Automating Visual Privacy Protection Using a Smart LED

Shilin Zhu*
University of California-San Diego
shz338@eng.ucsd.edu

Chi Zhang*
University of Wisconsin-Madison
czhang296@wisc.edu

Xinyu Zhang
University of California-San Diego
xiz368@eng.ucsd.edu

ABSTRACT

The ubiquity of mobile camera devices has been triggering an outcry of privacy concerns, whereas privacy protection still relies on the cooperation of the photographer or camera hardware, which can hardly be guaranteed in practice. In this paper, we introduce LiShield, which automatically protects a physical scene against photographing, by illuminating it with smart LEDs flickering in specialized waveforms. We use a model-driven approach to optimize the waveform, so as to ensure protection against the (uncontrollable) cameras and potential image-processing based attacks. We have also designed mechanisms to unblock authorized cameras and enable graceful degradation under strong ambient light interference. Our prototype implementation and experiments show that LiShield can effectively destroy unauthorized capturing while maintaining robustness against potential attacks.

CCS CONCEPTS

• **Computer systems organization → Special purpose systems**; • **Security and privacy → Security services**; *Systems security*; • **Human-centered computing → Ubiquitous and mobile devices**; • **Computing methodologies → Computer vision**;

KEYWORDS

Privacy Protection, Visible Light, Camera, Computer Vision

1 INTRODUCTION

Cameras are now pervasive on consumer mobile devices, such as smartphones, tablets, drones, smart glasses, first-person recorders [53], *etc.* The ubiquity of these cameras, paired with pervasive wireless access, is creating a new wave of visual sensing applications, *e.g.*, autonomous photographer [54], quantified-self (life-logging) [24, 95], photo-sharing social networks, physical-analytics in retail stores [64], and augmented reality applications that navigate users across unknown environment [55, 96]. Zooming in the photo-sharing application alone, statistics report that 350 million photos/videos are uploaded to Facebook every day, majority of which are from mobile users [76]. Many of these applications automatically upload batches of images/videos online, with a simple one-time permission from the user. While these technologies bring significant convenience to individuals, they also trigger an outcry of privacy concerns.

Privacy is ultimately a subjective matter, and often varies with context. Yet many of the privacy-sensitive scenes occur in indoor environment, and are bound to specific locations. For example, recent user studies [17] showed that people's acceptability of being recorded by augmented reality glasses has a strong correlation with location. User studies of life-logging cameras [37] also indicate that 70.2% of the cases when the user disables capturing is associated with specific locations. In numerous real-world scenarios, cameras are forbidden, *e.g.*, concerts, theaters, museums, trade shows, hospitals [34], dressing rooms and exam rooms [57], manufacturing plants [6], *etc.* However, *visual privacy protection in such passive physical spaces still heavily relies on rudimentary approaches* like warning signs and human monitors, and there is no way to automatically enforce the requirements. In personal visual sensing applications like life-logging, even if a user were to disable the camera in private space (*e.g.*, bedroom and washroom), malware could perform remote reconnaissance and targeted visual theft by hijacking the victim's camera [81, 98].

In this paper, we propose LiShield, a system that deters photographing of sensitive indoor physical space, and automatically enforces location-bound visual privacy protection. LiShield protects the physical scenes against undesired recording without requiring user intervention, and without disrupting the human visual perception. Our key idea is to illuminate the environment using smart LEDs, which are intensity-modulated following specialized waveforms. We design the waveform in such a way that its modulation pattern is imperceptible by human eyes, but can interfere with the image sensors on mobile camera devices.

More specifically, our basic waveform follows an ON-OFF modulation, which causes the reflection intensity of the scene to "flicker" at high frequency. Digital cameras commonly adopt rolling-shutter image sensors, which sample the scene row by row during capturing. Consequently, LiShield will impose a striping effect on the captured image, as long as its flickering frequency exceeds the camera frame rate. To protect against a wide range of camera settings, we build a numerical model to explore the relation between the image quality degradation and the (uncontrollable) camera configurations (*e.g.*, exposure time). Accordingly, we derive common guidelines to maximize the effectiveness through waveform parameter configurations (*e.g.*, frequency, peak intensity, duty cycle). To further enhance the protection, we take two measures: *(i.)* scramble the color patterns, taking advantage of the array of multi-channel RGB chips commonly available on commercial smart LEDs; *(ii.)* randomize the waveform frequency to counteract exposure time manipulation that may circumvent the striping effect, while ensuring no low-frequency components are generated that affect human perception.

*Co-primary authors.

MobiCom '17, October 16-20, 2017, Snowbird, UT, USA

ACM ISBN 978-1-4503-4916-1/17/10…$15.00
https://doi.org/10.1145/3117811.3117820

In addition, LiShield can tailor the waveform for two special use cases: *(i.)* allowing an authorized camera, which shares secret configuration information with the LED, to recover the image or video frames it captures. *(ii.)* when strong ambient light interferes with the smart LED, LiShield cannot ensure full protection, but it can still emit structured light which embeds invisible "barcode" into the physical environment. The embedded information can convey a "no distribution" message, allowing online servers (*e.g.*, from Facebook and Instagram) to block and prevent the image from being distributed.

We have implemented LiShield based on a customized smart LED, which allows reconfiguration of intensity modulation waveforms on each color channel. Our experiments on real world scenes demonstrate that LiShield can corrupt the camera capturing to an illegible level, in terms of the image brightness, structure, and color. The impact is resilient against possible post-processing attacks, such as multi-frame combining and denoising. On the other hand, it enables authorized cameras to recover the image perfectly, as if no modulation is present. Even under strong sunlight/flashlight interferences, LiShield can still sneak barcode into the physical scenes which can be decoded with around 95% accuracy.

Preventing all privacy leaks, particularly those by determined attackers with professional global-shutter cameras, is likely impossible. Instead, LiShield aims for preventing ad-hoc capturing from benign camera-phone holders, by simply installing customized smart LEDs to fully cover the target environment. Our main contributions can be summarized as follows:

(i.) Proposing a new concept of automating privacy protection against cameras by modulating an LED's waveforms, and deriving general guidelines for optimizing the waveforms against possible camera settings and image recovery.

(ii.) Designing mechanisms to authorize desired capturing, and to embed protection information into the scene under strong ambient light interference.

(iii.) Verifying the system through a full-fledged testbed implementation and experiments in real environments.

2 ADVERSARY MODEL AND PROTECTION GOALS

LiShield's end goal is to prevent camera recording in protected indoor physical areas, without affecting normal human perception. The scene can be static or dynamic. In either case, we assume one or multiple LiShield-enabled smart LEDs can cover the whole area, while providing illumination similar to normal office lighting without human-perceptible flickering. Whereas conventional lighting and sunlight may co-exist with LiShield's smart LEDs (as to be verified in our experiments), covering the entire target scene with LiShield will ensure the strongest protection.

Now consider an unauthorized user (attacker) who wants to take pictures or videos within the protected space, with cameras and flashes embedded in smartphones, but no professional equipment such as global shutter cameras, filters or tripods. The attacker has full control over the camera parameters (*e.g.*, exposure time, capturing time, white-balancing), and can run any post processing on the captured images. Nonetheless, with LiShield's protection, the image frames are corrupted, so that major fraction of each frame is either blank or overexposed while colors are distorted (Sec. 3),

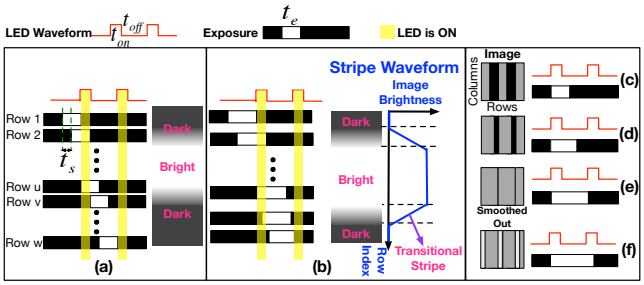

Figure 1: (a)-(b) Bright, dark and transitional stripes and their width changing with exposure time; (c)-(f) Stripe pattern of image changes under different exposure times.

which deters image viewing/sharing. In addition, LiShield should maintain its protection while allowing authorized users to capture the same scene simultaneously without distortion (Sec. 4). In case strong ambient interference may degrade LiShield's protection, LiShield embeds barcodes in images/videos captured by the attacker to convey privacy policies and ensures they are detectable even after common post-processing (Sec. 5).

3 PHYSICAL SCENE DISRUPTION

3.1 A Primer on Camera Image Disruption in LiShield

Cameras and human eyes perceive scenes in fundamentally different ways. Human eyes process continuous vision by accumulating light signals, while cameras slice and sample the scene at discrete intervals. Consequently, human eyes are not sensitive to high frequency flickers beyond around 80 Hz either in brightness or chromaticity [4, 41, 87, 104], while cameras can easily pick up flicker above a few kHz [44, 102]. Equally importantly, human eyes perceive brightness in a non-linear fashion [77], which gives them huge dynamic range, while cameras easily suffer from overexposure and underexposure when signals with disparate intensities mix in the same scene [66].

Unlike professional or industrial cameras which may have global shutters that mimic human eyes to some degree, nearly all consumer digital cameras, pinhole cameras, and smartphones use the rolling shutter sampling mechanism [49, 62], which is the main contributor to their high-frequency sensitivity. When capturing an image frame, a *rolling shutter camera exposes each row sequentially.*

LiShield harnesses the disparity between cameras and eyes to disrupt the camera imaging without affecting human vision. It modulates a smart LED to generate high-frequency flickering patterns. The reflection intensity (or brightness) of target scene also flickers following the same pattern as the LED's illumination, albeit at reduced intensity due to reflection loss. LiShield uses the On-Off Keying (OOK) as the basic modulation waveform (Fig. 1), which does not require complicated analog front-ends and is widely supported by smart LEDs [25, 26]. Due to rolling-shutter sampling, the rows of pixels that are fully exposed in the ON period will be bright, and rows in the OFF period become dark, thus causing striped patterns on the captured image (Fig. 1(a)(b)). Partially exposed rows experience moderate brightness. Meanwhile, human eyes can only perceive the smooth averaged intensity, as long as the OOK frequency goes beyond 80 Hz [4, 41, 87, 104].

In addition, commercial LED fixtures often comprise multiple LED bulbs/chips, and sometimes separate RGB channels to allow color adjustments [60]. LiShield can turn different numbers of LED bulb/chip on to generate different intensities, and control the RGB channels of the LEDs to vary the color. Therefore, LiShield's flickering waveform is staircase-shaped on-off patterns, running independently in 3 color channels. In what follows, we will show how such flickering corrupts the spatial patterns captured by a camera.

3.2 Maximizing Image Quality Degradation

LiShield aims to *minimize the image capturing quality by optimizing the LED waveform, characterized by modulation frequency, intensity, and duty cycle.* To explore the optimization space and to provide guidelines for designing the basic waveform, we derive a model to predict the image quality as a function of the LiShield's waveform and attacker's camera parameters. For simplicity, we start with monochrome LED (equivalent to one with a single color channel) that illuminates the space homogeneously. We denote P as the reference image taken under a non-flickering LED, and Q as the one taken under LiShield's LED with the same average brightness. We assume each image has m rows and n columns, and the light energy received by each pixel is denoted by $P(i,j)$ and $Q(i,j)$, respectively. Our model focuses on two widely adopted image quality metrics: *PSNR*, which quantifies the disruption on individual pixel intensity levels; and *SSIM* [91], which measures the structural distortion to the image (*i.e.*, deformation effects such as stretching, banding and twisting). In general, the minimum PSNR and SSIM corresponding to acceptable viewing quality are in the range of 25∼30 and 0.8∼0.9, respectively [3, 5, 12, 29].

3.2.1 Decomposing the Image. To compute the image quality, we need to model the intensity and width of each stripe caused by LiShield. As illustrated in Fig. 1, we use $t_{\text{on}}, t_{\text{off}}, I_p$ to denote the on/off duration and peak intensity of the flickering light source, and t_e, t_s are the exposure time (controllable by software) and sampling interval (fixed in hardware) of the rolling shutter camera. For convenience, denote the period of the light source as $t_l = t_{\text{on}} + t_{\text{off}}$, and duty cycle as $D_c = t_{\text{on}}/t_l$. For pixel j in row i which starts exposure at time t_i, its light accumulation would be:

$$Q(i,j) = \alpha_{i,j} \int_{t_i}^{t_i+t_e} \pi_l(\tau) d\tau \tag{1}$$

where $\alpha_{i,j}$ is the aggregated path-loss for pixel (i,j), including attenuation and reflection on the photographed object, and $\pi_l(\tau)$ represents the illumination waveform of the LED:

$$\pi_l(\tau) = \begin{cases} I_p, & 0 < \tau \bmod t_l \leqslant t_{\text{on}} \\ 0, & t_{\text{on}} < \tau \bmod t_l \leqslant t_l \end{cases} \tag{2}$$

When the camera's exposure time is equal or shorter than the LED's OFF period ($t_e \leqslant t_{\text{off}}$), the image will contain rows that are completely dark (Fig. 1(c)). On the other hand, when $t_e > t_l$, one row-exposure period of the camera will overlap multiple ON periods of the LED, accumulating higher intensity (Fig. 1(f)). The special case happens when $t_e = t_l$ where the integration of LED waveform and exposure has fixed value, which eventually smooths out dark stripes (Fig. 1(e)). Without loss of generality, assume the exposure starts right at the beginning of the ON period. Let $N = \lfloor t_e/t_l \rfloor$ which

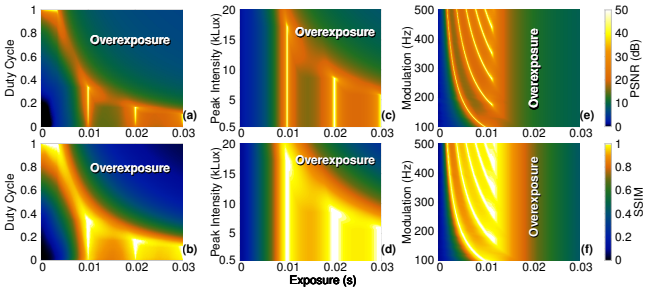

Figure 2: PSNR and SSIM with respect to exposure time, LED intensity, duty cycle, and modulation frequency.

is the number of whole flicker cycles covered by exposure time, and $t_{\text{rem}} = (t_e \bmod t_l)$ which is the remaining duration after multiple whole cycles, the light accumulation of the brightest rows Q_B is:

$$Q_B(i,j) = \begin{cases} \alpha_{i,j} I_p (N t_{\text{on}} + t_{\text{rem}}), & 0 < t_{\text{rem}} \leqslant t_{\text{on}} \\ \alpha_{i,j} I_p (N+1) t_{\text{on}}, & t_{\text{on}} < t_{\text{rem}} \leqslant t_l \end{cases} \tag{3}$$

Since the brightest rows appear when the exposure captures most ON periods possible (*e.g.*, row 2 to row u in Fig. 1 (a)), and rolling shutter effect converts temporal variation into pixels with sampling interval t_s, the width of Q_B is:

$$W_B = |t_{\text{rem}} - t_{\text{on}}|/t_s \tag{4}$$

Likewise, when the exposure captures least ON periods possible (*e.g.*, from row v to row w in Fig. 1 (a)), we get the darkest rows with light accumulation Q_D:

$$Q_D(i,j) = \begin{cases} \alpha_{i,j} I_p N t_{\text{on}}, & 0 < t_{\text{rem}} \leqslant t_{\text{off}} \\ \alpha_{i,j} I_p (N t_{\text{on}} + t_{\text{rem}} - t_{\text{off}}), & t_{\text{off}} < t_{\text{rem}} \leqslant t_l \end{cases} \tag{5}$$

and the width of Q_D is:

$$W_D = |t_{\text{rem}} - t_{\text{off}}|/t_s \tag{6}$$

We refer to a collection of consecutive brightest rows as "bright stripe" and consecutive dark rows as "dark stripe", as shown in Fig. 1(b). In addition, there exist intermediate rows containing linear intensity transition between dark and bright, referred to as "transitional stripe".

Meanwhile, if the LED were not flickering and provided the same average brightness, the pixel intensity would be:

$$P(i,j) = \alpha_{i,j} I_p \cdot D_c \cdot t_e \tag{7}$$

Since $D_c \cdot t_e$ remains constant within each frame, *the image captured under LiShield is equivalent to the original image multiplied by a piecewise function* (cf. Eqs. (3) and (5)).

Other common camera parameters (*i.e.*, ISO, white balance, and resolution) do not affect the structure of the stripe pattern, since they are unrelated to rolling shutters and they only affect the average pixel intensity. By default, we assume the attacker sets the ISO to its minimum (usually 100) to maximally suppress noise.

3.2.2 Optimizing the LED Waveform. Since the stripe pattern follows a piecewise function, a closed form expression of PSNR and SSIM becomes infeasible. We thus use numerical simulation to evaluate the impact of LiShield, based on the above model. We generate the piecewise function with $Q_B(i,j), W_B, Q_D(i,j), W_D$ and multiply it on a reference image to obtain the disrupted image Q just like the process inside real cameras. We use the well-known Lena image as a reference, and stitch the original 512×512 version into a 3264×2448 (8-mega-pixel) image, assuming $t_s = 1/75000$ s, which

matches the capability of a Nexus 5 camera. The quality metrics are calculated between the reference image P and LiShield-corrupted image Q, which are set to the same average intensity by scaling pixel values in Q. Note that if P and Q are both overexposed into the same white image, PSNR = ∞ and SSIM = 1 can no longer reflect image quality. Thus, we make P's pixel intensity range infinite, which allows quantifying quality loss caused by overexposure.

By default, we use OOK waveform with frequency $f = 100$ Hz, peak intensity $I_p = 10$ kLx and duty cycle $D_c = 0.5$. We vary one parameter while keeping others to the defaults. Note that the typical light intensity is ~ 700 Lx in office environments (considering energy efficiency), $\sim 5,000$ Lx for overcast sky and $\sim 100,000$ Lx for sunny days [33]. Our numerical results (Fig. 2 show a few general trends, which lead to the following design choices for LiShield.

(i) A single frequency cannot ensure robust protection. Fig. 2(e) and (f) show that for a given waveform frequency f, there exist several exposure time settings that lead to high-quality images. This is because when $t_e \approx N t_l$, the stripes become smoothed out (Fig. 1(e)). Although the waveform parameters are unknown to the attacker, a determined attacker may launch a brute-force search for the t_e that satisfies this condition, thus circumventing the protection. To counteract such attackers, *LiShield includes a countermeasure called frequency randomization*, which we discuss in Sec. 3.3.1.

(ii) LiShield must prevent attackers from using long exposures. The image quality increases with exposure time t_e, until overexposure happens (Fig. 2(a) and (b)), because longer exposure leads to more waveform cycles being included as a constant base in the brightness of the stripes (larger N in Eqs. (3) and (5)), making the contrast of stripes Q_B/Q_D lower and weakening the quality degradation. Since overexposure limits the maximum exposure time, *LiShield should leverage overexposure to limit attacker's exposure time.*

(iii) LiShield should keep a high peak intensity to expand the overexposure zone. We observe that when t_e falls below a threshold ($\approx$1/100 s in Fig. 2(c) and (d)), the image is always corrupted due to the dominance of dark stripes (Fig. 1 (c)). On the other hand, when t_e goes beyond a threshold, the image always suffers from overexposure. A larger I_p leads to a smaller overexposure threshold for t_e, which limits the attacker's ability to tune t_e to improve image quality. When $I_p \geq 10$ kLx (Fig. 2(c)), there almost exists only a single t_e setting ($t_e \approx 1/100$ s) that can avoid overexposure and dark stripes simultaneously. But even this setting fails under LiShield's frequency randomization mechanism (Sec. 3.3.1). With power efficiency and eye health in mind (Sec. 8), *LiShield sets I_p to 20 kLx by default.*

(iv) Duty cycle should be kept at a moderate level. Without overexposure, a lower D_c yields lower PSNR and SSIM (Fig. 2(a) and (b)), as it widens the dark stripes (Eq. (6)). On the other hand, a larger D_c means more light accumulation, resulting in overexposure across a wider range of t_e settings. Since higher I_p has the same impact given a fixed D_c, we design the LED waveform *to have maximum peak intensity with moderate duty cycle, empirically set to $D_c = 0.5$.*

The above conclusions hold for all scenes since the trend of quality does not vary with scenes (Sec. 7). Optimal parameters may vary slightly across different scenes (*e.g.* different reflectivity), and can be easily obtained by taking one photo of the scene and running the aforementioned simulation.

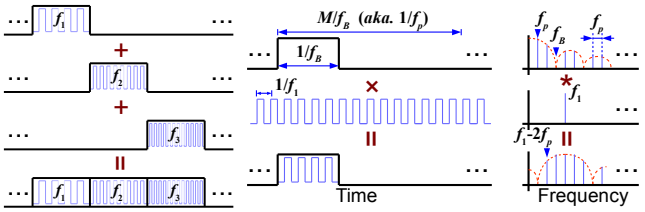

Figure 3: Decomposition of frequency randomization waveform and modulation generating side lobes.

3.2.3 Adding Color to the Model. Occasionally sensitive information is in the color channel of images, which requires LiShield to distort color for protection. LiShield extends to multi-channel setup by generating waveforms for each of the RGB channels independently, turning white stripes in the previous model into colored ones. To compensate the intensity loss compared with the white stripes, we need to make the new peak intensity $I'_p = 3I_p$, assuming R, G and B appear with equal probability.

3.3 Circumventing Potential Attacks

Based on the foregoing analysis, we identify the following potential holes that can be exploited by attackers to overcome the striping effect. *(i) Manual exposure attack.* If an attacker can configure the t_e to satisfy $t_e \approx N t_l$, it can guarantee every row receives almost the same illumination, thus eliminating the stripes during a capture (Fig. 1(e)). In practice, t_l is unknown to the attacker, but it can try to capture images with different t_e, until seeing a version without obvious stripes[1]. *(ii) Multi-frame attack.* When the scene is static, an attacker may also combine multiple frames (taking a video and playback) to mitigate the stripes with statistical clues, *e.g.* by averaging or combining rows with maximum intensities from multiple frames. Note that the attacker must keep the camera highly stable, otherwise even pixel-level shift will cause severe deformation when combining multiple frames. *(iii) Post-processing attack.* Common post-processing techniques (*e.g.*, denoising and de-banding) might be used to repair the corrupted images.

In what follows, we introduce countermeasures to the first two attacks. In Sec. 7.4, we will verify that LiShield's distortion does not fit empirical noise or banding models, so the common post-processing schemes become ineffective.

3.3.1 Frequency Scrambling. To thwart the manual exposure attack, we design a *frequency scrambling*[2] mechanism, which packs multiple waveforms with randomly selected frequencies within each image frame duration. Since the camera exposure time t_e is always fixed within each frame, no single t_e can circumvent all the frequency components.

However, we cannot choose and switch the flickering frequencies in an arbitrary manner, for three reasons. *(i)* Multiple frequency values that share a common divisor can satisfy $t_e = N t_l$ under the same t_e (recall N can be an arbitrary integer). We need to ensure the common divisor is small enough (*i.e.*, least common multiplier of t_l large enough), so that overexposure occurs even for the smallest N. *(ii)* Frequencies should be kept low to maximize image corruption,

[1] Digital camera's exposure time cannot be set arbitrarily due to hardware limitation. On Nexus 5, the granularity is around 13μs, *e.g.*, the actual exposure time is 1/1950 when the attacker needs 1/2000. Note that exposure time must be fixed within each frame.

[2] Scrambling and randomization are exchangeable in this paper.

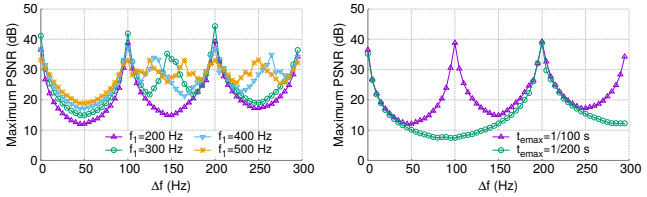

Figure 4: Worst-case PSNR with different frequency increment Δf under different (a) f_1 (b) t_{emax}.

as evident in Fig. 2(e) and (f), since camera's analog gain decreases at high frequencies [102]. *(iii)* Switching between different frequencies may create an additional level of modulation, which will spread the spectrum and generate unexpected low frequency components that become perceivable by eyes.

To explore the design space under these constraints, suppose we switch among M frequencies $f_1, f_2, \ldots, f_M$ (in ascending order) at a switching rate f_B. The whole pattern thus repeats itself at rate $f_p = f_B/M$. To pack at least M different frequencies in an image frame, we need $f_B > (M-1)f_r$, or preferably, $f_B > Mf_r$, where f_r is the frame rate, typically around 30 Hz (fps). Note that the switching rate cannot be higher than the lowest scrambling frequency, *i.e.* $f_B \leq f_1$, otherwise the waveforms of f_1 will be truncated. To maximize image corruption, we choose the smallest value for f_1 (*i.e.*, $f_1 = f_B$), and empirically set $f_n = f_B + (n-1)\Delta f, n \in 2, 3, \ldots, M$, where Δf is frequency increment, set to $\Delta f \neq f_B$ to lower the common divisor frequency.

The frequency scrambling can be considered as an M-FSK modulation: essentially, we multiply the waveform corresponding to each frequency with a rectangular wave of frequency f_p and duty cycle $1/M$, which convolves harmonics of the pattern repetition frequency f_p to the spectrum, creating side lobes around each scrambling frequency, spacing f_p apart, as shown in Fig. 3. These side lobes might appear at low-frequency region and become perceptible by human eyes.

To tackle this challenge, note that for waveforms with frequencies $f_2, f_3, \ldots$, their side lobes are dampened more at lower frequencies compared with f_1, so we only need to focus on f_1. The side lobes of f_1 are located at $f_1 + kf_p$, where k is an integer. For the side lobe with the lowest frequency, $k = \lfloor f_1/f_p \rfloor$. Since we selected $f_1 = f_B = Mf_p$, the lowest non-DC side lobe is at $f_p = f_B/M$. Therefore, to ensure no side lobe exists below the perceivable threshold $f_{\text{th}} \approx 80$ Hz, we need a small M and large f_B, and hence higher flickering frequency components f_n. Yet increasing the flickering frequencies may weaken LiShield's protection. Fortunately, since LiShield does not require large M (which leads to high f_M) to circumvent the manual exposure attack, the degradation should be tolerable.

To find the optimal Δf and showcase the effectiveness of the frequency scrambling, we choose the case $M = 2$ and $f_1 = f_B$ under 20 kLx peak intensity (to be consistent with our testbed setup in Sec. 6). We then repeat the numerical simulation (Sec. 3.2) to evaluate the attacker's maximum image quality. Fig. 4(a) shows that the quality has two peaks at $\Delta f = 0$ and 100 Hz, as well as a valley at $\Delta f = 50$ Hz. Note that the positions of these peaks/valleys are independent of f_1 and M, because quality always reaches the maximum at the longest t_e before overexposure happens (denoted as t_{emax} in Fig. 4(b)). Thus, we set $\Delta f = (1/2)/t_{emax} = 50$ Hz to maximize image disruption. The optimal Δf for other peak intensity settings can be obtained following a similar procedure. Fig. 4 also shows that, once set to the optimal Δf, *frequency randomization can significantly improve LiShield's robustness against manual exposure attacks.* Sec. 7 will show more evidence through testbed experiments.

3.3.2 Illumination Intensity Randomization. If attackers repetitively capture a static scene for a sufficiently long duration, they may eventually find at least one clean version for each row across all frames, thus *recovering* the image. LiShield does not guarantee complete protection against such brute-force attacks. However, it can increase the number of frames needed for image recovery, so that the attack becomes infeasible unless the camera can stay perfectly still over a long period of time, during which the attackers may have already been discovered by the owners of the physical space. LiShield achieves the goal by employing illumination intensity randomization, where it randomly switches the magnitude of each ON period across multiple predefined levels, which extends the attacker's search space. We note that the intensity randomization adds another level of modulation, but similar analysis in Sec. 3.3.1 still applies and can ensure imperceptible operation.

To understand the effectiveness of this scheme, we build a statistical model to estimate the number of frames needed to perfectly recover the image, as if LiShield did not function at all. Suppose the LED waveform has K intensity levels, and the camera has m rows. For simplicity, we assume the intensity levels of each row become uncorrelated after the randomization. Then the probability that one row gets any illumination is $p = t_{\text{on}}/(t_{\text{on}} + t_{\text{off}}) = D_c$. Observe that on average same intensities would reappear approximately every K frames, the possibility of combining L frames to fully recover an image of the static scene is thus:

$$P_{\text{rec}} = \begin{cases} \left[1 - (1 - D_c)^{L/K}\right]^{m} & \text{(monochrome)} \\ \left[1 - (1 - D_c)^{L/K}\right]^{3m} & \text{(RGB)} \end{cases} \tag{8}$$

Therefore, achieving a given level of P_{rec} becomes increasingly difficult as D_c and K increases, and for higher camera resolution (larger m). For example, to have $P_{\text{rec}} = 90\%$ for $D_c = 0.5$ and $m = 2448$ for 8-mega-pixel cameras, the attacker needs $L = 300$ frames under $K = 10$, and ~ 3000 frames under $K = 100$. For lower duty cycles, recovery becomes even more challenging (*e.g.*, ~ 7000 frames are needed for $D_c = 0.2$, $K = 100$). As we will show later (Sec. 7), in practice, the attackers cannot keep cameras completely still and all frames aligned at pixel level, even with a tripod and across a short duration. So Eq. (8) gives the best performance for such attacks. Note that *if the target scene is mobile, then the multi-frame attack becomes impossible, as long as the scene has certain variation across K frames.* The effectiveness of intensity randomization will be further justified in our testbed experiments (Sec. 7).

4 SCENE RECOVERY WITH AUTHORIZED CAMERAS

To allow authorized users to capture the scene while maintaining protection against unauthorized attackers, we need to impose additional constraints on the LED waveform. LiShield's solution leverages a secure side channel (*e.g.* visible light communication [16] or Wi-Fi) between authorized users and the smart LED, which

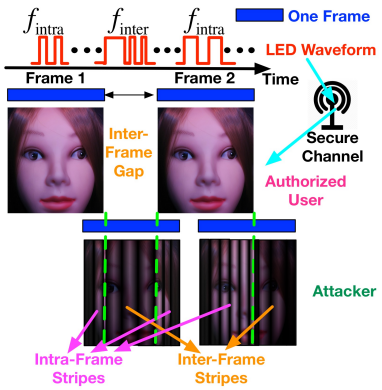

Figure 5: Enabling authorized users to capture dynamic scenes while corrupting unauthorized users.

conveys secret information such as frame timing and waveform parameters[3].

A naive solution is to stop flickering when authorized users are recording. However, since attackers may be co-located with the authorized users, this enables them to capture one or more frames that have part of the clean scene, which compromises privacy and security. To counteract such cases, we design special waveforms for the LED to minimize the flicker-free duration.

4.1 Authorized Video Recording

To authorize a camera to capture a dynamic scene, each individual frame within the video must be recoverable. To achieve this, the authorized camera needs to convey its exposure time setting t_e^u to the smart LED via the secure side channel, and synchronize its clock (for controlling capturing time) with the smart LED's clock (for controlling the waveform), so the smart LED can send recoverable waveforms precisely during the capture of the authorized camera. State-of-the-art time synchronization mechanisms through visible light [48] or wireless side-channels [23, 69, 74] can already achieve μs of accuracy, sufficient to synchronize the LiShield smart LED with camera at a resolution that is finer than the rolling shutter period (typically tens of μs).

Recall that the camera can evade the striping effects if $t_e = Nt_l$ (phase does not matter, see Sec. 3.3). So to authorize the user with exposure t_e^u, LiShield simply needs to set its flickering frequency $f_a = 1/t_l = N/t_e^u$ $(N = 1, 2, \ldots)$ and maintain its peak intensity within each frame. In addition, the t_e^u and corresponding flickering frequency f_a can be varied on a frame by frame basis, making it impossible for an attacker to resolve the correct exposure time by trial-and-error (Sec. 3.3).

Meanwhile, when the authorized camera is not recording at its maximum possible rate (*e.g.*, a 30 fps camera recording at 25 fps), there will be an interval (*i.e.*, inter-frame gap) where the camera pauses capturing. LiShield packs random flickering frequencies other than f_a into the inter-frame gap, so as to achieve the same scrambling effect as described in Sec. 3.3.1, without compromising the authorized capturing. Fig. 5 depicts one example, where f_{intra} and f_{inter} denote intra-frame and inter-frame frequencies, respectively.

[3] Such information can be protected by existing encryption algorithms and systems, which are already mature and thus beyond the scope of this paper.

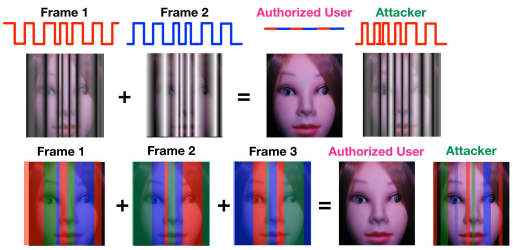

Figure 6: The impact of multi-frame recovery on authorized user and attacker, respectively.

4.2 Static Scene Recovery

When the target scene is static, the authorized user may capture a few complementary frames at a specific time to recover the scene as depicted in Fig. 6, where frequency and intensity randomization (Sec. 3.3) are employed in each frame to ensure robustness. While it does require recording a very short video, the process is extremely short (200ms at most) and barely noticeable to the authorized user. Meanwhile, an out-of-sync attacker will still receive corrupted images that cannot reconstruct the original scene even after combined.

Suppose a static scene is to be recovered using L_f frames, referred to as *critical frames*. To prevent attackers from launching the multi-frame attack, the timing of the critical frames is negotiated only between the smart LED and the authorized user through the secure side channel. These L_f frames together must contain the information of the entire scene, *i.e.* they must be complementary, as shown in Fig. 6. Meanwhile, all other frames will follow the normal flickering pattern as discussed in Sec. 3. Since the attackers cannot identify nor predict the timing of the critical frames , the best they can do is to launch the brute-force multi-frame attack, which has been discussed in Sec. 3.3.2.

5 AUTOMATIC PHYSICAL WATERMARKING FOR PRIVACY ENFORCEMENT

High-intensity ambient light sources (*e.g.* sunlight, legacy lighting, flash lights) can create strong interference to LiShield's illumination waveform, degrading the contrast by adding a constant intensity to both the bright and dark stripes, which may weaken LiShield's protection. In such scenarios, LiShield degrades itself to a *barcode mode*, where it embeds barcode in the physical scene to convey privacy policies. The barcode forms low-contrast stripes, which may not fully corrupt the images of the scene, but can still be detected by online photo-distributing hubs (*e.g.*, social website servers) who automatically enforce the policies, without cooperation of the uploader or evidence visible by naked eye. LiShield forms the watermark with just a single light fixture, instead of active displays (*e.g.*, projectors) that are required by conventional systems. The key challenge here is: how should LiShield encode the information, so that it can be robustly conveyed to the policy enforcers, despite the (uncontrollable) attacker camera settings? We now describe how LiShield's barcode design meets the challenge.

Embedding. LiShield's barcode packs multiple frequencies in every image (or in every frame of a video) following Sec. 3.3.1, but aims to map the *ratios between frequencies* into digital information. Suppose LiShield embeds two waveforms with frequencies F_0 and F_1, it chooses the two frequency components such that F_1/F_0 equals

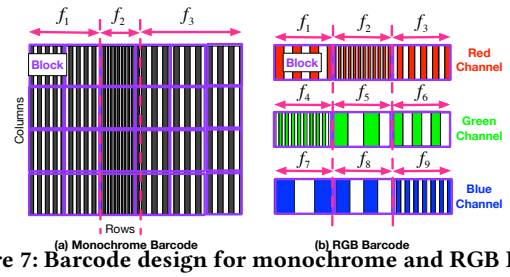

Figure 7: Barcode design for monochrome and RGB LED.

to a value R_p well known to the policy enforcers. In other words, the presence of R_p conveys "no distribution/sharing allowed". This encoding mechanism is robust against camera settings [4].

Since physical scenes usually comprise a mix of spatial frequencies, and spectral power rolls off in higher spatial frequencies thanks to camera lenses' limited bandwidth [31] while temporal frequencies are unaffected, LiShield's barcode uses frequencies that are much higher than the natural frequencies (> 400Hz) in the scene to avoid interference. It is worth noting that since the rolling-shutter sampling rate of all cameras falls in a range (30 kHz to slightly over 100 kHz [102]), LiShield limits its highest flickering frequency to 15 kHz, which respects the Nyquist sampling theorem so that the barcode can eventually be recovered without any aliasing effect.

To further improve robustness, LiShield leverages redundancy. It embeds multiple pairs of frequency components to make multiple values of R_p. In this way, LiShield can pack different R_p either at different rows of the image or in different color channels, further mitigating interference caused by intrinsic spatial patterns within the scene. Fig. 7 illustrates an example of monochrome ($C_3^2 = 3$ R_p values) and RGB LEDs ($C_{3\times3}^2 = 36$ R_p values). Note that the same mechanism can be used to increase the amount of information in the barcode, but this is beyond the scope of the present work.

Detection. Since the barcode contains M frequencies, *i.e.* $f_n = f_B + (n-1)\Delta f, n \in 2, 3, \ldots, M$ (Sec. 3.3.1), there are $M_R = C_M^2$ possible frequency ratio values across the image for monochrome barcode ($M_R = C_{M\times3}^2$ for RGB barcode). Δf must be set large enough to avoid confusion ($\Delta f = 200$ Hz in experiments). The barcode decoder, running on the policy enforcer, recognizes the image as protected if there are at least M_b values that roughly match the known ratio R_p, *i.e.*, when the value falls within T_b of R_p. We empirically set $M_b = \lceil \gamma M_R + M_{\text{att}} \rceil$ where M_{att} is number of R_p removed by manual exposure attack (Sec. 3.3). γ and T_b are determined by bounding the false positive rate following an empirical procedure (to be discussed in Sec. 7.3).

To detect the frequency ratios, LiShield first partitions the image into $n_{b_r} \times n_{b_c}$ blocks, across both rows and columns, either within the monochrome channel or among all 3 RGB channels. Dividing image by columns provides LiShield multiple copies of the same frequency block, in case some of them are interfered by spatial patterns in the scene. For example, in Fig. 7, $n_{b_r} = 6$ and $n_{b_c} = 4$. For each block, LiShield averages the intensity of each row to get a one-dimension time series s_r of length $L_f = m/n_{b_r}$, given total m rows on the image. LiShield then runs FFT over each series to extract the M_p strongest frequencies. Note that LiShield's detector

[4] Although width of stripes is affected by sampling interval t_s and exposure time t_e (Fig. 1(a) and (b)), ratio of stripe widths resulted from two frequencies (which equals to R_p) remains constant.

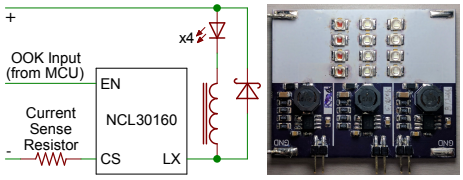

Figure 8: Simplified circuit diagram and photo for the smart LED module.

allows more than one frequencies to appear in one block. Finally, LiShield combines all unique frequencies extracted from each block and computes all frequency ratios (within and across color channels in the case of RGB barcode). Algorithm 1 describes the procedure of barcode detection.

Algorithm 1: Barcode Detection

Input: image I ($m \times n$), n_{b_r}, n_{b_c}, T_b, S_b, M_b, $m_b = 0$, $F = \emptyset$, $D_p = \emptyset$, $B = \emptyset$, $f_s = 30$ kHz
Output: whether I is protected
crop I to $n_{b_r} \times n_{b_c}$-size blocks, store in set B;
for $b \in B$ **do**
  $s_r \leftarrow n_{b_r} \times 1 \leftarrow \text{mean}(n_{b_r} \times n_{b_c})$;
  $F_D \leftarrow \text{detrend}(FFT(s_r, f_s))$;
  pick M_p maximum peaks $F_p \in F_D$,
  400 Hz$\leqslant F_p \leqslant$15 kHz;
  $F \leftarrow F \cup F_p$;
end
$D_p \leftarrow D_p \cup (f_i/f_j), \forall f_i, f_j \in F$;
for $d_p \in D_p$ **do**
  if $d_p \in [R_p - T_b, R_p + T_b], \forall R_p \in S_b$ **then**
    $m_b \leftarrow m_b + 1$;
  end
end
if $m_b \geq M_b$ **then**
  I is protected;
end

LiShield's redundancy in barcode ensures that the barcode cannot be completely destroyed, unless nearly all frequencies are distorted by processing the image, which will in turn cause strong deformation on the scene. We will verify the robustness of this scheme through testbed experiments (Sec. 7.4).

6 IMPLEMENTATION

Testbed setup. Fig. 9 shows our smart LED prototype, and the target scenes containing 5 capture-sensitive objects (document and painting are 2-D objects and others are all 3-D objects). We mount the LED inside a diffusive plastic cover similar to conventional ceiling light covers. We use a programmable motor [13] to hold the camera and control its distance/orientation, in order to create static or dynamic scene setup in a repeatable manner.

Smart LED modules. Commercial-of-the-shelf (COTS) household LED bulbs rely on integrated drivers to regulate LED's current [51, 88]. A dimming input is usually available on these drivers for controlling the current dynamically. We build our smart bulb based on the same topology as these COTS LED bulbs. For safety, we use 19V DC laptop power supplies instead of wall AC power, and NCL30160 [58] LED drivers which allow dimming at nearly 100

Figure 9: Experimental setup and multiple scenes we used.

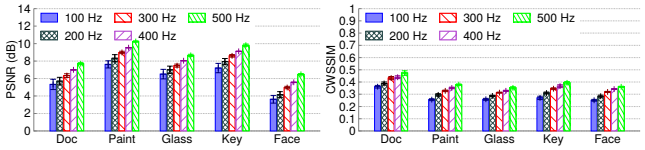

Figure 10: Impact of flickering frequency on quality.

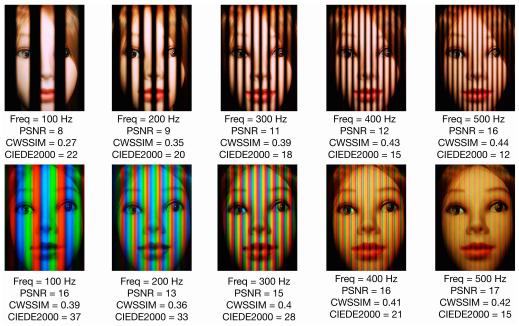

Figure 11: Image quality levels on a benchmark image.

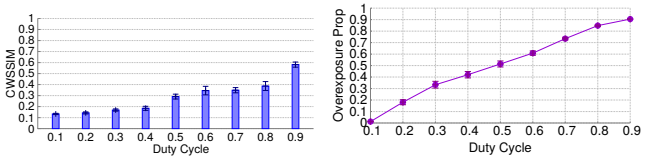

Figure 12: Impact of duty cycle on quality with auto-exposure.

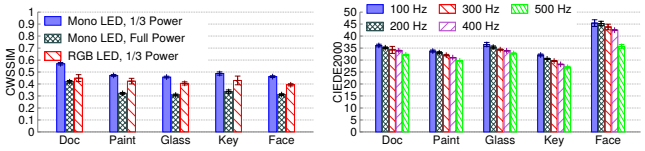

Figure 13: Impact of duty cycle on overexposure area with fix-exposure.

Figure 14: Impact of color on quality.

Figure 15: CIEDE2000 for color distortion.

kHz with arbitrary OOK waveform. The smart bulb has built-in independent RGB/white channels for controlling color/intensity. Each channel can be controlled by a separate waveform, with 4 LED chips in series, at driving current of 800 mA. In total, the 3 channels consume approximately 25 W peak power, close to common office LED troffer fixtures. However, since LiShield's OOK waveform has a duty cycle much lower than 1 (Sec. 3), the actual perceptible brightness is significantly lower. As a result, multiple LED modules can be used to improve light intensity. Fig. 8 depicts the circuit for each color channel and shows a photo of the whole module.

The dimming input signals of each channel are controlled by an STM32 [78] micro-controller unit (MCU), which generates the OOK waveform as specified by LiShield. For flexible reconfiguration, we generate digitized waveforms in MATLAB on a laptop or Android app on a smartphone instead, which are then passed to the MCU via USB.

Android app for normal, authorized and attacker's cameras. Unless otherwise noted, we use Nexus 5 [46] with stock ROM as our benchmark device. We assume that normal users use the stock camera app with default settings (including auto exposure), while a malicious attacker can manually tune the camera parameters (*e.g.*, using the Open Camera app [35]). By default, the camera ISO is set to the lowest value (100) since it is the most beneficial for attackers, as it allows longer exposure to smooth out the stripes without causing overexposure. To implement the authorization mechanism (Sec. 4), we develop a specialized app for the authorized smartphone, which uses Android's Camera2 API [32] to precisely control the exposure time, as well as communicating with the smart LED's MCU via USB. Since current Android camera APIs do not support precise frame timing, the app requests the smart LED to synchronize with the camera by altering its waveform.

Attacker's image processing. We have implemented the attacking algorithms in Sec. 3.3, which are specifically designed to combine/process the captured image, aiming to eliminate LiShield's stripe distortion. In addition, we implement classical image processing techniques, including denoising and debanding, which may be attempted by attackers. For denoising, we use the Haar-wavelet thresholding [14], non-local-means (NLmeans) [9] and BM3D [15], which are among the most popular algorithms [70]. For Haar-wavelet and NLmeans, we use the G'MIC [40] plugin of the GIMP [83] image processing program. For BM3D, we use a CUDA implementation [36] since it is significantly faster and practical than CPU-base implementations. As for debanding, we use the Banding Denoise [22] and Unstrip [8] in the G'MIC [40] plugin.

Metrics. Small displacement and vibration of the camera are inevitable in physical environment, which is known to affect the SSIM of captured images significantly [92]. Thus, we quantify the image quality degradation with the enhanced CW-SSIM [67], which is insensitive under such translations, but similar to SSIM otherwise. Since PSNR shows similar trends with SSIM, we omit it in the experiments except for a few cases. Besides, we employ the CIEDE2000 [50] to compute the degradation of the images' color quality when the RGB LED is used.

7 EXPERIMENTAL EVALUATION

7.1 Effectiveness of Physical Scene Disruption

Impact of flickering frequency. We first verify LiShield's basic protection scheme (Sec. 3) with 5 static scenes, monochrome LEDs, and OOK waveform without frequency randomization, while attacker's camera uses auto-exposure. Without LiShield, the measured

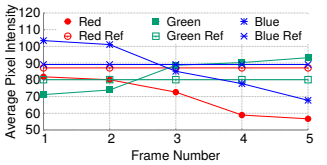

Figure 16: Impact on automatic white balance.

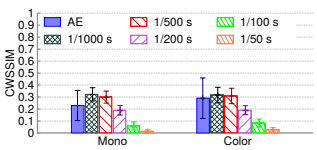

Figure 17: Impact on dynamic scene.

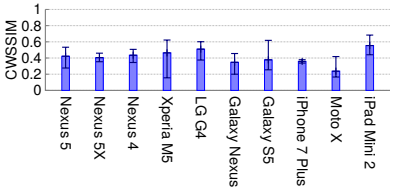

Figure 18: Impact of device heterogeneity. Error bars show *std.* across OOK waveforms with different frequencies (100 Hz to 500 Hz).

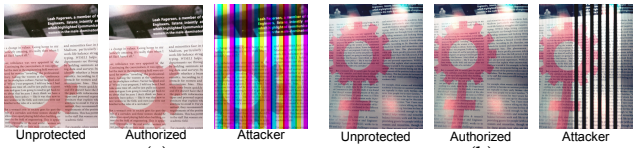

Figure 19: Frames observed by authorized users and attackers for (a) static scene (b) dynamic scene.

image quality stays high, with PSNR> 30 dB and CW-SSIM> 0.9 (slightly lower than simulation results due to digital noises in real cameras). Despite the use of a basic configuration, LiShield degrades the image quality by 3 to 10 dB for PSNR and 0.25 to 0.45 for CW-SSIM (Fig. 10). We notice that *the quality worsens slightly as flickering frequency decreases* from 500 Hz to 100 Hz, as the image sensor has higher analog gain at lower flickering frequencies [102]. In addition, *different scenes suffer from different levels of disruption*, depending on the scene's structure and reflection rate. As a visual *quality benchmark*, Fig. 11 plots the same scene with different qualities under flickering.

Impact of waveform duty cycle. We use 100 Hz flickering frequency on the document scene as a representative setup [5] to study the impact of duty cycle of emitted waveform. Here we enable auto-exposure to study the stripes' impact alone. Fig. 12 shows that lowering the duty cycle from 0.9 to 0.1 degrades the image quality dramatically, with CW-SSIM from nearly 0.6 to just over 0.1. However, higher duty cycle leads to more light influx and larger overexposure area (Fig. 13) when fix-exposure is used by attacker (here $t_e = 1/200$ s). To leverage both types of quality degradations (*i.e.*, flickering stripes and overexposure), *the LED should adopt a relatively moderate duty cycle but high peak intensity,* which echoes our model in Sec. 3.2.

Impact of RGB color distortion. We further verify the color-distortion impact when the RGB flickering is turned on. The results (Fig. 14) demonstrate slightly weaker quality degradation when its peak intensity is the same as monochrome LED (and thus average intensity is only 1/3). But *the quality degradation is stronger if the RGB LED has the same average intensity with monochrome LED.* Besides, the color distortion makes an additional independent impact.

[5]Unless otherwise noted, the rest of experiments use the same setup.

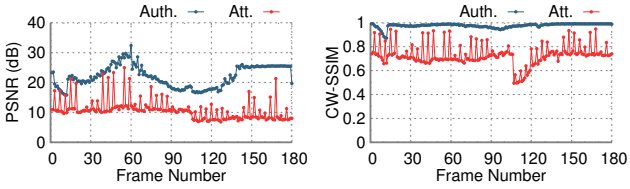

Figure 20: Video quality with and without authorization.

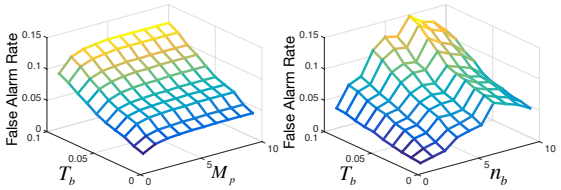

Figure 21: False alarm ratio across detector settings.

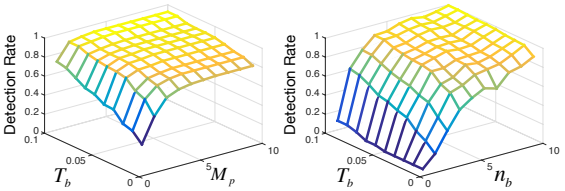

Figure 22: Detection rate across detector settings.

The corresponding CIEDE2000 metric (Fig. 15) escalates up to 45, way beyond the *human-tolerable threshold 6* [50]. This implies *the scene is no longer considered viewable by average viewers.*

Two *bonus effects* from our RGB LED are observed: *(i)* The structural distortion from the stripes disrupts the camera's auto-focus function, often *making the captured scene extremely blur.* This is because under LiShield, contrast of bands no longer depend on focusing accuracy, which breaks the assumption of auto-focus mechanism. *(ii) The color bands also mislead the automatic white balance function across all 5 different scenes*, since the camera can no longer identify a clean region in the image to calibrate itself and thus hesitates as shown in Fig. 16.

Impact on dynamic scenes. To create a dynamic scene, we use the motor to rotate the smartphone, creating relative motion at three different speeds (45, 100 and 145 degrees/second). Fig. 17 shows the average quality among all 3 speeds, which indicates that *dynamic scene experiences worse quality under LiShield* due to motion blur. Moreover, if the exposure time is larger than 1/100 s, then overexposure and motion blurs together further reduce the quality (PSNR < 6, CW-SSIM < 0.1). Thus, *dynamic objects further decrease the adjustment range of exposure time and make manual exposure attack more ineffective.*

Impact of device heterogeneity. We cross-validate the impact of LiShield on 10 common smartphone cameras. Fig. 18 shows the image quality varies slightly, due to different sampling rates across devices resulting in stripes of different widths. However, the quality remains at an intolerably low level across devices. Thus *LiShield's protection mechanism works across typical smartphone camera models.*

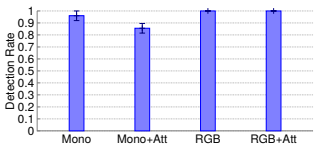

Figure 23: Detection rate of monochrome and RGB barcode design.

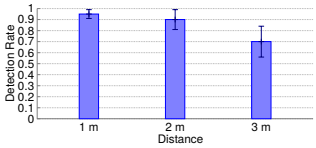

Figure 24: Barcode detection rate across distance under a single LED.

Table 1: Flicker-free configurations for monochrome barcode.

Seq	f_1 (Hz)	f_2 (Hz)	f_3 (Hz)	t_{att} (s)
1	400	600	800	1/400, 1/600, 1/800
2	1000	1200	1400	1/1000, 1/1200, 1/1400
3	1600	1800	2000	1/1600, 1/1800, 1/2000

Table 2: Flicker-free configurations for RGB barcode.

Color	f_1 (Hz)	f_2 (Hz)	f_3 (Hz)	t_{att} (s)
Red	400	600	800	1/2000 ~ 1/400
Green	1000	1200	1400	1/2000 ~ 1/400
Blue	1600	1800	2000	1/2000 ~ 1/400

7.2 Effectiveness of User Authorization

We developed an app (Sec. 6) that allows a user to capture critical frames on static scene protected by our RGB LED, and then recover the scene following Sec. 4. The resulting image quality (PSNR = 25dB, CW-SSIM = 0.9, CIEDE2000 = 5) is comparable to the ideal setting when we disable LiShield's LED modulation (Fig. 19 shows example frames extracted from a recorded video). In contrast, the attacker suffers intolerable image corruption (PSNR = 13dB, CW-SSIM = 0.56, CIEDE2000 = 34) by combining same number of randomly selected frames (Sec. 3.3.2).

For the dynamic scene, we set f_{intra} =1 kHz and f_{inter} =300 Hz (Sec. 4.1). From Fig. 20, we can see the authorized user has much higher quality (PSNR=25dB, CW-SSIM=0.98 in average) compared with attacker (PSNR = 10dB, CW-SSIM = 0.6 in average). This can be seen by resulting image frames in Fig. 19 where attacker suffers from both intra-frame and inter-frame stripes. Thus *LiShield's authorization scheme is effective in unblocking specific users while maintaining protection against attackers.*

7.3 Effectiveness of Barcode Embedding

We first determine general optimal parameters for LiShield's barcode detector (γ, T_b, n_{b_r} and n_{b_c} in Sec. 5), based on the following metrics. *(i)* False alarm rate. We run the detector on 200 images (random real-world scenes) in the SIPI database [93], and measure the probability that a barcode is detected from clean image. *(ii)* Detection rate. We embed monochrome barcodes with different f_1 from 400 Hz to 10 kHz with 200 Hz switching frequency. For each f_1, we embed 3 frequencies (*i.e.*, $M_R = 3$ in Sec. 5) with $\Delta f = 200$ Hz interval and capture 300 images with these barcodes over a benchmark scene (without loss of generality) to obtain detection rate. For simplicity we set $n_b = n_{b_r} = n_{b_c}$. Fig. 21 plots the fraction of falsely detected frequency ratios (*i.e.*, R_p in Sec. 5) over total number of ratios, while Fig. 22 shows successful detection rate under the same set of parameters. Considering the trade-off between false alarm and detection, we choose $T_b = 0.05$, $M_p = 2$ and $n_b = 4$ to bound the false alarm rate below 5%, and set $M_b = \lceil 2 \times 5\% \times M_R + M_{\text{att}} \rceil = 3$ to guarantee no false alarm for barcodes with 3 frequencies ($M_{\text{att}} = 2$ since manual exposure attack can remove two R_p), while ensuring around 90% detection rate for monochrome barcode.

Using the above configuration and frequencies in Tables 1 and 2, Fig. 23 shows that detection rate for RGB barcodes is close to 100% with or without manual exposure attack, while being slightly below 90% for monochrome barcodes if attacked. We conclude that *LiShield's barcode detector provides reliable detection, while RGB barcodes are more detectable and robust than monochrome ones*, thanks to extra redundancy provided by color channels.

An attacker may post-process the image in attempt to remove the watermark. However, thanks to the redundancy of the barcode, the attacker will have to deform most parts of the image, which greatly reduces the image quality and makes the attack nonviable.

Table 3: Flicker-free configurations for frequency randomization. f_c, t_{att} represent center frequency and attacker's exposure time.

M	$f_B = f_1$ (Hz)	Δf (Hz)	f_c (Hz)	t_{att} (s)
2	200	50	225	1/225
3	300	50	350	1/350
4	400	50	475	1/475
5	500	50	600	1/600
6	600	50	725	1/725

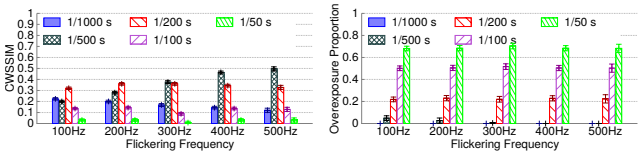

Figure 25: Quality and overexposed proportion with fix-exposure camera.

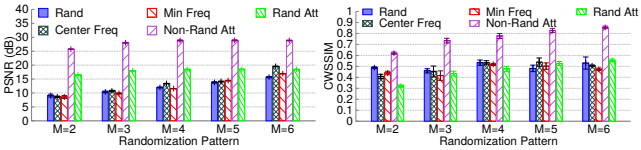

Figure 26: Quality with and without frequency randomization.

7.4 Robustness and Counteracting Attacks

Manual exposure attack. One possible attack against LiShield is to manually set the exposure time t_e to smooth out the flickering patterns (Sec. 3.3). Fig. 25 shows that although the image quality first increases with t_e, it drops sharply as overexposure occurs. Therefore, *LiShield traps the attacker in either extremes by optimizing the waveform* (Sec. 3.2), and *thwarts any attempts through exposure time configuration.*

We further test the effectiveness of randomization as configured in Table 3 with auto-exposure (except for attacker). Fig. 26 shows that the image quality with scrambling is comparable with single frequency of f_1 and f_c, thus *frequency randomization does not cause much difference in image quality.* Note that the image quality varies only slightly with number of frequencies, implying *it is insensitive to LiShield's frequency randomization pattern.* We assume the exposure time is $t_{\text{att}} = 1/f_c$, which is optimistic for the attacker. Results show that image quality does not vary significantly (compared with attacks to stripes without randomization), showing *LiShield's robustness against such attacks.*

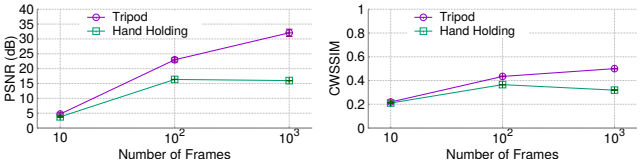

Figure 27: Image quality with number of frames of multi-frame attack.

Multi-frame attack. Fig. 27 plots the recovered scene's quality under the multi-frame attack. Here we set t_e to be 1/500 s to avoid overexposure and then record a video in 30 fps. When a tripod is used, PSNR goes to 30 dB but CW-SSIM remains low at 0.5 using 1000 frames, which means *the impact of stripes on structure of scene is still strong although intensity does not differ substantially, making quality still unacceptable for professionals who spend such a great cost.* We also ask 5 volunteers to hold the smartphone as stable as they can on a table, and Fig. 27 shows the quality is even lower, because it is impossible to completely avoid dithering with hands even with anti-shake technology, making recovered scene unviewable. Extending the recording duration increases the number of frames recorded by the attacker, but it also increases disturbance and probability of being identified by the protected user, making it risky and impractical for the attacker to pursue higher quality.

Image recovery processing attack. Fig. 28 shows the image quality after post-processing with denoising or de-banding (Sec. 6). The denoising methods fail to improve the quality significantly as the disruption pattern of LiShield does not fit any known noise model (*e.g.* the commonly used Gaussian model). BM3D can improve the CW-SSIM slightly because it decreases contrast slightly, but the PSNR remains low. The deformation removal methods (*i.e.*, de-banding and unstriping) do not help either, since interpolation process cannot bring back the correct pixel values. The CIEDE2000 color metric also shows a low quality (well above 6). Thus, *it is practically impossible to remove LiShield's impact by image processing, despite some unnoticeable increase of the image quality.* More advanced computer vision techniques may provide better recovery, but even they will not recover the *exactly original scene* since information is already lost at capture time.

Impact of ambient light. We evaluate LiShield's performance under different types of ambient lights and LED power to verify LiShield's robustness. As shown in Fig. 29, the stripes are almost completely removed under direct sunlight due to its extremely high intensity, making the quality comparable with the unprotected case (PSNR>30dB, CW-SSIM>0.9). However, CIEDE2000 remains relatively high as LED's light affects the scene's color tone significantly, which explains unexpected image quality degradation under diffused sunlight and office light in Fig. 29. Flash light can increase the quality slightly thanks to its close distance to the scene, but the improvement is marginal and far from unprotected. In addition, Fig. 30 shows a slight decrease of detection rate of barcode under direct sunlight, but the decrease is marginal in every case. Thus, we conclude that *LiShield is robust against ambient light, and still guarantees protection with barcode under direct sunlight.*

Impact of distance. We vary the distance between camera and *a single LED* from 1 m to 3 m. The scene resolution lowers at longer distance. Fig. 24 shows the barcode detection rate remains high (> 90%) at 2 m range (where the bright area is only 1/4 on the image compared with 1 m case). However only 70% rate can be achieved at 3 m range, since the bright area is too small on the image (1/9). But multiple LEDs may be distributed to increase the coverage. To make a fair comparison on quality, we tailor the same scene from image to avoid interference from surrounding objects. Fig. 31 shows that even under 3 m, CW-SSIM is still way below 0.9 and the quality only increases slightly with distance. Thus, *LiShield's working range can cover most of common applications with only a single smart LED.* With multiple smart LEDs, LiShield's coverage can be scaled up just like normal lighting (Sec. 8).

8 DISCUSSION

Considerations for high peak intensity. Considering the hardware capability and energy costs, we estimate the optimal LED peak intensity to be 20 kLux, and average intensity is 10 kLux with 0.5 duty cycle, which is an order of magnitude lower than outdoor intensity in a sunny day [85], and generally considered safe even for long activities [82]. Our smart LED is brighter than typical indoor lighting, which is usually less than 1 kLux. But we found the intensity is always acceptable by perception in our experiments, likely because the brightness perceived by human eyes and actual light intensity follow a logarithmic relationship. Since the privacy protection has higher priority than power savings, we expect slight increase of illumination brightness is acceptable in the target use cases.

Multiple LEDs and multiple cameras. When a large indoor environment needs LiShield's protection, the smart LEDs can be deployed pervasively to cover the whole area, just like regular lighting. Availability of multiple LEDs can also increase the diversity of the protection, since each of them can be controlled independently. We leave the optimization of such multi-LED setup for future work.

With the presence of multiple unauthorized cameras, LiShield needs to ensure no additional information can be recovered by combining images across them, which may impose extra constraints on waveform design. Meanwhile, when multiple authorized cameras (Sec. 4) are present, LiShield can serve them in a round-robin manner. Better strategies may require synchronization between cameras and we leave them for future work.

Attacker with special equipment. Global shutter cameras, ND filters (optical attenuators) and similar professional devices may compromise LiShield's protection. While this is inevitable, we note that such devices are usually bulky and costly, which makes them obtrusive and less accessible by everyday photographers. Thus, LiShield may still protect the privacy of its users by demotivating such attacks. An advanced version of LiShield that fully prevents such attacks will be left for our future work.

Recording a high speed video (*e.g.*, 120 FPS) by advanced cameras will not significantly weaken LiShield's protection, as stripes across frames will be similar. High FPS also requires shorter exposure, which actually amplifies LiShield's effect. Along with the backup barcode protection, which is not affected by the camera's frame rate, the threat posed by high speed camera is limited.

9 RELATED WORKS

Anti-piracy and capturing-resistant technologies. Camera recording of copyright screen-displayed videos (*e.g.*, in a movie theater) accounts for 90% of pirated online content [104]. Since

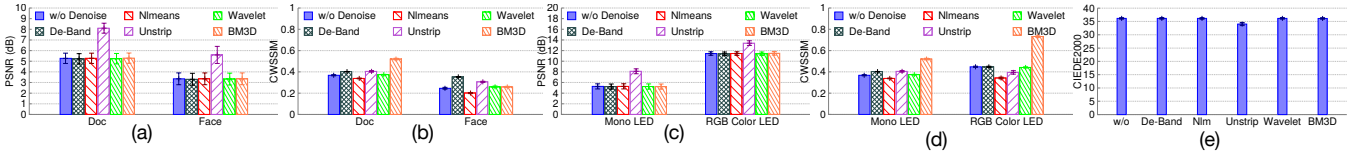

Figure 28: Effects of denoising and de-banding image processing algorithms.

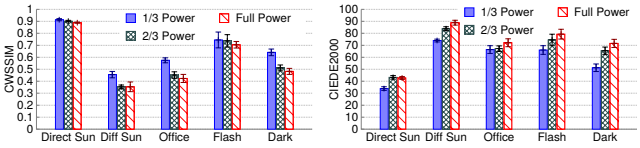

Figure 29: Image Quality under ambient lights.

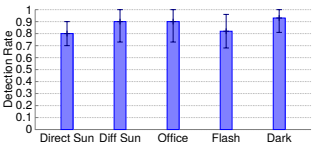

Figure 30: Barcode detection rate with ambient light intensity.

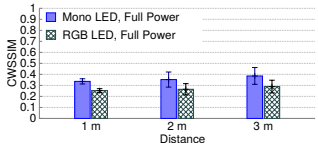

Figure 31: Image quality with different distances under a single LED.

screen refresh rate is much higher than video frame rate, Kaleido [104] scrambles multiple frames within the frame periods to deter recording, while preserving viewing experience by taking advantage of human eyes' flicker fusion effects. Many patented technologies addressed the same issue [10, 11, 20, 21, 30, 43, 61, 68, 71–73, 75, 79, 94, 100]. In contrast, the problem of automatic protection of private and passive physical space received little attention. Certain countries [19, 28] dictate that smartphone cameras must make shutter sound to disclose the photo-capturing actions, yet this does not enforce the compliance, cannot block the photo distribution, and cannot automatically protect against video recording.

Certain optical signaling systems can remotely ban photography in concerts, theaters and other capturing-sensitive sites. Courteous Glass [42] and a recent Apple patent [84] augment wearable devices with near-infrared LEDs, which are invisible to human but can be captured by camera, to convey hidden privacy appeal of the wearers. These LEDs cannot enforce protection (*e.g.*, through image corruption as in LiShield), and convey information only when they fall in the camera's field of view. BlindSpot [86] adopts a computer vision approach to locate retro-reflective camera lenses, and pulses a strong light beam towards the camera to cause overexposure. Despite its sophistication, approach fails when multiple cameras coexist with arbitrary orientations.

Invisible screen-camera communications. Recent research also explored novel ways of screen-to-camera visible light communication, by hiding information behind the display. VRCodes [97] carries information through high frequency changes of selected color, which can be decoded by rolling-shutter cameras. Hilight [47] conveys information by modulating the pixel translucency change in subtle ways. ARTcode [99] embeds data into images by modifying the pixels' colors, which is imperceptible due to human eyes' limited pixel resolution. This line of research is also related with the classical watermarking which hides copyright and authentication information in images/videos through spatial/frequency domain re-encoding [1, 59]. These mechanisms are applicable when the users have full control over the image/video source, but cannot prevent malicious capturing/distribution of physical scenes. On the other hand, conventional luminaries bear natural flickering effects that have been leveraged for localization purposes [102, 103, 106], but the frequencies are too high to cause visible corruption on the camera images.

Privacy protection for images/videos. Conventional visual privacy-protection systems have been replying on post-capture processing. Early efforts employed techniques like region-of-interest masking, blurring, mosaicking, *etc.* [56], or re-encoding using encrypted scrambling seeds [18]. There also exists a vast body of work for hiding copyright marks and other information in digital images/videos [27, 38, 39, 45, 52, 63, 89, 90, 97, 101]. LiShield's barcode protection is inspired by these schemes, but it aims to protect physical scenes prior to capturing.

One common challenge in visual privacy protection is to identify the privacy preference. Location-bound privacy expression can be achieved in everyday life using special signs. Privacy.Tag [7] allows people to express their privacy preference by wearing QR codes. I-Pic [2] allows people to broadcast their privacy preferences using Bluetooth. COIN [105] matches a user's face to a prescribed privacy preference, and can automatically detect and mask people who do not want to be captured. P3 [63] protects photo-sharing privacy by encoding an image into a private, encrypted part, and a public, standards-compatible part. PrivacyEye [65] allows a user to manually mark regions on an image that permit access from mobile apps. PlaceAvoider [80] allows first-person camera users to capture and blacklist sensitive spaces *a priori*, and use image matching to block subsequent pictures containing such spaces. These systems only work when the user has full control over the camera.

10 CONCLUSION

Privacy protection in passive indoor environment has been an important but unsolved problem. In this paper we propose LiShield, which uses smart-LEDs and specialized intensity waveforms to disrupt unauthorized cameras, while allowing authorized users to record high quality image and video. We implemented and evaluated LiShield under various representative indoor scenarios, which demonstrates LiShield's effectiveness and robustness. We consider LiShield as a first exploration of automatic visual privacy enforcement and expect it can inspire more research along the same direction.

ACKNOWLEDGEMENT

We would like to thank the anonymous reviewers and our shepherd Marco Grutester for their valuable comments and feedback. This project was partially supported by the NSF under Grant CNS-1343363, CNS-1350039, CNS-1506657, CNS-1518728, and CNS-1617321.

REFERENCES

[1] 2010. Digital Image Steganography: Survey and Analysis of Current Methods. *Signal Processing* 90, 3 (2010).

[2] Paarijaat Aditya, Rijurekha Sen, Peter Druschel, Seong Joon Oh, Rodrigo Benenson, Mario Fritz, Bernt Schiele, Bobby Bhattacharjee, and Tong Tong Wu. 2016. I-Pic: A Platform for Privacy-Compliant Image Capture. In *Proceedings of ACM Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*.

[3] Saeed Al-Mansoori and Alavi Kunhu. 2012. Robust watermarking technique based on DCT to protect the ownership of DubaiSat-1 images against attacks. *International Journal of Computer Science and Network Security (IJCSNS)* 12, 6 (2012), 1.

[4] Stephen J. Anderson and David C. Burr. 1985. Spatial and temporal selectivity of the human motion detection system. *Vision Research* 25, 8 (1985), 1147 – 1154.

[5] Mauro Barni. 2006. *Document and Image compression*. CRC press.

[6] BBC. 2004. The Camera Phone Backlash. (2004). http://news.bbc.co.uk/2/hi/uk_news/magazine/3793501.stm

[7] Cheng Bo, Guobin Shen, Jie Liu, Xiang-Yang Li, YongGuang Zhang, and Feng Zhao. 2014. Privacy.Tag: Privacy Concern Expressed and Respected. In *ACM Conference on Embedded Network Sensor Systems (SenSys)*.

[8] Jérôme Boulanger. [n. d.]. jboulanger.gmic. ([n. d.]). https://github.com/jboulanger/jboulanger-gmic

[9] Antoni Buades, Bartomeu Coll, and J-M Morel. 2005. A non-local algorithm for image denoising. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, Vol. 2. IEEE, 60–65.

[10] Herschel Clement Burstyn. 2009. Cinema anti-piracy measures. (2009). US Patent 7,634,089.

[11] Herschel Clement Burstyn, George Herbert Needham Riddle, Leon Shapiro, and David Lloyd Staebler. 2008. Method and apparatus for film anti-piracy. (2008). US Patent 7,324,646.

[12] Tung-Shou Chen, Chin-Chen Chang, and Min-Shiang Hwang. 1998. A virtual image cryptosystem based upon vector quantization. *IEEE transactions on Image Processing* 7, 10 (1998), 1485–1488.

[13] Cinetics. [n. d.]. Axis360 Pro. http://cinetics.com/axis360-pro/. ([n. d.]).

[14] Ronald R Coifman and David L Donoho. 1995. *Translation-invariant de-noising*. Springer.

[15] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. 2006. Image denoising with block-matching and 3D filtering. In *Electronic Imaging*. International Society for Optics and Photonics.

[16] Christos Danakis, Mostafa Afgani, Gordon Povey, Ian Underwood, and Harald Haas. 2012. Using a CMOS camera sensor for visible light communication. In *Proc. of IEEE Globecom Workshops*.

[17] Tamara Denning, Zakariya Dehlawi, and Tadayoshi Kohno. 2014. In Situ with Bystanders of Augmented Reality Glasses: Perspectives on Recording and Privacy-mediating Technologies. In *SIGCHI Conference on Human Factors in Computing Systems (CHI)*.

[18] F. Dufaux and T. Ebrahimi. 2008. Scrambling for Privacy Protection in Video Surveillance Systems. *IEEE Transactions on Circuits and Systems for Video Technology* 18, 8 (2008).

[19] Engadget. 2016. Japan's noisy iPhone problem. (2016). https://www.engadget.com/2016/09/30/japans-noisy-iphone-problem/

[20] Michael Epstein and Douglas A Stanton. 2003. Method and device for preventing piracy of video material from theater screens. (2003). US Patent 6,529,600.

[21] James A Fancher, David H Sitrick, and Gregory P Sitrick. 2003. Movie film security system utilizing infrared patterns. (2003). US Patent 6,559,883.

[22] Iain Fergusson. [n. d.]. External Filters by Iain Fergusson. ([n. d.]). https://github.com/dtschump/gmic-community/blob/master/include/iain_fergusson.gmic

[23] Federico Ferrari, Marco Zimmerling, Lothar Thiele, and Olga Saukh. 2011. Efficient network flooding and time synchronization with glossy. In *Proc. of ACM/IEEE IPSN*. 73–84.

[24] Forbes. 2011. Adventures in Self-Surveillance, aka The Quantified Self, aka Extreme Navel-Gazing. (Apr. 2011).

[25] Nobuhiro Fujimoto and Hikari Mochizuki. 2013. 477 Mbit/s visible light transmission based on OOK-NRZ modulation using a single commercially available visible LED and a practical LED driver with a pre-emphasis circuit. In *National Fiber Optic Engineers Conference*. Optical Society of America, JTh2A–73.

[26] Nobuhiro Fujimoto and Shohei Yamamoto. 2014. The fastest visible light transmissions of 662 Mb/s by a blue LED, 600 Mb/s by a red LED, and 520 Mb/s by a green LED based on simple OOK-NRZ modulation of a commercially available RGB-type white LED using pre-emphasis and post-equalizing techniques. In *Optical Communication (ECOC), 2014 European Conference on*. IEEE, 1–3.

[27] Zhongpai Gao, Guangtao Zhai, and Chunjia Hu. 2015. The invisible qr code. In *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 1047–1050.

[28] Golf News Net. 2015. Why is it illegal in South Korea to silence mobile phone camera sounds? (2015). https://thegolfnewsnet.com/golfnewsnetteam/2015/10/07/illegal-south-korea-silence-mobile-phone-camera-sounds-13503/

[29] Rafael L Gomes, Luiz F Bittencourt, Edmundo RM Madeira, Eduardo Cerqueira, and Mario Gerla. 2015. QoE-Aware dynamic virtual network resource adaptation for EaaS environment. In *Communications (ICC), 2015 IEEE International Conference on*. IEEE, 6836–6841.

[30] Dean K Goodhill and Ty Safreno. 2011. Method and apparatus for inhibiting the piracy of motion pictures. (2011). US Patent 8,018,569.

[31] Joseph W Goodman. 2005. *Introduction to Fourier optics*. Roberts and Company Publishers.

[32] Google. [n. d.]. android.hardware.camera2. ([n. d.]). https://developer.android.com/reference/android/hardware/camera2/package-summary.html

[33] Phil Green and Lindsay MacDonald. 2011. *Colour engineering: achieving device independent colour*. Vol. 30. John Wiley & Sons.

[34] Stacey L. Gulick. 2004. Preventing Unauthorized Audio and Video Recording at Your Practice. (2004). http://medicaleconomics.modernmedicine.com/medical-economics/content/tags/hipaa/preventing-unauthorized-audio-and-video-recording-your-practice?page=full

[35] Mark Harman. [n. d.]. Open Camera. ([n. d.]). http://opencamera.sourceforge.net/

[36] David Honzátko. [n. d.]. CUDA implementation of BM3D. ([n. d.]). https://github.com/DawyD/bm3d-gpu

[37] Roberto Hoyle, Robert Templeman, Steven Armes, Denise Anthony, David Crandall, and Apu Kapadia. 2014. Privacy Behaviors of Lifeloggers Using Wearable Cameras. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.

[38] Chin-Wei Hsu, Kevin Liang, Hung-Yu Chen, Liang-Yu Wei, Chien-Hung Yeh, Yang Liu, and Chi-Wai Chow. 2016. Visible light encryption system using camera image sensor. In *OptoElectronics and Communications Conference (OECC) held jointly with 2016 International Conference on Photonics in Switching (PS), 2016 21st*. IEEE, 1–3.

[39] Chunjia Hu, Guangtao Zhai, and Zhongpai Gao. 2015. Visible light communication via temporal psycho-visual modulation. In *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 785–788.

[40] Image Team of the GREYC laboratory. [n. d.]. G'MIC - GREYC's Magic for Image Computing. ([n. d.]). http://gmic.eu/

[41] Yi Jiang, Ke Zhou, and Sheng He. 2007. Human visual cortex responds to invisible chromatic flicker. *Nature Neuroscience* 10, 5 (2007), 657–662.

[42] Jaeyeon Jung and Matthai Philipose. 2014. Courteous Glass. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.

[43] Masayuki Karakawa. 2006. Laser video projection system and method with anti-piracy feature. (2006). US Patent 7,103,074.

[44] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. 2014. Luxapose: Indoor Positioning with Mobile Phones and Visible Light. In *Proc. of ACM MobiCom*.

[45] Hui-Yu Lee, Hao-Min Lin, Yu-Lin Wei, Hsin-I Wu, Hsin-Mu Tsai, and Kate Ching-Ju Lin. 2015. Rollinglight: Enabling line-of-sight light-to-camera communications. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 167–180.

[46] LG Electronics. [n. d.]. Nexus 5. ([n. d.]). http://www.lg.com/ca_en/cell-phones/lg-D820-White-nexus5

[47] Tianxing Li, Chuankai An, Xinran Xiao, Andrew T. Campbell, and Xia Zhou. 2015. Real-Time Screen-Camera Communication Behind Any Scene. In *Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*.

[48] Zhenjiang Li, Cheng Li, Wenwei Chen, Jingyao Dai, Mo Li, Xiang-Yang Li, and Yunhao Liu. 2012. Clock Calibration Using Fluorescent Lighting. In *Proceedings of International Conference on Mobile Computing and Networking (MobiCom)*.

[49] C. K. Liang, L. W. Chang, and H. H. Chen. 2008. Analysis and Compensation of Rolling Shutter Effect. *IEEE Transactions on Image Processing* 17, 8 (2008).

[50] M Ronnier Luo, Guihua Cui, and B Rigg. 2001. The development of the CIE 2000 colour-difference formula: CIEDE2000. *Color Research & Application* 26, 5 (2001), 340–350.

[51] Maxim Integrated Products, Inc. 2004. Why Drive White LEDs with Constant Current? https://www.maximintegrated.com/en/app-notes/index.mvp/id/3256. (Aug. 2004).

[52] Moni Naor and Adi Shamir. 1995. Visual Cryptography. In *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques (EUROCRYPT)*.

[53] Narrative. 2016. Narrative Clip 2 Wearable HD Video Camera. (2016). http://getnarrative.com/

[54] T. Naseer, J. Sturm, and D. Cremers. 2013. FollowMe: Person Following and Gesture Recognition With a Quadrocopter. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*.

[55] Alaeddin Nassani, Huidong Bai, Gun Lee, and Mark Billinghurst. 2015. Tag It!: AR Annotation Using Wearable Sensors. In *SIGGRAPH Asia Mobile Graphics and Interactive Applications*.

[56] Elaine M. Newton, Latanya Sweeney, and Bradley Malin. 2005. Preserving Privacy by De-Identifying Face Images. *IEEE Transactions on Knowledge and Data Engineering* 17, 2 (2005).

[57] S. John Obringer and Kent Coffey. 2007. Cell Phones in American High Schools: A National Survey. *Journal of Technology Studies* 33, 1 (2007).

[58] ON Semiconductor. [n. d.]. NCL30160: LED Driver, Constant Current Buck Regulator, 1.0 A. ([n. d.]). http://www.onsemi.com/PowerSolutions/product.do?id=NCL30160

[59] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn. 1999. Information Hiding–a Survey. *Proc. IEEE* 87, 7 (1999).

[60] Philips Lighting B.V. [n. d.]. Philips Hue. ([n. d.]). http://meethue.com

[61] John D Price. 2009. Methods and apparatus for detection of motion picture piracy for piracy prevention. (2009). US Patent App. 12/322,915.

[62] QImaging. 2014. Rolling Shutter vs. Global Shutter. (2014). https://www.qimaging.com/ccdorscmos/pdfs/RollingvsGlobalShutter.pdf

[63] Moo-Ryong Ra, Ramesh Govindan, and Antonio Ortega. 2013. P3: Toward Privacy-preserving Photo Sharing. In *USENIX Conference on Networked Systems Design and Implementation (NSDI)*.

[64] Swati Rallapalli, Aishwarya Ganesan, Krishna Chintalapudi, Venkat N. Padmanabhan, and Lili Qiu. 2014. Enabling Physical Analytics in Retail Stores Using Smart Glasses. In *Annual International Conference on Mobile Computing and Networking (MobiCom)*.

[65] Nisarg Raval, Animesh Srivastava, Ali Razeen, Kiron Lebeck, Ashwin Machanavajjhala, and Lanodn P. Cox. 2016. What You Mark is What Apps See. In *Annual International Conference on Mobile Systems, Applications, and Services (MobiSys)*.

[66] Erik Reinhard, Wolfgang Heidrich, Paul Debevec, Sumanta Pattanaik, Greg Ward, and Karol Myszkowski. 2010. *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann.

[67] Mehul P Sampat, Zhou Wang, Shalini Gupta, Alan Conrad Bovik, and Mia K Markey. 2009. Complex wavelet structural similarity: A new image similarity index. *IEEE transactions on image processing* 18, 11 (2009), 2385–2401.

[68] Yosef Sanhedrai, Ariel Schwarz, Liad Ben Yishai, and Zeev Zalevsky. 2007. System and method for preventing photography. (2007). US Patent App. 12/308,525.

[69] Oren Shani. [n. d.]. Precise Time Synchronization Over WLAN. ([n. d.]). http://www.ti.com/lit/an/swaa162a/swaa162a.pdf

[70] Ling Shao, Ruomei Yan, Xuelong Li, and Yan Liu. 2014. From heuristic optimization to dictionary learning: A review and comprehensive comparison of image denoising algorithms. *IEEE Transactions on Cybernetics* 44, 7 (2014), 1001–1013.

[71] David H Sitrick and James A Fancher. 2004. Anti-piracy protection system and methodology. (2004). US Patent 6,771,349.

[72] David H Sitrick and James A Fancher. 2007. Targeted anti-piracy system and methodology. (2007). US Patent 7,170,577.

[73] David H Sitrick and James A Fancher. 2011. System and methodology for validating compliance of anti-piracy security and reporting thereupon. (2011). US Patent 8,006,311.

[74] Fikret Sivrikaya and Bülent Yener. 2004. Time Synchronization in Sensor Networks: a Survey. *IEEE network* 18, 4 (2004), 45–50.

[75] Vincent So. 2009. Anti-piracy image display methods and systems. (2009). US Patent 7,634,134.

[76] Social Pilot. 2016. 125 Amazing Social Media Statistics You Should Know. (2016). https://socialpilot.co/blog/125-amazing-social-media-statistics-know-2016/

[77] S.S. Stevens. 1975. *Psychophysics: introduction to its perceptual, neural, and social prospects*. Transaction Publishers.

[78] STMicroelectronics. [n. d.]. STM32F103C8. ([n. d.]). http://www.st.com/en/microcontrollers/stm32f103c8.html

[79] Emil Tchoukaleysky. 2013. Digital cinema anti-piracy method and apparatus for liquid crystal projection systems. (2013). US Patent 8,559,791.

[80] Robert Templeman, Mohammed Korayem, David J. Crandall, and Apu Kapadia. 2014. PlaceAvoider: Steering First-Person Cameras away from Sensitive Spaces. In *Network and Distributed System Security Symposium (NDSS)*.

[81] R. Templeman, Z. Rahman, D. Crandall, and A. Kapadia. 2013. PlaceRaider: Virtual Theft in Physical Spaces With Smartphones. In *Network and Distributed System Security Symposium (NDSS)*.

[82] Michael Terman and Jiuan Su Terman. 2005. Light therapy for seasonal and nonseasonal depression: efficacy, protocol, safety, and side effects. *CNS spectrums* 10, 08 (2005), 647–663.

[83] The GIMP Team. [n. d.]. GIMP - GNU Image Manipulation Program. ([n. d.]). https://www.gimp.org/

[84] Victor Tiscareno, Kevin Jonhson, and Cindy Lawrence. 2011. Systems and Methods for Receiving Infrared Data with a Camera Designed to Detect Images. (2011).

[85] PR Tregenza. 1980. The daylight factor and actual illuminance ratios. *Lighting Research & Technology* 12, 2 (1980), 64–68.

[86] Khai N. Truong, Shwetak N. Patel, Jay W. Summet, and Gregory D. Abowd. 2005. *Preventing Camera Recording by Designing a Capture-Resistant Environment*.

[87] Christopher W. Tyler. 1985. Analysis of visual modulation sensitivity. II. Peripheral retina and the role of photoreceptor dimensions. *Journal of the Optical Society of America A* 2, 3 (1985), 393–398.

[88] H. van der Broeck, G. Sauerlander, and M. Wendt. 2007. Power driver topologies and control schemes for LEDs. In *Annual IEEE Applied Power Electronics Conference and Exposition*.

[89] Anran Wang, Zhuoran Li, Chunyi Peng, Guobin Shen, Gan Fang, and Bing Zeng. 2015. Inframe++: Achieve simultaneous screen-human viewing and hidden screen-camera communication. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services*. ACM, 181–195.

[90] Anran Wang, Chunyi Peng, Ouyang Zhang, Guobin Shen, and Bing Zeng. 2014. Inframe: Multiflexing full-frame visible communication channel for humans and devices. In *Proceedings of the 13th ACM Workshop on Hot Topics in Networks*. ACM, 23.

[91] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[92] Zhou Wang and Eero P Simoncelli. 2005. Translation insensitive image similarity in complex wavelet domain. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on*, Vol. 2. IEEE, ii–573.

[93] Allan G Weber. 1997. The USC-SIPI image database version 5. *USC-SIPI Report* 315 (1997), 1–24.

[94] Donald Henry Willis. 2008. Method, apparatus and system for anti-piracy protection in digital cinema. (2008). US Patent App. 12/736,774.

[95] H. James Wilson. 2012. You, By the Numbers. *Harvard Business Review* (Sep. 2012).

[96] W. Winterhalter, F. Fleckenstein, B. Steder, L. Spinello, and W. Burgard. 2015. Accurate Indoor Localization for RGB-D Smartphones and Tablets Given 2D Floor Plans. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*.

[97] Grace Woo, Andy Lippman, and Ramesh Raskar. 2012. VRCodes: Unobtrusive and Active Visual Codes for Interaction by Exploiting Rolling Shutter. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*.

[98] Nan Xu, Fan Zhang, Yisha Luo, Weijia Jia, Dong Xuan, and Jin Teng. 2009. Stealthy Video Capturer: a New Video-Based Spyware in 3G smartphones. In *Proceedings of the ACM conference on Wireless Network Security (WiSec)*.

[99] Zhe Yang, Yuting Bao, Chuhao Luo, Xingya Zhao, Siyu Zhu, Chunyi Peng, Yunxin Liu, and Xinbing Wang. 2016. ARTcode: Preserve Art and Code in Any Image. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.

[100] Hong Heather Yu and Prabir Bhattacharya. 2006. Methods and apparatus for digital content protection. (2006). US Patent 7,006,630.

[101] Guangtao Zhai and Xiaolin Wu. 2014. Defeating camcorder piracy by temporal psychovisual modulation. *Journal of Display Technology* 10, 9 (2014), 754–757.

[102] Chi Zhang and Xinyu Zhang. 2016. LiTell: Robust Indoor Localization Using Unmodified Light Fixtures. In *Proc. of ACM MobiCom*.

[103] Chi Zhang and Xinyu Zhang. 2017. Pulsar: Towards Ubiquitous Visible Light Localization. In *Proc. of ACM MobiCom*.

[104] Lan Zhang, Cheng Bo, Jiahui Hou, Xiang-Yang Li, Yu Wang, Kebin Liu, and Yunhao Liu. 2015. Kaleido: You Can Watch It But Cannot Record It. In *ACM International Conference on Mobile Computing and Networking (MobiCom)*.

[105] Lan Zhang, Kebin Liu, Xiang-Yang Li, Cihang Liu, Xuan Ding, and Yunhao Liu. 2016. Privacy-friendly Photo Capturing and Sharing System. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.

[106] Shilin Zhu and Xinyu Zhang. 2017. Enabling High-Precision Visible Light Localization in Today's Buildings. In *Proc. of ACM MobiSys*.