Rolling Colors: Adversarial Laser Exploits against Traffic Light Recognition

Chen Yan[1], Zhijian Xu[1,2]*, Zhanyuan Yin[3], Xiaoyu Ji[1]†, Wenyuan Xu[1]
[1]*Zhejiang University,* [2]*The Chinese University of Hong Kong,* [3]*The University of Chicago*

Abstract

Traffic light recognition is essential for fully autonomous driving in urban areas. In this paper, we investigate the feasibility of fooling traffic light recognition mechanisms by shedding laser interference on the camera. By exploiting the rolling shutter of CMOS sensors, we manage to inject a color stripe overlapped on the traffic light in the image, which can cause a red light to be recognized as a green light or vice versa. To increase the success rate, we design an optimization method to search for effective laser parameters based on empirical models of laser interference. Our evaluation in emulated and real-world setups on 2 state-of-the-art recognition systems and 5 cameras reports a maximum success rate of 30% and 86.25% for Red-to-Green and Green-to-Red attacks. We observe that the attack is effective in continuous frames from more than 40 meters away against a moving vehicle, which may cause end-to-end impacts on self-driving such as running a red light or emergency stop. To mitigate the threat, we propose redesigning the rolling shutter mechanism.

1 Introduction

A full autonomous driving requires the vehicle to behave independently not just on highways but also in urban settings, where traffic light recognition is essential. Such systems enable the vehicle to visually detect and recognize the traffic light signals so that it can respond properly. According to NHTSA [20], intersection accidents are the second leading cause of vehicle collisions, led only by rear-end crashes. Reliable traffic light recognition can effectively prevent such accidents by provoking automatic responses of the vehicle. For this reason, car manufacturers such as BMW [5] and parts suppliers such as Mobileye [15] are actively developing and testing their own traffic light recognition systems. Tesla [18] has recently deployed traffic light recognition on their vehicles in the U.S. Looking forward, we will not be surprised to see a prevalence of such systems in the near future.

Before that day, a critical question is how safe the vision-based traffic light recognition systems are. The consequence of erroneous recognition is severe: recognizing a red light as green may cause intersection collisions; recognizing a green light as red may force stop the vehicle and jam the traffic. Previous studies have shown the insecurity of similar vision-based object detection and recognition systems. For example, a strong light can over-expose the camera and cause denial-of-service of the collision avoidance system [23, 37]; a projector can inject adversarial ghost images on an object [13, 17] or into the camera [14] to fool the advanced driver-assistance system. However, the security of traffic light recognition has not been studied yet. In addition, previous attacks generally used light that may raise stealthiness concerns and are constricted to an attack distance of a few meters.

We are motivated to investigate the feasibility of fooling vision-based traffic light recognition in the real world. To achieve a stealthier effect and longer attack distance, we choose laser as the attack signal, which is a very narrow and energy-concentrated beam of light. We envision that a laser shed into the camera's lens may modify the captured images and affect any subsequent recognition systems. However, such an attack is non-trivial due to the following reasons. (1) Though cameras are known to be susceptible to optical interference, a laser can easily saturate the CMOS sensor in the camera and render the entire image unrecognizable. It is unknown whether a laser can modify the image in a controllable way, especially only around the traffic light's area. (2) Traffic light detection is normally performed before recognition. The attacker needs to ensure that the laser interference does not disrupt the traffic light detection so that it can fool the recognition. (3) The red, green, and yellow lamps are on different parts of a traffic light. It is unknown whether and how an attacker can cause a red light to be recognized as a green light (or vice versa) without physically altering the lamps.

In this paper, we overcome these challenges and validate the attack's feasibility. After studying all potential effects a laser may impose on camera imaging, we manage to create

*This research was conducted when the author was at Zhejiang University.
†Corresponding author: xji@zju.edu.cn.

a color stripe overlapped on the traffic light in the image by exploiting the inherent vulnerability of the camera's rolling shutter. Our experiments show that color stripes of specific cases will not disrupt the traffic light detection and can cause desired color recognition results. To achieve a higher success rate, we empirically model the laser attack process and search for effective laser parameters based on simulated images under attack. We find that the attack can succeed with a wide range of laser parameters, suggesting high robustness to the uncertainties in real attack scenarios. We evaluate the attack in both emulated and real-world setups using real traffic light, laser diodes, and 5 cameras on 2 state-of-the-art open-source traffic light recognition systems. To demonstrate the potential real-world impacts, we experiment in practical settings and demonstrate successful attacks against a moving vehicle from more than 40 meters away in continuous frames. We propose rolling shutter redesign to mitigate the threat. We hope this work can help build more security into traffic light recognition for future autonomous vehicles.

Our contributions are summarized as follows:

- We discovered an inherent vulnerability of the rolling shutters in CMOS cameras that can be exploited by laser to create adjustable color stripes in an image, introducing a new approach to inject adversarial image patterns.
- We experimentally validated the feasibility of fooling traffic light recognition using laser, and proposed an optimization method to search for the effective laser parameters by simulating the attack with empirical models.
- We evaluated the attack in emulated and real-world setups on 2 traffic light recognition systems and 5 cameras, including one used on Tesla vehicles. The highest average success rates are 30% and 86.25% for Red-to-Green and Green-to-Red attacks. We built an attack equipment and demonstrated the feasibility of attacks in practical settings against a moving vehicle.

2 Background

2.1 Traffic Light Detection and Recognition

Traffic light (TL) recognition is generally performed after traffic light detection. They are differentiated by the target used when taking traffic lights as input [16]. Traffic light detection's target is to box out the traffic lights' positions in a given image. Traffic light recognition, usually taking the result of traffic light detection as input, will further distinguish the traffic light's color. Usually, the autonomous vehicle system will combine the detection and recognition step to provide instructions to the vehicle. We briefly introduce a few state-of-the-art methods and open-source systems.

Apollo [1] is an open autonomous driving platform developed by Baidu. It uses YOLO [24] to detect the traffic lights and treats the recognition as a typical CNN classification task. The RetinaNet receives an image with a list of bounding boxes with traffic lights as inputs and outputs a list with four elements, representing four possibilities for each box to be black, red, green, and yellow. Within red, yellow, and green, the class with the highest possibility will be regarded as the light's status when the possibility is over a certain threshold. Otherwise, the light's status will be set to black, meaning uncertain status.

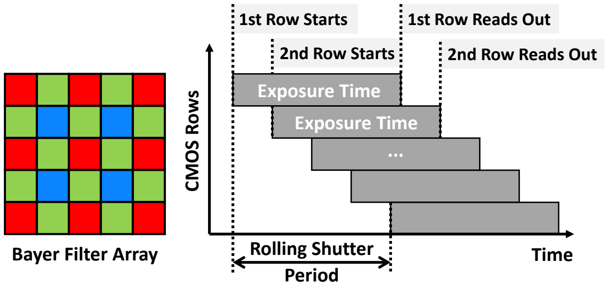


Figure 1: Bayer filter array and rolling shutter.

Other than separating the detection and recognition, there is another approach to combine these steps and treat traffic lights with different colors as different objects when training a classification model such as DNN. Therefore, systems [6] developed in such an approach will directly detect and classify red/yellow/green/black traffic lights in one step.

2.2 Camera Imaging

A camera is an optical system that can capture images. Basically, a camera uses a lens to gather light and a light-sensitive surface (usually photographic film or a digital sensor) placed on a focal plane to capture the image. A camera can control the luminous flux that reaches the surface by adjusting the aperture's size or exposure time. The latter corresponds to the amount of time the photosensitive surface is exposed to light, which is controlled by a shutter. Digital cameras use digital image sensors as the photosensitive surface that can transfer light into digital signals. There are two major types of electronic image sensors, i.e., charge-couple device (CCD) and complementary metal-oxide-semiconductor (CMOS) sensors. Cameras in consumer products generally use CMOS sensors as they are cheaper and have lower power consumption.

Image sensors capture the light intensity and can only create grayscale images. To acquire the color information, image sensors are typically overlaid with a "color filter array," which consists of tiny color filters that only allow the light within specific wavelength ranges to pass through. Thus, the image processing algorithm can restore the color information by checking each pixel's color filter arrangement. The most commonly used color filter array is called "Bayer Filter Array," which consists of alternating rows of red-green and green-blue filters, as shown in the left of Fig. 1.

Most digital cameras adjust the exposure time by controlling the "on-off" of the photodiodes on the image sensor,

Table 1: Attacker's goals, attack scenarios, and outcomes.

Goal	DoS	Color Spoofing	Outcome
Stop→Move	N/A	R→G	Collision
Move→Stop	Probable	G→R, G→Y, G→B	Trap

i.e., via electronic shutters. There are two types of electronic shutters: global shutter and rolling shutter. Global shutters enable the whole sensor to turn on and off simultaneously, while cameras with rolling shutters turn on and off the photodiodes row-by-row as if it is "rolling." Therefore, cameras with rolling shutters do not capture the entire scene at a single time instant. Due to the readout bottleneck of CMOS image sensors, most of them use rolling shutters. Fig. 1 demonstrates the timing diagram of a simplified rolling shutter sensor. Rows of pixels start exposure successively and get read out when the row exceeds exposure time. Meanwhile, the pixels are cleared for the next exposure. Since image rows are captured at different time instances, taking a picture of rapidly moving objects or flashing light can cause image distortion and skewness. We find that most cameras on vehicles (e.g., Tesla) use rolling shutters. Apollo Auto recommends the LI-USB30-AZ023WDRB camera, which is also a rolling shutter sensor.

3 Threat Model

3.1 Attacker's Goal and Attack Scenarios

The attacker's goal is to make self-driving vehicles make incorrect decisions near traffic lights by fooling the traffic light recognition system. In particular, there may be two scenarios: *1) Make a vehicle move while it should stop:* a vehicle should stop at red or yellow lights. Failing to stop accordingly may cause intersection accidents. *2) Make a vehicle stop while it should move:* a vehicle should move at green lights when there is no obstacle. Such a case may trap the vehicle and even jam the traffic.

We envision that denial-of-service attacks and various types of color spoofing attacks against the traffic light detection and recognition systems may cause the above cases. Table 1 summarizes the potential attack scenarios.

Denial-of-Service (DoS) Attacks. DoS attacks aim to disable traffic light detection, i.e., making a vehicle fail to detect existing traffic lights. To fail safe, a vehicle is supposed to slow down or stop at an intersection when no traffic light is detected. However, a vehicle may easily notice from the map if a traffic light is missing and take safety precautions.

Color Spoofing Attacks. Such attacks aim to alter the results of traffic light recognition, i.e., causing the traffic light's color status to be interpreted incorrectly. A traffic light has four color statuses: red (R), green (G), yellow (Y), and black (B), where black means that the light is off. We consider the most common traffic light status to be either red or green because black is rare in real life and yellow only lasts for a very short time. An attacker may make a vehicle move by making a red light recognized as green, and force a vehicle to stop by making a green light recognized as red, yellow, or black. In this paper, we focus on color spoofing attacks as they are stealthier and pose a more significant threat.

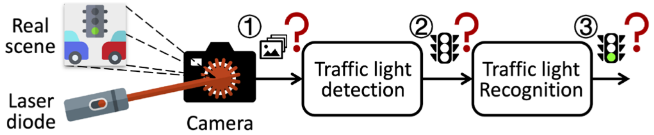


Figure 2: Illustration of the three attack stages.

3.2 Attacker's Capabilities

We consider an attacker with the following capabilities.

Laser Interference to the On-board Camera. The attacker can only alter the results of traffic light detection and recognition by remotely shedding a laser interference to the camera on the target vehicle. She cannot physically alter existing traffic lights or build a spurious traffic light.

No Direct Access to the Target Vehicle. The attacker has no direct physical or digital access to the target vehicle before or during the attack. She cannot make any changes to the on-board camera or the traffic light detection and recognition system. The target vehicle remains enclosed to the attacker.

Camera and System Awareness. The attacker may infer the model and structure of the target camera and system based on public information or reverse engineering. She may obtain the same camera or vehicle to assess the attack beforehand in setups similar to the planned attack scenario. For example, she could drive a vehicle of the same model through the target intersection and record images of the target traffic light. However, she can only treat the target camera as a black box.

4 Feasibility Study

In this section, we study the feasibility of attacking vision-based traffic light detection and recognition mechanisms using laser interference. We consider the three attack stages shown in Fig. 2.

1. Camera Imaging: the attacker generates a laser beam and points it at the camera. Key questions in this stage include a) what types of laser interference an attacker can generate and b) what images will a camera capture under laser interference.

2. Traffic Light Detection: disabling traffic light detection is sufficient for a DoS attack. However, for color spoofing attacks, the attacker must ensure that the traffic light can be detected with sufficiently high confidence.

3. Traffic Light Recognition: in this stage, the attacker intends to make a traffic light recognized as showing a different color. However, it is unknown whether this attack is feasible and what type of laser interference will be effective.

We examine the feasibility of these stages in the following.

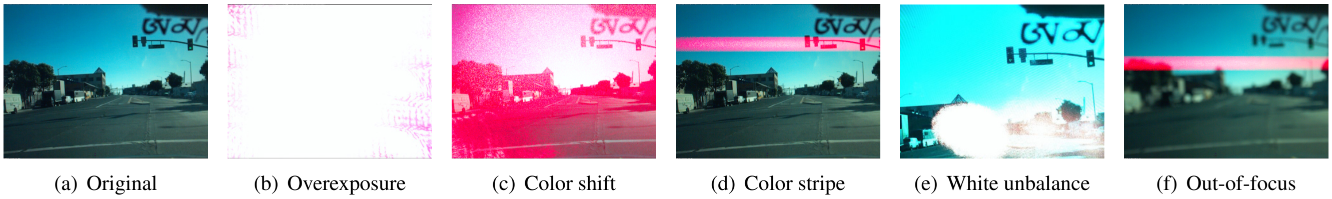

(a) Original (b) Overexposure (c) Color shift (d) Color stripe (e) White unbalance (f) Out-of-focus

Figure 3: Images captured by an AR0132AT evaluation board camera under various types of laser interference.

4.1 Laser Interference Examination

We choose semiconductor laser diodes as the laser generator due to their low-cost, high accessibility, and high power-efficiency. There are four attack parameters to consider that determine the laser interference pattern on the image:

Wavelength. A laser's wavelength corresponds to its color, which directly affects the color of the resultant interference on the image. A laser diode's wavelength is generally fixed as it is determined by the semiconductor's structure and material. Therefore, an attacker can inject interference of various colors using a series of laser diodes with different wavelengths.

Power. A laser's power affects the interference's intensity as well as the attack distance. A higher power enables a longer attack distance, but it may also saturate the CMOS sensor and blind the camera. In a color spoofing attack, the attacker has to carefully adjust the laser's power to ensure that the interference on the image is neither too weak nor too strong. Previous work [30] has shown that a laser's power increases linearly with the driving current once it is above a diode-specific threshold.

Pulse Width and Period. The laser can be continuous or in the form of a series of short pulses. A continuous laser affects all image frames in a video, while laser pulses can affect selected frames in a video and even a part of an image. An attacker can change the width and period of laser pulses by applying pulse width modulation (PWM) to the laser diode's driving current.

Incidence angle. The incidence angle is the angle between the laser beam and the optical axis of the camera lens. The incidence angle mainly affects the brightness distribution of the interference. For example, a laser emitted from the right side of the camera will cause a brighter interference on the right part of the image.

4.2 Camera Imaging under Laser Interference

Existing studies [11,23,33,37] have shown that light and laser interference can significantly change the images captured by cameras. However, their work focused on DoS attacks, i.e., overexposing the camera, and cannot be used for color spoofing. In this paper, we thoroughly examine and report all potential impacts of laser interference by experimenting with the laser parameters mentioned above. Besides overexposure, we discover that laser interference can cause the color shift, color stripe, white unbalance, and out-of-focus of an image, as shown in Fig. 3. The images are taken by an AR0132AT evaluation board [27] camera.

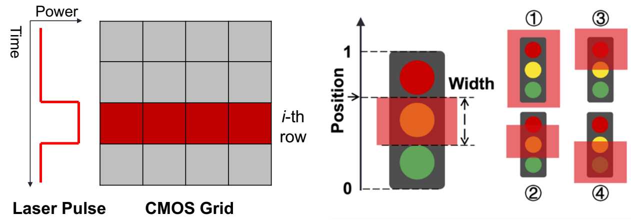
Overexposure. A laser of sufficient power can easily saturate all three color channels of a CMOS sensor and cause overexposure of the image, as shown in Fig. 3(b). Overexposure is the most effective way to cause the denial-of-service of any camera-based system.

Color Shift. By lowering the laser's power below the saturation point, we can reduce the interference's brightness to a range that only affects a single color channel, causing a "color shift" of the image, as shown in Fig. 3(c). The shifted color is determined by the laser's wavelength. In this example, a laser of 650 nm wavelength increases the red channel's value much more than the green and blue channels in the CMOS, making the image more reddish.

Color Stripe. The color shift appears on the entire image and may unintentionally impede the recognition of other objects. We wonder whether the color shift can occur only in a part of the image, especially where the traffic lights are. We manage to create a "color stripe" on the image by exploiting the rolling shutter of CMOS sensors with modulated laser pulses, as shown in Fig. 3(d). The principle is illustrated in Fig. 4(a). As the shutter "rolls" from the first row of the image to the last one, a laser pulse that appears only when the i-th row of CMOS is activated will interfere with only the i-th row on the image, thus leaving a color stripe. An attacker can change the width, position, brightness, and number of the stripes by adjusting the modulation parameters of the diode's driving signal. She could even stabilize the color stripe in a sequence of video frames by synchronizing the pulse period with the rolling shutter speed (video frame rate), e.g., 30 Hz.

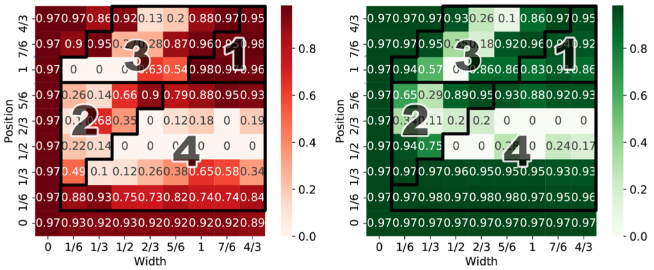
White Unbalance. White balance is an in-built function in most digital cameras that automatically balances the image's color temperature. A laser interference on a part of an image can make the white balance algorithm wrongly adjust the color gains of the rest of the image, as shown in Fig. 3(e). An attacker may exploit this phenomenon to cause "white unbalance" of the traffic light, i.e., changing its color without overlaying interference on it. However, the color change may not be intense due to the limited white balance ability.

Out-of-Focus. We discover that moving color stripes can make the camera out-of-focus, potentially leading to denial-of-service. Rapidly moving color stripes can periodically change the contrast of the sampling area of the focus algorithm, and



(a) Generation of a color stripe (b) Stripe width and position

Figure 4: Illustrations of (a) the generation of a color stripe, and (b) the stripe's width and position compared with the traffic light and four typical cases.



(a) Red laser (650 nm) (b) Green laser (520 nm)

Figure 5: The confidence scores of detecting a traffic light covered by color stripes of various widths and positions. The 4 cropped regions correspond to the 4 cases in Fig. 4(b). It shows that case ① affects traffic light detection the least.

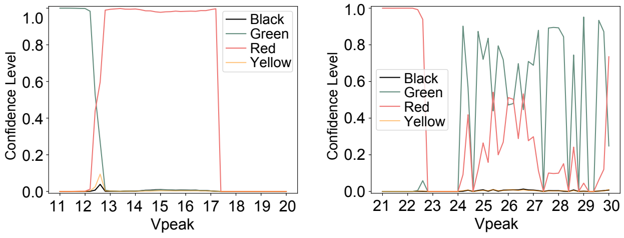
hence force the camera to refocus. During the refocusing process, the background will be out-of-focus and become blurry, as shown in Fig. 3(f).

In summary, an attacker may achieve denial-of-service with overexposed or out-of-focus images and achieve color spoofing with color-shifted or color-striped images. A color-shifted image is essentially a color-striped image when the stripe is as wide as the image. In a preliminary experiment reported in Fig. 30, we find that compared with a narrow stripe, a wider stripe will greatly reduce the confidence score of traffic light detection and unnecessarily causing denial-of-service. Thus, we will focus on attacks using a narrow stripe in this paper.

4.3 TL Detection under Laser Interference

To successfully spoof the results of traffic light recognition, the attacker has to first ensure that the color stripes do not affect traffic light detection. Thus, a question of interest is what type of color stripe less impedes traffic light detection.

We examine the impact of color stripes by comparing the confidence scores of detecting a traffic light under various combinations of stripe parameters, where we are mainly concerned with the stripe's width and position. We use a trained YOLOv4 classifier as the traffic light detection system and use red and green lasers to inject color stripes of various widths and positions into a traffic light image.



(a) Turn a green light to red light (b) Turn a red light to green light

Figure 6: Confidence scores of traffic light recognition as the laser diode's driving voltage (i.e., power) varies. It shows that though the effective laser parameters may exist in a wide range, they need to be adjusted in a case-by-case manner.

Two heatmaps in Fig. 5 show the confidence scores of detecting the traffic light under red and green laser interference. The x-axis is the stripe's width, which varies from 0 to 4/3 of the traffic light's height. The y-axis indicates the position of the stripe's upper edge compared with the vertical size of the traffic light, also varying from 0 to 4/3. Fig. 4(b) provides a visual illustration of the position and width. Position "0" means that the stripe's upper edge is at the bottom edge of the light, while position "1" suggests that the stripe's upper edge coincides with the light's upper edge. To help understand the results in Fig. 5, we divide the heatmaps into four regions that correspond to the four typical cases in Fig. 4(b). In case ①, the stripe covers the entire traffic light; in case ②, the stripe covers only a part of the light between its upper and lower edges; while in case ③ and ④, the stripe covers the upper or lower parts of the light including its edges.

The results in Fig. 5 show that, regardless of the laser's color, the accuracy of traffic light detection is barely affected by the color stripe if it covers the entire light, i.e., case ①. We observe a great decrease in the confidence scores in other cases, which we assume is because the color stripe breaks the traffic light's continuity, especially on its original edges. This experiment motivates us to create color stripes similar to case ① for color spoofing attacks.

4.4 TL Recognition under Laser Interference

To examine the feasibility of spoofing traffic light recognition, we consider two attack cases: (a) turn a green light to red with a red stripe, and (b) turn a red light to green with a green stripe. The stripes are generated using 650 nm red and 520 nm green laser diodes, respectively. The images of the interfered traffic lights are sent to Apollo for recognition.

In Apollo, the system is programmed to first detect traffic lights within a cropped region of interest (ROI) that is previously known from the HD-Map that contains traffic lights' locations. Since the HD-Map is unavailable, we provide the ROI to the system manually to make Apollo function normally. If a traffic light is detected, Apollo outputs the confidence

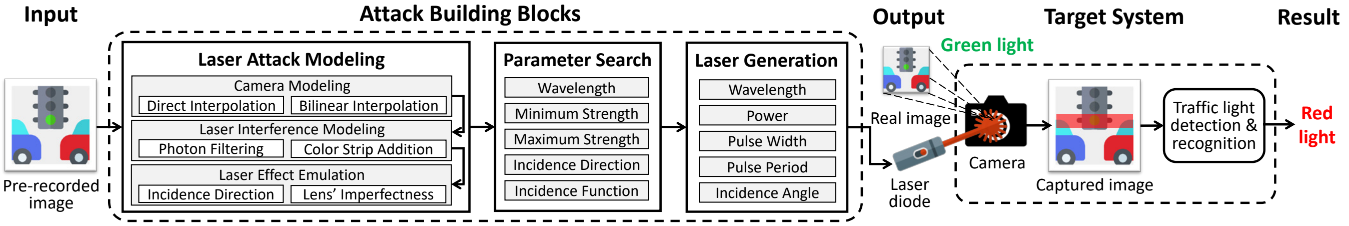


Figure 7: The attack workflow. Based on laser attack modeling, the attacker first simulates laser interference on pre-recorded images of the target traffic and search for the most effective sets of laser attack parameters that can achieve the desired result of traffic light recognition. The attacker then uses the parameters to generate a laser attack signal and point it at the target camera.

scores of four possibilities: red, yellow, green, and black.

The results show that color spoofing attacks are feasible for both cases. We are able to make a red light recognized as a green one and vice versa. However, we also observe that the attack does not succeed all the time—the confidence scores can be affected greatly by the laser parameters, especially the power. For example, Fig. 6 shows the scores of the recognition result when we adjust the laser's power by the diode's driving voltage. It shows a red laser can spoof a green light with high confidence scores when the voltage is between 13 to 17 volts. However, there is no "safe zone" of power when we try to make a red light recognized as green. The effective voltage region is discrete and appears random. In our experiment, we find that the attack result can also be affected greatly by the laser's incidence angle, the traffic light itself, and even the image background. Therefore, it would be inadvisable to use the same set of laser parameters to attack different traffic lights. In practice, it is critical to fine-tune the laser parameters according to specific traffic lights and attack scenarios.

5 Attack Design

To obtain the most effective laser parameters case-by-case, we model the laser attack process and emulate the attack based on pre-recorded images of the target traffic light. The attack workflow is summarized in Fig. 7.

First, to simulate laser interference on cameras, we develop an empirical laser attack model that takes a raw image as input and outputs an image with simulated laser interference. The model consists of three parts: camera modeling, laser interference modeling, and laser effect emulation. Camera modeling describes the transformation from the RAW images captured by the CMOS to common image files such as JPG. Laser modeling simulates the laser interference by mathematically modeling the addition of color stripes to an image. Laser effect emulation further involves a laser's incidence direction and noise to the model to make the simulated color stripe look more realistic. Then based on the laser attack model, we use a grid search method to seek the most effective combinations of attack parameters. Finally, we map the attack parameters derived from our model to adjustable laser parameters and generate laser attack signals in the real world.

5.1 Camera Modeling

The first step is to transform the CMOS sensor's raw images to jpg images. As introduced in Section 2, CMOS, as well as the raw file, will store the color information of the image in the form of R, Gr, Gb, B in the four adjacent pixels respectively that form a square. Therefore, interpolation is required to transform the matrix from a raw file to a tensor that represents RGB values. There are mainly two types of interpolation: direct and bilinear. We consider both of them in our model, and the specific model to use depends on the target camera. We leave the mathematical details to Appendix A.1.

Direct Interpolation. For each pixel in the jpg image, its RGB values are determined by the closest pixel that stores Red, the mixture of Gr and Gb, or Blue in the matrix from the raw file. As for each two-by-two block in the raw file, there will be at least one pixel representing "red," one representing "blue," and two representing "Gr" and "Gb," respectively, the RGB value of the jpg-image-pixels in this block will be directly determined by those pixels values in the raw file.

Bilinear Interpolation. For each pixel in the jpg image, its RGB value is determined by at least two closest pixels in the raw file. In general, for each interior pixel in the jpg file, it will take a three-by-three block centered at the pixel. All the pixel values representing the corresponding color (R, G, or B) in the raw file are averaged to calculate the RGB value in the jpg file. For each pixel on the edge of the image where the three-by-three block is not available, we will still find the two closest pixels in the raw file representing the same color and calculate their average.

5.2 Laser Interference Modeling

We consider two steps in the laser interference modeling: photon filtering and color stripe addition.

Photon Filtering. When a laser is shed into a CMOS, the photons that each photodiode can detect are determined by its quantum efficiency, which describes the ratio of the number of electrons measured by a photodiode to the number of photons fed to it. Each CMOS has its unique quantum efficiency. Also, the color filters (R, Gr, Gb, B) on each photodiode make the quantum efficiency unique for the light of different wavelengths. For example, Fig. 8 shows the quantum

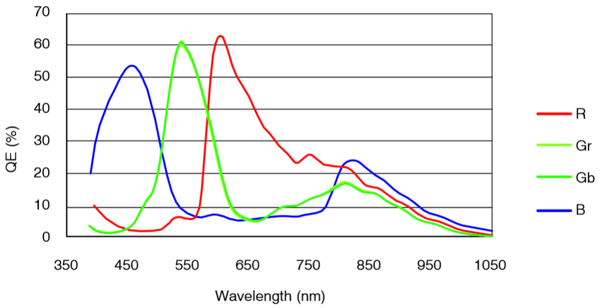


Figure 8: Quantum Efficiency curves for ON Semiconductor CMOS Digital Image Sensor MT9P006 [26].

efficiency curves for an ON Semiconductor CMOS Digital Image Sensor MT9P006. Green light of 550 nm wavelength can be measured by the Gr and Gb photodiodes with a quantum efficiency of 60%, while it is less received by the R and B photodiodes because the quantum efficiency is less than 10%. However, when the light strength is sufficiently high, its effect on the red and blue pixels becomes obvious, which explains why in Fig. 3(b), a single-colored laser can saturate all color channels and make the entire image white.

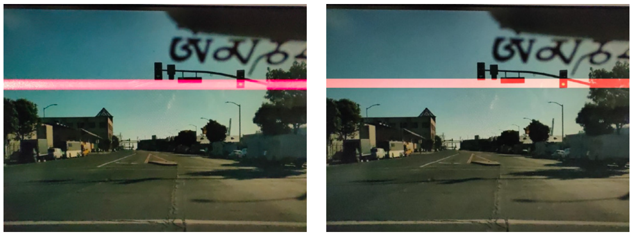
Suppose I stands for the laser's strength, λ stands for its wavelength, and R^*, G^*, B^* stands for the effect of the laser on the Red, Green, Blue channels of the image, respectively, the filtering effect of the CMOS sensor due to quantum efficiency could be represented by $X^* = I f_X(\lambda), X \in \{R, G, B\}$. Here, $f_R(\cdot), f_G(\cdot), f_B(\cdot)$ stand for the quantum efficiency curves of the CMOS sensor on the target camera. Since the curves could not be expressed parametrically, we interpolate the functions by tracing the quantum efficiency curves of the target camera.

Color Stripe Addition. Suppose the effect a laser interference has on an RGB image is the addition on the three color channels, we can express the RGB values of the post-image (image taken under the laser attack), R_p, G_p, B_p, by the RGB values of the original image, R_o, G_o, B_o, and the effect of the laser, R^*, G^*, B^* as $X_p = H(X_o, X^*), X \in \{R, G, B\}$. Here, there is no specific choice for function $H(\cdot, \cdot)$. Considering the color channel overflow, here we use a relu-like function for $H(\cdot, \cdot)$: $H(x, y) = \min(x + y, c_{max})$. Here, c_{max} stands for the overflow value for a specific channel, and we choose 255 for simplicity.

Combining the photon filtering and color stripe addition processes, the effect of the laser interference is determined by the RGB values of the original image X_o, the wavelength λ and strength I of the laser, and the quantum efficiency $f_X(\cdot)$ of the CMOS sensor as:

$$X_p = H(X_o, I f_X(\lambda)) \qquad X \in \{R, G, B\}$$

Fig. 9 shows a comparison between the real laser interference and our simulation. Notice that our simulated laser interference is highly similar to the real one, indicating our model's efficacy. However, there are slight differences in the details, which we assume are caused by the laser's incidence direction and noise.



(a) Real laser interference

(b) Simulated laser interference

Figure 9: Comparison between the real laser interference and our simulation, showing a high similarity.

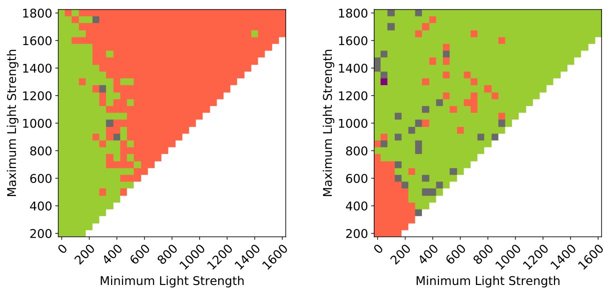
5.3 Laser Effect Emulation

To make the simulation more realistic, we include the laser's incidence direction and lens imperfectness into our model. The former determines the color stripe's brightness distribution on the image, and the latter produces noise.

Laser's Incidence Direction. The laser's incidence direction has a significant effect on the color stripe's distribution in the image. For example, in Fig. 9(a), the laser is irradiated from the left side of the camera. Therefore, the laser's intensity is higher on the left than on the right, which results in a channel overflow (white pixels) on the left and an increased level of noise. For simplicity, in our model, we consider the laser from left, right, or front of the camera. Instead of using a single intensity I for the laser, we define the minimum and maximum intensities I_{min}, I_{max} measured by the CMOS, and assume the width and height of the color stripe are w, h, respectively. The intensity of the laser at a specific point (x, y) can be represented as $I = D(I_{min}, I_{max}, x, y, h)$. Empirically, we consider three potential cases for the function $D(\cdot)$: linear, sigmoid, and Gaussian. Linear or sigmoid functions are used when the incidence direction is from the sides; the Gaussian function is used when the light is irradiated from the camera's front. We leave the details to Appendix A.2.

Lens Imperfectness. Empirically, we observe snow-flake-like noise in the color stripe, which is due to the abrasion of the lens. Therefore, if the lens has abrasion, when the laser is shed into the camera, the diffuse reflection will occur instead of direct reflection, which will result in some areas of the CMOS receiving a stronger laser beam that may cause overexpose at the corresponding pixels in the image. As it is hard to measure or accurately model the abrasion inside the lens, we assume the horizontal distribution of the noise follows an exponential distribution, and the vertical distribution of the noise follows a normal distribution. For laser irradiated from the left, we model the horizontal distribution of the noise as:

$$h(y) = \frac{e^{-\beta_1 y/w}}{\int_0^\infty e^{-\beta_1 t/w} dt}$$



(a) Red laser (wide stripe)

(b) Green laser (wide stripe)

Figure 10: Refined grid search of light strength for red and green lasers that create wide stripes. The feasible solutions are red and green pixels, respectively.

Here, β_1 is the hyper-parameter that controls the decaying rate of the distribution, and the denominator is a standardized function to ensure the sum of the probability density function is 1. If the incidence direction is right, then we only need to replace y by $w-y$.

The vertical distribution of the noise follows:

$$v(x) = Norm\left(\mu = x_0 + \frac{h}{2}, \sigma = \frac{h}{\beta_2}\right)$$

Here, β_2 is the hyper-parameter that controls the distribution's sparsity. The vertical distribution is identical for both incidence directions.

Furthermore, as the noise does not always result in overflow of the target color channel, we randomly choose a value between 240 and 255 to replace the channel's value. The procedure of adding noise is detailed in Appendix A.2. The effect of incidence direction and noise emulation is shown in Fig. 20, which highly resembles the real scenario in Fig. 9(a).

5.4 Searching Laser Attack Parameters

Based on the laser attack modeling, we can simulate a large number of color stripes added on a target traffic light using various combinations of laser parameters. To acquire the most effective range of parameter sets, we apply the grid-search method. We do not use the optimization methods that are common in generating adversarial examples because our attack can succeed with a range of laser parameters instead of a particular set. This property may mitigate the influence of model inaccuracy and real-world uncertainties, therefore increasing the attack's robustness. The parameters we consider in the grid-search include the minimum light strength I_{min}, maximum light strength I_{max}, wavelength of the laser λ, and incidence direction. The noise function follows the incidence direction. The search grids are reported in Appendix A.3.

We tested the parameter search on red and green traffic light images. The search results show a range of effective laser attack parameters (**feasible solutions**). 9.63% and 22.59% of all parameter combinations are effective for red laser attack and green laser attack, respectively, which proves that the feasible solutions of our attack are not unique. In addition, we conduct a more thorough search on the minimum and maximum light strength using a more fine-grained search grid. Fig. 10 shows the results for red and green stripes with various combinations of minimum and maximum light strengths. In the figures, the recognition results of red, yellow, green, black, and DoS (failed to detect the light) are represented by red, yellow, green, dark gray, and purple pixels, respectively. The results show that feasible solutions for light strength exist in a wide range and can be regarded as nearly continuous.

5.5 Generating Laser Attack Signals

After acquiring the feasible attack parameters, we map them to adjustable laser parameters to recurrent the model-generated images with real laser. We mainly consider the wavelength, pulse width and period, power, and incidence angle.

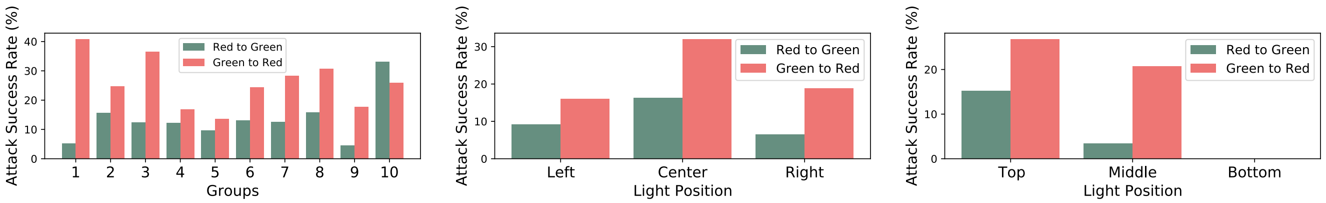
Wavelength. The relationship between wavelength in the model, λ, and reality is explicit. The wavelength of the laser diode is the same as the wavelength in the model.

Pulse Width and Period. The pulse width determines the width of the color stripe in the image. Suppose the image has N rows of pixels and the traffic light takes n rows. Our previous observations in the real-world experiment and emulation both suggest that the color stripe affects traffic light detection the least if its width is larger than that of the light. Therefore, we consider a color stripe with a width of $1.5n$ rows. Suppose the rolling shutter speed is f, then the pulse width is $1.5n/Nf$. The pulse period is set the same as the rolling shutter time $1/f$ to stabilize the stripe's position.

Power. Power is related to the minimum and maximum light strength, I_{min}, I_{max} in the model. The minimum/maximum light strength is a constructed parameter in the model which does not have direct physical meaning. Though there is no deterministic transformation between light strength and power, as they both change monotonically, the attacker could measure the light strength with various laser power beforehand in a similar setup and fit the curve.

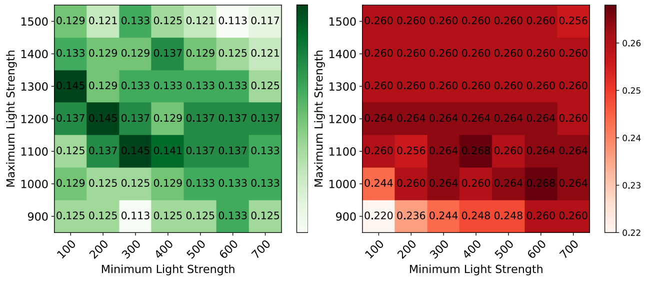
Incidence Angle. The incidence angle is determined by the incidence direction and the relationship between the minimum and maximum light strength. The incidence direction provides a definitive guide on where the laser shall be radiated from. However, it is hard to acquire the incidence angle from the min./max. light strength mathematically because their relationship varies case-by-case for various cameras. Similarly, the attacker can derive their relationship empirically.

Despite the challenges in obtaining accurate mappings between model parameters and laser parameters, the wide range of feasible solutions for our attack can alleviate this problem, as we will show in the next section. The necessity of the above design in comparison with a naive attacker that subjectively adjusts the attack parameters is discussed in Appendix A.4.



(a) Impact of the traffic light's size (b) Impact of the traffic light's horizontal position (c) Impact of the traffic light's vertical position

Figure 11: Impact of the traffic light's size and position on the attack's success rates.



(a) R→G attacks (b) G→R attacks

Figure 12: The attack's average success rates on all images using various combinations of min. and max. light strengths.

6 Evaluation

We evaluate the attack's performance in 3 scenarios: (a) emulated attacks based on public traffic light images and simulated laser interference, (b) real-world attacks in stationary setups using real laser and cameras, and (c) real-world attacks against a moving vehicle in practical settings.

6.1 Emulated Attacks

6.1.1 Dataset and Experimental Setup

We select 240 images that contain traffic lights from the BDD100K dataset [38]. We target all traffic lights in each image that can be correctly detected and recognized by Apollo. In total, there are 250 green lights and 248 red lights. The hyper-parameters of the model are determined as follows. For all green lights, we use the red laser with a wavelength of 650 nm; for all red lights, we use the green laser with a wavelength of 520 nm. The laser wavelengths correspond to the laser diodes in real-world experiments. Based on our previous observations, we adopt an incidence direction opposite to the light's position in the image to avoid over-exposure in the light's area and use the linear function to model the intensity distribution. For the minimum and maximum light strength, we applied a grid search for each image. For each variable, we set 7 levels in total: the minimum light strength varies from 100 to 700 with a step length of 100, and the maximum light strength varies from 900 to 1500 with a step length of 100.

6.1.2 Overall Performance

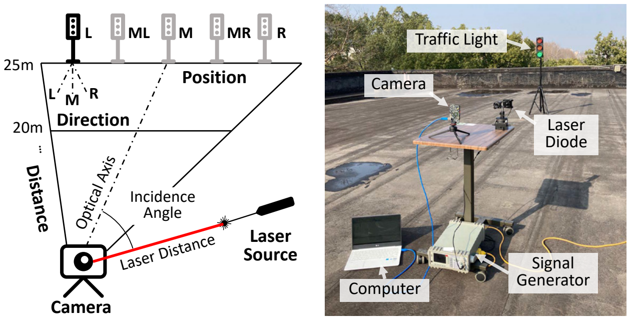
By using the optimal set of attack parameters for each traffic light, we are able to make a red light recognized as a green light (R→G) in 45 cases, i.e., with an attack success rate of 18.2%, and make a green light recognized as a red light (G→R) in 86 cases, with an attack success rate of 34.4%. In addition, we find that the attack can succeed on different images even using the same set of parameters. For example, Fig. 12 shows that R→G is successful on 14.5% of all images when the min. and max. light strengths are (100,1300), (200,1200), or (300,1100), and G→R is successful on 26.8% of all images when the min. and max. light strengths are (400,1100) or (600,1000). These results show that our attack is feasible on various traffic lights and does not require unique parameters as feasible solutions.

6.1.3 Impact of the Traffic Lights

Our dataset contains 489 traffic lights of various sizes and positions in the image. In the following, we investigate whether the traffic light can affect the attack's success rate.

Impact of Size. To explore the effect of traffic light's size, we sort all traffic lights into 10 groups by their height in the image (in pixels), and calculate the attack success rate for each group. Group 1 contains the smallest (0%–10% of all heights) lights, while group 10 represents the largest (90%–100%). The result is shown in Fig. 11(a). We observe that for the G→R cases, the attack success rates are comparatively higher for relatively smaller or larger traffic lights. However, for R→G cases, there is no clear relationship. Overall, the correlation between the traffic light's size and the attack's success rate is not significant.

Impact of Position. We consider both the horizontal and vertical positions of the traffic light in the image. We divide an image into three-by-three blocks: left, center, and right horizontally; top, middle, and bottom vertically. All blocks have the same size. We plot the attack's success rates for the horizontal and vertical divisions, respectively, in Fig. 11(b) and 11(c). Horizontally, the attack's success rates for both G→R and R→G cases are higher when the traffic light is in the center of the image than at the left or right. Vertically, the attack's success rate appears to be higher if the traffic light is at a higher position in the image. Since no image contains



(a) Illustration of setup

(b) Real setup

Figure 13: Illustration and image of the experiment setup.

a traffic light in the bottom blocks, the attack's success rate is not available. In summary, we assume that such impacts of traffic light's position are mainly due to the background. In most cases, the background will be clearer at the top and center positions of the image, which typically is the sky. In contrast, in other positions of the image, the background is usually buildings or traffic that are more complex.

6.2 Real-World Attacks in Stationary Setups

We validate the attack's effectiveness in stationary setups using real cameras, laser diodes, and traffic lights at various distances, positions, and directions.

6.2.1 Experimental Setup

The experimental setup is shown in Fig. 13. The attack target is an experimental traffic light that can shine in green, red, and yellow. The diameter of the lamps is 10 cm[1]. We experiment on 5 cameras: an AR0132AT evaluation board camera [27], two dashcams from Xiaomi [34] and Hikvision [8], an OPPO K3 smartphone [22], and an OpenMV H7 Devboard camera [21]. The AR0132AT sensor is considered to be the camera used on Tesla vehicles [4, 32]. We attack the cameras using two laser diodes (505 nm for green laser, 650 nm for red laser) driven and controlled by a signal generator. The images of the traffic light under laser interference is sent to two traffic light detection and recognition systems: Apollo [1] and a deep-learning-based model developed for the Nexar traffic light recognition challenge [6], which are the most popular and state-of-the-art open-source models we can find for traffic light recognition. To increase the variety of traffic light instances, we conduct the attack in various combinations of the traffic light's distance, position, and facing direction, as illustrated in Fig. 13(a). The traffic light's distance to the camera in the optical axis varies from 5 m to 15 m. At each distance, the traffic light is placed in 5 positions, left (L), middle left (ML), middle (M), middle right (MR), and right (R). On each position, we set the light to face three directions: left (L), middle (M), and right (R). In total, there are 120 instances of traffic lights for a single color. We repeat the 120 instances for different light colors and cameras. For safety reasons, all the experiments are conducted on a rooftop in an enclosed environment with proper laser protection.

Table 2: Success rates of attacking 2 systems and 5 cameras.

Sys.	Attack Scenario	Target Camera					Avg.
		Tesla	Xiaomi	Hikv	OPPO	OpMV	
Apollo	R→G	31.67%	32.47%	15.38%	4.44%	17.39%	20.27%
	R→DoS	21.67%	40.26%	15.38%	73.33%	13.04%	32.74%
	G→R	73.33%	17.24%	39.13%	32.22%	47.37%	41.86%
	G→DoS	21.67%	32.18%	39.13%	36.67%	52.63%	36.46%
Nexar	R→G	28.33%	14.77%	1.23%	59.55%	0%	20.78%
	R→DoS	12.50%	57.95%	38.27%	31.46%	100%	48.04%
	G→R	99.17%	64.00%	21.31%	45.21%	100%	65.94%
	G→DoS	0.83%	30.67%	78.69%	53.42%	0%	32.72%

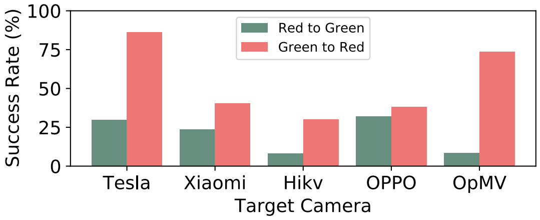


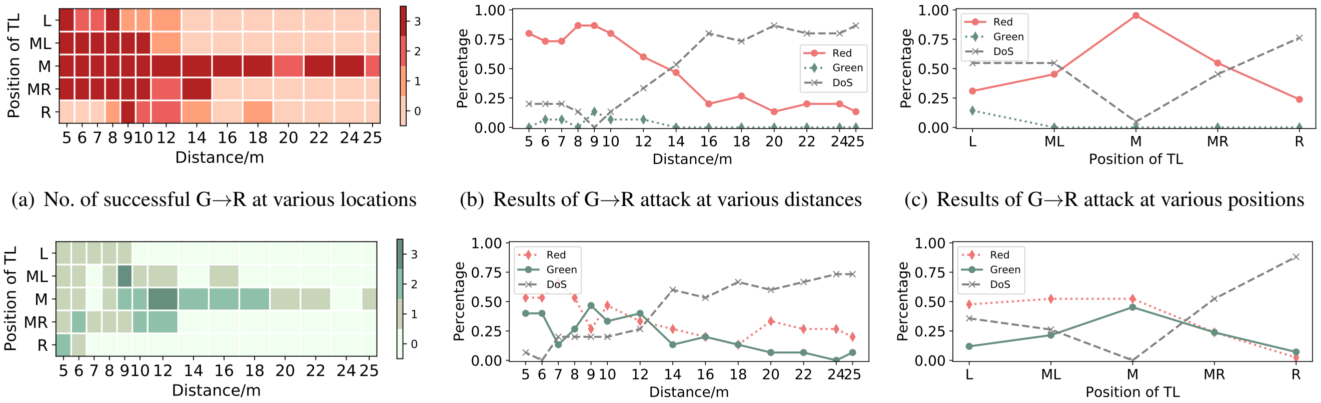
Figure 14: Average success rates of attacking 5 cameras.

6.2.2 Overall Performance

We report the attack's overall success rates in Table 2. The experiments are conducted in two attack scenarios: R→G and G→R. We also report the percentage of DoS cases. The overall success rate is 20.53% for R→G attacks and 44.95% for G→R attacks, suggesting that it is relatively easier to cause a green light to be recognized as a red light using a red laser. DoS happens 34.59% and 26.76% of the time for R→G and G→R attacks, respectively. Note that the above results involve all the instances. The average success rate is higher than 90% when the distance is less than 10 m (equivalent to 20 m for regular traffic lights, which is the width of a typical four-lane urban road).

In addition, we find that the attack result varies according to the target recognition system and camera. The average attack success rate on Apollo is 31.07%, while it is 43.36% on Nexar. The comparison of attacking 5 cameras in Fig. 14 shows that the AR0132AT camera used on Tesla is the most vulnerable—the average success rates for R→G and G→R are 30% and 86.25%. The lower success rates on other cameras may be caused by the low image qualities or large field-of-view as listed in Table 4. For example, Xiaomi and Hikvision are both dashcams that have a large field-of-view (130°) in comparison with the Tesla camera (70°). The wide angle

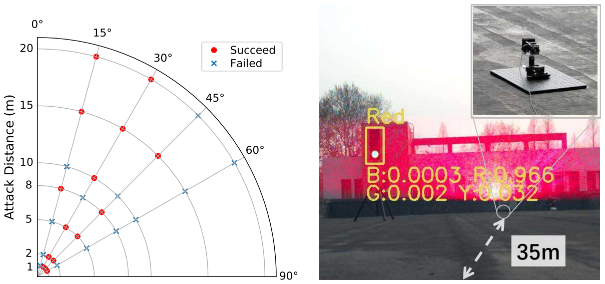
[1] The diameter of lamps on regular traffic lights is 20 cm. Therefore, the traffic light's distance to the camera that we report in this paper shall be doubled so that it is equivalent to the attacks against regular traffic lights.



(a) No. of successful G→R at various locations (b) Results of G→R attack at various distances (c) Results of G→R attack at various positions

(d) No. of successful R→G at various locations (e) Results of R→G attack at various distances (f) Results of R→G attack at various positions

Figure 15: Attack results showing impacts of the traffic light's distance and position based on the AR0132AT camera.



(a) Attack angles and distances (b) Attack from 35 m away

Figure 16: Results at various attack distances and angles.

makes the traffic light smaller in the image and thus more difficult to be recognized. As the traffic light becomes closer and larger, it moves toward the edge of the image and suffers from more wide-angle distortions than others. Nonetheless, G→R attacks show higher success rates than R→G regardless of the recognition system and camera.

6.2.3 Impact of the Traffic Light

We investigate the impacts of traffic light's distance, position, and direction based on the AR0132AT camera. In this experiment, we extend the maximum traffic light distance to 25 m. The number of successful attacks when the traffic light is at various locations is visualized in Fig. 15(a) and 15(d).

Impact of Distance. The results in Fig. 15(b) and 15(e) show that the attack success rate decreases as the traffic light's distance increases, while the percentage of DoS increases. We speculate that the traffic light becomes smaller as the distance increases, making it harder to be detected after being overlapped with a color stripe. However, we can still attack a traffic light that is 25 m away (equivalent to 50 m for regular traffic lights) if it is in the middle position.

Impact of Position. Fig. 15(c) and 15(f) show that the attack success rate is higher when the traffic light is in the middle of the image. DoS dominates when the traffic light is on the right part of the image because the laser is emitted from the right direction. As the light strength is not strong enough on the left part of the image, the traffic light is recognized as its original color in the left positions. The above observations are consistent with our emulation results.

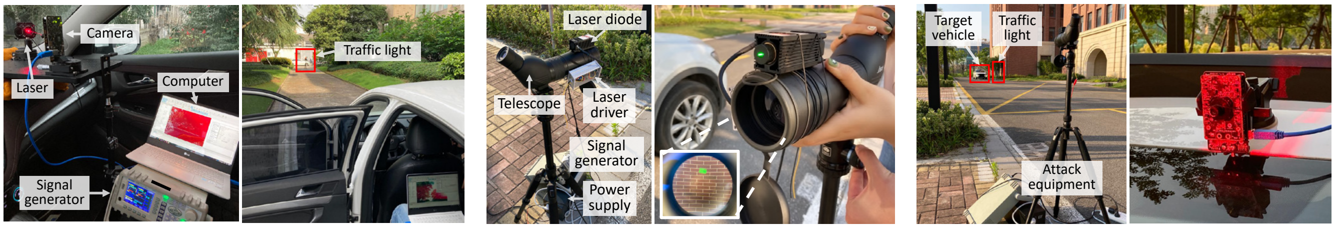
Impact of Direction. A successful attack requires the camera to have a clear view of the traffic light's lamp. Under this condition, we did not find a clear correlation between the attack result and the traffic light's facing direction.

6.2.4 Attack Distance and Angle

We investigate the attack's effectiveness when the laser source is at various distances and off-axis angles. We attack the camera from [1, 2, 5, 8, 10, 15, 20] meters away at [15, 30, 45, 60] degrees off the optical axis. The results are shown in Fig. 16(a). It shows that the attack is affected more by the laser's angle than the distance. Successful attacks appear mostly within 45 degrees off-axis from 1 m to 20 m. We assume the reason is that the color stripe becomes fainter as the incidence angle increases. To find the longest attack distance, we move the laser source to the outermost location allowed by our test site (a rooftop), which is approximately 35 m away from the camera. As shown in Fig. 16(b), the attack is still successful at such a distance—making a green light recognized as red with a confidence score of 0.966. We envision that the longest attack distance will be larger than 35 m, especially if we use a high-power laser diode.

6.3 Real-World Attacks in Motion

Attacks in practical settings require to target a moving vehicle and achieve the following: (1) be effective across continuous video frames when the captured scene is changing, (2) track and aim laser at the target camera from a distance away from the vehicle, and (3) cause end-to-end impact on driving. We study the attack's capability from the



(a) Setup for continuous video frame experiment (b) Attack equipment for tracking and laser aiming (c) Setup for long-range laser aiming experiment

Figure 17: Experimental setups for real-world attacks: (a) the setups inside and outside a vehicle for continuous video frame experiments, (b) the self-made equipment for manual target tracking and laser aiming, and (c) the setup for long-range laser aiming experiment with the attacker on the roadside and an AR0132AT camera mounted on the target vehicle's windshield.

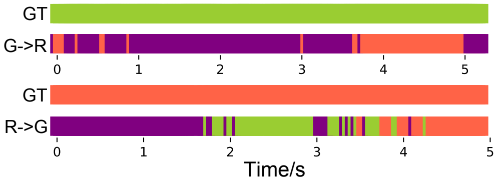


Figure 18: Attack results across continuous frames, showing continuous success for more than 1 second for both cases.

three perspectives separately on a real vehicle and a driving simulator. Videos of attacks can be found at `https://sites.google.com/view/rollingcolors`.

6.3.1 Effectiveness across Continuous Video Frames

First of all, we study the attack's effectiveness across continuous video frames when the target camera is on a moving vehicle. As a preliminary experiment, we rule out the influence of aiming and assume an ideal setting, where the attacker can *continuously inject laser into the camera* when the vehicle is moving. The feasibility of continuous laser injection will be studied separately.

Setup. To achieve this setting, we placed the laser diode inside the vehicle as shown in Fig. 17(a). We performed 27 attack trials in total. In each trial, we drove the vehicle backward or forward (with speeds between 5-20 km/h) towards an experimental traffic light 30 m ahead and recorded videos with the camera while launching the attack. For safety considerations, all experiments were performed in enclosed fields.

Results. We analyzed the attack results based on Apollo and found that the attack could successfully spoof traffic light recognition across continuous frames in 23 trials, rendering an attack success rate of 85.2%. Fig. 18 shows two best cases for G→R and R→G attacks, which achieve the longest duration of successful spoofing. The recognition results are represented in the corresponding colors (except for purple, which means Apollo fails to detect any traffic light), and "GT" represents the ground truth of the light color. The results show that the attack can continuously spoof traffic light recognition for more than one second. This preliminary experiment demonstrates the attack's effectiveness across continuous video frames when the captured scene is changing.

6.3.2 Feasibility of Tracking and Laser Aiming

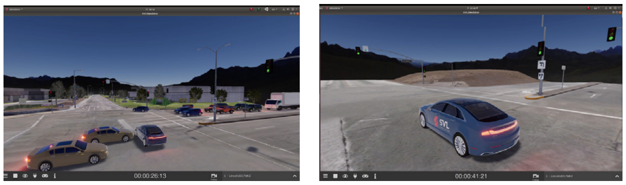
In the above experiment, we assume the attacker can continuously inject laser into the camera. However, in practice, the attacker and laser diode should be outside the target vehicle, which makes it challenging to continuously aim laser at the camera's tiny lens especially from far away.

Manual Tracking and Aiming Equipment. To overcome this challenge, we built a manual tracking and aiming equipment that combined a 500-1000 mW laser diode with a monocular telescope mounted on a tripod, as shown in Fig. 17(b). We used the high-power laser diode to create a larger speckle that eases aiming while ensuring the light injected into the camera has sufficient strength. We fixed the laser diode on top of the telescope in a position where the speckle is visible from the eye lens of the telescope. With this equipment, an attacker can manually track the target and aim the laser at the same time even from far away.

Setup. Based on the equipment, we conducted real-world experiments in a practical setup to examine the feasibility of continuous aiming. As shown in Fig. 17(c), the attacker was on the roadside and 40-80 m away from the vehicle. The target camera was mounted near the top of the vehicle's windshield. We drove the vehicle forward or backward towards a traffic light, while another researcher acted as the attacker and controlled the attack equipment on the roadside. All laser operations were performed in an enclosed field following safety regulations and coordinated with the safety committee[2].

Results. Out of the 88 trials conducted, we found the attacker can continuously inject laser into the camera in 77 trials, rendering an aiming success rate of 89.8%. The results show that manual aiming is feasible with the equipment we built even when the vehicle is moving at 20 km/h.

[2] According to the FDA, laser pointers with power more than 500 mW are Class-4 laser products that must avoid direct or scattered exposure to for eye and skin protection. Though the large speckle we used has greatly reduced the laser's power density at the target vehicle, eye damage may happen to the passengers when viewed directly for long periods of time. For example, given that the laser power is 1 W, speckle diameters at the diode and at 40 m are 0.8 cm and 15 cm, then based on a rough calculation, the laser's power density is equivalent to 2.9 mW, 11.4 mW, and 213.3 mW at 40 m, 20 m, and 10 m away respectively. Safety measures such as laser goggles must be ensured in all experiments, especially when in proximity to the diode.



(a) Running a red light

(b) Emergency stop

Figure 19: Simulated end-to-end impacts of the attack.

6.3.3 End-to-End Impact on Driving

We believe the experimental setup in Section 6.3.2 represents a practical setting for attacks in the real world. Thus, we further analyze the experiment's result on traffic light recognition and investigate the attack's potential end-to-end impact on Apollo's driving behavior with the LGSVL simulator.

Result of Attacking TL Recognition. Under the practical setup in Section 6.3.2, the attack can successfully spoof a red light to green in 16 trials (out of 38) and spoof a green light to red in 9 trials (out of 50), rendering an attack success rate of 42.1% for R→G and 18% for G→R. The average attack success rate is 28.4%, which is lower than the result in Section 6.3.1 due to the unstable laser injection caused by manual aiming from a distance. We envision that the attack success rate can be increased with better equipment, e.g., an automatic aiming and laser stabilization system.

Impact on Driving Behavior. As we could not find production self-driving vehicles, we study the attack's potential end-to-end impact on Apollo running in the LGSVL driving simulator. For simplicity, we replay the best attack results in the previous experiments as the color of the virtual traffic light in the simulator. We find the vehicle will start to brake or move in less than 0.5 seconds after recognizing a sudden change of the traffic light. In some cases such as Fig. 19, such behaviors can make the vehicle run a red light and collide with the traffic, or suddenly decelerate and stop in the middle of the intersection, potentially causing rear-end collisions.

7 Discussion

Universal Black-box Attack. An universal black-box attack is desirable as it does not require any knowledge of the target camera. Though our attack does not require a full white-box knowledge of the camera, knowing the refresh rate is necessary for attacking all cameras in order to stabilize the injected interference. As the refresh rate varies between devices, it is difficult to achieve an universal black-box attack that uses the same attack parameters for different target cameras.

Timing Challenge. The timing of laser injection determines the stripe's position in the image and directly affects the attack's effectiveness. Unfortunately, it is impossible to predict the exact timing without direct feedback from the camera, which can lower the attack's effectiveness in practice. We consider the influence of timing as a probability problem. A wider stripe will increase the probability of overlapping with the traffic light, however, it may also unnecessarily disrupt traffic light detection and the recognition of other objects in the image, such as cars and lanes, which may trigger alarms and disable self-driving. The precise effect of a wide stripe on object detection is unknown and shall be investigated in future work. The attacker may also inject a stripe that slowly moves in the image across consecutive frames. Therefore, the attack may succeed during the time when the stripe overlaps with the traffic light, e.g., for a few seconds depending on the stripe's moving speed, which is sufficient to affect self-driving according to our simulation.

Countermeasures. To mitigate the threat, we suggest two methods to redesign the rolling shutter mechanism. The electronic shutter may (1) expose the CMOS rows in a random sequence, or (2) start the exposure from a random row for every frame, as illustrated in Fig. 31. We simulate the first method, and Fig. 32 shows the images before and after applying random rolling shutter sequence. It shows that this method will scatter the injected stripe over the whole image, thus making the attack ineffective. The second method will make the injected stripe appear at different locations in the image across continuous frames. With this design, the attack may only succeed occasionally in discontinuous frames, and thus is difficult to affect the driving decision.

8 Related Work

A few studies have shown the insecurity of camera-based object detection and recognition systems. Earlier studies [11, 23, 33, 37] showed that strong light could saturate the camera and disable the system. To spoof the system, Man et al. [14] managed to inject ghost images of a road sign into the camera using a regular projector, and Li et al. [12] injected adversarial image patterns by physically placing stickers on the camera's lens. Other studies change the scene being filmed by physically modifying the object of interest [3, 7, 39] or projecting light to the scene [13, 17, 19, 40] as adversarial patterns. Our work differs from these papers in that we use laser to inject adjustable color stripes into the image as adversarial patterns, which could be conducted from a longer distance. More recently, several works proposed to fool camera-based systems by attacking relevant non-camera sensors. For example, Ji et al. [9] used sound to attack a camera's image stabilization sensor, causing blurry images that deceive object detectors. Tang et al. [31] proposed a region-of-interest attack that fools traffic light detection by fabricating localization errors.

Some studies have used laser to attack non-camera-based recognition systems, including microphones [30] and LIDAR [2, 23, 28]. These attacks exploit sensors' vulnerabilities and are known as transduction attacks [36]. Other transduction attacks against vehicles have investigated ultrasonic sensors [35], rotational speed sensors [29], and radars [37].

Recently, two concurrent works [10, 25] also exploited the rolling shutter to attack camera-based recognition systems. Like ours, they injected stripe-shaped interference on images. However, the goal of [25] is to cause targeted *image misclassification* by flashing the LED illumination on the object in the scene, and [10] aimed to maximally *disrupt object detection*, i.e., to hide objects in the image, by irradiating blue laser at the camera. Both of them generated adversarial patterns for the entire image. While in this paper, we focus on a completely different scenario, i.e., *fooling traffic light recognition*, which requires different attack methods that inject adversarial patterns onto a specific part of the image where the traffic light is and considering aiming and synchronization in dynamic attack scenarios.

9 Conclusion

In this paper, we show the feasibility of fooling traffic light recognition with a laser. We manage to create adjustable color stripes in the image that can alter the recognition results by exploiting the vulnerability of rolling shutters in CMOS cameras. By modeling the laser attack process, we are able to simulate the attack and search for effective laser parameters. Our evaluation with real cameras shows the attack's effectiveness and demonstrates its potential real-world impact. To alleviate the threat, we propose defenses for existing and future vehicles.

Acknowledgments

We thank our shepherd Prof. Earlence Fernandes and the reviewers for their constructive feedback. This work is supported by China NSFC Grant 61941120, 62071428, 61925109 and 2019 Zhejiang University Academic Award for Outstanding Doctoral Candidates.

References

[1] ApolloAuto. Apollo. https://github.com/ApolloAuto/apollo, 2020.

[2] Yulong Cao, Chaowei Xiao, Benjamin Cyr, Yimeng Zhou, Won Park, Sara Rampazzi, Qi Alfred Chen, Kevin Fu, and Z Morley Mao. Adversarial sensor attack on lidar-based perception in autonomous driving. In *Proceedings of ACM CCS 2019*, pages 2267–2281, 2019.

[3] Alesia Chernikova, Alina Oprea, Cristina Nita-Rotaru, and BaekGyu Kim. Are self-driving cars secure? evasion attacks against deep neural networks for steering angle prediction. In *Proceedings of SPW 2019*, pages 132–137. IEEE, 2019.

[4] Tesla Motors Club. Hw2.5 capabilities. https://teslamotorsclub.com/tmc/threads/hw2-5-capabilities.95278/page-74, 2017.

[5] Nancy Cohen. Bmw puts traffic light recognition to the test. https://techxplore.com/news/2019-06-bmw-traffic-recognition.html, 2019.

[6] deep-learning-traffic lights. davidbrai. https://github.com/davidbrai/deep-learning-traffic-lights, 2020.

[7] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. Robust physical-world attacks on deep learning visual classification. In *Proceedings of CVPR 2018*, pages 1625–1634, 2018.

[8] Hikvision. Hikvision C6 Driving Recorder HD Night Vision. https://shopee.com.my/Hikvision-C6-Driving-Recorder-HD-Night-Vision-Car-Hidden-Free-Installation-Wirel-i.189517295.4757236362, 2020.

[9] Xiaoyu Ji, Yushi Cheng, Yuepeng Zhang, Kai Wang, Chen Yan, Wenyuan Xu, and Kevin Fu. Poltergeist: Acoustic adversarial machine learning against cameras and computer vision. In *Proceedings of IEEE S&P 2021*, pages 1–16. IEEE, 2021.

[10] Sebastian Köhler, Giulio Lovisotto, Simon Birnbach, Richard Baker, and Ivan Martinovic. They see me rollin': Inherent vulnerability of the rolling shutter in cmos image sensors. *arXiv preprint arXiv:2101.10011*, 2021.

[11] Feng Lan, Guangtao Zhai, Zhongpai Gao, and Xiaokang Yang. Live demonstration: Screen piracy protection using saturation laser attack and tpvm. In *Proceedings of ISCAS 2016*, pages 2376–2376. IEEE, 2016.

[12] Juncheng B Li, Frank R Schmidt, and J Zico Kolter. Adversarial camera stickers: A physical camera attack on deep learning classifier. *arXiv preprint arXiv:1904.00759*, 2(2), 2019.

[13] Giulio Lovisotto, Henry Turner, Ivo Sluganovic, Martin Strohmeier, and Ivan Martinovic. Slap: Improving physical adversarial examples with short-lived adversarial perturbations. *arXiv preprint arXiv:2007.04137*, 2020.

[14] Yanmao Man, Ming Li, and Ryan Gerdes. Ghostimage: Remote perception attacks against camera-based image classification systems. In *Proceedings of RAID 2020*, pages 317–332, 2020.

[15] Mobileye. Our technology. https://www.mobileye.com/our-technology/, 2020.

[16] Guo Mu, Zhang Xinyu, Li Deyi, Zhang Tianlei, and An Lifeng. Traffic light detection and recognition for autonomous vehicles. *The Journal of China Universities of Posts and Telecommunications*, 22(1):50–56, 2015.

[17] Ben Nassi, Yisroel Mirsky, Dudi Nassi, Raz Ben-Netanel, Oleg Drokin, and Yuval Elovici. Phantom of the adas: Securing advanced driver-assistance systems from split-second phantom attacks. In *Proceedings of ACM CCS 2020*, pages 293–308, 2020.

[18] Andrei Nedelea. Tesla launches autopilot traffic light and stop sign control. https://insideevs.com/news/414820/tesla-autopilot-traffic-light-recognition/, 2020.

[19] Dinh-Luan Nguyen, Sunpreet S Arora, Yuhang Wu, and Hao Yang. Adversarial light projection attacks on face recognition systems: A feasibility study. In *Proceedings of CVPR 2020*, pages 814–815, 2020.

[20] NHTSA. Crash factors in intersection-related crashes: An on-scene perspective. https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/811366, 2010.

[21] OpenMV. OpenMV Cam H7. https://openmv.io/products/openmv-cam-h7, 2020.

[22] OPPO. OPPO K3. https://www.amazon.in/Test-Exclusive-676/dp/B07HGLBZ9J, 2020.

[23] Jonathan Petit, Bas Stottelaar, Michael Feiri, and Frank Kargl. Remote attacks on automated vehicles sensors: Experiments on camera and lidar. *Black Hat Europe*, 11(2015):995, 2015.

[24] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[25] Athena Sayles, Ashish Hooda, Mohit Gupta, Rahul Chatterjee, and Earlence Fernandes. Invisible perturbations: Physical adversarial examples exploiting the rolling shutter effect. *arXiv preprint arXiv:2011.13375*, 2020.

[26] ON Semiconductor. MT9P006 - 1/2.5-Inch 5 Mp CMOS Digital Image Sensor. https://www.mouser.com/datasheet/2/308/MT9P006-D-606273.pdf, 2017.

[27] ON Semiconductor. AR0132AT: CMOS Image Sensor, 1.2 MP, 1/3". https://www.onsemi.com/products/sensors/image-sensors-processors/image-sensors/ar0132at, 2020.

[28] Hocheol Shin, Dohyun Kim, Yujin Kwon, and Yongdae Kim. Illusion and dazzle: Adversarial optical channel exploits against lidars for automotive applications. In *Proceedings of CHES 2017*, pages 445–467. Springer, 2017.

[29] Yasser Shoukry, Paul Martin, Paulo Tabuada, and Mani Srivastava. Non-invasive spoofing attacks for anti-lock braking systems. In *Proceeding of CHES 2013*, pages 55–72. Springer, 2013.

[30] Takeshi Sugawara, Benjamin Cyr, Sara Rampazzi, Daniel Genkin, and Kevin Fu. Light commands: laser-based audio injection attacks on voice-controllable systems. In *Proceedings of USENIX Security 2020*, pages 2631–2648, 2020.

[31] Kanglan Tang, Junjie Shen, and Qi Alfred Chen. Fooling perception via location: A case of region-of-interest attacks on traffic light detection in autonomous driving. In *Proceedings of AutoSec 2021*, pages 1–4, 2021.

[32] TeslaTap. Autopilot/fsd in late afternoon into the sun. https://forums.tesla.com/discussion/102759/autopilot-fsd-in-late-afternoon-into-the-sun, 2018.

[33] Khai N Truong, Shwetak N Patel, Jay W Summet, and Gregory D Abowd. Preventing camera recording by designing a capture-resistant environment. In *Proceedings of UbiComp 2005*, pages 73–86. Springer, 2005.

[34] Xiaomi. Xiaomi Dash Cam 2 DVR 2K 3” Display WIFI Dash Cam. https://www.aliexpress.com/i/4001129431320.html, 2020.

[35] Wenyuan Xu, Chen Yan, Weibin Jia, Xiaoyu Ji, and Jianhao Liu. Analyzing and enhancing the security of ultrasonic sensors for autonomous vehicles. *IEEE Internet of Things Journal*, 5(6):5015–5029, 2018.

[36] Chen Yan, Hocheol Shin, Connor Bolton, Wenyuan Xu, Yongdae Kim, and Kevin Fu. Sok: A minimalist approach to formalizing analog sensor security. In *Proceedings of IEEE S&P 2020*, pages 233–248. IEEE, 2020.

[37] Chen Yan, Wenyuan Xu, and Jianhao Liu. Can you trust autonomous vehicles: Contactless attacks against sensors of self-driving vehicle. *Def Con*, 24(8):109, 2016.

[38] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of CVPR 2020*, pages 2636–2645, 2020.

[39] Yue Zhao, Hong Zhu, Ruigang Liang, Qintao Shen, Shengzhi Zhang, and Kai Chen. Seeing isn’t believing: Towards more robust adversarial attack against real world object detectors. In *Proceedings of ACM CCS 2019*, pages 1989–2004, 2019.

[40] Zhe Zhou, Di Tang, Xiaofeng Wang, Weili Han, Xiangyu Liu, and Kehuan Zhang. Invisible mask: Practical attacks on face recognition with infrared. *arXiv preprint arXiv:1803.04683*, 2018.

Appendix

A Laser Attack Modeling

A.1 Camera Modeling

Direct Interpolation. Suppose T represents the tensor in RGB form, and M represents the matrix in raw file. Then for pixel in the $i\,th$ row and $j\,th$ column in T, mathematically,

$$T[i,j,0] = \begin{cases} M[i,j] & \text{if } i \; mod \; 2 = 0 \text{ and } j \; mod \; 2 = 0 \\ M[i-1,j] & \text{if } i \; mod \; 2 \neq 0 \text{ and } j \; mod \; 2 = 0 \\ M[i,j-1] & \text{if } i \; mod \; 2 = 0 \text{ and } j \; mod \; 2 \neq 0 \\ M[i-1,j-1] & \text{else} \end{cases}$$

$$T[i,j,1] = \begin{cases} M[i,j+1] & \text{if } i \; mod \; 2 = 0 \text{ and } j \; mod \; 2 = 0 \\ M[i,j] & \text{if } i \; mod \; 2 \neq 0 \text{ and } j \; mod \; 2 = 0 \\ M[i,j] & \text{if } i \; mod \; 2 = 0 \text{ and } j \; mod \; 2 \neq 0 \\ M[i,j-1] & \text{else} \end{cases}$$

$$T[i,j,2] = \begin{cases} M[i+1,j+1] & \text{if } i \; mod \; 2 = 0 \text{ and } j \; mod \; 2 = 0 \\ M[i,j+1] & \text{if } i \; mod \; 2 \neq 0 \text{ and } j \; mod \; 2 = 0 \\ M[i+1,j] & \text{if } i \; mod \; 2 = 0 \text{ and } j \; mod \; 2 \neq 0 \\ M[i,j] & \text{else} \end{cases}$$

Bilinear Interpolation. For the edges of the image, bilinear interpolation takes the same procedure as direct interpolation. For the interior pixels, we use the following rules to implement interpolation for pixel in $i\,th$ row and $j\,th$ column:

$$T[i,j,0] = \frac{\sum_{k=-1}^{1}\sum_{l=-1}^{1} M[i+k,j+l]\mathbf{1}\{\delta_{i+k}+\delta_{j+l}=0\}}{\sum_{k=-1}^{1}\sum_{l=-1}^{1} \mathbf{1}\{\delta_{i+k}+\delta_{j+l}=0\}}$$

Here, for simplicity, we define

$$\delta_i = i \quad \text{mod } 2$$

$$\mathbf{1}\{A\} = \begin{cases} 1 & \text{if A is true} \\ 0 & \text{else} \end{cases}$$

Similarly, we can define for green and blue, respectively:

$$T[i,j,1] = \frac{\sum_{k=-1}^{1}\sum_{l=-1}^{1} M[i+k,j+l]\mathbf{1}\{\delta_{i+k}+\delta_{j+l}=1\}}{\sum_{k=-1}^{1}\sum_{l=-1}^{1} \mathbf{1}\{\delta_{i+k}+\delta_{j+l}=1\}}$$

$$T[i,j,2] = \frac{\sum_{k=-1}^{1}\sum_{l=-1}^{1} M[i+k,j+l]\mathbf{1}\{\delta_{i+k}+\delta_{j+l}=2\}}{\sum_{k=-1}^{1}\sum_{l=-1}^{1} \mathbf{1}\{\delta_{i+k}+\delta_{j+l}=2\}}$$

After interpolation, the raw file matrix is transformed into an tensor that could be used for image processing in RGB.

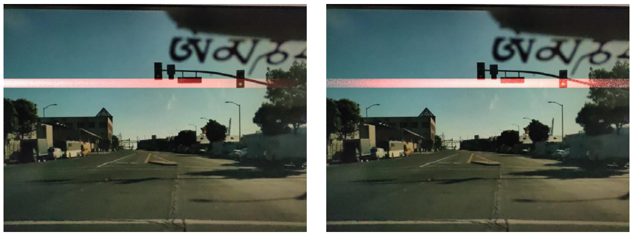

(a) Incidence from the left (b) Incidence from the front

Figure 20: Illustration of the emulated color stripe with noise and irradiated from the left and front of the camera. (Sigmoid model, $I_{max} = 1200, I_{min} = 0, \beta_1 = 8, \beta_2 = 5$).

A.2 Laser Effect Emulation

Linear Function: Linear function is used when the incidence direction is from the left or the right. We made the following assumptions:

- The effect of laser attack is uniform vertically: the light intensity difference only occurs among the x-axis;
- The light intensities at the two ends are I_{max}, I_{min}: For example, if the incidence direction is from the left, the I_{max} will occur at the left side of the image, while I_{min} at the right;
- The light intensity changes linearly horizontally;

Therefore, given I_{max}, I_{min}, we can express the light intensity function for any point (x,y) on the color strip as (if the incidence direction is from the left):

$$I = I_{min} + \frac{y}{w}\left(I_{max} - I_{min}\right)$$

If the incidence direction is from the right, we only need to replace y by $w-y$. The illustrations of the linear model are shown as following:

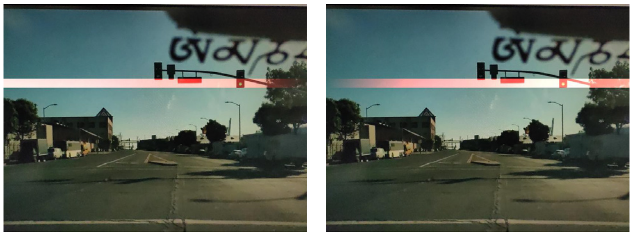
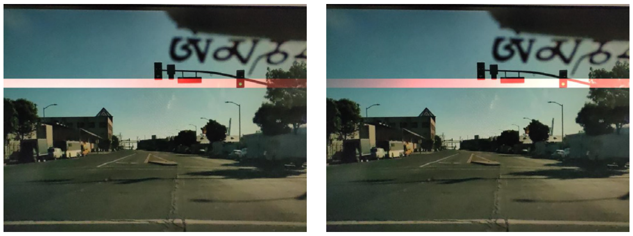

(a) Incidence from the left (b) Incidence from the right

Figure 21: Illustration of the Linear Model for both directions with $I_{max} = 2000, I_{min} = 0$.

Sigmoid Function: Another reasonable assumption is that the color intensity changes exponentially instead of linearly. Sigmoid Function is also used for the case when the incidence direction is from left or right. The assumption of the sigmoid function is the same as the linear case except for the last one: the light intensity changes according to a sigmoid-like function horizontally. Therefore, given I_{max}, I_{min}, we can express the light intensity function for any point (x,y) on the color strip as (if the incidence direction is from the left):

$$I = I_{min} + \frac{1}{1+e^{-\alpha_1/(y-w/\alpha_2)}}\left(I_{max} - I_{min}\right)$$

Here, α_1, α_2 are hyper-parameters. If the incidence direction is from the right, we need to replace $y - \frac{w}{\alpha_2}$ by $\frac{w}{\alpha_2} - y$. The illustrations of the sigmoid model are shown as following:



(a) Incidence from the left

(b) Incidence from the right

Figure 22: Illustration of the Sigmoid Model for both directions with $I_{max} = 2000, I_{min} = 0, \alpha_1 = 5, \alpha_2 = 2$.

Gaussian Function: If the incidence direction of the light is from the front, then the channel overflow is most likely to occur in the middle of the color strip. The assumptions of the Gaussian model is different:

- The effect of laser attack follows two dimensional multi-normal distribution with the peak at the center of the stripe;
- The light intensity at the peak of the distribution is I_{max} and converges to 0 at the periphery;
- The correlation between two directions is 0 ($\rho = 0$, can be relaxed);

To implement the Gaussian model, we need the maximum light intensity I_{max}, and the top position of the color strip x_0. The light intensity function for any points (x, y) on the color strip can be expressed as following:

$$a = \frac{(x - x_0 - h/2)^2}{(h/\sigma_1)^2}$$

$$b = \frac{(y - w/2)^2}{(w/\sigma_2)^2}$$

$$c = \rho \frac{2\,(x - x_0 - h/2)\,(y - w/2)}{(hw)/(\sigma_1\sigma_2)}$$

$$I = \frac{1}{2\pi hw\sqrt{1 - \rho^2}/(\sigma_2\sigma_2)} e^{-\frac{1}{2(1-\rho^2)}(a+b+c)} I_{max}hw$$

Here, σ_1, σ_2 are hyper-parameters representing the decaying rate of the light intensity from the center to the periphery. The illustration of the Gaussian model is shown as following:

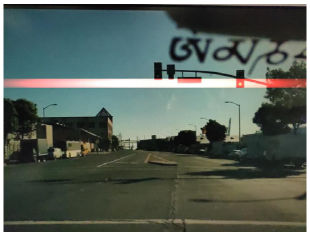


Figure 23: Illustration of the Gaussian Model with $I_{max} = 2000, \sigma_1 = 2, \sigma_2 = 4$.

Adding Noise: The procedure is described in the following:

1. Pick a random number n from n_1 to n_2 to represent the number of noisy pixels;
2. Randomly select $\{h_0, h_1, \cdots, h_{n-1}\} \times \{w_0, w_1, \cdots, w_{n-1}\}$, where $h_i, w_j \in \{1, 2, 3, 4, 5\}$ to represent the sizes of the noisy pixels;
3. If the incidence direction is left/right, then according to distributions $v(x), h(y)$, generate tuples $\{(x_0, y_0), (x_1, y_1), \cdots, (x_{n-1}, y_{n-1})\}$ to represent the upper left corner of the noisy pixels; If the incidence direction is from the front, then generate tuples $\{(x_0, y_0), (x_1, y_1), \cdots, (x_{n-1}, y_{n-1})\}$ according to two-dimensional random distribution;
4. Randomly select $\{b_0, b_1, \cdots, b_{n-1}\}$ to represent the brightness of the pixels;
5. Then for $i = 0, \cdots, n-1$, we replace the red channel value of all pixels in the square from (x_i, y_i) (upper left corner) to $(x_i + h_i, y_i + w_i)$ (lower right corner) by b_i.

A.3 Searching Laser Attack Parameters

Table 3: Grids for parameter search.

Parameter	Symbol	Grids
Min. Light Strength	I_{min}	0,400,800,1200,1600
Max. Light Strength	I_{max}	200,600,1000,1400,1800
Wavelength (Red)	λ_{red}	632,650,660
Wavelength (Green)	λ_{green}	505,520
Incidence Direction	None	Left, Right, Front
Incidence Function	D	Linear, Sigmoid, Gaussian

A.4 Necessity of the Laser Attack Modeling

We conducted baseline experiments to evaluate the necessity of the methodology design. We performed two groups of experiments, one with laser parameters derived from the model-based search, and the other without (as the naive baseline). In particular, the laser parameters in the naive baseline experiments were picked randomly by a volunteer without knowledge of the recommended parameters, e.g., he can use any laser strength as long as he thinks it may fool traffic light recognition. Green-to-red attacks were conducted 20 times for each group using our previous setup (Fig. 15) at various traffic light distances and angles. The model-guided group was successful 14/20 times, rendering a success rate of 70%, while the naive baseline group succeeded 6/20 times, rendering a success rate of 30%. Thus, the comparison shows that the methodology we designed can significantly increase the attack's effectiveness.

B Evaluation and Defense

The cameras we tested are shown in Fig. 24 and their specs are listed in Table 4. The images they capture under attack are shown in Fig. 25 to 29.

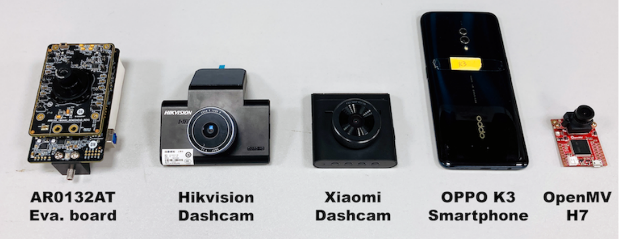
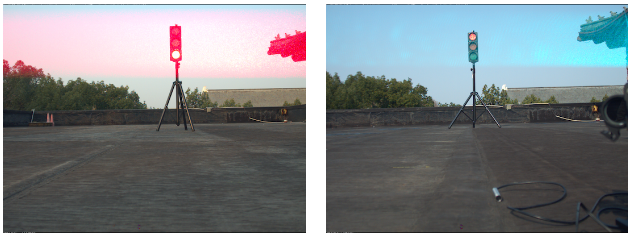


Figure 24: The five cameras experimented in the evaluation.

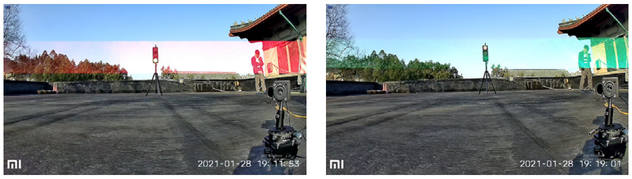
Table 4: Specs of the five experimented cameras.

Parameter	Tesla	Hikv	Xiaomi	OPPO	OpMV
Sensor	AR0132	N/A	OV OS05A10	IMX519	OV7725
Refresh Rate (Hz)	45.458	20	30	30.012	N/A
Resolution	1280* 964	2560* 1600	1920* 1080	4608* 3456	320* 240
Lens	Sunex DSL945D	N/A	N/A	N/A	N/A
Field of View (°)	70	130	130	N/A	70.8
F	2.5	1.6	1.8	1.8	2.0



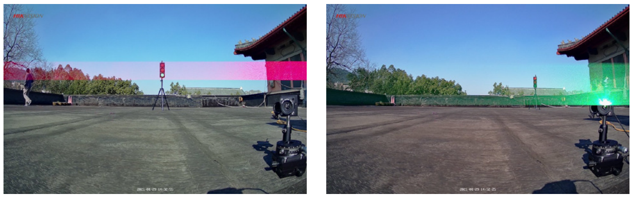
(a) G→R (b) R→G

Figure 25: Images of attacking the AR0132AT eval. board.



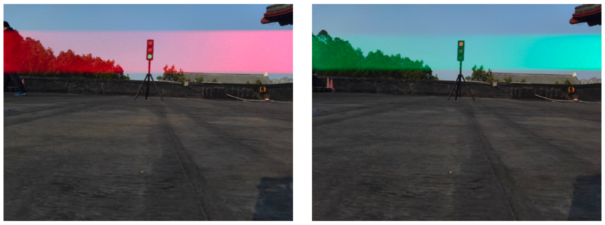
(a) G→R (b) R→G

Figure 26: Images of attacking the Xiaomi Dashcam.



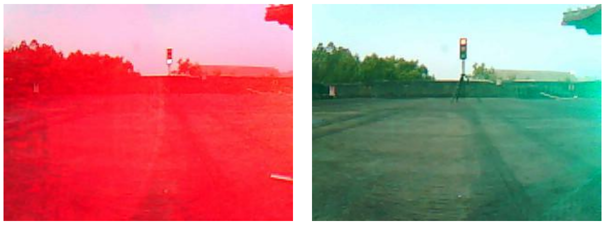
(a) G→R (b) R→G

Figure 27: Images of attacking the Hikvision Dashcam.



(a) G→R (b) R→G

Figure 28: Images of attacking the OPPO K3 camera.



(a) G→R (b) R→G

Figure 29: Images of attacking the OpenMV H7 devboard.

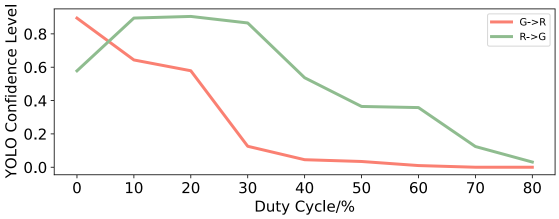
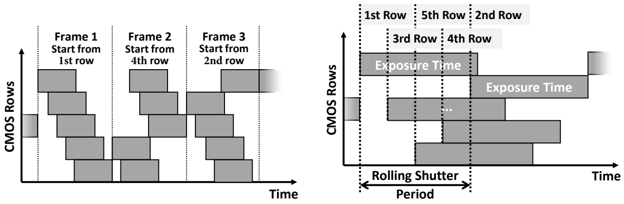
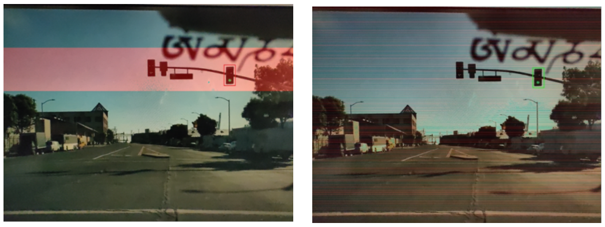


Figure 30: The confidence score of traffic light detection while injecting stripes of various sizes. A larger duty cycle indicates a wider stripe. It shows that a wide stripe can significantly disrupt traffic light detection, and thus shall be avoided.



(a) Random starting row (b) Random rolling sequence

Figure 31: Random rolling shutter mechanisms that can destabilize or atomize the color stripe as a defense.



(a) Before defense (b) After defense

Figure 32: Images under attack before and after applying the random rolling sequence countermeasure.