RoLMA: A Practical Adversarial Attack against Deep Learning-based LPR Systems

Mingming Zha[1,2], Guozhu Meng[1,2,⋆], Chaoyang Lin[1,2], Zhe Zhou[3], and Kai Chen[1,2]

[1] State Key Laboratory of Information Security, Institute of Information Engineering Chinese Academy of Science, Beijing, China
[2] School of Cyber Security, University of Chinese Academy of Sciences, China
{zhamingming,mengguozhu,linchaoyang,chenkai}@iie.ac.cn
[3] Fudan University, Shanghai, China
zhouzhe@fudan.edu.cn

Abstract. With the advances of deep learning, license plate recognition (LPR) based on deep learning has been widely used in public transport such as electronic toll collection, car parking management and law enforcement. Deep neural networks are proverbially vulnerable to crafted adversarial examples, which has been proved in many applications like object recognition, malware detection, etc. However, it is more challenging to launch a practical adversarial attack against LPR systems as any covering or scrawling to license plate is prohibited by law. On the other hand, the created perturbations are susceptible to the surrounding environment including illumination conditions, shooting distances and angles of LPR systems. To this end, we propose the first practical adversarial attack, named as RoLMA, against deep learning-based LPR systems. We adopt illumination technologies to create a number of light spots as noises on the license plate, and design targeted and non-targeted strategies to find out the optimal adversarial example against HyperLPR, a state-of-the-art LPR system. We physicalize these perturbations on a real license plate by virtue of generated adversarial examples. Extensive experiments demonstrate that RoLMA can effectively deceive HyperLPR with an 89.15% success rate in targeted attacks and 97.3% in non-targeted attacks. Moreover, our experiments also prove its high practicality with a 91.43% success rate towards physical license plates, and imperceptibility with around 93.56% of investigated participants being able to correctly recognize license plates.

Keywords: pratical adversarial attack, license plate recognition

1 Introduction

Attributed to the rapid development of deep learning, license plate recognition (LPR) systems are experiencing a dramatic improvement in recognition accuracy and efficiency. The state-of-the-art deep learning-based license plate recognition systems (hereafter referred to as DL-LPR) can achieve high accuracy over 99% [14]. The great success boosts its wide deployment in many areas such as

⋆ Corresponding author

electronic toll collection, car parking management and law enforcement. However, modern deep learning is vulnerable to *adversarial examples* [12]. For instance, a slight perturbation added to an image, which is imperceptible to humans, can easily fool a model of deep neural networks [5]. Analogically, DL-LPR is also suffering from the threat of adversarial examples that incur wrong recognitions. However, it is non-trivial to ensure adversarial examples to be still effective in the physical world. To date, no prior work to our knowledge has explored the practical adversarial attacks against DL-LPR systems.

Challenges of a practical adversarial attack against DL-LPR. To fool a DL-LPR system is much more difficult than to deceive an image classifier. There are two main challenges for performing a practical adversarial attack against modern DL-LPR systems in the physical world.

C1. The perturbations to license plates are extremely restrictive. License plates are generally issued by a local government department that regulates communications and transport for official identification purposes [2]. They are allegedly not allowed to be altered, obliterated or covered by anything. Therefore, we cannot make any permanent modifications, even minor ones that are imperceptible to a human, to a license plate.

C2. Launching adversarial attacks against DL-LPR systems in the physical world is much more challenging [10]. When DL-LPR systems recognize the license plates attached to fast-moving motor vehicles, the distance and shooting angle to DL-LPR systems are changing over time. Besides, the sunlight or supplement light around the vehicle can also degrade the photographing of license plate. All the above can negatively impact on the effectiveness and robustness of adversarial examples.

Robust Light Mask Attacks against DL-LPR. In this paper, we put forward the first robust yet practical adversarial attack, termed **Ro**bust **L**ight **M**ask **A**ttacks (RoLMA), against DL-LPR systems in the physical world. We select a popular DL-LPR system HyperLPR [22] as the target model, and execute two types of adversarial attacks (see Section 4.3)–*a targeted attack* is to create an adversarial license plate in the disguise of a designated one; *a non-targeted attack* is to make a original license plate recognized as any different one.

To address challenge C1, we employ illumination technologies to illuminate license plates instead of scrawling them. The produced light spots can persistently make noises to LPR cameras during the process of photographing, and moreover be removed once away from the monitor areas. To improve its effectiveness and robustness under different circumstances, *i.e.* C2, we identify three environmental factors of most influence: light noise from many other light sources, shooting distances, and shooting angles. Subsequently, we perform *image transformation* on a digital license plate during adversarial example optimization. In particular, we adjust brightness to simulate the varying light, rescale the image to simulate the shooting distances, and rotate the image to simulate the shooting angles (see Section 4.2).

Physical deployment of RoLMA. We install several LED lamps in a license plate frame and create designed spots. Then we adjust the position, size,

brightness of light spots, and conduct extensive experiments to evaluate RoLMA: RoLMA achieves an 89.15% success rate in targeted attacks and a 97.30% success rate in non-targeted attacks; RoLMA also proves to be very effective in the physical world and obtains a 91.43% success rate of physical attacks; the adversarial license plates are imperceptible to human beings as most of the investigated volunteers attribute the perturbations to natural light (78.32%) rather than artificial light. Additionally, we have reported our findings to Zeusee [22], and they acknowledged the importance of the problems we discovered. More details can be found here[4].

Contributions. We summarize our contributions as follows:

- *Effective algorithm to generate adversarial examples.* We developed an effective algorithm to make appropriate perturbations and generate adversarial license plates of high robustness. These adversarial license plates are effective in deceiving the target LPR system.
- *Practical adversarial attacks against DL-LPR systems.* We designed and developed the first practical adversarial attack against DL-LPR systems, which is still effective under different circumstances of the real world, such as variable-sized shooting distances and angles.
- *Extensive and comprehensive experiments.* We conducted extensive experiments to evaluate our approach including effectiveness, practicality, and imperceptibility. The results demonstrated that the adversarial examples generated by our approach could effectively devastate the modern LPR systems.

2 Background

2.1 License plate recognition

License plate recognition (LPR) is a technology that recognizes vehicle registration plates from images automatically. To date, it has a broad use in transportation, for example, levying tolls on pay-per-use roads, charging parking fees, capturing traffic offenses. LPR usually employs *optical character recognition* (OCR) to convert images into machine-readable text. Typically, OCR technologies can be categorized into two classes: *character-based recognition* and *end-to-end recognition*.

Character-based recognition is the traditional approach to recognize the text from images of license plates [15]. Given an image of a license plate, the character-based recognition system first segments it into several pieces, ensuring that one piece only contains one character [11]. The classifier, oftentimes equipped with classification algorithms (*e.g.*, SVN, ANN, and k-nearest neighbors), can output the most likely character. The performance of LPR does not only rely on a recognition algorithm but also character segmentation to a large extent.

End-to-end recognition is a more recent technology that gains the majority of attention in the field of LPR. It recognizes the entire sequence of characters in a variable-sized "block of text" image with deep neural networks. It is able to

[4] https://sites.google.com/view/rolma-adversarial-attack/responses

produce the final results (*i.e.*, machine-encoded text), without feature selection, extraction, and even character segmentation. A number of deep learning models including Recurrent Neural Networks, Hidden Markov Models, Long Short Term Memory Networks, and Gated Recurrent Units, have been applied in LPR and obtain superior results [8, 9].

2.2 HyperLPR

HyperLPR [22] is a high-performance license plate recognition framework developed by Zeusee Technologies. It employs an end-to-end recognition network GRU, which takes a graphical license plate of size $h \times w$ as input and produces the most likely sequence of characters as output. It starts with a convolution layer (Conv2D) with a $3 \times 3 \times 32$ filter, a batch-normalization and *relu* activation, followed by a 2×2 max-pooling layer(MaxPooling2D). Then two layers follow which have the same architecture as above but with different filters, *i.e.*, one is with $3 \times 3 \times 64$ and the other is with $3 \times 3 \times 128$. The probabilities from the last activation function are passed to a network with 4 gated recurrent units (GRUs) of 256 hidden units, and a dropout layer (its rate is 0.25). Last, the output layer utilizes *softmax* to normalize an 84-unit probability distribution, corresponding to the number of possible license plate characters. In this study, we choose HyperLPR as our attack target, then develop the approach RoLMA to generate a massive number of adversarial license plates that can evade the recognition.

3 Problem Statement

In this section, we present the attack goal, attack scenarios, and the capability of adversaries.

3.1 Attack Goal

We aim at constructing a practical adversarial attack against DL-LPR. The adversarial license plates are expected to be misclassified by DL-LPR but recognized correctly by humans. Without the loss of generality, we define the following terms involved in this study: one registration number $\mathcal{L}$ of a motor vehicle is a sequence of characters $\langle c_1, c_2, \ldots, c_n \rangle$. Assuming that only m characters can be used as a license plate, *i.e.*, the available character set $\mathcal{V}$, we then have $c_i \in \mathcal{V}$. In addition, there are some constraints in a license plate, such as the length of characters n. So we use $\mathcal{C}$ to denote these constraints. Lastly, we have $\mathcal{L} : \langle c_1, c_2, \ldots, c_n \rangle \sim \{\mathcal{V}, \mathcal{C}\}$. One LPR system is able to convert an image G to a machine-readable license number, *i.e.*, $f(G) = \mathcal{L}$.

Adversarial License Plate. We generate an adversarial license plate by adding the slight perturbation p to the original graphical license plate G. We use G' to denote the adversarial plate and $G' = G + p$. With respect to G', the target LPR system can output a new license number $\mathcal{L}'$, *i.e.*, $f(G') = \mathcal{L}'$, $\mathcal{L}' \sim \{\mathcal{V}, \mathcal{C}\}$, and $\mathcal{L}' \neq \mathcal{L}$. That is, *the goal is to disguise the original license plate as the other for DL_LPR systems*. To ensure practicality, the adversarial license plates should satisfy all constraints $\mathcal{C}$ as the original one does.

3.2 Attack Scenarios

In this section, we design two attack scenarios for our RoLMA approach.

- *Car parking management.* More and more car parks start to equip automatic DL-LPR systems for parking management [1], *e.g.*, parking access automation and automated deduction of parking fees. The license plate serves as an access token for identity authentication, and only registered licenses could access the parking service. In such a case, the adversaries can resort to the adversarial licenses to elevate their privileges. On the other hand, if the automated deduction of parking fees is based on DL-LPR systems, the adversaries can counterfeit others' license plates and get free parking.
- *Law Enforcement.* Since LPR has been long used for identifying vehicles in a blacklist, an adversarial license plate can escape from the detection successfully. Generally, one well-formed and legal license plate would not trigger LPR's attention. But if the adversarial license plate is recognized as being of the wrong format, it is probable that a specialized staff is sent for manual inspection [6]. It is well-known that adversarial examples can be correctly recognized by a human. Besides, this attack can also affect other common law enforcement applications such as border control and red-light enforcement.

3.3 The capability of adversaries

In this study, we aim to generate adversarial license plates with respect to the DL-LPR system. Since HyperLPR is open-source and high-performance, we select it as the target model, then know the details of its model. So the process of adversarial license plate generation is a kind of white-box attack. In order to attack the deployed DL-LPR systems in reality, the adversaries have to decorate the license plate in a "mild" fashion. It is because license plates should comply with many regulations allegedly by law. More specifically, the adversaries cannot cover, scrawl or discharge license plates in any manner. *In this study, we use the spotlight as a decoration method to confuse DL-LPR systems. The rationale is that light is ubiquitous such as the natural light and license plate light, so that it is hard to determine how comes a light spot on the license plate.*

4 The RoLMA Methodology

To convert the original license plate to an adversarial one, we propose the **Ro**bust **L**ight **M**ask **A**ttack (RoLMA). It proceeds with three key phases in Figure 1: *illumination, realistic approximation, loss calculation.* However, these digital adversarial images cannot be directly fed to LPR systems for recognition. Instead, we apply several spot light bulbs to irradiate the license plate in order to get light spots. Next, we adjust the positions, size, brightness of light spots, photograph the irradiated license plate and compare it with the digital adversarial image. Finally, we use the irradiated license plate to apply practical attack. More details can be found here[5].

[5] https://sites.google.com/view/rolma-adversarial-attack

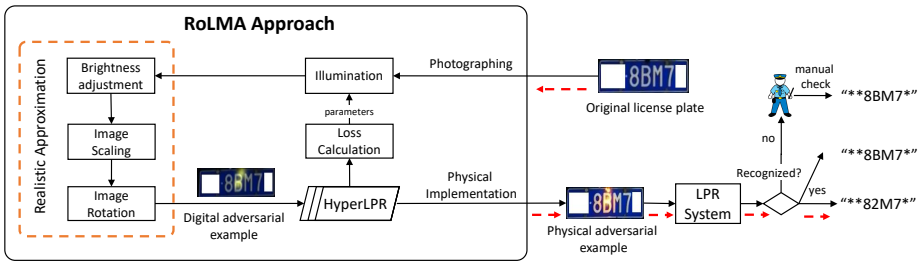

Fig. 1: The system overall of RoLMA

4.1 Illumination

Adversarial examples differ from the original samples in crafted perturbations. The perturbation could be a change of pixels in image classification, an adjustment of an acoustic wave in speech recognition [3]. Generally, license plate recognition reads machine-readable text from an image. Although pixel changes can also make LPR systems misrecognize in the digital space, it has several problems in the physical world: 1) changed pixels are susceptible to shooting settings by LPR cameras (*e.g.*, distance and angle) and the circumstance conditions (*e.g.*, air quality and sunlight intensity); 2) a license plate should remain tidy, uncovered, and unaltered. As a result, it is nearly impossible to scrawl it with previous ways [16]. In this study, we propose an illumination technology and decorate the target license plate with visible lights. The light mask can be taken on and off at any time, without making a permanent scratch to the license plate. In addition, when the LPR system is recognizing a vehicle, the circumstance around the vehicle is full of light, either sunlight or a street light, headlights or rear lights. If the decorated license plate can still be correctly recognized by a human, it will likely not incur a violation of laws.

In this study, we select LED lamps as our illumination source. LED lamps are installed at the rear of a vehicle, and make several light spots on the license plate. To work out an illumination solution, we draw several light spots on a digital license plate, which is captured from a physical license plate. This decorated image is then passed to HyperLPR to check whether it is an adversarial example. We model such a light spot according to its color, position, size, brightness, but not shape.

- *Color.* The background of license plates usually varies from colors. In this study, the color c is modeled as RGB values and optimized gradually during the computation of adversarial examples.
- *Position.* A light spot is positioned by its circle center. We establish a rectangular coordinate system on a license plate. The point at the left bottom has a coordinate (0, 0), and the point (x, y) denotes that it is x away from the left border and y away from the bottom border. In such a fashion, we can represent the center p of a light spot with (c_x, c_y).
- *Size.* It indicates the irradiated area of a light spot, which is measured by the radius r of the circle, *i.e.*, $s = \pi r^2$. As mentioned beforehand, our physical light spots may be not an accurate circle, and more often an ellipse.

– *Brightness.* When a spotlight emits to a plane, the center of the spot is brightest and the light scatters in a decaying rate. Given a point (x, y) inside the spot, the brightness of this point $b(x, y)$ obeys normal distribution probability density function (norm_pdf), *i.e.*, $b(x, y) \sim N(r, \sigma^2)$. Let λ be the brightness coefficient, $b(x, y) = \lambda \times norm\_pdf(\sqrt{(x - c_x)^2 + (y - c_y)^2})$ and the brightness of the circle center is $\frac{\lambda}{\sqrt{2\pi}\sigma}$.

Until now, a light spot can be characterized by its color, position, size and brightness, that is $spot = (C, P, S, B)$. As mentioned above, the color is determined by its RGB values rgb, the position is decided by the coordinates of the circle center (c_x, c_y), the size is determined by the radius r, and the brightness is determined by its standard deviation σ. To search an adversarial example, we intend to make our illuminated license plate misrecognized to a wrong number and the *loss* function reaches the approximately minimal value.

$$\arg\min_{rgb,(c_x,c_y),r,\sigma} \quad L(X)$$

where X is an input image, and $L(X)$ is the loss function for adversarial examples.

4.2 Realistic Approximation

Adversarial attacks are seriously sensitive to external noises from the physical world [4]. With regards to the two scenarios mentioned in Section 3.2, there are many challenges as shown in Section 1. As a consequence, we propose three tactics to approximate the reality and improve the robustness of RoLMA as follows: 1) *Brightness Adjustment.* To simulate the impact of different lights in the real environment, we utilize TensorFlow via the API "`tf.image.random_brightness`" to adjust the brightness of images randomly. 2) *Image Scaling.* It is used to simulate the varying shooting distances of LPR cameras away from the vehicle. Here we adopt "`tf.image.resize_images`" to resize the license plate randomly. Moreover, the scaling holds a fixed width-height ratio, avoiding a badly distorted license plate which is nearly impossible to happen. 3) *Image Rotation.* The robustness of adversarial examples is susceptible to shooting angles of LPR cameras. In the same manner, we invoke the API "`tf.contrib.image.rotate`" of TensorFlow to shift the image with a random angle, departing from its coordinates.

4.3 Loss Calculation

In this section, we present the details about how to determine the efficiency of perturbations and provide finer parameters for illumination.

Oracle. To generate adversarial examples, we take HyperLPR as the *oracle* to guide the process. Given an input of image X, HyperLPR outputs a sequence of characters $\langle c_1, c_2, \ldots, c_n \rangle$. As mentioned in Section 3.1, we aim to make LPR

systems produce a wrong license $\mathcal{L}'$ from a real license $\mathcal{L}$. They are of the same length and both comply with lawful constraints, but different in at least one character. Assuming the rth character is c_r, we obtain the probability distribution for this character as $\{(c_1, p_1), (c_2, p_2), \ldots, (c_n, p_n)\}$ where $p_1 = max\{p_i\}$ and $c_1 \neq c_r$. Surely, the overall confidence of this recognition should be higher than the requirement $C \geq \theta$. In this study, we define the following two attacks in terms of generated adversarial examples.

Targeted Adversarial Attack. This is a directed attack, where RoLMA can cause HyperLPR to recognize the adversarial license plate as a specific license number. For example, we attempt to make the license plate "N92BR8" recognized as "N925R8". Then all the adjustments of parameters are targeting this goal. *This attack is especially suitable for the scenario of car parking management, as it can disguise a privilege license number to access the parking service.*

In a targeted adversarial attack, the original license is $\mathcal{L} : \langle c_1, c_2, \ldots, c_n \rangle$, and the targeted one is $\mathcal{L}' : \langle c'_1, c'_2, \ldots, c'_n \rangle$. The inconsistent characters in between are $\{(c_i, c'_i)\} \in \mathcal{D}$. In order to generate an adversarial example G', we utilize a loss function to measure the differences between the real sequence of characters and the targeted one. The optimization process is conducted in two directions: (1) decreasing the loss of the whole sequence against the target; (2) decreasing the loss of specifically targeted characters $c_i \in \mathcal{D}$ against the target characters. Thus, the loss function is as follows.

$$\arg\min_{G'} \quad \alpha \times L_{CTC}(f(G^{'}), \mathcal{L}^{'}) + \sum_{(c_i, c'_i) \in \mathcal{D}} L(c_i, c'_i) \tag{1}$$

where L_{CTC} is the CTC loss function for label sequence and $\sum_{(c_i, c'_i) \in \mathcal{D}} L(c_i, c'_i)$ is the sum of losses which are the editing distances between all targeted characters and the original ground true characters. The coefficient α balances the two variables in the loss function.

Non-targeted Adversarial Attack. The goal of non-targeted adversarial attacks is to fool a LPR system by producing any wrong recognition. *This attack is very suitable for the scenarios of escaping electronic tolls collection and blacklisted vehicle detection.* A non-targeted attack contains two uncertainties–which characters will be changed in adversarial examples at the sequence level, and what the original characters will become at the character level. As such, we aim to find an optimal solution to minimize the distance between adversarial examples with the original at the sequence level. Moreover, this solution leads to a wrong recognition with its confidence satisfied. Let $d(\mathcal{L}, \mathcal{L}')$ be the editing distance between the two licenses $\mathcal{L}$ and $\mathcal{L}^{'}$ and $f(G^{'}) = \mathcal{L}^{'}$ as aforementioned. Moreover, $C_{f(G')}$ is the confidence of the targeted license G', and θ is a threshold of confidence, here we set it as 0.75. The optimization process can be formulated as Equation 2.

$$\arg\min_{G'} \quad d(f(G'), \mathcal{L}) \ \cap \ C_{f(G')} \geq \theta \tag{2}$$

Here we utilize Simulated Annealing (SA) to guide the process of non-targeted adversarial attacks as shown in algorithm 1. In particular, the iteration process

Algorithm 1: Non-targeted adversarial attacks based on SA

Input: $\{(c_i, p_i)|1 \leq i \leq n\}$: a descending list of possible chars by probabilities; T: the intial degree of temperature and $T > 0$; λ: the annealing rate and $0 < \lambda < 1$; MAX: the maximal number of iterations for adversarial example generation; G: the original image of license plate

Output: $G^{'}$: adversarial license plate, where $c_1^{'} \neq c_1$

1 $iter \leftarrow 0, c_i^{'} \leftarrow c_i,\ p_i^{'} \leftarrow p_i, i \in [1, n]$;
2 **while** $c_1^{'} = c_1$ ***and*** $iter < MAX$ **do**
3 $\quad \Delta p \leftarrow p_2^{'} - p_1^{'}$;
4 $\quad G^{'} \leftarrow G + \delta_{c_1, c_1^{'}}$;
5 $\quad$ **for** $i \leftarrow 2$ **to** n **do**
6 $\quad\quad \{(c_i^{''}, p_i^{''})\} \leftarrow license\_plate\_recognition(G^{'})$;
7 $\quad\quad$ sort $\{(c_i^{''}, p_i^{''})\}$ where $p_i^{''} \geq p_{i+1}^{''}$;
8 $\quad\quad$ **if** $c_1^{''} \neq c_1^{'}$ **then**
9 $\quad\quad\quad c_i^{'} \leftarrow c_i^{''},\ p_i^{'} \leftarrow p_i^{''}, i \in [1, n]$;
10 $\quad\quad\quad$ break;
11 $\quad\quad \Delta p_{new} \leftarrow p_2^{''} - p_1^{''}$;
12 $\quad\quad$ **if** $\Delta p_{new} < \Delta p$ ***or*** $e^{\frac{\Delta p - \Delta p_{new}}{T}} > rand(0, 1)$ **then**
13 $\quad\quad\quad c_i^{'} \leftarrow c_i^{''},\ p_i^{'} \leftarrow p_i^{''}, i \in [1, n]$;
14 $\quad\quad\quad$ break;
15 $\quad T \leftarrow \lambda \times T$;
16 $\quad iter \leftarrow iter + 1$;
17 $\quad$ **if** $G^{'}$ *satisfies the constraints* $\mathcal{C}$ **then**
18 $\quad\quad G \leftarrow G^{'}$;
19 **return** $G^{'}$;

is continuing unless one wrong character gains the largest probability or it exceeds the maximal iteration number MAX (line 2). Line 3 is to compute the probability gap between the first two characters. It can roughly measure the chance to accomplish a wrong recognition. Line 4 is to generate the perturbed license plate $G^{'}$ by adding the perturbation $\delta_{c_1, c_1^{'}}$, and $\delta_{c_1, c_1^{'}}$ is computed by the targeted adversarial attack as described above. Line 5 to 14 present which wrong characters will be selected for the next decoration. Following with a descending order of probability, we select the 2nd character as our first decoration target. A new probability distribution is produced by LPR system (line 6) and sorted as per probabilities (line 7). If a wrong recognition is achieved (line 8), we terminate the iteration process. Otherwise, we compute the chance of wrong recognition in the current probability distribution (line 11) and compare it with the previous one. If the chance is increased, *i.e.* $\Delta p_{new} < \Delta p$, we accept this decoration. Otherwise, we accept this decoration with a probability calculated in line 12. We evolve the value of temperature at line 15. When we get G', we need to check whether G' follows the constraints $\mathcal{C}$ on the license plate numbering system in order not to be rejected at line 17. If the G' satisfies the constraints $\mathcal{C}$, then we will update G at line 18.

5 Evaluation

We implement RoLMA on the base of TensorFlow and Keras. The experiments are conducted on a server with 32 Intel(R) Xeon(R) CPUs of E5-2620 and 64GB memory. Through these experiments, we intend to answer:

RQ1. How effectively does RoLMA generate adversarial license plates and how successfully do these adversarial examples deceive the HyperLPR system?
RQ2. How is the success rate of the practical attacks guided by these adversarial examples?
RQ3. Are these adversarial examples imperceptible enough for ordinary audiences?

Experiment Subject. We prepare two types of data sets for the experiments as follows. All the license plates can be recognized correctly by HyperLPR.

- **Real license plates**. We have collected 1000 images of license plates from CCPD [18]. Due to the influences of the surrounding environment, many of the images are blurred and of low quality.
- **Synthesized license plates**. We also synthesize a number of license plates by ourselves following the design specification of a legal license plate. We randomly select characters from the limited alphabet. Constraints are checked to guarantee these license plates are valid. In total, we create 1000 license plates of high quality without any noise from the physical environments.

Parameter Determination. RoLMA uses illumination technique to create spots on the license plate to fool a LPR system. However, if the number of light spots is too small, we may not be able to gain a high success rate, *i.e.*, failure on generating adversarial examples. Inversely, installing a larger number of light spots is also not a good choice since it may cause a failed recognition and too remarkable for ordinary audiences. Therefore, we first design an experiment to identify the favored number of light spots that could effectively fool LPR systems. We randomly select 100 license plates from the data set, and commence to generate adversarial examples with an increasing number of light spots from 1 to 10. We set a maximal iteration number as 5,000 in each trial, and then one trial will stop if either an adversarial example is generated or the iteration number exceeds 5,000. It is worth mentioning that we use a non-targeted strategy for adversarial attacks. The result shows the success rates of attacks along with the number of light spots. The success rate is raised slightly after 5. As a result, we only make 5 light spots to license plate in the following experiments.

5.1 RQ1: Effectiveness

In this experiment, we aim to explore the effectiveness of RoLMA in digital space, *i.e.*, the generated adversarial images are directly passed to HyperLPR for performance assessment. More specifically, we conduct two types of attacks: *Targeted adversarial attack.* For each license plate, we aim to receive a specific

wrong license number from HyperLPR. We employ random algorithms first to identify which character to be disturbed, then disguise the character as a different one. One attack is terminated once the target is accomplished or the iteration exceeds 5,000 times; *Non-targeted adversarial attack.* Target is not necessarily designated in a non-targeted adversarial attack. Therefore, we will not specify a target for each license plate. One attack is terminated once an adversarial example is obtained or it exceeds the maximal iterations.

Table 1: Success rate of targeted and non-targeted attacks

Data	**Targeted Attack**		**Non-targeted Attack**	
	Success	Confidence	Success	Confidence
Real	92.60%	86.55%	99.70%	91.59%
Synthesized	85.70%	85.64%	94.90%	90.88%
Average	89.15%	85.95%	97.30%	91.28%

Table 1 shows the results of these attacks on both real license plates and synthesized license plates. The success rate of non-targeted attacks is 97.3% outperforming targeted attacks (89.15%). That is because one character has varying difficulties to pretend to other characters as concluded above. Some characters cannot be even achieved regardless of how to optimize. There are still a number of trial instances failing to deceive HyperLPR. For example, we cannot find an adversarial example for the license plate "A40F29" in a limited time. In addition, we find that the success rate in synthesized license is always smaller than real license's in both attacks. The reason is that the synthesized license plates have relatively higher definition compared to the real license plates, which means the correct characters can be recognized with a higher probability. In contrast, when HyperLPR is recognizing a blurred image, it is prone to making the results with lower confidence or even cannot determine the final characters. As a consequence, fewer additional perturbations may cause a wrong recognition for real license plates and much more perturbations have to be made to the synthesized license plates for adversarial examples.

Comparison with random illumination attack. We launch another attack by randomly illuminating the 2000 images in our data set. The randomness of the illumination attack lies in the number of light spots, the color, brightness, size and position for each spot. After all, we obtain 2000 decorated images with random spots. HyperLPR can correctly recognize 96.95% of them. Only 1.90% of them can deceive HyperLPR, which is far less effective than the non-target attack of RoLMA (97.30%). It is concluded that modern LPR systems have great resistance to this random illumination attack. It is non-trivial to generate adversarial examples effectively without considering LPR algorithms. This experiment also proves that RoLMA achieves superior performance by exploring the weaknesses residing in LPR algorithms.

5.2 RQ2: Practicability

In this section, we apply targeted attack to evaluate the practicability of RoLMA by instantiating adversarial perturbations on real license plates.

Experiment Design. 1) We install these electronic devices on a car and calibrate these LEDs carefully. If the captured license plates are remarkably different from the digital adversarial image, then we will adjust the supply current, illumination direction, and used lenses to change formed light sports. The calibration is stopped if two images are different within a tolerant threshold θ. And the limitation of physical calibration time is set to 5 minutes. 2) We record two continuous videos for the decorated license plate: the first video is filmed at the horizontal plane with the license plate in a "Δ" route. More specifically, the camera is at the back of the stationary car with a distance of 2 meters. Then we move the camera to the left-back with a 30^o horizontal angle till to a location with a 3-meter distance. We then move the camera horizontally to the right till the symmetric location, and finally move to the left front till the start point; the second video is filmed at a higher position with a 45^o depression angle to the license plate. The camera is moved from the left ($\approx 15^o$ horizontal angle) of the license plate to the right ($\approx 15^o$ horizontal angle). The distance of the camera to the license plate is 2 meters. This experiment lasts around 2 hours and gets two one-minute videos.

Experiment Results. In our recorded videos, there are 1600 frames of image totally and 922 valid frames remain after filtering out blurred images. We feed these valid images to HyperLPR and 843 of them are misrecognized. Hence, the success rate of our physical attack is 91.43%. The averaged confidence of recognition results is 87.24%. Moreover, the average time of physical calibration is about 3 minutes.

Table 2: Recognition results in the physical attacks

No	**Distance** (meters)	**Depress.**	**Horizon.**	**Text**	**Conf.** (%)
1	2	0^o	0^o	■■8BM7■	98.06
2	2	0^o	0^o	■■82M7■	86.93
3	3	0^o	-30^o	■■82M7■	85.91
4	3	0^o	+30^o	■■82M7■	86.35
5	2	45^o	0^o	■■82M7■	90.92
6	2	45^o	-15^o	■■82M7■	91.40
7	2	45^o	+15^o	■■82M7■	87.64

Examples. We select six images recorded in this physical attack shown on the website[6], and the recognition results in Table 2. These images are captured with varying distances and shooting angles. In particular, the first image is shot with the original license plate and the camera is 2 meters away behind. HyperLPR can output "■■8BM7■" correctly with a confidence of 98.06%. To protect privacy, we use "■" to cover specific characters in both the images and recognized

[6] https://sites.google.com/view/rolma-adversarial-attack/practicability

text. The other six images, shot from the decorated license plate, can all make HyperLPR output "[redacted]82M7[redacted]". As shown in Table 2, "Distance" denotes the distance of the camera to the license plate, "Depress." means the depression angle of photographing, "Horizon." means the horizontal angle of photographing, and "Conf." denotes the confidence of HyperLPR with regard to recognition results. Noted that "-30°" and "-15°" indicate the camera is at the left side of the license plate while "+30°" and "+15°" mean the right side. These decorated license plates are all recognized wrongly, according to our computation in the experiments. It shows that RoLMA is very effective in generating adversarial examples, and these adversarial examples are very robust in the physical world.

5.3 RQ3: Imperceptibility

Imperceptibility is another important feature for adversarial examples, which means the perturbations do not affect users' decision. In the field of license plate recognition, practical adversarial examples impose a new implication to this concept: the license plate is still recognized correctly, and the crafted perturbations are indistinguishable from other noises of the real world. In this experiment, we conduct a survey and it is designed with carefully-designed questions about these adversarial examples. In particular, one survey is composed of 20 generated adversarial examples, randomly selected from our data set. More details can be found here[7]. We release the survey via a public survey service[8], and receive 121 questionnaires in total within three days. We have filtered out 20 surveys of low quality if the survey is finished too fast (less than 60s) or the answers all point to a single choice.

Survey Results. Among the 101 valid surveys, the median age of the participants is 22, 66.34% of them are male and 33.66% are female. 93.07% hold a Bachelor or higher degree. From the survey, we find that 93.56% of the participants can recognize the text of the license plate successfully, which means our adversarial examples do not affect users' recognitions. 8.23% of them do not notice any light spots in adversarial examples, indicating that the perturbations are inconspicuous to them. As for the remaining participants noticing the light spots, 78.32% think the light spots are caused by license plate light or other natural light as we expected, and only 21.68% consider the light spots are from artificial illumination. Thus, we can find out that our practical attack can easily pretend as some normal lighting sources, such as license plate light and the light of other vehicles from the back.

6 Discussion

Potential Defenses for RoLMA. To defend against RoLMA and other alike attacks, we propose the following strategies for LPR systems that are learned in the course of experiments. From the aspect of the recognition algorithm, LPR

[7] https://sites.google.com/view/rolma-adversarial-attack/imperceptibility
[8] https://www.wjx.cn/

systems can employ *denoising* techniques [7] to elevate image quality by eliminating noises added by adversarial examples. Noises in a license plate could be light spots, stains caused by haze or rain, character overlap due to small shooting angles. To overcome these noises, LPR systems are encouraged to sharpen the borders of characters in a low-quality license plate, and the areas out of characters are made consistent with the background. Meanwhile, the stains inside of the characters are colored as the surrounding area. Based on the investigation result of its underlying recognition mechanism, we found that it employs denoising techniques that can crack our perturbations and thus the LPR systems are capable of recognizing the correct text. Besides, training with a variety of adversarial examples can also greatly improve the resistance to future adversarial examples. From the aspect of the system, security experts of the system have to work out more complete and comprehensive protection mechanisms for a specific risky task. Imaging that one car parking management system solely relies on license plate recognition for authentication, attackers can easily break into the car parking system with small efforts committed in case LPR fails or ceases to work. In such a case, multi-factor authentication [13] is a promising method to enhance security. The unique identification code of vehicle which is widely used in the field of IoT can be used in this scenario. Even the car owner changes or heavily scrawls the license plate, the unique identification code can assist in vehicle identification. Moreover, manual checks by specialists are the last obstacles hindering these attacks.

7 Related Work

There are a lot of works on adversarial attacks.

Adversarial attacks against license plate recognition. There are few works on adversarial attacks against LPR systems. For example, Song and Shmatikov [16] explore how the deep learning-based Tesseract [15] system is easily smashed in adversarial settings. They have generated adversarial images to lead a wrong recognition of Tesseract in digital space but not in the practical world. *Unlike the above attack, we are the first one to apply practical adversarial examples in the field of license plate recognition, and implement a full-stack attack from the digital world to the physical world. It helps unveil the weaknesses of modern LPR systems and facilitates the improvement of robustness indirectly.*

Physical implementation of adversarial examples. Although adversarial examples have gained a surprisingly great success in defeating deep learning systems [17], to work in the physical world is not that worrisome [10]. There are emerging research works aiming at making the adversarial attacks come true in reality. In order to generate more robust adversarial attack, Yue Zhao *et al.* [21] proposed the feature-interference reinforcement method and the enhanced realistic constraints generation to enhance robustness. Zhe Zhou *et al.* [23] constructed a new attack against FaceNet with an invisible mask but without the consideration of disturbances from the surrounding environment. Moreover, Xuejing Yuan *et al.* [20] implemented a practical adversarial attack against ASR systems,

working across air in the presence of environmental interferences. In addition, they proposed REEVE attack which can remotely compromise Amazon Echo via radio and TV signals [19]. However, as shown in Section 1-C2, environmental factors can reduce the effectiveness and robustness under different circumstances. Thus, *we design three transformations(e.g., adjust brightness, rescale the image and rotate the image) to simulate the realistic environment in Section 4.2.*

8 Conclusion

We propose the first practical adversarial attack RoLMA against deep learning-based LPR systems. We employ illumination technologies to perturb the license plates captured by LPR systems, rather than making perceivable changes. To resolve a workable illumination solution, we adopt targeted and non-targeted strategies to determine how license plates are illuminated including the color, size, and brightness of light spots. Based on the illumination solution, we design a physical implementation to simulate these light spots on real license plates. We conducted extensive experiments to evaluate the effectiveness of our illumination algorithm and the efficacy of physical implementation. The experiment results show that RoLMA is very effective to deceive HyperLPR with an averaged 93.23% success rate. We have tested RoLMA in the physical world with 91.43% of shoot images are wrongly recognized by HyperLPR.

9 Acknowledgments

IIE authors are supported in part by National Key R&D Program of China (No. 2016QY04W0805), NSFC U1836211, 61728209, 61902395, National Top-notch Youth Talents Program of China, Youth Innovation Promotion Association CAS, Beijing Nova Program, Beijing Natural Science Foundation (No. JQ18011), National Frontier Science and Technology Innovation Project (No. YJKYYQ20170070) and a research grant from Huawei. Fudan university author is supported by NSFC 61802068, Shanghai Sailing Program 18YF1402200.

References

1. License Plate Recognition. https://parking.ku.edu/license-plate-recognition (2018)
2. Vehicle registration numbers and number plates. Tech. Rep. INF104 (2018)
3. Carlini, N., Wagner, D.A.: Audio adversarial examples: Targeted attacks on speech-to-text. In: 2018 IEEE Security and Privacy Workshops. pp. 1–7 (2018). https://doi.org/10.1109/SPW.2018.00009, https://doi.org/10.1109/SPW.2018.00009
4. Evtimov, I., Eykholt, K., Fernandes, E., Kohno, T., Li, B., Prakash, A., Rahmati, A., Song, D.: Robust physical-world attacks on deep learning models. arXiv preprint arXiv:1707.08945 **1** (2017)
5. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. CoRR **abs/1412.6572** (2014)
6. Gravelle, K.: Video tolling system with error checking (2011)

7. Guo, C., Rana, M., Cissé, M., van der Maaten, L.: Countering adversarial images using input transformations. CoRR **abs/1711.00117** (2017), http://arxiv.org/abs/1711.00117
8. Jain, V., Sasindran, Z., Rajagopal, A.K., Biswas, S., Bharadwaj, H.S., Ramakrishnan, K.R.: Deep automatic license plate recognition system. In: Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP). pp. 6:1–6:8 (2016)
9. Li, H., Shen, C.: Reading car license plates using deep convolutional neural networks and lstms. CoRR **abs/1601.05610** (2016), http://arxiv.org/abs/1601.05610
10. Lu, J., Sibai, H., Fabry, E., Forsyth, D.A.: NO need to worry about adversarial examples in object detection in autonomous vehicles. CoRR **abs/1707.03501** (2017), http://arxiv.org/abs/1707.03501
11. Nomura, S., Yamanaka, K., Katai, O., Kawakami, H., Shiose, T.: A novel adaptive morphological approach for degraded character image segmentation. Pattern Recognition **38**(11), 1961–1975 (2005)
12. Papernot, N., McDaniel, P., Jha, S., Fredrikson, M., Celik, Z.B., Swami, A.: The limitations of deep learning in adversarial settings. In: 2016 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 372–387. IEEE (2016)
13. Rosenblatt, S., Cipriani, J.: Two-factor authentication: What you need to know (FAQ). https://www.cnet.com/news/two-factor-authentication-what-you-need-to-know-faq/ (june 2015)
14. Silva, S.M., Jung, C.R.: License plate detection and recognition in unconstrained scenarios. In: Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XII. pp. 593–609 (2018). https://doi.org/10.1007/978-3-030-01258-8_36, https://doi.org/10.1007/978-3-030-01258-8_36
15. Smith, R.: An overview of the tesseract OCR engine. In: 9th International Conference on Document Analysis and Recognition (ICDAR). pp. 629–633 (2007)
16. Song, C., Shmatikov, V.: Fooling OCR systems with adversarial text images. CoRR **abs/1802.05385** (2018)
17. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I.J., Fergus, R.: Intriguing properties of neural networks. CoRR **abs/1312.6199** (2013)
18. Xu, Z., Yang, W., Meng, A., Lu, N., Huang, H.: Towards end-to-end license plate detection and recognition: A large dataset and baseline. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 255–271 (2018)
19. Yuan, X., Chen, Y., Wang, A., Chen, K., Zhang, S., Huang, H., Molloy, I.M.: All your alexa are belong to us: A remote voice control attack against echo. In: 2018 IEEE Global Communications Conference (GLOBECOM). pp. 1–6. IEEE (2018)
20. Yuan, X., Chen, Y., Zhao, Y., Long, Y., Liu, X., Chen, K., Zhang, S., Huang, H., Wang, X., Gunter, C.A.: Commandersong: A systematic approach for practical adversarial voice recognition. In: 27th {USENIX} Security Symposium ({USENIX} Security 18). pp. 49–64 (2018)
21. Yue Zhao, Hong Zhu, R.L.Q.S.S.Z.K.C.: Seeing isnt believing: Towards more robust adversarial attack against real world object detectors. In: Proceedings of the 26th ACM Conference on Computer and Communications Security (CCS) (2019)
22. Zeusee: High Performance Chinese License Plate Recognition Framework (2018)
23. Zhou, Z., Tang, D., Wang, X., Han, W., Liu, X., Zhang, K.: Invisible mask: Practical attacks on face recognition with infrared. CoRR **abs/1803.04683** (2018), http://arxiv.org/abs/1803.04683