What Machines See Is Not What They Get: Fooling Scene Text Recognition Models with Adversarial Text Images

Xing Xu, Jiefu Chen, Jinhui Xiao, Lianli Gao, Fumin Shen, Heng Tao Shen*
Center for Future Media & School of Computer Science and Engineering
University of Electronic Science and Technology of China

Abstract

The research on scene text recognition (STR) has made remarkable progress in recent years with the development of deep neural networks (DNNs). Recent studies on adversarial attack have verified that a DNN model designed for non-sequential tasks (e.g., classification, segmentation and retrieval) can be easily fooled by adversarial examples. Actually, STR is an application highly related to security issues. However, there are few studies considering the safety and reliability of STR models that make sequential prediction. In this paper, we make the first attempt in attacking the state-of-the-art DNN-based STR models. Specifically, we propose a novel and efficient optimization-based method that can be naturally integrated to different sequential prediction schemes, i.e., connectionist temporal classification (CTC) and attention mechanism. We apply our proposed method to five state-of-the-art STR models with both targeted and untargeted attack modes, the comprehensive results on 7 real-world datasets and 2 synthetic datasets consistently show the vulnerability of these STR models with a significant performance drop. Finally, we also test our attack method on a real-world STR engine of Baidu OCR, which demonstrates the practical potentials of our method.

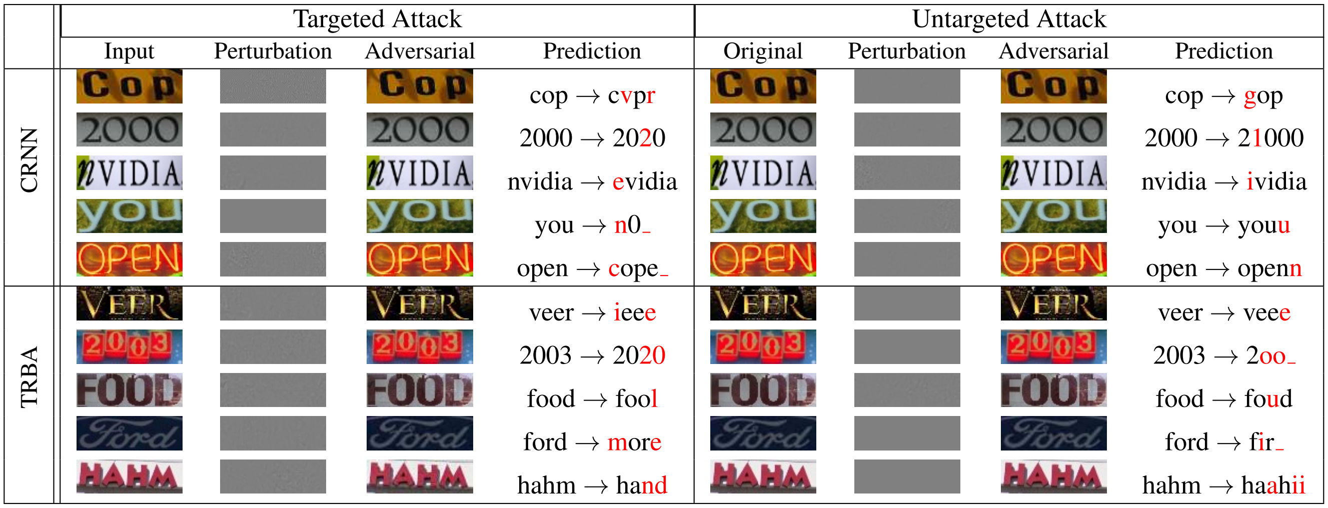
1. Introduction

The scene text recognition (STR) [15, 26, 37, 42] aims at reading sequential characters of varied-length from a text image in natural scene. STR has been an active research field in computer vision because it is a critical element of a lot of real-world applications, such as human computer interaction [59], road sign recognition in the autonomous vehicles [58], assistive reading for the blind and low-vision people [29], etc. Developing accurate and robust STR models has been a research challenge for years, due to the huge diversity in the visual appearance of texts, the illumination artifacts, the complex image background, etc.

The advantages of deep neural networks (DNNs) and their successes in various computer vision tasks have also boosted the development of STR in recent years [4, 20, 41, 55, 2, 56]. Most of these DNN-based works convert the STR tasks into *sequence recognition* (labeling) [50] problems. Specifically, they firstly encode the input image into a feature sequence through a certain encoding technique, such as convolution neural network (CNN) or recurrent neural network (RNN), and then apply decoders, such as connectionist temporal classification (CTC) [14, 42, 18] or attention mechanism [43, 8], to predict the linguistic strings in the image. Despite their current success, recent studies [46, 12, 5] have shown that DNNs are extremely vulnerable to adversarial examples, *i.e.*, by adding small perturbations on the input images. Intuitively, the STR also has this problem since DNN models are prevalent in such a security-critical scenario. For an online STR system, incorrectly recognizing even a single word may possibly change the overall meaning of the text image.

Current studies on adversarial examples mainly focus on non-sequential vision tasks, such as image classification [46, 48, 10], video classification [28, 21], object detection [24, 57], semantic segmentation [49, 33], face recognition [13, 38], etc. Differently, attacking STR models has rarely been explored in the literature since STR is considered as a sequence recognition task and STR models are more difficult to deal with than above non-sequential tasks. Attacking STR models is significantly more challenging due to three major issues: 1) The output of the modern STR models (DNN-based) is a label sequence of varied-length, rather than a single label in the non-sequential attacks (*e.g.*, object classification model). As discussed in [54], the common attack strategies in the non-sequential attacks only involve the *substitution* operation (*e.g.*, modifying the ground-truth class label), while the sequential attacks in STR is expected to consider operations on both character level and word level: *insertion*, *substitution* and *deletion* (*e.g.*, insertion: “horse” ↦ “hoarse”; substitution: “horse” ↦ “house”, “horse” ↦ “zebra”; deletion: “horse” ↦ “hose”). 2) The adversarial examples for attacking STR models should be

[*] Corresponding author.



	Targeted Attack Prediction	Untargeted Attack Prediction
CRNN	cop → cvpr	cop → gop
	2000 → 2020	2000 → 21000
	nvidia → evidia	nvidia → ividia
	you → n0_	you → youu
	open → cope_	open → openn
TRBA	veer → ieee	veer → veee
	2003 → 2020	2003 → 2oo_
	food → fool	food → foud
	ford → more	ford → fir_
	hahm → hand	hahm → haahii

Figure 1: Typical adversarial examples generated by our method to fool two latest STR models: CRNN [42] and TRBA [1] with targeted and untargeted attack modes. Interestingly, we can mislead the models to predict "cvpr", "ieee" and "2020".

guided by linguistic information and each character in the output target sequential labels needs to be well aligned, not just in arbitrary and meaningless sequences of character. 3) The encoder module in STR models usually leverages RNN structures instead of CNNs to capture the sequential context in visual features of text images.

In this paper, we make the first attempt to fool the cutting-edge STR models that are deployed upon DNNs. In principle, we propose a novel and efficient solution for the adversarial attacks on both CTC-based and attention-based STR models. We firstly prove the feasibility of attacking these two kinds of models on the theoretical aspect. Considering the diverse and complex loss functions of these models for sequential labeling task, the popular attack algorithm (*e.g.*, C&W [5], FGSM [12]) designed for non-sequential tasks cannot be used. To this end, we develop a novel optimization-based attacking algorithm that iteratively clips the perturbation value under the constraint. Besides, it allows larger learning rate value with robust training procedure for boosting the attacking speed with much fewer iterations. Our proposed algorithm can be flexibly equipped to the objective functions of different STR models with both targeted and untargeted attack modes. Fig. 1 shows the adversarial examples and perturbations generated by our method, including both untargeted and targeted attack modes, on two state-of-the-art STR models. The perturbations are displayed in gray-scale since images are commonly converted to gray-scale ones in these models. Notably, we further use our adversarial examples to attack the real-world STR system, *i.e.*, Baidu OCR, and observe that the perturbations on original text images can also corrupt the predictions of the commercial STR system.

In summary, our main contributions are:

- We propose a novel and efficient optimization-based adversarial attack approach, which derives generic loss functions for both CTC-based and attention-based models with both targeted and untargeted attack modes. It can learn adversarial examples with robust and efficient training processes. To our best knowledge, this is the very first attempt and comprehensive study on crafting adversarial examples to fool the state-of-the-art STR models.

- We conduct extensive experiments on 7 benchmark datasets for evaluating the attacking effect on five state-of-the-art STR models. Experimental results show that our proposed method attains a remarkable attack success rate when crafting adversarial examples for both targeted and untargeted attacks. In addition, the successful attacking results on the commercial STR system further demonstrate the generalization capability of our proposed method.

2. Related Work

Scene Text Recognition (STR). Most earlier STR methods [37, 47] adopt a bottom-up pipeline, *i.e.*, firstly detecting and recognizing individual characters from certain hand-crafted features, and then linking up the recognized characters into words or text lines via dynamic programming and language models. For the general information of text recognition, readers can refer to the survey [53].

With the advances of DNNs in recent years, some researchers [8, 26, 43] treated the STR task as a sequence learning problem: first encoding a text image into a sequence of features with DNN, then directly generating character sequence with sequence recognition techniques. Note that the recent emergence of CTC [14, 42, 30] and attention mechanism [43, 1] are promising to tackle this sequential training problem by constructing the alignment between the input images and their corresponding label sequence. These CTC-based and attention-based STR models have achieved state-of-the-art performance [1].

Existing studies on STR mainly focus on improving the recognition performance, while rarely considering the issues of model reliability and safety. As aforementioned in Sec. 1, an STR model may be misled to make an incorrect prediction with adversarial examples. To our best knowledge, there are two informally published works [54, 45] that have made an initial step on the adversary of document image recognition in optical character recognition (OCR) area. However, recognizing texts from natural images in the STR area is more challenging. Besides, they only take the simple CTC-based approach [42] as a prototype for discussion. On the contrary, we focus on the reliability of both CTC-based and attention-based STR models and propose a more generic and efficient solution to learn adversarial examples.

Adversarial Examples. In the pioneering work of [46], Szegedy *et al*. have demonstrated that DNNs can be easily attacked by adversarial examples, *i.e.*, by adding minor perturbation that is not noticeable by human eyes. The attack modes can be targeted and untargeted. Taking image classification as an instance, the targeted attack requires a pre-specific label that is expected to be predicted by the classifiers; while the untargeted attack only requires that the prediction of the classifiers differs from the ground-truth without pre-specification.

To attack a model, an adversary is able to fully access the model parameters and training configurations under the white-box setting. In this case, an adversarial example is generated by applying one-step [12] or multiple steps (*e.g.*, I-FGSM [11], MI-FGSM [25]) perturbations on an input image, along with the direction of the adversarial gradient. However, the adversarial gradient may not be accessible to an adversary due to the unknown model parameters, which is called the black-box setting. More recently, learning the universal adversarial perturbation (UAP) [35] has received more attention as this kind of image-agnostic perturbation is able to corrupt most natural images. Several UAP approaches [9, 33, 27] have been developed to learn perturbations based on specific models or training datasets to fool other models or datasets via various schemes such as transfer attack [12, 46] and knowledge distillation [17].

Despite the increasing research attention on learning adversarial examples for the non-sequential tasks, the STR as a sequential recognition task has not been thoroughly explored yet. Recently, a few works have studied the adversarial examples on the other related tasks, such as speech-to-text [6, 52], visual question answering [36, 31] and image caption [7, 51]. However, these tasks have intrinsically different objective functions with the existing models for STR. Therefore, we explicitly focus on the CTC-based and attention-based STR models and propose a unified attack algorithm to efficiently find adversarial examples with both untargeted attack and targeted attack modes.

3. The Proposed Method

3.1. Problem Formulation

We first formally introduce the problem definition to crafting adversarial examples for STR. Given an input scene text image $x \in [-1,1]^{|n|}$ with n normalized pixels, its ground-truth sequence of labels $l = \{l_0, l_1, ..., l_T\}$, where T is the length of the sequence. For a STR model R, our goal is to find an adversarial example $x' = x + \delta$, where δ denotes the adversarial perturbations to x, $x' \in [-1,1]^{|n|}$ ensures it to be a valid input to R. It is expected that x' can guide the R to predict another sequence $l' = \{l'_0, l'_1, ..., l'_{T'}\}$ for x', where l' is different with l (and T' is unnecessarily equal to T), to accomplish the attack on the given image x. Note that x' can be either targeted or untargeted, it depends on the attacking sequence l' is whether pre-specific or not. Finally, generating adversarial example x' for x can be cast as the following optimization problem

$$\begin{aligned} &\min_{x'} L(x', l') + \lambda D(x, x'), \\ &s.t.\ \ x' = x + \delta,\ \ x' \in [-1,1]^{|n|}, \end{aligned} \tag{1}$$

where $R(x) = l$, $R(x') = l'$, $D(x, x') = \|\delta\|_2^2$ is an L_2 distance metric between the original image and the adversarial image. $L(\cdot)$ is an attack loss function which takes different forms in different STR models. λ is a pre-specified hyper-parameter that balances the importance of two terms $L(\cdot)$ and $D(\cdot)$. Intuitively, with smaller λ, the attack is more likely to succeed but with the cost of higher distortion on δ. In our experiment, we use the binary search scheme to select λ. In the following sections, we present the different form of Eq. 1 of our attack method regarding to CTC-based (CTC) and attention-based (Attn) STR models.

3.2. Attack on CTC-based Models

The original CTC [14, 42, 30] provides an alignment-free pipeline for training an end-to-end neural network for sequence labeling tasks. In STR, given an input sequence l of x, the network model R will output a sequential probability distribution $y = \{y_1, y_2, ..., y_M\}$ over the output domain for each character $\{l_i\}_{i=1}^T$ in l, where $M \geq T$. As M is not necessarily the same as T, a valid alignment path is adopted in CTC to remove *blank* and sequentially duplicate characters in the output sequence. For example, $\{c, v, v, p, blank, r\}$ is a valid alignment for $\{c, v, p, r\}$.

Training a CTC-based model requires calculating the probabilities of all possible valid alignment paths for the sequence l. Generally, the probability of one valid alignment path π can be written as $p(\pi|x) = \prod_{t=1}^T y_{\pi_t}^t$, where y_{π_t} is the probability of a valid character in π. Let S be the set of all possible valid alignments for the sequence l, then the

CTC-based models require to calculate the log probability of all the valid alignments given l, as

$$L_{ctc}(x,l) = -\log \sum_{\pi \in S(l)} p(\pi|x). \tag{2}$$

Targeted Attack on CTC-based Models. Suppose we have a targeted sequence of labels l', intuitively, the CTC loss of searching the respective adversarial example x' can be derived as:

$$L_{ctc}(x',l') = -\log \sum_{\pi \in S(l')} p(\pi|x'). \tag{3}$$

Due to the flexible combination of characters in the targeted sequence l', we first expand l' to $l' = \{l'_1, l'_{t_1}, l'_3, ..., l'_{t_k}, ..., l'_{T'}\}$, to highlight the changed k characters $\{l'_{t_1}, .., l'_{t_k}\}$ in l'. Then the probability of a valid path π' over the targeted sequence l' is derived as:

$$p(\pi'|x') = y^1_{\pi'_1} \times y^2_{\pi'_{t_1}} \times ... \times y^k_{\pi'_{t_k}} \times y^{k+1}_{\pi_{k+1}} ... \times y^{T'}_{\pi_{T'}}. \tag{4}$$

As the position of the changed characters in l' is known, we can maximize the probability of the partial sequence before the last changed characters l'_{t_k} to accomplish the attack, which is written as:

$$\max p(\pi'|x') = \max(y^1_{\pi'_1} \times y^2_{\pi'_2} \times ... \times y^k_{\pi'_{t_k}}) \times \prod_{t=k+1}^{T'} y_{\pi^{T'}_t}. \tag{5}$$

We notice that $\prod_{t=k+1}^{T'} y_{\pi^{T'}_t}$ is fixed as the characters after l'_{t_k} is unchanged, then Eq. 5 is further simplified as

$$\max p(\pi^{'}|x^{'}) \propto \max(y^1_{\pi'_1} \times y^2_{\pi'_2} \times ... \times y^k_{\pi'_{t_k}}). \tag{6}$$

According to the above Eq. 6 and the CTC loss term for the targeted sequence l' in Eq. 3, the final objective function for finding adversarial example x' is formulated as

$$\min_{x'}((-\log \sum_{\pi' \in S(l')} \max p(\pi'|x')) + \lambda D(x,x')). \tag{7}$$

Untargeted Attack on CTC-based Models Different from the targeted attack, the untargeted attack only requires the attacking sequence l' is not equal to the input sequence l. As the definition of probability of a valid path π for l, here we can find another path π' that reduces (minimizes) the probability of π to find the adversarial example x' that accomplishes the untargeted attack. The probability of $p(\pi'|x')$ can be derived as

$$p(\pi'|x') = \min(y^1_{\pi_1} \times y^2_{\pi_2} \times ...y^{i+1}_{\pi_{i+1}} \times ...y^T_{\pi_T}), \tag{8}$$

where $\{y_{\pi_i}\}_{i=1}^T$ are probabilities of characters in valid path π.

According to the above Eq. 8 and the general CTC loss term in Eq. 3, the final objective function for finding adversarial example x' with untargeted attack is formulated as

$$\min_{x'}((-\log \sum_{\pi' \in S(l')} \min p(\pi'|x')) + \lambda D(x,x')). \tag{9}$$

3.3. Attack on Attention-based Models

In encoder-decoder networks with attention mechanism *e.g.*, [43, 1], suppose we have a ground-truth sequence $l = \{l_1, l_2, ..., l_t, ..., l_T\}$ of an input image x, where l_t indicates the index of the t-th character in the vocabulary list V. The network model R commonly contains a RNN/LSTM/BiLSTM cell $f(\cdot)$ that outputs the probability $p(l_t|x, l_1, ..., l_{t-1})$ of the t-th character l_t in l according to its hidden state h_{t-1} previous character l_{t-1}, as

$$p_t = softmax(z_t),\ and\ z_t = f(h_{t-1}, l_{t-1}), \tag{10}$$

where $z_t := \{z^1_t, z^2_t, ..., z^{|V|}_t\} \in \mathbb{R}^{|V|}$ is a vector of the logits (unnormalized probabilities) for each possible character in the vocabulary. The vector p_t represents a probability distribution on V with each coordinate p^i_t defined as $p^i_t := p(l_t = i|x, l_1, ..., l_{t-1})$, $i \in [1, |V|]$. Note that the attention matrix is implicitly incorporated when computing p_t, thus the Eq. 10 is a general form in the attention-based models. Following the definition of softmax function:

$$p(l_t|x, l_1, ..., l_{t-1}) = \exp(p^{l_t}_t) / \sum_{i \in V} \exp(p^i_t). \tag{11}$$

To maximize the probability of the ground-truth sequence l, we can directly take its negative log probability $-\log p(l|x)$ as a loss function, which can be formulated as:

$$L_{att}(x,l) = -\log p(l|x) = -\sum_{t=2}^{T} \log p(l_t|x, l_1, ..., l_{t-1}). \tag{12}$$

Here the Eq. 11 can be directly applied to compute the last log term in Eq. 12.

Targeted Attack on Attention-based Models. Similar as the case of the CTC-based models, we also expand the targeted sequence l' to $l' = \{l'_1, l'_{t_1}, l'_3, ..., l'_{t_k}, ..., l'_{T'}\}$, to explicitly show the changed k characters $\{l'_{t_1}, .., l'_{t_k}\}$ in l' comparing with ground-truth sequence l. Considering the attention mechanism, to achieve the attack effect, it is expected to shift attention to other "incorrect" characters during the decoding procedure.

To simplify the derivation, we first discuss the case of only one character l'_{t_1} is changed in l'. To maximize the probability $p(l'|x')$ of the targeted sequence l', finding the adversarial image x' can be formulated according to the above Eq. 12 as:

$$\begin{aligned} L_{att}(x^{'}, l^{'}) &= -\sum_{t=2}^{T'} \log p(l'_t|x', l'_1, ..., l'_{t_1}, ..., l'_{t-1}) \\ &\approx -\sum_{t=2, t \neq t_1}^{T'} \log p(l'_t|x', l'_1, ..., l'_{t_1}, ..., l'_{t-1}) \\ &\quad - \log p(l'_t|x', l'_1, ..., l'_{t_1}). \end{aligned} \tag{13}$$

For the more general case of k changed characters $\{l'_{t_1}, ..., l'_{t_k}\}$ in l', we can still derive the similar form as

Eq. 13 by dividing l' into $k+1$ parts where continuously unchanged characters are grouped. Note that in Eq. 13, the first term is a constant value given the unchanged characters in l' compared with l, then the final targeted attack loss function can be rewritten as:

$$\min L_{att}(x', l') \propto \max\left(-\log p(l'_t | x', x', l'_1, ..., l'_{t_1})\right). \tag{14}$$

Applying Eq. 14 to the general form of Eq. 1 , the final objective formula of the targeted attack on attention-based models is:

$$\min\left(\max(-\log p(l'_t | x, l'_1, ..., l'_{t_1})) + \lambda D(x, x')\right). \tag{15}$$

Untargeted Attack on Attention-based Models. To accomplish the attack, we adopt the similar strategy as for the CTC models, *i.e.*, reducing the probability of each character in the ground-truth $l = \{l_1, l_2, ..., l_T\}$ to find the adversarial example x'. If any character in l obtains lower probability, l will be attacked and changed to an untargeted sequence l'. According to the probability definition in Eq. 12, we can derive the loss term for (x', l') as:

$$L_{att}(x', l') = -\log p(l' | x') = -\sum_{t=2}^{T'} \log p(l'_t | x', l'_1, ..., l'_{t-1}),$$
$$s.t. \log p\left(l'_t | x', l'_1, ..., l'_{t-1}\right) \neq \log p\left(l_t | x, l_1, ..., l_{t-1}\right), \exists t \in T. \tag{16}$$

Here the index t can be any position according to l. Applying Eq. 16 to Eq. 1 , the final optimization formula of untargeted attack on attention-based model is:

$$\min\left(\max(-\log p(l' | x') + \lambda D(x, x')\right). \tag{17}$$

3.4. The Optimization

According to the objective functions Eq. 7, Eq.9, Eq.15, and Eq. 17 of (un)targeted attacks on CTC and attention-based models, we adopt the stochastic gradient descent (SGD) algorithm to update the perturbation vector δ iteratively. Finally the perturbations are added on the original images to generate adversarial examples. The detailed optimization algorithm is depicted in Alg. 1.

4. Experiments

4.1. Experimental Setup

STR Models. Five state-of-the art STR models are selected as the targets for adversarial attack, including three CTC-based models (*i.e.*, CRNN [42], Rosetta [3], STAR-Net [30]) and two attention-based ones (*i.e.*, RARE [43], TRBA [1]). As summarized in [1], these STR models adopts different DNN network architectures of VGG [44] and ResNet [16] for visual feature extraction. In addition, Bidirectional LSTM (Bi-LSTM) is used as the (de)-selection in sequence modeling; CTC and attention scheme (Attn) are employed for sequence prediction.

Algorithm 1 The detailed procedure of our method to attack STR models.

```
Input: Original image $x$, target sequence $l'$, a STR model $R(\cdot)$, with attacking objective function $L(\cdot)$, attack mode $m$, learning rate $\mu$;
if $l' \neq$ NULL then
    $m \leftarrow -1$  // Targeted attack
else
    $m \leftarrow 1$  // Untargeted attack
end if
Initialize $\delta \leftarrow 0$
repeat
    $g \leftarrow m \nabla_{\delta} L(x', l')$,
    $g \leftarrow \frac{g}{\|g\|_2}$,
    $\delta \leftarrow \delta + \mu * g$,
    Update $\mu$ by learning rate annealing,
    $\delta \leftarrow clip(\delta)$
    $x' \leftarrow x + \delta$
until $R(x') = l$
Output: Adversarial image $x'$.
```

Datasets. Since it is costly to obtain enough labeled scene text images in real scenarios, most STR models use synthetic data for training. The MJSynth [19] (MJ) and SynthText [15] (ST) are two widely-used synthetic datasets designed for STR, which contain 8.9 and 5.5 million word box images, respectively. Unlike the prior works that have used diverse combinations of the two datasets, Baek *et al.* [1] suggest unifying the two datasets to avoid inconsistent and unfair comparison. Therefore, we follow their settings and use a combination of the two datasets (*i.e.*, MJ+ST) as our training data, which contains 14.4 million images in total.

Moreover, 7 real-world STR datasets are used for evaluating a trained STR model, they are CUTE80 [40], ICDAR2003 [32], ICDAR2013 [23], ICDAR2015 [22], IIIT5K-Words (IIIT5K) [34], Street View Text (SVT) [47], and SVT Perspective (SP) [39]. All these datasets have been fairly evaluated in the latest work of [1]. Besides, we also build 2 synthetic test datasets by randomly selecting 4000 images in each of MJ and ST datasets in our experiment.

Implementation Details. All pre-trained scene text recognition models are trained based on MJ+ST, and all model parameters are set to be the best values which are released by [1]. We recalculate the accuracy of all pre-trained models to ensure the reliability of our experiment. We employ the PyTorch toolkit to implement our method and all the experiments are conducted on a desktop with one GeForce GTX 1080 Ti GPU. In general, these STR models originally adopt the Adam optimizer with the learning rate as 0.005 for model training. For our attacking method, we use SGD to optimize the objective functions of all STR models and adopt a learning rate annealing strategy to reduce learning rate from 0.1 to 0.01, since our method can support large learning rate with stable training procedure and fast attacking speed. For the parameter λ, we adopt a binary search for $\lambda \in [10^{-3}, 10^{4}]$ and take an early-stop strategy to avoid unnecessary iterations. The detailed analysis of our model parameters is shown in the ablation study.

Attack Setting and Evaluation Metric. We conduct both targeted and untargeted attacks on each test dataset. As the attacking sequence is optional to make *insertion*, *substitution*, and *deletion* of the groundtruth sequence, we use the edit distance to measure the difference between the two sequences. Specifically, for the untargeted attack, since the attacking sequence is arbitrary, the edit distance is not fixed. On the contrary, as the attacking sequence is pre-specific, the edit distance is fixed, *e.g.*, 1, 2, 3, etc, depending on the length of the groundtruth sequence. In our experiment, we run the attack to find an adversarial example until the edit distance is not 0 (*i.e.*, attack succeeds). For all datasets, we set the maximum adversarial perturbations magnitude with clip norm to 0.2. We use the widely-adopted metrics of 1) success rate (**SR**), the ratio of successful generation of adversarial examples under the perturbation bound within the limited number of iterations; 2) the averaged L_2 distance (**Dist**) between the input images and the generated adversarial examples; and 3) average number of iterations (**Iter**), required for a successful attack (excluding failed attacks).

4.2. Overall Results

Results on Targeted Attack. We first evaluate the targeted attack against five STR models on all testing datasets. We consider a typical targeted attack case, *i.e.*, the targeted sequence has a 2-Edit distance with the groundtruth sequence by the *substitute* operation. The results in terms of original prediction accuracy, attack success rate, average difference and average iterations can be found in Table 1. We can clearly observe that all the STR models are vulnerable to adversarial examples, as the SRs on them reach almost 100%. For the models with larger recognition accuracy, *e.g.*, STAR-Net and TRBA, more number of iterations are needed to accomplish the attack to targeted sequence as they have more complicate and deeper network architecture. In general, our proposed method needs smaller number of iterations (*e.g.*, 20-50) for the targeted attack on all datasets, which is remarkably efficient than traditional attack algorithms such as C&W [5] and the reported results in [54] that usually needs hundreds (even thousands) of iterations. Moreover, our method also achieves low distances (around 1.0) of perturbations for all models on all datasets. In most cases, the STAR-Net model has smaller perturbations than the other models, indicating that the adversarial examples generated for it are more similar to the original images.

Results on Untargeted Attack. We then report the results of untargeted attack against all STR models in Table 2, under the same case of the above targeted attack. Similar to the above results on the targeted attack with 2-Edit distance, here all models are easily fooled by our attack method with nearly 100% success rate. Moreover, the required numbers of iterations for attacking all models are remarkably less than those of the targeted attack case, *i.e.*, less than 10 iterations in general. Besides, the distances for all models are also much smaller (less than 1.0), indicating that just slight perturbations on the original images of all datasets would lead to incorrect predictions.

Results on Transfer Attack. The transfer attack is to fool models or datasets with a perturbation generated on another model or dataset. Since the STR models are trained on synthetic dataset MJ+ST and tested on real-world datasets, therefore, the above untargeted and targeted attacks can also belong to the *cross-dataset* transfer attack. The results in both Table 1 and Table 2 demonstrate that our attack method can effectively achieve the cross-dataset transfer attack. Furthermore, we conduct experiments to investigate the *cross-model* transfer attack, where we use the adversarial examples generated from one STR model to fool another model. Table 3 shows the results in terms of mean SR score for the cross-model transfer attack across five different STR models on 7 real-world datasets. Each row in the table shows the SRs for perturbations crafted by a given STR model, and each column shows the succeeded rates on another model. We can see that the attack across pairwise models is asymmetric, showing the diverse properties of each model on adversarial examples generated by another model. The CRNN model obtains the best average SR score while the state-of-the-art TRBA method obtains the worst scores. It indicates that CRNN can generate more effective adversarial examples to fool the other models though it has much simpler network architecture. Moreover, as different feature extraction schemes (*e.g.*, VGG and ResNet) and prediction schemes (CTC and Attn) are used in the STR models, they also have an effect on the results of cross-model transfer attack.

4.3. Further Analysis

Visualization of Perturbations. In this experiment, we investigate the changes on the learned perturbations during optimization in our method. We choose two models: CRNN (CTC-based) and TRBA (attention-based) as prototypes to generate adversarial examples on the IIIT5K and CUTE80 datasets. Fig. 2 visualizes the generated perturbations and the corresponding adversarial examples in different steps of the optimization procedure. We can see that at the beginning (*e.g.*, iteration 1), the perturbations are too weak to change the prediction results, with more iterations (*e.g.*, 20-80), the prediction results are changed with more effective perturbations. In practice, our method will stop iteration around 20 as the attack has accomplished.

Effect of Different Operations in Targeted Attack. Indeed, a sequence can be modified with various operations: insertion, substitution and deletion in the targeted attack. To fully explore these operations, we again use the edit dis-

	CUTE08 (247 images)				ICDAR03 (867 images)				ICDAR13 (857 images)			
	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓
CRNN (TPAMI'17)	65.5	100	1.20	14.6	92.6	100	1.33	16.0	91.0	99.94	1.35	16.7
Rosetta (KDD'18)	69.2	99.19	1.15	24.5	92.9	99.82	1.19	26.5	90.9	99.84	1.21	29.9
STAR-Net (BMVC'16)	71.7	100	1.03	26.2	94.0	100	1.08	22.3	92.8	99.94	1.12	24.5
RARE (CVPR'16)	64.0	99.59	1.27	16.9	91.2	99.91	1.41	18.2	69.4	99.94	1.43	18.6
TRBA (ICCV'19)	74.0	99.19	1.24	47.9	94.4	99.82	1.25	47.2	93.6	99.79	1.26	48.6
	IIIT5K (2556 images)				SVT (647 images)				SVT-P (645 images)			
	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓
CRNN (TPAMI'17)	82.9	99.93	1.25	16.1	81.6	99.94	1.22	15.7	70.0	99.94	1.20	15.2
Rosetta (KDD'18)	84.3	99.59	1.12	28.7	84.7	99.63	1.10	27.6	73.8	99.66	1.08	26.6
STAR-Net (BMVC'16)	87.0	99.86	1.07	24.1	86.9	99.87	1.05	23.3	77.5	99.88	1.04	22.5
RARE (CVPR'16)	81.7	99.84	1.27	16.8	80.8	99.85	1.25	16.3	69.4	99.87	1.23	15.7
TRBA (ICCV'19)	87.9	99.36	1.21	45.1	87.5	99.42	1.20	43.9	79.2	99.47	1.19	42.9
	ICDAR15 (1927 images)				MJ (4000 images)				ST (4000 images)			
	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓
CRNN (TPAMI'17)	69.4	99.97	1.12	13.6	93.9	99.59	1.36	18.6	94.8	100	1.34	16.9
Rosetta (KDD'18)	71.2	99.92	1.02	23.4	95.2	99.39	1.33	61.1	95.7	100	1.12	25.4
STAR-Net (BMVC'16)	76.1	99.97	0.96	19.4	94.9	99.39	1.26	41.6	97.1	100	1.07	27.1
RARE (CVPR'16)	70.6	99.97	0.96	19.4	88.8	100	1.62	26.8	87.3	100	1.20	13.9
TRBA (ICCV'19)	77.6	99.87	1.12	39.2	96.1	99.61	1.27	51.7	97.3	99.97	1.10	34.4

Table 1: Results of targeted attack on 7 real-world datasets and 2 synthetic datasets against five state-of-the-art STR models.

	CUTE08 (247 images)				ICDAR03 (867 images)				ICDAR13 (857 images)			
	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓
CRNN (TPAMI'17)	65.5	99.19	0.64	5.4	92.6	94.91	1.25	10.2	91.0	96.14	1.00	7.7
Rosetta (KDD'18)	69.2	100	0.25	1.6	92.9	99.76	0.31	2.4	90.9	99.88	0.33	3.1
STAR-Net (BMVC'16)	71.7	100	0.45	3.6	94.0	99.88	0.64	6.0	92.8	99.76	0.69	6.4
RARE (CVPR'16)	64.0	100	0.29	2.0	91.2	100	0.43	3.3	69.4	99.88	0.42	3.3
TRBA (ICCV'19)	74.0	100	0.39	3.0	94.4	99.76	0.56	5.1	93.6	99.88	0.57	5.1
	IIIT5K (2556 images)				SVT (647 images)				SVT-P (645 images)			
	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓
CRNN (TPAMI'17)	82.9	98.32	0.89	9.3	81.6	99.38	0.58	3.8	70.0	99.53	0.47	3.1
Rosetta (KDD'18)	84.3	99.88	0.32	2.3	84.7	100	0.27	1.8	73.8	100	0.25	1.6
STAR-Net (BMVC'16)	87.0	99.49	0.70	6.3	86.9	100	0.50	4.3	77.5	99.84	0.42	3.7
RARE (CVPR'16)	81.7	99.96	0.43	3.3	80.8	100	0.41	3.3	69.4	100	0.37	2.8
TRBA (ICCV'19)	87.9	99.96	0.56	5.1	87.5	100	0.43	3.8	79.2	100	0.35	2.9
	ICDAR15 (1927 images)				MJ (4000 images)				ST (4000 images)			
	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓	Acc ↑	SR ↑	Dist ↓	Iter ↓
CRNN (TPAMI'17)	69.4	99.84	0.43	3.0	93.9	99.09	0.89	7.4	94.8	92.26	1.15	12.9
Rosetta (KDD'18)	71.2	100	0.25	1.6	95.2	99.95	0.23	1.6	95.7	99.74	0.41	4.3
STAR-Net (BMVC'16)	76.1	100	0.38	3.2	94.9	99.87	0.62	6.0	97.1	99.77	0.56	5.5
RARE (CVPR'16)	70.6	100	0.35	2.6	88.8	99.93	0.23	1.4	87.3	99.51	0.63	5.4
TRBA (ICCV'19)	77.6	100	0.33	2.6	96.1	100	0.49	4.3	97.3	100	0.44	4.1

Table 2: Results of untargeted attack on 7 real-world datasets and 2 synthetic datasets for five state-of-the-art STR models.

	CRNN	Rosetta	STAR-Net	RARE	TRBA	Avg.
CRNN	-	25.11	16.74	33.33	15.96	22.79
Rosetta	11.93	-	9.14	13.48	6.20	10.18
STAR-Net	15.19	16.27	-	19.37	9.76	15.14
RARE	16.27	14.41	10.38	-	9.14	12.55
TRBA	13.64	14.41	9.45	15.81	-	13.32

Table 3: Results of cross-model transfer attack.

tance to measure the difference between the targeted sequence and the groundtruth sequence. Table 4 shows the attacking results of our method against five STR models on the ICDAR15 dataset by specifying the targeted sequence with 1 and 2-Edit distance, respectively. We can see that our method is stable with different operations and keep the attacking SR scores on all models with fast attacking speed. In the 2-Edit distance case, to keep the SR score, our method needs more iterations to accomplish the attack and the average distance is larger. The reason is that the targeted sequence has more inequality with the groundtruth, the perturbations are respectively more difficult to be learned.

	1-Edit distance			2-Edit distance		
	SR ↑	Dist ↓	Iter ↓	SR ↑	Dist ↓	Iter ↓
CRNN	99.97	1.11	14.1	99.87	1.43	26.1
Rosetta	99.87	0.97	25.8	99.84	1.25	44.9
Star-Net	99.97	0.94	22.7	99.92	1.22	46.0
RARE	99.97	1.10	13.9	99.97	1.36	19.3
TRBA	99.76	1.16	42.1	99.51	1.41	72.2

Table 4: The results of different operations in our targeted attack with 1, 2-Edit distance.

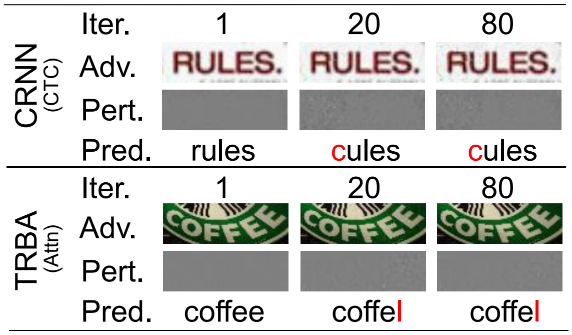
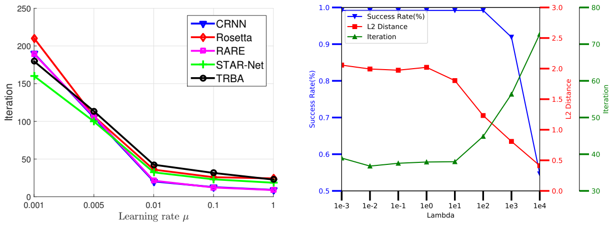


Figure 2: Visualization of the perturbations and adversarial examples obtained by our method in different optimization steps on IIIT5K (top) and CUTE80 (bottom) datasets.

Effect of Model Parameters. We further assess the effect of our model parameters: the learning rate μ and trade-off coefficient λ in our method. We take the ICDAR15 dataset as a testbed and evaluate our method on all five STR models for the untargeted attack. Fig. 3(a) shows the numbers of iterations with different μ on all methods. It can be seen that with larger μ, fewer iterations (*e.g.*, 20-30) are required to accomplish the attack. Note that, using large μ (*e.g.*, [0.1, 1]) in existing models, such as C&W and [54], may lead to collapse. However, our method is able to use a larger value of μ . Furthermore, Fig. 3(b) illustrates the change of perturbations with different λ. Indeed, λ controls the importance of both perturbations and the sequence decoding in the attack objective functions. With larger λ, smaller perturbations are expected, which may lead to the failure attack due to the negligible difference between the generated adversarial examples and the original images. In practice, $\lambda \in [0.01, 10]$ achieves the best attacking results.

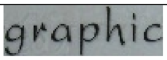
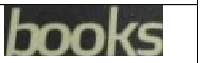


(a) Effect of μ

(b) Effect of λ on TRBA model

Figure 3: The effect of parameter μ and λ in our method for all STR models on CUTE80 dataset.

Attack on Real-world STR System. Finally, we investigate the generated adversarial examples by our method on attacking a real-world commercial STR system, *i.e.*, Baidu OCR. In particular, we select 800 images in total from the 7 real-world datasets to generate their adversarial examples using CRNN and TRBA models, then we use the API toolkit (https://cloud.baidu.com/doc/OCR/OCR-API.html) to make prediction. Fig. 4 shows the overall success rate of the targeted and untargeted attacks on the STR system. We can observe that the system has a considerably high risk to be attacked, as the SR is more than 20%. Moreover, two typical adversarial examples presented in the table show that the predicted results of these perturbed images are completely different from the original ones. Another potential reason is that the characters' vocabulary of the Baidu OCR system may be different from the ones we used for the original CRNN and TRBA models. Nevertheless, this experiment again indicates building a real-world commercial STR system also needs to consider the issue of reliability for more robust recognition.

	Targeted Attack	Untargeted Attack
CRNN	 wrappers → wrapper_	 logistic → _ogistIC
TRBA	 graphic → 9raphic	 books → _OOKS

CRNN	26.70	22.31
TRBA	25.05	20.90

Figure 4: Typical adversarial examples tested on Baidu OCR (top panel) and the overall attacking results (SR) of the two STR models (bottom panel).

5. Conclusion

In this paper, we are the first to propose a generic and efficient attack methods against scene text recognition (STR). We firstly derived the objective functions for attacking both CTC-based and attention-based models with targeted and untargeted attack modes. We then conducted extensive experiments to evaluate our proposed attack method on 7 real-world datasets, 2 synthetic datasets as well as a commercial STR system (*i.e.*, Baidu OCR), in which our method consistently has shown high attack performance and almost completely fooled five state-of-the-art STR models with high efficiency. Our work can therefore serve as an inspiration in designing more robust and secure STR models against the proposed attack schemes.

Acknowledgements. This work was supported in part by the National Key Research and Development Program of China under grant 2018AAA0102200; the National Natural Science Foundation of China under grants 61976049, 61632007 and 61872064; the Sichuan Science and Technology Program, China, under grants 2019ZDZX0008, 2019YFG0003 and 2018GZDZX0032. Jiefu Chen and Jinhui Xiao were with the internships at Afanti AI Lab (Beijing, China) when this work was performed.

References

[1] Jeonghun Baek, Geewook Kim, Junyeop Lee, Sungrae Park, Dongyoon Han, Sangdoo Yun, Seong Joon Oh, and Hwalsuk Lee. What is wrong with scene text recognition model comparisons? dataset and model analysis. In *IEEE International Conference on Computer Vision*, pages 652–661, 2019.

[2] Fan Bai, Zhanzhan Cheng, Yi Niu, Shiliang Pu, and Shuigeng Zhou. Edit probability for scene text recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 1508–1516, 2018.

[3] Fedor Borisyuk, Albert Gordo, and Viswanath Sivakumar. Rosetta: Large scale system for text detection and recognition in images. In *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD*, pages 71–79, 2018.

[4] Michal Busta, Lukas Neumann, and Jiri Matas. Deep textspotter: An end-to-end trainable scene text localization and recognition framework. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2223–2231, 2017.

[5] Nicholas Carlini and David A. Wagner. Towards evaluating the robustness of neural networks. In *IEEE Symposium on Security and Privacy, SP*, pages 39–57, 2017.

[6] Nicholas Carlini and David A. Wagner. Audio adversarial examples: Targeted attacks on speech-to-text. In *IEEE Security and Privacy Workshops, SP Workshops*, pages 1–7, 2018.

[7] Hongge Chen, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, and Cho-Jui Hsieh. Attacking visual language grounding with adversarial examples: A case study on neural image captioning. In *Proceedings of the Association for Computational Linguistics, ACL*, pages 2587–2597, 2018.

[8] Zhanzhan Cheng, Fan Bai, Yunlu Xu, Gang Zheng, Shiliang Pu, and Shuigeng Zhou. Focusing attention: Towards accurate text recognition in natural images. In *IEEE International Conference on Computer Vision, ICCV*, pages 5086–5094, 2017.

[9] Wenjie Ding, Xing Wei, Xiaopeng Hong, Rongrong Ji, and Yihong Gong. Universal adversarial perturbations against person re-identification. In *IEEE International Conference on Computer Vision*, pages 4635–4643, 2019.

[10] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 9185–9193, 2018.

[11] Yinpeng Dong, Fangzhou Liao, Tianyu Pang, Hang Su, Jun Zhu, Xiaolin Hu, and Jianguo Li. Boosting adversarial attacks with momentum. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 9185–9193, 2018.

[12] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations, ICLR*, 2015.

[13] Gaurav Goswami, Nalini K. Ratha, Akshay Agarwal, Richa Singh, and Mayank Vatsa. Unravelling robustness of deep learning based face recognition against adversarial attacks. In *AAAI Conference on Artificial Intelligence, AAAI*, pages 6829–6836, 2018.

[14] Alex Graves, Santiago Fernández, Faustino J. Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *International Conference on Machine Learning ICML*, pages 369–376, 2006.

[15] Ankush Gupta, Andrea Vedaldi, and Andrew Zisserman. Synthetic data for text localisation in natural images. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 2315–2324, 2016.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 770–778, 2016.

[17] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531, 2015.

[18] Kyuyeon Hwang and Wonyong Sung. Sequence to sequence training of ctc-rnns with partial windowing. In *International Conference on Machine Learning, ICML*, pages 2178–2187, 2016.

[19] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition. *CoRR*, abs/1406.2227, 2014.

[20] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep structured output learning for unconstrained text recognition. In *International Conference on Learning Representations, ICLR*, 2015.

[21] Linxi Jiang, Xingjun Ma, Shaoxiang Chen, James Bailey, and Yu-Gang Jiang. Black-box adversarial attacks on video recognition models. In *ACM International Conference on Multimedia, MM*, pages 864–872, 2019.

[22] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman K. Ghosh, Andrew D. Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, Faisal Shafait, Seiichi Uchida, and Ernest Valveny. ICDAR 2015 competition on robust reading. In *International Conference on Document Analysis and Recognition, ICDAR*, pages 1156–1160, 2015.

[23] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez i Bigorda, Sergi Robles Mestre, Joan Mas, David Fernández Mota, Jon Almazán, and Lluís-Pere de las Heras. ICDAR 2013 robust reading competition. In *International Conference on Document Analysis and Recognition, ICDAR 2013*, pages 1484–1493, 2013.

[24] Edgar Kaziakhmedov, Klim Kireev, Grigorii Melnikov, Mikhail Pautov, and Aleksandr Petiushko. Real-world attack on MTCNN face detection system. *CoRR*, abs/1910.06261, 2019.

[25] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial examples in the physical world. In *International Conference on Learning Representations, ICLR, Workshop Track Proceedings*, 2017.

[26] Chen-Yu Lee and Simon Osindero. Recursive recurrent nets with attention modeling for OCR in the wild. In *IEEE Con-*

ference on Computer Vision and Pattern Recognition, CVPR, pages 2231–2239, 2016.

[27] Jie Li, Rongrong Ji, Hong Liu, Xiaopeng Hong, Yue Gao, and Qi Tian. Universal perturbation attack against image retrieval. In *IEEE International Conference on Computer Vision*, pages 1326–1335, 2019.

[28] Shasha Li, Ajaya Neupane, Sujoy Paul, Chengyu Song, Srikanth V. Krishnamurthy, Amit K. Roy-Chowdhury, and Ananthram Swami. Stealthy adversarial perturbations against real-time video classification systems. In *Annual Network and Distributed System Security Symposium, NDSS*, 2019.

[29] Minghui Liao, Jian Zhang, Zhaoyi Wan, Fengming Xie, Jiajun Liang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Scene text recognition from two-dimensional perspective. In *AAAI Conference on Artificial Intelligence*, pages 8714–8721, 2019.

[30] Wei Liu, Chaofeng Chen, Kwan-Yee K. Wong, Zhizhong Su, and Junyu Han. Star-net: A spatial attention residue network for scene text recognition. In *British Machine Vision Conference, BMVC*, 2016.

[31] Jiaying Lu, Xin Ye, Yi Ren, and Yezhou Yang. Good, better, best: Textual distractors generation for multi-choice VQA via policy gradient. *CoRR*, abs/1910.09134, 2019.

[32] Simon M. Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, and Robert Young. ICDAR 2003 robust reading competitions. In *International Conference on Document Analysis and Recognition (ICDAR)*, pages 682–687, 2003.

[33] Jan Hendrik Metzen, Mummadi Chaithanya Kumar, Thomas Brox, and Volker Fischer. Universal adversarial perturbations against semantic image segmentation. In *IEEE International Conference on Computer Vision, ICCV*, pages 2774–2783, 2017.

[34] Anand Mishra, Karteek Alahari, and C. V. Jawahar. Scene text recognition using higher order language priors. In *British Machine Vision Conference, BMVC*, pages 1–11, 2012.

[35] Konda Reddy Mopuri, Aditya Ganeshan, and R. Venkatesh Babu. Generalizable data-free objective for crafting universal adversarial perturbations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(10):2452–2465, 2019.

[36] Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. Did the model understand the question? In *Proceedings of the Association for Computational Linguistics, ACL*, pages 1896–1906, 2018.

[37] Lukas Neumann and Jiri Matas. Real-time lexicon-free scene text localization and recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(9):1872–1885, 2016.

[38] Mikhail Pautov, Grigorii Melnikov, Edgar Kaziakhmedov, Klim Kireev, and Aleksandr Petiushko. On adversarial patches: real-world attack on arcface-100 face recognition system. *CoRR*, abs/1910.07067, 2019.

[39] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *IEEE International Conference on Computer Vision, ICCV*, pages 569–576, 2013.

[40] Anhar Risnumawan, Palaiahnakote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images. *Expert Syst. Appl.*, 41(18):8027–8048, 2014.

[41] José A. Rodríguez-Serrano, Albert Gordo, and Florent Perronnin. Label embedding: A frugal baseline for text recognition. *International Journal of Computer Vision*, 113(3):193–207, 2015.

[42] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(11):2298–2304, 2017.

[43] Baoguang Shi, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Robust scene text recognition with automatic rectification. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4168–4176, 2016.

[44] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations, ICLR*, 2015.

[45] Congzheng Song and Vitaly Shmatikov. Fooling OCR systems with adversarial text images. *CoRR*, abs/1802.05385, 2018.

[46] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian J. Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations, ICLR*, 2014.

[47] Kai Wang, Boris Babenko, and Serge J. Belongie. End-to-end scene text recognition. In *IEEE International Conference on Computer Vision, ICCV*, pages 1457–1464, 2011.

[48] Chaowei Xiao, Dawei Yang, Bo Li, Jia Deng, and Mingyan Liu. Meshadv: Adversarial meshes for visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 6898–6907, 2019.

[49] Cihang Xie, Jianyu Wang, Zhishuai Zhang, Yuyin Zhou, Lingxi Xie, and Alan L. Yuille. Adversarial examples for semantic segmentation and object detection. In *IEEE International Conference on Computer Vision, ICCV*, pages 1378–1387, 2017.

[50] Zecheng Xie, Yaoxiong Huang, Yuanzhi Zhu, Lianwen Jin, Yuliang Liu, and Lele Xie. Aggregation cross-entropy for sequence recognition. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 6538–6547, 2019.

[51] Yan Xu, Baoyuan Wu, Fumin Shen, Yanbo Fan, Yong Zhang, Heng Tao Shen, and Wei Liu. Exact adversarial attack to image captioning via structured output learning with latent variables. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 4135–4144, 2019.

[52] Zhuolin Yang, Bo Li, Pin-Yu Chen, and Dawn Song. Characterizing audio adversarial examples using temporal dependency. In *International Conference on Learning Representations, ICLR*, 2019.

[53] Qixiang Ye and David S. Doermann. Text detection and recognition in imagery: A survey. *IEEE Trans. Pattern Anal. Mach. Intell.*, 37(7):1480–1500, 2015.

[54] Xiaoyong Yuan, Pan He, and Xiaolin Andy Li. Adaptive adversarial attack on scene text recognition. *CoRR*, abs/1807.03326, 2018.

[55] Fangneng Zhan and Shijian Lu. ESIR: end-to-end scene text recognition via iterative image rectification. In *IEEE Con-*

ference on Computer Vision and Pattern Recognition, CVPR, pages 2059–2068, 2019.

[56] Fangneng Zhan, Shijian Lu, and Chuhui Xue. Verisimilar image synthesis for accurate detection and recognition of texts in scenes. In *European Conference on Computer Vision ECCV*, pages 257–273, 2018.

[57] Haichao Zhang and Jianyu Wang. Towards adversarially robust object detection. *CoRR*, abs/1907.10310, 2019.

[58] Yingying Zhu, Minghui Liao, Mingkun Yang, and Wenyu Liu. Cascaded segmentation-detection networks for text-based traffic sign detection. *IEEE Trans. Intelligent Transportation Systems*, 19(1):209–219, 2018.

[59] Yingying Zhu, Cong Yao, and Xiang Bai. Scene text detection and recognition: recent advances and future trends. *Frontiers Comput. Sci.*, 10(1):19–36, 2016.