Perceptual-Sensitive GAN for Generating Adversarial Patches

Aishan Liu,[§] Xianglong Liu,[§*] Jiaxin Fan,[§] Yuqing Ma,[§] Anlan Zhang,[§]
Huiyuan Xie,[†] Dacheng Tao[‡]
[§]State Key Laboratory of Software Development Environment, Beihang University, China
[†]Department of Computer Science and Technology, University of Cambridge, UK
[‡]UBTECH Sydney AI Centre, SIT, FEIT, University of Sydney, Australia
{liuaishan, xlliu, jxfan, mayuqing, zal1506}@buaa.edu.cn, hx255@cam.ac.uk, dacheng.tao@sydney.edu.au

Abstract

Deep neural networks (DNNs) are vulnerable to adversarial examples where inputs with imperceptible perturbations mislead DNNs to incorrect results. Recently, adversarial patch, with noise confined to a small and localized patch, emerged for its easy accessibility in real-world. However, existing attack strategies are still far from generating visually natural patches with strong attacking ability, since they often ignore the perceptual sensitivity of the attacked network to the adversarial patch, including both the correlations with the image context and the visual attention. To address this problem, this paper proposes a perceptual-sensitive generative adversarial network (PS-GAN) that can simultaneously enhance the visual fidelity and the attacking ability for the adversarial patch. To improve the visual fidelity, we treat the patch generation as a patch-to-patch translation via an adversarial process, feeding any types of seed patch and outputting the similar adversarial patch with high perceptual correlation with the attacked image. To further enhance the attacking ability, an attention mechanism coupled with adversarial generation is introduced to predict the critical attacking areas for placing the patches, which can help producing more realistic and aggressive patches. Extensive experiments under semi-whitebox and black-box settings on two large-scale datasets GTSRB and ImageNet demonstrate that the proposed PS-GAN outperforms state-of-the-art adversarial patch attack methods.

Introduction

Recent advances in deep neural networks (DNNs) have enabled researchers to achieve great success in various tasks handling the massive image (Krizhevsky, Sutskever, and Hinton 2012), text (Bahdanau, Cho, and Bengio 2014) and speech (Hinton et al. 2012) data. Despite the successful progress, deep learning models have been proved to be vulnerable and susceptible to adversarial examples (Szegedy et al. 2013; Goodfellow, Shlens, and Szegedy). On one side, adversarial examples pose potential security threats by attacking or misleading the practical deep learning applications like auto driving and face recognition system, which may cause pecuniary loss or people death with severe impairment. On the other side, adversarial examples are also valuable and beneficial to the deep learning models, as they are able to provide insights into their strengths, weaknesses, and blind-spots (Tramèr et al. 2017; Ross and Doshivelez 2018).

Figure 1: Traffic signs with scrawls and patches on them in the real world.

The straightforward solution is to intentionally add small-magnitude perturbations to the input instances like images, generating the maliciously perturbed examples that can fool DNNs to make wrong predictions. In the past years, various typical techniques have been developed to produce adversarial examples along this direction, such as gradient-based algorithms (Goodfellow, Shlens, and Szegedy ; Kurakin, Goodfellow, and Bengio 2016), optimization-based methods (Szegedy et al. 2013; Athalye and Sutskever 2017) and network-based techniques (Xiao et al. 2018; Poursaeed et al. 2018). Network-based techniques have achieved satisfying performance owning to their great power for generating high-quality synthetic data. Among them, the generative adversarial networks (GANs) technique is capable to approximate the true data distribution (Goodfellow et al. 2014; Ma et al. 2018; Song et al. 2018; Pathak et al. 2016), and recently has attracted great attention in producing perceptually realistic adversarial examples with the state-of-of-the-art attacking performance (Xiao et al. 2018).

Besides the well-designed perturbations, the adversarial patch serves as an alternative way to generate adversarial examples, which can be directly localized in the input instance to the deep model (Brown et al. 2017; Karmon, Zoran, and Goldberg 2018). Compared to the traditional perturbation based adversarial examples, the adversarial patch enjoys the advantages of being input-independent and scene-independent, and can be easily placed on any input data with general attack ability. In real world, the patch scenarios often happen where the patches can be invisible or imperceptible to human. For example, the traffic signs with scrawls and

*Corresponding author

patches on them are quite common on the streets of many old cities as shown in Figure 1. The fact makes it much more convenient to be applied to attack a deep learning system like auto driving and face recognition, by generating adversarial patches and simply sticking them on the traffic signs, the face pictures, etc.

To generate adversarial patches, Brown et al. first introduced the concept focusing on the security implications, and attempted to generate universal noise "patches" that can be physically printed and put on any images. Similarly, Karmon, Zoran, and Goldberg created adversarial patches using an optimization-based approach with a modified loss function. But different from the prior research, they concentrated on investigating the blind-spots of state-of-the-art image classifiers, and studying the kinds of noise that can cause misclassification. Evtimov et al. adopted the traditional perturbation techniques to generate the attacking noises, which further can be mixed into the black and white stickers to attack the recognition of the stop sign.

Prior studies in perception and psychophysics indicate that the perceptual sensitivity plays a quite important role in helping accomplish the robust visual recognition (Theeuwes and Chen 2005). Therefore, to complete a high-quality attacking, it is also important to make sure that the generated adversarial patches can beat the perceptual sensitivity of the attacked network. Namely, the adversarial patch should be visually natural with strong perceptual correlations with the image context, and meanwhile spatially localized at the perceptual sensitive positions in the attacked image. Though adversarial patch techniques own the flexibility for attacking and have achieved encouraging performance in the past years, however, most of them usually ignore the perceptual sensitivity, and fail to generate background-harmonious, yet aggressive patches, and thus often resulting in unstable attack effects.

To address the problem, our paper proposes a novel attack framework named perceptual-sensitive GAN (PS-GAN) to generate adversarial patches. Different from existing studies, our PS-GAN exploits the perceptual sensitivity of the attacked network to the adversarial patch, and enhances both the visual fidelity and the attacking ability of the generated adversarial patches. PS-GAN allows adversaries to generate any types of adversarial patches they prefer and specify, and employs a patch-to-patch translation process to pursue the visually natural and context-correlated adversarial patches. Moreover, to further improve the attacking ability, PS-GAN adopts the visual attention to capture the spatially distributed sensitivity and guide the attacking localization of the adversarial patches for the stable attack effects. More importantly, our PS-GAN can instantly generate adversarial patches individually without access to target models anymore at inference time (i.e., semi-whitebox attack (Xiao et al. 2018)). To the best of our knowledge, we are the first to devise an efficient adversarial patch technique that can generate any styles of patch based on the specified seed patch, which enjoys both strong attacking ability and natural appearance in the real world. To evaluate the effectiveness of the proposed method, extensive experiments are conducted on GTSRB (Houben et al. 2008) and ImageNet (Deng et al. 2009) towards different target models, under semi-whitebox and blackbox settings in both digital world and physical world. The experimental results show that our PS-GAN can not only consistently outperforms state-of-the-art adversarial patch attack methods, but also owns strong generalization ability and transferability.

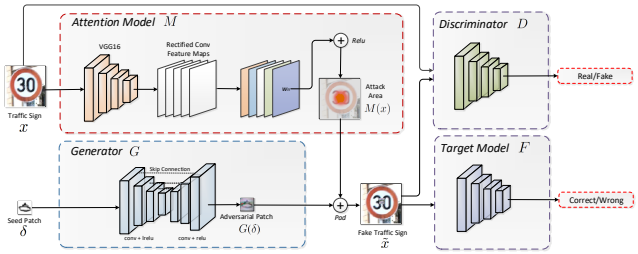


Figure 2: The framework of our PS-GAN consists of a generator G, a discriminator D and an attention model M, attacking a target model F.

Perceptual-Sensitive GAN

In this section, we will first introduce the problem definition, and then elaborate the framework, formulation and corresponding network architecture of our proposed Perceptual-Sensitive GAN (PS-GAN).

Problem Definition

Assuming $\mathcal{X} \subseteq \mathbb{R}^n$ is the feature space with n the number of features. Supposing (x_i, y_i) is the ith instance in the data with feature vector $x_i \in \mathcal{X}$ and $y_i \in \mathcal{Y}$ the corresponding class label. The deep learning model tries to learn a mapping or classification function F: $\mathcal{X} \rightarrow \mathcal{Y}$. Specifically, in this paper we consider the visual recognition problem and an adversarial patch δ is used to mislead the target model F to wrong predictions. Given an original clean image x with its original class label y_{real}, the new image $\tilde{x}$ with attack ability is composed of the original image x, an additive adversarial patch $\delta \in \mathbb{R}^z$ and a location mask $m \in \{0,1\}^n$:

$$\tilde{x} = (1 - m) \odot x + m \odot \delta, \tag{1}$$

where $\odot$ is element-wise multiplication. To simplify, we will use the below equation for the rest of the paper:

$$\tilde{x} = x +_m \delta. \tag{2}$$

The prediction result of $\tilde{x}$ by F is y_{pre}: $y_{pre} = F(\tilde{x})$. The adversarial patch makes the model predict wrong label, namely $y_{pre} \neq y_{real}$.

The Framework

Motivated by the fact that the deep convolutional neural networks (CNNs) usually own strong perceptual-sensitivity to the visual fidelity and spatial localization of the objects in the input images, in this paper we will develop a perceptual-sensitive GAN framework that can generate adversarial patches naturally correlated with the context and visually attentional to the localization of the input images.

In order to improve visual fidelity, a patch-to-patch translation process is introduced to allow adversaries to specify the patch style. In this process, the specified seed patch and attacked image are both taken as input, and the output is an adversarial patch which is visually similar to the input patch and meanwhile harmonious to the attacked images. Specifically, we adopt the adversarial learning process, where a generator G is motivated to create adversarial patches with high perceptual correlation with the image to be attacked. Namely, for the input seed patch δ and the image x to be attacked, it can generate an adversarial patch $G(\delta)$. A discriminator D promises the perceptual similarity. Intuitively, D encourages $\tilde{x}$, the attacked image with the generated adversarial patch, to be harmonious and indistinguishable with the attacked image x, result in high visual fidelity and perceptual correlation.

Besides, to obtain the adversarial attack ability, the target model F is introduced in our framework. Specifically, F behaves as the target model to be attacked, which is responsible for the guidance of adversarial attack ability of the generated patch. The generated patch must be qualified to mislead F.

As to the modelling of the spatial localization sensitivity, in our framework an attention model M is integrated into the end-to-end patch generation. It can capture the attention distribution of the attacked network with respect to the patch localization, and thus help determine the critical areas $M(x)$ to place the patches with strong attacking ability. In the predicted areas, adversarial patch can be generated by G with low distortion rate and high attack success rate.

Our PS-GAN equipped with both the adversarial generation and the attention prediction can generate more realistic and aggressive patches $G(\delta)$, and guide the model to stick the patch to the image x at the position $M(x)$ forming the attacked image $\tilde{x}$:

$$\tilde{x} = x +_{M(x)} G(\delta). \tag{3}$$

Figure 2 illustrates the overall architecture of our PS-GAN.

Formulation

As aforementioned, in this paper we mainly focus on the two key aspects of perceptual-sensitivity including the visual fidelity and the spatial localization. Therefore, in our PS-GAN framework there are two corresponding parts that fully exploits the perceptual-sensitivity: a patch-to-patch adversarial translation that help produce the desired adversarial patch with visual fidelity and perceptual correlation, and a visual attention model that predicts the critical areas to be attacked by the adversarial patch. The two parts are coupled together to guarantee the strong attacking ability.

Visual Fidelity & Perceptual Correlation To improve visual fidelity, the adversarial generation process is developed for its surprising capability of generating realistic images. Specifically, we expect to encourage the model to generate adversarial patches with good visual fidelity. Based on this motivation, the adversarial generation loss can be written as:

$$\begin{aligned} L_{GAN}(G, D) =& \mathbb{E}_x[\log D(\delta, x)] + \\ & \mathbb{E}_{x,z}[\log(1 - D(\delta, x +_{M(x)} G(z, \delta)))], \end{aligned} \tag{4}$$

where x, δ and z are the image to be attacked, the input patch and the noise, respectively. Note that our our PS-GAN differs widely from the conditional GAN (cGAN) (Mirza and Osindero 2014), even we can also treat the input patch as the conditions. In PS-GAN the generator actually can work without z, which could still learn a mapping. In practice, we only provide noise in the form of dropout, applied on some layers of our G. Besides, we simultaneously combine both the discriminator and the target model to guide the pursuit of good generator, which distinguished the original image and the attacked one, rather than the conditioned input in cGAN.

At the same time, a patch loss is further appended to capture and enhance the high perceptual correlation of the generated patch with the context of the input image. Meanwhile, the loss is also responsible for constraining the distortion of the generated patch from the seed patch. Intuitively, we expect the generated patch to share the similar visual perception with the image to be attacked, which means that they have common correlated perceptual meanings, and the generate patch in this case usually should be visually harmonious with the image context in both pixel-wise and perceptual levels. Therefore, we introduce the following loss to guide the learning of the adversarial networks for patch generation:

$$L_{patch}(\delta) = \mathbb{E}_\delta ||G(\delta) - \delta||_2. \tag{5}$$

Attention Sensitivity & Attacking Ability Since our goal is to generate adversarial patches with strong attacking ability, it is required to introduce an adversarial attacking loss. The loss obligates the generator G to produce the patches and the attention model M predicts the localization, which together can mislead the target model further.

Specifically, on one side, we should push the prediction label y_{pre} of the adversarial $\tilde{x}$, a clean input x appended with the adversarial patch $G(\delta)$, away from its original prediction label y_{real}. Therefore, the loss can be defined as follows:

$$L_{adv}(G, F) = \mathbb{E}_{x,\delta}[\log P_F(\tilde{x})]. \tag{6}$$

On the other hand, the attacking performance highly relies on the visual attention sensitivity of the attacked networks, which tries to explain which part of the image contributes more to the model decisions (Zeiler and Fergus 2014; Cao et al. 2015). Therefore, adversarial patches placed in these areas will have more sensitive effect on the model performance. In our PS-GAN framework, we borrow the visual attention technique to predict the critical attacking area. Specifically, to obtain the class-discriminative localization map, Gradient-weighted Class Activation Mapping (Grad-CAM) (Selvaraju et al. 2016) helps compute the gradient of y^c (score for class c) with respect to feature maps A of a convolutional layer, i.e., $\frac{\partial y^c}{\partial A^k_{ij}}$. These gradients flow back to obtain the importance weights α^c_k after being global-average-pooled. This weight α^c_k captures the "importance" of feature map k for a target class c, forming an attention mask for placing the adversarial patch.

Finally, the adversarial attacking loss combined with the above visual fidelity and perceptual correlation losses leads to our final patch generation formulation:

$$\min_G \max_D \quad L_{GAN} + \lambda L_{patch} + \gamma L_{adv}, \tag{7}$$

where $\lambda > 0$ and $\gamma > 0$ balance the contribution of each part. With the guidance of the attention model, optimizing the above problem drives our PS-GAN model to find the near-optimal generator, which can produce adversarial patches with strong attacking ability and perceptual sensitivity of the attacked networks.

Network Architecture

Next, we briefly introduce the network architectures in our PS-GAN framework.

Generator Following the generator and discriminator architectures adopted in (Isola et al. 2017; Johnson, Alahi, and Li 2016) for image translation and perceptual information capturing, we employ the U-Net architecture in our generator, which allows low-level information to shortcut across the network, leading to better results. Let `C`k denote a Convolution-LayerNorm-LeakyReLU layer with k filters. All convolutions are (4×4) spatial filters applied with stride 2. The encoder-decoder architecture consists of:

Encoder: `C16-C32-C64-C128`

Decoder: `C64-C32-C16-C3`

Discriminator Our discriminator architecture is:

`C64-C128-C256-C512`

The last layer of discriminator network is fed into a linear layer to generate a 1-dimensional output, followed by a Sigmoid function.

Target Model Our target model F could be any given deep networks with the last two layers accessible (e.g., Softmax layer and the layer before it). These two layers are used as a part of L_{adv}. To perform adversarial attack, the loss L_{adv} encourages $\tilde{x}$ to be misclassified by F.

Attention Model As our basic visual attention method, Grad-CAM can achieve surprising performance. We use a pre-trained VGG16 on ImageNet dataset to obtain the attention region of images. We firstly compute the gradient of the last fully connected layer with respect to the output feature maps of the fourth convolutional layer $conv4$. Specifically, to obtain the class-discriminative localization map, gradient of y^c (score for class c) with respect to feature maps $A^k \in \mathbb{R}^{u \times v}$ of $conv4$ is calculated. Then these gradients are used to compute the weight of each feature map. They flow back to obtain the importance weights α_k^c after being global-average-pooled:

$$\alpha_k^c = \frac{1}{u \times v} \sum_{i=1}^{u} \sum_{j=1}^{v} \frac{\partial y^c}{\partial A_{ij}^k}. \tag{8}$$

After that, the attention map of input image is calculated and acquired by these feature maps. Specifically, this weight α_k^c represents a partial linearization of the deep network downstream from A, and captures the "importance" of feature map k for a target class c. Our Attention map is a weighted combination of feature maps, but followed by a ReLU:

$$L_{Grad-CAM}^c = ReLU\left(\sum_k \alpha_k^c A^k\right). \tag{9}$$

Algorithm 1 Perceptual-Sensitive Generative Adversarial Network (PS-GAN).

Input: training image set $X_{image} = \{x_i | i = 1, ..., n\}$, and training patch set $\delta_{patch} = \{\delta_i | i = 1, ..., n\}$
Output: non-linear parameters set $\mathbf{W}_D$ and $\mathbf{W}_G$.
for the number of training epochs **do**
  for k steps **do**
    sample minibatch of m images $\psi_x = \{x_1, ..., x_m\}$;
    sample minibatch of m patches $\psi_\delta = \{\delta_1, ..., \delta_m\}$;
    generate minibatch of m adversarial patches $\psi_\delta^G = \{G(\delta_1), ..., G(\delta_m)\}$;
    obtain attention map $M(\psi_x)$ by *Grad-CAM*;
    construct minibatch of m^2 adversarial images $\psi_{\tilde{x}} = \{x_i +_{M(x_i)} \delta_j | i, j = 1, ..., m\}$;
    optimize $\mathbf{W}_D$ to $\max_D L_{GAN}$ with G fixed.
  end for
  sample minibatch of m images $\psi_x = \{x_1, ..., x_m\}$.
  sample minibatch of m patches $\psi_\delta = \{\delta_1, ..., \delta_m\}$.
  obtain attention map $M(\psi_x)$ by *Grad-CAM*.
  optimize $\mathbf{W}_G$ to $\min_G L_{GAN} + \lambda L_{patch} + \gamma L_{adv}$ with D fixed.
end for

As a result, the output attention map can highlight important regions of the image which correspond to any decision of interest. Thus, critical areas are provided where modifications in these areas are more sensitive to final predictions achieving strong attacking ability.

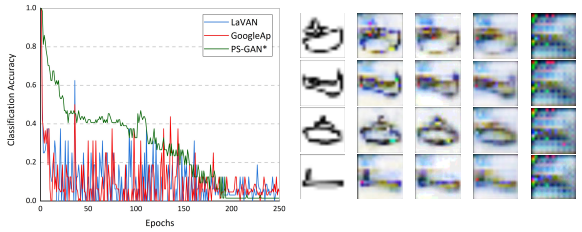
Training Process

The entire training process is detailed in Algorithm 1. It is mainly a recurrent and iterative training process of G and D. In each iteration, we train D for k times while once for G. Attention map for each image in each minibatch is acquired by Grad-CAM to guide areas to place patch. Standard gradient-based optimization methods can be used to learn $\mathbf{W}_D$ and $\mathbf{W}_G$. We use Adam and SGD for G and D in our experiments, respectively.

Experiments

In this section, we will evaluate our proposed algorithm PS-GAN in the classification attacking task. Firstly, we compare our method with the state-of-the-art adversarial patch methods: GoogleAp (Brown et al. 2017) and LaVAN (Karmon, Zoran, and Goldberg 2018) on GTSRB (Houben et al. 2008) and ImageNet (Deng et al. 2009) from three aspects: attacking success rate, visual fidelity and time consumption. Secondly, we investigate the performance of our PS-GAN under semi-whitebox and blackbox settings on GTSRB, which will demonstrate the excellent transferability and generalization ability of PS-GAN. At last, the attacking experiment in the physical world will be conducted to prove the practicability of PS-GAN in the real world.

Datasets and Models German Traffic Sign Recognition Benchmark (GTSRB) is a large multi-category classification benchmark for traffic sign classification. There are more than 40 classes and 50,000 images in the dataset. In order



(a) (b) (c) (d) (e) (f)

Figure 3: The attacking performance when using different training epochs. (a) The classification accuracy with respect to the training epochs; (c)-(f) present the generated adversarial patches with increasing patch distortion, where (b) corresponds to the original input patch without any distortion, and the classification accuracy decreases, i.e., 85.6%, 12.5%, 10.9%, 7.8% and 1.5%, respectively.

to keep high perceptual correlation between the patches and the attacked images, we choose QuickDraw (J. Jongejan and Fox-Gieg. 2016) as the corresponding patch dataset. QuickDraw is a collection of 50 million drawings and scrawls across 345 categories. As people like to draw scrawls on public facilities in the real world, it will be natural for human being if we generate scrawl-like adversarial patches and stick them on the traffic signs. Here, we use 20 classes (SpeedLimit, NoPass, Stop, etc.) in GTSRB and 20 classes (Aircraft Carrier, Backpack, Basket, etc.) in QuickDraw. We also test our PS-GAN on the natural images and choose the samples from ImageNet dataset. Specifically, we choose "monkey", "dining-table", "dog" and "cat" as the attacked image classes, and the corresponding patch classes are "apple", "orange" and "baseball", for their high perceptual correlation. Each image and patch is normalized to $[-1, 1]$ and scaled to $128 \times 128 \times 3$ and $16 \times 16 \times 3$, respectively. As we can see, the size of patch only accounts for 1.5% of the size of image.

In our experiments, we will attack the following models: VGG16, ResNet-34, VY (Yadav 2016) and some variants of them.

Implementation Details In our experiments, we use Tensorflow and Keras for the implementation and test them on a NVIDIA Tesla K80 GPU cluster. We train PS-GAN for 250 epochs with a batch size of 64, with the learning rate of 0.0002, decreased by 10% every 900 steps. As for the hyper-parameters in loss function, we set λ range from 0.002 to 0.005 and γ to 1.0 and δ to 0.0001, respectively. For the attention model, we retrained two pre-trained VGG16 models on GTSRB and ImageNet respectively.

Comparative Experiments

Firstly, we compare the attacking performance of our method with GoogleAp and LaVAN on GTSRB and ImageNet.

Attacking Ability Figure 3(a) shows the attacking ability of different methods when trained with different epochs. We can see that all the methods converge at about 200 epochs.

Table 1: Classification accuracy of examples with adversarial patches attacking the target model VY. The lower accuracy indicates better attacking performance.

Dataset	PS-GAN	PS-GAN*	GoogleAp	LaVAN
GTSRB	12.5%	**1.5%**	4.7%	3.1%
ImageNet	25.0%	**4.7%**	6.3%	**4.7%**

Subsequently, we report the classification accuracy of all methods with 200 epochs training.

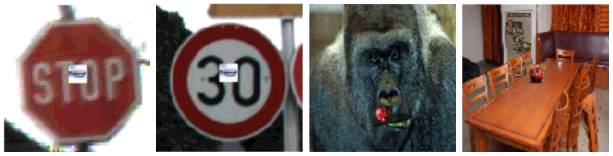
Table 1 shows the classification accuracy of examples with adversarial patches generated by each method. Since GoogleAp and LaVAN do not control the noise distortion of the patch, to make fair comparison we derive two different versions of our PS-GAN, namely PS-GAN is our original model with strong constraint of patch distortion and PS-GAN* is the one with weak constraint as GoogleAp and LaVAN. From the table we can see that both of the two PS-GAN versions can generate adversarial patches with satisfying attacking success rate. Note that the weak distortion constraint makes PS-GAN*, GoogleAp and LaVAN perform well, and our PS-GAN* get the best performance. However, these methods ignore the visual fidelity and perceptual correlations with the attacked images, and thus produce unnatural patches that are offensively conspicuous to human being. Instead, our PS-GAN can alleviate this problem by limiting the patch distortion. Figure 3(b)-(f) show the variation when using different levels of distortion, where we can also conclude that the large distortion helps obtain strong attacking ability but at the cost of obvious visual fidelity loss.

Besides the classification accuracy, Figure 3(a) also depicts the attacking stability of the generated adversarial patches. With a number of training epochs, the attacking performance of our PS-GAN becomes stable and keeps the best among all methods. However, even with the same number of training epochs, the performance of both GoogleAp and LaVAN still vibrates sharply. The main reason might be that the adversarial patches produced by GoogleAp and LaVAN owns good attacking ability for train set images stemming from the sufficiently large perturbations, which generalize poorly for the unseen images in the testing set. Contrarily, our PS-GAN is able to learn the distribution of adversarial patches for both training and testing data. Besides, PS-GAN equipped with the attention prediction can gradually localize the most perceptually sensitive area to attack. This can be easily observed from the sharp and consistent accuracy decrease as we run more epochs at the training stage.

Visual Fidelity & Perceptual Correlation It is important to make the adversarial patches keep the natural and friendly appearance when performing the attack in the real world. Figure 4 shows the adversarial patches generated by different methods, and we can see that GoogleAp and LaVAN output very unnatural and inharmonious patches with the attacked images. Besides, some patches are even generated outsides the reasonable areas, e.g., the traffic sign. It means that the patches generated by the existing state-of-the-art solutions like GoogleAp and LaVAN can be easily noticed in practice, which subsequently limits their attack-

(a) GoogleAp

(b) LaVAN



(c) PS-GAN

Figure 4: Adversarial patches generated by GoogleAp, LaVAN and PS-GAN on GTSRB and ImageNet in the semi-whitebox setting. All the attacked images are misclassified.

Training Set	
Traffic Sign	Speed Limit 30
Patch	Aircraft Carrier
Accuracy with SP	85.6%
Accuracy with AP	1.5%
Testing Set	
Traffic Sign	Warning Sign
Patch	Backpack, Basket
Accuracy with SP	81.3%
Accuracy with AP	15.6%

Figure 5: Adversarial patches generated by PS-GAN with input patch class it has never seen at train time. The left table is experiment configurations, while the figures on the right are two test results.

ing ability. On the contrast, the generated patches by PS-GAN look more like the commonly appeared scrawls well placed on the traffic signs. This is because our model tries to modify the patches using the confined and high perceptually correlated noise, so that the perturbations added to the images are inapparent to human beings but deadly to deep learning models leading to misclassification.

Time Consumption We also investigate the time consumption for generating adversarial patches by each method. GoogleAp and LaVAN respectively spend 61.2s and 65.4s on producing one patch on GTSRB datasets, and similarly 72.3s and 81.5s on ImageNet. PS-GAN only takes 0.106s and 0.111s per patch for GTSRB and ImageNet, which means that PS-GAN enjoys both the fast computation and the ease for use in practice.

Table 2: Semi-whitebox attacking performance.

	GTSRB	ImageNet
Accuracy without patches	89.5%	87.6%
Accuracy with seed patches	85.6%	67.6%
Accuracy with adversarial patches	12.5%	25.0%

Semi-whitebox and Blackbox Attack

We apply different structures for the target model F in this experiment. The target models include many different deep models with different activation functions and training data.

Semi-whitebox Setting Firstly, we generate adversarial patches to perform semi-whitebox attack against VY and $VGG16$ on GTSRB and ImageNet respectively. As observed in Table 2, PS-GAN has the ability to generate adversarial patches to attack all the listed target models with high attack success rate. Compared with images placed with seed patches (SP), those with the generated adversarial patches (AP) largely decline the classification performance of the target networks.

Generalization Ability We also test the generalization ability of our proposed model on GTSRB. After we have the well trained model, at the inference time we feed the model with the images and patches of unseen classes in the training set, and Figure 5 illustrates some representative results. As we can see, even though the traffic sign and the patches in the training and testing sets are from different classes, our PS-GAN is still be able to preserve the strong attacking ability, which means our model also enjoys the good generalization ability in practice.

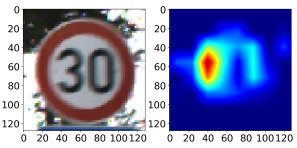
Transferability & Blackbox Attack Now we evaluate the transferability in the blackbox attacking settings. Table 3 shows the classification accuracy of PS-GAN, when transferring attacks between different classification models. In this case, we use 640 random testing inputs. We first generate adversarial patches for a source attacked model, and then apply the patches to attack all other target models. The target models include a number of deep models with different activation functions like $lrelu$ and $tanh$: VY, $VGG16$, VY_{lrelu}, $VGG16_{tanh}$, $\overline{VY}$, $\overline{VGG16}$ and $ResNet$. As for $\overline{VY}$ and $\overline{VGG16}$, we train them using the training data disjoint with those of VY and $VGG16$. Note that the diagonal results correspond to the white-box or semi-whitebox attacking settings, and the others are results of the blackbox attack.

The transferability performance is listed in Table 3, from which we can get the following conclusions:

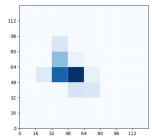
- Adversarial patches generated by PS-GAN have very encouraging transferability among different target models, which means our PS-GAN can perform quite well in black-box setting.
- Attacking ability is highly correlated with the capacity of the learning model generating the adversarial patches. For example, adversarial patches generated by $VGG16$ show good attacking performance on VY, because $VGG16$ usually owns much more complicated network structure and thus better capability than VY in practice.

Table 3: Classification accuracy of adversarial examples transferred between different models on GTSRB.

Source Models \ Target Models	VY	$VGG16$	VY_{lrelu}	$VGG16_{tanh}$	$\overline{VY}$	$\overline{VGG16}$	$ResNet$
VY	**12.5**%	25.0%	37.5%	12.5%	15.6%	31.3%	37.5%
$VGG16$	1.6%	**31.3**%	15.6%	37.5%	1.6%	31.3%	34.4%
VY_{lrelu}	4.7%	25.0%	**7.8**%	23.4%	12.5%	29.7%	26.6%
$VGG16_{tanh}$	3.1%	25.0%	32.8%	**34.4**%	7.8%	25.0%	25.0%
$\overline{VY}$	9.4%	25.7%	14.1%	25.0%	**14.1**%	28.1%	37.5%
$\overline{VGG16}$	3.1%	37.5%	9.4%	34.4%	7.8%	**31.4**%	21.9%
$ResNet$	3.1%	15.6%	4.7%	21.9%	9.4%	26.6%	**34.4**%



(a) Attention map



(b) Distortion heat map

Figure 6: The consistence between the visual attention map and distortion heat map.

(a) $-15°$, 1m (b) $0°$, 1m (c) $15°$, 3m (d) $0°$, 3m

Figure 7: Traffic signs with adversarial patches on them. The photos are taken in the campus of Beihang University with different distances and angles using camera. Traffic signs are respectively classified as "No Entry", "Slippery Road", "Speed Limit 60" and "Over-weighted Vehicles Prohibited".

Visual Attention

In this section, we analyze the attention prediction results by conducting a distortion heatmap experiment. First we can adopt the relatively complicated deep model $VGG16$, and get the attention map based on the layer $conv4$ from a pre-trained $VGG16$, which is shown in Figure 6(a). To check whether it is the most sensitive area for placing adversarial patches, we try to experimentally find the area by repeatedly training the attacking model with a number (i.e., 64) of fixed patch localization positions. For each training, we change the constraint of the distortion and force the attack success rate to reach a threshold value of 50%. Based on the batch of experiments, finally we can get a distortion heat map indicating the probability of the successful attack in Figure 6(b), where the dark zones indicate less distortion and thus the sensitive places for classification. From the figure, it is easy to observe that the attention map and the most sensitive attacking areas are perfectly matched.

Attention Area v.s. Attack Area According to the attention map experiment, we can conclude that the more attentive area we attack the less distortion is needed and the better effect is. If we attack the area of image that the model really cares for classification, we only need a very small distortion. Different from existing studies, we appreciate the significance of attack areas and result in stable attack effects. This conclusion is very valuable and has some relations with the gradient-based method (Goodfellow, Shlens, and Szegedy). In that type of algorithms, adversarial examples are generated in a gradient-guided way. The noises are added intentionally at pixels where the gradient is more critical to change the final prediction label. Both of gradient-based and our method pay attention to the critical place of images to attack in order to get good attack success rate and low perturbations.

Physical World Attack

In this section, a physical world attack experiment is conducted to validate the practical effectiveness. We first take 64 pictures of a real-world traffic sign "Speed Limit 20" in the street, with varying angles $\{0°, 15°, 30°, -15°, -30°\}$ and distances $\{1m, 3m, 5m\}$. The accuracy of the target classification model on these images is 86.7%. Then, the PS-GAN model is further trained to generate four different patches based on input patches from "Aircraft Carrier". After printing these patches by a Fuji Xerox DocuPrint CM318z, we place them on the real-world traffic sign "Speed Limit 20" and take pictures with the combination of different distances and angles as before using a Huawei P20 camera. Figure 7 shows the different examples and the classification results, where the adversarial patches generated by PS-GAN possess strong attacking ability, decreasing the classification accuracy from 86.7% to 17.2% on average.

Conclusion

In this paper, we proposed a perceptual sensitive GAN (PS-GAN) for generating adversarial patches. By exploiting the perceptual sensitivity of the attacked network, PS-GAN can guarantee that the generated adversarial patch enjoys a natural appearance, i.e., the high visual fidelity and perceptual correlation with the context of image to be attacked. Besides, it couples the attention mechanism in the adversarial generation process, and promises the strong attacking ability for the generated adversarial patches. The extensive experimental results, under semi-whitebox and blackbox settings in both digital and physical world, demonstrate that PS-GAN owns strong generalization ability and transferability, and achieves state-of-the-art performance.

Acknowledgements

This work was supported by National Natural Science Foundation of China 61690202, 61872021, MSRA Collaborative Research Grant, FL-170100117, DP-180103424 and IH-180100002.

References

Athalye, A., and Sutskever, I. 2017. Synthesizing robust adversarial examples. *arXiv preprint arXiv:1707.07397*.

Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Brown, T. B.; Mané, D.; Roy, A.; Abadi, M.; and Gilmer, J. 2017. Adversarial patch. *arXiv preprint arXiv:1712.09665*.

Cao, C.; Liu, X.; Yang, Y.; Yu, Y.; Wang, J.; Wang, Z.; Huang, Y.; Wang, L.; Huang, C.; Xu, W.; et al. 2015. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *IEEE International Conference on Computer Vision*, 2956–2964.

Deng, J.; Dong, W.; Socher, R.; Li, L. J.; Li, K.; and Li, F. F. 2009. Imagenet: A large-scale hierarchical image database. In *The IEEE Conference on Computer Vision and Pattern Recognition, 2009*, 248–255.

Evtimov, I.; Eykholt, K.; Fernandes, E.; Kohno, T.; Li, B.; Prakash, A.; Rahmati, A.; and Song, D. 2017. Robust physical-world attacks on deep learning models. *arXiv preprint arXiv:1707.08945* 1.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *Advances in Neural Information Processing Systems,* 2672–2680.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. Explaining and harnessing adversarial examples (2014). *arXiv preprint arXiv:1412.6572*.

Hinton, G.; Deng, L.; Yu, D.; Dahl, G. E.; Mohamed, A.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; and Sainath, T. N. 2012. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine* 29(6):82–97.

Houben, S.; Stallkamp, J.; Salmen, J.; Schlipsing, M.; and Igel, C. 2008. Detection of traffic signs in real-world images: The german traffic sign detection benchmark. In *International Joint Conference on Neural Networks*, 1–8.

Isola, P.; Zhu, J. Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 5967–5976.

J. Jongejan, H. Rowley, T. K. J. K., and Fox-Gieg., N. 2016. The quick, draw! - a.i. experiment. https://github.com/googlecreativelab/quickdraw-dataset.

Johnson, J.; Alahi, A.; and Li, F. F. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European Conference on Computer Vision*, 694–711.

Karmon, D.; Zoran, D.; and Goldberg, Y. 2018. Lavan: Localized and visible adversarial noise. *arXiv preprint arXiv:1801.02608*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *International Conference on Neural Information Processing Systems*, 1097–1105.

Kurakin, A.; Goodfellow, I.; and Bengio, S. 2016. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*.

Ma, Y.; He, Y.; Ding, F.; Hu, S.; Li, J.; and Liu, X. 2018. Progressive generative hashing for image retrieval. In *27th International Joint Conference on Artificial Intelligence*, 871–877.

Mirza, M., and Osindero, S. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

Pathak, D.; Krahenbuhl, P.; Donahue, J.; Darrell, T.; and Efros, A. A. 2016. Context encoders: Feature learning by inpainting. In *The IEEE Conference on Computer Vision and Pattern Recognition*, 2536–2544.

Poursaeed, O.; Katsman, I.; Gao, B.; and Belongie, S. 2018. Generative adversarial perturbations. In *The IEEE Conference on Computer Vision and Pattern Recognition*.

Ross, A. S., and Doshivelez, F. 2018. Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients. *32nd AAAI Conference on Artificial Intelligence*.

Selvaraju, R. R.; Das, A.; Vedantam, R.; Cogswell, M.; Parikh, D.; and Batra, D. 2016. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR, abs/1610.02391,* 7.

Song, J.; Zhang, J.; Gao, L.; Liu, X.; and Shen, H. T. 2018. Dual conditional gans for face aging and rejuvenation. In *27th International Joint Conference on Artificial Intelligence*, 899–905.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Theeuwes, J., and Chen, C. Y. D. 2005. Attentional capture and inhibition (of return): The effect on perceptual sensitivity. *Perception & Psychophysics* 67(8):1305–1312.

Tramèr, F.; Kurakin, A.; Papernot, N.; Goodfellow, I.; Boneh, D.; and McDaniel, P. 2017. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204*.

Xiao, C.; Li, B.; Zhu, J.-Y.; He, W.; Liu, M.; and Song, D. 2018. Generating adversarial examples with adversarial networks. *arXiv preprint arXiv:1801.02610*.

Yadav, V. 2016. p2-trafficsigns. https://github.com/vxy10/p2-TrafficSigns.

Zeiler, M. D., and Fergus, R. 2014. Visualizing and understanding convolutional networks. In *European conference on computer vision*, 818–833. Springer.