Learning Universal Adversarial Perturbations from Local and Global Perspectives

Anonymous CVPR submission

Paper ID 11541

Abstract

Deep neural networks (DNN) have been proven to be vulnerable to adversarial attacks. The early attacks mostly involved image-specific approaches that generated specific adversarial perturbations for each individual image. More studies have further demonstrated that neural networks can also be fooled by image-agnostic perturbations, called "universal adversarial perturbation". In this paper, we consider the success rate of adversarial attacks and the quasi-imperceptibility of perturbations and introduce two novel, simple but efficient universal adversarial perturbation generation methods. We first concentrate mainly on the attack success rate of universal adversarial samples and develop an optimization-based generation method (SUAN) to achieve visible local adversarial noises, while taking into account the pixel intensity and the amount of perturbation. The method achieves excellent fooling rates in both targeted and non-targeted attacks, almost all above 95%, and performs well in cross-data and cross-model settings. Then, considering the imperceptibility of the perturbation, we propose an optimization algorithm combined with deep steganography (UAP-DS) on this basis to map the local adversarial noises to the global perturbation. This method achieves equivalent or even surpasses advanced methods in non-targeted attacks and has good transferability. Finally, we verify the attack effect of the universal adversarial perturbations generated by both methods, discuss their limitations and make a trade-off between the two in specific application scenarios.

1. Introduction

Deep neural networks(DNNs) have been revealed to be vulnerable [32] to adversarial examples, i.e., they can easily be fooled by quasi-imperceptible perturbations for humans, causing them to output incorrect predictions. This property has raised security concerns about the applicability of deep networks in security-critical domains such as computer vision [4], autonomous driving [11], speech recognition [5], etc. Subsequently, several other studies [15, 18, 19, 21] have also investigated this interesting property. Most adversarial perturbations [8, 10, 15, 19, 22, 29, 32] are image-dependent, which means that the applied perturbations vary with the input images. Moosavi-Dezfooli et al. [18] indicated that a universal image-agnostic adversarial perturbation may exist; this type of perturbation is a fixed pattern that causes a large proportion of natural images to be misclassified with high probability. The universal adversarial perturbations are more harmful to DNNs compared to the universal image-dependent adversarial examples. On the one hand, no information of the model is required when using the universal adversarial examples against the target model in the test phase. On the other hand, the type of perturbation greatly lowers the threshold of attack, making it easy to abuse. Currently, researches [6, 9, 13, 20, 21, 23–25, 27, 36] on universal adversarial perturbations are emerge in an endless stream.

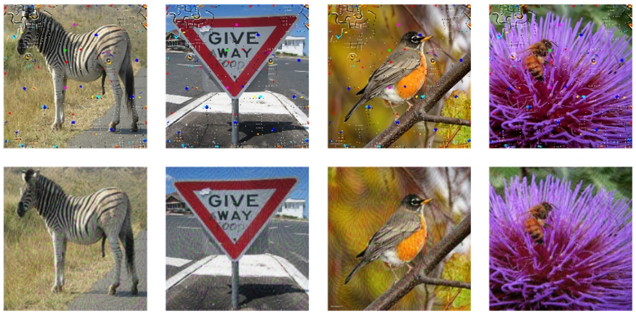

Figure 1. Universal adversarial examples generated by SUAN (the first row) and UAP-DS (the second row).

In this paper, we propose two universal adversarial samples generation methods with considering the attack success rate and the quasi-imperceptibility, generating locally visible adversarial noises and global perturbations to realize effective attack respectively. On the one hand, the superior attack success rate is the main purpose of constructing adversarial samples, and the focus is on achieving a powerful attack. On the other hand, small perturbations are the basic premise of adversarial samples, which show obvious differences in deep neural networks but are quasi-imperceptible for humans. It is a major feature of adversarial samples, which exposes the dead area of machine learning or deep learning algorithms that indicates that the deep neural network has hidden features, which are related to the data distribution in an inconspicuous way. Both of our proposed two universal adversarial perturbations have achieved good attack performance. Figure 1 shows some universal adversarial examples generated by the two methods. In addition, we analyzed and discussed their respective advantages and limitations, and achieved a trade-off between the two. Our main contributions are summarized as follows:

- We introduce a simple optimization-based strategy, called SUAN, to generate visible universal adversarial noises to realize high attack success rates. We first start with the targeted attacks, develop an optimization method to find the common characteristics of the target category to fool target models, and then further extend it to non-targeted attacks. The results show that the method achieves excellent performance with success rates exceeding 95% for almost all attacks against target models, outperforming the state-of-art method, and performs well in cross-data and cross-model settings.
- Based on SUAN, taking into account quasi-imperceptibility for humans, we present another generation method with combining deep image steganography, called UAP-DS. By mapping the adversarial noise distributed in the local area of the image into a global inconspicuous perturbation, the attack is realized. Although sacrificing a little attack success rate, our method still shows comparable performance to current advanced methods, and even surpasses them.
- Finally, we analyze the two methods, compare their different effects in the attack, discuss their respective advantages and limitations, and make a choice between the two based on specific attack and application scenarios.

2. Related Work

Non-targeted universal attack. Moosavi-Dezfooli et al. [18] were the first to report the existence of universal adversarial perturbations (UAP). They applied an iterative method, DeepFool [19], for each training example to get perturbation and update the non-targeted universal perturbation until the accuracy of the target model on the training set is lower than a certain threshold, which was time-consuming. Besides, this method has a strict requirement regarding the original training samples of the target model and their amounts. Konda et al. [25] used a generative model to establish an adversarial perturbation distribution that ensures the generated perturbations are different from each other. They also introduced Fast Feature Fool [21], an image-agnostic perturbation generation method that did not require training data that maximizes the product of the average activations of multiple layers of the network to generate adversarial perturbations. An extended version [20] improves performance by using l_2 norm of activations instead of the mean activations. However, the lack of training data leads to poor performance compared with the UAP approach in [18]. Konda et al. [24] further proposed a two-stage process to generate the adversarial perturbation that first simulates the real data samples with class impression and then uses generative models to obtain perturbations by using randomly sampled vectors in latent space, which performs better than UAP. Shafahi et al. [27] proposed an improved optimization algorithm based on UAP that maximized the cross-entropy loss through parameter limitations using stochastic gradient methods. This method reduced the time consumption under the premise of achieving the same effect as UAP.

The targeted universal attack. Karmon D et al. [13] proposed LaVAN, which implements a targeted universal adversarial attack by adding a visible local patch to the natural image. This method achieved high success rates. The local noise occupies only 2% of the area of the original image, and it is translation invariant. Nevertheless, due to the locality and the marginality of the position of the noise, this obviously visible perturbation is easily detected by anomalies and masked or partially removed [17, 33]. Omid et al. [23] presented a unifying framework called GAP to create both non-targeted and targeted approaches in universal and dependent situations. They parameterized an end-to-end trained model to seek a mapping from a random pattern to a universal perturbation. One significant contribution of GAP is that their method was the first to consider targeted universal attacks. Zhang et al. [35] proposed a new method MIIP for generating targeted universal adversarial perturbations using random source images. It completes targeted universal attacks without using original training data, and achieves performance equivalent to the state-of-the-art baseline using the original training data set. Targeted universal adversarial attacks are more challenging than non-targeted attacks because finding a single pattern that can mislead the target model to output a specific target label is a more restrictive problem. The results of these targeted attacks still had room for improvement.

Universal adversarial perturbation and steganography. Din et al. [9] proposed a universal adversarial perturbation generation method based on image steganography. The proposed perturbations are computed in a transform domain and the secret image is embedded in any target image using wavelet decomposition to fool deep neural networks with high probability. However, the attack effect of the method has a great difference on different secret images. Anshumaan et al. [6] pointed that the frequency spectrum has played a significant role in learning unique and discriminating features for object recognition. On this basis, they use "WaveTransform" to separate the low-frequency and high-frequency subbands of natural images, and optimize and generate adversarial noise on the wavelet band. Regrettably is the adversarial noises are image-dependent. Zhang et al. [36] explained the success of universal adversarial perturbation and deep image steganography tasks from Fourier perspective, indicating that frequency is a key factor affecting classify task performance and attributed its success to DNN's high sensitivity to high-frequency content, which is also mentioned in [36]. Based on this, they proposed two new variants of universal perturbations with deep steganography, USAP, and HP-UAP, which realize both attack and hiding.

Our work proposes two optimization-based universal adversarial sample generation algorithms, which have nothing to do with the specific manifestation of interference. One is local adversarial noises and the other is global adversarial perturbations. Our main goal is to improve the attack performance and to guarantee the invisibility of the perturbations as much as possible, with the expectation that the different adversarial perturbations can be maximally useful.

3. SUAN: Simple Universal Adversarial Noises Generation Method

In this section, starting with targeted attacks, we discussed the generation process of universal adversarial noises, established the connection between target labels and noises, and developed a simple and efficient generation algorithm. Subsequently, it was extended to non-targeted attacks.

3.1. Generation Method

For the targeted attacks, a universal adversarial noise is designed to move each original sample from its original category across the decision boundary to the target category. In other words, the universal adversarial noise can be regarded as a "shortcut" for feature transformation between different categories in the multi-dimensional space. Intuitively, the cost of generating noise aimed at a certain category for different samples is different, and it is not easy to search for the minimum noise for each sample in the pixel space and superimpose it to achieve generality. A simpler and more effective way is to find the common characteristics of each specific category related to the model, that is, to build the relationship between the noise and the target label. On this basis, we developed a simple generation method to find universal adversarial noises. The optimization process mainly follows the work of neural cleanse [34], which is designed to reverse engineer the backdoor trigger of a backdoor neural network.

$$x_{adv} = m \odot p + (1 - m) \odot x \tag{1}$$

We use a generic form to describe adversarial examples generation of SUAN, shown in Eq.(1), where x_{adv} refers to synthetic adversarial example, x denotes the raw image and belongs to the training set X. The mask, m, is a single-channel image that has the same width and height as the original image, and the value of each element in m ranges from [0,1], meaning the rewrite proportion of the adversarial noise at each corresponding position. Analogously, $1 - m$ represents the proportion of the original image. The pattern, p, is an RGB image with the same size as x. In extreme cases, when the mask value is 0, the corresponding value is the original pixel intensity. Similarly, when the mask value is 1, the pixel intensity of the pattern p completely overwrites the original image pixel. In short, we generate adversarial samples by overlapping the original image with a masked adversarial pattern.

For the targeted attack, the noises need to fool the neural network to output a specific category. The prediction should be close to the original target category and the mask should be as small as possible. The optimization function of the targeted attack is shown in Eq.(2), where y_{target} is a one-hot vector of the target category, $f(\cdot)$ denotes the target neural network's prediction under our attack, $CE(\cdot)$ describes the cross-entropy loss. $|m|$ means the L_1 norm of the mask. λ is a hyperparameter that can be adjusted dynamically to balance the first and the second parts of the objective function, where the former guarantees a high attack success rate and the latter determines the pixel intensity and amount of adversarial noises as small as possible.

$$\min_{m,p} \{CE(y_{target}, f(x_{adv})) + \lambda \cdot |m|\} \tag{2}$$

Compared with targeted attacks, non-targeted attacks are easier to perform. We don't need to establish the relationship between the noise and a certain category, but only need to find the minimum amount of noises that destroys the image features. Therefore, for non-targeted attacks, we only need the neural network to output an incorrect answer. In this case, the goal is to maximize the distance between the predicted probability and the true label while constraining the size of the noise. The objective function is shown in Eq.(3), where y_{true} means the one-hot vector of the correct category.

$$\min_{m,p} \{-CE(y_{true}, f(x_{adv})) + \lambda \cdot |m|\} \tag{3}$$

We used the Adam [14] optimizer to solve this multi-objective optimization task. The pattern p and mask m were initialized randomly. The optimization process terminated when the average attack success rates over the expected threshold ε and the L_1 norm of mask no longer decrease. To reduce the time consumption, the early stopping scheme was used. When the average attack success rate of the noise in an iteration for n consecutive batch data reaches the threshold ε, and the hyperparameter λ reaches a balance between the cross-entropy loss and the norm loss of m, the optimization stops, and the optimal m is together with the optimal p form the final adversarial noise. The generation algorithm of SUAN is shown in Algorithm 1.

Algorithm 1 SUAN: simple universal adversarial noises generation.

```
Require: image $x \in X$, $y_{target}$, $y_{true}$, threshold $\varepsilon$
  Random initialize mask $m$ and pattern $p$; $|m_{best}|$ ←INF
  for $Epoch$=1 to $N$ do
    for minibatch $B \subset$ X do
      Compute loss function as Eq.(2) or Eq.(3).
      Update $m$, $p$ through Adam Optimizer.
      $x_{adv}$=$m \odot p + (1-m) \odot x$
    end for
    if average_attack_acc≥ $\varepsilon$ and $|m| < |m_{best}|$ then
      $m_{best}$=$m$,   $p_{best}$=$p$
    end if
    Ajust $\lambda$ and use the early stop scheme to break.
  end for
  return $m_{best}$, $p_{best}$
```

3.2. Experiment Results

In this section, we conducted targeted and untargeted attacks to verify the effectiveness of the proposed method. For targeted attacks, we discussed that the generated noises have obvious categorical characteristics, and at the same time proved that the method has good transferability across datasets and is data-independent. For untargeted attacks, we evaluated the high fooling rates aimed at target model and verified the transferability of the generated universal adversarial noises on different models.

3.2.1 Targeted Attacks

We performed our attack on the standard Inception-v3 [31] network for ImageNet [26] classification task. For the 1,000 ImageNet classes, we selected 10 images for each class as a training set X and used all 50,000 validation images as testing samples. To construct the targeted attack, we chose 100 random classes as the target categories and attacked them sequentially. Figure 2 shows some examples of the generated universal adversarial noises. Based on the noise images, we found that the generated noises are mostly concentrated primarily in a certain area of the image, while others are distributed over the image in blocks. It is worth mentioning that our adversarial noises involve only small areas and the noise-to-image ratio is less than 2%, which is comparable to LaVAN [13]. Our attack achieved good results on all 100 classes; the highest result was 99.17% and the lowest was 97.73%. The average fooling rates of the GAP [23] and LaVAN [13] are 52.0% and 74.1%, respectively, both of which are much lower than those of our method 98.23%. MIIP [35] achieved their targeted attack on the MS-COCO dataset [16], with the highest attack success rate of 93.62%, which is still slightly lower than ours. Our algorithm achieves state-of-the-art performance.

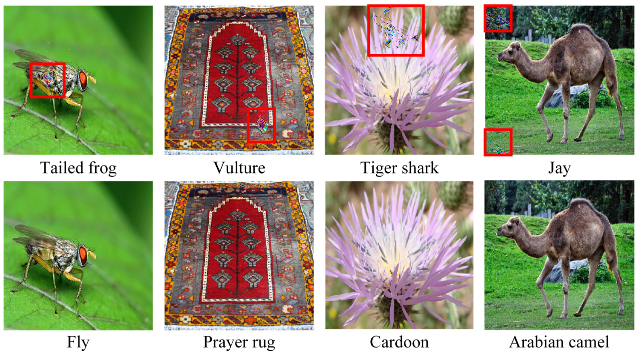

Figure 2. Examples of ImageNet samples with different targeted universal adversarial noises where the first row shows the adversarial images and the second row shows the clean images.

Transferability across data. To verify the conjecture that whether the generated targeted adversarial noises represent the common characteristics of the attacked category, we added some supplementary experiments to explore the relationship between the universal adversarial noises themselves and the target categories. For each target category in the ImageNet classification task, we input the generated universal adversarial noises directly to the classification model and found that the model output matched the target category 100% of the time. The result strongly supports our hypothesis.

Furthermore, we consider a realistic situation in which the attackers do not have access to the original training datasets for the target model. We performed the same targeted attacks on ImgaeNet classification aimed at the Inception-v3 model but took the training images from the MS-COCO datasets. We used 5,000 MS-COCO images as training samples to generate noises for 20 different classes in ImageNet and added the generated noises to 50,000 images in the original ImageNet validation datasets to test. The attacks on all 20 categories achieved high success and the average fooling rate ranged up to 94.71%. Figure 3 shows the generated noises for the object classes 'stingray' and 'chickadee' in the ImageNet and MS-COCO datasets, which are similar in shape, size, and location. Although the noises were generated by different data, they activated the same area. The situation is already obvious that our method is still effective in the absence of data-driven conditions. The results demonstrate that despite using distinct categories to train the universal adversarial noises, the noise corresponding to the same target class has similar appearances. This indicates that the optimization method may possess good transferability across different datasets. This also illustrates that our universal adversarial noises have found the characteristic information of the target model about the specific category, and have nothing to do with the data.

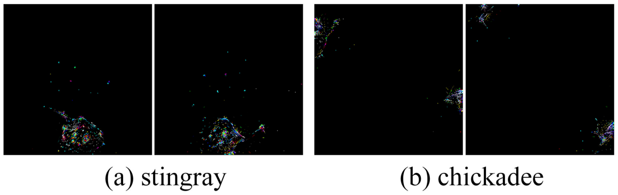
(a) stingray (b) chickadee

Figure 3. Universal adversarial noises of two classes generated by the ImageNet(Left) and COCO(Right) datasets.

3.2.2 Non-targeted Attacks

We selected a group of mainstream deep neural networks as our non-targeted attacks models, including VGG-16,19 [28], ResNet152,50 [12], GoogLeNet [30], and Inception-v3. These models were pre-trained on the ImageNet datasets using the weights published by Keras [1], except ResNet-152 [2] and GoogLeNet [1]. We conducted all experiments using the Keras [3] framework on an NVIDIA GeForce TITAN-X GPU and the experiment setting of targeted attacks is the same.

Table 1 shows the results of the non-targeted attacks for each model, with the best highlighted in bold. Clearly, our approach achieves the highest attack success rates for different classification models compared to the methods from previous studies [18, 20, 21, 23–25]. The fooling rates for the untargeted attacks by different models are all over 98%, except for GoogLeNet, whose attack success rate is slightly lower at 93.62% (which is still much higher than the results of other methods). For the non-targeted attacks against ResNet50 and Inception-v3 that are not listed in Table 1, our method reached success rates of 98.59% and 98.60%, respectively. These results show that SUAN's performance is not limited to any specific model architecture. We visualized the generated perturbation noises for the six attacked models, as shown in Figure 4.

Transferability across models. Transferability across models means that the adversarial perturbation generated for a specific network will fool other models as well. Table 1 also shows the transferability of different untargeted universal adversarial samples. The first column lists the different adversarial attack approaches, the second column lists the models on which the adversarial perturbations are trained, and the first row lists the names of the models that are victims of the untargeted attacks. The experimental results demonstrate that a similar model structure leads to higher transferability. The transferability of all approaches achieves higher scores across models with similar architectures; for example, the fooling rates of our method reach 93.43% and 95.42% for an adversarial attack transferred between VGG16 and VGG19. When transferring from VGG16 or VGG19 to other model structures, the attack success rate decreased by more than 10%.

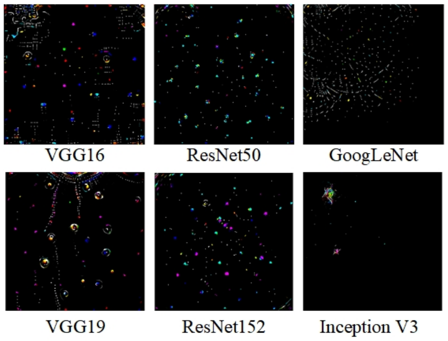
VGG16 ResNet50 GoogLeNet
VGG19 ResNet152 Inception V3

Figure 4. Visualized non-targeted universal adversarial noises for different models.

Table 1. Fooling rates of untargeted attacks and the transferability.

		VGG16	VGG19	GoogLeNet	ResNet152
UAP [18]	VGG16	78.30%	73.10%	56.50%	63.40%
	VGG19	73.50%	77.80%	53.60%	58.00%
	GoogLeNet	39.20%	39.80%	78.90%	45.50%
	ResNet152	47.00%	45.50%	50.50%	84.00%
Fast Feature Fool [21]	VGG16	47.10%	41.98%	34.33%	-
	VGG19	38.19%	43.62%	30.71%	-
	GoogLeNet	40.91%	40.17%	56.44%	-
GAP [23]	VGG16	93.90%	89.60%	-	52.20%
	VGG19	88.00%	94.90%	-	49.00%
	VGG16+VGG19	90.50%	90.10%	-	54.10%
	ResNet152	31.90%	30.60%	-	79.50%
NAG [25]	VGG16	77.57%	73.25%	67.38%	54.38%
	VGG19	80.56%	83.78%	74.48%	65.43%
	GoogLeNet	56.40%	59.14%	90.37%	59.22%
	ResNet152	52.17%	53.18%	62.33%	87.24%
AAA [24]	VGG16	71.59%	65.64%	60.74%	45.33%
	VGG19	69.45%	72.84%	68.79%	51.74%
	GoogLeNet	59.12%	48.61%	75.28%	47.81%
	ResNet152	47.21%	48.78%	56.41%	60.72%
GD-UAP [20]	VGG16	63.08%	56.04%	46.59%	36.84%
	VGG19	55.73%	64.67%	40.90%	35.81%
	GoogLeNet	37.95%	37.90%	71.44%	34.56%
	ResNet152	27.76%	26.52%	33.22%	37.30%
Ours(SUAN)	VGG16	**98.41%**	93.43%	80.59%	85.76%
	VGG19	95.42%	**98.10%**	81.19%	82.87%
	GoogLeNet	72.08%	70.51%	**93.62%**	74.21%
	ResNet152	71.89%	71.30%	75.11%	**98.87%**
Ours(UAP-DS)	VGG16	78.18%	77.91%	-	67.02%
	VGG19	75.73%	80.09%	-	65.73%
	ResNet152	65.64%	66.64%	-	85.82%

4. UAP-DS: Universal Adversarial Perturbation Generation Method with Deep Steganography

In this section, we took into account the quasi-imperceptibility of perturbation, proposed a new universal adversarial perturbation generation method based on SUAN combining with depth image steganography, called UAP-

DS, and conducted experiments on non-targeted attacks.

4.1. Generation Method

Zhang et al. [36] have proved that universal adversarial perturbation and deep steganography tasks have similar properties, i.e., the success of both stems from the destruction of high-frequency content of image content and DNN is highly sensitive to high-frequency content. Inspired by this, we applied the steganography module in the optimization process of USAN, and mapped the generated local adversarial noise to global perturbation through deep image steganography, and added it into the original images to achieve concealment. To implement the global mapping of noise, we abandon the adversarial sample synthesis method in Eq.(1). Instead of directly replacing the pixels, we use a steganography model to hide the noise $m \odot p$ composed of the mask m and the pattern p in the original image, as shown in Eq.(4).

$$x'_{adv} = HNet(m \odot p, x) \tag{4}$$

Algorithm 2 UAP-DS: universal adversarial perturbation generation with deep steganography.

```
Require: image $x \in X$, $y_{target}$, $y_{true}$
  Random initialize mask $m$ and pattern $p$; $|m_{best}|$ ←INF
  for $Epoch$=1 to $N$ do
    for minibatch $B \subset$ X do
      Compute loss function as Eq.(5) or Eq.(6).
      Update $m$, $p$ through Adam Optimizer.
      $x'_{adv}$=$HNet(m \odot p, x)$
    end for
    if average_attack_acc≥ $\varepsilon$ and $|m| < |m_{best}|$ then
      $m_{best}$=$m$,   $p_{best}$=$p$
    end if
  end for
  return $m_{best}, p_{best}$
```

Among them, $HNet(\cdot)$ represents the pre-trained steganography model and x'_{adv} refers to the new generated adversarial samples. According to the equation, we can map the locally visible perturbation to the global image. Because of the global distribution of perturbation, there is no need to control the amount of the added noises like USAN. At this time, the constraint on the mask is unnecessary, and rather the distance between the original sample and the adversarial sample needs to be constrained to ensure the imperceptibility of the perturbation. So, optimization functions of the targeted attack and the untargeted attack are updated to Eq.(5) and (6) respectively, where $L_2(\cdot)$ means the L_2 distance between the original sample x and the adversarial sample x'_{adv}, and restrict its use of the standard metric 2000. Other optimization details are consistent with SUAN. The universal adversarial perturbation generation algorithm with steganography is shown in Algorithm 2.

$$\min_{m,p} \{CE(y_{target}, f(x'_{adv})) + L_2(x, x'_{adv})\} \tag{5}$$

$$\min_{m,p} \{-CE(y_{true}, f(x'_{adv})) + L_2(x, x'_{adv})\} \tag{6}$$

4.2. Experiments Results

We conducted all experiments using the same experiment settings as the above untargeted attacks in section 3.2.2. The steganography model uses the pre-trained model published by [7], the size of the input image is 256×256, and the channel is 6. We used the original image as the cover image, and adversarial perturbation as the secret image, the input image is connected by the two images.

Figure 5 shows the original sample and the adversarial sample after adding adversarial perturbation through deep image steganography. The results show that deep steganography successfully realized the invisibility of adversarial perturbation. We applied deep steganography to carry out non-targeted attacks on each model and tested the success rate of the attack, as shown in Table 1.

Transferability across models. Similarly, we evaluated the transferability of the generated universal adversarial perturbation between multiple models through steganography. Table 1 also shows the transferability of the improved method. The experimental results also demonstrate that a similar model structure leads to higher transferability. But for models with different architectures, more than 50% transferability can also be achieved.

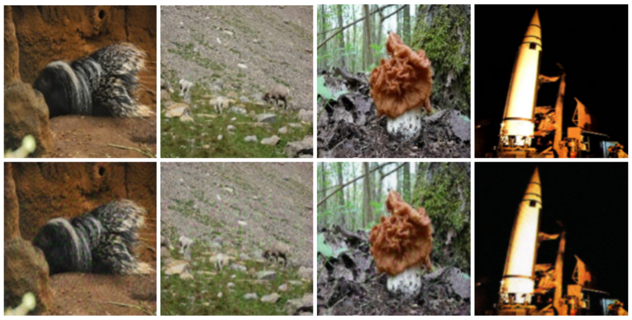

Figure 5. Original samples (the first row) and adversarial samples (the second row) with universal adversarial perturbation.

Performance comparison. We compared UAP-DS with other universal adversarial perturbation generation methods listed in Table 1, and the results show that our approach combined with deep steganography surpasses them overall in attack performance. Compared with these methods, our results have higher attack success rates on the four models, exceeding the range from 0.61 to 53.57, which is quite impressive. Meanwhile, our method is better than them in terms of model transferability. Unfortunately, compared with GAP [23], our method is inferior to them on VGG16 and VGG19 models, but exceeded them on ResNet152. We also compared UAP-DS with [9], which is also a universal adversarial sample generation method based on steganography. The results are shown in Table 2, where the first row represents the attacked target model, and images 1 $\sim$ 5 in [9] respectively represent the secret images used to hide in the clean samples. For ResNet50 and Inception-v3 models, our method is almost close to their best results, with a difference of only about 2%, and we are better than the remaining four secret images. But their method has considerable limitations, which is equivalent to a pre-processing process, without generating an adversarial perturbation but hiding a secret image in the original image instead, which relies heavily on steganographic content. We can also achieve better attack performance by performing steganography on perturbations, and it is significantly better than most all existing methods.

Table 2. Fooling rates of universal adversarial perturbations with steganography on untargeted attacks.

		VGG16	ResNet50	Inception-v3
SUAP [9]	Image 1	87.19%	84.77%	79.19%
	Image 2	84.96%	82.11%	74.51%
	Image 3	73.88%	71.76%	64.59%
	Image 4	38.65%	42.72%	41.25%
	Image 5	38.45%	42.61%	41.78%
UAP-DS		78.18%	84.64%	77.36%

5. Discussion

In this section, we analyzed different adversarial examples generated by the two methods to discuss the impact of different types of perturbations. We explored how universal adversarial examples work. Specifically, we probed into the impact of universal adversarial perturbation on natural images. In addition, we compared the two methods and discussed their advantages and limitations, and made a trade-off between the two in the actual attack and application scenario.

The attack effects of two universal adversarial perturbations. Intuitively, the two generation methods we proposed are consistent in the core optimization algorithm, but the forms of interference to natural images have their own merits, where one is the local visible noises, and the other is the global imperceptible perturbation. This lead to a slight difference in their impact on natural images. We focus on the high-frequency content of different adversarial samples, to which deep neural networks are more sensitive.

We first decompose the perturbations and clean samples of the first 10 target classes of the generated Imagenet into four frequency domain subbands through DWT. Next, we calculate the cosine similarity on the four frequency domain subbands, LL(low-frequency), LH(vertical high frequency), HL(horizontal high frequency), and HH(high-frequency) of the corresponding categories of the clean image and the perturbation image. For high-frequency content, their cosine similarity is close to 1, indicating that the impact of the generated universal perturbation on high-frequency content is very important and its characteristics are quite close to the target category. The result is shown in Figure 6.

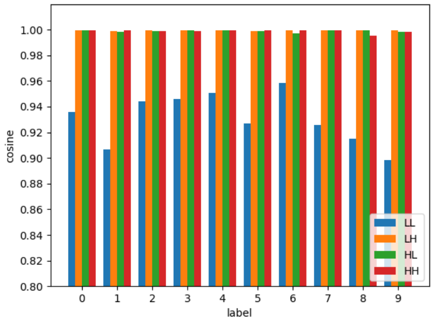

Figure 6. Cosine similarity between clean images and adversarial images in different frequency domains. (LL, LH, HL and HH represent the content information of the original image, the high-frequency information of the horizontal direction, the vertical direction and the diagonal of the image, respectively.)

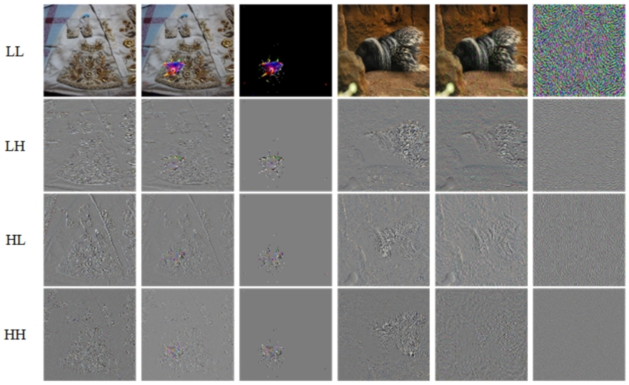

Figure 7. Discrete wavelet transform (DWT) of natural and adversarial images. The first three columns and the last three columns are the visualization results of SUAN and UAP-DS.

Meanwhile, we visualized the wave transformation of the universal adversarial perturbations with steganography, and the four frequency components of the clean sample, adversarial sample and perturbation were shown in Figure 7. It can be observed that for clean samples, after adding generated the universal adversarial perturbation, the frequency features of the clean sample will obviously be covered. It can be clearly seen that the adversarial perturbation after steganography has minimal impact on the LL, LH and HL components, but only produces obvious interference on the HH. We also performed the same visual analysis on the adversarial noises generated by the SUAN, as shown in the first three columns of Figure 7. It can be observed that for clean samples, after adding generated the universal adversarial noise, the high-frequency features of the original sample will be also obviously covered. However, The noise generated by the SUAN has a certain impact on the four components, which proves from the side that the UAP-DS can effectively map the noise to the global perturbation and maintain good concealment. However, this is not enough to illustrate which algorithm is better, and we then analyze the advantages and limitations of both respectively.

Advantage and limitation. On the one hand, for the SUAN, although the universal adversarial noises generated by the simple method has high fooling rates, the restriction of the method on the perturbation causes the generated adversarial perturbation only to be concentrated in a local area of the image while others are distributed over the image in blocks and has obvious visibility. The reason is that mask and pattern interact with each other, that is, fewer disturbed pixels lead to higher pixel intensity. They determine the effectiveness of the synthetic adversarial noises, which results in that the generated adversarial perturbation is not strictly imperceptible. Whether it is considered from the perspective of the models or humans, it is not a good thing, even though our perturbations involve only small areas and the noise-to-image ratio is less than 2%. These noises are not limited to a regular local area as [13] but have a certain degree of dispersion. But it still has similar limitations, that is, the perturbation area is easily partially covered or the image is compressed, which may lead to a decrease in attack performance.

On the other hand, regrettably, although the perturbation generated by UAP-DS satisfies the quasi-imperceptibility to humans, it sacrifices a little attack success rate (It is still comparable or higher than some current generation methods). We analyzed the possible reasons for this. We added a steganography module in the optimization process to ensure quasi-imperceptibility, but it does not guarantee that it can produce the same effect for all images. In other words, steganography maps a fixed pattern of noise to different content for different samples, which leads to a reduction in performance, as shown in Figure 8. It can be clearly seen that for the two different images, the same adversarial noises are hidden to them, which produces different disturbance effects, as shown in the third column of Figure 8. This phenomenon limits the optimization process of universal adversarial perturbations, making it difficult for the average fooling rate to reach the attack threshold ε. And in this case, the loss is difficult to further decrease and stabilizes after a few iterations, which explains why the UAP-DS method does not achieve a attack success rate comparable to SUAN.

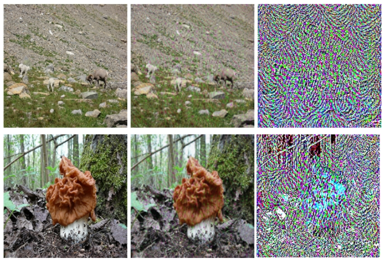

Figure 8. Universal adversarial perturbation steganographic content for different images.

So, what is the point of these two universal adversarial perturbations? From an attacker‘s perspective, when implementing a malicious adversarial attack, the attackers need to consider not only the performance and cost of the attack, but also the concealment and imperceptibility of the perturbation. Regardless of whether it is for humans or for models, adversarial samples need to survive from detection or preprocessing algorithms, where UAP-DS seems more practical in this case. However, if in order to implement high-efficiency and fast attacks, such as assessing the robustness of the target model and investigating the blind spots of models [13], universal adversarial noises can be regarded as an effective solution. In addition, SUAN provides the possibility to implement physical attacks in real-world scenarios, which has an intuitive difference from performing adversarial attacks in the digital domain directly. That is, the adversarial perturbation added to real objects must be perceptible to ensure it can be captured by the physical device and that the image information is not lost during the data transmission process. We believe that both generation methods can make sense.

6. Conclusions

In this paper, we proposed two simple and effective methods for generating universal adversarial samples, SUAN and UAP-DS, which can effectively generate locally visible adversarial noise and imperceptible global perturbation. First, we mainly focused on the attack success rate of universal adversarial samples and proposed an optimization method to constraint pixel sensitivity and amount of perturbation, which achieves excellent results in both targeted and non-targeted, including cross-model and cross-data. Based on SUAN, we further considered the imperceptibility of perturbation and propose an optimization algorithm combined with deep steganography, UAP-DS, to map local adversarial noises to global perturbation. It achieved equivalent even surpass than advanced methods in the non-targeted attack. In addition, we analyzed and discussed their respective advantages and limitations, and achieved a trade-off between the two. In future work, we intend to deeply study the relationship between attack success rate and imperceptibility of universal adversarial perturbation, and extend these two methods to other fields based on deep neural networks.

References

[1] Keras pre-trained model of googlenet in ilsvrc 2012. https://gist.github.com/joelouismarino/a2ede9ab3928f999575423b9887abd14. 5

[2] Keras pre-trained model of resnet-152 in ilsvrc 2012. https://gist.github.com/flyyufelix/7e2eafb149f72f4d38dd661882c554a6. 5

[3] Keras: The python deep learning library. https://keras.io/. 5

[4] Naveed Akhtar and Ajmal Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, 6:14410–14430, 2018. 1

[5] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*, pages 173–182, 2016. 1

[6] Divyam Anshumaan, Akshay Agarwal, Mayank Vatsa, and Richa Singh. Wavetransform: Crafting adversarial examples via input decomposition. In *European Conference on Computer Vision*, pages 152–168. Springer, 2020. 1, 2

[7] Shumeet Baluja. Hiding images in plain sight: Deep steganography. *Advances in Neural Information Processing Systems*, 30:2069–2079, 2017. 6

[8] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017. 1

[9] Salah Ud Din, Naveed Akhtar, Shahzad Younis, Faisal Shafait, Atif Mansoor, and Muhammad Shafique. Steganographic universal adversarial perturbations. *Pattern Recognition Letters*, 135:146–152, 2020. 1, 2, 7

[10] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 1

[11] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of Field Robotics*, 2019. 1

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[13] Danny Karmon, Daniel Zoran, and Yoav Goldberg. Lavan: Localized and visible adversarial noise. *arXiv preprint arXiv:1801.02608*, 2018. 1, 2, 4, 8

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3

[15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016. 1

[16] Tsung Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context. 2014. 4

[17] Xuankai Liu, Fengting Li, Bihan Wen, and Qi Li. Removing backdoor-based watermarks in neural networks with limited data. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 10149–10156. IEEE, 2021. 2

[18] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1765–1773, 2017. 1, 2, 5

[19] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2574–2582, 2016. 1, 2

[20] Konda Reddy Mopuri, Aditya Ganeshan, and R Venkatesh Babu. Generalizable data-free objective for crafting universal adversarial perturbations. *IEEE transactions on pattern analysis and machine intelligence*, 41(10):2452–2465, 2018. 1, 2, 5

[21] Konda Reddy Mopuri, Utsav Garg, and R Venkatesh Babu. Fast feature fool: A data independent approach to universal adversarial perturbations. *arXiv preprint arXiv:1707.05572*, 2017. 1, 2, 5

[22] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversarial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016. 1

[23] Omid Poursaeed, Isay Katsman, Bicheng Gao, and Serge Belongie. Generative adversarial perturbations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4422–4431, 2018. 1, 2, 4, 5, 7

[24] Konda Reddy Mopuri, Phani Krishna Uppala, and R Venkatesh Babu. Ask, acquire, and attack: Data-free uap generation using class impressions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–34, 2018. 1, 2, 5

[25] Konda Reddy Mopuri, Utkarsh Ojha, Utsav Garg, and R Venkatesh Babu. Nag: Network for adversary generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 742–751, 2018. 1, 2, 5

[26] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 2015. 4

[27] Ali Shafahi, Mahyar Najibi, Zheng Xu, John Dickerson, Larry S Davis, and Tom Goldstein. Universal adversarial training. *arXiv preprint arXiv:1811.11304*, 2018. 1, 2

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[29] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019. 1

[30] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent

Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015. 5

[31] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 4

[32] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

[33] Sakshi Udeshi, Shanshan Peng, Gerald Woo, Lionell Loh, Louth Rawshan, and Sudipta Chattopadhyay. Model agnostic defence against backdoor attacks in machine learning. *arXiv preprint arXiv:1908.02203*, 2019. 2

[34] Bolun Wang, Yuanshun Yao, Shawn Shan, Huiying Li, Bimal Viswanath, Haitao Zheng, and Ben Y Zhao. Neural cleanse: Identifying and mitigating backdoor attacks in neural networks. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 707–723. IEEE, 2019. 3

[35] Chaoning Zhang, Philipp Benz, Tooba Imtiaz, and In So Kweon. Understanding adversarial examples from the mutual influence of images and perturbations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14521–14530, 2020. 2, 4

[36] Chaoning Zhang, Philipp Benz, Adil Karjauv, and In So Kweon. Universal adversarial perturbations through the lens of deep steganography: Towards a fourier perspective. *arXiv preprint arXiv:2102.06479*, 2021. 1, 3, 6