Machine Learning (2023) 112:1597–1626
https://doi.org/10.1007/s10994-023-06306-z

Generalizing universal adversarial perturbations for deep neural networks

Yanghao Zhang[1] · Wenjie Ruan[1] · Fu Wang[1] · Xiaowei Huang[2]

Received: 15 December 2021 / Revised: 30 November 2022 / Accepted: 19 January 2023 /
Published online: 22 March 2023

Abstract

Previous studies have shown that universal adversarial attacks can fool deep neural networks over a large set of input images with a single human-invisible perturbation. However, current methods for universal adversarial attacks are based on additive perturbation, which enables misclassification by directly adding the perturbation on the input images. In this paper, for the first time, we show that a universal adversarial attack can also be achieved through spatial transformation (non-additive). More importantly, to unify both additive and non-additive perturbations, we propose a novel unified yet flexible framework for universal adversarial attacks, called GUAP, which can initiate attacks by ℓ_∞-norm (additive) perturbation, spatially-transformed (non-additive) perturbation, or a combination of both. Extensive experiments are conducted on two computer vision scenarios, including image classification and semantic segmentation tasks, which contain CIFAR-10, ImageNet and Cityscapes datasets with a number of different deep neural network models, including GoogLeNet, VGG16/19, ResNet101/152, DenseNet121, and FCN-8s. Empirical experiments demonstrate that GUAP can obtain higher attack success rates on these datasets compared to state-of-the-art universal adversarial attacks. In addition, we also demonstrate how universal adversarial training benefits the robustness of the model against universal attacks. We release our tool **GUAP** on https://github.com/TrustAI/GUAP.

Keywords Deep learning · Adversarial examples · Security · Deep neural networks

Editor: Lijun Zhang.

✉ Wenjie Ruan
wjie.ruan@gmail.com

Yanghao Zhang
yanghao.zhang@outlook.com

Fu Wang
fw377@exeter.ac.uk

Xiaowei Huang
xiaowei.huang@liverpool.ac.uk

[1] College of Engineering, Mathematics and Physical Sciences, University of Exeter, Exeter EX4 4QF, England, UK

[2] Department of Computer Science, University of Liverpool, Liverpool L69 3BX, England, UK

1 Introduction

Although deep neural networks (DNNs) have achieved great success in a wide range of applications, such as computer vision (Russakovsky et al., 2015), natural language processing (Collobert et al., 2011), yet recently some researchers have demonstrated that DNNs are vulnerable to adversarial examples or attacks (Szegedy et al., 2014; Carlini and Wagner, 2017; Huang et al., 2019, 2020; Yin et al., 2022). Adversarial examples are generated by adding small perturbations to an input, sometimes imperceptible to humans, that can enable the neural network to make an incorrect classification result (Zhang et al., 2019; Wu et al., 2020; Sun et al., 2018b; Mu et al., 2021; Wang et al., 2022). Taking Fig. 1 as an example, by adding a human-invisible perturbation, a well-trained VGG19 neural network can be easily fooled such that it incorrectly classifies the image 'ice lolly' as 'candle'.

Thus, adversarial examples (Goodfellow et al., 2014b; Xu et al., 2022) have become a severe risk, especially when DNNs are applied to safety-critical applications such as medical record analysis (Sun et al., 2018a), malware detection (Wang et al., 2017), and autonomous vehicles (Zhang et al., 2023; Wu and Ruan, 2021; Mu et al., 2022). Most existing adversarial attack methods focus on generating an adversarial perturbation over a specific input (Carlini and Wagner, 2017; Zhang et al., 2019; Goodfellow et al., 2014b; Ruan et al., 2018). These perturbations are image-specific, i.e., different perturbations are generated for different inputs. An adversarial perturbation of this type may expose the weakness of the network within the local precinct of the original image in the input domain, but it cannot directly support the analysis of global robustness (Ruan et al., 2019). In order to support this, the concept of universal adversarial perturbation is considered, which can fool a well-trained neural network on a set of, ideally, all input images from the data distribution. (Moosavi-Dezfooli et al., 2017) firstly showed the existence of the Universal Adversarial Perturbation (UAP) and presented an iterative algorithm to compute it based on a set of input images. Unlike UAP which employs the iterative method, some other works also showed that generative models could be used for crafting universal perturbation (Hayes and Danezis, 2018; Poursaeed et al., 2018; Reddy Mopuri et al., 2018), with the aim of capturing the distribution of adversarial perturbations and producing a higher fooling rate.

Until today, existing universal adversarial attacks are all additive (i.e., they make DNNs misclassified when the perturbation is directly added to images) and based on ℓ_p-norm distance to constrain the magnitude of the perturbation. However, a transformation-based perturbation can also be out of the range of ℓ_p-norm ball, but maintains imperceptibility. For example, as shown in Fig. 1, an adversarial example generated by spatial transformation (Xiao et al., 2018b) is almost "the same" as human perception but results in a large ℓ_∞ distance. This observation led to another type of adversarial perturbation, i.e., non-additive perturbations. Generally speaking, a non-additive perturbation can be seen as a function generating the transformation for the input, and hence a generalisation to the additive perturbation.

Recently, a particular type of non-additive perturbation, i.e., adversarially spatial transformation, has increasingly attracted the attention of the community. Some researchers observed that deep neural models suffer from spatial variants of input data, whereas humans are usually less sensitive to such spatial distortions (Lenc and Vedaldi, 2015; Jaderberg et al., 2015; Wang et al., 2021; Zhang et al., 2022). In this regard, some pioneering works have emerged recently with the aim of generating spatially transformed adversarial examples (Engstrom et al., 2009; Xiao et al., 2018b; Zhang et al., 2020). For instance, (Engstrom et al., 2009) identified that even simply rotating and/

or translating the benign images can significantly degrade classification performance in DNNs. (Xiao et al., 2018b) proposed an adversarial attack method that can generate perceptually realistic adversarial examples by perturbing the spatial locations of pixels.

However, those non-additive methods can only generate a specific perturbation that is workable on a given image, rather than a universal one that can fool a deep neural network over the whole dataset. And current works on universal perturbation are mostly based on additive approaches. Therefore, in this paper, our first aim is to design a novel universal adversarial attack method that can generate *non-additive* perturbation, in this paper specifically we use spatial transformation to fool DNNs over a large number of inputs simultaneously. We then try to further surpass current universal attack approaches with an aim to *unify* both additive and non-additive perturbations under the *same* universal attack framework. As a result, we propose a *unified* and *flexible* framework, called *GUAP*, that can capture the distributions of unknown additive and non-additive adversarial perturbations *jointly* for crafting Generalized Universal Adversarial Perturbations. Specifically, the generalised universal adversarial attack can be achieved via spatial transformation (non-additive) perturbation or ℓ_∞-norm based (additive) perturbations or the combination of both. Extensive experiments are conducted to evaluate the effectiveness of our framework. In summary, the contributions of this paper lie in the following aspects:

- We propose a novel *unified* framework, named GUAP, for universal adversarial attacks. As the first of its kind, GUAP can generate either ℓ_∞-bounded (additive) or spatial transformation (non-additive) perturbations, or a combination of both, which considerably generalises the attacking capability of current universal attack methods.
- To our knowledge, GUAP is also one of the first attempts to initiate *universal* adversarial attacks on DNNs by *spatial* transformations. We show that, with spatial transformations, GUAP is able to generate less distinguishable adversarial examples with significantly better attacking performance than existing state-of-the-art approaches, leading to significant improvement in the attack success rate for some computer vision tasks such as image classification and semantic segmentation.
- The proposed method fits the setting of semi-white attack, which can synthesise adversarial images without accessing the structures and parameters of the original target model. In addition, with the universal and input-agnostic properties, the produced perturbation can be used directly in the attacking phase without any further computation, which provides excellent efficiency in practice.

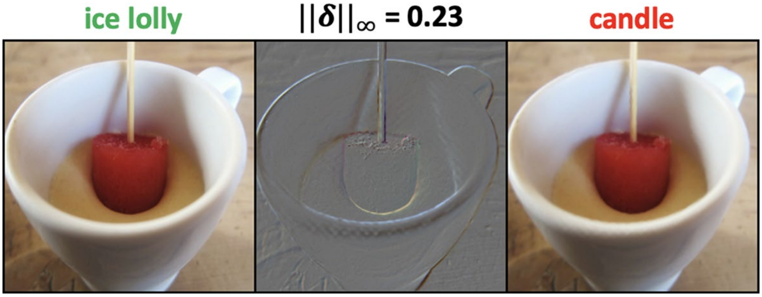


Fig. 1 Adversarial example generated by spatial transformation, VGG19 neural network can be easily fooled so it incorrectly classifies the "ice lolly" as "candle", the image in the middle column represents the intermediate perturbation scaled with the minimum of 0 and the maximum of 1

- We show that the proposed GUAP can also work on image semantic segmentation, thus this framework can be adapted to other similar tasks. Moreover, we explore the universal adversarial training (AT) strategy as a defence method against UAP, which demonstrates the effectiveness of GUAP-AT against traditional UAP attacks.

2 Related work

We firstly review the local adversarial attacks that are spatial-based methods and non-universal. Then we discuss the related works in universal adversarial attacks, which are all based on additive perturbations so far. We also summarise some current adversarial defence methods against universal perturbations.

Different from all the other research in universal perturbation, our proposed GUAP framework is more flexible than all current adversarial attacks in terms of attacking capability. Our work is not only universal but also could be additive (i.e., ℓ_∞-bounded), non-additive (i.e., spatial transformation), or both, under different scenarios. Last but not least, we take a further look at universal adversarial training by injecting the universal perturbation from our framework during training to observe its robustness against conventional universal attacks. Table 1 indicates the uniqueness of our research. To our knowledge, there is no existing work that can exactly achieve the same functionalities as ours.

2.1 Local adversarial perturbations

The concept of local adversarial perturbation is inherited from the adversarial example (Szegedy et al., 2014), considering an image classifier and given a specific image, when the elaborate local adversarial perturbation is applied to the images, it will mislead the well-trained neural network to make an incorrect prediction (Goodfellow et al., 2014b; Carlini and Wagner, 2017). The common type of local adversarial perturbation is an additive noise that directly perturbs the image and its magnitude is constrained by the ℓ_p norm. However, there exist other types of local adversarial perturbation, such as some transformations, and they are not bounded by the ℓ_p norm but still maintain high imperceptibility to humans. We review the relevant literature in the following.

Fawzi and Frossard (2015) firstly studied the in-variance of deep networks to spatial transformations, revealing that convolutional neural networks are not robust against rotations, translations, and dilation. Xiao et al., (2018b) also argued that the traditional ℓ_p-norm based constraint may not be an ideal criterion for measuring the similarity of human perception on two images. They proposed an optimisation method, which is capable of generating perceptually realistic adversarial examples with a high fooling rate by perturbing the positions of pixels instead of adding perturbation to the clean image directly. It manipulates an image according to a pixel replacement rule named ‘flow field’. To ensure that an adversarial image is perceptually close to the benign one, it also minimises local geometric distortion instead of the ℓ_p-norm distance in the objective function. Xiao et al., (2018b) also conducted a human perceptual study, which showed that spatially transformed adversarial perturbations are more indistinguishable for humans, compared to additive adversarial perturbations generated by Goodfellow et al., (2014b), Carlini and Wagner, (2017).

Engstrom et al., (2009) noted that existing adversarial methods are too complicated, and generate contrived adversarial examples that are highly unlikely to occur ‘naturally’. They thus showed that neural networks are even quite vulnerable to simple rotations. They

Table 1 Comparison among existing related works

Method	Attacking capability			Computer vision task		Defense
	Additive	Non-additive	Universal	Image classification	Semantic segmentation	Adversarial training
UAP[1]	✓		✓	✓		
FFF[2]	✓		✓	✓		
UAN[3]	✓		✓	✓		
GAP[4]	✓		✓	✓	✓	
NAG[5]	✓		✓	✓		
StAdv[6]		✓		✓		
Engstrom et al.[7]	✓	✓	✓	✓		
UAP-Seg[8]	✓		✓		✓	
UAT[9]	✓		✓	✓		✓
Cosine-UAP[10]	✓		✓	✓		
GUAP (Ours)	✓	✓	✓	✓	✓	✓

[1] Moosavi-Dezfooli et al., (2017)

[2] Mopuri et al., (2017)

[3] Hayes and Danezis, (2018)

[4] Poursaeed et al., (2018)

[5] Reddy Mopuri et al., (2018)

[6] Xiao et al., (2018b)

[7] Engstrom et al., (2009)

[8] Hendrik Metzen et al., (2017)

[9] Shafahi et al., (2020)

[10] Zhang et al., (2021a)

restricted the transformation to a range of ±30° × ±3 pixels, then adopted grid search to explore the parameters space, and exhaustively tested all possibilities. It reveals that simply rotating and/or translating benign images can result in a significant degradation of the performance of the target classifier. Furthermore, a combination adversary was also considered, which performs all possible spatial transformations (through exhaustive grid search) and then applies a ℓ_p-bounded PGD attack (Madry et al., 2017) on top. From the experimental observation, they further indicated that the robustness of these two kinds of perturbation is orthogonal to each other.

2.2 Universal adversarial perturbations

Different from local adversarial perturbation that is designed for a specific image, universal adversarial perturbation (Moosavi-Dezfooli et al., 2017) (UAP) aims to fool a well-trained neural network on a set of, ideally, all input images. Currently, the existing universal adversarial attacks are all additive-based noise, whose magnitude is constrained by ℓ_p-norm distance. Below, we review some classic methods for generating universal adversarial perturbation and also some defence approaches against UAP.

UAP proposed by Moosavi-Dezfooli et al., (2017) is the first work that identifies the vulnerability of DNNs to universal adversarial perturbations. To create a universal perturbation, UAP integrates the learned perturbations from each iteration. If the combination cannot mislead the target model, UAP will find a new perturbation followed by projecting the new perturbation onto the ℓ_p norm ball to ensure that it is small enough and meets the distance constraints. This method will keep running until the empirical error of the sample set is sufficiently large or the threshold error rate is satisfied. The optimisation strategy to find minimal noise is adapted from previous work, i.e., DeepFool (Moosavi-Dezfooli et al., 2016). Recently, (Shafahi et al., 2020) proposed to optimise the perturbation directly with a sum of the projected gradient, which shows a promising result. (Zhang et al., 2021a) followed a similar process but utilised the cosine similarity as the loss function.

On the other hand, there were three previous works that leveraged the generative model for universal adversarial attacks, i.e., UAN (Hayes and Danezis, 2018), GAP (Poursaeed et al., 2018) and NAG (Reddy Mopuri et al., 2018). All of them attempted to capture the distribution of adversarial perturbations, which show some improvements compared to UAP (Moosavi-Dezfooli et al., 2017). However, their implementations have some differences. Specifically, 'Universal Adversarial Network' ('UAN') (Hayes and Danezis, 2018) is composed of stacks of deconvolution layers, batch normalisation layers with activation function, and several fully-connected layers on the top. For UAN, it includes a ℓ_p distance minimisation term in the objective function, and the magnitude of the generated noise is controlled by a scaling factor, which increases gradually during training. (Poursaeed et al., 2018) employed a ResNet-based generator from (Johnson et al., 2016) to generate universal adversarial perturbations, named 'GAP'. Before adding the perturbation to an image, the constraint of the noise is restricted directly by a fixed scaling factor. Furthermore, this work was also extended to semantic image segmentation, such as UAP-Seg (Hendrik Metzen et al., 2017), which generates a targeted universal adversarial perturbation for semantic segmentation models through a similar iterative algorithm in (Moosavi-Dezfooli et al., 2017). (Reddy Mopuri et al., 2018) proposed a generative model called 'NAG', which consists of seven deconvolution layers and a fully-connected layer. NAG presented an objective function to reduce the prediction confidence of the true label and increase that of other labels. In addition, a diversity term was introduced in the objective function to encourage the diversity of perturbations. Furthermore, (Mopuri et al., 2017, 2019) revealed that, a single perturbation can fool the majority of images in a data-free setting by maximising the output after activation function over multiple layers in the target model, but this may sacrifice the success rate of the attack compared to UAPs (Moosavi-Dezfooli et al., 2017).

Regarding the defence against universal attacks, only a few works exist. The first defence strategy is the perturbation rectifying network (Akhtar et al., 2018), which preprocesses input images to remove universal perturbations. It builds a binary classifier to detect the existence of the universal perturbation, and then replaces the original input with the rectified image when performing the final classification. Later, borrowing the idea of per-instance adversarial training, (Mummadi et al., 2019) proposed a shared adversarial training strategy to use shared gradients in the image heap. Although it maintains good clean accuracy, the robust accuracy under UAP is still not satisfactory. Recently, (Shafahi et al., 2020) proposed universal adversarial training to resist UAPs. In essence, it utilises every image in the training dataset and generates FGSM-based batch universal adversarial perturbations for adversarial training, but it requires many epochs during training to achieve decent performance.

3 Generalized universal adversarial perturbations

Previous research on UAPs is based on ℓ_p-bounded adversarial perturbations, which requires generated images to be close to the benign examples within a given ℓ_p norm ball. This is based on the assumption that human perception can be quantified by ℓ_p norm. However, for a non-additive method such as spatial transformation, the norm ℓ_p is difficult to capture the perceptional indistinguishability of humans, as proved in (Engstrom et al., 2009). On the other hand, prior non-additive approaches only generate specific perturbations for a given image, rather than a universal one over the whole dataset.

As a result, we propose a framework to work with both ℓ_p norm-based (additive) and spatial transformation-based (non-additive) methods to craft universal adversarial examples. The proposed framework leverages the training process in an end-to-end generative model to jointly generate universal flow and universal noise. Overall, the framework of GUAP is elaborated in Algorithm 1.

Given a random fixed input noise $z \sim \mathcal{N}(0, 1)$, it will be fed into our generalized adversarial perturbation generator, which outputs a universal noise and a universal flow field at the same time. The existing generative model-based methods can only produce a universal additive noise, while our proposed approach will simultaneously yield another universal flow field. After the scaling operation according to the pre-defined perturbation constraints τ and ϵ, the spatial and noise perturbations are then performed successively to obtain the adversarial examples, such that the task-related objective loss can be calculated for optimising the parameters of the generator. More details will be described in the following sections.

Algorithm 1 Generalized Universal Adversarial Perturbations

```
Input: Training set $X$, total epochs $T$, initial input vector $z \sim \mathcal{N}(0,1)$, adversarial radius $\epsilon$, Generator $\mathcal{G}_\theta$, maximum flow $\tau$ , target model $h$, function $\mathcal{F}_f(\cdot)$ for performing spatial transformation and number of mini-batches $M$
Output: A universal flow field $f$ and a universal perturbation noise $\delta$
for $t = 1 \dots T$ do
    for $i = 1 \dots M$ do
        $x = X_i$
        $\delta_0, f_0 = \mathcal{G}_\theta(z)$
        $f = \tau / \hat{L}_{flow}(f_0) \cdot f_0,\ \delta = \epsilon / \|\delta_0\|_\infty \cdot \delta_0$        ▷ Scaling operations
        $x_{adv} = \mathcal{F}_f(x) + \delta$        ▷ Perform spatial and noise perturbations
        $x_{adv} = \text{Clip}(x_{adv}, 0, 1)$
        $\theta = \theta - \nabla_\theta L_{adv}\left(h(x_{adv}), h(x)\right)$        ▷ Update model weights with loss
    end for
end for
```

3.1 Problem definition

As our applications mainly focus on image classification, here we give the problem definition based on this task, while this can be easily extended to other tasks such as semantic segmentation (Hendrik Metzen et al., 2017) and object detection (Zhang et al., 2021b). Given a data sample $x := \{x_i \in \mathbb{R}^d, i = 1, 2, ..., n\}$ belonging to the benign data set $\mathcal{X}$ from $\mathcal{C}$ different classes, and there exist ground truth relations from inputs to labels: $\mathcal{X} \in \mathbb{R}^{n \times d} \rightarrow \mathcal{Y} \in \mathbb{R}^n$, a target DNN classifier h will confidently output a prediction $h(x) \in \{1, 2, ..., \mathcal{C}\}$ to each input image x_i. We assume that all features of the images are normalised in the range [0, 1], and h has achieved high accuracy on the benign image set, such that $h(\mathcal{X}) \approx \mathcal{Y}$. Moreover, we denote $\mathcal{A}$ as the space spanned by adversarial examples, such that, given an input x, the corresponding adversarial example $x_{adv} \in \mathcal{A}$ is able to fool the target model h with high probability while resembles the natural image x. For untargeted adversarial attacks, we can express this as $h(x) \neq h(x_{adv})$ formally, meanwhile satisfying the defined distance metric. Universal attacks focus on finding universal noise δ for all inputs, which generates adversarial examples $x + \delta$ that can fool the target classifier. Commonly, the maximum perturbation constraint is controlled by the ℓ_p norm ball, e.g. $\|\delta\|_\infty \leq \epsilon$.

However, we will consider both non-additive and additive perturbations, i.e., spatial transformation-based and ℓ_∞ norm-based. The adversarial sample x_{adv} with respect to *any* input data x can be represented as:

$$x_{adv} = \mathcal{F}_f(x) + \delta \tag{1}$$

where $\mathcal{F}_f(\cdot)$ is the function of adversarially spatial transformation (Xiao et al., 2018b), containing a flow field f to indicate the replacement rules for each pixel. Therefore, in our case, f and δ are the *universal* flow field and *universal* noise for performing spatial transformation-based perturbation and ℓ_∞-bounded perturbation, respectively, over the whole dataset. We inherit a hyper-parameter ϵ from traditional UAPs to denote the magnitude of the universal noise and further introduce another parameter τ to restrict the perturbation caused by the spatial transformation.

3.2 Universal spatial transformations

The adversarial spatial transformed attack was proposed in (Xiao et al., 2018b). It is an image-specific method that optimises flow fields for different input images, and generates adversarial example by manipulating each pixel value based on a learned flow field. We also utilise the flow field to perform the spatial transformation but in an image-agnostic manner. Formally, we define the input space of the image as $x \in [0, 1]^{c \times h \times w}$. A universal flow field $f \in [-1, 1]^{2 \times h \times w}$ represents a rule of pixel replacement: for a pixel $x^{(i)}$ at the location $(u^{(i)}, v^{(i)})$, its corresponding coordinate in f, i.e., $f_i = (\Delta u^{(i)}, \Delta v^{(i)})$, denotes the direction and magnitude for replacement of the pixel value of $x^{(i)}$. Let x_{st} stand for the spatial transformed image from the benign image x via the flow field f, the relation between the renewed coordinate and the original pixel location can be expressed as:

$$\left(u^{(i)}, v^{(i)}\right) = \left(u_{st}^{(i)} + \Delta u^{(i)}, v_{st}^{(i)} + \Delta v^{(i)}\right) \tag{2}$$

for all i in $\{1, \ldots, h \times w\}$. Note that, in this paper, the same flow field $(\Delta u^{(i)}, \Delta v^{(i)})$ is applied to all channels for a given pixel. Since a pixel coordinate only accepts the integer format, the flow field with shape $(2 \times h \times w)$ is necessary for handling the pixel transformation, which allows the flow field to transform a pixel value to a location along the vertical and horizontal directions, respectively, even though it does not lie in the integer grid. To ensure that f is differentiable during training, bi-linear interpolation (Jaderberg et al., 2015) is used to compute an appropriate pixel value over the current neighbourhood for the transformed image x_{st}:

$$x_{st}^{(i)} = \sum_{q \in N\left(u^{(i)}, v^{(i)}\right)} x^{(q)} \left(1 - |u^{(i)} - u^{(q)}|\right)\left(1 - |v^{(i)} - v^{(q)}|\right) \tag{3}$$

Here, the neighbourhood $N\left(u^{(i)}, v^{(i)}\right)$ is the four defined positions of (top left, top right, bottom right, bottom left) that tightly surround the target pixel x^i. In this way, the spatially perturbed image remains in the same shape as the original image. To encourage the flow field f to generate images with high perceptual quality, (Xiao et al., 2018b) introduced the flow loss based on the total variation (Rudin et al., 1992) to enforce local smoothness with respect to the neighbourhood $\mathbb{N}(p)$ of each pixel p:

$$\mathcal{L}_{flow}(f) = \sum_{p}^{\text{all pixels}} \sum_{q \in \mathbb{N}(p)} \sqrt{\|\Delta u^{(p)} - \Delta u^{(q)}\|_2^2 + \|\Delta v^{(p)} - \Delta v^{(q)}\|_2^2} \tag{4}$$

This flow loss is included in the objective function, together with a fooling loss introduced in (Carlini and Wagner, 2017), which will be minimised during training. However, the hyper-parameter for balancing these two losses in each dataset may be different, thus it becomes unclear when measuring the magnitude of the spatial distortion.

When applying bi-linear interpolation, the new value of x_{st} depends on the direction and magnitude toward which the original x changes. As we want the perturbation to be as imperceptible as possible, intuitively, we can constrain that every single pixel can only move its mass to nearby neighbours. Different from (Xiao et al., 2018b), here we introduce a hyper-parameter τ to budget the perturbations caused by the spatial perturbation, which satisfies the constraint: $\hat{L}_{flow}(f) \leq \tau$, where $\hat{L}_{flow}(f)$ is defined as:

$$\hat{L}_{flow}(f) = \max_{q_j \in \mathbb{N}(p)} \sqrt{\frac{1}{n} \sum_{p}^{n} \left(\|\Delta u^{(p)} - \Delta u^{(q_j)}\|_2^2 + \|\Delta v^{(p)} - \Delta v^{(q_j)}\|_2^2\right)} \tag{5}$$

Here n is the number of pixels for the corresponding image, and $\mathbb{N}$ is the function of the defined Von Neumann neighbourhood (Toffoli and Margolus, 1987), which defines the notion of 4-connected pixels that have a Manhattan distance of 1. This constraint can be further derived as follows:

$$\|\Delta u^{(p)} - \Delta u^{(q)}\|_2^2 + \|\Delta v^{(p)} - \Delta v^{(q)}\|_2^2 \leq \tau^2 \tag{6}$$

Intuitively speaking, for each pixel, this implies that the explicit τ budgets the mass it can move along the horizontal and vertical directions. This notion is a critical component for quantifying the intensity of the spatial transformation, rather than optimising a loss with an arbitrary value in the objective function. If we assign the value of τ as 0.1, and consider an extreme case where one of the terms on the left side of (6) is zero, it will merely move toward one direction (horizontal or vertical) with 10% of the pixel mass by one pixel. Intuitively speaking, this can also be understood as moving less than 10% of the pixel mass along two orientations for more than one pixel.

This concept can be approximately considered as the notion of the Wasserstein distance (Wong et al., 2019b), which projects the generated noise onto a Wasserstein ball by employing a low-cost transport plan. We also restrict the spatial transformation in a local configuration, which only moves the adjacent pixel's mass to the nearby pixels. The

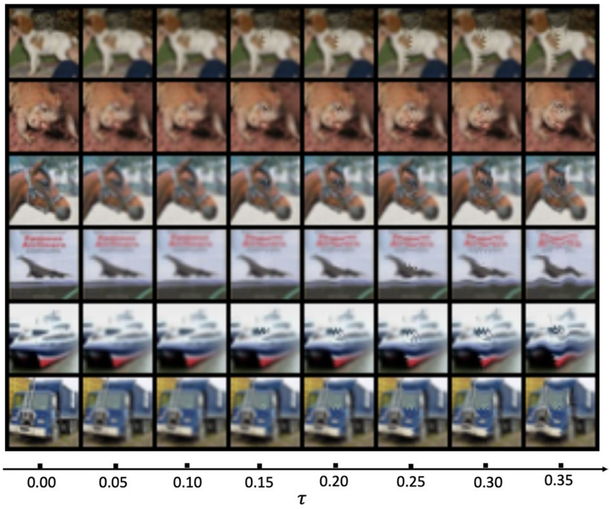


Fig. 2 Universal spatially transformed adversarial images with the increasing value of τ

key difference is that (Wong et al., 2019b) utilised the sinkhorn iterations for calculating the projected Wasserstein distance based on additive noise, while in our method, this is achieved via interpolating from its neighbourhood without extra additive perturbation, which is more natural for image manipulations and leads to a quasi-imperceptible effect for the human eye.

Note that different images may have different sensitivity to spatial perturbation. For example, as shown in Fig. 2, the deformations in the first three rows are almost indistinguishable for humans even with large τ, however, in the last three rows, the distortion is perceptible when τ is greater than 0.1. For images that have a regular structure, such as a straight line, their distortion leads to a curve shape and becomes noticeable to human eyes. In natural creation, most things are not in a straight-line shape, which makes this spatial perturbation useful in the physical world. In empirical, the setting $\tau = 0.1$ with indistinguishable visual quality is adapted for our experiments in Sects. 4-5, where we will show how it can help for the universal adversarial attack.

3.3 Generalized adversarial perturbation generator

In our unified framework, there are two key components to produce generalised universal adversarial perturbations: *i)* universal spatial perturbation; *ii)* universal noise perturbations. If we set τ to 0, the spatial transformation will turn to identity transformation, and our framework is reduced as a universal attack method to generate additive perturbation only. On the other hand, when ϵ is set to 0, our framework results in learning spatially transformed universal perturbations. Most importantly, our framework enables it to work collaboratively and generalise universal adversarial attacks by considering both ℓ_∞ norm-based (additive) and spatial transformation-based (non-additive) perturbations together, or any combination of both.

Taking advantage of the generative architectures described in (Hayes and Danezis, 2018), Poursaeed et al. (2018), (Reddy Mopuri et al., 2018), we employ the generative model to find a small universal noise, rather than the iterative approach in (Moosavi-Dezfooli et al., 2017). Differently, we also capture the distribution of the universal spatial perturbation at the same

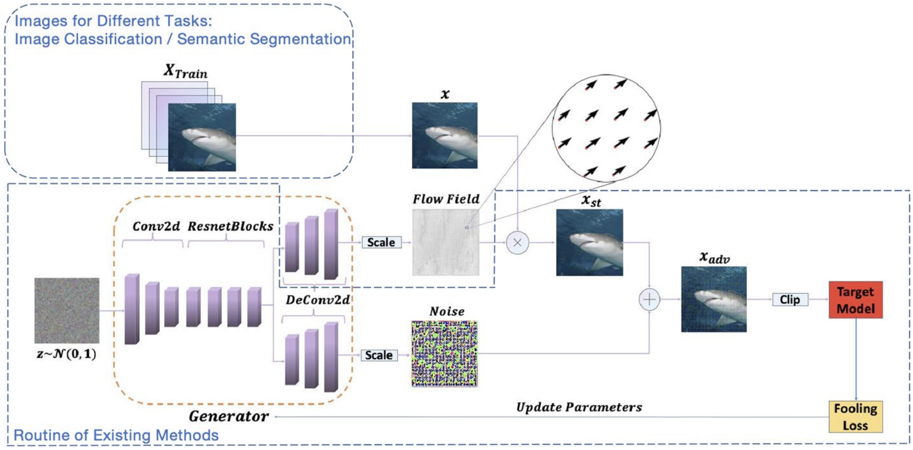


Fig. 3 Overview of Generalized Universal Adversarial Perturbation, here ⊗ represents the spatial transformation operation, ⊕ is the additive implementation. It is noted that in our framework, the target model (red box) can be an image DNN classifier or a semantic segmentation DNN model

time. By doing so, the learned perturbations will not directly depend on any input image from the dataset, so-called universal perturbations. Here, we show how to learn these two universal perturbations jointly via an end-to-end generative model. We adopt a similar architecture as image-to-image translation adversarial networks (Zhu et al., 2017; Isola et al., 2017; Xiao et al., 2018a) for perturbation generation, which utilises an encoder-bottleneck-decoder structure to transfer an input vector to the desired outputs. Fig. 3 indicates the whole workflow for the generation of universal perturbations. It can be seen that in the generator, the encoder consists of three convolutional layers followed by instance normalisation and ReLU. After going through four ResNet blocks, there are two decoders producing two outputs for learning different universal perturbations, respectively, each of them contains 3 deconvolution layers, followed by instance normalisation and ReLU activation functions again. The largest difference from other generative models for universal learning (Hayes and Danezis, 2018; Poursaeed et al., 2018; Reddy Mopuri et al., 2018) is that we only take a single input noise vector as input, i.e., batch-size equals 1; in the meanwhile, we apply instance normalisation during training. The output of the generator consists of two parts: a universal flow field f, and a small universal perturbation noise δ, so we apply another decoder to craft spatial perturbations. In addition, we eschew the discriminator in generative adversarial networks (GAN) (Goodfellow et al., 2014a) because the natural-looking attribute has already been controlled by ϵ and τ, respectively.

Formally, for our generalised adversarial generator $\mathcal{G}_\theta(z)$ parameterized by θ, it is fed a fixed random vector $z \sim \mathcal{N}(0,1)$ and outputs a flow field $f_0 \in [-1,1]^{2\times h\times w}$ and a noise $\delta_0 \in [-1,1]^{c\times h\times w}$, activated by two *Tanh* functions, respectively. Then the output f_0 is scaled to obtain the universal flow field f. This operation ensures that the prerequisite constraint for spatial distortion is met, controlled by the parameter τ:

$$f = \frac{\tau}{\hat{L}_{flow}(f_0)} \cdot f_0 \tag{7}$$

which is then applied to any $x \in \mathcal{X}$ to perform spatial distortion. On the other hand, in terms of the output noise, we also scale it to satisfy the ℓ_∞ constraint, i.e., assigning $\epsilon = 0.03$ for colour image values ranging from [0, 1]:

$$\delta = \frac{\epsilon}{\|\delta_0\|_\infty} \cdot \delta_0 \tag{8}$$

In the next step, we combine it with the spatially perturbed image generated by the learned flow field f, then the clipping implementation is carried out to guarantee that each pixel in the image has a valid value in [0, 1]. The final adversarial example can be expressed as follows:

$$x_{adv} = \mathcal{F}_f(x) + \delta \tag{9}$$

In this way, we can define a loss function that leads to misclassification and update the parameters of the generator accordingly.

3.4 Objective function

Given an input x, the corresponding adversarial example x_{adv} aims to fool the target model h with high probability. We denote the original prediction of x as $y := \arg\max h(x)$, then the objective function for an untargeted generalised perturbation attack attempts to find the universal perturbation that misleads the original prediction to a wrong class. Here, we define it as:

$$L_{adv}(x_{adv}, y) = -\hat{l}_{ce}(h\big(x_{adv}\big), y) = -\hat{l}_{ce}(h\big(\mathcal{F}_f(x) + \delta\big), y) \tag{10}$$

where $\hat{l}_{ce}$ is the surrogate loss function of the conventional cross-entropy l_{ce} which computes the cost between the output logit p of model and the given label y:

$$\hat{l}_{ce}(p, y) = \frac{1}{N} \sum_{i=1}^{N} \log(l_{ce}(p, y) + 1) \tag{11}$$

Since there is no upper bound for traditional cross-entropy loss, a single evaluating data point is potentially capable of causing an arbitrarily low loss value from 0 to ∞. The worst case happens when there is a single image turning into a perfect adversarial example, this causes misclassification, but it dominates the cross-entropy loss and forces the average loss to infinity. There is no doubt that this raises the difficulty of finding the optimal parameter and leads to slow convergence. To tackle this problem, we propose the scaled cross-entropy above to enforce our optimiser to search for the perturbation that targets at as many data points as possible. The '+1' operation ensures that the output of the log function remains positive. On the other hand, the natural log function scales the original loss for each image. This avoids any single image from standing over the objective during the optimization. We intensively evaluate the effectiveness of this scaled loss function in Sect. 7.

4 Experiments of image classification

Extensive experiments are conducted on two benchmark image datasets to evaluate the performance of the proposed framework, i.e., CIFAR-10 (Krizhevsky et al., 2009) and ImageNet (Deng et al., 2009). The code is created with PyTorch library, all experiments are run on one or two GeForce RTX 2080Ti GPUs. Unless otherwise specified, in the following experiments, we conduct our GUAP training using Adam optimiser with weight decay 1×10^{-4} for 20 epochs, the learning rate is set to 0.003 with a batch size of 100 on CIFAR-10 dataset. For ImageNet dataset, the learning rate is set to 0.01 with a batch size of 32 due to the large image size.

To evaluate the proposed method, here the attack success rate (ASR) is used to measure the performance of the attack, which reflects the percentage of the images in a test set that can be used by the adversary to successfully fool the victim neural networks. A higher ASR represents the better attacking capacity of the adversary and the higher vulnerability of the target neural networks. Since the proposed method can have several different setups controlled by two parameters, i.e., ϵ and τ. We refer the configuration ($\epsilon = 0.04, \tau = 0$) as GUAP_v1, which only performs the universal attack under the traditional ℓ_∞ norm. Regarding the combination attack, for convenience we refer the setup, GUAP_v2: $\epsilon = 0.03, \tau = 0.1$; and GUAP_v3: $\epsilon = 0.04, \tau = 0.1$. A small value of τ ensures that the caused spatial distortion is imperceptible to humans, while we show that, combining with small ℓ_∞-bounded perturbation can achieve state-of-the-art results on both small and large benchmark datasets.[1]

4.1 Universal attack on CIFAR-10 dataset

CIFAR-10 dataset contains 60,000 colour images from 10 different classes, with 6,000 images per class. Each image has 32×32 pixels. Normally, they are split into 50,000 images for training purposes and 10000 images used for evaluation. For comparison, we follow (Hayes and Danezis, 2018) and use the same neural network structures as the target classifiers for generating universal adversarial perturbations, i.e., VGG19, ResNet101, and DenseNet121. The standard accuracy of these models is 93.33%, 94.00%, and 94.79%, respectively.

Table 2 reports the experimental results achieved by our methods and the comparison with the other six universal attacks, including UAP (Moosavi-Dezfooli et al., 2017), Fast Feature Fool (Mopuri et al., 2017), and UAN (Hayes and Danezis, 2018), GAP (Poursaeed et al., 2018), UAT (Shafahi et al., 2020) and Cosine-UAP (Zhang et al., 2021a). In particular, when τ is set to 0, our framework degrades to craft universal noise constrained by ℓ_∞ norm, i.e., GUAP_v1, but it still obviously outperforms most universal adversarial attacks in terms of ASR. We find that VGG19 is the most resistant model to universal attacks, which is in line with the observation in (Hayes and Danezis, 2018). For this challenging VGG19 model, our method can still achieve 84.25% ASR, with a nearly 17% improvement over the strongest universal attacks generated by the generative models (GAP and UAN). When compared to state-of-the-art universal adversarial attacks (UAT and Cosine-UAP), GUAP_v1 is able to achieve comparable results, and GUAP_v2 and GUAP_v3 even boost the performances further, this also indicates the effectiveness of the proposed method.

On the other hand, when ϵ equals 0, our framework turns to universal spatial perturbation. Here, we conduct an ablation study to investigate the relationship between the spatial perturbation and the ℓ_∞-bounded perturbation. The visualisation in Fig. 4 demonstrates the corresponding attack success rates over the test set of the CIFAR-10, trained for the target VGG19 model. Different heat cap colours represent the performance of the adversarial

[1] Our code can be found in https://github.com/TrustAI/GUAP.

Table 2 Comparison with the state-of-the-art universal attack methods on CIFAR-10 dataset

Universal attack	Configuration		Attack success rate		
	ϵ	τ	VGG19	ResNet101	DenseNet121
UAP	0.04	–	57.20%	76.00%	67.90%
FFF	0.04	–	20.10%	36.50%	34.10%
UAN	0.04	–	66.60%	85.10%	75.00%
GAP	0.04	–	67.35%	74.75%	74.94%
UAT	0.04	–	84.63%	88.59%	87.49%
Cosine-UAP	0.04	–	84.26%	86.89%	88.75%
GUAP_v1	0.04	0.0	84.25%	88.58%	89.23%
GUAP_v2	0.03	0.1	86.86%	89.45%	90.09%
GUAP_v3	0.04	0.1	89.59%	89.56%	90.09%

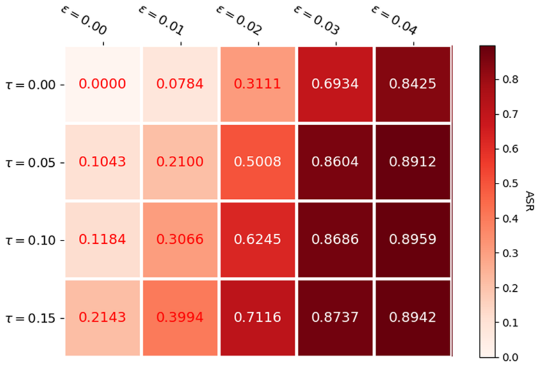


Fig. 4 Attack success ratio for VGG19 under different combination settings on CIFAR-10 dataset

attack with different $\epsilon \in \{0.0, 0.01, 0.02, 0.03, 0.04\}$ and $\tau \in \{0.0, 0.05, 0.1, 0.15\}$. From the x-axis, it can be observed that with the increasing magnitude of ϵ, a higher fooling rate can be achieved. Similarly, towards the y-axis direction, a larger τ also leads to a higher attack success rate.

Since the constrained ball for spatial perturbation is different from the ℓ_∞ norm ball, it can be inferred that these two universal perturbations are not strictly contradictory to each other, and hence performing them together brings improvements in terms of ASR but will not compromise the imperceptibility. As expected, the proposed combination attack GUAP_v2 is able to fool the target classifier with a very high ASR. And our attack, GUAP_v3, obtains the highest ASR rate among all attack methods with a nearly 5% improvement to the state-of-the-art methods on VGG19 DNN. Fig. 5 displays several adversarial examples generated by GUAP_v2 among ten different classes. The second and fourth rows represent the differences between the original image (first row) and perturbed images (third and fifth rows) caused by the spatial transform and additive noise, respectively. We can easily observe that the spatial-based attack mainly focuses on the edge of images, which confirms that the edge of the image plays a significant role in deep neural networks.

In addition, as (Reddy Mopuri et al., 2018) suggested, visual diversity plays a significant effect in the effective exploration of the latent space. As shown in Fig. 6, we also draw some universal perturbations generated on CIFAR-10 dataset by GUAP_v1 when using the different random seeds for input noise initialisation, it can see that our perturbations

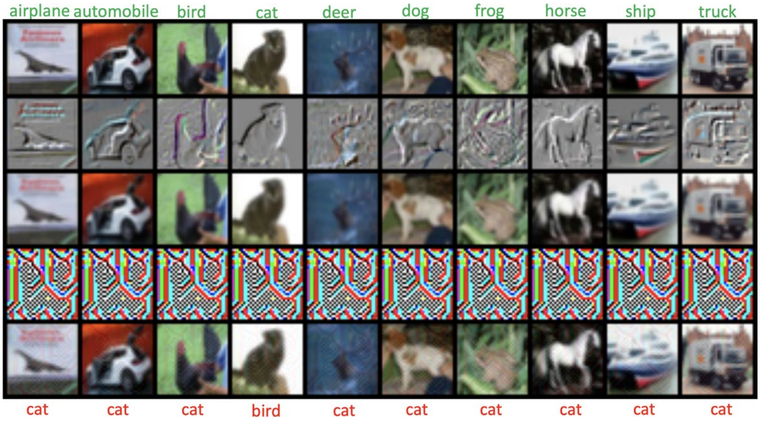


Fig. 5 Attacking performance of GUAP_v2 on CIFAR-10 dataset among 10 classes against VGG19

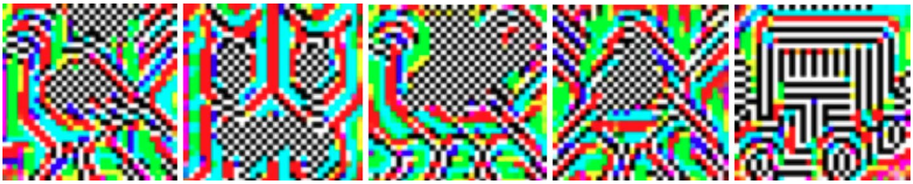


Fig. 6 Different Universal additive perturbations using different input noises generated by GUAP_v1 against VGG16 on CIFAR-10 dataset

present a higher level of diversities compared to NAG. This also confirms that the proposed method indeed has a strong capacity to capture the distribution of perturbations. Last but not least, we can see that actually each universal perturbation at least resembles one specific category in the dataset. This indicates that when finding the universal perturbation, it will move towards a specific class and cross the decision boundary, which is more promising to achieve better fooling performance.

4.2 Universal attack on ImageNet dataset

We also try to perform universal attacks on a large image dataset. Following the instruction in (Moosavi-Dezfooli et al., 2017), a subset of the training set of the ILSVRC 2012 dataset (Deng et al., 2009) is utilized as our training dataset, which contains 10,000 images from 1000 classes, i.e., 10 images per object. In addition, 50,000 images in the validation set of ImageNet are treated as the test set for evaluation. We adopt four different neural networks as target models: VGG16 (Simonyan and Zisserman, 2014), VGG19 (Simonyan and Zisserman, 2014), ResNet152 (He et al., 2016), and GoogLeNet (Szegedy et al., 2015), whose top-1 accuracy can reach 71.59%, 72.38%, 78.31% and 69.78%, respectively. We compare GUAP with six baseline methods, including UAP (Moosavi-Dezfooli et al., 2017), Fast Feature Fool (Mopuri et al., 2017), two generative model-based methods including NAG (Reddy Mopuri et al., 2018), UAN (Hayes and Danezis, 2018) and GAP (Poursaeed

Table 3 Comparison with the state-of-the-art universal attack methods on imagenet dataset

Universal attack	Configuration		Attack success rate			
	ϵ	τ	VGG16	VGG19	ResNet152	GoogleNet
UAP	0.04	–	78.30%	77.80%	84.00%	78.90%
FFF	0.04	–	47.10%	43.62%	–	56.44%
NAG	0.04	–	77.57%	83.78%	87.24%	90.37%
GAP	0.04	–	83.70%	80.10%	–	82.70%
UAT	0.04	–	94.80%	96.06%	92.08%	90.91%
Cosine-UAP	0.04	–	97.40%	96.40%	90.20%	90.50%
GUAP_v1	0.04	0.0	97.06%	94.80%	95.15%	90.60%
GUAP_v2	0.03	0.1	98.25%	97.00%	96.74%	94.42%
GUAP_v3	0.04	0.1	98.47%	99.24%	99.03%	97.82%

et al., 2018), and two state-of-the-art methods UAT (Shafahi et al., 2020) and Cosine-UAP (Zhang et al., 2021a) for crafting universal perturbation.

Table 3 reports the experimental results on this dataset. We can see that our proposed model GUAP_v1 is able to obtain better fooling rates under the same magnitude of ℓ_∞ norm, i.e., over 90% for all target models. This proves the fragility of deep neural networks even the perturbation is universal. These results are comparable to or exceed the state-of-the-art methods for generating universal perturbations. UAP (Moosavi-Dezfooli et al., 2017) suggests that VGG19 is the most resilient DNN for ImageNet dataset. However, we find out that GoogLeNet is the least sensitive model to additive perturbation compared to other models, this is in line with the result from the state-of-the-art methods UAT (Shafahi et al., 2020) and Cosine-UAP (Zhang et al., 2021a).

The ablation study in Fig. 7 demonstrates the interplay between spatial perturbation and additive L_p-norm perturbation. When the universal ℓ_∞ attack is integrated with spatial perturbation, a stronger attack can be achieved. In particular, our combination attack, GUAP_v3, obtains the highest attack success rate compared to other universal ℓ_∞ attacks with a larger ϵ, including the proposed GUAP_v1. The experiment demonstrates the superior performance of our universal attack when it contains multiple types of perturbations, e.g., a combination of both spatial and additive perturbations. we also visualise the learned perturbation on ImageNet for four different target models in Fig. 8. We can see that perturbations from GUAP present a high level of diversities across different models.

In Sect. 7, we will investigate more properties of the proposed method, especially when spatial perturbations are involved. Specifically, we study the imperceptibility of the adversarial perturbations, the effect of training samples, and the transferability of our method – GUAP.

5 Experiments of semantic segmentation

5.1 Universal attack on cityscapes dataset

In this section, we also generalise the proposed method for semantic segmentation tasks. Different from image classification, semantic image segmentation denotes dense prediction

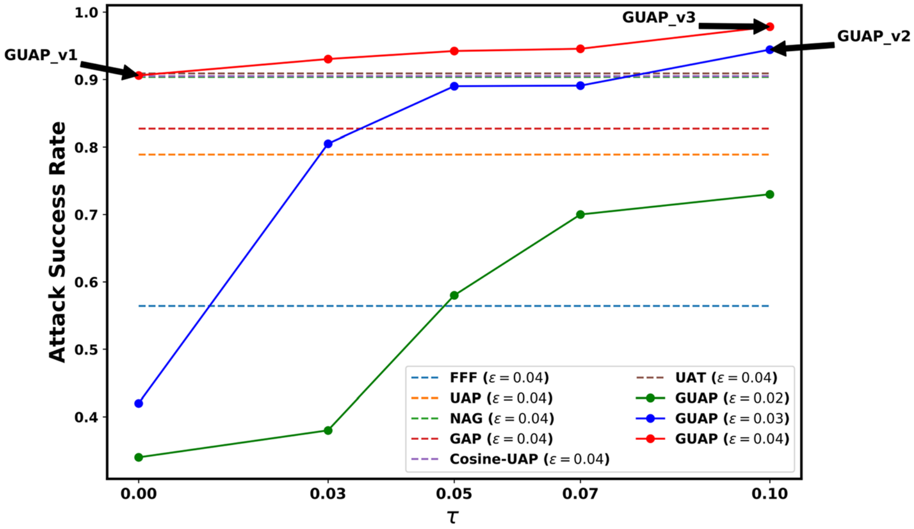


Fig. 7 Attack success rates against GoogLeNet under different combination settings on ImageNet dataset

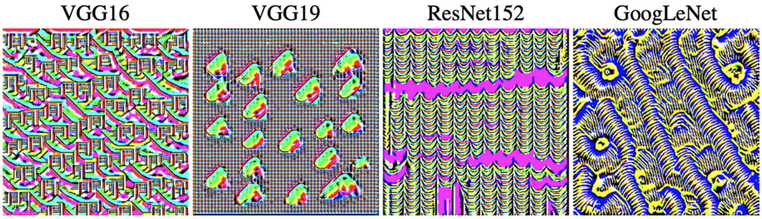


Fig. 8 Universal additive perturbations generated by GUAP_v1 on ImageNet dataset against VGG16, VGG19, ResNet152 and GoogLeNet, respectively

missions, which provide a class label (prediction) to each pixel of the image that answer the question: “what is where in an image?”. Regarding the aim of adversarial attack under the segmentation setting, we consider the static target segmentation used in (Hendrik Metzen et al., 2017). In this scenario, the adversary can define a fixed segmentation at a time step t_0 as a target for all subsequent time steps such that $y^t_{target} = y^t_{pred}$ $\forall t > t_0$. This can be used to attack a monitor system based on a static camera and the adversary can hide suspicious activities during a time span $t > t_0$. In our experiment, this case study is conducted on the Cityscapes dataset (Cordts et al., 2016) against the FCN-8s model (Long et al., 2015). This database consists of 2975 training images and 500 validation samples with size 2048× 1024. Following the settings in (Poursaeed et al., 2018), (Hendrik Metzen et al., 2017), we resize images and label maps into 1024×512 with bilinear and nearest-neighbour interpolation, respectively. Regarding the aim of adversarial attack, we consider the static target segmentation used in (Hendrik Metzen et al., 2017). Following the same setting, an arbitrary ground-truth segmentation (monchengladbach_000000_026602_gtFine) is chosen, as shown in the left part of Fig. 9. And the attack success rate on the validation set is used as the evaluation metric, which measures the pixel accuracy between static target segmentation and predicted segmentation of the network on the adversarial example, i.e.,

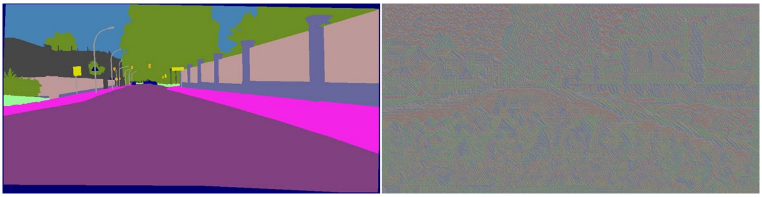


Fig. 9 Static Target label map and the universal additive perturbation generated by GUAP against the FCN-8s semantic segmentation model on Cityscapes dataset

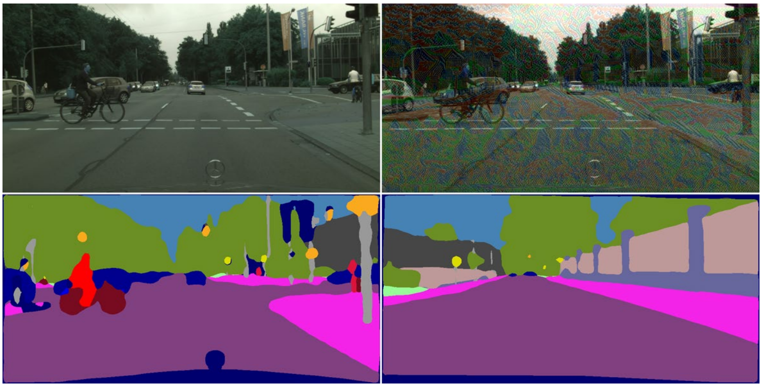


Fig. 10 Example of the targeted universal perturbations for GUAP_v1 against the FCN-8s semantic segmentation model on Cityscapes dataset

Table 4 Comparison with the state-of-the-art universal attack methods on cityscapes dataset

Universal attack method	τ	Configuration of ϵ		
		0.02	0.04	0.08
UAP-Seg	–	80.30%	91.00%	96.30%
GAP	–	79.50%	92.10%	97.20%
GUAP	0.0	80.04%	91.90%	96.96%
	0.1	82.24%	93.54%	97.42%

the percentage of pixels that are accurately classified in the image. Similar to the image classification task in Sec. 4, we implement the proposed GUAP to generate the universal spatial perturbation and universal ℓ_∞ noise on the training data with the defined surrogate loss function Eq. 10.

In Table 4, we report the success rate of the attack in the validation set as the evaluation metric, which measures the categorical accuracy between static target segmentation and the predicted segmentation of the network on the adversarial examples. Compared to two existing state-of-the-art approaches, i.e., GAP (Poursaeed et al., 2018) and UAP-Seg (Hendrik Metzen et al., 2017), GUAP achieves comparable results in fooling the targeted classifier. In particular, when performing the universal spatial perturbation, our method is able to yield better fooling rates. Given the static target label map in the left image in Fig. 9, GUAP generates the universal perturbation on the right, which resembles the target segmentation to a large extent. We can see an example demonstrated in Fig. 10, the first row shows the original image and the adversarial image, respectively; in the second row, the left and right figures plot the corresponding segmentation predictions, which will mislead the driver to hit the pedestrian dangerously.

6 Experiments of adversarial training

6.1 Universal adversarial training on CIFAR-10 dataset

In this section, we explore universal adversarial training against universal perturbations. As a defensive strategy, adversarial training can be formulated as the following Min-Max optimisation problem (Madry et al., 2017).

$$\min_{\theta} \mathbb{E}_{(x,y)\sim\mathcal{D}} \left(\max_{\delta \in B_{\epsilon}} L(F_{\theta}(x+\delta)) \right), \tag{12}$$

The inner maximisation problem in 12 can be approximated by adversarial attacks, such as FGSM (Goodfellow et al., 2014b) and PGD (Madry et al., 2017).

Some existing work also explores adversarial training in a universal manner. (Mummadi et al., 2019) proposed the shared adversarial training to utilise the shared gradients of the image heap. (Shafahi et al., 2020) utilised FGSM-based UAP with a simple application in adversarial training, and it requires many iteration steps.

Differently, we leverage the generative model to generate the universal perturbation for the inner maximisation to obtain a resistant model. We show that it is possible to utilise the fast adversarial setting (Wong et al., 2019a) with cycle loss to efficiently train a model robust to universal attack. Similarly to (Shafahi et al., 2020), we generate the (batch) universal perturbation and add it to the clean images during the training process.

For each attack method, we use 5,000 images (only use 1/10 data in order to reduce the training time) from CIFAR-10 dataset to construct the universal perturbation to attack four Wide-ResNet (WRN-34-10) models with different robustness, i.e., the clean trained, adversarially trained with PGD-7 (Madry et al., 2017), GUAP_v1 and GUAP_v2. Note that here the ϵ for ℓ_∞ norm attack is set to 8/255 for GUAP_v1 and GUAP_v2 (with $\tau = 0.1$). As shown in Table 5, we report the clean accuracy and robust accuracy on CIFAR-10 test dataset, the columns represent the (adversarially) trained models, and each row illustrates the robust accuracy under various universal attack approaches to evaluate the robustness.

We can see that adversarially trained models with GUAP are highly resistant to universal attacks (Moosavi-Dezfooli et al., 2017; Poursaeed et al., 2018), where the robust accuracy has just dropped a little compared to the clean accuracy. And in most cases, the attacking capacities are mostly in line with the observation in Sect. 4.1. An interesting observation is that when adversarially training the model with GUAP_v2, the attacking performance of GUAP_v1 is better than GUAP_v2. And it is difficult for the joint optimisation in GUAP_v2 to coverage to a global minimum because the adversarial examples generated by GUAP_v2 have already been considered during the training process. In this case, the pure ℓ_∞-bounded attack GUAP_v1 is able to focus on optimisation of additive noise and obtain slightly lower robust accuracy. Furthermore, it can be seen that universal adversarially trained models can also achieve some robustness when defending strong instance (local) adversarial attacks such as PGD-7 and PGD-20 (Madry et al., 2017). For

Table 5 Evaluation on CIFAR-10 dataset with Wide-ResNet34-10 model under (Universal) adversarial training

Attack	Adversarially trained with			
	Natural	PGD-7	GUAP_v1	GUAP_v2
Clean	95.46%	86.07%	92.16%	90.17%
UAP	35.65%	85.83%	91.78%	89.61%
GAP	35.69%	85.69%	90.24%	89.30%
GUAP_v1	18.87%	85.43%	90.88%	85.40%
GUAP_v2	10.90%	81.06%	83.18%	87.01%
FGSM	13.65%	68.41%	40.33%	44.56%
PGD-7	00.00%	58.41%	16.79%	21.94%
PGD-20	00.00%	55.87%	13.01%	16.50%
Training speed (seconds/epoch)	89.1	670.6	245.4	245.8

weak instance (local) adversarial attack such as FGSM, universal adversarially-trained models are able to achieve at least 40% robust accuracy.

7 More properties of GUAP

In this part, we further investigate various properties of GUAP, especially when the spatial perturbation is involved.

7.1 Imperceptibility of adversarial perturbations

When the spatial transform is involved for adversarial attacks, even it is invisible for humans, and the learned perturbations are not strictly in the ℓ_∞ norm ball. In other words, for the proposed GUAP, the constrained ball for combination attack is beyond that of the additive ℓ_∞ perturbations. Here, for fairness, we employ two other distance metrics, to measure the imperceptibility between the original and adversarial images, i.e., SSIM (Wang et al., 2004) and LPIPS (Zhang et al., 2018). As shown in Fig. 11, when achieving a similar attack success rate, applying the combination attack can achieve more imperceptibility most of the time, i.e., higher SSIM scores and smaller LPIPS distances, compared to the universal perturbation that uses the ℓ_∞-norm noise only.

It is observed that under a similar ASR, the similarity achieved by the combination attack is better than using a pure ℓ_∞-bounded attack with large ϵ. Since spatial transformation does not modify the pixel directly, it leads to more natural perturbations for human beings. Thus, by combining spatially transformed perturbation with universal ℓ_∞ attack together, the proposed GUAP is able to achieve a comparable or even better attack performance while having a smaller LPIPS distance and high SSIM score, compared to only using the additive ℓ_∞-norm perturbation that usually requires larger ϵ. Some adversarial examples can be seen in Fig. 12, we can see that the adversarial examples generated by GUAP_v2 and GUAP_v3 still maintain high imperceptibility to human beings. In other words, a spatial perturbation will significantly benefit crafting strong universal adversarial examples with better visual imperceptibility, and our method provides a flexible framework to achieve this purpose.

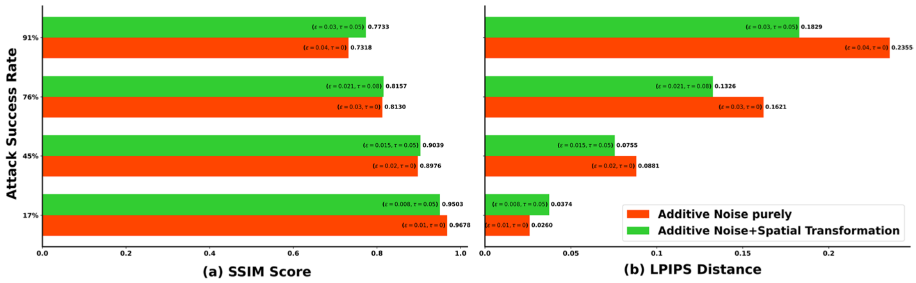


Fig. 11 **a** Average SSIM score between original images and adversarial examples for GoogLeNet on ImageNet dataset; **b** Average LIPIS distance between original images and adversarial examples for GoogLeNet on ImageNet dataset

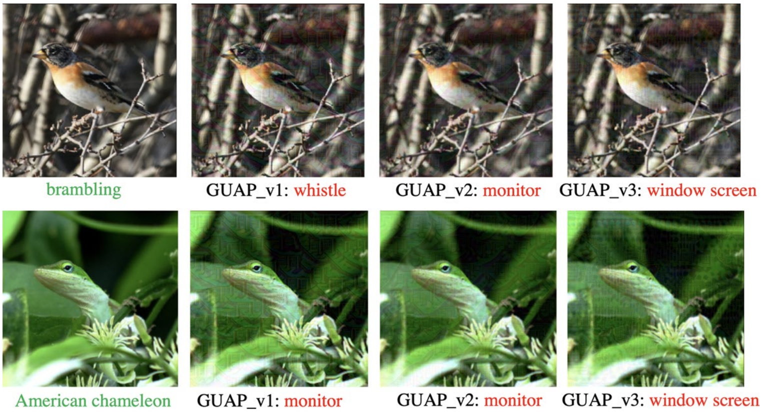


Fig. 12 Set of benign and corresponding adversarial examples against VGG16 on ImageNet dataset

7.2 How many training samples are needed?

There is no doubt that the number of training samples plays a crucial role not only in classification, but also in adversarial attacks. In this section, our aim is to investigate the impact of the amount of training data when the spatial perturbation is taken into account to generate universal perturbations. Here, we pick up the most robust model GoogLeNet and probe the impact of universal spatial perturbation on the attack strength. We use four sets of training data with different sizes, and generate adversarial examples on GoogLeNet, successively. For each setup, the proposed GUAP_v1, GUAP_v2 and GUAP_v3 are conducted to search for the universal perturbations, respectively, and then we report the attack success rate with respect to the prediction of the target model on the whole validation set (50,000 images).

As shown in Fig. 13, it can be observed that, by increasing the number of training samples, the fooling rate can be improved. Surprisingly, the combined attack method GUAP_v2 shows a powerful capacity. Although only 500 images are used for crafting adversarial examples, GUAP_v2 can still fool more than 60% of the images in the validation set, which is two times more than the iterative UAP method (Moosavi-Dezfooli et al., 2017). This demonstrates that our method has a more remarkable generalisation power over unseen

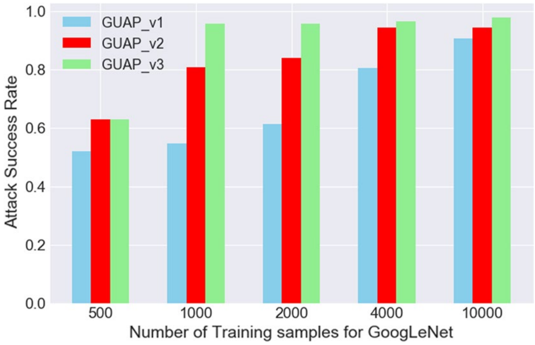


Fig. 13 Attack success ratio on the validation set versus the amount of training samples

data. Compared to the universal ℓ_∞-bounded attack with 4,000 training data, our combined attack method GUAP_v2 can use fewer images (i.e., 1,000 images) but achieve a similar attack success rate (i.e., 80%). In particular, only one image per class is sufficient for GUAP_v3 to achieve more than 95% fooling rate. In other words, universal attacks that contain both spatial and additive perturbations, such as GUAP_v2 and GUAP_v3, have a superior capacity compared to the pure universal ℓ_∞-bounded attack, i.e., GUAP_v1, especially when only limited training samples are available.

7.3 Transferability of GUAP

We further explore the generalisation ability of the learned UAPs across different models. We create 10,000 adversarial examples over the test dataset by our proposed GUAP_v1 and GUAP_v2 method, then feed them to a target classifier that is not used to learn universal adversarial examples. As shown in Table 6, UAP and NAG have better transferability from VGG16 and VGG19 to ResNet152 and GoogleNet. For both methods, the universal noise learned by using a less complex structure such as VGG16 as a source model can bring a better generalisation to other unseen models. Interestingly, different from UAP and NAG, the results of our proposed methods are just the opposite of UAP and NAG, the proposed GUAP approaches have a stronger connection with more complex models like ResNet152 and GoogleNet. In particular, the learned UAP based on GoogLeNet by GUAP_v2 is able to mislead all other models with more than 82% fooling rate, which is even better than that obtained by using UAP under a white-box attack setting. Surprisingly, the learned universal perturbation by GUAP_v1 and GUAP_v2 can achieve the fooling rate for these four different victim models up to 83.57% and 90.95% respectively, on average. However, when using VGG16 and VGG19 as source models, the proposed model seems to overfit the source model, which makes them not transferable to other complex models. In addition, because of spatial transformation, this phenomenon become more severe. This also reveals that, when the proposed method is used to construct a universal perturbation, the average transferability of learned adversarial noise becomes stronger when a more complex structure is employed as the source model.

7.4 Effect of the surrogate loss function

In this part, we investigate the effect of the proposed scaled loss function by comparing it with the performance of the original cross-entropy method on CIFAR-10 dataset.

Table 6 Transferability study cross different models, universal perturbations are constructed using source models and tested against pre-trained victim classifiers

Source model	Method	Victim model				Average
		VGG16	VGG19	ResNet152	GoogleNet	
VGG16	UAP	78.30%	73.10%	63.40%	56.50%	67.83%
	NAG	77.57%	73.25%	54.38%	67.38%	68.15%
	GUAP_v1	97.06%	90.96%	35.35%	52.37%	68.94%
	GUAP_v2	98.25%	93.95%	34.75%	41.59%	67.14%
VGG19	UAP	73.50%	77.80%	58.0%	53.6%	65.73%
	NAG	80.56%	83.78%	65.43%	74.48%	76.06%
	GUAP_v1	89.91%	94.80%	27.30%	26.60%	59.66%
	GUAP_v2	90.09%	97.00%	31.97%	31.68%	62.69%
ResNet152	UAP	47.00%	45.50%	84.00%	50.50%	56.75%
	NAG	52.17%	53.18%	87.24%	62.33%	63.73%
	GUAP_v1	88.87%	85.79 %	95.15%	64.47%	83.57%
	GUAP_v2	92.62%	92.77%	96.74%	57.95%	85.02%
GoogLeNet	UAP	39.20%	39.80%	45.50%	78.90%	50.85%
	NAG	56.40%	59.14%	59.22%	90.37%	66.37%
	GUAP_v1	82.57%	81.41%	59.83%	90.60%	78.60%
	GUAP_v2	93.76%	93.04%	82.19%	94.42%	90.85%

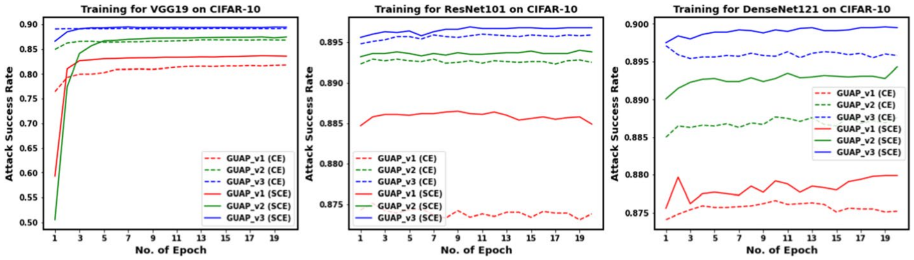


Fig. 14 Training process by maximising the original cross-entropy loss and the scaled cross-entropy loss under different setups

As shown in Fig. 14, it demonstrates the whole training process by applying the original cross-entropy loss (CE) and the scaled cross-entropy loss (SCE), respectively. We can see the results of the surrogate loss function always outperforms that of the traditional cross-entropy loss steadily, especially on the combination attacks.

7.5 Efficiency of GUAP

In terms of the efficiency of the proposed GUAP, The proposed method falls into the setting of a semi-white attack (Xiao et al., 2018a), which can synthesise adversarial images without accessing the structures and parameters of the original victim model. In addition, with the universal and input-agnostic properties, the produced perturbation can be

Table 7 Run time comparison when performing adversarial attack(s) on the CIFAR-10 test dataset

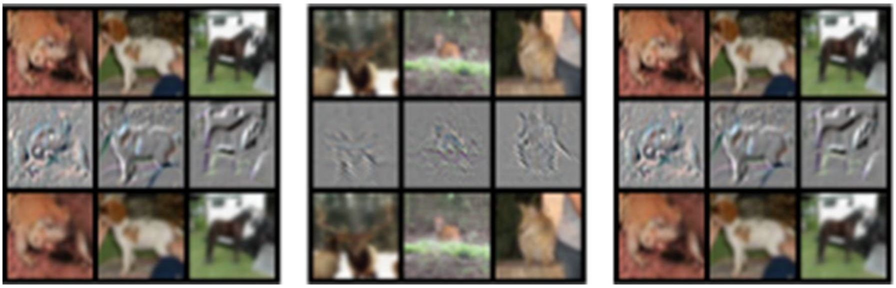
Method	Adversary requirement		Total run time on CIFAR-10 test dataset		
	Access to victim model	Access to extra generative model	VGG19	DenseNet121	ResNet101
FGSM	✓	–	4.31s	23.31s	30.66s
GAP	✗	✓	5.88s	9.13s	11.48s
GUAP (Ours)	✗	✗	2.33s	6.10s	7.27s

used directly in the attacking phase without any further computation. Specifically, after the generator has been trained, the universal flow field and universal noise learned from the training set can be directly employed for evaluating the performance of the universal attack, without accessing the generator again. These properties will be beneficial for the offline system in practice. Table 7 reports the total run time when performing the (universal) adversarial example(s) on the test dataset of CIFAR-10. Compared with the single-step white-box attack FGSM and another generative model-based method GAP, our proposed GUAP approach illustrates better efficiency, which requires less time to evaluate the robustness against universal adversarial attack during test time.

8 Conclusion

In conclusion, we propose a unified framework for crafting universal adversarial perturbations, which can be either ℓ_∞-norm (additive), spatial transformation (non-additive), or a combination of both. We show that, by combining spatial transformation with a small universal ℓ_∞-norm attack, our approach is able to obtain state-of-the-art attack success rates for universal adversarial perturbations, significantly outperforming existing approaches. Moreover, compared to current universal attacks, our approach can obtain a higher fooling performance, but with *i)* less training data, *ii)* superior transferability of the attack with a complex source model, and *iii)* without compromising human imperceptibility to adversarial examples. Except for the experiments on image classification, we illustrate that the proposed framework can be easily extended to adversarial attacks on semantic segmentation tasks. Furthermore, we also investigate the robustness of the DNN model under the universal adversarial training strategy. We believe that this work provides an alternative but more powerful universal adversarial attack/defence solution, which marks a step forward to understand the distributional robustness of deep neural networks.

The proposed framework combines two different universal adversarial attacks with different perturbation constraints in sequence, since the spatial perturbation provides more semantic features compared to the additive adversarial noise, future work can be conducted to investigate how to explore the mutual adversarial subspace for different perturbations. Besides, more properties between local adversarial training and universal adversarial training can be explored in the future, to identify how to improve the model's robustness against local and universal (distributional) adversarial attacks simultaneously.



(a) with Von Neumann neighbourhood (b) with corner neighbourhood (c) with square neighbourhood

Fig. 15 **a** With Von Neumann neighbourhood calculation, Benign images, their adversarial examples, and their intermediate perturbations; **b** With corner neighbourhood calculation, Benign images, their adversarial examples, and their intermediate perturbations; **c** With square neighbourhood calculation, Benign images, their adversarial examples, and their intermediate perturbations

Appendix A: Neighbourhood section for spatial constraint

Regarding the neighbourhood selection for spatial transformation, in our preliminary experiments, we consider 3 kinds of neighbourhoods: (a) Von Neumann (top mid, bottom mid, mid left, mid right); (b) corner (top-left, top-right, bottom-right, bottom-left); (c) square (top-left, top-right, bottom-right, bottom-left, top-left, top-right, bottom-right, bottom-left). All the neighbourhoods can be used to calculate the budget tau, as shown in Fig. 15. Under similar attacking results, Von Neumann and square neighbourhood perform similarly as Von Neumann is contained in the square neighbourhood, while the result of corner neighbourhood has the pixelation effect, in which the distortion is more visibly perceptible and less smooth than other twos. Based on this observation, we choose to simply use the Von Neumann neighbourhood for the constraint calculation.

Appendix B: Ablation study for different generative model-based UAP

As we notice that there is still a gap between GUAP_v1 and other approaches employing the generative model, i.e., UAN, NAG, and GAP. Table 8 demonstrates the model structures among these methods. It can be observed that UAN and NAG directly map the input noise to the final perturbation using Deconvolutions and FC layers, while the proposed GUAP method has a similar encoder-bottleneck-decoder structure to the network of GAP. It is hard to theoretically compare which kind of architecture is better for generating universal perturbation, but empirically from the experimental results, we can see that the encoder-bottleneck-decoder model outperforms the model that uses deconvolution and fully connected layers only.

Therefore, we add an ablation study to compare the benefits brought by different components with GAP. Except for the different designs in architecture, we highlight other three factors which may affect the final performance, i.e., scaling factor, attacking objective and loss function. In the original GAP, after getting the output δ_0 from the generator, they map the output $\delta_0 \in [-1, 1]$ into [0, 1] by performing $\delta_0 = (\delta_0 + 1) * 0.5$, then multiply it by

Table 8 Comparison of four types of model architecture

UAN	NAG	GAP	GUAP_v1
Input	Input	Input	Input
Deconv+BN+ReLU	FC+BN+ReLU	Conv+BN+ReLU	Conv+IN+ReLU
Deconv+BN+ReLU	Deconv+BN+ReLU	Conv+BN+ReLU	Conv+IN+ReLU
Deconv+BN+ReLU	Deconv+BN+ReLU	Conv+BN+ReLU	Conv+IN+ReLU
Deconv+BN+ReLU	Deconv+BN+ReLU	ResnetBlock	ResnetBlock
Deconv+BN+ReLU	Deconv+BN+ReLU	ResnetBlock	ResnetBlock
FC+BN+ReLU	Deconv+BN+ReLU	ResnetBlock	ResnetBlock
FC+BN+ReLU	Deconv+BN+ReLU	ResnetBlock	ResnetBlock
FC	Deconv	ResnetBlock	Deconv+IN+ReLU
	Tanh	ResnetBlock	Deconv+IN+ReLU
		Deconv+BN+ReLU	Deconv
		Deconv+BN+ReLU	Tanh
		Conv	
		Tanh	

Table 9 Ablation study of generative model-based UAP against VGG19 on CIFAR-10 dataset

[1]Net 1	[2]Net 2	[3]SO 1	[4]SO 2	[5]Obj 1	[6]Obj 2	[7]Loss 1	[8]Loss 2	ASR
✓		✓		✓		✓		65.48%
	✓	✓		✓		✓		61.89%
$\underline{✓}$		$\underline{✓}$		$\underline{✓}$			$\underline{✓}$	$\underline{67.35\%}$
	✓	✓		✓			✓	62.42%
✓			✓	✓		✓		66.90%
	✓		✓	✓		✓		66.01%
✓			✓	✓			✓	77.04%
	✓		✓	✓			✓	68.42%
✓		✓			✓	✓		80.97%
	✓	✓			✓	✓		80.74%
✓		✓			✓		✓	81.05%
	✓	✓			✓		✓	81.83%
✓			✓		✓	✓		81.70%
	✓		✓		✓	✓		82.87%
✓			✓		✓		✓	83.09%
	$\overline{✓}$		$\overline{✓}$		$\overline{✓}$		$\overline{✓}$	$\overline{84.33\%}$

[1] Net 1: Network structure of GAP

[2] Net 2: network structure of GUAP_v1

[3] SO 1: Scaling operation 1: map to [0,1] and then multiply by $\min(1, \frac{\epsilon}{\|\delta_0\|_\infty})$, used in GAP

[4] SO 2: Scaling operation 2: multiply by $\frac{\epsilon}{\|\delta_0\|_\infty}$, used in GUAP_v1

[5] Obj 1: Attack using least likely class as target, used in GAP

[6] Obj 2: Untargeted attack, used in GUAP_v1

[7] Loss 1: $-\log(\frac{1}{N}\sum_{i=1}^{N} l_{ce}(h(x_{adv}), y))$, used in GAP

[8] Loss 2: $-\frac{1}{N}\sum_{i=1}^{N}\log(l_{ce}(h(x_{adv}), y) + 1)$, used in GUAP_v1 Row with underline represents the default setting and result of GAP

Row with overline represents the default setting and result of GUAP_v1

$\min(1, \frac{\epsilon}{\|\delta_0\|_\infty})$ to obtain the final δ. In addition, they perform the untargeted attack by setting the least likely class as the target and conducting a targeted attack with the log function of cross-entropy loss. In this ablation study, we generate universal adversarial perturbations against VGG19 under different settings on CIFAR-10 dataset. The results are shown in Table 9, we can observe that the main factor affecting the attacking performance is the attacking objective, and then the scaling operation follows. Our previous experiments in Sect. 4.1 have indicated that the universal perturbation will towards a specific class and cross the decision boundary, which is more promising to achieve better fooling performance, therefore assigning different classes for training the generative model will harden its difficulty to find a powerful universal perturbation. After fixing this, GAP can achieve comparable results, and our scaling operation and loss function indeed bring some improvements to the attacking performance. From Table 9, we observe that, under the same settings (scaling operation, attack objective, and loss function), they are able to achieve a similar attack success rate, hence the differences in architecture between GAP and GUAP (including the batch/instance normalisation, the final Conv/Deconv layer before Tanh, and the number of ResNetBlock) do not have a huge effect on the final results. However, the proposed method requires fewer layers and parameters for training, which demonstrates the effectiveness of GUAP. Overall, with a suitable training aim, proper noise projection, and robust objective function together, the generative model can achieve better performance, while how to design and prove a generative architecture that brings the best performance still remains an open question.

Author Contributions YZ contributed to the idea, algorithm, theoretical analysis, writing, and experiments. WR contributed to the idea, theoretical analysis, and writing. FW contributed to the experiments. XH contributed to the theoretical analysis. All authors read and approved the final manuscript.

Funding This work is supported by Partnership Resource Fund (PRF) on Towards the Accountable and Explainable Learning-enabled Autonomous Robotic Systems from UK EPSRC project on Offshore Robotics for Certification of Assets (ORCA) [EP/R026173/1], and the UK Dstl project on Test Coverage Metrics for Artificial Intelligence.

Data availability There are some open-resourced datasets are used in this work, including CIFAR-10 https://www.cs.toronto.edu/~kriz/cifar.html, ImageNet, https://www.image-net.org and Cityscapes https://www.cityscapes-dataset.com.

Code availability We make our code available on Github (https://github.com/TrustAI/GUAP) for the purpose of reproducibility.

Declarations

Conflict of interest The authors have no competing interests to declare that are relevant to the content of this article.

Ethical approval This work does not involve any human subjects or animals, so has no ethical concerns.

Consent to participate Not Applicable.

Consent for publication Not applicable.

References

Akhtar, N., Liu, J., Mian, A. (2018). Defense against universal adversarial perturbations. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3389–3398.

Carlini, N., & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In 2017 IEEE symposium on security and privacy (sp), IEEE, pp. 39–57.

Collobert, R., Weston, J., Bottou, L., et al. (2011). Natural language processing (almost) from scratch. *Journal of Machine Learning Research, 12*(ARTICLE), 2493–2537.

Cordts, M., Omran, M., Ramos, S., et al. (2016). The cityscapes dataset for semantic urban scene understanding. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3213–3223.

Deng, J., Dong, W., Socher, R., et al. (2009). Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, IEEE, pp. 248–255.

Engstrom, L., Tran, B., Tsipras, D., et al. (2019). Exploring the landscape of spatial robustness. In International conference on machine learning, pp. 1802–1811.

Fawzi, A., & Frossard, P. (2015). Manitest: Are classifiers really invariant? In British machine vision conference (BMVC), CONF.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., et al. (2014a). Generative adversarial nets in Advances in neural information processing systems, pp. 2672–2680.

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

Hayes, J., & Danezis, G. (2018). Learning universal adversarial perturbations with generative models. In 2018 IEEE security and privacy workshops (SPW), IEEE, pp. 43–49.

He, K., Zhang, X., Ren, S., et al. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770–778.

Hendrik Metzen, J., Chaithanya Kumar, M., Brox, T., et al. (2017). Universal adversarial perturbations against semantic image segmentation. In Proceedings of the IEEE international conference on computer vision, pp. 2755–2764.

Huang, W., Sun, Y., Sharp, J., et al. (2019). Coverage guided testing for recurrent neural networks. arXiv preprint arXiv:1911.01952.

Huang, X., Kroening, D., Ruan, W., et al. (2020). A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. *Computer Science Review, 37*(100), 270.

Isola, P., Zhu, J. Y., Zhou, T., et al. (2017). Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1125–1134.

Jaderberg, M., Simonyan, K., Zisserman, A., et al. (2015). Spatial transformer networks. *Advances in Neural Information Processing Systems, 28,* 2017–2025.

Johnson, J., Alahi, A., & Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In European conference on computer vision, pp. 694–711. Springer

Krizhevsky, A., Hinton, G., et al. (2009). *Learning multiple layers of features from tiny images*. Toronto, ON, Canada: University of Toronto.

Lenc, K., & Vedaldi, A. (2015). Understanding image representations by measuring their equivariance and equivalence. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 991–999.

Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3431–3440.

Madry, A., Makelov, A., Schmidt, L., et al. (2017). Towards deep learning models resistant to adversarial attacks. arXiv preprint arXiv:1706.06083.

Moosavi-Dezfooli, S. M., Fawzi, A., & Frossard, P. (2016). Deepfool: a simple and accurate method to fool deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2574–2582.

Moosavi-Dezfooli, S. M., Fawzi, A., Fawzi, O., et al. (2017). Universal adversarial perturbations. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1765–1773.

Mopuri, K., Garg, U., Venkatesh, & Babu, R. (2017). Fast feature fool: A data independent approach to universal adversarial perturbations. In British machine vision conference 2017, BMVC 2017, BMVA Press.

Mopuri, K., Ganeshan, A., & Babu, R. (2019). Generalizable data-free objective for crafting universal adversarial perturbations. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 41*(10), 2452–2465.

Mu, R., Ruan, W., Soriano Marcolino, L., et al. (2021). Sparse adversarial video attacks with spatial transformations. In The 32nd British machine vision conference (BMVC'21).

Mu, R., Ruan, W., Marcolino, L. S., et al. (2022). 3dverifier: Efficient robustness verification for 3d point cloud models. *Machine Learning*. https://doi.org/10.1007/s10994-022-06235-3.

Mummadi, C. K., Brox, T., & Metzen, J. H. (2019). Defending against universal perturbations with shared adversarial training. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 4928–4937.

Poursaeed, O., Katsman, I., Gao, B., et al. (2018). Generative adversarial perturbations. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4422–4431.

Reddy Mopuri, K., Ojha, U., Garg, U., et al. (2018). Nag: Network for adversary generation. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 742–751.

Ruan, W., Huang, X., & Kwiatkowska, M. (2018). Reachability analysis of deep neural networks with provable guarantees. In International joint conference on artificial intelligence (IJCAI), pp. 2651–2659.

Ruan, W., Wu, M., Sun, Y., et al. (2019). Global robustness evaluation of deep neural networks with provable guarantees for the hamming distance. In Proceedings of the 28th international joint conference on artificial intelligence (IJCAI), pp. 5944–5952.

Rudin, L. I., Osher, S., & Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena, 60*(1–4), 259–268.

Russakovsky, O., Deng, J., Su, H., et al. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision, 115*(3), 211–252.

Shafahi, A., Najibi, M., Xu, Z., et al. (2020). Universal adversarial training. In Proceedings of the AAAI conference on artificial intelligence, pp. 5636–5643.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Sun, M., Tang, F., Yi, J., et al. (2018a). Identify susceptible locations in medical records via adversarial attacks on deep predictive models. In Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining, pp. 793–801.

Sun, Y., Wu, M., Ruan, W., et al. (2018b). Concolic testing for deep neural networks. In The 33rd ACM/IEEE international conference on automated software engineering (ASE).

Szegedy, C., Zaremba, W., Sutskever, I., et al. (2014). Intriguing properties of neural networks. In International conference on learning representations (ICLR).

Szegedy, C., Liu, W., Jia, Y., et al. (2015). Going deeper with convolutions. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1–9.

Toffoli, T., & Margolus, N. (1987). *Cellular automata machines: A new environment for modeling*. Cambridge: MIT press.

Wang, F., Zhang, Y., Zheng, Y., et al. (2021). Gradient-guided dynamic efficient adversarial training. arXiv preprint arXiv:2103.03076.

Wang, F., Zhang, C., Xu, P., et al. (2022). Deep learning and its adversarial robustness: A brief introduction. *Handbook on computer learning and intelligence: Volume 2: Deep learning, intelligent control and evolutionary computation* (pp. 547–584). Singapore: World Scientific.

Wang, Q., Guo, W., Zhang, K., et al. (2017). Adversary resistant deep neural networks with an application to malware detection. In Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining, pp. 1145–1153.

Wang, Z., Bovik, A. C., Sheikh, H. R., et al. (2004). Image quality assessment: Ffrom error visibility to structural similarity. *IEEE Transactions on Image Processing, 13*(4), 600–612.

Wong, E., Rice, L., & Kolter, J. Z. (2019a). Fast is better than free: Revisiting adversarial training. In International conference on learning representations.

Wong, E., Schmidt, F., & Kolter, Z. (2019b). Wasserstein adversarial examples via projected sinkhorn iterations. In International conference on machine learning, pp. 6808–6817.

Wu, H., & Ruan, W. (2021). Adversarial driving: Attacking end-to-end autonomous driving systems. arXiv preprint arXiv:2103.09151.

Wu, M., Wicker, M., Ruan, W., et al. (2020). A game-based approximate verification of deep neural networks with provable guarantees. *Theoretical Computer Science, 807,* 298–329.

Xiao, C., Li, B., Zhu, J. Y., et al. (2018a). Generating adversarial examples with adversarial networks. arXiv preprint arXiv:1801.02610.

Xiao, C., Zhu, J. Y., Li, B., et al. (2018b). Spatially transformed adversarial examples. In International conference on learning representations.

Xu, P., Ruan, W., & Huang, X. (2022). Quantifying safety risks of deep neural networks. Complex & Intelligent Systems pp 1–18.

Yin, X., Ruan, W., & Fieldsend, J. (2022). Dimba: Discretely masked black-box attack in single object tracking. *Machine Learning*. https://doi.org/10.1007/s10994-022-06252-2.

Zhang, C., Benz, P., Karjauv, A., et al (2021a). Data-free universal adversarial perturbation and black-box attack. In Proceedings of the IEEE/CVF international conference on computer vision, pp. 7868–7877.

Zhang, C., Ruan, W., & Xu, P. (2023). Reachability analysis of neural network control systems. In Proceedings of the AAAI conference on artificial intelligence (AAAI'23).

Zhang, R., Isola, P., Efros, A. A., et al. (2018). The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 586–595.

Zhang, T., Liu, S., Wang, Y., et al. (2019). Generation of low distortion adversarial attacks via convex programming. In 2019 IEEE international conference on data mining (ICDM), IEEE, pp. 1486–1491.

Zhang, T., Ruan, W., & Fieldsend, J. E. (2022). Proa: A probabilistic robustness assessment against functional perturbations. In Joint European conference on machine learning and knowledge discovery in databases (ECML/PKDD'22).

Zhang, Y., Ruan, W., Wang, F., et al. (2020). Generalizing universal adversarial attacks beyond additive perturbations. In 2020 IEEE international conference on data mining (ICDM'20), IEEE, pp. 1412–1417.

Zhang, Y., Wang, F., & Ruan, W. (2021b). Fooling object detectors: Adversarial attacks by half-neighbor masks. arXiv:2101.00989.

Zhu, J. Y., Park, T., Isola, P., et al. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE international conference on computer vision, pp 2223–2232.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.