DeepSearch: Deep, Deeper, and Wider

Minsoo Kang, Mai Tung Duong, Aitolkyn Baigutanova, Sanjarbek Rakhmonov
KAIST, School of Computing, Undergraduate

***Abstract*—Blackbox adversarial attack poses huge security risks to AI-based software. However, current researches usually focus on computer vision - a well-explored domain and often assumes ideal, somewhat impractical setting. In this work, we attempted to generate adversarial reaction using a simple search-based method called DeepSearch. We than test the method on three more challenging but realistic settings and expand the method into audio domain.**

***Index Terms*—Adversarial attack, Audio, Targeted attack, Categorical feedback, Artifact**

Our GitHub repository

I. Introduction

Deep Neural Networks are being used in many aspects of our lives. The last few years of research in the field made significant improvements to the accuracy and efficiency of the deep neural networks. Because neural networks are becoming increasingly involved in our lives, security of the networks is more important than ever. Therefore identifying vulnerabilities of the networks and preventing them is very important.

One of those security risks is adversarial attacks to the deep neural network. Adversarial attack is a machine learning technique attempting to fool models by supplying corrupted input. By inserting a small noise to the original input, which is undetectable to humans, a different output is produced by the network.

Initial research on the topic was done on white box attacks. White box attack means an attack made to the networks whose internal network structure is available to the attacker. That is usually not the realistic setting, because softwares usually do not disclose their internal network structure, but only release the API. Focus of our project is black-box attack which is done without knowing the internal structure of the model, where only output of the model can be utilized as a feedback.

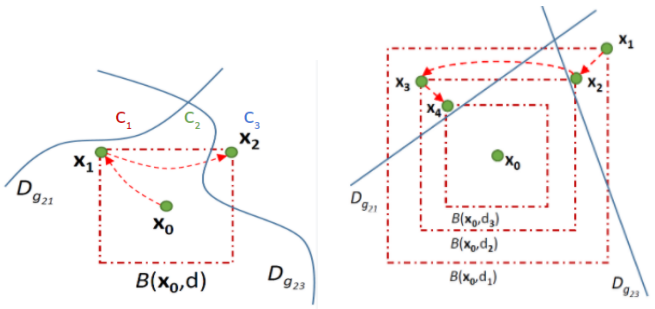
Our project is based on paper "DeepSearch: A Simple and Effective Blackbox Attack for Deep Neural Networks". Paper introduces simple search based black-box attack called DeepSearch. We successfully replicated their work and made the following improvements:

(1) Targeted attack
(2) Improved grouping
(3) Categorical attack

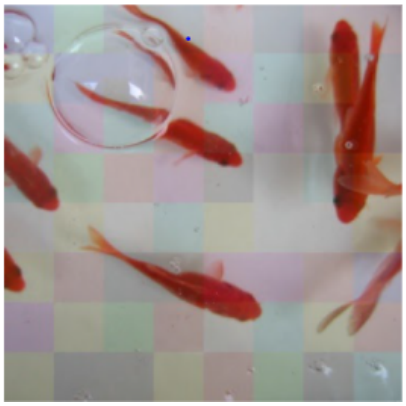
Moreover, we tried to expand the scope of the black-box attack to less studied areas. To achieve this, we modified the DeepSearch algorithm to make an attack on audio classification networks.

II. Related work

Our baseline method is DeepSearch algorithm described in the related paper "DeepSearch: A Simple and Effective Blackbox Attack for Deep Neural Networks" [1]. Simply



((a)) Hill climbing (source: [1]) ((b)) Iterative refinement (source: [1])



((c)) Hierarchical grouping and its artifact

Figure 1. DeepSearch algorithm

speaking, we are given an original image, and the access to the classifier. The only information we are given to make searching decisions would be the probability output of the classifier. And we would like to slightly mutate the image to get the wrong output. But we don't want to query the model too many times and want our adversarial input as similar as possible to the original one. The paper suggests a search-based method, assisted with query reduction and difference reduction.

Essentially, DeepSearch aims to find the noise pattern for adversary by hill-climbing-ish method on each pixel RGB value (Fig. 1(a)). The algorithm uses a fixed-size step. It takes a step forward and stays if the point is closer to the decision boundary, else it moves backward. We called this

process *mutation*. The mutation is iteratively performed until out image cross the decision boundary (misclassified). The algorithm is proven effective for the most general cases of multiclass nonlinear classifiers.

However, the complexity is exploding as our image gets larger, as DeepSearch performs on each pixel value. Hierarchical Grouping is introduced to reduce query usage, utilizing the locality to group adjacent pixels together (Fig. 1(c)).

Then, the algorithm use Iterative Refinement to reduce distortion, simply narrowing down the distortion as long as the input stay misclassified. (Fig. 1(b)) The algorithm is assisted with bisect search.

Even though the paper proposed an effective methods, we have some critical comments:

- The attack is somewhat benign, in that it only make the adversarial image but we have no control over which class the model misclassified it into.
- It assumes the availability of probability output, which is not always the case. In fact most application in software engineering does not have the full probability information revealed to the user.
- The grouping although simple, it creates 'square blocking' artifact that looks unnatural to human eyes (Fig. 1(c)).
- The paper is also only explore the image domain. For other domain which is not well-studied such as audio or NLP, it is unsure that the algorithm works

Our project aims to solve these problem, and the main contribution is summarized as follow:

- **Deep** (Replication): Successfully replicating the DeepSearch algorithm
- **Deeper** (Improvement): Test the algorithm in some more challenging setting: targeted attack, categorical feedback. Successfully implement a working grouping scheme with less artifact.
- **Wider** (Expansion): Successfully adapt the algorithm into the audio domain.

III. METHODOLOGY

In this section, we talk about our contribution in more details. Our work can be divided into three parts: Replication, improvements, and expansion. More details of the implementation can be found in our codes.

A. Replication

In replication, we re-implemented the baseline study's algorithm to facilitate our experiments. The "iterative refinement" of the original study was omitted as the generation of adversity was sufficient for our intention of research.

Our implementation can be divided into 4 parts, which are wrapper, mutation, evaluation, and the DeepSearch algorithm. The wrappers are the parts that is used as interfaces to the model, the mutation part handles the grouping and modifying of the image patches to its lower or upper bound, the evaluation part contains various evaluation scheme that will suits each attack mode, and the DeepSearch algorithm followed exact procedures provided by the baseline study.

An important part of the Zhang et al.'s study didn't get its fair share of attention, so we would like to make a quick mention. The 'batching' part, which gives intermediate steps to the whole mutation of the image. After a few trials, we found out that this batching mechanism makes the most significant impact to the performance since it makes better approximation to the assumption of the algorithm.

B. Improvement

In order to seek more realistic attack scenarios in software engineering perspective, we made the following three improvements: Targeted attack, improved grouping, and categorical attack.

1) Targeted attack [2]: In targeted attack, the goal was to generate specifically labeled adverse reaction. Whereas the baseline method only focused on achieving adversity regardless of the resulting class.

In order to achieve this goal, we designed the algorithm to only use the probability of a targeted class, and try to increase the value. Meanwhile, the baseline method tried to decrease the probability gap between the original class and the next most probable class.

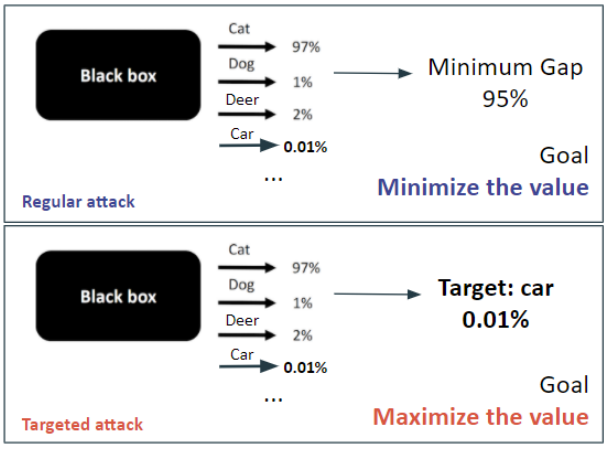


Figure 2. Targeted Attack implementation

2) Improved grouping: In this part of our research, we were trying to fix visual artifact problem present in baseline method's grouping. Zhang et al. used square pattern grouping to exploit the locality of visual images. However, depending on the scene, even a slight abrupt change in intensity value in an image can create a noticeable mark as can be seen in Fig. 1(c).

To solve this, we avoid unnatural pattern by simply sampling the pixels randomly to create groups. Hence, no particularly distinctive group border is generated, and the mutation could be disguised as 'noise'.

3) Categorical attack [2]: One of the assumptions by Zhang et al. is that classifiers output the probability of all the classes. Realistically, classifier APIs are not always expected to return full outcome, but only a few of them or even only

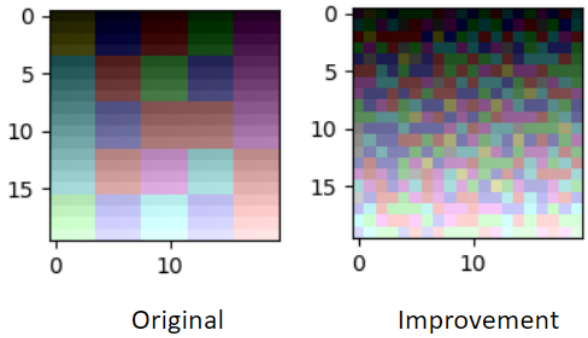


Figure 3. Alternative grouping scheme example on a simple image

labels of top few probable classes.
In this part of experiment, we implemented a new evaluation method to only use the information of 'top 3' classes. The idea is to use statistic data for probability approximation. The implementation of this evaluation requires proportionally more query. The new evaluation challenges the confidence of the prediction by querying slightly perturbed image. These multiple results are then compiled into a histogram and converted into scalable confidnce values. In particular, gaussian noise of acceptable scale was added for perturbation of the images.

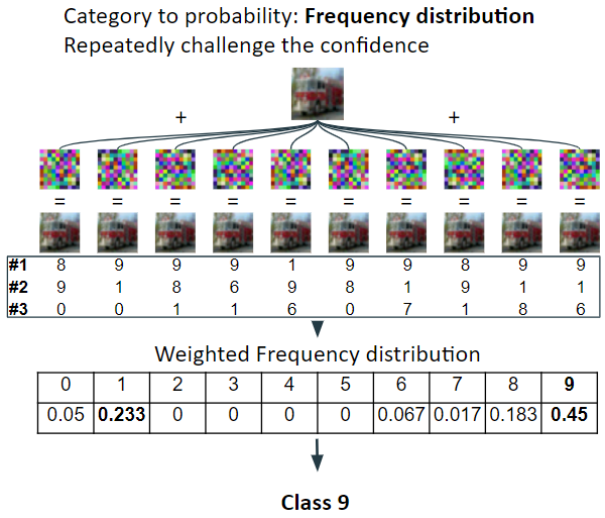


Figure 4. Categorical probability assessment

C. Expansion

In expansion, we wanted to try the robustness of the algorithm on other domains. For the time being, we've only succeeded in testing on audio domain and retrieving back the audio data.
There were several options of audio representation to adopt for our purpose such as PCM audio data and spectrogram matrix, and we chose spectrogram. This decision was because spectrogram is already a 2-dimensional data that can be input right away, and also because spectrogram perturbation will hide our mutation due to its robustness to noise.

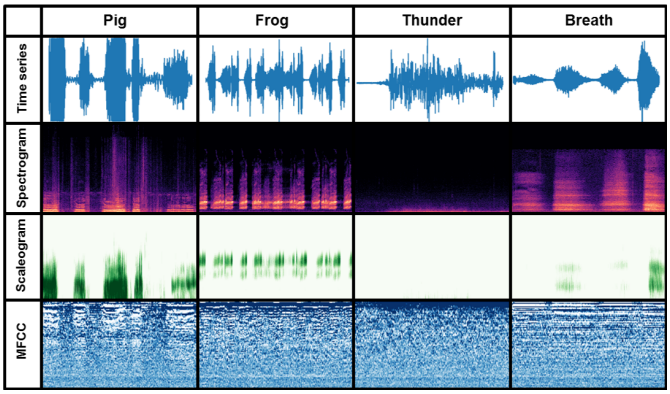


Figure 5. Sound representations (source)

We built the audio classifier based on spectrograms and ResNet with 5 classes (cat, dog, human, kid, parrot) to accuracy of 98%.

IV. Results

As in the original paper, we used two datasets for evaluation of our results: CIFAR10 and ImageNet. Due to the lack of computational power resources, we tested our algorithms on a smaller sample size compared to the baseline method.

A. Replication

As Table 1 shows, the success rate achieved with in our replication part is comparable with the original work, whereas the average query number is higher except for the case with CIFAR-10 defended dataset.

Table I
Replication results

		Success Rate (%)		Average Query	
		Reseach (on 1000)	Ours (on 50 - 100)	Research (on 1000)	Ours (on 50 -100)
ImageNet		99.3	98	561	666
Cifar-10	Undefended	100	100	247	531
	Defended	47.7	44	963	925

B. Improvement

1) Targeted attack: We have used two target classes: "street sign" for ImageNet and "airplane" for CIFAR-10 datasets. Table 2 shows the results we obtained with the targeted attack. We managed to successfully attack more than a half of images we used from ImageNet and all from CIFAR-10.

Table II
Targeted attack results

	Success Rate, %	Average Query
ImageNet	54	5547
Cifar-10 (Undefended)	100	931

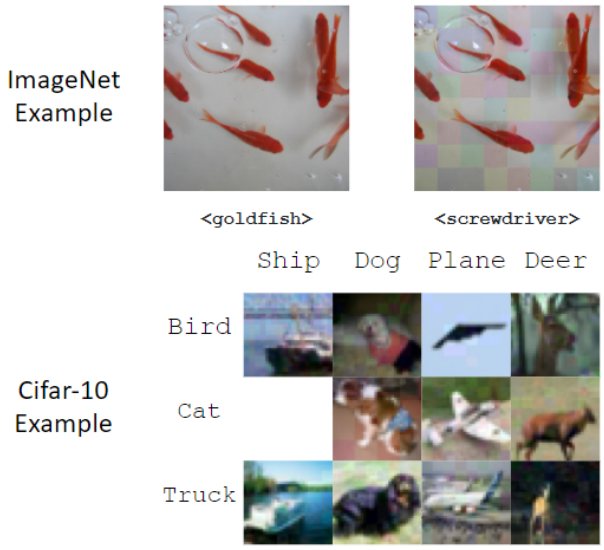


Figure 6. Replicated results

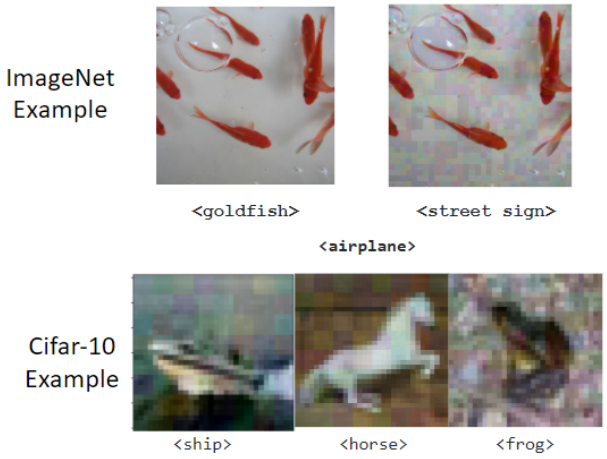


Figure 7. Targeted attack images

2) Improved grouping: Table 3 demonstrates results we could achieve in our experiments with random grouping. It successfully attacked more than 80% of the images from both testing datasets. When compared with the baseline methodology, random grouping performs worse in terms of success rate and average query rate but significantly reduces artifacts of perturbed images.

Table III
IMPROVED GROUPING RESULTS

	Success Rate, %	Average Query
ImageNet	80	1522
Cifar-10 (Undefended)	96	581

3) Categorical attack: For our third improvement, categorical attack, we only tested with 50 images from CIFAR-10. The success rate that we achieved is 6%, while average query is 5617.

C. Expansion

For our final step, expanding the original work, we tested DeepSearch algorithm in the audio domain. We could successfully attack the spectrogram obtained from the audio files and reproduce the sound from the perturbed examples. The result

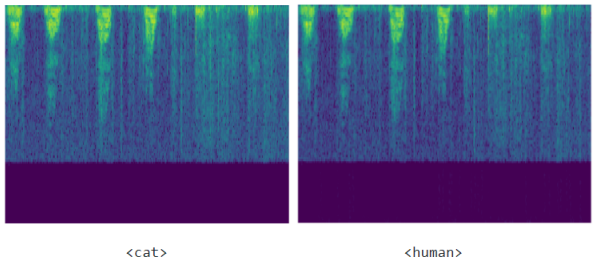


Figure 8. Audio perturbation

sounds can be heard in our GitHub.

V. CONCLUSION

In this work we have proposed several ways of how to improve baseline algorithm DeepSearch for adversarial blackbox attacks. We have found that the algorithm can be effectively applied on targeted attacks and audio domain. We have also observed that categorical attacks might not be successful in query usage. Finally, the random grouping is effective in reducing artifacts, but the quality of the images was also reduced.

VI. FUTURE WORK

In this work, we use spectrogram to represent sound, but there are actually much more representation (Fig. 5).

This poses the next problem. Out implementation assumes that we know the model uses spectrogram to represent sound. It is not completely blackbox. We would like to make it representation-independent, by either direct mutation on sound or back-and-forth implementation (raw audio to image, perform mutation on image, convert back to sound and feed to the the model).

REFERENCES

[1] F. Zhang, S. P. Chowdhury, and M. Christakis, "DeepSearch: A Simple and Effective Blackbox Attack for Deep Neural Networks," 2019, accessed: 20-12-2020. [Online]. Available: https://arxiv.org/abs/1910.06296

[2] J. Chen, M. Su, S. Shen, H. Xiong, and H. Zheng, "POBA-GA: Perturbation Optimized Black-Box Adversarial Attacks via Genetic Algorithm," 2019, accessed: 20-12-2020. [Online]. Available: https://arxiv.org/pdf/1906.03181.pdf