Impact of reverberation through deep neural networks on adversarial perturbations

Romain Cohendet, Miguel Solinas, Rémi Bernhard, Marina Reyboz,
Pierre-Alain Moellic, Yannick Bourrier, Martial Mermillod

▶ **To cite this version:**

Romain Cohendet, Miguel Solinas, Rémi Bernhard, Marina Reyboz, Pierre-Alain Moellic, et al.. Impact of reverberation through deep neural networks on adversarial perturbations. 2021. hal-03173407

HAL Id: hal-03173407
https://hal.science/hal-03173407

Preprint submitted on 18 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Impact of reverberation through deep neural networks on adversarial perturbations

Romain Cohendet
CEA, List
romain.cohendet@laposte.net

Miguel Solinas
CEA, List
miguelangel.solinas@cea.fr

Rémi Bernhard
CEA, Tech
remi.bernhard@cea.fr

Marina Reyboz
CEA, List
marina.reyboz@cea.fr

Pierre-Alain Moellic
CEA, Tech
pierre-alain.moellic@cea.fr

Yannick Bourrier
Université Grenoble-Alpes
yannick.bourrier@univ-grenoble-alpes.fr

Martial Mermillod
Université Grenoble-Alpes
martial.mermillod@univ-grenoble-alpes.fr

Abstract

The vulnerability of Deep Neural Network (DNN) models to maliciously crafted adversarial perturbations is a critical topic considering their ongoing large-scale deployment. In this work, we explore an interesting phenomenon that occurs when an image is reinjected multiple times into a DNN, according to a procedure (called reverberation) that has been first proposed in cognitive psychology to avoid the catastrophic forgetting issue, through its impact on adversarial perturbations. We describe reverberation in vanilla autoencoders and propose a new reverberant architecture combining a classifier and an autoencoder that allows the joint observation of the logits and reconstructed images. We experimentally measure the impact of reverberation on adversarial perturbations placing ourselves in a scenario of adversarial example detection. The results show that clean and adversarial examples – even with small levels of perturbation – behave very differently throughout reverberation. While computationally efficient (reverberation is only based on inferences), our approach yields promising results for adversarial examples detection, consistent across datasets, adversarial attacks and DNN architectures.

1. Introduction

Deep neural networks (DNNs) are vulnerable to maliciously crafted inputs that visually resembles the learned data but translate into erroneous predictions, known as *adversarial examples* [28]. This problem has the potential to cause dramatic damages considering the large deployment of DNNs, in particular when the predictions involve real-world decisions (e.g. autonomous cars, banking system, flow regulation). In consequence, a variety of defenses against adversarial examples have been proposed.

The defenses to mitigate adversarial attacks can be broadly classified into two categories: proactive or reactive defenses. Proactive defenses intervene before testing time by modifying the architecture of the attacked model or its training for it to become more robust against adversarial examples. For examples, adversarial learning belongs to this category. Reactive defenses play the part at testing time. Methods based on inputs transformation to filter out adversarial examples typically belong to this category.

The first range of defenses aims at correctly classifying the adversarial examples. Some of these methods rely on a preprocessing of the inputs to reduce or remove the adversarial perturbations, for example as in [11] by using a denoising autoencoder. However, most of these defenses were proved to be ineffective against advanced attackers. A popular robustness-based defense is adversarial training, which consists in augmenting the training set with adversarial examples [9, 21]. Subsequent improvements were proposed to make the models even more robust to adversarial examples within a given range [30, 3, 7, 32]. While adversarial training improves robustness against specific attacks, these computationally expensive defenses target a limited range of adversarial perturbations, and the DNNs trained this way are still vulnerable to counter-attacks.

The difficulty of correctly classifying adversarial examples shift some of the effort towards detecting them instead. In the same vein as adversarial training for robustness-based

defenses, learning-based adversarial detection methods utilize adversarial examples during the training phase, but of the model used as a detector (e.g. [10]). Although both popular and reliable, these methods suffer from the weaknesses mentioned above for adversarial training. Another class of methods is based on extracting some knowledge from a set of data and use it to differentiate adversarial inputs, for example using some statistics [10], after having added random noise to the input [24] or transformed it [12]. Other approaches capitalize on the fact that an adversarial example may not fool all the classifiers and accordingly pass the input through various models to detect adversarial examples [19, 27]. Number of adversarial detection methods were proved not to be robust [4] and adversarial examples detection is still a challenging topic.

This paper introduces a novel approach to apprehend adversarial perturbations based on a reverberation of the inputs through an artificial neural network (ANN) . Reverberation refers to the process of reinjecting an input multiple times through an ANN by using as inputs the successive outputs produced by the model. It requires an architecture implementing an autoencoder function for the output to be used unchanged as input. When going through such a procedure, we observe that clean and adversarial examples behave very differently, which allows us to distinguish one from the other. To characterize the phenomenon, we place ourselves in an adversarial examples detection scenario to measure the impact of reverberation on adversarial perturbations. More precisely, we adopt a reactive defense scenario where we transform the inputs using reverberation through autoencoders trained with clean data. The principal contributions of this paper are as follow:

- We present the reverberation procedure as proposed in cognitive psychology to deal with catastrophic forgetting (Section 2.1) and give our motivations to introduce it in the study of adversarial perturbations (Section 3.1);
- We detail a way to implement reverberation for adversarial examples detection, in particular by proposing a new "reverberant" architecture (Section 3);
- We conduct an experiment on different datasets, varying DNN architectures and adversarial attacks, to propose a first characterization of the impact of reverberation on adversarial perturbations (Sections 4 and 5).

2. Related work

In this section, we first describe the reverberation procedure that has been first proposed in cognitive psychology (Section 2.1) and then present the adversarial examples detection methods that relates to the work presented in this paper (Section 2.2).

2.1. The reverberation procedure as proposed in cognitive psychology

In contrast to human brain, ANNs tend to forget their previous knowledge when learning new information, a well-known problem called *catastrophic forgetting*. Research in cognitive psychology [2], that aims at computationally modelling human memory in a biologically plausible way, proposed an original solution to this problem. The core idea is to reinject several times a noise into an ANN to create "pseudo-examples" that reflect its current state of knowledge, to be learned with new information. To implement a "reverberant" architecture into a ANN, Ans and Rousset add backward connections between some layers besides the regular forward connections. They found that using these pseudo-examples along with the new information during subsequent training benefit the preservation of the anterior knowledge. In our case this means that a noise that reverberates into an ANN converges one way or another towards the internal information distributed within its parameters since it enables to capture it. To the best of our knowledge, nowhere this thinking was applied and reverberation-based approaches proposed in the study of adversarial perturbations or detection in the broad sense (including detection of novelty, anomalies and adversarial examples).

2.2. Adversarial examples detection

Adversarial examples detection methods try to categorize inputs as clean or adversarial. Among them, the learning-based detection methods [10, 8] utilize a network to detect adversarial examples. For example, the authors of [22] trained on both clean and adversarial examples a sub-network classifier to detect adversarial examples for each adversarial attack considered. The detector sounds to achieve a good performance when the adversarial examples to be detected at testing time and the ones used to train the detector were generated with the same adversarial attack, but the generalization across different adversarial attacks and attacks parameters is poor. This highlights a major drawback of these methods: to achieve a proper accuracy, they require to train the detector on adversarial examples generated using all adversarial attacks in addition to clean data. Besides being computationally expensive, they may not be useful against new adversarial attacks. In [5], MagNet uses detectors, but they are not trained on adversarial examples: they learn the manifold of clean data and compare the inputs with the manifold at testing time. The detection scenario in which we explore the impact of reverberation on adversarial perturbations (see Section 3.3) also requires only clean data.

Some methods try to remove the effect of adversarial attacks by adding random noise to the inputs [25, 13] or by transforming them. In [20], the authors compare the classification results of the input and its denoised version to

detect adversarial examples. The authors of [31] compare the predictions of the model for the original input and for a squeezed version of the same input to detect adversarial examples. In [12], PCA is used to transform the inputs. In [23], noises are added to the PCA transformations of the inputs which substantially help to detect adversarial examples which are close to or far from the decision boundary. In this paper, reverberation is explored as an input transformation-based procedure that operates the transformation directly into the detector model as explained in Section 3.2.

DNNs give wrong predictions with high confidence values to adversarial examples. Some methods try to detect adversarial examples based on the observation of the logits (i.e. the absolute logit activation values that usually are the input of the last, softmax layer in classifiers). In [1], the authors study how the logits are distributed for adversarial examples compared to clean data and show that the logits provide relevant information to differentiate them. Consequently, they train a network that takes the logits as input and predict if the classification is correct or not. In [25], the authors observe that robustness of logits to noise depend on whether the input is clean or adversarial, which allows them to differentiate the two sorts by using a statistical test. In this paper, we introduce a new reverberant architecture that enables us to quantify the change in the logits while the inputs are reverberated (see Section 3.2.2).

3. Proposed method

In this section, we first reveal our motivations behind introducing reverberation in the study of adversarial perturbations (Section 3.1). Then, we detail the reverberation procedure and propose a new reverberant architecture (Section 3.2). We finally propose to measure the impact of reverberation on adversarial perturbations in an adversarial examples detection scenario (Section 3.3).

3.1. Motivations

Our main source of inspiration is [2], where the authors propose the reverberation procedure to model the functioning of human memory, and in particular to deal with catastrophic forgetting. They show that reinjecting multiple times a noise within an ANN captures a part of its internal knowledge. Moreover, reinjecting different noises allows them to sample the learned data distribution to generate synthetic examples diverse enough to reflect the anterior state of knowledge of the ANN. Since reverberation produces from noises synthetic data that resembles the learned data, then it makes the noises to converge one way or another towards the ANN internal knowledge and not all noises converge towards the same part of this knowledge. This inspired us this hypothesis: while reverberated, the inputs converge differently depending on their resemblance with the learned data distribution, which enables to detect novel or abnormal data. Accordingly, we expect adversarial data not to behave the same way as clean data when reverberated, and that this to translate into a quantifiable difference.

Additionally, several studies proposing defenses against adversarial examples inspired us to import reverberation to study adversarial perturbations, in particular the ones focusing on input transformation and using autoencoders to remove amounts of the adversarial perturbations. In some ways, by applying reverberation to study adversarial perturbations, we push the idea of input transformation by multiplying inferences in an autoencoder. Moreover, studies that observe the logits to detect adversarial examples inspired us the reverberant hybrid model we introduce in Section 3.2.2, which combines a classifier and an autoencoder and enables us to observe the displacement of the logits through reverberation.

3.2. Reverberant models

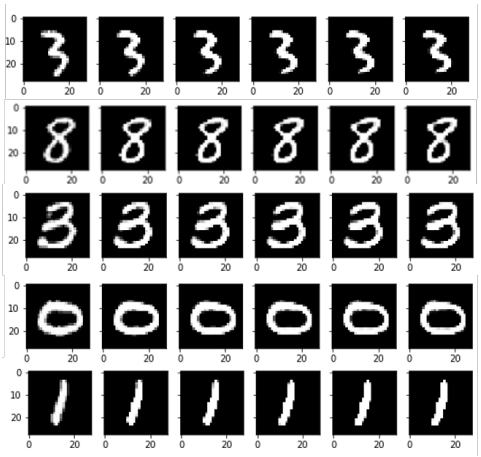
We define reverberation as the process of reinjecting an input into an ANN at least one time. One way to implement it is to use an autoencoder that reconstructs the input for the output to be reinjected in the same format. In this section, we present two reverberant architectures: a vanilla autoencoder (Section 3.2.1) and a hybrid model combining an autoencoder and a classifier (Section 3.2.2).

3.2.1 Reverberation through autoencoders

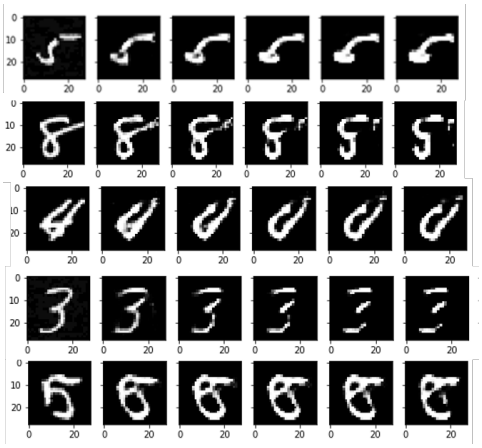
The simplest architecture, inherently reverberant, is the basic autoencoder. As such, this is one of the two architectures we propose to conduct a first analysis of the effect of reverberation on adversarial perturbations. Reverberation into an autoencoder simply consists in using as input of the model its own output. Figure 1 shows examples of reverberated MNIST test set images [18] and adversarial examples generated by using the Fast Gradient Sign Method (FGSM) [9] (see Section 4.1 for details about the parameters of the attack). We can observe that clean images tend to remain stable when reverberated through the autoencoder while adversarial examples change, even for a small level of perturbation. Figure 2 illustrates a similar phenomenon for reverberated Fashion-MNIST test set images [29] and adversarial examples generated by using the L_∞-norm Projected Gradient Descent attack (L_∞-PGD) [16].

3.2.2 Reverberation through hybrid models

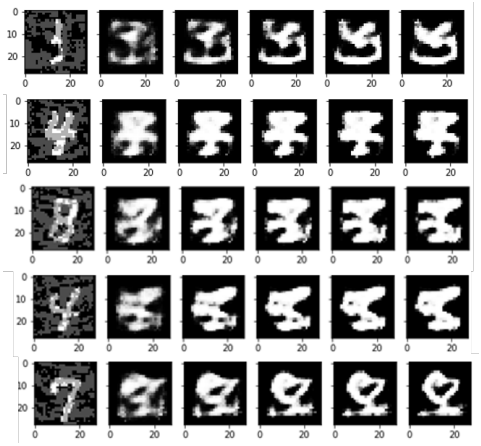
To study reverberation in more complex datasets, we propose a hybrid architecture that combines a classifier and an autoencoder. Figure 3 shows such a vanilla hybrid reverberant architecture. The ANN has two outputs: an output intended for the input replication and an output for the classification. Besides enabling us to use the logits information, we propose this hybrid architecture for two reasons. Carlini



(a) Clean images

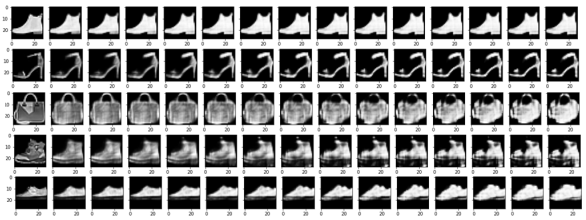


(b) $\epsilon = 0.05$

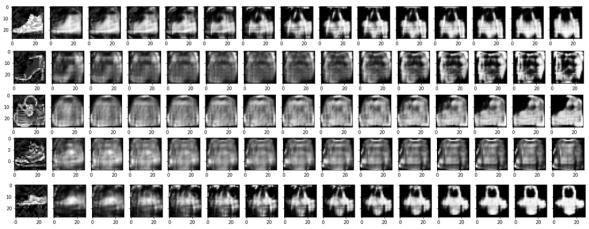


(c) $\epsilon = 0.3$

Figure 1: Reverberations through a vanilla autoencoder of MNIST images and adversarial examples generated by attacking a LeNet model with FGSM attack. We can observe that while clean images remain stable during reverberation, adversarial images change. This is particularly the case when the perturbation size ϵ is high. Measuring the difference between the original input and the output resulting from the fifth reverberation is enough to distinguish adversarial examples from clean images, as proved by the experiment presented below (see Section 4).



(a) Clean images



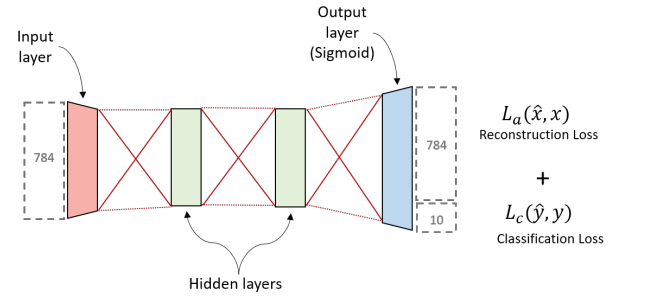
(b) $\epsilon = 0.2$

Figure 2: Reverberations through a vanilla autoencoder of Fashion-MNIST images and adversarial examples generated by using the L_∞-PGD attack. We can observe a phenomenon similar to the one observed in Figure 1 for MNIST images and adversarial examples generated with the FGSM attack.

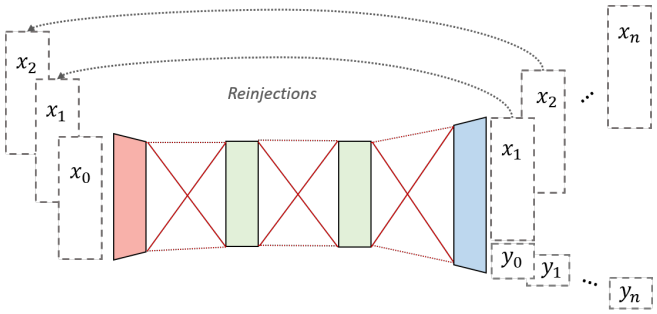
et al. [4] find that using a second neural network to identify adversarial examples is the least effective defense among ten tested. While we propose in this paper a first study of reverberation in an adversarial examples detection scenario, we want the reverberation to possibly be integrated directly into the attacked model. This a point we will address in a further paper (while we briefly discuss this point in Section 6): the hybrid architecture enables to directly address the problem of robustness of DNNs to adversarial examples (i.e. their correct classification) instead of the problem of detection. As part of the experiment presented in this paper, to study the impact of reverberation on adversarial examples generated from CIFAR-10 images [15] we build a hybrid architecture on a DenseNet classifier (see Section 4.2).

3.3. Adversarial examples detection

We observed that clean and adversarial examples behave very differently when reverberated. To measure the impact of reverberation on adversarial perturbation, one can place himself in an adversarial examples detection scenario (this is the case in our experiment detailed in Section 4). To detect adversarial examples with a basic autoencoder trained on clean data, each new incoming input is reverberated several times through the model. After n reverberations, an image similarity metric can be used to measure the difference between the original input and the last reconstructed input.



(a) Training



(b) Reverberations

Figure 3: Training and reverberations into a vanilla hybrid reverberant model for adversarial examples detection. The inputs are MNIST images (784 pixels, 10 classes). (a) The training is done by reducing the binary cross-entropy loss functions for reconstruction and classification. (b) Once the training is complete, the model can be used in detection mode. Each new incoming input is reverberated multiple times, and we measure the discrepancy between the logits and reconstructed images throughout the reverberation process.

The detection is slightly more complex in reverberant hybrid models. As shown in Figure 3b, this new architecture allows us to measure the difference between logits across reverberations besides the image similarity. The histogram is Figure 4 shows the average Mean Squared Error (MSE) between the logits resulting from the first inference and the logits resulting from the 10^{th} reverberation for clean CIFAR-10 images and their adversarial examples counterparts generated by attacking our hybrid model (presented in Section 4.2) with FGSM (detailed in Section 4.1). We can observe that the MSE between the logits enables to discriminate to a certain extent between clean and adversarial examples.

4. Experimental setup

We conduct an experiment to measure the impact of reverberation on adversarial perturbations by adopting an adversarial examples detection scenario. More precisely, we

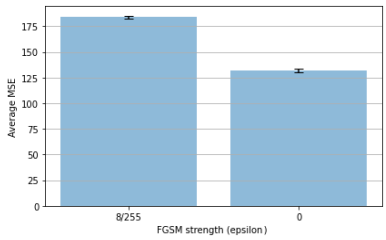


Figure 4: Average MSE between logits resulting from the first inference and logits resulting from the 10^{th} reverberation for clean CIFAR-10 images and adversarial examples generated from these images with FGSM. Error bars reflect the Standard Error of the Mean.

set up a reactive defense scheme where the detector is independent from the model under attack (i.e. it does not modify the target model) and from the adversarial attack. Consequently, the proposed models for adversarial examples detection use only clean data (in our case, only the training set of the considered datasets) to build their knowledge.

4.1. Data and adversarial examples generation

We measure the effect of reverberation on adversarial examples detection on three datasets: MNIST [18], Fashion-MNIST [29] and CIFAR-10 [15]. For each dataset, we generate adversarial examples from $10,000$ images of the test set by attacking classifiers with an untargeted attack (i.e. to generate adversarial examples misclassified by the classifier into any class as long as it is different from the true class). For MNIST, the attacked classifier is a LeNet [17] that achieves a classification accuracy of 98.76% on the test set. For Fashion-MNIST, it is a CNN-classifier that achieves a classification accuracy of 90.91% on the test set. For CIFAR-10, we vary the architecture of the models under attack to evaluate its impact on the detection rate of our detector (which is based on a DenseNet architecture [14], as explained in Section 4.2). Hence we use a DenseNet-121 and a VGG-13 [26] classifiers, that achieve a very close classification accuracy on the test set: 94.14% and 94.29% respectively.

We consider three adversarial attacks: FGSM [9], L_∞-PGD [16] and L_2 Carlini and Wagner attack (L_2-CW) [6]. Depending of their parameters, not all attacks are always successful at creating a "true" adversarial example that fool the classifier under attack. Moreover, some of the test set images are misclassified by the models before the adversarial attack. We discard these two sorts of images before adversarial examples detection, retaining only true adversarial examples. Consequently, in each condition tested, the attacked model obtain a classification accuracy of 0% on the adversarial examples retained for detection. Note that this implies a number of true adversarial examples different

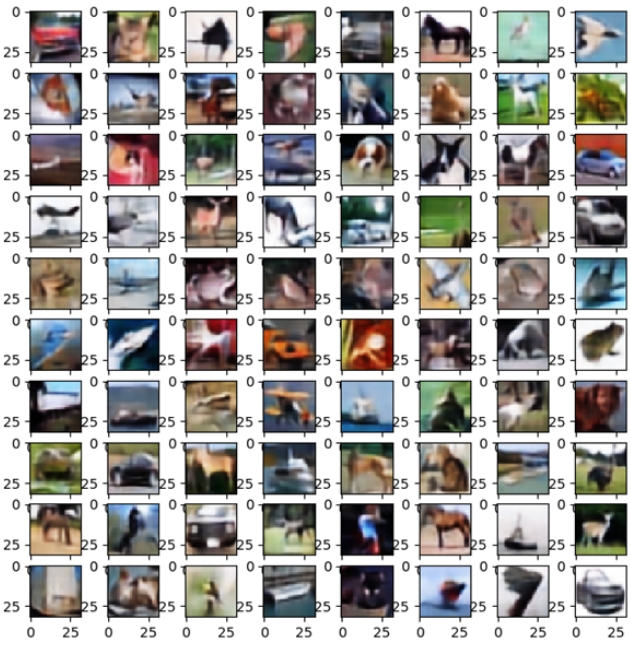


Figure 5: A batch of CIFAR-10 images reconstructed by our hybrid model.

in each condition, the minimum of all the conditions being 2356 true examples at testing time. Table 1 provides details about the parameters of the adversarial attacks in each experimental condition.

4.2. Reverberant models parameters and training

For MNIST and Fashion-MNIST, we train basic autoencoders with the images of the training set of the respective datasets. As previously mentioned, for CIFAR-10 we train a hybrid model, combining a classifier and an autoencoder, as presented in Figure 3. However, while the figure presents a vanilla hybrid model for the sake of clarity, we use a more complex model, based on a DenseNet architecture augmented with an autoencoder. As illustrated in Figure 3a, we use two loss functions for the training: the binary cross-entropy to translate the difference between the input images and their replications and the cross-entropy to translate the difference between the outputted labels and ground truth labels. The total loss function to be minimized by iteratively updating the model's parameters was the sum of these two functions. Our trained hybrid model achieves a classification accuracy of 92.83% on CIFAR-10 test set. Figures 5 shows a batch of test set CIFAR-10 images reconstructed by our hybrid model.

4.3. Metrics and method

At testing time, in each experimental condition the detector model is provided with $10,000$ clean examples from the test set of the considered dataset and the corresponding true adversarial examples (as defined in Section 4.1). For MNIST and Fashion-MNIST, each input is reverberated five times into the autoencoder, then we calculate the MSE between the original input and the image resulting of the last inference. For CIFAR-10, we calculate the MSE between the logits resulting from the first inference and the logits resulting from the 10^{th} inference. Note that we use only MSE for the sake of comparison, while we found other metrics (such as the Pearson correlation coefficient or Structural Similarity Index Measure) being occasionally better. A pretest enabled us to determine that five reverberations for MNIST and Fashion-MNIST, ten for CIFAR-10, if not always optimal, is enough to observe a clear divergence in the behaviour of clean and adversarial examples. Furthermore, for MNIST and Fashion-MNIST, we observed that the function that associates our metrics and reverberation is always monotonic during the first five reverberations: the MSE between the first inference output and the n^{th} inference output is always smaller that with the $n^{th}+1$ output.

To decide if an input is adversarial, a natural way is to extract some statistical descriptor from clean data to be compared with the inputs during testing time. In our experiment, we reverberate five times (ten for CIFAR-10) a sample of the clean data drawn from the considered dataset. Then, we compute a threshold to decide if the input is adversarial that depends on the tolerated false positive rate (FPR). We vary the FPR from 0 to 100 with steps equal to 0.1 and obtain a threshold T for each FPR i calculated as follows:

$$T_i = max(MSE_{clean}) \tag{1}$$

with MSE_{clean} being the list of MSE for each of the reverberated clean inputs minus the i^{th} highest MSE values. For example, for $FPR = 1$, which means that 1% of false positives is tolerated, the threshold is equal to the maximum of the 99% lowest values of the clean MSE list. At testing time, for each input we compare the MSE calculated after five (10 for CIFAR-10) reverberations with the threshold of the corresponding dataset in such a way that, if it is lower, the detector classifies the input as clean, or as adversarial otherwise. By varying the FPR, we calculate the Area Under the Receiver Operating Characteristic curve (AUROC).

5. Results

Table 2 shows the results obtained for adversarial example detection. The perturbation size ϵ tends to change depending on the considered dataset. We report in the table the results for attack parameters commonly found in the literature. The results show that reverberation has an important impact on adversarial perturbations, as measured through their detection. This is true for all the considered datasets and adversarial attacks, while the detection values are lower for CIFAR-10 and L_2-C&W. This is result one could expect, however, as CIFAR-10 is the most complex of the

	FGSM	L_∞-PGD	L_2-C&W
MNIST	ϵ=0.3	ϵ=0.3, α=0.01, nb_{iter}=100	c=5, κ=0, steps=1000, lr=0.1
Fashion-MNIST	ϵ=0.2	ϵ=0.2, α=0.01, nb_{iter}=100	c=5, κ=0, steps=1000, lr=0.1
CIFAR-10	ϵ=8/255	ϵ=8/255, alpha=2/255, nb_{iter}=100	c=5, κ=0, steps=1000, lr=0.1

Table 1: Parameters of the conducted adversarial attacks.

three datasets considered and L_2-C&W the most powerful adversarial attack of the three considered with the retained parameters. Keep in mind, however, that the difference between attack parameters and reverberant models makes not all the conditions fully comparable.

As mentioned, our reverberant hybrid model for CIFAR-10 is built on a DenseNet architecture similar to one of the classifier under attack – the other classifier having a VGG architecture. The results show no substantial difference regarding detection between the two architectures, which is consistent with the scenario of a detector fully independent from the model under attack.

For MNIST and Fashion-MNIST, we additionally tested a range of perturbation sizes for FGSM and PGD. Figure 6 shows the corresponding ROC curves. We can observe that, regardless of the dataset or attack, the detection is always made easier by a greater perturbation size (ϵ). At a certain level of perturbation, almost all the adversarial examples are detected even for low levels of tolerated false positives.

6. Discussion

We proposed a first analysis of the impact of reverberation on adversarial perturbations through a scenario of adversarial examples detection. Our experiment shows that, while simple and fast, reverberation has a different effect on clean and adversarial examples that allows for their differentiation to some extent. This is the first characterization of this phenomenon: there is probably room for improvement. The first thing one could do to maximise the distortions of adversarial examples under the effect of reverberation is to conduct an extensive study of the parameters that affect them. Besides obvious parameters that widely affect ANNs, a list would include: the reverberant ANN architecture (e.g. classification and autoencoding can be realized in different parts of the network), the size of the bottleneck in autoencoders, the number of reverberations and the images and logits similarity measures.

We measured the impact of reverberation on adversarial perturbations thanks to the implementation of an adversarial examples detection-based reactive defense. Another interesting scenario to explore would be the embedding of reverberation in a robustness-based defense. This way, one could measure whether a reverberant DNN under attack better classify adversarial examples than its regular counterpart. If we take a careful look at the Figures 1 and 2, we can observe that adversarial examples tend to change into something else that resembles a prototypical image of another class under the effect of reverberation. It would be interesting to evaluate if adversarial examples converge towards their true class. As the case may be, a built-in reverberation-based defense could be simply based on multiple reverberations of the inputs before classification.

7. Conclusion

We introduced reverberation in the study of adversarial perturbations and proposed a first experimental analysis of its impact through an adversarial detection scenario. We also proposed an original reverberant hybrid DNN architecture that combines a classifier and an autoencoder. Our results show that reverberation has a different impact on clean and adversarial images. There are a number of unanswered questions regarding the reasons of this phenomenon, and a research effort is still to be done to fully benefit from its implementation. Further research on reverberation may advance our understanding of adversarial perturbations. Beyond this topic, reverberation could bring solutions in other machine learning tasks. The most obvious are anomaly and novelty detection tasks. Reverberation has also a natural potential for data augmentation.

References

[1] Jonathan Aigrain and Marcin Detyniecki. Detecting adversarial examples and other misclassifications in neural networks by introspection. *arXiv preprint arXiv:1905.09186*, 2019. 3

[2] Bernard Ans and Stéphane Rousset. Avoiding catastrophic forgetting by coupling two reverberating neural networks. *Comptes Rendus de l'Académie des Sciences-Series III-Sciences de la Vie*, 320(12):989–997, 1997. 2, 3

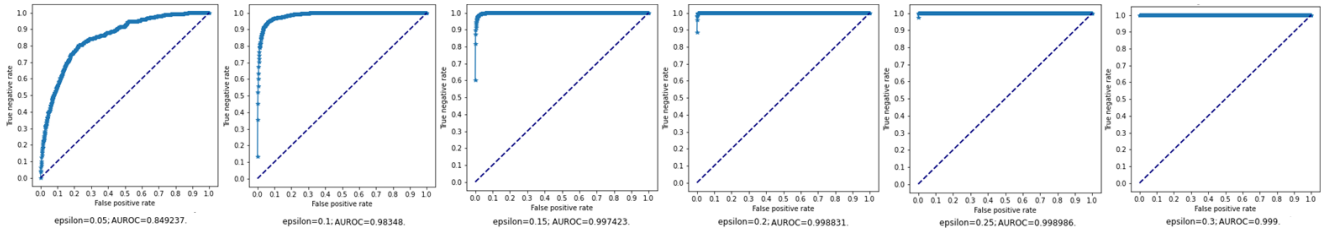
[3] Nicholas Carlini, Guy Katz, Clark Barrett, and David L Dill. Provably minimally-distorted adversarial examples. *arXiv preprint arXiv:1709.10207*, 2017. 1

[4] Nicholas Carlini and David Wagner. Adversarial examples are not easily detected: Bypassing ten detection methods. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 3–14, 2017. 2, 4

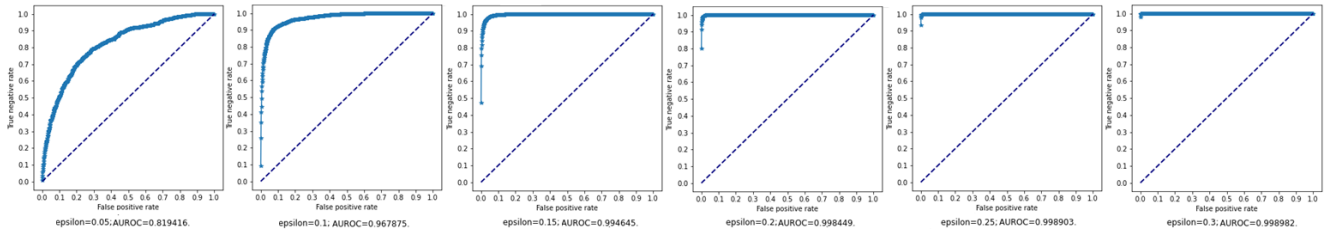
[5] Nicholas Carlini and David Wagner. Magnet and" efficient defenses against adversarial attacks" are not robust to adversarial examples. *arXiv preprint arXiv:1711.08478*, 2017. 2

	MNIST	Fashion-MNIST	CIFAR-10	
	LeNet	Basic CNN	VGG-13	DenseNet-121
FGSM	0.999 ($\epsilon = 0.3$)	0.9869 ($\epsilon = 0.2$)	0.7286 ($\epsilon = 8/255$)	0.7449 ($\epsilon = 8/255$)
L_∞-PGD	0.9989 ($\epsilon = 0.3$)	0.9725 ($\epsilon = 0.2$)	0.7015 ($\epsilon = 8/255$)	0.7224 ($\epsilon = 8/255$)
L_2-C&W	0.7712	0.7308	0.6518	0.6625

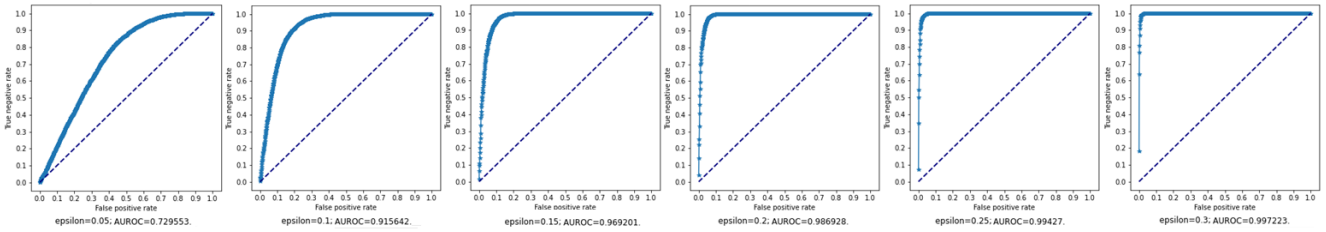
Table 2: AUROC (%) scores for adversarial examples detection. Only clean data have been used to train the ANNs and to estimate the thresholds. The detection is based on the MSE, calculated between original inputs and outputs resulting from the last reverberation, or between logits resulting from the first and last inferences.



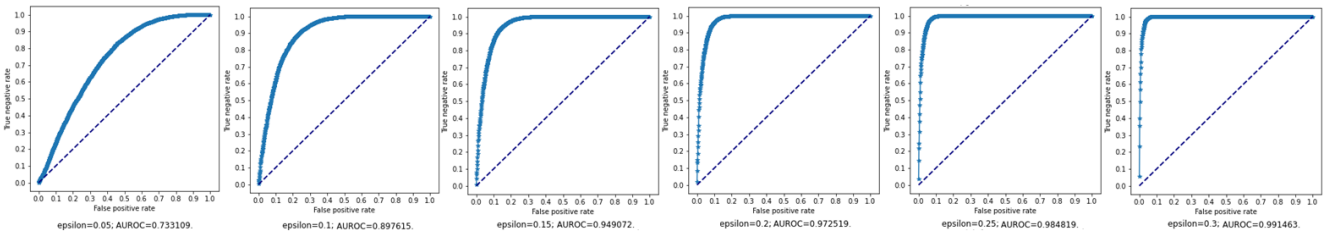
(a) MNIST, FGSM



(b) MNIST, L_∞-PGD



(c) Fashion-MNIST, FGSM



(d) Fashion-MNIST, L_∞-PGD

Figure 6: ROC curves for different perturbation sizes ϵ. The higher the perturbation, the lower the detection performance.

[6] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017. 5

[7] Ruediger Ehlers. Formal verification of piece-wise linear feed-forward neural networks. In *International Symposium on Automated Technology for Verification and Analysis*, pages 269–286. Springer, 2017. 1

[8] Zhitao Gong, Wenlu Wang, and Wei-Shinn Ku. Adversarial and clean data are not twins. *arXiv preprint arXiv:1704.04960*, 2017. 2

[9] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *Inter-*

national Conference on Learning Representations, 2015. 1, 3, 5

[10] Kathrin Grosse, Praveen Manoharan, Nicolas Papernot, Michael Backes, and Patrick McDaniel. On the (statistical) detection of adversarial examples. *arXiv preprint arXiv:1702.06280*, 2017. 2

[11] Shixiang Gu and Luca Rigazio. Towards deep neural network architectures robust to adversarial examples. *arXiv preprint arXiv:1412.5068*, 2014. 1

[12] Dan Hendrycks and Kevin Gimpel. Early methods for detecting adversarial images. *International Conference on Learning Representations (Workshop Track)*, pages 1–9, 2017. 2, 3

[13] Shengyuan Hu, Tao Yu, Chuan Guo, Wei-Lun Chao, and Kilian Q Weinberger. A new defense against adversarial images: Turning a weakness into a strength. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[14] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5

[15] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. Cifar-10 (canadian institute for advanced research). *URL http://www. cs. toronto. edu/kriz/cifar. html*, 5, 2010. 4, 5

[16] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236*, 2016. 3, 5

[17] Yann LeCun, Bernhard Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne Hubbard, and Lawrence D Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989. 5

[18] Yann LeCun, C Cortes, and C Burges. The mnist dataset of handwritten digits. 1998. 3, 5

[19] Xin Li and Fuxin Li. Adversarial examples detection in deep networks with convolutional filter statistics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5764–5772, 2017. 2

[20] Bin Liang, Hongcheng Li, Miaoqiang Su, Xirong Li, Wenchang Shi, and XiaoFeng Wang. Detecting adversarial image examples in deep neural networks with adaptive noise reduction. *IEEE Transactions on Dependable and Secure Computing*, 2018. 2

[21] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. 1

[22] Jan Hendrik Metzen, Tim Genewein, Volker Fischer, and Bastian Bischoff. On detecting adversarial perturbations. *arXiv preprint arXiv:1702.04267*, 2017. 2

[23] Kartik Mundra, Rahul Modpur, Arpan Chattopadhyay, and Indra Narayan Kar. Adversarial image detection in cyber-physical systems. In *Proceedings of the 1st ACM Workshop on Autonomous and Intelligent Mobile Systems*, pages 1–5, 2020. 3

[24] Rafael Pinot, Laurent Meunier, Alexandre Araujo, Hisashi Kashima, Florian Yger, Cedric Gouy-Pailler, and Jamal Atif. Theoretical evidence for adversarial robustness through randomization. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[25] Kevin Roth, Yannic Kilcher, and Thomas Hofmann. The odds are odd: A statistical test for detecting adversarial examples. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5498–5507. PMLR, 09–15 Jun 2019. 2, 3

[26] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 5

[27] Kirthi Shankar Sivamani, Rajeev Sahay, and Aly El Gamal. Non-intrusive detection of adversarial deep learning attacks via observer networks. *IEEE Letters of the Computer Society*, 3(1):25–28, 2020. 2

[28] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations*, 2014. 1

[29] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 3, 5

[30] Kai Y. Xiao, Vincent Tjeng, Nur Muhammad (Mahi) Shafiullah, and Aleksander Madry. Training for faster adversarial robustness verification via inducing reLU stability. In *International Conference on Learning Representations*, 2019. 1

[31] Weilin Xu, David Evans, and Yanjun Qi. Feature squeezing: Detecting adversarial examples in deep neural networks. In *25th Annual Network and Distributed System Security Symposium NDSS*. The Internet Society, 2018. 3

[32] Huan Zhang, Tsui-Wei Weng, Pin-Yu Chen, Cho-Jui Hsieh, and Luca Daniel. Efficient neural network robustness certification with general activation functions. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. 1