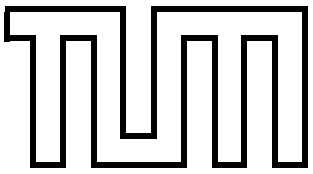


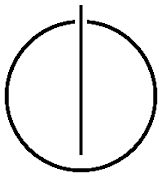
DEPARTMENT OF INFORMATICS

TECHNICAL UNIVERSITY OF MUNICH

Bachelor's Thesis in Informatics

A Comprehensive Benchmark for Defenses Against Black-Box Adversarial Attacks

Chantal Marie Pellegrini



DEPARTMENT OF INFORMATICS

TECHNICAL UNIVERSITY OF MUNICH

Bachelor's Thesis in Informatics

A Comprehensive Benchmark for Defenses Against Black-Box Adversarial Attacks

Ein umfassender Benchmark für Abwehrmechanismen gegen adversarial Black-Box-Angriffe

Author:	Chantal Marie Pellegrini
Supervisor:	Prof. Dr.-Ing. habil. Alois Christian Knoll
Assessor:	Thomas Brunner
Submission Date:	16.09.2019

I confirm that this bachelor's thesis in informatics is my own work and I have documented all sources and material used.

Munich, 16.09.2019 Chantal Marie Pellegrini

Abstract

Although deep neural networks are state of the art in the field of image classification, they are vulnerable to adversarial examples [1]. These are generated through attacks that slightly perturb the input to cause misclassification [2]. Addressing this risk, researchers propose various defense mechanisms to increase network robustness.
This thesis explores several relevant defense approaches, namely DefenseGAN, Ensemble Adversarial Training and Mahalanobis Detector and RCE + K-Density Detector [3–6]. It compares them to each other in a comprehensive study. It is a known issue that evaluating defenses under attack is difficult and often insufficiently done [22]. In particular, defense validations often use only specific white-box attacks, and additionally papers differ in their way of evaluation. This study evaluates all defenses in an unified environment using an iterative decision-based attack, an improved version of Boundary Attack [7, 8]. It is model-agnostic and realistic due to the limited information the attacker needs. All experiments are carried out on the MNIST [9] dataset and, if applicable, additionally on the Cifar10 dataset [10]. In this way the evaluated defenses are made comparable.
The results suggest defending against this type of attack is difficult, as only one of the four defenses leads to visually confusing adversarial examples. Moreover, only two did lead to more visible perturbations on MNIST, whereas on Cifar10 the changes are never visible. The study further reveals that it is insufficient to assess robustness solely based on distance metrics. Rather the visibility of the perturbation should be taken into account, as those approaches indicate different judgments of the defenses. A defense should only be considered effective if the perturbation is clearly visible.

Contents

1 Introduction and Theory

1.1 Introduction

In image classification no other method achieves similar performance as Deep Learning [11]. Nevertheless, Deep Learning can be vulnerable to adversarial attacks, methods designed to add small perturbations to images aiming to fool the network [1]. This can cause serious dangers in safety critical applications like autonomous driving. One striking example is the real-world adversarial attack recently performed against Tesla's lane recognition system. It was tricked into seeing lane markers by applying small patches on the street. What is remarkable is that a human would not have interpreted them as lane markers and thus would not have made a mistake. Even though Tesla shortly after declared this gap as fixed, it is an illustrative example of what is possible through such attacks [12].
There is plenty of active research in the field of adversarial examples, and while many papers are published conducting new attacks, many others counter with defense mechanisms claiming robustness improvements of Deep Learning models under attack.
Many defenses are validated only against specific types of attacks, namely white-box and transfer-based attacks, whereas there is a lack in knowledge of how the defenses work against decision-based black-box attacks. Those attacks only need query access to the classifier to receive the final labels for given inputs. Needing no more than this information they are more relevant for real-world scenarios as an attacker mostly does not have access to other information about the classifier.
This thesis evaluates the robustness improvement some of the most important defense mechanisms gain against decision-based black-box attacks using a consistent test environment to make sure those defenses are comparable to each other.
For evaluation this work presents a study, in which different classifiers combined with defense approaches are attacked by the HopSkipJumpAttack [7], an iterative attack similar to BoundaryAttack [8]. The study compares the robustness of the classifiers with and without the defenses proposed in the respective papers. All experiments are carried out on the MNIST [9] as well as the Cifar10 dataset [10] where it is applicable. For tuning attack parameters where necessary, the well-known FGSM attack [2] is used. The defenses that are evaluated are Defense-GAN [3], Ensemble Adversarial Training [4], a detector based on the Mahalanobis distance [5] and the RCE + K-density detector approach [6].

Structure of the work The following section provides the necessary background knowledge about adversarial examples in general as well as current attack and defense approaches and types. The related work section presents other studies carried out that investigate the effectiveness of defense mechanisms. Part two describes the study conducted in this work in detail and presents the achieved results. The last section concludes taking into consideration the state of the art research as well as the results achieved in this thesis and points out possible future work.

1.2 Background Knowledge

This section informs about the definition of adversarial examples, attack methods for their generation as well as proposed defense types. It further defines essential terms used in this work.

1.2.1 Adversarial Examples

Szegedy et al. first introduced adversarial examples [1]. They discovered that neural networks are vulnerable to small perturbations applied to the input. These perturbations lead to misclassification although the true label of the input, resulting from human perception, remains the same. Frequently humans hardly notice that the perturbation in the image like in the example shown in Figure 1.1. Previous contributions developed several different ways to generate such adversarial examples, so-called adversarial attacks.

Figure 1.1: Adversarial example generated during the study on MNIST. On the left is the original picture, classified as '7', in the middle the adversarial example, classified as '9' and on the right the difference between them.

The quality of adversarial examples is usually defined by some distance metric, quantifying the difference to the original image. The three most popular distance metrics are the L_{inf}, the L_2-norm and the L_0-norm [17]. This work uses the mean squared distance (MSE), which is calculated by squaring and normalizing the L_2 norm. Thus the distance between the original image and the generated adversarial example is defined as $\|x_{adv} - x\|_2{}^2$ normalized by image size. It is aimed to reduce this value during the attack.

1.2.2 Adversarial Attacks

The term adversarial attack refers to algorithms generating adversarial examples to fool some target machine learner, mostly a neural network. This section provides an overview of the existing types of adversarial attacks and presents some specific attacks relevant to this work.

White-box attacks In white-box attacks, the attacker has access to all information about the model, e.g. the network structure, the training procedure, sometimes including the training data and the trained weights. Those attacks mostly exploit the information

about the gradient of the loss for the given input, but not every white box adversary uses the same information about the target network. [8, 13]

Fast Gradient Sign Method Goodfellow et al. [2] present a simple yet effective method of generating adversarial examples. The Fast Gradient Sign Method, also known as FGSM, linearizes the cost function around the current network parameter values θ and thus applies the following perturbation to the image:

$$\eta = \epsilon \, sign(\nabla_x J(\theta, x, y))$$

The method aims for the smallest epsilon for which the perturbation still results in an adversarial example. It is based on their hypothesis that neural networks have a linear enough character to be susceptible to linear adversarial perturbations.

Black-box attacks Black-box attacks do not require knowledge about the target model. In some cases, the attacker may have limited knowledge about specific properties like the architecture, but never has access to the model parameters or gradients. Instead, black-box attacks either rely on transferability or get information solely by querying the attacked model. [13]

Attacks using transferability A lot of proposed black-box attacks are based on the transferability of adversarial examples. Transferability describes the characteristic that adversarial examples crafted by attacking one model often also fool other models. These models can even have another architecture, different hyperparameters, or train on different training data [1, 2]. By using this property, black-box attacks can be designed that apply some white-box attack on a substitute model and use the resulting adversarial examples for fooling the actual target model. In this way, no detailed information about the target model is necessary.

Boundary Attack Instead of using a substitute model like in transfer-based attacks, Boundary Attack [8] is solely decision-based. This means it only needs the final labels the target model predicts when querying it with some input to attack a classifier.
Boundary Attack gets initialized with a high perturbation that fulfills the adversarial criterion in general misclassification. In the following steps, the attack iteratively tries to reduce the distance to the target image by performing a random walk along the boundary of the non-adversarial region. In detail, each step of this walk consists of two parts. First, a step in a random direction is made, chosen through drawing from an i.i.d. Gaussian distribution. This direction is rescaled, clipped, and projected on a sphere around the target image. Second, a small step towards the target image is taken. At the end of the step, the new image should lie closer to the target image than the prior one while still being adversarial.

HopSkipJumpAttack The HopSkipJumpAttack [7] is another iterative decision-based attack and is used in this work. It follows a similar approach as the Boundary Attack but incorporates some changes leading to higher query efficiency. It is initialized as well at an adversarial starting image and then tries to minimize the distance to the target image with iterative steps. Each step consists of the following parts:

- the current image is pushed towards the boundary with a binary search
- with those results, a gradient direction estimation is performed
- the update in the direction of the gradient estimate is initialized and reduced until the perturbation is successful

What makes this attack more query-efficient and faster than Boundary Attack is mainly that the estimate of the gradient uses all tried perturbations. In comparison, Boundary Attack discards all drawn perturbations that are not adversarial. In Figure 1.2 the attack is visualized.

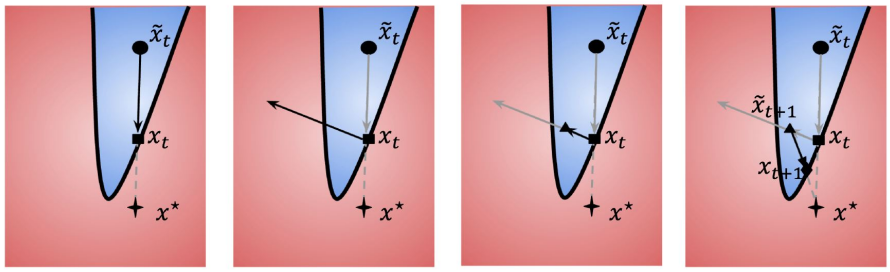


Figure 1.2: Visualisation of the HopSkipJumpAttack. (a) Perform a binary search to find the boundary, and then update $\tilde{x}_t \rightarrow x_t$. (b) Estimate the gradient at the boundary point x_t. (c) Geometric progression and then update $x_t \rightarrow \tilde{x}_{t+1}$. (d) Perform a binary search, and then update $\tilde{x}_{t+1} \rightarrow x_{t+1}$. [7]

Targeted vs. untargeted attacks All adversarial attacks pursue the goal to cause a misclassification. In the untargeted scenario, this alone is sufficient to claim the attack as successful. Targeted attacks, on the other hand, aim a classification of a specific label, different from the true class of the input. The attack used in this thesis, HopSkipJumpAttack [7], is capable of both objectives. This thesis works with the untargeted version of the attack.

1.2.3 Defenses against Adversarial Attacks

Parallel to the development of adversarial attacks presented in the previous section, there have been published numerous papers presenting defense approaches. This section gives an overview of the developments in this area.

Classifier robustness When designing a defense, the objective is to increase the robustness of the classifier. Robustness describes the ability to resist against adversarial attacks and can be achieved by either classifying correctly despite the attack or detect the attack in order to deny to classify this particular input. In literature, robustness often is measured as the fooling rate at a specific perturbation size [8]. The perturbation size can be measured with different distance metrics like L_{inf}-norm, L_2-norm or L_0-norm.

Visibility of adversarial perturbations The assessment of the robustness of a classifier with the measures mentioned above is not sufficiently expressive on its own. Instead, the visibility of the perturbation should be taken into account as well. Figure 1.3 shows two perturbed images with similar perturbation values but a recognizable difference concerning visibility. The same model classifies the left image as a nine and the right as a three. The nine is easily recognizable, whereas one cannot notice the three. Thus the right image is a much better adversarial example, even though they have almost the same perturbation size.
As there is no quantitative measure for this visibility, the results in the conducted study base the measure of robustness on the achieved minimal perturbation as MSE. Additionally, qualitative examples from the generated adversarial images are shown to address this gap of expressiveness.

Figure 1.3: Two adversarial examples with similar perturbation sizes measured as mean squared distance (MSE). On the left: 0.0034 and on the right: 0.0033

Types of defense strategies Adversarial defenses can roughly be divided into three categories:

- **Modifying data:** Modifying the data includes modifying the training data like for adversarial training [1, 2, 4] as well as modifying the input during testing, e.g. by using Data Compression or Randomization.
- **Modifying the network:** These defenses build a network in a way that they claim to be more robust, e.g., through adding layers or changing the loss or activation functions.

- **Usage of auxiliary tools:** Here some additional tool like an external network is used for defending against attacks. They usually can be applied as an add-on to already trained classification models.

The latter two cases can further be divided into defenses aiming for correct classification and detection only defenses. Detection only defenses will reject any input that they detect as adversarial using some additional classifier or a metric calculating a score that represents the likelihood of the input being adversarial [13, 14].

Defenses useful against specific attacks Against several attacking methods useful defenses have already been found [8]:

- **Gradient-based attacks:** Against white-box attacks using the gradients, gradient masking can be used. This can be done by, e.g. adding non-differentiable elements in the network.
- **Score-based attacks:** Attacks using prediction scores like class probabilities can be defended by adding e.g. dropout or use training methods that mask gradient estimates. In general, they can be defended by stochastic defenses as the resulting stochastic gradient will lead to a wrong estimate of the actual gradient if the attack only uses one sample. [8, 15]
- **Transfer-based attacks:** For transfer-based attacks, it is often sufficient to train a robust classifier on adversarial samples like with Ensemble Adversarial Training [4].

1.3 Related Work

There exists extensive literature on adversarial attacks and defenses building the foundation for this thesis. As the previous parts already present the work about attacks and defenses which are most relevant for this thesis, this section concentrates on other studies evaluating the robustness of models using defense mechanisms. In particular, it presents two studies comparable to the one in this thesis. Both gain similar results suggesting most defenses are not effective enough when attacked by strong adversaries.

Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods Carlini and Wagner [16] conduct a study bypassing ten detection methods. They use three different threat models:

- **Zero-knowledge attack:** In this attack setting, a strong generic attack, not knowing anything about the defense mechanism, is utilized. The authors use the C&W attack [17]. The zero-knowledge attack is the weakest threat model as the attacker is not aware of the defense.

- **Perfect knowledge attack:** Here the attack used is a perfect knowledge white-box attack tailored on the detector it is attacking. For this, they present a new loss function for building an attack fooling the model and the detector simultaneously. The loss function is shaped to fit the used defense, which makes this the strongest attack model.
- **Limited knowledge black-box attack:** In this threat model, the adversary knows the used detector but not its parameters. The attack makes use of transferability by training a substitute model like the original model but with a different training set. If the detector is a separate model, the attack can only access the classifier. If not, the attack generates adversarial examples on both, the unsecured and the secured classifier, and the results are compared.

In this research, six of the ten defenses are already a lot less effective under the zero-knowledge attack. The perfect knowledge attack can break all defenses and also the limited access black-box attack does decrease model accuracies significantly.
One of the detectors studied by them is Kernel Density Estimation [18]. RCE + K-Density [6], one defense investigated in this thesis, builds on top of the Kernel Density Estimation by using it as a detection mechanism. Carlini and Wagner come to the following results. Against the zero-knowledge attack, the K-density approach can defend successfully on MNIST, but not on Cifar10. On Cifar10, 80% of the adversarial images even have a higher likelihood score than the original ones. With the perfect and the limited knowledge attack the defense can be broken on MNIST as well.
Another of their conclusions relevant to us is that MNIST results may not hold on Cifar10. For that reason, the study in this thesis evaluates the defenses on both datasets where applicable.

Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples In a more recent work Athalye et al. carry out a study on all ICLR 2018 defenses [15]. They define the term obfuscated gradient as cases where the defense design itself masks the gradients. From the seven ICLR attacks relying on this property, they were able to break six entirely and one partially. For attacking they propose Backward Pass Differentiable Approximation. This technique computes the backward pass using a differentiable approximation after computing the forward pass. In this way, they can approximate the gradients.
They investigate Adversarial Training, the base of Ensemble Adversarial Training [4], and can reduce the accuracy by using an optimization-based attack, that relies on the gradients. Thus they conclude that Adversarial Training does not use obfuscated gradients. Even though they succeed with some probability, they are not able to falsify the claims of the authors.
They as well include Defense-GAN [3] in their study and evaluate it on MNIST. They can generate adversarial examples using the mentioned attack, but only with a 45% success rate.

2 Conducted Study

The study aims to evaluate the robustness of common adversarial defenses against adversarial black-box attacks, tested with a decision-based iterative black-box attack. It pursues two main goals with the first being to enable a comparison of different defenses by testing them all in the same environment. The results may help to get an idea of the achievable robustness for neural networks against such an attack with current approaches.
The following section describes details about the study, including the procedure and the study objects. In the course of this, it also gives detailed explanations of the four defense approaches tested.

2.1 Study Objects

The Study Objects comprise the datasets for evaluation MNIST [9] and Cifar10 [10], the used HopSkipJumpAttack [7], that is closely related to Boundary Attack [8] and presents the studied defenses [3–6] in detail.

2.1.1 Datasets

The evaluation employs the two standard datasets MNIST [9] and CIFAR10 [10] as those are widely used for evaluation of image classification approaches in general and as well as of adversarial attacks and defenses.
All defenses are evaluated using MNIST as the base dataset, enabling a fair comparison. Furthermore, Mahalanobis Detector and RCE + K-Density Detector are applied to Cifar10 classifiers. Defense-GAN and Ensemble Adversarial Training are not implemented for Cifar10, because the implementation would require a significant amount of further development. It would have been necessary to choose many parameters and adapt the training procedures as well as the network architecture. The functionality of those defenses relies on the training as well as the used network. Therefore any misapplication could have harmed the validity of the study.

2.1.2 Used Attack

For attacking, the study uses an iterative decision-based black-box attack closely related to the Boundary Attack [8]. The reason for using this attack family is that those kinds of attacks are entirely independent of the used model. They only require the final prediction labels. This property is essential for the study as, for evaluation, the same

attack needs to run against several different models paired with different defenses. Moreover, Boundary Attack shows good results among the state of the art black-box attacks.
As HopSkipJumpAttack is shown to reach competitive performance while being more query-efficient, it is better suited for this thesis as some defenses require a relatively long time processing a query. Since iterative attacks need to send many queries to the classifier, evaluation is done using this variant of decision-based attacks.

2.1.3 Evaluated Defense Mechanisms

The study evaluates the following four different defenses that represent a variety of some of the most important defense approaches. All of the defenses were presented in important conferences or published in peer-reviewed journals and are referenced quite frequently. Furthermore, all of the papers provide their source code, so it was not necessary to decide on implementation strategies for the defenses.[1]

Defense-GAN Defense-GAN [3] is an add-on network that aims to remove the perturbations from the input to and thus generates an image that is similar to the clean data. This output can then be fed to the classifier. In detail, Defense-GAN performs the following steps. The steps are also visualized in Figure 2.1.

- Beforehand a generative adversarial network G is trained using a generator and a discriminator model. Its objective is to generate samples modeling the training data by projecting every input onto the distribution of the unperturbed images. Consequently, given a clean input the output should stay very similar, given an adversarial input the adversarial noise should decrease.

- At testing time, given the input $x, z*$ is chosen by approximating the minimization problem $\min_z ||G(z) - x||_2^2$. For this a fixed number of gradient descent steps is run using R random initializations $z_0^{(1)}$ to $z_R^{(1)}$.

- The output $G(z*)$ is fed to the classifier that then predicts the label $\hat{y}$.

This defense neither needs knowledge of the attack nor of the used classifier. Therefore it can be applied to any trained classifier.

[1] The GitHub repositories used can be found here:
Defense-GAN: https://github.com/kabkabm/defensegan (last visited: 21/08/2019)
Ensemble Adv. Training: https://github.com/ftramer/ensemble-adv-training (last visited: 21/08/2019)
Mahalanobis: https://github.com/pokaxpoka/deep_Mahalanobis_detector (last visited: 21/08/2019)
RCE: https://github.com/P2333/Reverse-Cross-Entropy (last visited: 21/08/2019)
K-Density: https://github.com/rfeinman/detecting-adversarial-samples (last visited: 07/03/2019)

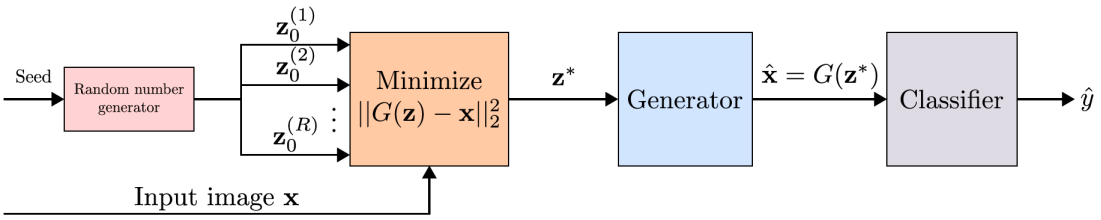


Figure 2.1: Overview of the Defense-GAN algorithm [3]

Ensemble Adversarial Training Adversarial Training approaches belong to the defenses modifying the training data. They aim to increase the model robustness by training not only on the regular training set but additionally on adversarial examples crafted for the trained model. The training should result in a model that can predict the correct label with a higher probability, especially for perturbed samples [1, 2]. This concept is also the basis for Ensemble Adversarial Training [4]. Here the adversarial examples used for training are transferred from several other models instead of being crafted directly for the target model. Transferring the adversarial examples increases the diversity of the perturbations in the training set. The gained diversity should avoid one-sided robustness against only a specific type of adversarial examples, which is a common problem of adversarial training methods.

Detector using Mahalanobis distance score Lee et al. [5] propose a framework to detect adversarial examples during testing time. Their detector can be appended to every softmax trained neural network without retraining being necessary, so it as well belongs to the last category of defenses, the auxiliary tools.

- All classes are described with class-conditional Gaussian distributions using the empirical class means and covariance of the training data.
- Using this information, the paper defines a new confidence score based on the Mahalanobis distance between the input sample and the closest class-conditional distribution. In general, the Mahalanobis distance measures the distance between two points in a multivariate space related to some central point that can be seen as the overall mean of the data [19].
- This score is extracted from features of several layers of the network, and afterwards, logistic regression is used on the obtained scores from all layers to calculate a final score.
- An input gets rejected if the score falls below some threshold.

RCE + K-density detector The last defense consists of two parts. It modifies the training process and additionally calculates a k-density score at test time to detect adversarial examples [6].

- **RCE training:** Instead of using cross-entropy loss, the new training objective is to minimize the reverse cross-entropy(RCE). RCE training is supposed to improve the classifier robustness and lead to features that can be better used to distinguish between normal and adversarial examples. The RCE is defined as $-R_y^T \ logF(x)$, where F is the classifier and R_y is the reverse label vector for the training data (x, y). The y^{th} element of this vector is zero and all other are equal to $\frac{1}{L-1}$. This part of the defense modifies the network by changing the training objective.
- **K-density detector:** For each input the kernel density is calculated after the forward pass through the network using the last hidden layer features. Kernel density measures how far some data point is from the submanifold of the predicted class in this case modeled by a Gaussian distribution [18]. This second part applies at the testing time similar to the Mahalanobis distance detector.

With those defenses, the aimed variety of approaches is covered. Ensemble adversarial training represents the branch of adversarial training based approaches, Defense-Gan the ones that modify the input itself in this case as an add-on network. Mahalanobis Detector and RCE + K-density are concentrating mainly on the detection of adversarial examples. RCE + K-density furthermore introduces a new optimization method during training. All of the defenses can realistically help against black-box attacks as they do not defend with techniques aimed specifically against white-box attacks, e.g. gradient masking.

2.2 Study Design

The conducted study compares the defenses listed in the section above and evaluates how robust the defenses are in general. All defenses are studied using an uniform testing environment to provide comparability. In this, all defenses are attacked with the same attack on the same datasets.
Evaluation of every defense follows the subsequent procedure: The first step is the extraction of the underlying classifier used in the respective paper. This step is necessary to attack the models of all papers with and without defense. Then HopSkipJumpAttack [7] is run against both models with 100 images. Each attack uses an adversarial image from the dataset as a starting point. The attack is only allowed to send a limited number of queries to the classifier.
For evaluation, the perturbation size of the current adversarial example at each attack step is set into relation to the queries used at this point. With the data of all attack images, a mean graph showing the development of the perturbation given the query count is generated. The graphs obtained from the undefended and defended version of the model are compared to each other. Furthermore, the mean and the median of the

perturbations reached for all images is calculated. Also, the accuracies of the models, defended and undefended, are measured.

2.3 Details on the Procedure

In the following section details about the realization of the described study are provided. It outlines the parameters set for the attack as well as the defenses. Further, it describes the integration of the different defense types in the evaluation framework and the generation of the results needed for evaluation.

2.3.1 Attack Parameters

As stated earlier, this study uses the HopSkipJumpAttack [7], an iterative black-box attack working only with the prediction given by the attacked model. Foolbox [20] provides an implementation of this attack. In the course of this work, the implementation had to be extended. It was necessary to provide the possibility to track the number of queries made during the attack. This extension enables stopping the attack when a certain number of queries is reached. Furthermore, the original attack implementation can diverge. Indeed, this occurs when attacking a classifier defended by Defense-GAN. This is undesirable as the final perturbation would not represent the best reachable adversarial example. For that reason, the adapted attack used in the evaluation always works with the adversarial example with the smallest perturbation generated so far. The current example, which is the starting point for the following step, will only be updated if an example closer to the target image, is found. [2]
The maximum batch size is set to 64, describing the number of images the attack is allowed to get predictions for at once. In comparison to using a batch size of one, this only speeds up attacking but does not change the results. The attack terminates after requesting 100000 predictions per image, where each query in a batch counts as a single prediction. As a distance measure, the attack uses mean squared error. The attack starts with an adversarial image from the original dataset, instead of random noise, which decreases the time until convergence. For every defense, a subset of the data containing 2000 images is sampled under the condition that the corresponding model in the defended and undefended version predicted the correct label for the whole set. For each attacked image, the closest adversarial in this subset serves as the starting point. The closest adversarial is the image with the smallest L_2-norm difference to the original, which is still classified different than the original image.

2.3.2 Defense Usage

Preparation All papers on defense mechanisms are using different underlying prediction models to apply the defense. For the study, those models are extracted to compare

[2]The modified version of Foolbox can be found at https://github.com/ChantalMP/foolbox.

the defended and undefended version of the same model. There were some small changes in the defense code needed to make it possible to defend the model against a single query at attack time. In the original code, for some defenses, all computations of the defense are precomputed for the whole dataset. That is not possible for an iterative attack as the inputs the attack will query can not be known in advance. None of the changes modifies the logic of the defenses.

Training Data Training uses the MNIST and Cifar10 train data delivered by the PyTorch data loader, which already splits the data into a train and a test set. For each defense, the data is normalized with the same normalizations as in the belonging paper.

Choosing of Thresholds For two of the used defenses, Mahalanobis Detector [5] and RCE + K-Density, it is necessary to choose a threshold determining when an input gets rejected as adversarial. Establishing this threshold is done by running the FGSM attack [2] against the models and measure the scores obtained for those samples. Together with the same measurements for clean data, the value best separating the scores of clean and adversarial data can be estimated. The resulting thresholds are listed in table 2.1.

	mahalanobis	rce + K-Density
mnist	0.5	-105
cifar10	0.5	-38

Table 2.1: Thresholds used in detectors. All inputs with a score higher (for Mahalanobis Detector) or lower (for RCE + K-Density) than the threshold are classified as adversarial.

Defense Integration All defenses interfere either while training the model or while attacking it. In the case of interference during the attack, the defense can act during or after the prediction. All those cases need a different handling, which is described in the following:

- **Modifying the input:** Defense-Gan [3] is the only defense used modifying the given input. It is used during the prediction at attack time. Every input and adversarial image generated is given to the pre-trained gan, the attack then resumes with the reconstructed image and is unaware of this change.
- **Modified network training:** Ensemble Adversarial Training [4] and parts of RCE + K-Density [6] change the training procedure. In both cases, the training process provided in the code of the original papers is used to train the defended, and the undefended models. For Ensemble Adversarial Training, this means to train either only on the train set or additionally on the adversarial examples generated for

the ensemble models. The reverse cross-entropy training of RCE + K-Density can be replaced by a standard optimization using cross-entropy loss to turn off the defense.

- **Detectors:** Mahalanobis Detector [5], as well as RCE + K-density, are meant to detect adversarial inputs. They are used at attack time but after the prediction. To be able to reflect a detection, the predictions are modified as follows: an additional class is added to the classes of the dataset which represents the label 'adversarial'. The respective scores for this class are generated after computing the prediction on the given input and the scores obtained by the detection metric. If according to the threshold the image is detected as adversarial an one-hot encoding representing the additional adversarial class as prediction is returned. Otherwise, the probability of the adversarial class is set to be lower than all of the other class probabilities.

2.3.3 Result Generation

All experiments for one dataset use the same image set consisting of 100 randomly chosen pictures from the test data of the dataset. They fulfill the precondition that all used networks predict those pictures correctly with and without defense. While attacking the images, the modified attack saves the queries made and the perturbation of the current adversarial after each attack step. The allowed query amount of 100000 arose during the study after seeing the results for different values. Most of the attacks nearly converge until this point, so the improvement expected when allowing more queries is very small. Moreover, a significantly larger value of queries would not be realistic, taking into account the time needed to run the attack against 100 images.
When allowing 100000 queries, we get around 100 perturbation values per attacked image. The values in between are approximated using linear interpolation. Based on the interpolated values, the graphs in the result section are generated using the mean of the values of all images at a specific number of queries.
Additionally, the median and the mean perturbation value of the last iteration, are calculated. Also, some of the generated adversarial images are visualized. The prediction accuracies reached by the single models with and without defense are measured on the whole test set provided by the datasets. In all parts, the defended versions of the models are set into relation to the undefended versions to be able to describe and compare the improvements achieved by the defenses.

3 Results and Evaluation

After describing the study set up in the last part, the following presents the achieved results, their interpretation and discusses the limitations of the study.

3.1 Performance of the Evaluated Defenses

This section presents the results of the study. In Figure 3.1 and Figure 3.2 some qualitative results can be found. They demonstrate how the generated adversarial examples look when using the different defense mechanisms. Moreover, the following tables show the model accuracies and the perturbation values reached by the attack.

Model accuracies Table 3.1 lists the accuracies of all models with and without using the particular defense are computed to show if the usage of the defense harms the accuracies. No models or training processes are fine-tuned for achieving higher accuracies as for fulfilling the aim of this study this is not necessary. The accuracies are measured on the unperturbed test sets without including any adversarial examples.

	Accuracy clean model	Accuracy with defense
Ensemble Adversarial training	99.28 %	99.15 %
Defense-GAN	98.22 %	93.01 %
Mahalanobis	98.59 %	91.43 %
RCE + K-Density	99.54 %	RCE: 99.58 % RCE + K-Density: 91.27 %
Mahalanobis Cifar10	93.67	93.46
RCE + K-Density CIFAR10	91.56	RCE: 88.45 % RCE + K-Density: 70.42 %

Table 3.1: Accuracies of all used models with and without defense. Here clean model refers to the underlying classifier used in the respective paper without the usage of the defense. It can be seen the accuracy decreases for the detectors as well as Defense-GAN.

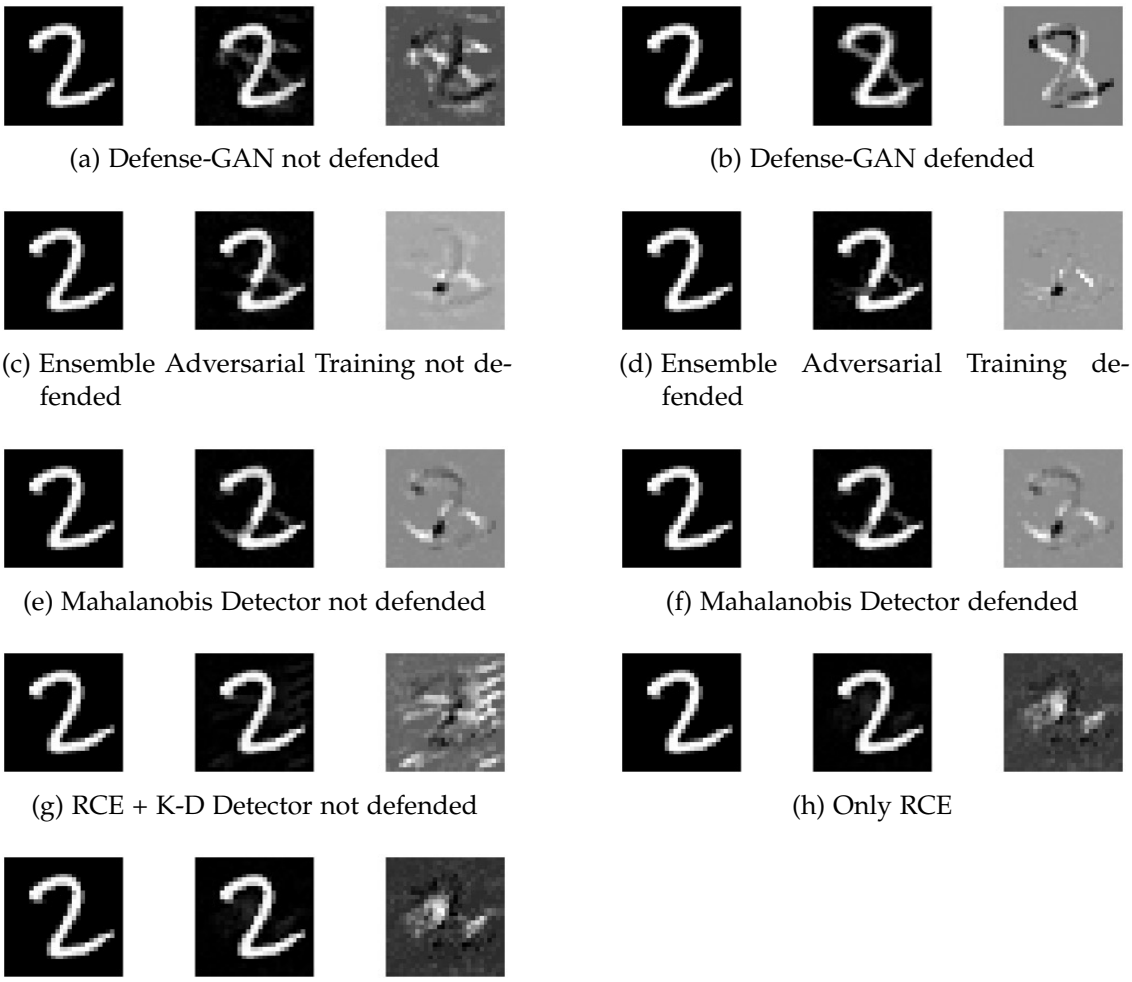
The measurements reveal a significant drop in accuracy for Mahalanobis detector as

well as the RCE + K-Density approach. Their usage results in accuracies of close to 90% what is a weak result for MNIST. For Cifar10 a similar drop can be seen using the RCE + K-Density detector. For Defense-GAN the accuracy with defense is also quite poor. Ensemble Adversarial Training on MNIST and Mahalanobis detector on Cifar10 have a stable accuracy.

Achieved perturbations In the following the development graphs for all defenses, plotting the reached adversarial perturbation against the attack queries, can be found. The graphs are shown from Figure 3.3 to Figure 3.8. Table 3.2 presents the reached mean and median perturbation in the last attack step as well as the change in robustness caused by the defense. A value above one is describing an increased perturbation value of the final adversarial example generated by the attack. Thus the defense can gain a robustness improvement. Accordingly, a change below one denotes a decreased perturbation. The values are the mean squared distances of the original to the perturbed image. To better understand those values in Figures 3.1 and 3.2 qualitative examples of the generated adversarial examples can be found.

	Mean		Median	
	achieved perturbation	change in robustness	achieved perturbation	change in robustness
Defense-GAN	0.01371	3.34	0.01221	3.36
Ensemble Adversarial Training	0.00376	1.30	0.00267	1.30
Mahalanobis Detector MNIST	0.00616	2.12	0.00298	1.36
RCE MNIST	0.00079	0.62	0.00064	0.61
RCE + K-Density MNIST	0.00081	0.63	0.00067	0.64
K-Density[1] MNIST	0.00081	1.02	0.00067	1.06
Mahalanobis Detector Cifar10	$1.001 \times 10^{-}5$	1.03	$5.760 \times 10^{-}6$	0.98
RCE Cifar10	$8.431 \times 10^{-}6$	1.26	$5.072 \times 10^{-}6$	1.54
RCE + K-Density Cifar10	$2.162 \times 10^{-}5$	3.24	$1.343 \times 10^{-}5$	4.08
K-Density[1] Cifar10	$2.162 \times 10^{-}5$	2.57	$1.343 \times 10^{-}5$	2.65

Table 3.2: Mean and median achieved perturbation against the defended models and the according change in robustness compared to the undefended model. The robustness change is calculated as the perturbation against the defended divided by the perturbation against the undefended model. E.g., the perturbation achieved against Ensemble Adversarial Training is 1.3 times higher when it is defended. Clearly Defense-GAN has the highest positive impact on MNIST and RCE + K-Density on Cifar10. Also all Cifar10 perturbations are substantially lower than MNIST.
[1] K-Density refers to the comparison between only RCE Training and RCE Training combined with K-Density Detector.



(a) Defense-GAN not defended

(b) Defense-GAN defended

(c) Ensemble Adversarial Training not defended

(d) Ensemble Adversarial Training defended

(e) Mahalanobis Detector not defended

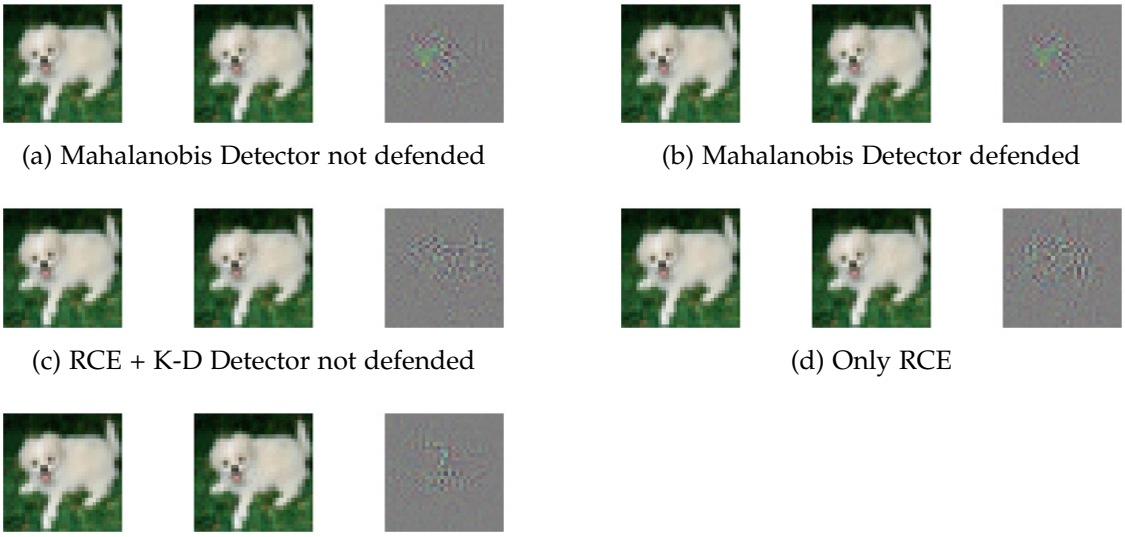
(f) Mahalanobis Detector defended

(g) RCE + K-D Detector not defended

(h) Only RCE

(i) RCE + K-D Detector defended

Figure 3.1: Qualitative results on MNIST. Each sub-figure shows the original image, the generated adversarial example and the difference between them from left to right. The original label is '2'. The adversarial examples a), b), g), h) and i) are classified as '8' and c), d), e) and f) as '3'.



(a) Mahalanobis Detector not defended

(b) Mahalanobis Detector defended

(c) RCE + K-D Detector not defended

(d) Only RCE

(e) RCE + K-D Detector defended

Figure 3.2: Qualitative results on Cifar10. Each sub-figure shows the original image, the generated adversarial example and the difference between them from left to right. The original label is 'dog'. The adversarial examples a), b), c) and e) get classified as 'frog', d) as 'cat'.

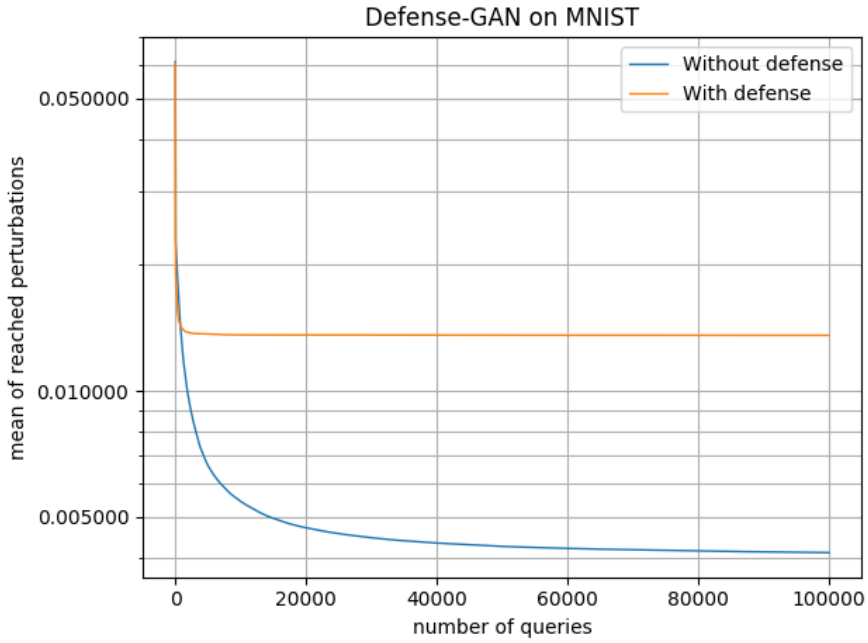


Figure 3.3: Reached perturbation against the model from the Defense-GAN paper with and without using a Defense-GAN.

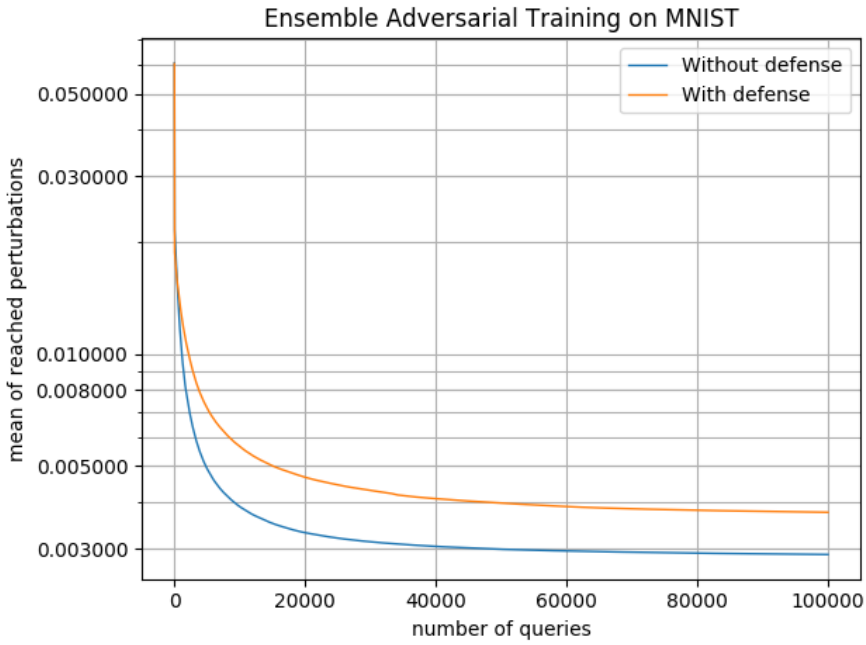


Figure 3.4: Reached perturbation against the ensemble adversarial training model trained normal or adversarial.

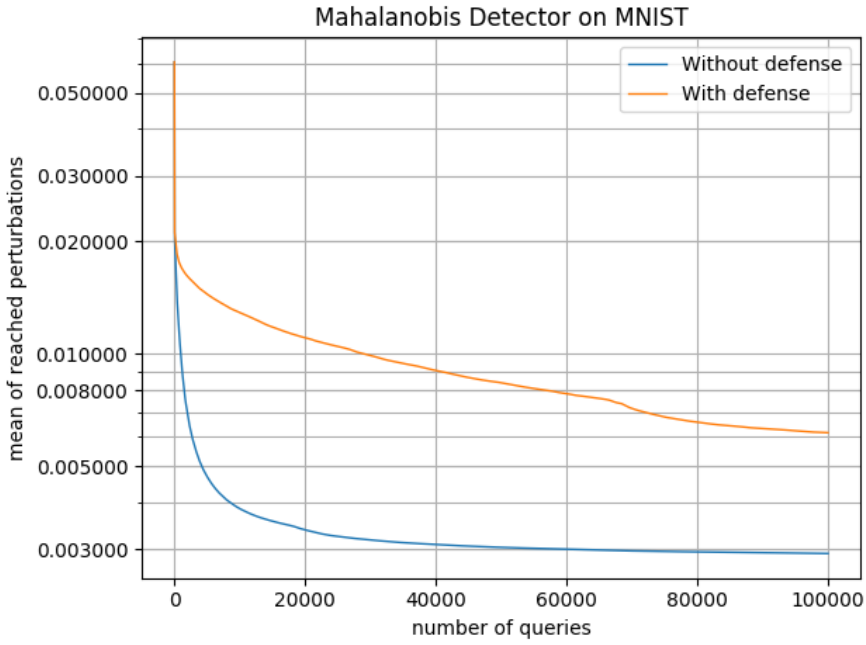


Figure 3.5: Reached perturbation against the Mahalanobis model with and without Detector on MNIST.

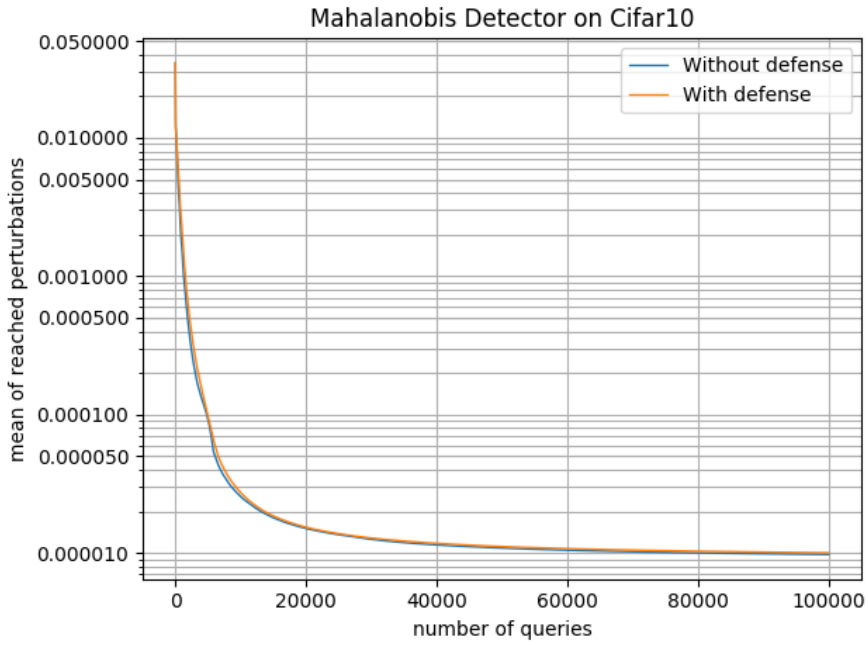


Figure 3.6: Reached perturbation against the Mahalanobis model with and without Detector on Cifar10.

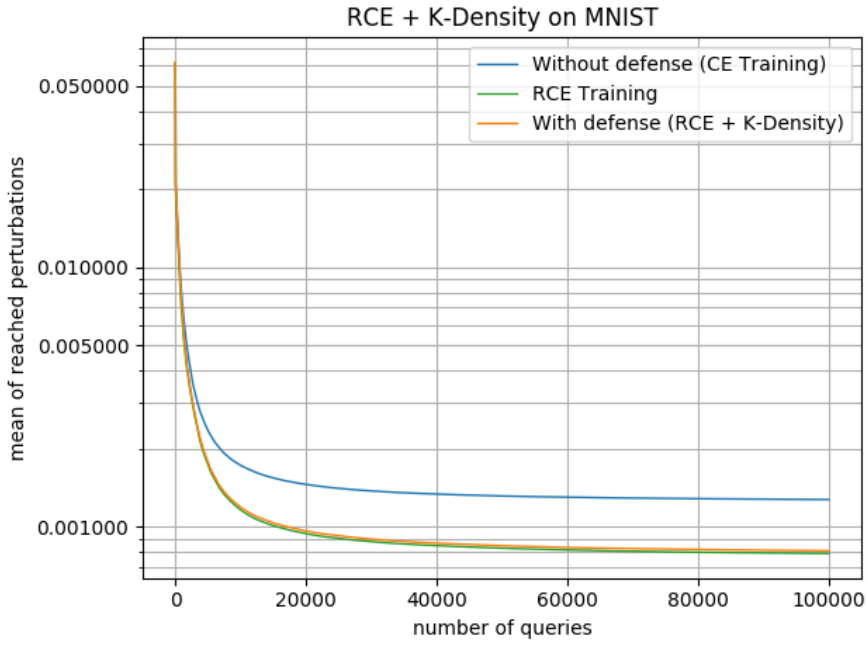


Figure 3.7: Reached perturbation against the model from the RCE+K-density paper trained with CE or RCE training or RCE training plus K-density detector on MNIST.

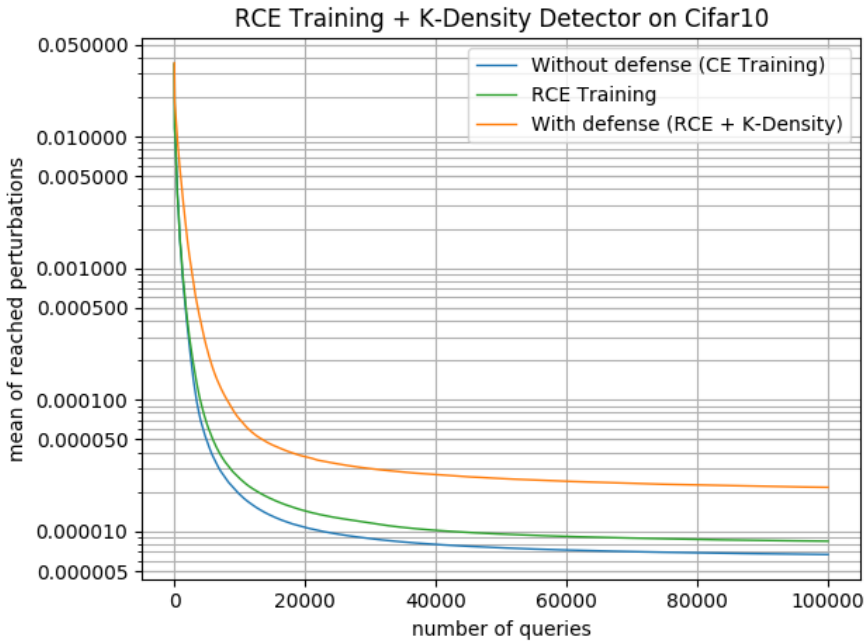


Figure 3.8: Reached perturbation against the model from the RCE+K-density paper trained with CE or RCE training or RCE training plus K-density detector on Cifar10.

Defense-GAN Defense-GAN is getting quite good results on MNIST. We can see the absolute perturbation, as well as the improvement, are both substantially higher as for the other compared defenses. Moreover, Figure 3.3 shows the graph is nearly constant, and thus, the perturbation is not expected to improve with more iterations (Fig. 3.3). The generated adversarial examples against Defense-GAN shown in Figure 3.1 do differ relevantly from the ones generated against the undefended model. A human may even classify the image as the adversarial class.

Ensemble Adversarial Training. Figure 3.4 shows that Ensemble Adversarial Training reaches only a slight improvement of the perturbation. Nevertheless, the perturbation noticeable changes how it looks. Using this defense on MNIST, the number of the adversarial class might be guessed as the missing parts are drawn as a sharp line on the original image (Fig. 3.1).

Mahalanobis Detector This detector does noticeably increase the minimal perturbation reached after 100000 queries on MNIST, but there is still a trend downwards noticeable in the graph shown in Figure 3.5, so with more iterations, the perturbation might improve. The examples for MNIST in Figure 3.1 do have an apparent perturbation, but the strength of the perturbation does not change relevantly from the undefended to the defended model. Moreover, it is hard to recognize the adversarial class. On Cifar10

the detector has almost no influence (see Figure 3.6), and thus the examples for Cifar10 in Figure 3.2 have no perceptible perturbation.

RCE + K-density Detector Figure 3.7 shows the experiments result in a lower final perturbation when using the defense on MNIST, which is a quite surprising result. The following conclusion is that the defense does not help against the attack type this study applies. The Reverse Cross-Entropy Training, the first part of the defense, causes the drop in robustness. The K-Density detector on the other hand causes minimal improvement. On Cifar10 both stages achieve some improvement like displayed in Figure 3.8. Here this defense succeeds Mahalanobis detector, the other defense working with Cifar10. For none of the datasets, the perturbation is more visible when using the defense in comparison to the undefended model (see Fig. 3.1 and 3.2).

3.2 Interpretation

The following part interprets the presented results focused on the comparison and the effectiveness of the defenses. It also elaborates on the differences between the used datasets.

3.2.1 Effectiveness of Defense Mechanisms in Comparison

Only one of the defenses can prevent the attack. Still, some of the others do have a notable influence on MNIST. By looking at the generated examples in Figure 3.1 and 3.2 it can be seen that the perturbations differ in their strength. A human can recognize the true label for all, except Defense-GAN, even though the networks fail to do so. Nevertheless, there are differences in the quality of the examples, especially on MNIST. For Defense-GAN, one might understand the true label, but it is not a straightforward choice. For Cifar10 the perturbation is not perceptible for both tested defenses.
One further distinctive feature of the defenses is their influence on the prediction accuracy on the unperturbed test set. Table 3.1 displays falling accuracies for Defense-GAN as well as the Mahalanobis and the K-Density Detector. The accuracy is relatively stable when using pure RCE training without K-Density detection. Concerning Defense-GAN this accuracy drop is caused when the generator maps clean images into an adversarial region. Even though the generator is trained to apply a minimal change to a clean image, this cannot be excluded as even small changes can lead to a different classification label.
For the detectors, this is an expected effect as some clean images mistakenly are classified as adversarial, which adds up with the actual misclassification. For both detectors, there exists no precise separation between the scores of clean and adversarial examples. Instead, the value ranges overlap. This issue does not apply for Mahalanobis on Cifar10. In this constellation, a good separation of the values is possible. The issue mentioned above leads to a trade-off between classification accuracy and detection rate

as a threshold closer to the clean score values will lead to better precision in detection but worsen the accuracy. Moving the threshold towards the adversarial score values will have the opposite effect. Its significant influence makes choosing the threshold critical for the detector and classifier performance. It is a definite disadvantage of those methods as for no other defense type the trade-off between robustness and accuracy is so present.

Defense-GAN Using MNIST Defense-GAN [3] stands out with a mean improvement of more than three times. Figure 3.9 shows some generated adversarial examples for MNIST using Defense-GAN. The perturbation is clearly recognizable, and the trend of the graph in Figure 3.3 suggests they would not improve with more iterations. It is undoubtedly hard to identify the true class of the image because the shape of the adversarial class is good to see, and thus, the images are visually confusing. An adversarial defense is successful if the perturbations do change the true label of the image, what Defense-GAN manages against the used attack on MNIST.
In order to exclude that the used attack can not handle the randomness of the defense, it can be shown in experiments that the results stay similar without random initializations. When always using the same initializations for the generator, the defense is a bit weakened, but the perturbation stays on a high level. This information reassures that for MNIST the defense is working to a certain degree. The generative network seems to be capable of representing the data distribution good enough and leads to the adversarial data being separable from the unperturbed data.

(a) original labels: ’1’, ’5’ and ’3’

(b) adversarial labels: ’6’, ’3’ and ’5’
perturbation size as MSE: 0.01999, 0.02166 and 0.01485

Figure 3.9: Original images and according generated adversarial examples above attacking Defense-GAN.

Ensemble Adversarial Training Ensemble Adversarial Training accomplished medium improvements in the numerical robustness values. Nevertheless the perturbation is easy to notice (see Figure 3.10). The main difference to Defense-GAN is that the perturbations are very easy to distinguish from the original parts of the picture as they have rather sharp edges, so for a human, they are less visually confusing. As the slope of the graph in Figure 3.4 after the 100000 iterations is quite small, no significant reduction of the perturbation is to be expected when running the attack for a longer time.
Evaluating Ensemble Adversarial Training on Cifar10 requires to train several Cifar10 classifiers adversarially. Using the training procedure provided for MNIST the accuracies of the resulting classifiers are as low as 60% which is not acceptable for Cifar10. Moreover using an untargeted attack against a model with a too low accuracy makes the attack a lot easier and thus leads to unrepresentative results. A longer training time and an adapted training procedure would be necessary as Cifar10 has more channels than MNIST and Adversarial Training needs to push the boundary away in every dimension to improve the network robustness successfully.

(a) original labels: ’4’, ’5’ and ’2’

(b) adversarial labels: ’9’, ’3’ and ’8’
perturbation size as MSE: 0.00272, 0.00335 and 0.00472

Figure 3.10: Original images and according generated adversarial examples above attacking a model trained using Ensemble Adversarial Training.

Mahalanobis Detector On MNIST, the use of the Mahalanobis Detector leads to a relevant higher perturbation value, especially regarding the perturbation mean. In the visual samples, on the other hand, the difference between the perturbed images generated against the undefended and defended model is only hardly recognizable. Interesting is also the discrepancy of mean and median improvement for this detector. Considering the median, the improvement is substantially lower indicates there are some outliers where the defense improves the robustness a lot, what leads to the high mean value, but for the majority of images the improvement is lower, explaining the lower median values.
The evaluation on Cifar10 shows a change close to one for mean and median meaning the robustness stays at the same level as for the undefended model. These results imply that in the scope of the conducted study Mahalanobis Detector has no positive effect used for a Cifar10 classifier.

RCE + K-Density Detector RCE + K-density [6] does not reach an improvement on MNIST, surprisingly instead the robustness is decreasing when using RCE instead of CE training. Figure 3.1 shows there is as well no visual improvement for this defense. Using the K-Density Detector does on its own provide a minimal improvement compared to the pure RCE training.
The original paper tests the defense under several different attacks including FGSM [2], BIM [21] and the C & W attack both as a white and as a black-box attack using

transferability. None of the used attacks is an iterative black-box attack like Boundary Attack. Furthermore, their experiments are carried out in the targeted attack setting while in this thesis, an untargeted attack setting is used, which makes attacking easier. As the last point, it has to be taken into account how detectors interfere in the study. If they detect an adversarial example, the classifier predicts the label 'adversarial' and thus the attacker is aware of being detected. With the iterative decision-based attack this label can be used to change the perturbation in the next iteration aiming not only at misclassification but also at being unnoticed. In this way, the attack can successfully circumvent the detector.
On Cifar10, both stages of the defense have a positive influence. Nevertheless, there is no perturbation visible in the generated images in Figure 3.2. These qualitative examples demonstrate that a defense mechanism needs to improve the robustness significantly more than RCE + K-Density Detector to work on Cifar10 effectively.

3.2.2 Differences on MNIST and Cifar10

To be able to compare the results for the two datasets first the general differences of these two datasets have to be clarified. MNIST is a straightforward task for state of the art classifiers, so a network needs to consider much less complex features than for Cifar10. It is considerably easier to reach a low perturbation in an adversarial example for Cifar10 than for MNIST due to the higher complexity. This leads, like expected, to significantly smaller perturbation values on Cifar10 in the undefended as well as the defended case.
Furthermore, the images of MNIST are divided into clearly separable black and white areas, whereas Cifar10 contains real low-resolution photos with a wide variety of colors and shapes. It follows every change of a pixel value is better observable for a human on MNIST than the same amount of change would be on CIFAR10.
Mahalanobis Detector, as well as RCE+K-Density Detector, were also tested on Cifar10. The results reveal two central observations. For one, they confirm that MNIST results can not reliably be adopted for Cifar10 as both defenses change their behavior significantly. Secondly, the results suggest the task of defending a Cifar10 classifier is harder as for MNIST, as a substantially greater improvement of robustness is necessary. The evaluation shows that RCE + K-Density does indeed increase the perturbation value round about four times. Nevertheless, the generated adversarial images do not contain any visible perturbation. They are not visible because the absolute value of the perturbation size is still very small as attacking Cifar10 is an easier task.
Taking into account that the perturbation improvements as well as the visibility of the perturbations for all defenses, Defense-GAN yields the best results on MNIST and is the only defense making it hard to identify the true class of the image. In general, it can be seen the defenses work a lot better on MNIST as on Cifar10 what suggests they can not realistically be used on complex datasets. Evaluation of the visibility approved it is indeed important to take this into account as it gives different insights than the norm of the perturbation alone.

3.3 Discussion

It would further strengthen the statistical expressiveness of the study to use more images as attack set, but the 100 images used, provide a good trade-off between expressiveness and feasibility. Given the time frame of the thesis and the relatively long attacking times against some defenses, it is not realistic to evaluate many more images. Furthermore, experiments show there is no relevant change in the results when using a different set. These experiments indicate the results are not strongly correlated with the input images which shows that the study is able to provide a qualitative evaluation of the defenses.
A high number of defense approaches were published over time and it is not reasonable to test all of them within one work. So there always exists the possibility that some of the conclusions do not hold for different approaches. To counter this, the defense selection includes some of the most relevant approaches with a variety of defense types. Therefore the results can be considered reflecting the state of the art robustness against decision-based black-box attacks.
The last critical point is that Defense-GAN and Ensemble Adversarial Training are only evaluated on MNIST despite using them for Cifar10 would probably provide different insights. As those defenses are not built for Cifar10 in the first place and therefore, no implementation is provided, it is questionable if they would yield improvements using the attacking methods from the original papers. In any case, it is out of scope for this thesis to implement those defenses for a new dataset. Still, comparing how the behavior changes for the other two defenses reveals some relevant assumptions about how the defenses would react on Cifar10.

4 Conclusion

This thesis evaluates the effectiveness of common defense mechanisms against adversarial attacks in the field of image classification. It uses a unified evaluation environment to enable a fair comparison of different defense approaches. Using this approach, the work tackles the well-known problem that many papers on defense mechanisms lack a comprehensive and stable evaluation of the proposed defense. [22]
The thesis addresses these objectives by conducting a study that compares four defenses, namely Defense-GAN [3], Ensemble Adversarial Training [4], Mahalanobis Detector [5], and RCE + K-Density Detector [6]. For every defense, the classifier used in the implementations of the according paper was extracted. The extraction provides the same models in an undefended and a defended version. Afterward, both versions were attacked with the HopSkipJumpAttack [7], an iterative decision-based black-box attack of the same type as Boundary Attack [8]. The experiments were carried out on the MNIST [9] and Cifar10 [10] dataset. The mentioned attack aims to reduce the perturbation in the adversarial example it generates in each iteration. For evaluation, the mean and median perturbation reached with respect to the number of executed iterations for a small test set of images is measured. Moreover, qualitative examples of the generated adversarial examples are taken into concern.
From the four defenses, three reach some improvement on the MNIST dataset regarding the mean and median of the reached perturbation. Only for two of them, Ensemble Adversarial Training and Defense-GAN, the visibility of the perturbation increases because of the defense and only Defense-GAN leads to visually confusing adversarial examples and is thus successfully defending against the attack. Regarding accuracy, Ensemble Adversarial Training is the only defense that does not have a negative impact. The use of Mahalanobis and K-Density detector leads to the most significant drop in accuracy as clean examples mistakenly get detected as adversarial and hence will not be classified further. In the comparison between MNIST and Cifar10, we can see the Cifar10 classifiers are a lot easier to attack, leading to adversarial examples not distinguishable from the original images and much smaller perturbation values. Similar studies [15, 16] presented as related work suggest similar results with different attacking methods. The authors of those papers as well disprove most of the defenses they investigate.
As this thesis only evaluates a subset of the existing defenses and new ones will be developed, the task this thesis performs is never fully completed. It is a part of future work to evaluate others, especially new defenses, as well. Additionally, it would be interesting to run similar experiments on a dataset like ImageNet [23] what is again closer to the real world than Cifar10.
The presented findings confirm that existing defenses can, at least on MNIST, reach slight

improvement in robustness against iterative black-box attacks but mostly not enough to build a robust classifier. On Cifar10 being closer to real-world images than MNIST, even defended classifiers are very easy to attack, suggesting none of the evaluated defenses would be strong enough to defend in real applications. The presented study adds to earlier research disproving introduced defense approaches. It shows that the insufficient evaluation in papers on defense mechanisms is still a present problem, even for contributions that are presented at important conferences and generally accepted in the field.

Bibliography

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. "Intriguing properties of neural networks." In: *International Conference on Learning Representations*. 2014.

[2] I. Goodfellow, J. Shlens, and C. Szegedy. "Explaining and Harnessing Adversarial Examples." In: *International Conference on Learning Representations*. 2015.

[3] P. Samangouei, M. Kabkab, and R. Chellappa. "Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models." In: *CoRR* abs/1805.06605 (2018). arXiv: 1805.06605.

[4] A. Kurakin, D. Boneh, F. Tramèr, I. Goodfellow, N. Papernot, and P. McDaniel. "Ensemble Adversarial Training: Attacks and Defenses." In: *International Conference on Learning Representations*. 2018.

[5] K. Lee, K. Lee, H. Lee, and J. Shin. "A Simple Unified Framework for Detecting Out-of-distribution Samples and Adversarial Attacks." In: *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*. NIPS'18. Montreal, Canada: Curran Associates Inc., 2018, pp. 7167–7177.

[6] T. Pang, C. Du, Y. Dong, and J. Zhu. "Towards Robust Detection of Adversarial Examples." In: *Proceedings of the 32Nd International Conference on Neural Information Processing Systems*. NIPS'18. Montreal, Canada: Curran Associates Inc., 2018, pp. 4584–4594.

[7] J. Chen and M. I. Jordan. "HopSkipJumpAttack++: Query-Efficient Decision-Based Adversarial Attack." In: *CoRR* abs/1904.02144 (2019). arXiv: 1904.02144.

[8] W. Brendel, J. Rauber, and M. Bethge. "Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models." In: *International Conference on Learning Representations*. 2018.

[9] Y. Lecun, L. Bottou, Y Bengio, and P. Haffner. "Gradient-Based Learning Applied to Document Recognition." In: *Proceedings of the IEEE* (1998).

[10] A. Krizhevsky. *Learning multiple layers of features from tiny images*. Tech. rep. University of Toronto, 2009.

[11] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[12] E. A. in IEEE Spektrum. *Three Small Stickers in Intersection Can Cause Tesla Autopilot to Swerve Into Wrong Lane*. 2019. URL: https://spectrum.ieee.org/cars-that-think/transportation/self-driving/three-small-stickers-on-road-can-steer-tesla-autopilot-into-oncoming-lane (visited on 06/21/2019).

[13] N. Akhtar and A. S. Mian. "Threat of Adversarial Attacks on Deep Learning in Computer Vision: A Survey." In: *IEEE Access* 6 (2018), pp. 14410–14430.

[14] S. Qiu, Q. Liu, S. Zhou, and C. Wu. "Review of Artificial Intelligence Adversarial Attack and Defense Technologies." In: *Applied Sciences* 9 (Mar. 2019), p. 909. DOI: 10.3390/app9050909.

[15] A. Athalye, N. Carlini, and D. Wagner. "Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples." In: *CoRR* (Feb. 2018).

[16] N. Carlini and D. A. Wagner. "Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods." In: *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. AISec '17. Dallas, Texas, USA: ACM, 2017, pp. 3–14. ISBN: 978-1-4503-5202-4. DOI: 10.1145/3128572.3140444.

[17] N. Carlini and D. A. Wagner. "Towards Evaluating the Robustness of Neural Networks." In: *2017 IEEE Symposium on Security and Privacy (SP)* (2017), pp. 39–57.

[18] R. Feinman, R. R. Curtin, S. Shintre, and A. Gardner. "Detecting Adversarial Samples from Artifacts." In: *arXiv preprint arXiv:1703.00410v3* (Mar. 2017).

[19] statisticshowto. *What is the Mahalanobis distance?* 2017. URL: https://www.statisticshowto.datasciencecentral.com/mahalanobis-distance/ (visited on 06/20/2019).

[20] J. Rauber, W. Brendel, and M. Bethge. "Foolbox: A Python toolbox to benchmark the robustness of machine learning models." In: *arXiv preprint arXiv:1707.04131* (2017). arXiv: 1707.04131.

[21] A. Kurakin, I. Goodfellow, and S. Bengio. "Adversarial examples in the physical world." In: *ICLR Workshop* (2017).

[22] N. Carlini, A. Athalye, N. Papernot, W. Brendel, J. Rauber, D. Tsipras, I. Goodfellow, and A. Madry. "On Evaluating Adversarial Robustness." In: *CoRR* (Feb. 2019).

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. "ImageNet: A Large-Scale Hierarchical Image Database." In: *CVPR09*. 2009.