Defending against adversarial attacks by randomized diversification

Olga Taran, Shideh Rezaeifar, Taras Holotyak, Slava Voloshynovskiy*
Department of Computer Science, University of Geneva
7, route de Drize, 1227 Carouge, Switzerland
{olga.taran, shideh.rezaeifar, taras.holotyak, svolos}@unige.ch

Abstract

The vulnerability of machine learning systems to adversarial attacks questions their usage in many applications. In this paper, we propose a randomized diversification as a defense strategy. We introduce a multi-channel architecture in a gray-box scenario, which assumes that the architecture of the classifier and the training data set are known to the attacker. The attacker does not only have access to a secret key and to the internal states of the system at the test time. The defender processes an input in multiple channels. Each channel introduces its own randomization in a special transform domain based on a secret key shared between the training and testing stages. Such a transform based randomization with a shared key preserves the gradients in key-defined sub-spaces for the defender but it prevents gradient back propagation and the creation of various bypass systems for the attacker. An additional benefit of multi-channel randomization is the aggregation that fuses soft-outputs from all channels, thus increasing the reliability of the final score. The sharing of a secret key creates an information advantage to the defender. Experimental evaluation demonstrates an increased robustness of the proposed method to a number of known state-of-the-art attacks.

1. Introduction

Besides remarkable and impressive achievements, many machine learning systems are vulnerable to adversarial attacks [6]. The adversarial attacks attempt at tricking a decision of a classifier by introducing bounded and invisible perturbations to a chosen target image. This weakness seriously questions the usage of the machine learning in many security- and trust-sensitive domains.

Many researchers have proposed various defense strategies and countermeasures to defeat adversarial attacks. However, the growing number of defenses naturally stimulates the invention of new and even more universal attacks.

*S. Voloshynovskiy is a corresponding author. The research was supported by the SNF project No. 200021_182063.

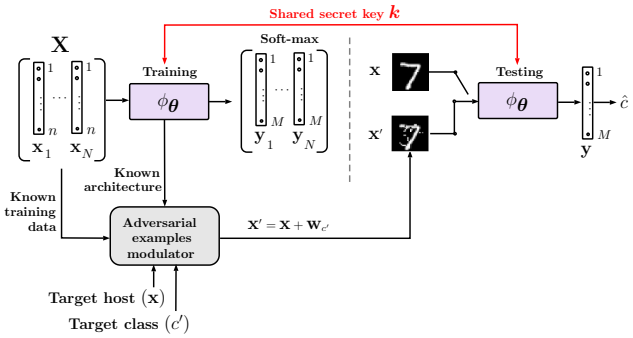


Figure 1: Setup under investigation: the attacker knows the labeled training data set $\mathbf{X}$ and the system architecture but he does not have access to secret key k of the defender shared between the training and testing.

An overview and classification of the most efficient attacks and defenses are given in [15, 10].

In this paper, we consider a "game" between the defender and the attacker according to the diagram presented in Figure 1.

The defender has access to the classifier $\phi_{\boldsymbol{\theta}}$ and the training data set $\mathbf{X}$. The defender shares a secret key k between training and testing. The classifier outputs a soft-max vector $\mathbf{y}$ of length M, where M corresponds to the total number of classes, and each y_c, $1 \leq c \leq M$ is treated as a probability that a given input $\mathbf{x}$ belongs to a class c. The trained classifier $\phi_{\boldsymbol{\theta}}$ is used during testing.

The attacker in the *white-box* scenario has full knowledge about the classifier architecture, defense mechanisms, training data and, quite often, can access the trained parameters of the classifier. In the *gray-box* scenario, considered in this paper, the attacker knows the architecture of the classifier, the general defense mechanism and has access to the same training data $\mathbf{X}$ [3, 15]. Using the above available knowledge, the attacker can generate a *non-targeted* or *targeted*, with respect to a specified class c', adversarial perturbation $\mathbf{w}_{c'}$. The attacker produces an adversarial example by adding this perturbation to the target host sample $\mathbf{x}$ as

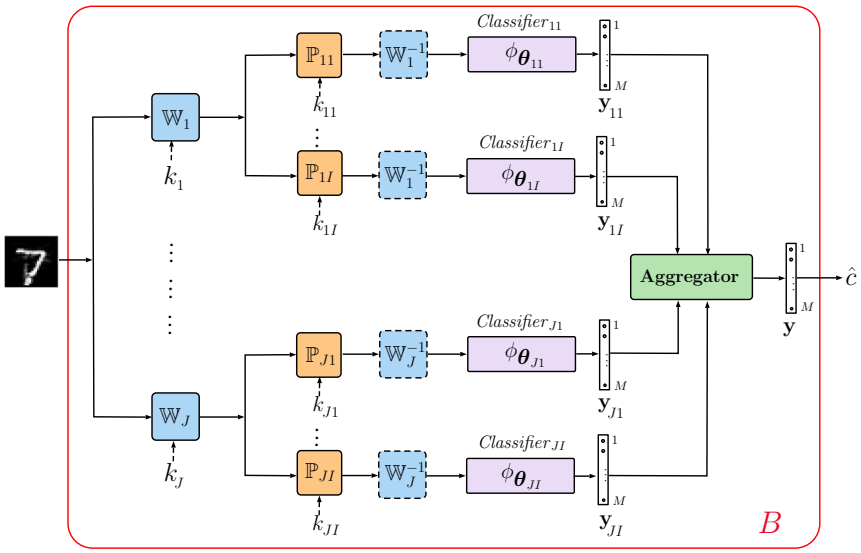


Figure 2: Generalized diagram of the proposed multi-channel classifier.

$\mathbf{x}' = \mathbf{x} + \mathbf{w}_{c'}$. The adversarial example is presented to the classifier at test time in an attempt to trick the classifier $\phi_{\boldsymbol{\theta}}$ decision.

Without pretending to be exhaustive in our overview, we group existing defense strategies into three major groups:

1. **Non key-based defenses**: This group includes the majority of state-of-the-art defense mechanisms based on detection and rejection, adversarial retraining, filtering and regeneration, etc. [10]. Besides the broad diversity of these methods, a common feature and the main disadvantage of these approaches is an absence of "cryptographic" elements, like for example a secret key, that would allow to create an information advantage of the defender over the attacker.

2. **Defense via randomization and obfuscation**: The defense mechanisms of this group are mainly based on the ideas of randomization avoiding the reproducible and repeatable use of parameters of the trained system. This includes gradient masking [1] and introducing an ambiguity via different types of key-free randomization. The example of such randomization can be noise addition at different levels of the system [14], injection of different types of randomization like, for example, random image resizing or padding [13] or randomized lossy compression [5], etc.

   The main disadvantage of this group of defense strategies consists in the fact that the attacker can bypass the defense blocks or take this ambiguity into account during the generation of the adversarial perturbations [1]. Additionally, the classification accuracy is degraded since the classifier is only trained on average for different sets of randomization parameters unless special ensembling or aggregation is properly applied to compensate this loss. However, even in this case the mismatch between the training and testing stages can only ensure the performance on average whereas one is interested to have the guaranteed performance for each realization of randomized parameters. Unfortunately, this is not achievable without the common secret sharing between the training and testing.

3. **Key-based defenses**: The third group generalizes the defense mechanisms, which include a randomization explicitly based on a secret key that is shared between training and testing stages. For example, one can mention the use of random projections [11], the random feature sampling [4] and the key-based transformation [10], etc.

   Nevertheless, the main disadvantage of the known methods in this group consists of the loss of performance due to the reduction of useful data that should be compensated by a proper diversification and corresponding aggregation.

In this paper, we target further extension of the key-based defense strategies based on the cryptographic principles to create an information advantage of the defender over the attacker yet maximally preserving the information in the classification system. The generalized diagram of the proposed system is shown in Figure 2. It has two levels of randomization, each of which can be based on unique secret keys. An additional robustification is achieved via the aggregation of the soft outputs of the multi-channel classifiers trained for their own randomizations. As it will be shown throughout

the paper, usage of multi-channel architecture diminishes the efficiency of attacks.

The main contribution of this paper is twofold:

- A new multi-channel classification architecture with defense strategy against *gray-box* attacks based on the cryptographic principle.
- An investigation of the efficiency of the proposed approach on three standard data sets for several classes of well-known adversarial attacks.

The remainder of this paper is organized as follows: Section 2 introduces a new multi-channel classification architecture. Section 3 provides an extension of the defense strategy based on the data independent permutation proposed in [10] to multi-channel architecture. The efficient key-based data independent transformation is investigated in Section 4. The filtering by a hard-thresholding in the secret domain is analyzed in Section 5. Section 6 concludes the paper.

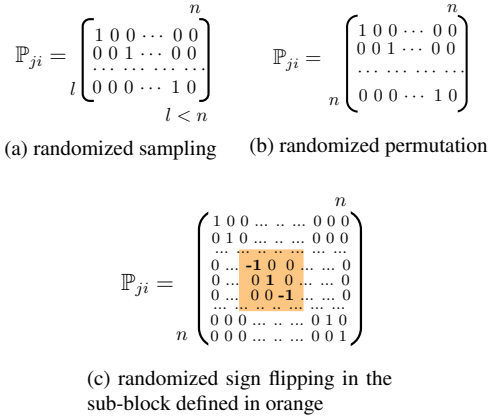
2. Multi-channel classification algorithm

A multi-channel classifier, which forms the core of the proposed architecture, is shown in Figure 2. It consists of four main building blocks:

1. Pre-processing of the input data in a *transform domain* via a mapping $\mathbb{W}_j$, $1 \leq j \leq J$. In general, the transform $\mathbb{W}_j$ can be any linear mapper. For example it can be a random projection or belong to the family of orthonormal transformations ($\mathbb{W}_j \mathbb{W}_j^T = \mathbb{I}$) like DFT (discrete Fourier transform), DCT (discrete cosines transform), DWT (discrete wavelet transform), etc. Moreover, $\mathbb{W}_j$ can also be a learnable transform. However, it should be pointed out that from the point of view of the robustness to adversarial attacks, the data independent transform $\mathbb{W}_j$ is of interest to avoid key-leakage from the training data. Furthermore, $\mathbb{W}_j$ can be based on a secret key k_j.

2. *Data independent processing* $\mathbb{P}_{ji}$, $1 \leq i \leq I$ presents the second level of randomization and serves as a defense against gradient back propagation to the direct domain.

   One can envision several cases. As shown in Figure 3a, $\mathbb{P}_{ji} \in \{0,1\}^{l \times n}$, $l < n$, presents a lossy sampling of the input signal of length n, as considered in [4]. In Figure 3b, $\mathbb{P}_{ji} \in \{0,1\}^{n \times n}$ is a lossless permutation, similar to [10]. Finally, in Figure 3c, $\mathbb{P}_{ji} \in \{-1,0,+1\}^{n \times n}$ corresponds to sub-block sign flipping. The yellow color highlights the key defined region of key-based sign flipping. This operation is reversible and thus lossless for an authorized party. Moreover, to make the *data independent processing* irreversible for the attacker, it is preferable to use a $\mathbb{P}_{ji}$ based on secret key k_{ji}.



(a) randomized sampling (b) randomized permutation

(c) randomized sign flipping in the sub-block defined in orange

Figure 3: Randomized transformation $\mathbb{P}_{ji}$, $1 \leq j \leq J$, $1 \leq i \leq I$ examples. All transforms are key-based.

3. *Classification block* can be represented by any family of classifiers. However, if the classifier is designed for classification of data in the direct domain then it is preferable that it is preceded by $\mathbb{W}_j^{-1}$.

4. *Aggregation block* can be represented by any operation ranging from a simple summation to learnable operators adapted to the data or to a particular adversarial attack.

As it can be seen from Figure 2, the chain of the first 3 blocks can be organized in a parallel multi-channel structure that is followed by one or several *aggregation blocks*. The final decision about the class is made based on the aggregated result. The rejection option can be also naturally envisioned.

The training of the described algorithm can be represented as:

$$(\hat{\boldsymbol{\vartheta}}, \{\hat{\boldsymbol{\theta}}_{\boldsymbol{ji}}\}) = \arg\min_{\boldsymbol{\vartheta}, \{\boldsymbol{\theta}_{ji}\}} \sum_{t=1}^{T} \sum_{j=1}^{J} \sum_{i=1}^{I_j} \mathcal{L}(\mathbf{y}_t, A_{\boldsymbol{\vartheta}}(\phi_{\boldsymbol{\theta}_{ji}}(f(\mathbf{x}_t)))), \tag{1}$$

with:

$$f(\mathbf{x}_t) = \mathbb{W}_j^{-1} \mathbb{P}_{ji} \mathbb{W}_j \mathbf{x}_t,$$

where $\mathcal{L}$ is a classification loss, $\mathbf{y}_t$ is a vectorized class label of the sample $\mathbf{x}_t$, $A_{\boldsymbol{\vartheta}}$ corresponds to the aggregation operator with parameters $\boldsymbol{\vartheta}$, $\phi_{\boldsymbol{\theta}_{ji}}$ is the ith classifier of the jth channel, $\boldsymbol{\theta}$ denotes the parameters of the classifier, T equals to the number of training samples, J is the total number of channels and I_j equals to the number of classifiers per channel that we will keep fixed and equals to I.

The attacker might discover the secret keys k_j and/or k_{ji} or make the full system end-to-end differentiable using the Backward Pass Differentiable Approximation technique proposed in [1] or via replacing the key-based blocks by

the bypass mappers. To avoid such a possibility, we restrict the access of the attacker to the internal results within the block B. This assumption corresponds to our definition of the gray-box setup.

In the proposed system, we will consider several practical simplifications leading to information and complexity advantages for the defender over the attacker:

- The defender training can be performed per channel independently until the *aggregation block*. At the same, the attacker should train and back propagate the gradients in all channels simultaneously or at least to guarantee the majority of wrong scores after aggregation.
- The blocks of *data independent processing* $\mathbb{P}_{ji}$ aim at preventing gradient back propagation into the direct domain but the classifier training is adapted to a particular $\mathbb{P}_{ji}$ in each channel.
- It will be shown further by the numerical results that the usage of the multi-channel architecture with the following aggregation stabilizes the results' deviation due to the use of randomizing or lossy transformations $\mathbb{P}_{ji}$, if such are used.
- The right choice of the *aggregation* operator A_{ϑ} provides an additional degree of freedom and increases the security of the system through the possibility to adapt to specific types of attacks.
- Moreover, the overall security level considerably increases due to the independent randomization in each channel. The main advantage of the multi-channel system consists in the fact that each channel can have an adjustable amount of randomness, that allows to obtain the required level of defense against the attacks. In a one-channel system the amount of randomness can be either insufficient to prevent the attacks or too high which leads to classification accuracy loss. Therefore, having a channel-wise distributed randomness is more flexible and efficient for the above trade-off.

The described generalized multi-channel architecture provides a variety of choices for the transform operators $\mathbb{W}$ and data independent processing $\mathbb{P}_{ji}$. In Section 3, we will consider a variant with multiple $\mathbb{P}_{ji}$ in the form of the considered permutation in the direct domain $\mathbb{W}_j = \mathbb{I}$. In Section 4, we will investigate a sign flipping operator $\mathbb{P}_{ji}$ for the common DCT operator $\mathbb{W}$. Section 5 will be dedicated to the investigation of a denoising version of $\mathbb{P}_{ji}$ based on hard-thresholding in a secret sub-space of the DCT domain $\mathbb{W}_j$.

3. Classification with multi-channel permutations in the direct domain

The simplest case of randomized diversification can be constructed for the direct domain with the permutation of input pixels. In fact, the algorithm proposed in [10] reflects this idea for a single channel. However, despite the reported efficiency of the proposed defense strategy, a single channel architecture is subject to a drop in classification accuracy, even for the original, i.e., non-adversarial, data.

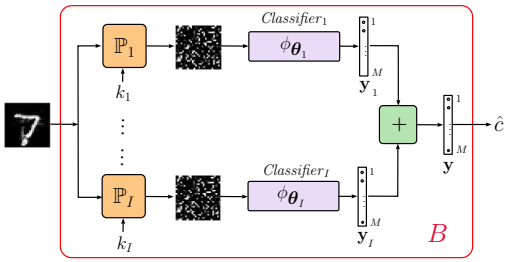


Figure 4: Classification via multi-channel permutations in the direct domain.

Therefore, this paper investigates the performance of a permutation-based defense in a multi-channel setting.

3.1. Problem formulation

The generalized diagram of the corresponding extended multi-channel approach is illustrated in Figure 4. The permutation in the direct domain implies that $\mathbb{W}_j = \mathbb{I}$ with $J = 1$ and I permutation channels. Therefore, each channel $1 \leq i \leq I$ has only one data independent permutation block $\mathbb{P}_i$ represented by a lossless permutation of the input signal $\mathbf{x} \in \mathbb{R}^{n \times n \times m}$ in the direct domain, where n corresponds to the size of the input image and m is the number of the channels (colors) in this image. Thus, the permutation matrix $\mathbb{P}_i$ is a matrix of size $n \times n$, generated from a secret key k_i, whose entries are all zeros except for a single element of each row, which is equal to one. In addition, as illustrated in Figure 3b, all non-zero entries are located in different columns. For our experiments, we assume that $\mathbb{P}_i$ is the same for each input image color channel but it can be a different one in the general case to increase the security of the system. As an aggregation operator A, we use a summation for the sake of simplicity and interpretability. The aggregated result represents a M-dimensional vector $\mathbf{y}$ of real non-negative values, where M equals to the number of classes and each entry y_c is treated as a probability that a given input $\mathbf{x}$ belongs to the class c.

Under the above assumptions, the optimization problem (2) reduces to [1]:

$$\{\hat{\boldsymbol{\theta}}_i\} = \arg\min_{\{\boldsymbol{\theta}_i\}} \sum_{t=1}^{T} \sum_{i=1}^{I} \mathcal{L}(\mathbf{y}_t, A(\phi_{\boldsymbol{\theta}_i}(\mathbb{P}_i \mathbf{x}_t))). \quad (2)$$

3.2. Numerical results

To reveal the impact of multi-channel processing, we compare our results with an identical single channel system

[1] https://github.com/taranO/defending-adversarial-attacks-by-RD

Data type	I					
	1	5	10	15	20	25
MNIST: original classifier error is 1%						
Original	2.83	1.73	1.37	1.43	1.57	1.4
CW ℓ_2	8.85	4.56	3.82	3.53	3.55	3.51
CW ℓ_0	13.87	5.98	4.98	4.69	4.47	4.4
CW ℓ_∞	11.67	4.72	4.03	3.87	3.59	3.69
Fasion-MNIST: original classifier error is 7.5%						
Original	11.40	9.4	9.27	9.2	9.23	9.2
CW ℓ_2	12.16	10.15	9.78	9.41	9.49	9.4
CW ℓ_0	13.45	10.15	9.62	9.56	9.82	9.63
CW ℓ_∞	11.99	9.72	9.69	9.24	9.26	9.32
CIFAR: original classifier error is 21%						
Original	47.03	41.47	40.2	39.8	39.2	39
CW ℓ_2	47.76	41.82	39.83	39.59	39.4	39.04
CW ℓ_0	48.39	42.27	40.87	39.73	39.85	39.76
CW ℓ_∞	47.41	42.12	40.53	39.58	39.62	39.21

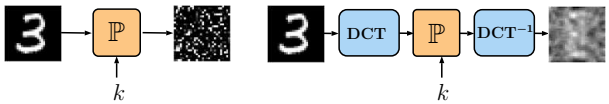
Table 1: Classification error (%) on the first 1000 test samples for I-channel system with the direct domain permutation.

reported in [10]. For each classifier $\phi_{\boldsymbol{\theta}_i}$ we use exactly the same architecture as mentioned in Table 2 in [10][2]. Moreover, taking into account that the generation of adversarial examples is quite a slow process, as well as in [10], we verify our approach on the first 1000 test samples of the MNIST [8] and Fashion-MNIST [12] data sets. Additionally, we investigate the CIFAR-10 data set [7].

The obtained results are given in Table 1. For all data sets, a single channel set up with $I = 1$ corresponds to the results of the approach proposed in [10] and *CW* denotes the attacks proposed by Carlity and Wagner in [2].

As one can note from Table 1, increasing the number of channels leads to a decrease of the classification error. In the case of the MNIST data set, our multi-channel algorithm allows to reduce the error on the original non-attacked data in 2 times, from 2.8% to 1.4%. For the attacked data, the classification error decreases 2.5 times from almost 9-14% to 3.5-4.5%. In case of the Fashion-MNIST data set, one can observe a similar dynamic, namely, the classification error decreases from 11.5-13.5% to 9-9.5. For the CIFAR-10 data set using the multi-channel architecture allows to reduce the error from 47-48% to only about 39.5%. The CIFAR-10 natural images are more complex in comparison to the MNIST and Fasion-MNIST and the introduced permutation destroys local correlations. This has a direct impact on classifier performance.

[2]The Python code for generating adversarial examples is available at https://github.com/carlini/nn_robust_ attacks



(a) coordinate domain (b) DCT domain

Figure 5: Global permutation in the coordinate domain (a) and DCT based encoding using key-based sign flipping (b).

4. Classification with multi-channel sign permutation in the DCT domain

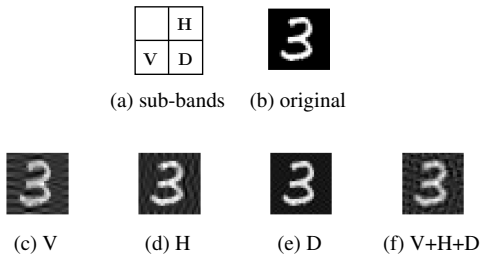
The results obtained in Section 3 for the CIFAR-10 data set show a high sensitivity to the gradient perturbations that degrade the performance of the classifier. In this Section we investigate the other data independent processing functions $\mathbb{P}_{ji}$ based on a secret key k_{ji} preserving the gradient in a special way that is more suitable for the classification of complex natural images. We will consider a general scheme for demonstrative purposes to justify that the permutations should be localized rather than global.

4.1. Global permutation

We will consider sign flipping in the DCT domain as a basis for the multi-channel randomization. For the visual comparison of the effect of global permutation in the coordinate domain versus global sign flipping in the DCT domain, we show an example in Figure 5. From this Figure one can note that the permutation in the coordinate domain disturbs the local correlation in the image that will impact the local gradients. In turn, this might impact the training of modern classifiers that are mostly based on gradient techniques. At the same time, preservation of the gradients makes the data more vulnerable to adversarial attacks. Keeping this in mind, we can conclude that the global DCT sign permutation also "randomizes" the images but, in contrast to the permutation in direct domain, it keeps the local correlation.

To answer the question whether the preservation of local correlation at the randomization can help preserve the loss of the gradients, we investigate the global DCT sign permutation for the classification architecture shown in Figure 4 with $\mathbb{W}_j$ is DCT and $\mathbb{P}_i \in \{-1, 1\}^{n\times n}$. It should be noted that the transform $\mathbb{W}_j$ is fixed for all channels. Therefore, the secrecy part consists in the key-based flipping of DCT coefficients' signs.

In the experimental results we obtained the classification accuracy to be very close to the results represented in Table 1. For the sake of space, we do not present this table in the paper. Nevertheless, we can conclude that the global sign permutation in the DCT domain does not improve the previous situation with the global permutation in direct domain.



(a) sub-bands (b) original (c) V (d) H (e) D (f) V+H+D

Figure 6: Local randomization in the DCT sub-bands by key-based sign flipping.

4.2. Local permutation

Taking into account the above observation, we investigate the behaviour of the local DCT sign permutations, i.e., we will use a global DCT transform but will flip the signs only for the selected number of coefficients as shown in Figure 6c.

The general idea, illustrated in Figure 6, consists in the fact that the DCT domain can be split into overlapping or non-overlapping sub-bands of different size. In our case, for the simplicity and interpretability, we split the DCT domain into 4 sub-bands, namely, (1) top left that represents the low frequencies of the image, (2) vertical, (3) horizontal and (4) diagonal sub-bands. After that we apply the DCT sign flipping as randomization in each sub-band keeping all other sub-bands unchanged and apply the inverse DCT transform. The corresponding illustrative examples are shown in Figures 6c - 6e. Finally, we apply the DCT sign permutation in 3 sub-bands. The corresponding result is shown in Figure 6f. It is easy to see that local DCT sign flipping applied in one individual sub-band creates a specific oriented distortion due to the specificity of chosen sub-bands but preserves the local image content quite well. The simultaneous permutation of 3 sub-bands creates more degradation which might be undesirable and can have a negative influence on the classification accuracy.

To investigate the behaviour of the local DCT sign permutations we use the multi-channel architecture shown in Figure 7. It is a three-channel model with I sub-channels. As a $\mathbb{W}_j$ we use a standard DCT transform. The sub-channels' data independent processing blocks $\mathbb{P}_{ji} \in \{-1, 0, 1\}^{n\times n}$ are based on the individual secret keys k_{ji} and are represented by the matrices that allow to change the elements' signs only in the sub-band of interest, like illustrated in Figure 3c. In general case, the sub-bands can be overlapping or non-overlapping and have different positions and sizes. As discussed, we use only 3 non-overlapping sub-bands of equal size as illustrated in Figure 6a. The architecture of the classifiers $\phi_{\boldsymbol{\theta}_{ji}}$ is identical to the ones used in Section 2. As an aggregation operator we use a simple

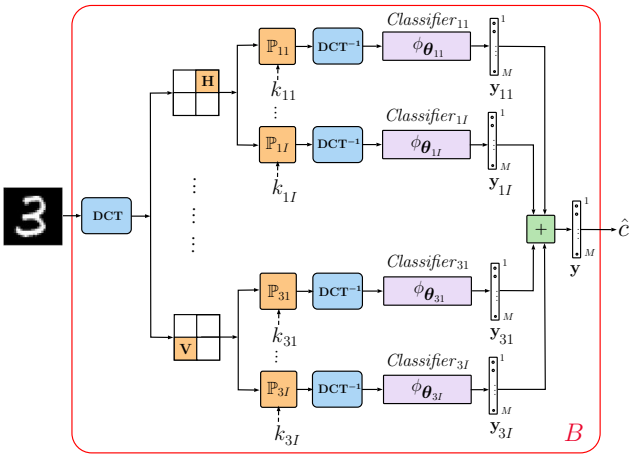


Figure 7: Classification with local DCT sign permutations.

summation.

The corresponding optimization problem becomes:

$$\{\hat{\boldsymbol{\theta}}_{ji}\} = \arg\min_{\{\boldsymbol{\theta}_{ji}\}} \sum_{t=1}^{T} \sum_{j=1}^{3} \sum_{i=1}^{I} \mathcal{L}(\mathbf{y}_t, A(\phi_{\boldsymbol{\theta}_{ji}}(\mathbb{P}_{ji}\mathbf{x}_t))). \quad (3)$$

4.3. Numerical results

The results obtained for the architecture proposed in Figure 7 are shown in Table 2. The column "Classical" corresponds to the results of the one-channel classical classifier for the original non-permuted data, that is referred to as the classical scenario.

It should be pointed out that in the previous experiments we observed a drop in the classification accuracy even for the original non-attacked data. In the proposed scheme with the 12 and 15 sub-channels, the obtained classification error on the adversarial examples corresponds to those of the original data and, in some cases, is ever lower. For example, we obtained a 2 times decrease in the classification error on the MNIST for the original data in comparison to the classical architecture.

The CIFAR-10 data set presents a particular interest for us as a data set with natural images. For *CW* ℓ_2 and *CW* ℓ_∞ attacks the classification error is the same as in the case of the classical scenario on the original data. This demonstrates that the proposed method does not cause a degradation in performance due to the introduced defense mechanism. In case of *CW* ℓ_0 attack there is only about 2% of successful attacks.

In the case of the Fashion-MNIST data set, the obtained results are better than the results for the permutation in the direct domain given in Table 1. For the original non-attacked data the classical scenario accuracy is achieved.

Data type	Classical	$J \cdot I$				
		3	6	9	12	15
MNIST						
Original	1	0.5	0.5	0.5	0.5	0.5
CW ℓ_2	100	6.28	5.34	4.66	4.44	4.73
CW ℓ_0	100	19.3	18.48	17.6	16.7	17.42
CW ℓ_∞	99.99	2.81	2.37	2.22	2.12	2.06
Fashion-MNIST						
Original	7.5	8.1	7.4	7.6	7.2	7.4
CW ℓ_2	100	9.27	8.67	8.87	8.62	8.62
CW ℓ_0	100	10.62	9.99	10.13	9.87	9.86
CW ℓ_∞	99.9	9.2	8.41	8.66	8.47	8.49
CIFAR-10						
Original	21	21.2	19.6	19.5	18.6	19.2
CW ℓ_2	100	22.42	21.3	21.04	20.79	20.92
CW ℓ_0	100	25.72	24.52	23.84	23.43	23.28
CW ℓ_∞	100	22.8	21.39	21.21	20.81	20.92

Table 2: Classification error (%) on the first 1000 test samples for the DCT domain with the local sign flipping in 3 sub-bands ($J = 3$).

For the attacked data the classification error exceeds the level of those on the original data only with 1-2%.

The situation with the MNIST data set is even more interesting. First of all, we would like to point out that we decrease in 2 times the classification error in comparison to the classical scenario. However, for the *CW* ℓ_0 the results are surprisingly worse. To investigate the reasons of the performed degradation we visualize the adversarial examples. The results are shown in Table 3. It is easy to see that, in general, the *CW* ℓ_∞ noise manifests itself as a background distortion and doesn't affect considerably the regions of useful information. The *CW* ℓ_2 noise affects the regions of interest but the intensity of the noise is much lower than the intensity of the meaningful information. As well as in *CW* ℓ_2, the *CW* ℓ_0 noise is concentrated in the region near the edges but its intensity is as strong as the informative image parts. Thus, it becomes evident why local DCT sign permutation is not capable to withstand such kind of noise. In general, such a strong noise is easy detectable and the corresponding adversarial examples can be rejected by many detection mechanisms, like for example an auxiliary "detector" sub-network [9]. Moreover, as it can be seen from the Fashion-MNIST examples, the influence of such noise and successful attacks drastically decreases with increasing image complexity. As it has been shown by the CIFAR-10 results, the local DCT sign permutation produces a high level of defense against such an attack for natural images.

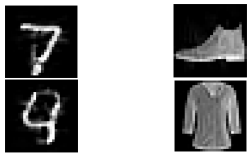
Attack	MNIST	Fashion-MNIST
CW ℓ_2		
CW ℓ_0		
CW ℓ_∞		

Table 3: Adversarial examples.

5. Classification with multi-channel hard thresholding in the sub-bands of the DCT domain

As it can be seen from Figure 6, the local DCT sign permutation creates sufficiently high image distortions. As a simple strategy to avoid this effect, we investigate hard thresholding of the DCT coefficients in the defined sub-bands. In this case the matrix $\mathbb{P}_{ji}$ contains zeros for the coefficients of key-defined sub-bands. Alternatively, one can consider this strategy as a random sampling as illustrated in Figure 3a, where one retains only the coefficients used by the classifier. In this sense, the considered strategy is close to the randomization in a single channel without the aggregation considered in [4].

Note that the considered processing is a data independent transform. The secret keys can be used for choosing the sub-bands positions. Thus, the attacker can not predict in advance, which DCT coefficients will be used or suppressed.

For simplicity and to be comparable with the previously obtained results, we use the multi-channel architecture that is shown in Figure 7, the DCT sub-band division as illustrated in Figure 6a with fixed 3 sub-band sizes and positions. Instead of applying the sign permutation, the corresponding DCT frequencies are set to zero and the result is transformed back to the direct domain. The visualization of the results of such a transformation is shown in Figure 8. The resulting images are slightly blurry but less noisy than in the case of the DCT sign permutation.

The obtained numerical results for the MNIST, Fashion-MNIST and CIFAR-10 data sets are given in Table 4. In

(a) original (b) sub-band V (c) sub-band H (d) sub-band D

Figure 8: Local zero filling in the DCT domain.

	Original	*CW* ℓ_2	*CW* ℓ_0	*CW* ℓ_∞
MNIST	0.6	7.59	21.3	3.03
Fashion-MNIST	8.8	9.6	11.23	9.58
CIFAR	21.1	23.28	27.08	23.27

Table 4: Classification error (%) for the DCT based hard thresholding over the first 1000 test samples ($J = 3,\ I = 1$).

general, the results are very close to the results of using the DCT sign permutation represented in Table 2 with the number of classifiers equals to 3. For the original non-attacked data the classification error is almost the same. In case of the attacked data, the classification error is about 0.5-1 % higher. This is related to the fact that the zero replacement of DCT coefficients leads to a loss of information and, consequently, to a decrease in classification accuracy.

Hence, replacing the DCT coefficients by zeros might also serve as a defense strategy.

6. Conclusions

In this paper, we address a problem of protection against adversarial attacks in classification systems. We propose the randomized diversification mechanism as a defense strategy in the multi-channel architecture with the aggregation of classifiers' scores. The randomized diversification is a secret key-based randomization in a defined domain. The goal of this randomization is to prevent the gradient back propagation or use of bypass systems by the attacker. We evaluate the efficiency of the proposed defense and the performance of several variations of a new architecture on three standard data sets against a number of known state-of-the-art attacks. The numerical results demonstrate the robustness of the proposed defense mechanism against adversarial attacks and show that using the multi-channel architecture with the following aggregation stabilizes the results and increases the classification accuracy.

For the future work we aim at investigating the proposed defense strategy against the gradient based sparse attacks and non-gradient based attacks.

References

[1] A. Athalye, N. Carlini, and D. Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 274–283, Stockholmsmssan, Stockholm Sweden, 10–15 Jul 2018. PMLR. 2, 3

[2] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57. IEEE, 2017. 5

[3] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26. ACM, 2017. 1

[4] Z. Chen, B. Tondi, X. Li, R. Ni, Y. Zhao, and M. Barni. Secure detection of image manipulation by means of random feature selection. *CoRR*, abs/1802.00573, 2018. 2, 3, 7

[5] N. Das, M. Shanbhogue, S.-T. Chen, F. Hohman, S. Li, L. Chen, M. E. Kounavis, and D. H. Chau. Shield: Fast, practical defense and vaccination for deep learning using jpeg compression. In *Proceedings of the 24nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2018. 2

[6] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations (ICLR)*, 2015. 1

[7] A. Krizhevsky, V. Nair, and G. Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 2014. 5

[8] Y. LeCun, C. Cortes, and C. Burges. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010. 5

[9] J. H. Metzen, T. Genewein, V. Fischer, and B. Bischoff. On detecting adversarial perturbations. In *International Conference on Learning Representations (ICLR)*, 2017. 7

[10] O. Taran, S. Rezaeifar, and S. Voloshynovskiy. Bridging machine learning and cryptography in defence against adversarial attacks. In *Workshop on Objectionable Content and Misinformation (WOCM), ECCV2018*, Munich, Germany, September 2018. 1, 2, 3, 4, 5

[11] N. X. Vinh, S. Erfani, S. Paisitkriangkrai, J. Bailey, C. Leckie, and K. Ramamohanarao. Training robust models using random projection. In *Pattern Recognition (ICPR), 2016 23rd International Conference on*, pages 531–536. IEEE, 2016. 2

[12] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 5

[13] C. Xie, J. Wang, Z. Zhang, Z. Ren, and A. Yuille. Mitigating adversarial effects through randomization. In *International Conference on Learning Representations (ICLR)*, 2018. 2

[14] Z. You, J. Ye, K. Li, and P. Wang. Adversarial noise layer: Regularize neural network by adding noise. *arXiv preprint arXiv:1805.08000*, 2018. 2

[15] X. Yuan, P. He, Q. Zhu, R. R. Bhat, and X. Li. Adversarial examples: Attacks and defenses for deep learning. *arXiv preprint arXiv:1712.07107*, 2017. 1