12

# New technique artificial intelligent model against image classification attacks using Hadamard codes.

**Ali Abdul Razak Hussein**
**Al-Mustansiriya University**

**Information Technology Center,**

**Dr. Mohammed Audi**

**Modern University for Business and Science_ Lebanon**

**Abstract**

Deep Neural Networks (DNN) can learn to classify, synthesize and, in general, infer correlations in large collections of data. They have become black box solutions for countless everyday tasks. However, DDNs are susceptible to adversarial attacks, which consist of instances that contain modifications imperceptible to a human observer which are designed so that the DNNs make mistakes when making a prediction. It was noted that all DNN-based architectures usage one-hot encoding later a softmax coat. The Hamming detachment among pairs of one-hot codes is 2, regardless of the number of classes, and it follows that this is essential for a successful attack. The attacker only needs to generate a disturbance capable of changing a couple of bits of information to incorrectly classify an image. Something like the transmission of information through a noisy channel. With the above observation in mind, in this study it is postulated that the use of error correcting codes can improve the resistance of DNNs against adversary attacks. Hadamard codes are used for this purpose. Hadamard codes of order n, for classes O(n), have a Hamming distance of n/2. Therefore, an adversary attack would need to change a larger number of bits to misclassify an image. This hypostudy was tested against six kinds of white case attacks and one black case attack. The models with Hadamard codes presented greater resistance that far exceeded the alternatives in the state of the art. We outperform distilled models, considered the best adversarial defense in the related literature.

## تقنية جديدة لنموذج الذكاء الاصطناعي ضد هجمات تصنيف الصور

## باستخدام أكواد هادمورد

**علي عبد الرزاق حسين jlybdalrzaq@gmail.com**

**مركز تكنولوجيا المعلومات ـ الجامعة المستنصرية- العراق**

**د.هيثم معروف jlybdalrzaq@gmail.com**

**خلاصة**

يمكن للشبكات العصبية العميقة (DNN) أن تتعلم كيفية تصنيف وتوليف واستنتاج الارتباطات بشكل عام في مجموعات كبيرة من البيانات. لقد أصبحت حلول الصندوق الأسود لعدد لا يحصى من المهام اليومية. ومع ذلك، فإن شبكات DDN عرضة للهجمات العدائية، والتي تتكون من حالات تحتوي على تعديلات غير محسوسة للمراقب البشري والتي تم تصميمها بحيث ترتكب شبكات DNN الأخطاء عند التنبؤ. وقد لوحظ أن جميع البنى المستندة إلى DNN تستخدم ترميزًا ساخنًا واحدًا لاحقًا أ معطف سوفت ماكس. إن انفصال هامينج بين أزواج الرموز الساخنة الواحدة هو 2، بغض النظر عن عدد الفئات، ويترتب على ذلك أن هذا ضروري لهجوم ناجح. يحتاج المهاجم فقط إلى إحداث اضطراب قادر على تغيير بضع أجزاء من المعلومات لتصنيف صورة بشكل غير صحيح. شيء مثل نقل المعلومات عبر قناة صاخبة. مع أخذ الملاحظة المذكورة أعلاه في الاعتبار، من المفترض في هذه الدراسة أن استخدام رموز تصحيح الأخطاء يمكن أن يحسن المقاومة DNNs ضد هجمات العدو. وتستخدم رموز Hadamard لهذا الغرض. رموز هادامارد من الرتبة n، للفئات O(n)، لها مسافة هامينغ n/2. لذلك، سيحتاج هجوم الخصم إلى تغيير عدد أكبر من البتات لإساءة تصنيف الصورة. تم اختبار هذه الفرضية ضد ستة أنواع من هجمات الحالة البيضاء وهجوم حالة سوداء واحدة. قدمت النماذج التي تحتوي على رموز Hadamard مقاومة أكبر تجاوزت بكثير البدائل المتوفرة في أحدث ما توصلت إليه التكنولوجيا. نحن نتفوق على النماذج المقطرة، التي تعتبر أفضل دفاع عدائي في الأدبيات ذات الصلة.

***Acknowledgement**: The authors would like to thank Mustansiriyah University (www.uomustansiriyah.edu.iq), Baghdad-Iraq for its support in the present work.*

## *1.Introduction*

Advances in deep neural networks (DNN), the development of hardware (GPUs), and the convenience of a large amount of data have allowed us to solve a variety of problems. In particular, the Imagenet image set (Deng et al., 2009), together with the AlexNet model (Krizhevsky et al., 2012), popularized the study of the image classification problem using DNN.

Hypostudy 1.3

A convention used during the training of a DNN consists of associating each image to a vector whose size is the total number of classes in the problem. In the position corresponding to its class, the vector has a value of one and zero in the rest, this is known as a one-hot encoding.

One-hot coding was introduced in (Huffman, 1954) as a memory reduction needed in sequential circuit switching, in the area of electrical engineering. An advantage of this encoding is that it can binarize the categorical entries (labels) so that they are considered a vector in a Euclidean space, which is used to calculate distances or similarities in many classification algorithms (Cassel and Lima, 2006). Due to its simplicity, it is the most popular coding strategy in the machine learning area (Rodríguez et al., 2018).

For its part, an adversary attack needs to change a couple of bits in the one-hot encoding for a model to get the classification prediction wrong. An attack easily makes this change because the Hamming distance among a pair of one-hot codes is 2, regardless of the total number of classes.

On the other hand, in the information transmission problem, error correction codes are used to add redundancy to a message through encoding. In some cases, this encoding has a Hamming distance of n/2 for any pair of codes, where n the size of the code.

Based on the above observations, this paper seeks to validate the following hypostudy: Increasing the Hamming distance in class encoding using error correction techniques will increase the tolerance of a model against adversary attacks.

From the model of Krizhevsky et al., multiple DNN-based proposals such as VGG (Simonyan and Zisserman, 2014) and ResNet (He et al., 2016) have been developed to solve the image classification problem (Cheng et al., 2017). These same models have allowed advances in other computer

image problems, such as object detection (Redmon and Farhadi, 2018), image segmentation (Zhang et al., 2017), pose estimation (Toshev and Szegedy, 2014), among others.

Despite the advances mentioned above, Szegedy et al. reported a weakness in the DNNs called adversarial examples. These examples are images modified in such a way that a model misclassifies the resulting image, while to the naked eye, a person does not notice the changes.

For the generation of adversarial examples, two main approaches have been used. The first of them is based on geometric transformations, such as rotation and translation (Engstrom et al., 2019). On the other hand, the most popular approach is to create a perturbation from white noise, which is combined with an input image to generate an adversarial example (Szegedy et al., 2013), >

## 2. Problem statement

Computer vision systems have gone from being a futuristic idea to becoming a burgeoning field of research. Today, computers are capable of outperforming humans at some vision tasks, such as classifying images. The way these systems process information is increasingly different from how humans do it.

DNN models accept image as input, course it in series of steps where they first notice pixels, then edges and contours, until they reach full objects, and finally produce a prediction about what they are observing.

Although DNN models are trained using large amounts of information, their style is fixed and does not complete over time as a human does. For example, when we see an image of a cat, we are most likely able to recognize the animal in the image, even if it is red or striped, if the image is black and white, or if the image is worn and faded. discolored We can probably detect this pet when it appears curled up behind a pillow or jumping on a piece of furniture. Naturally, we have learned to identify a cat in almost any situation. On the other hand, although DNNs can outperform humans in recognizing a cat in still conditions, new images that are a bit novel or noisy can cause these models to fail altogether.

In (Geirhos et al., 2018), they argue that humans pay more attention to the shapes of the represented objects, while DNN models focus on the textures

of the objects. This finding highlights the stark contrast between how humans and machines "think" and illustrates how misleading our intuitions about what makes artificial intelligence work can be.

On the other hand, this problem is not limited to image processing, in (Bou cher et al., 2021) it was shown that small modifications in a text, which simply include white spaces, can wreak havoc on the understanding of the text by part of a learning model. Furthermore, these alterations also have consequences for users, in one example it is shown that changing a single character can cause a user to send money to the wrong bank account.

For that, It is proposed to use error correction codes to increase the Hamming distance between classification codes and increase the Replace the one-hot robustness of a model against adversary attacks. encoding traditionally used to train DNN, by a Hadamard encoding. Estimate the act of the planned coding against different adversary attacks compared to the best defensive method mentioned in the related literature.

## 3. methodology proposed

the proposed system implements a class classification founded on Hadamard codes as shown in Figure 20 when training a DNN model, as a self-protective strategy against adversary attacks. It is worth mentioning that (Yang et al., 2015) suggest the usage of error correction codes to improve the discriminatory characteristics in DNN. However, the authors do not study the problem of adversarial examples.

| Class | One-hot codification | Hadamard codification |
|---|---|---|
| 0 | 1000000000 | 1111111111111111 |
| 1 | 0100000000 | 1010101010101010 |
| 2 | 0010000000 | 1100110011001100 |
| 3 | 0001000000 | 1001100110011001 |
| 4 | 0000100000 | 1111000011110000 |
| 5 | 0000010000 | 1010010110100101 |
| 6 | 0000001000 | 1100001111000011 |
| 7 | 0000000100 | 1001011010010110 |
| 8 | 0000000010 | 1111111100000000 |
| 9 | 0000000001 | 1010101001010101 |

Figure 1 Shows the one-hot and Hadamard coding of the 10 classes corresponding to the MNIST dataset.

For the implementation of a robust encoding of classes, the following training process is carried out. Given any DNN model, the softmax output layer is removed and replaced by a dense layer, whose output vector corresponds in size to the codes being used. In case of using a pre-trained model, it is recommended to freeze the intermediate weights of the model, known as deep features, and only train the classifier part of the model.

Subsequently, the training process is like that performed with one-hot encoding. For each of the problem classes, a Hadamard code is assigned as shown in Figure 20, which is associated with each of the images of said class.

In a classification model that uses one-hot coding, the prediction of a class corresponds to the position with maximum value in the output vector. While a prediction in the proposed model corresponds to the code with the shortest distance between the model output and the matrix of Hadamard codes used.

## 4. Preprocessing proposed work

To assess the robustness of the classification when using Hadamard codes for class labeling in comparison to a traditional coding, in this chapter different experiments are presented that validate the proposed method. Each one of them is linked giving to a top -1 precision metric in a validation set, that is, the relation among the precise guesses and the entire amount of estimates made for a set of images that were not considered during the process. of training.

In each experiment, a base model is used, which does not put on a little security approach and is used as a reference to show how an adversary attack affects classification accuracy as its intensity increases. Additionally, to make a comparison against the best known defensive strategy, distilled variants of the base model are included, where each variant was trained with a temperature value T equal to 1, 5, 10 and 25. Finally, a variant of the base model using the Hadamard coding.

To generate the necessary adversary examples for each experiment, the Foolbox library (Rauber et al., 2020) was used. It was decided to use this library since it has the implementation of the main attacks mentioned in the related literature. It is worth mentioning that separately attack was generated by means of its defaulting variables. The only parameter that was modified was the pertubation perception value ε, where values within the interval [0, 1]. It is important to remember that an adversarial example will accomplish its goal of misclassifying an image as the value of ε, however, this also increases the chance that a human observer will discover the deception.

.

## 5. Discussion experiments and results:

For the first experiment, we chose to use MNIST (Deng, 2012), this set of hand written digit photos covers 60 thousand training photos and 10 thousand validation photos. LeNet5 (LeCun et al., 2015) was used as the base model, since it has been used successfully to solve this classification problem. The distilled variants of the LeNet5 model correspond to the temperature values T mentioned above . While the variants trained with a Hadamard coding consisted of codes with values of [0, 1] of length 10 and 16, this is because when constructing the Hadamard codes a size corresponding to 2k. However, MNIST has 10 classes, so it was decided to train a model with the original code size (16) and another with the code size according to the classes number (10).

Finally, the enemy attacks castoff in this trial were the white box attacks: Fast Gradient Sign Method (FGSM), Basic Iterative Method (BIM) and Projected Gradient Descent Attack (PGD), since they are the most mentioned in the related literature. For each image in the validation set, an adversarial example was generated using one of the previous attacks with the intention of affecting only the model in charge of the evaluation. The results corresponding to this first experiment are shown in Figure 22.

In the results shown in the previous figure, it is important to observe that all the models have a similar classification percentage when there are no attacks ($\varepsilon = 0$) and this percentage decreases as the value of $\varepsilon$ increases. However, the models that use a Hadamard coding present a higher classification accuracy in each generated attack, despite increasing the value of $\varepsilon$. In some cases, accuracy better than $\sim 40\%$ compared to the base model and distilled models. In addition, it is possible to observe that when

training with Hamming codes in their original length, there is a greater tolerance against adversary attacks, compared to the case in which the codes are trimmed. This is because trimming the Hamming codes decreas

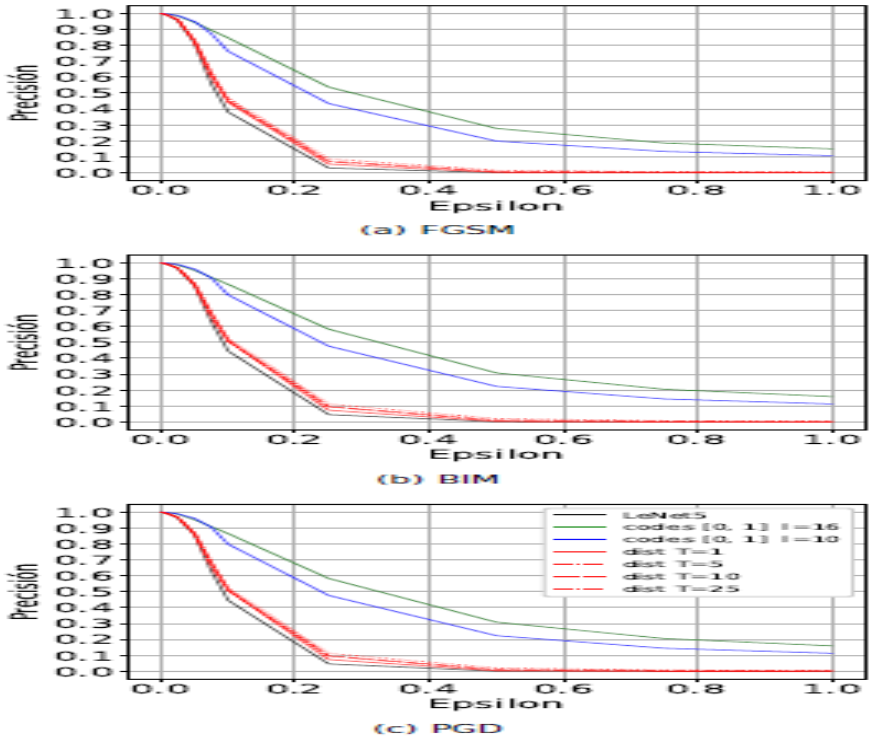


Figure *2* Top-1 accuracy on the MNIST dataset proof usual using the LeNet5 classification model below 3 different attacks: (a) FGSM, (b) BIM and (c) PGD

In the previous chapter it was mentioned that the output of a model is a vector of real numbers. For this reason, in the models that use the Hada mard coding, MSE is used to calculate the distance between the output and the corresponding class codes. Now, observing that there is a greater tolerance to attacks by increasing the length of the output vector, it was decided to train a couple of m (a) FGSM easing the intrinsic distance of the Hadamard codes using values of [−5, 5] instead of [0, 1] and in the same way as in the first experiment, this modification was tested with codes of length 10 and 16. The results obtained can be seen in Figure 23.



(b) BIM

In the previous results, it seems enough to use codes in their length original, since this model presents a greater precision when the value of $\varepsilon \leq 0.4$. However, it is possible to notice that the use of Hadamard codes with values of $[-5, 5]$ results in better robustness in contradiction of adversary attacks compared to codes with standards of $[0, 1]$ from a value of $\varepsilon \geq 0.5$. Note that for this data set it is well to variety the range of codes as well as their distance.

In Figure 3, the accuracy for the training and validation set at each training epoch is shown. It is possible to observe that the base model and the model trained with a Hadamard encoding, achieve a very similar precision after a certain number of epochs, experimentally showing together models meet at a like rate, that is, for the set of MNIST images, no additional training cycles are required when performing the encoding change.

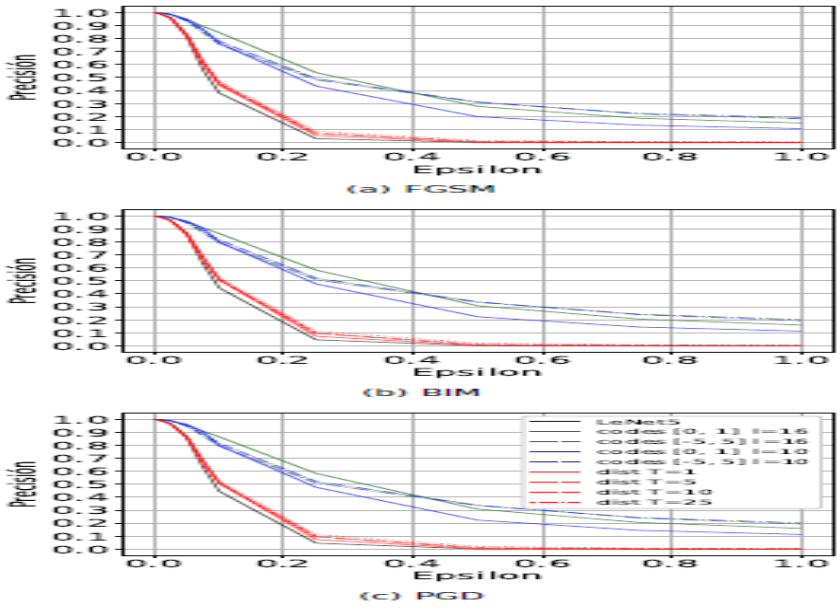


Figure 3 Top-1 accuracy on the MNIST dataset proof usual using the LeNet5 classification model for 3 ditinct attacks: (a) FGSM, (b) BIM and (c) PGD

***White Box Attacks***

The used in this section were: BIM, DDN, FGSM, PGD, VA and C and W. For each attack, an advesarial example was generated from the validation images using the Foolbox library, with the perturbation intensity values ε between [0, 1]. Like the previous experiments, each adversary example was generated with the intention of attacking the model performing the evaluation.

The Top-1 precision results of the Mini-Imagenet proof usual shown in Figure 26 for the ResNet18 model, Figure 4 for the ResNet50 model, and Figure 28 for the ResNet101 model. For the experiments in this section, the distilled models show a lower initial precision compared to the rest of

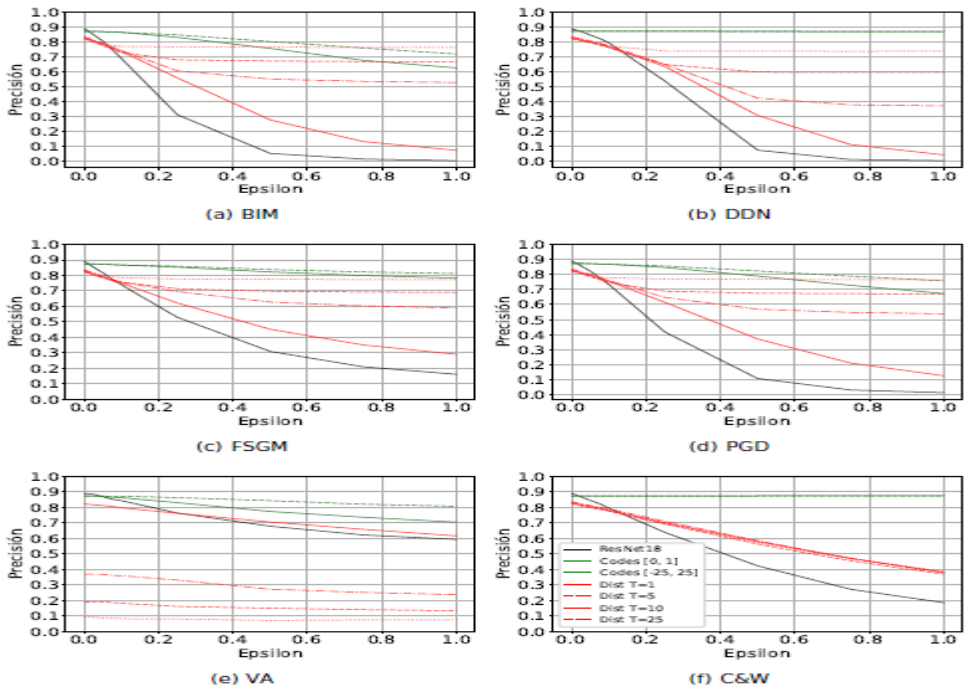


Figure 4 Top-1 accuracy on the Mini-Imagenet dataset (a) BIM, (b) DDN, (c) FGSM, (d) PGD, (e) VAA, and (f) C&W

## 6. Conclusion

This study adopted a system Using error-correcting codes, particularly Hadamard codes, for class labeling makes neural network models more robust in contrast to multiple adversarial white box and black box attacks. , performing only one training cycle.

For FGSM, BIM, PGD, DDN, VA, and C&W white box attacks, a decrease in classification accuracy of ~ 90 % in a ResNet base model as the attack intensity ε, while for the proposed model with Hadamard codes with values of [0, 1] the worst drop was ~ 23% and the Hadamard coding variant with values of [−25, 25] had its worst decrease of ~ eleven %. For its part, although the best distillate model T = 25 only presented a drop of ~ 8% in most of these attacks, the classification percentage is lower than that presented by the proposed model.

In the experiments carried out, the Hadamard codes proved to be a good strategy against the adversary examples in the face of multiple attacks. However, it is of interest to explore the performance of other linear fault improvement codes as Hamming codes (Hamming, 1950) or Reed Solomon codes (Reed, 1953), to name a few. Studying these or other codes could show some advantages compared to Hadamard's codes, be it greater robustness against a specific attack, exceed the results obtained in this study or consolidate the results obtained by showing that Hadamard's codes are the best option for encoding switching between multiple linear error correction codes.

## Acknowledgement

The authors would like to thank Mustansiriyah University (www.uomustansiriyah.edu.iq) , Baghdad-Iraq for its support in the present work.

## 7.References:

[1] Hamming, R.W. (1950). Error detecting and error correcting codes. The Bell system technical journal, 29(2): 147–160.

[2] Ahmad, K. and Conci, N. (2019). How deep features have improved event recognition in multimedia: A survey. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 15(2): 1–27.

[3]Athalye, A., Engstrom, L., Ilyas, A., & Kwok, K. (2018). Synthesizing robust adversarial examples. In: International conference on machine learning. PMLR, p. 284–293.

[4]Boucher, N., Shumailov, I., Anderson, R., & Papernot, N. (2021). Bad characters: Imperceptible nlp attacks. Retrieved on 10-26-2021 from https://arxiv.org/abs/2106.09898.

[5]Brown, TB, Mané, D., Roy, A., Abadi, M., & Gilmer, J. (2017). adversarial patch. Retrieved on 10-26-2021 from https://arxiv.org/abs/1712.09665.

[6]Carlini, N. & Wagner, D. (2017). Towards evaluating the robustness of neural networks. In: 2017 IEEE Symposium on Security and Privacy. IEEE, p. 39–57.

[7]Cassel, M. and Lima, F. (2006). Evaluating one-hot encoding finite state machines for their reliability in sram-based fpgas. In: 12th IEEE International On-Line Testing Symposium (IOLTS'06). IEEE, p. 6–pp.

[8]Chakraborty, A., Alam, M., Dey, V., Chattopadhyay, A., & Mukhopadhyay, D. (2018). Adversarial attacks and defenses: A survey. Retrieved on 10-26-2021 from https://arxiv.org/abs/1810.00069.

[9]Chen, P.-Y., Zhang, H., Sharma, Y., Yi, J., and Hsieh, C.-J. (2017). 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. pp. 15–26.

[10]Cheng, Y., Wang, D., Zhou, P., & Zhang, T. (2017). A survey of model compression and acceleration for deep neural networks. Retrieved on 10-26-2021 from https://arxiv.org/abs/1710.09282.

[11]Cho, JH & Hariharan, B. (2019). On the efficacy of knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), October.

[12]Deng, J., Dong, W., Socher, R., Li, L., Kai Li, and Li Fei-Fei (2009). Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255.

[13]Goodfellow, IJ, Shlens, J., & Szegedy, C. (2014). Explaining and harnessing adversarial examples. Retrieved on 10-26-2021 from https://arxiv.org/abs/1412.6572.

[15]Grosse, K., Manoharan, P., Papernot, N., Backes, M., & McDaniel, P. (2017). On the (statistical) detection of adversarial examples. Retrieved on 10-26-2021 from https://arxiv.org/abs/1702.06280.

[16]Guo, C., Rana, M., Cisse, M., & Van Der Maaten, L. (2017). Countering advertising versatile images using input transformations. Retrieved on 10-26-2021 from https://arxiv.org/abs/1711.00117.

[17]Hadamard, J. (1893). Resolution of a question relative to determinants. Bull. des sciences math., 2: 240–246.

[19]M. A. Mohammed, Z. H. Salih, N. Ţăpuş, and R. A. K. Hasan, "Security and accountability for sharing the data stored in the cloud," in RoEduNet Conference: Networking in Education and Research, 2016 15th, 2016, pp. 1-5: IEEE.