Sparse adversarial image generation using dictionary learning

Maham Jahangir[a,*] **and Faisal Shafait**[a,b]
[a]National University of Sciences and Technology, School of Electrical Engineering and Computer Science, Islamabad, Pakistan
[b]National Center of Artificial Intelligence, Deep Learning Laboratory, Islamabad, Pakistan

Abstract. Adversarial examples are used to evaluate the robustness of convolutional neural networks (CNNs) to input perturbations. Researchers have proposed different types of adversarial examples that attack CNNs to fool them. These attacks pose a serious threat to applications that use deep neural networks. Existing methods for adversarial image generation struggle in maintaining a balance between attack success rate and imperceptibility (measured in terms of ℓ_2-norm) of the generated adversarial examples. Recent sparse methods for this problem focus on limiting the number of pixels in an image but do not cater to the overall imperceptibility of the adversarial images. To address these problems, we introduce adversarial attacks based on K-singular value decomposition sparse dictionary learning. The dictionary is learned using feature maps of the targeted images from the first layer of CNN. The proposed method is evaluated in terms of attack success rate and ℓ_2-norm. The extensive experimentation shows our attack achieves a high success rate while maintaining a low imperceptibility score compared to state-of-the-art methods. © *2023 SPIE and IS&T* [DOI: 10.1117/1.JEI.32.3.033006]

Keywords: sparse attacks; K-singular value decomposition; sparse representation; dictionary learning.

Paper 220528G received May 21, 2022; accepted for publication Feb. 14, 2023; published online May 18, 2023.

1 Introduction

Deep neural networks have shown remarkable performance in various tasks such as face detection, person re-identification, image classification, speech recognition, and audio/video processing. Despite achieving human-level performance in these tasks these networks are vulnerable to adversarial attacks. Szedgy et al.[1] discovered the unusual mistake that deep networks make when human imperceptible adversarial examples attack to fool them. Deep neural networks have high representation power but this same quality is exploited to craft adversarial attacks. The adversarial attacks can be defined as imperceptible transformations applied to an input image with the goal to modify the input image correctly classified by the model into a new adversarial example that is incorrectly classified. In other words, adversarial attacks are malicious attacks on the input data, which may seem okay to the human eye but cause chaos in machine learning systems. These are specially designed noises carefully added as perturbations to the data. These vulnerabilities have intrigued researchers in this area and a lot of attacks and defense mechanisms have been proposed since then. The fast gradient sign method[2] attacks a neural network by using gradients. Unlike, normal deep neural networks, which minimize the loss during training, this method uses the gradients of the loss for the input image to create a new image that maximizes the loss. CW[3] attacks make use of two loss functions. The first one is the adversarial loss so that an image gets misclassified by the classifier. Second image distance loss is built to constrain the imperceptibility of the adversarial example. It produces adversarial examples with minimum ℓ_2-norm but is quite expensive as it requires thousands of iterations and is very time-consuming. JSMA[4] was proposed by Papernot. It is a gradient-based white box attack that uses the saliency

*Address all correspondence to Maham Jahangir, mjahangir.phdcs17seecs@seecs.edu.pk

1017-9909/2023/$28.00 © 2023 SPIE and IS&T

map to select the dimension, which produces the maximum error for every component of the image. Corner Search[5] is a type of sparse attack that aims to craft adversarial examples aiming at minimizing ℓ_0-distance to the original image.

These state-of-the-art attacks have several limitations. First, in terms of ℓ_2-norm distortions, the C&W is considered among very strong attacks but performs very slowly since it converges after thousands of iterations making it impracticable. This high time complexity makes it unfit for adversarial training too.[6] Moreover, it is also argued that the fast gradient method computes adversarial examples that are quite perceptible. A lot of noise has to be added to misclassify the clean input image.[7] In 2019, Shi et al.[8] explained that iterative attacks produce adversarial samples that accumulate redundant noises, which are hard to remove. The above-mentioned problems highlight the importance of a need to design attacks that should be computationally efficient and imperceptible in terms of ℓ_2-norm.

The ℓ_2-norm is a standard method to compute the length of a vector in Euclidean space. We use it to find the similarity between two images. Here it is the squared distance between the adversarial and original clean image. The lower the distance this means that the two images appear the same and the noise in adversarial images is imperceptible. In this research article, we have tried to overcome these problems using the ideas from the internal representation of convolutional neural network (CNN), sparse representation, and dictionary learning. We have tried to attack the internal representation of images. The feature maps constitute the internal representation of CNN. It is the output of the convolutional layer of CNN, which detects different features in the image to help in classification. We have used a sparse representation of activation maps generated from the first convolutional layer of CNN and introduced it as the perturbation. This sparse representation of feature maps is optimized using K-singular value decomposition (K-SVD)-based dictionary learning algorithm. Sparse dictionary learning is well known for its utility in tasks like image denoising,[9] image painting,[10] and camera calibration.[11] Sparse dictionary learning is a representation learning in which an image is represented linear sparse representation using only a few active non-zero coefficients.[12] For adapting the dictionary to achieve sparse representation, Aharon et al.[13] proposed a K-SVD method to learn the dictionary. Our research focuses on the application of K-SVD to learn a dictionary that transforms our perturbation vector to achieve imperceptibility by minimizing the ℓ_2-norm. As we mentioned earlier, adversarial attacks are an imperceptible transformation of the data. The motivation to use dictionary learning is based on its ability to transform data into a sparse linear combination of atoms. It can transform any data to another form that retains the representation of an image using the minimum number of pixels. This power can be used in two different ways. One is to use dictionary learning to remove noises from the input, hence as a defense mechanism against adversarial attacks. We have tried using its power for an alternate cause, which is to create adversarial attacks. Instead of adding random noise to the input images, we have used the internal representation of the same inputs to create the noise. This is the reason why we achieve a lower magnitude of ℓ_2-norm as this noise is, in fact, the sparse representation (simplified 76 linear combination of atoms) of its fellow images. The experiments are conducted on MNIST and Imagenet datasets. The comparisons with state-of-the-art methods show the efficacy of the proposed approach when we report the attack success rate and ℓ_2-norm. The salient contributions are as follows.

1. We use activation maps to generate perturbation vector.
2. These activation maps based perturbation vector is mapped to a sparse form using dictionary learning.
3. This simple approach gives promising results in terms of error and reduced ℓ_2 distance between an input image and adversarial image.
4. We explored this promising area for designing adversarial attacks not studied before.

The rest of the paper is structured as follows. Section 2 highlights major related contributions in literature. The methodology is discussed in Sec. 3. The experimental setup and results are mentioned in Sec. 4. The analysis is discussed and tabulated in Sec. 5. We propose a new black-box technique to craft adversarial examples aiming at minimizing the l_0-distance to the original image. The paper is concluded in Sec. 6.

2 Literature Review

Szegedy et al.[1] introduced the concept of adversarial examples and how they expose the vulnerability of neural networks. This introduction led to various attacks and defenses proposed by different researchers and practitioners. They proposed a box-constrained L-BFGS method to compute perturbations to fool the classifier. One of the early attacks introduced was the fast gradient method[2] exploits the gradients of the deep networks to create adversarial examples. Later, Deep Fool[14] and C&W attacks[3] were introduced, which are iterative algorithms. The C&W attacks are argued to be very strong and effective but are quite expensive in terms of computational resources. Then several methods are proposed to generate adversarial examples for high success rates and minimal size of perturbations.[2,15]

This work is inspired using activation maps or internal representations to yield adversarial attacks. Recently quite promising research focuses on the same concept. In 2019, Inkawhich et al. describe a transfer-based black-box targeted attack of feature space representations.[16] The attack is explicitly designed for transferability and achieves so by minimizing the euclidean distance of the feature space representation of a source image at a specific layer towards the representation of a target image at that layer. This method generated highly transferable adversarial examples. On the other hand, adversarial examples are also being generated by applying random transformations to input images[17] and by optimizing a perturbation over an ensemble of translated images.[18] These methods can be combined with existing techniques and are usually the extension of fast gradient sign method (FGSM).[2] Yucheng Shi et al.[8] highlighted and worked on two major problems. One is the need to add diversity in iterative trajectory and the second is minimizing repetitive noises.

The recent literature reveals that most of the methods are an extension of FGSM. The detection methods normally detect the difference in normal behavior.[19] Therefore, the activation map created after an adversarial attack should not have visible noise so that detection methods do not detect an attack as most of the defense strategies lie in noise reduction techniques.

Sparse representation is another closely related area. Some of the work regarding this includes[20] where Long et al. proposed a transfer sparse coding approach to construct robust sparse representations for classifying labeled and unlabeled images belonging to different distributions. For adapting the dictionary to achieve sparse representation, Aharon et al.[13] proposed a K-SVD method to learn the dictionary.

This research article took inspiration from all of the above-mentioned articles, which focus on using the internal representation of deep CNNs. Our goal here is to design an algorithm that yields adversarial examples with a significant decrease in noise magnitude in the ℓ_2-norm. For this purpose, we used K-SVD-based dictionary learning to convert the internal representation into sparse representation thereby, producing a minimal change in the original images to maintain imperceptibility. As far as we have reviewed, dictionary learning has been only applied for defense against adversarial attacks in literature.[21,22]

3 Methodology

The problem formulation followed by all three phases of our methodology is discussed in this section.

3.1 *Notation*

This section explains the methodology adopted in detail. Let x_l denotes the legitimate image, and y denotes the corresponding label. $F(x): X \rightarrow Y$ denotes a classifier that allots a specific class y to an image x_l. An adversarial image x_{adv} is generated, which is indistinguishable from x_l but misleads the classifier, i.e., $F(Xx_{\text{adv}} \neq Y)$. $F(x_{\text{adv}}) = y_t$ in case of targeted attacks where y_t is the desired class label we wish the classifier would predict. We evaluated our algorithms for two types of attacks type I: un-targeted and type: II targeted. This research is an attempt to imitate the deep internal features of the inputs associated with desired targeted labels. Sabour et al.[23] did something similar. They introduced a new framework for generating adversarial images that appear similar to a given input image, but whose deep internal representations mimic the

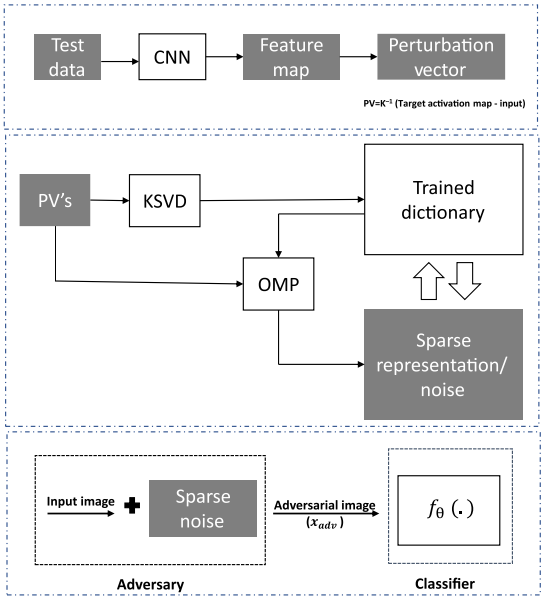

Fig. 1 Perturbation vectors are computed from CNNs. The vectors are optimized using K-SVD to obtain sparse representation. This sparse representation is added as noise to the original image to craft an adversarial example.

characteristics of natural target images. Activation maps are generated by the first layer of the CNN, which contain the information of high-level important features of the image.[24] This information is used to imitate the feature space of target images.

For better understanding, the methodology is divided into three phases: first, perturbation vector is computed using an activation map, which is detailed in Sec. 3.2. Next, these perturbation vectors are optimized using K-SVD-based dictionary learning to get the sparse representation of the perturbation vectors. The detailed description is provided in Sec. 3.3. The final step is to add sparse perturbation as noise to clean images as shown in Fig. 1.

3.2 *Perturbation Vector*

x_l denotes the clean image, and A_t is the activation map of the target image. The activation maps known as feature maps are the output of applying the filters to input, such as the input image fed to the CNN or another feature map. In this way, an activation map for a specific input image helps visualize and understand what features of the input are detected. These features are used by CNN in their classification process. At is the activation map of the target label image. We want the classifier to assign the label associated with this target image to our source image. K is the predefined Kernel. P are the perturbation vector to be computed. The perturbation vector is transformed into its sparse form and this will be added as the perturbation to the original input. This one-shot method of adding sparse noise to the input image is explained in Sec. 3.4. The CNNs are recognition algorithms and work on images. The convolutional layer detects features as it applies a filter/kernel on an image. The output generated is called the activation map. Our goal is to perturb the original image to generate a targeted activation map. This function can be expressed as

$$K(I_s + P) = A_t. \tag{1}$$

Calculating the perturbation vector from the above equation

$$P = K^{-}1(A_t) - I_s. \tag{2}$$

3.3 *K-SVD*

Sparse coding is an efficient way of data representation. The aim is to learn a set of basis functions such that each input signal can be well represented by a linear combination of minimum bases. It has a wide range of applications and has attracted increased interest due to its state-of-the-art performance. This paper uses it as the solution to the problem of redundant noises, and smaller ℓ_2 norm. We have used K-SVD based dictionary learning algorithm proposed by Aharon.[13] The K-SVD has been tailored and applied to our proposed problem to generate sparse adversarial images. The algorithm uses a SVD approach. It is a generalization of the k-means clustering method, which works iteratively between sparse coding and dictionary updating. It solves the problem by transforming an input signal X to a sparse linear combination of not more than T items and is given as

Algorithm 1 Sparse K-SVD dictionary learning for perturbations

Input: $P \rightarrow$ Matrix containing all perturbation vectors;

Result: D $\rightarrow$ Learned dictionary, $TP \rightarrow$ sparse representation of all perturbation vectors

$S \rightarrow$ Size of dictionary, i.e., total number of atoms;

$tp \rightarrow$ Sparse representation of single perturbation vector;

for $t = 1$ do iterations **do**

TP$\leftarrow$OMP(D, P);

Dictionary update stage;

for $i = 1$ do do **do**

$L =$ indices of non-zero elements in TP$(i,:)$;

$\hat{P} = P_L$;

$\hat{\text{TP}} = \text{TP}_L$;

$E = \hat{P} - \sum_{S \neq i} d_S \star tp_S^T$;

$U \star E \star V^T \approx \text{SVD}(E)$;

$d_i = U(:, 1)$;

Algorithm 2 Activation map as perturbation to learn sparse dictionary

Result: $p \rightarrow$ perturbation

Input: $A \rightarrow$ activation map;

$N \rightarrow$ total number of input image samples;

$K \rightarrow$ kernel;

$x_I \rightarrow$ clean input image;

for $i < |N|$ **do**

$p \leftarrow K^{-1}(A_i) - x_I$;

$i \leftarrow i + 1$;

Return p

$$X = D\alpha, \tag{3}$$

where α is a vector that contains coefficients of the linear combination mathematically, the problem of sparse representation is formulated as an optimization problem of finding D, which satisfies

$$\min_{D,\alpha} \|X - D\alpha\|_2 \quad \text{s.t. } \|\alpha\| < T, \tag{4}$$

where T is some predefined threshold that controls the sparseness $\|.\|_0$ denotes the ℓ_0 norm is the count of non-zero elements.

The above-formulated problem computes and performs sparse coding of every input signal using dictionary learning. The sparse representation of input signals is an NP-hard problem. The representation is therefore approximated and several pursuit algorithms are proposed to solve the approximation problem. K-SVD is flexible and normally used with greedy orthogonal matching pursuit algorithm.

This research is about the application of the K-SVD dictionary learning algorithm to solve the optimization problem in Eq. (2). Mapping the K-SVD algorithm to Eq. (2) gives us

$$\min_{D,\alpha} \|P - D\alpha\|_2 \quad \text{s.t. } \|\alpha\| < T, \tag{5}$$

where P = all perturbation vectors.

3.4 *Sparse Adversarial Image*

The final stage of our method is the computation of a sparse adversarial example. We add the sparse representation/noise of desired perturbation vector to our original legitimate image, referred to as the one-shot method. The magnitude of noise can be controlled by the value of ϵ. The magnitude is adjusted to achieve a low ℓ_2 norm. The ℓ_2 norm is the euclidean distance between clean and noisy images. The lower values of ℓ_2 norm help achieve imperceptibility and make it hard to detect such adversarial examples. The sparse adversarial image if $= s$ calculated as

$$X_{\text{adv}} = x_l + \epsilon P. \tag{6}$$

4 Experiments

The datasets used for the evaluation of the proposed algorithm are MNIST and Imagenet. We have compared our method with the C&W,[3] FGSM,[2] and Corner Search[5] methods. These are some of the most widely used and state-of-the-art methods in literature. Corner Search is relatively new and belongs to the category of sparse attacks. We have tried to cover a variety of competitive attacks i.e. gradient-based, iterative, and sparse attacks with which to compare our method. C&W[3] is a powerful attack method that generates adversarial examples with a promising minimum magnitude of ℓ_2 noise, yet its thousands of iterations make it almost impossible to use or implement due to its time complexity.[6,7]

We have used small values of epsilon (ϵ) mostly in the negative power of 10 to create our adversarial examples. The epsilon determines the strength of the attack, such that $\epsilon = 0$ means no attack. The common values used in the literature range from 0.1 to 0.8 to study the effect of the attack. The greater the value of ϵ, the higher the magnitude of the noise is, and the greater the ℓ_2-norm is. We have used smaller values as compared to the literature and show that even with such a smaller magnitude of the noise, we can create a successful attack while keeping imperceptibility low in terms of ℓ_2-norm.

4.1 *Metrics*

In this section, we list down mathematical formulae of various metrics to evaluate the performance of our proposed approach. We also discuss here the statistical standards and terms related

to the field. Machine learning algorithms or deep learning algorithms specifically learn from data. They find relationships, develop understanding, make decisions, and evaluate their performance from the training data they are given. The next section describes how accurately our chosen models perform on MNIST and Imagenet datasets. In supervised learning where the class labels to the input data are known, a deep learning algorithm learns a function that fits the training data with the goal to find a function that minimizes loss. Loss is a quantitative measure of how much your predictions differ from the actual output (label). Loss is inversely proportional to the correctness or accuracy of the model. The more the loss, the less accurate the model. First, classifiers loss on test data is calculated. The loss function compares the actual and predicted output values of a classifier and measures how well the CNN models the training data. The goal of a classifier is to minimize the loss between the predicted and actual outputs while training. A small value of loss indicates the classifier recognizes and classifies the images very well. In targeted attacks reported in the next section, the lower magnitude of loss indicates a stronger attack whereas, in the case of untargeted attacks the greater the loss, the stronger the attack is. We have also reported attack success rates. This indicates the rate at which the number of input samples is miss-classified by the classifier when it is fed with adversarial examples. The equation for mean and median ℓ_2-norm are given as follows:

$$\text{Distance}(x_l, x_a) = \|x_l - x_a\|_2, \tag{7}$$

$$\text{Median} = \text{median}(\text{distance}(x_l, x_a)|x \in X), \tag{8}$$

$$\text{Mean} = \frac{1}{N}\sum_{n=1}^{N} \text{distance}(x_l, x_a)|x \in X. \tag{9}$$

4.2 *Experimental Results*

The MNIST dataset is divided into training and testing sets. The training set consists of 50,000 images, whereas the testing set contains 10,000 black-and-white handwritten digit images. A CNN model is trained on the MNIST dataset for 50 epochs achieving 97.68% accuracy and a loss of 0.07. The adversarial images were generated with $\epsilon = 0.00005$ using the proposed strategy on the MNIST dataset.

The Imagenet dataset consists of images representing 1000 classes. The test set consists of 10,000 images. However, we evaluated our method on 1000 images from the test set after training a pretrained VGG-19 model. We have chosen 1000 images due to computational complexity. It achieved 70.2% accuracy and 1.20 error. The adversarial images were generated with $\epsilon = 0.0001$ using the proposed strategy on the Imagenet dataset. In the case of MNIST, all 10,000 images from the validation set are used for evaluation, whereas, in the case of Imagenet, 1000 images from the validation set are used for the evaluation purpose. The untargeted attacks are conducted using any sparse perturbation vectors, whereas, for targeted attacks, the specific sparse representation of the targeted label is used. The implementations from the adversarial robustness toolbox library are used to reproduce results for FGSM and C&W methods.[25] The original publicly available implementation for corner search is used for its evaluation. The metric values are computed for targeted as well as untargeted attacks. In the case of targeted attacks, following the framework from Ref. 6, we compute adversarial images for different target classes. Each individual attack result is calculated and finally, the average score of all attacks is reported. The values are reported for mean and median ℓ_2-norm. The smaller the value of ℓ_2-norm, the stronger the attack.

4.2.1 *Type I attack: untargeted*

The values of success rates mean and median for untargeted attacks are listed in Table 1. The values of mean and median ℓ_2-norm show the efficacy of the proposed approach. The mean and median magnitude of ℓ_2-norm is 0.0002 for MNIST and 0.2 for Imagenet. It is lower than any state-of-the-art attacks in the case of the MNIST dataset. In the case of Imagenet, it performs

Table 1 From left to right: type of attack, loss, attack success rate, mean, and median ℓ_2-norm our attack versus others on MNIST and Imagenet in untargeted case.

Attack method	Loss	Attack succ.	Mean ℓ_2	Median ℓ_2
Dataset: MNIST				
Fast gradient method	10.4	73%	0.1	0.1
Corner search	2.0	88%	7.9	8.8
C&W	2.8	43%	0.01	0.01
Sparse adversarial image	11.05	69%	0.0003	0.0003
Dataset: Imagenet				
Fast gradient method	3.4	72%	0.7	0.7
C&W	1.2	42%	0.0004	0.0004
Sparse adversarial image	2.26	48%	0.2	0.2

better than corner search and fast gradient method and is quite competitive compared to C&W. The ℓ_2-norm of C&W is 0.0004 in the case of Imagenet but its attack success rate and loss are lower than our sparse adversarial method.

4.3 *Type II attack: targeted*

The second column in Table 2 records the error. The lower error value implies a stronger attack. The magnitude of noise, i.e., value of ε is kept high, especially in the case of FGSM. Therefore, results recorded against FGSM and C&W are quite promising too. The high magnitude is chosen because the otherwise fast gradient method does not attack the classifier at all. Though C&W is the most effective targeted attack and reports comparable values for mean and median ℓ_2, it is quite impractical due to being time intensive.[6]

An illustration of the results can be interpreted in Fig. 2.

Table 2 The loss, attack success rate, mean, and median ℓ_2 of our attack versus MNIST and Imagenet datasets in targeted case. A lower value of the loss indicates a stronger attack.

Attack	Loss	Attack succ.	Mean ℓ_2	Median ℓ_2
Dataset: MNIST (targeted attack)				
Fast gradient method	8.7	18.3%	0.1	0.1
Corner search	46.6	1.20%	0.7	0
C&W	29.51	23%	0.005	0.005
Sparse adversarial image	16.79	15.11%	0.0002	0.0002
Dataset: imagenet (targeted attack)				
Fast gradient method	23.6	1%	1.36	1.36
C&W	18.62	1%	0.001	0.001
Sparse adversarial image	18.89	2.03%	0.2	0.2

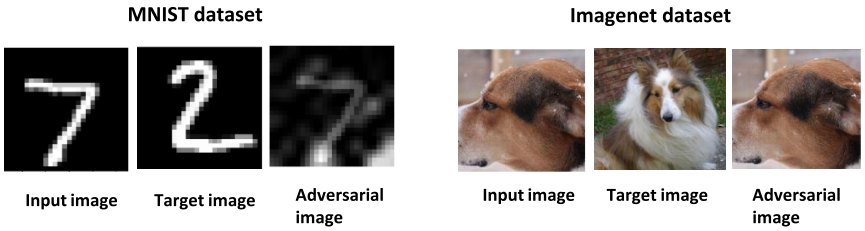

Fig. 2 Source/legitimate image, target image, and adversarial images for MNIST and Imagenet Datasets.

5 Effect of Dictionary Size on Attack Success Rate

This paper has motivated the application of dictionary learning to create adversarial examples. The approach is novel and introduces a new application area of dictionary learning. Since this paper revolves around dictionary learning and is the heart of the proposed method we dug a little deeper and investigated the effect of dictionary size on our attack method. This hyperparameter is very important as its tuning can save us a lot of time and effort. The dictionary size is determined by the number of components used to compute the dictionary. A series of experiments are conducted to explore this effect. We evaluated the performance of our attack for several possible options of the number of components and plotted a graph to analyze the effect. This analysis also helps justify the results reported in the previous section. It can be seen from Fig. 3 that the attack success rate slightly increases with the increase in dictionary size and then starts decreasing. In 2018, Moosavi et al.[22] conducted an extensive analysis of the hyper-parameters of dictionary learning when applied to an image-denoising problem. They analyzed that increasing the size of the dictionary decreases the robustness of the classifier. During a reconstruction task, the increase in dictionary size increases reconstruction because more information is retained and improves image denoising too. In our case, the attack success rate continuously decreases after the number of components = 100. It just oscillates between the number of components = 180 to the number of components = 280 but then gets stable and does not oscillate after that. The highest success rate is achieved at the number of components = 100. We have reported the results in the previous section of the number of components = 81. All these attacks are done on the test set from the MNIST dataset. This analysis highlights the following significant points: First, the main benefit can be visualized in the form that we achieve the desired result in a very small size of dictionary saving us time and cost. The results are in contrast to the conventional trend discussed in the literature. This is because here we are not using dictionary learning for denoising or reconstruction purposes, which would increase the accuracy. It is concluded that increasing the dictionary size (number of components) will retain more information but since we are learning a

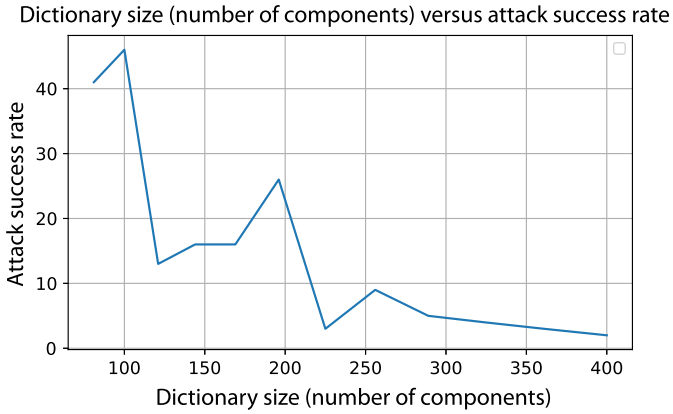

Fig. 3 The graph shows the effect of increasing dictionary size that is the number of components on attack success rate using sparse adversarial method.

dictionary of noises it will not increase the attack success rate. In the future, we can work on tuning these hyperparameters to improve our results (Fig. 3).

6 Conclusion

In this research, we have explored the application of sparse representation to create adversarial attacks against deep neural networks. We proposed a sparse adversarial image generation approach that achieves comparable performance when compared to state-of-the-art methods in terms of ℓ_2-norm, attack success rate, and loss on test data. The proposed strategy uses the activation map, which makes it possible to manipulate the internal working of the image to attack. The activation maps from the first layer of CNNs are used. These perturbation vectors are transformed into their respective sparse representations using K-SVD-based dictionary learning algorithm. The sparse representation is the sparse noise that we want to add to the original image to create an adversarial example. We can conclude from the experiments in Sec. 4.2 that the proposed approach can provide a good balance between the attack success rate and the imperceptibility of the attack. Our sparse adversarial attack shows promising results regarding the magnitude of noise measured as ℓ_2-norm. It either performs better than state of the art or is competitive in other cases. The specific perturbation is chosen randomly from the set of all perturbations generated. These perturbations can have different effects on the success rates of the attacks. In the future, we plan to optimize this approach into a more sophisticated selection approach.

References

1. C. Szegedy et al., "Intriguing properties of neural networks," in *2nd Int. Conf. Learn. Represent., ICLR*, April 14-16, Banff, Alberta (2014).
2. I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *3rd Int. Conf. Learn. Representat., ICLR*, May 7-9, San Diego, California (2015).
3. N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," *in IEEE Symp. Secur. and Privacy (SP)*, IEEE, pp. 39–57 (2017).
4. N. Papernot et al., "The limitations of deep learning in adversarial settings," in *IEEE Eur. Symp. Secur. and Privacy (EuroS&P)*, IEEE, pp. 372–387 (2016).
5. F. Croce and M. Hein, "Sparse and imperceivable adversarial attacks," in *Proc. IEEE/CVF Int. Conf. Comput. Vision*, pp. 4724–4732 (2019).
6. J. Rony et al., "Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 4322–4330 (2019).
7. N. Akhtar and A. Mian, "Threat of adversarial attacks on deep learning in computer vision: a survey," *IEEE Access* **6**, 14410–14430 (2018).
8. Y. Shi, S. Wang, and Y. Han, "Curls & Whey: boosting black-box adversarial attacks," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 6519–6527 (2019).
9. W. Dong et al., "Sparsity-based image denoising via dictionary learning and structural clustering," in *CVPR 2011*, IEEE, pp. 457–464 (2011).
10. T. S. Rao, M. V. G. Rao, and T. Aswini, "Image inpainting with group based sparse representation using self adaptive dictionary learning," in *Int. Conf. Signal Process. and Commun. Eng. Syst.* (2015).
11. H. He et al., "A novel efficient camera calibration approach based on k-svd sparse dictionary learning," *Measurement* **159**, 107798 (2020).
12. Z. Jiang, Z. Lin, and L. S. Davis, "Label consistent k-svd: Learning a discriminative dictionary for recognition," *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(11), 2651–2664 (2013).
13. M. Aharon, M. Elad, and A. Bruckstein, "K-svd: an algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.* **54**(11), 4311–4322 (2006).

14. S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard, "Deepfool: a simple and accurate method to fool deep neural networks," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 2574–2582 (2016).
15. A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial machine learning at scale," *CoRR* abs/1611.01236 (2016).
16. N. Inkawhich et al., "Feature space perturbations yield more transferable adversarial examples," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 7066–7074 (2019).
17. E. Raff et al., "Barrage of random transforms for adversarially robust defense," in *Proc. IEEE/CVF Conf. Comput. Vision and Pattern Recognit.*, pp. 6528–6537 (2019).
18. Y. Dong et al., "Evading defenses to transferable adversarial examples by translation-invariant attacks," in *Proc. IEEE/CVF Conf. Comput. Vision and Pattern Recognit.*, pp. 4312–4321 (2019).
19. C. Xie et al., "Feature denoising for improving adversarial robustness," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 501–509 (2019).
20. M. Long et al., "Transfer sparse coding for robust image representation," in *Proc. IEEE Conf. Comput. Vision and Pattern Recognit.*, pp. 407–414 (2013).
21. J. Mitro, D. Bridge, and S. Prestwich, "Denoising dictionary learning against adversarial perturbations," in *Workshops at the Thirty-Second AAAI Conf. Artif. Intell.*, pp. 364–371 (2018).
22. S. Moosavi-Dezfooli, A. Shrivastava, and O. Tuzel, "Divide, denoise, and defend against adversarial attacks," *CoRR* abs/1802.06806 (2018).
23. S. Sabour et al., "Adversarial manipulation of deep representations," in *4th Int. Conf. Learn. Represent., ICLR*, May 2-4, San Juan, Puerto Rico (2016).
24. S. R. Pericherla, N. Duvvuru, and D. B. Jayagopi, "Improving adversarial images using activation maps," in *IEEE 8th Joint Int. Inf. Technol. and Artif. Intell. Conf. (ITAIC)*, IEEE, pp. 843–847 (2019).
25. M. Nicolae et al., "Adversarial robustness toolbox v0.2.2," *CoRR* abs/1807.01069 (2018).

Maham Jahangir received her MS degree from Military College of Signal, National University of Sciences and Technology (NUST). Currently, she is a PhD scholar working under the supervision of professor Dr. Faisal Shafait at the School of Electrical Engineering and Computer Science, NUST, Islamabad, Pakistan. Her research interests include machine learning with special interests in deep neural networks and adversarial attacks.

Faisal Shafait received his PhD in computer engineering from TU Kaiserslautern, Germany, in 2008 with the highest distinction. Currently, he is working as a professor at the School of Electrical Engineering and Computer Science, NUST, Pakistan as well as director of the Deep Learning Laboratory at the National Center of Artificial Intelligence, NUST, Pakistan. He has more than 15 years of research and teaching experience in artificial intelligence with a primary focus on computer vision and deep learning. He has garnered a significant international reputation in the field of artificial intelligence by earning 11,000+ citations with i10 and h indices of more than 150 and 50, respectively. He received the prestigious IAPR/ICDAR Young Investigator Award from the International Association of Pattern Recognition in 2019 and has recently been included in the list of the world's top 2% scientists compiled by Stanford University.