Protecting JPEG Images Against Adversarial Attacks

Aaditya Prakash, Nick Moran, Solomon Garber, Antonella DiLillo and James Storer

Brandeis University

{aprakash,nemtiax,solomongarber,dilant,storer}@brandeis.edu

Abstract

As deep neural networks (DNNs) have been integrated into critical systems, several methods to attack these systems have been developed. These adversarial attacks make imperceptible modifications to an image that fool DNN classifiers. We present an adaptive JPEG encoder which defends against many of these attacks. Experimentally, we show that our method produces images with high visual quality while greatly reducing the potency of state-of-the-art attacks. Our algorithm requires only a modest increase in encoding time, produces a compressed image which can be decompressed by an off-the-shelf JPEG decoder, and classified by an unmodified classifier.

1 Introduction

Deep neural networks (DNNs) have shown tremendous success in image recognition tasks, even surpassing human capability [1]. DNNs have become components of many critical systems, such as self-driving cars [2], medical image segmentation, surveillance, and malware classification [3]. However, recent research has shown that DNNs are vulnerable to adversarial attacks, in which minute, carefully-chosen image perturbations can result in misclassification of the image by the neural network [4]. In most cases, this change is imperceptible to humans (the resulting image is visually indistinguishable from the original).

Current adversarial attacks take advantage of the *over-completeness* of the pixel representation of images – that is, storing images as an array of values results in storing much more information than is required to recognize the content of an image. Traditional image compression techniques, like JPEG, make assumptions about the human perception of natural images in order to compress them. Since adversarial attacks are imperceptible to the human eye, they can be viewed as hiding themselves in the over-complete part of the pixel representation of images. Removing these adversarial perturbations is thus precisely a lossy image compression problem: by effectively encoding natural images, image compression techniques can quantize away non-perceptible elements of an image, effectively blocking attacks that rely on imperceptible perturbations.

While the research in defending against such adversarial perturbations is still in its infancy, it has been shown that JPEG naturally removes some of these imperceptible perturbations, thereby restoring the original classification of the image [5, 6]. More advanced attacks, however, are robust against JPEG compression. In this work, we study the nature of the perturbations generated by adversarial methods, as well as their effect on classification. Additionally, we propose a solution which

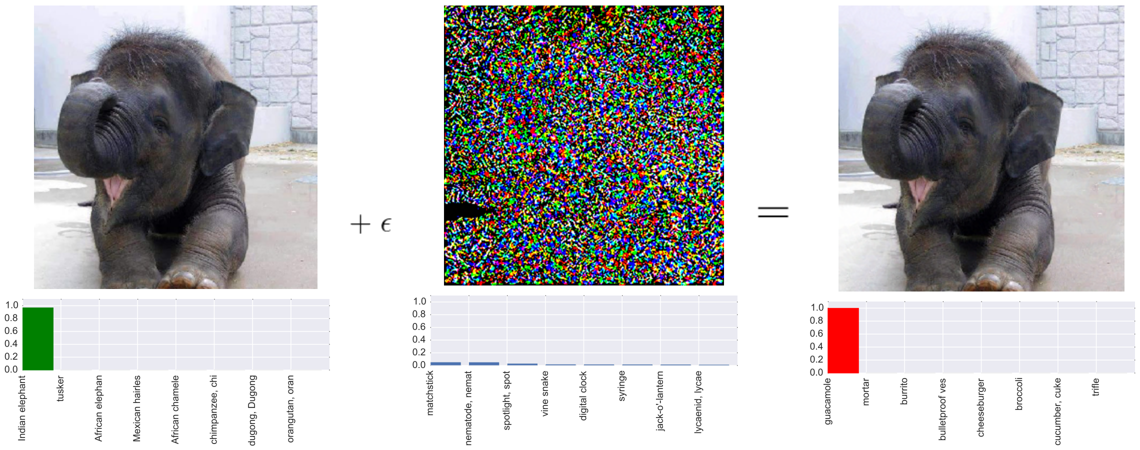

Figure 1: Left: Original Image, Center: Perturbed Noise, Right: Adversarial Image

yields a standard JPEG-decodable image, but which allows the recovery of accurate classifications at a higher rate than off-the-shelf JPEG compression.

Several techniques have been proposed which attempt to reduce the feasibility of generating adversarial images [7, 8]. Most of these methods involve augmenting the DNN training process to create a more robust classifier. However, these defenses often fail against more advanced attacks, and come at the cost of more computationally expensive training. A detailed discussion of proposed defenses is found in section 4.

We therefore seek a practical defense which hardens an image against such attacks as it is being compressed, and does not require changes to the classification model or its training process. Such a defense will aim to dampen the noise added by the adversary so as to allow the classifier to recover the correct classification. While this kind of defense may not be the most robust solution, it comes at a low computational cost, and the ubiquity of the JPEG decoder makes it easily implementable in practice.

Standard JPEG compression is able to provide some measure of defense against adversarial attacks. Since adversarial perturbations are often high frequency and low magnitude, the quantization step of JPEG frequently removes them. When the noise added by the adversary is removed or substantially modified, and the resulting image no longer fools the classifier. However, as the severity of quantization increases, important features of the original image are also lost, and the model's ability to correctly classify the resulting image deteriorates regardless of any adversarial perturbation. This trade-off limits the effectiveness of a simple JPEG-based defense, as defending against more sophisticated attacks requires harsh quantization that renders the output image difficult to classify.

We present a novel defense based on an adaptive quantization scheme which intelligently quantizes different blocks at different levels. Our analysis of adversarial images generated by various attacks shows that attack methods will tend to perturb all regions of an image with equal probability. As a result, the added noise is often outside the object of interest in the image. We thus tested the MSROI algorithm [9], which identifies multiple salient regions of the image, and quantizes those regions which are not salient more aggressively. MSROI adaptive quantization dampens a significant portion of the adversarial noise, but does so without deteriorating the quality of salient regions, thereby disrupting the attack while allowing for recovery of the original classification. However, initial experiments demonstrated that the unmodified MSROI algorithm is also susceptible to adversarial noise, weakening the accuracy of the generated saliency maps. Therefore, we present a more robust method of generating a saliency map, which is able to recover object saliency even in the presence of adversarial noise, without impacting performance on unperturbed images. We show that our proposed defense is successful against several state-of-the-art attack methods using an off-the-shelf classification model. We further demonstrate that our proposed technique greatly improves upon the results of a standard JPEG encoder.

2 Background

A deep neural network can be viewed as a parameterized function, F_θ, which maps $x \in \mathbb{R}^n$ to $y \in \mathbb{R}^m$ i. e. $F_\theta(x) = y$ where θ represents the model parameters. Feedforward neural networks are composed of multiple layers of successive operations, each applied to the output of the previous layer. For a deep neural network containing l layers, the final network is a function composition of l simpler functions i.e $F = F_1 \circ F_2 \circ \cdots F_l$. These networks are trained by minimizing some loss function, $\mathcal{L}$, which is a differentiable measure of the classification accuracy on the training data. The parameters of the model, θ, are learned by stochastic gradient descent (SGD).

Most adversarial attack methods use the gradient of the image pixels with respect to the the classification error to generate the desired perturbation. In order to compute this gradient, an attacker needs to have access to the internal model and parameters, θ. These kinds of attacks are called white box attacks, as the adversary needs access to the targeted model. White box attacks can fool many state-of-the-art convolutional neural networks. Many popular deep learning models are publicly available, making them vulnerable in practice to these attacks.

It is also possible to attack deep networks without access to the model parameters. Such attacks are called black box attacks, and they begin by training a secondary model to match the outputs of the target model. This secondary model can then be attacked using white box methods, and the resulting adversarial images are transferable to the target network [10].

3 Adversarial Attacks

Formally, the paradigm of adversarial attacks can be described as follows. Let x be an input image to a classifier. Then, the class label (the model's evaluation of what object the image contains), c, for the image x is: $c = \arg\max_{\widetilde{y}} F(x|\theta)$.

An adversarial attacker takes the image x and adds small perturbation δ_x, to obtain an adversarial image $x' = \mathrm{x} + \delta_x$. The attack is considered to be successful if, for a small δ_x, the class label of the adversarial image (c'), is not the same as that of original image (c). There is no consensus on the exact definition of “small” for δ_x; but attacks are generally considered successful when the difference between x and x' is indistinguishable to the human eye. Some attacks are noticeable to humans, but do not change our perception of the class label of the image, and these attacks are also considered successful. Attacks can be targeted to induce a specific, chosen class label, $\hat{c}$ such that $c' = \hat{c}$, or untargeted, meaning that they seek to induce any class label, as long as it does not match the original classification, i.e., $c' \neq c$.

In most cases, untargeted attacks are performed by iterating a targeted attack over all classes and choosing the class which requires the smallest perturbation. As our work focuses on defense, we will consider untargeted attacks, and therefore our methods should also be effective against targeted attacks.

Adversarial attacks work by exploiting the design of the model and the limitations of SGD. Just as gradients of the model parameters (θ) with respect to the loss function are calculated during SGD training, an attacker can compute gradients of the input pixels with respect to the loss function. These gradients instruct the attacker on how to modify the image as to change the model's classification. In the following sections, we will briefly describe each of the adversarial attacks used in our experiments.

Fast Gradient Sign Method (FGSM) [4] uses the sign of the gradient of the loss function to determine the direction of the perturbation δ_x.

$$x' = x + \epsilon \times \text{sign}(\nabla \mathcal{L}(F(x|\theta), \widetilde{y}))$$

Where ϵ is a small value which controls the magnitude of perturbation, chosen through trial and error. To ensure a sufficiently small δ_x, FGSM optimizes the L_∞ distance between x and x', and thus tries to minimize the maximum change to any of the pixels. It has been shown that with small values of ϵ, most deep neural networks trained on CIFAR-10 and MNIST can easily be fooled [4]. FGSM is a particularly efficient attack method, as it requires only a single computation of the gradients of the input pixels. However, FGSM does not always produce robust results.

Iterative Gradient Sign Method (IGSM) [11] is an improvement over FGSM. Starting with the original image, FGSM is applied iteratively until a satisfactory image is found, and, at each step, the generated adversarial image is clipped so that the generated image is within an L_∞ ϵ-neighborhood of the original image.

$$x'_0 = x, \qquad x'_{N+1} = \text{Clip}_{x,\epsilon}\Big\{x'_N + \alpha \times \text{sign}(\nabla \mathcal{L}(F(x|\theta), \widetilde{y}))\Big\}$$

Here, α is the amount of perturbation added per iteration, which is usually very small compared to the ϵ used in FGSM. This allows the attacker to find adversarial images which are closer to the original image than those found by FGSM [11].

Gradient Attack (GA) is where a single step is taken, as in FGSM, but the step is taken in the exact direction of the gradient, rather than the direction of the element-wise sign of the gradient. The gradient step is normalized to have unit length.

$$\mathcal{G} = \nabla \mathcal{L}(F(x|\theta), \widetilde{y}), \qquad x' = x + \epsilon \times \frac{\mathcal{G}}{||\mathcal{G}||_2} \qquad \text{where, } ||.||_2 \text{ denotes } L_2 \text{ norm}$$

Deep Fool makes the simplifying assumption that the classifier to be targeted is a linear model which uses a hyperplane to separate the classes. It tries to find an adversarial image by moving in the direction orthogonal to the linear decision boundary. Because the classifier is not actually linear, it is necessary to iterate this process until an adversarial image is found. Deep Fool is an untargeted attack and optimizes for L_2 (PSNR) instead of L_∞, unlike FGSM. Experiments reveal that perturbations generated by Deep Fool are less perceptible than those generated by other methods.

Jacobian-based Saliency Map Attack (JSMA) [12] uses an adversarial saliency map, which indicates the pixels in the image that the adversary should increase in value in order to fool the classifier. Let $\mathcal{J}(x)$ be the Jacobian of the output of the classifier with respect to the image x and c' be the targeted class. Then, the adversarial saliency map, $S(x_j, c')$, is given by:

$$S(x_j, c') = \begin{cases} 0, & \text{if } \mathcal{J}'_c(x_j) < 0 \text{ or } \sum_{i \neq c'} \mathcal{J}_i(x_j) > 0 \\ \mathcal{J}_{c'}(x_j) | \sum_{i \neq c'} \mathcal{J}_i(x_j)|, & \text{otherwise} \end{cases}$$

Where $\mathcal{J}_i(x) = \frac{\partial F_i(x)}{\partial x}$. The attacker tries to find the pixel x_j such that $S(x_j, c')$ is maximized. Then, those features are perturbed by ϵ. This process is iterated until the model's prediction for x is c'. JSMA produces alterations that are imperceptible to humans, but it is considerably slower than other techniques.

L-BFGS [13] is one of the earliest proposed techniques to generate adversarial images. The authors formulate the task of finding an adversarial image x' given an image x as a box-constrained optimization problem. The constraint is the class label and the function being minimized is the L_2 distance between the image x and the generated adversary x'. In practice, it is hard to find a satisfactory minimum under such strict constraints, i.e $\arg\max_{\widetilde{y}} F(x|\theta) \neq c$, where c is the true class label of the image. Instead, they reduce the minimization to:

$$\text{minimize} \quad \alpha ||x - x'||_2^2 + \mathcal{L}((F(x'|\theta), c'), \quad \text{subject to } x' \in [0, 1]^n$$

where α is an arbitrary value chosen by line search. This optimization is often carried out using L-BFGS, which is slower than other adversarial techniques.

4 Defenses

Defenses against adversarial systems are either specific to a given attack or attack-agnostic. We briefly discuss some well-known defenses from both categories.

One of the techniques to defend against adversarial images is to include such images as part of the training set, thereby encouraging the model to classify the perturbed images with same label as the original image. This method is very inefficient as the space of possible perturbations is very large and it would be not feasible to train a model long enough to become robust against all possible noise [14]. Another limitation of this method is that adversarial images trained to fool other networks are not defended against. It has been shown that adversarial images are transferable between models [15], and such transferred images are generally difficult to defend against. Adversarial systems which include random perturbations to the image before applying an attack technique are generally robust against this kind of defense.

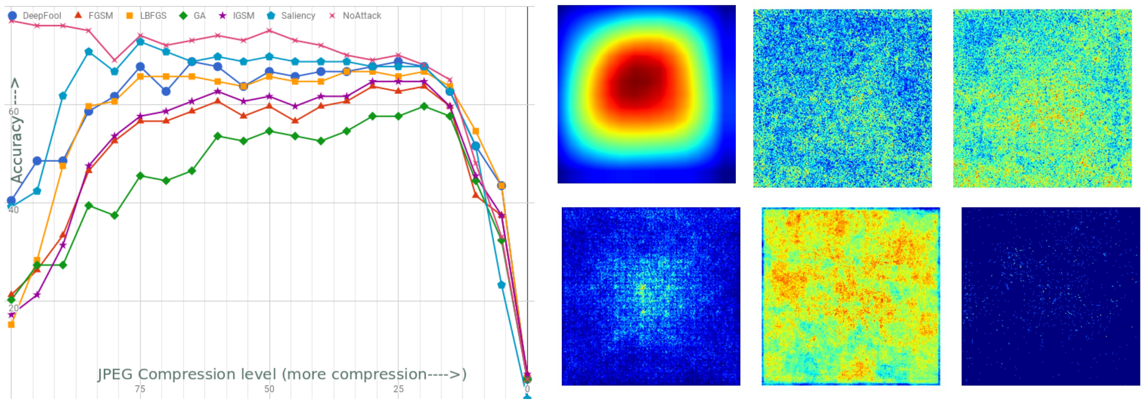

Figure 2: (a) Classification accuracy on adversarial images is recovered as quantization increases, but drops when quantization becomes too aggressive. (b) Clockwise from top left; average MSROI saliency map for original images, average adversarial perturbations for five attack methods (IGSM, FGSM, JSMA, Deep Fool, GA). Note that noise intensity is not concentrated in salient regions.

One of the most effective techniques towards defending against adversarial perturbations is called Defensive Distillation. In this scheme, the outputs of the classifier are used to train another model, called the distilled model. The distilled model is trained to match the output of the last layer (softmax layer) of the original classifier. This distilled model benefits from a smoother distribution of target outputs as compared to the all-or-nothing ground truth labels. This model has been shown to be robust against some of the less efficient attacks. However, this comes at an expense of training a new model.

4.1 Limitations of JPEG

Recently, it has been shown that image transformations such as Gaussian blur, Gaussian noise, change of contrast and brightness, and image compression like JPEG are able to recover some portion of correct classifications when applied to adversarial images [11]. The impact of JPEG compression was further studied by [6] and [5]. Their studies were limited to less effective attack techniques, such as FGSM, but demonstrated the baseline efficacy of JPEG. We show that standard JPEG does not defend against newer and more robust attacks, and analyze some of the limitations of pure JPEG for adversarial defense.

JPEG's sucsess in defending against some attack models is due to the quantization of high-frequency signals in the image. The high frequencies will contain some portion of the perturbations added by the attacker, and quantization will lessen their impact. However, adversarial perturbations in lower frequencies remain, and the more aggressive quantization (i.e. JPEG with lower Q values) required to remove them results in lower image quality, and makes it difficult to recover the original classification.

To maximize the effectiveness of our defense, while still preserving sufficient quality to allow for correct classification, we seek to strongly quantize those regions of the image which are not salient to the true class, while weakly quantizing salient regions. We present a method for achieving this in the following section.

5 Semantic Quantization

We propose a transformation $T(x)$ of a given image x which quantizes the image in the frequency domain like standard JPEG. However, unlike JPEG, we determine the salient regions of x , i.e. regions for which the presense of an object is likely, and quantize those regions at higher bit-rates than non-salient regions. We use a convolutional neural network to generate a saliency heat map, which guides quantization.

The center plot on Figure 3 shows the map of the perturbations added by an adversarial system (IGSM). The adversary has made modifications to every image region (except where the image is saturated), even though the elephant is in the center of the image. Our method takes advantage of this, and strongly quantize the image blocks outside of the salient region.

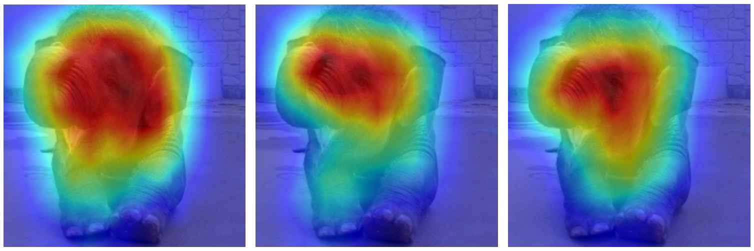

Figure 3: Left: MSROI on original image, Center: MSROI on adversarial image, Right: Augmented MSROI on adversarial image

Convolutional neural networks (CNNs) have been used to detect and locate objects in a given image [16, 17]. However, these works seek to genereate a hard bounding box around the objects which does not take into consideration partial or occluded objects. They are also limited in the number of classes they can detect, which ranges between 20 to 40. Saliency detection techniques can solve these issues, but such techniques are limited in their ability to detect multiple objects, and the identified salient region may only contain a limited subset of the objects in the image. To address these shrotcomings, we utilize MSROI [9], a CNN designed to retrieve all salient regions and provide a soft boundary over the image.

MSROI generates a heat map based on its confidence over all classes and their corresponding activation values. At each layer, l, of the network, MSROI has independent filters for each class in the dataset. Thus, summing the filter responses of each class over their respective filters represents the confidence of that class. For an image x at a given location of (i, j) and C classes, MSROI map is defined as:

$$M(i,j) = \sum_{c \in C} \begin{cases} \sum_l f_l^c(i,j) & \text{if} \sum_l \sum_{x,y} f_d^c(i,j) > \mathcal{T} \\ 0 & \text{otherwise} \end{cases}$$

When the confidence of the model on a specific class is below a threshold $\mathcal{T}$ (determined through experimentation on validation data), the class is not represented.

6 Augmented MSROI

MSROI was trained on natural images, but in this task it will be applied to an adversarial image. This means that the incorrect adversarial classification will be represented in the saliency map. This presents an issue, as the adaptive quantization

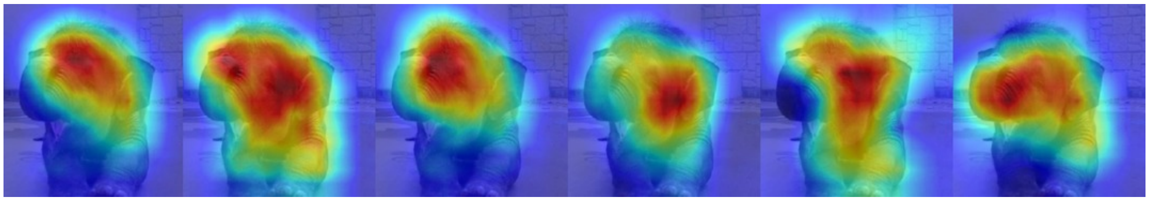

Figure 4: Combination of various jittered inputs

will try to preserve those areas which are salient to the incorrect classification. Thus, performing standard MSROI retains some adversarial noise which could have been quantized away. To counteract this, we augment the MSROI algorithm by generating several small random perturbations of the image and the intermediate activations of the MSROI network, and calculating a saliency map $\widetilde{M}$ for each such perturbation:

$$\widetilde{M}(i,j) = \sum_{c \in \hat{C}} \sum_{l} (a_l^c(i,j)) \quad \text{and} \qquad a_{l+1}^c = f_l(a_l^c(i,j) + \Delta)$$

where Δ is a random perturbation and $\hat{C}$ is subset of C found similarly using $\mathcal{T}$ as shown with MSROI. We take an average of $\widetilde{M}$ maps over several perturbations to generate a final map, which we term Augmented MSROI (Aug-MSROI). This process of averaging over many perturbations gives softer localization which is less vulnerable to attack. The number of perturbations to average over can be found empirically. Figure 4 shows several individual $\widetilde{M}$ maps which are combined to create the final heat map. In Figure 3 we show the difference between standard MSROI and Aug-MSROI. Most of the head of the baby elephant is retrieved in the Aug-MSROI heat map but not in the standard MSROI.

Once the semantic heat map is obtained, we use the technique proposed in [9], which quantizes different 8×8 blocks at different levels and combines them to obtain a final JPEG image. This final image retains high quality in salient regions, and is heavily quantizeed in non-salient regions. Strong quantization in non-salient regions is often sufficient to reverse the impact of adversarial perturbations, and high quality in salient regions allows recovery of the correct classification. Our modifications of the standard MSROI algorithm result in only a modest increase in encoding complexity, and, like standard MSROI, the resulting compressed image can be decompressed by a standard off-the-shelf JPEG decoder.

7 Experiments

We experimented with several adversarial attacks as described in section 3. We used Residual Network (ResNet-50) [1], a state-of-the-art deep CNNs as the classification model. Our test images are scaled and cropped to 256×256, and are drawn from the ImageNet dataset, which contains 1000 classes. We compare our results against the original MSROI technique, as well as off-the-shelf JPEG.

7.1 Metrics for comparison

Accuracy is the most common metric for assessing performance of deep neural networks on image classification tasks. Most publications report ‘Top-5’ accuracy, which is calculated by counting the number of instances for which the image’s true label is in the top five predicted labels. For the purposes of adversarial attacks, we consider an attack to be successful if the model's top predicted class c' for adversarial image x' is not the same as the top predicted class for the original image x. This is a very generous success metric for the adversary, as the predicted class label for most of the adversarial images are within the top five of the original image.

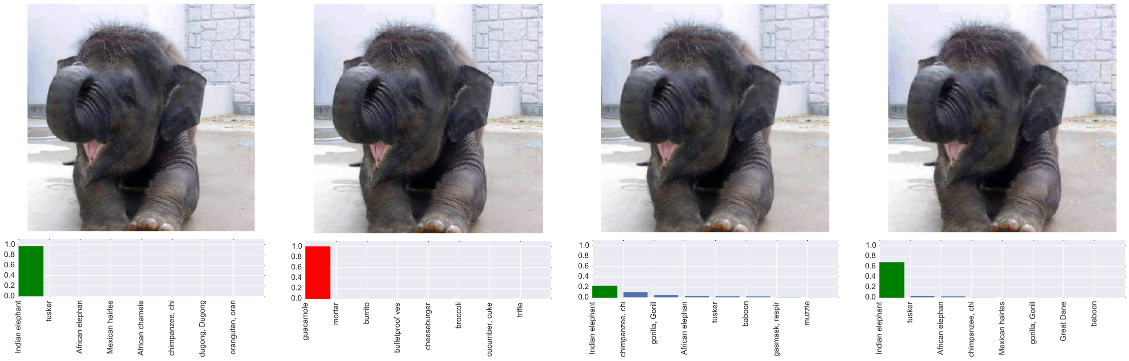

Figure 5: Original Image, Adversarial Image, JPEG, Aug-MSROI

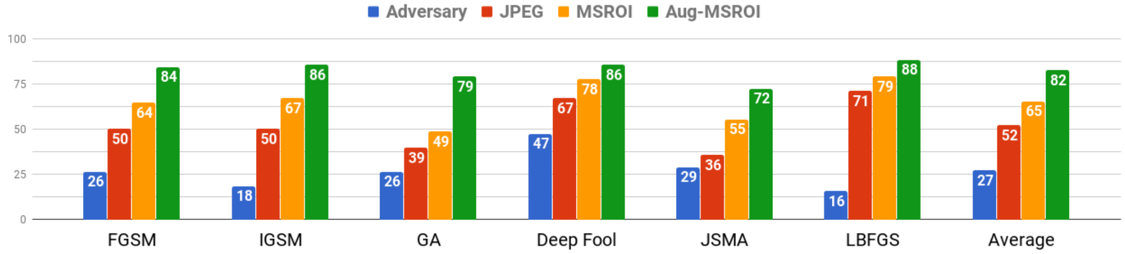

Figure 6: Classification accuracy of images on various attacks

8 Results

Figure 5 shows a typical example of the results for qualitative comparison. In this case, while JPEG was also able to retrieve the original class, it does so with significantly lower confidence. Figure shows the Table 1 shows the quantitative results averaged across 300 images of ImageNet.

	Accuracy↑	MSE↓	SSIM↑	PSNR↑
JPEG	52	16.55	0.62	24.33
MSROI	65	16.13	0.64	24.61
Aug-MSROI	82	10.92	0.86	27.36

Table 1: Results averaged across 300 images.

The JPEG quantization level was chosen to achieve the highest accuracy. For MSROI, we applied the technique described in [9] and set the quantization level to keep the final image quality as close as possible to the JPEG version. The results are shown in Figure 2(a). Our model improved classification accuracy while maintaining the perceptual quality as evaluated by PSNR, MSE and SSIM.

9 Conclusion

We have presented an augmentation of our previous MSROI algorithm [9] to employ semantic JPEG compression as an effective defense against state of the art adversarial attacks. Our experiments suggest that JPEG compression employing augmented MSROI produces compressed images, which, when decompressed, have high visual quality while at the same time preserving the accuracy of classification by a neural network. A key advantage of our approach is that, with only a modest increase in encoding time, like standard MSROI, it produces compressed images that can be decompressed by any off-the-shelf JPEG decoder.

References

[1] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).

[2] Mariusz Bojarski et al. "End to End Learning for Self-Driving Cars". In: *arXiv preprint arXiv:1604.07316* (2016).

[3] Razvan Pascanu et al. "Malware classification with recurrent networks". In: *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE. 2015.

[4] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. "Explaining and Harnessing Adversarial Examples". In: *CoRR* abs/1412.6572 (2014).

[5] Nilaksh Das et al. "Keeping the Bad Guys Out: Protecting and Vaccinating Deep Learning with JPEG Compression". In: *CoRR* abs/1705.02900 (2017).

[6] Gintare Karolina Dziugaite, Zoubin Ghahramani, and Daniel M. Roy. "A study of the effect of JPG compression on adversarial images". In: *CoRR* abs/1608.00853 (2016).

[7] Nicolas Papernot et al. "Distillation as a Defense to Adversarial Perturbations Against Deep Neural Networks". In: *2016 IEEE Symposium on Security and Privacy (SP)* (2016).

[8] Takeru Miyato et al. "Distributional Smoothing with Virtual Adversarial Training". In: 2015.

[9] Aaditya Prakash et al. "Semantic Perceptual Image Compression Using Deep Convolution Networks". In: *2017 Data Compression Conference (DCC)* (2017).

[10] Nicolas Papernot et al. "Practical Black-Box Attacks against Deep Learning Systems using Adversarial Examples". In: *CoRR* abs/1602.02697 (2016).

[11] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. "Adversarial examples in the physical world". In: *CoRR* abs/1607.02533 (2016).

[12] Nicholas Carlini and David A. Wagner. "Towards Evaluating the Robustness of Neural Networks". In: *2017 IEEE Symposium on Security and Privacy (SP)* (2017).

[13] Christian Szegedy et al. "Intriguing properties of neural networks". In: *CoRR* abs/1312.6199 (2013).

[14] Florian Tramèr et al. "The Space of Transferable Adversarial Examples". In: *CoRR* abs/1704.03453 (2017).

[15] Yanpei Liu et al. "Delving into Transferable Adversarial Examples and Black-box Attacks". In: *CoRR* abs/1611.02770 (2016).

[16] Dumitru Erhan et al. "Scalable Object Detection Using Deep Neural Networks". In: *2014 IEEE Conference on Computer Vision and Pattern Recognition* (2014).

[17] Rui Zhao et al. "Saliency Detection by Multi-Context Deep Learning". In: *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. June 2015.