Explaining Classifiers using Adversarial Perturbations on the Perceptual Ball

Andrew Elliott∗
Uni. of Glasgow/Turing Institute
andrew.elliott@glasgow.ac.uk

Stephen Law∗
UCL/Turing Institute
stephen.law@ucl.ac.uk

Chris Russell†
Amazon T¨ubingen
cmruss@amazon.com

Abstract

We present a simple regularization of adversarial perturbations based upon the perceptual loss. While the resulting perturbations remain imperceptible to the human eye, they differ from existing adversarial perturbations in that they are semi-sparse alterations that highlight objects and regions of interest while leaving the background unaltered. As a semantically meaningful adverse perturbations, it forms a bridge between counterfactual explanations and adversarial perturbations in the space of images.

We evaluate our approach on several standard explainability benchmarks, namely, weak localization, insertion-deletion, and the pointing game demonstrating that perceptually regularized counterfactuals are an effective explanation for image-based classifiers.

1. Introduction

We address the gap between counterfactual explanations [53] and adversarial perturbations [48], and show why minimal changes in image data that results in a change in classifier response does not result in semantically meaningful alteration. One might hope that the smallest edit to alter classifier response of an image labeled as bird should alter the bird pixels, but in practice adversarial perturbations make non-local changes that break the classifier. We show how penalizing changes in the mid-level classifier response with a perceptual loss stops this breakage and instead results in semantically meaningful changes that highlight the extent of objects in images (see Figs. 1.2).

Outside of computer vision [53], counterfactual explanations are a popular method in explainable AI. They find the smallest change needed to alter the decision of a classifier, and on tabular data, can give explanations such as:

“The loan was denied as your income was £30,000. If it had been £45,000, you would have been offered a loan.”

∗Equal contribution
†Much of this work was done prior to joining Amazon.

Figure 1. Object localization. From left to right: Original image; Magnitude of the perceptual perturbations; Dominant connected component and the bounding box from automatic object detection. Despite the flowers our method highlights the butterfly as salient.

The close relationship between adversarial perturbations and counterfactual explanations follows from the definitions in philosophy and folk psychology of a counterfactual explanation as answering the question “What is a minimal change that would result in a different outcome?” When applied to classifiers, these counterfactual explanations are simply adversarial perturbations, by a different name. As such, it is interesting to ask, why adversarial perturbations don’t work as explanations: Why are they imperceptible? and: Why don’t they localize on objects?

Two compelling arguments for the existence of imperceptible adversarial perturbations in images have been offered. The first due to [16] remarks that they are simply an artifact of a high-dimensional space, and thus, it is entirely expected that a small perturbation of every pixel can add up to a large change in the classifier response, and that in fact the same behavior is found in linear classifiers.

A second argument attempts to understand why sparse (potentially single pixel) attacks exist and attributes the effectiveness of adversarial perturbations to exploding gradients. ‘Exploding gradients’ refers to the phenomenon where changes in functional response grow exponentially with the depth of the network, relative to a change of input of fixed magnitude. These exploding gradients are an issue known to afflict the learning of Recurrent Neural Networks [30], and the deep networks common to computer vision. This phenomenon occurs because, by construction, neural networks form a product of (convolutional) matrix operations interlaced with nonlinearities; and for directions/locations
in which these nonlinearities act approximately linearly, the eigenvalues of the Jacobian can grow exponentially with depth. While this phenomenon is well-studied in the context of training networks with remedies such as normalization and gradient clipping, the same phenomenon occurs when generating adversarial perturbations. As such, a carefully chosen small perturbation to have an extremely large effect on the response of a deep or recurrent classifier.

To explore how these arguments fit together, and which explanation accounts for the familiar behavior of adversarial perturbations, we propose a simple novel regularization that bounds the exponential growth of the classifier response by regularizing the perceptual distance between the image and its adversarial perturbation.

One common criticism of adversarial perturbation is that the generated images lie outside the manifold of natural images, and if we could sample from the manifold, our adversarial perturbations would be both larger and more representative of the real world. Restricting adversarial perturbations to this manifold should limit the impact of exploding gradients – if samples are drawn from this space then a well-trained classifier should implicitly reflect the smoothness of the true labels of the underlying data distribution.

While it is believed that the manifold of natural images is low dimensional, characterizing this manifold outside of handwritten digits has proven extremely challenging. Our approach provides a complementary lightweight alternative. Rather than attempting to characterize the manifold, we penalize search directions that exploit exploding gradients as these encourage movement off the data manifold when searching for minimal adversarial perturbations.

We propose a novel regularization for adversarial perturbations based around the perceptual loss. Our new perturbations tend to highlight objects and regions of interest within the image (see Fig. 1). We evaluate on several standard explainability challenges for image classifiers and further validate using the sanity checks of [1].

2. Prior work

Numerous approaches to adversarial perturbations have been proposed previously. These can loosely be divided into white-box approaches that assume access to the underlying nature of the model and black-box methods which do not. The search for an adversarial perturbation is often formulated as trying to find the closest point to a particular image, under the $\ell_\infty$, $\ell_1$ or $\ell_2$ norm that takes a different class label. Numerous defenses have been proposed but they can often be circumvented.

Other works that add additional constraints to the perturbation to try to make the generated images more plausible. Such works restrict the space of perturbations considered by trying to find an adversarial perturbation that confounds many classifiers at once, or is robust to image warps. Other approaches considered only a single image and single classifier, but restricted adversarial perturbations to lie on the manifold of plausible images. The principal limitation of these approaches is that they require a plausible generator of natural images, something that is achievable with small simple datasets such as MNIST but currently out of reach for even the 224 by 224 thumbnails used by typical ImageNet classifiers.

Adversarial Perturbations and Counterfactuals There are substantial works relating adversarial perturbations and counterfactual explanations. This relationship follows from the definitions in philosophy and folk psychology of a counterfactual explanation as answering the question “What could have been different in order for outcome A to have occurred instead of B?” With full causal models of images being outside our grasp, such questions are commonly answered using Lewis’s Closest Possible World semantics, rather than Pearl’s Structured Causal Models. Under Lewis’s framework, an explanation for why an image is classified as ‘dog’ rather than ‘cat’ can be found by searching for the most similar possible world (i.e. image) which is assigned the label ‘cat’ by the classifier.

Conceptually, this is no different to searching for an adversarial perturbation sampled from the space of possible images. Several approaches have been proposed that ei-
Adversarial Perturbations and Gradient Methods  The majority of methods in the explainability of computer vision tend to be gradient or importance-based methods that assign an importance weight to every pixel in the image; every superpixel; or to mid-level neurons. These gradient methods and adversarial perturbations are strongly related. In fact, with most modern networks being piecewise linear, if the found adversarial perturbation and the original image lie on the same linear piece, the difference between the original image and closest adversarial perturbations under the $\ell_2$ norm is equivalent to the direction of steepest descent, up to scaling. As such, $\ell_2$ adversarial perturbations can be thought of as a slightly robustified method of estimating the gradient, that takes into account some local non-linearities.

Of the pure gradient-based approaches, [40] calculated the output gradient with respect to the input image to create a saliency map giving fine-grained, but potentially less interpretable results. Other gradient approaches include SmoothGrad [42] which stabilizes the saliency maps by averaging over multiple noisy copies, and Integrated Gradients [47] which accumulates gradients seen when perturbing an empty image to the input image.

CAM based approaches [38, 57] sum the activation maps in the final convolutional layer of the network. These small activation maps are up-sampled to obtain a heatmap that highlights particularly salient regions. Grad-CAM is a generalized variant which finds similar regions of interest to the perturbation based approaches [38]. Recently, [34] introduced a framework that tries to unify these various gradient approach by proposing NormGrad, which aggregates the spatial gradient contributions of individual layers.

Perturbation methods estimate the local sensitivity over a larger range than gradient methods. For example, [55] applied constant occlusion masks to different input patches repeatedly to find sensitive regions. LIME [35] constructed a linear model using the responses obtained from perturbing super-pixels. The recent work on Extremal Perturbation [10] estimates an optimal mask of the image to occlude which gives a maximal effect on the network’s output.

Various experiments have been proposed to test explanations including the pointing game [33, 38, 56], the weakly supervised object localization task [5, 12] and the insertion and deletion game [33, 54]. In particular, [1] developed experiments to test the suitability of saliency methods. A number of existing saliency techniques have been evaluated using these experiments, including: NormGrad [34], Extremal Perturbation [10], Gradient [40], RISE [33], Grad-CAM [38], SmoothGrad [42], GuidedBackprop [45], Integrated Gradients [47], Deconvolution [55], and Excitation Backpropagation [56]. We evaluate our approach on all these tests and compare against standard methods.

3. Methodology

We consider a classifier $C(\cdot)$ that takes an image $x$ as input, and returns a $k$ dimensional confidence vector.

For classifiers that assign a single class to each image, we assume the classifier $C(\cdot)$ assigns the label $i = \arg \max_j C_j(x)$ to the image $x$. Given image $x$ classified as label $i$ we consider the scalar multi-class margin:

$$M_i(x') = C_i(x') - \max_{j \neq i} C_j(x')$$

and note that $M_i(x') \leq 0$ if and only if $C(\cdot)$ does not assign label $i$ to image $x'$.

For classifiers $C(\cdot)$ that assign multiple classes to a single image (e.g. pointing game (Sec. 4.4)), we assume that the classifier $C(\cdot)$ assigns the labels $I = \{\forall j : C_j(x) > 0\}$
to the image \( x \). For each \( i \in I \), we are interested in the per label classifier response, and instead define the margin as:

\[
M_i(x') = C_i(x')
\]

Again, \( M_i(x') \leq 0 \) if and only if \( C(\cdot) \) does not assign label \( i \) to image \( x \). In both cases, an adversarial perturbation \( x' \) can be found by minimizing:

\[
(M_i(x') - T)^2
\]

where \( T \) is a target value smaller than zero. It is well-known [44] that minimizing a loss of the form:

\[
(M_i(x') - T)^2 + \lambda||x' - x||_2^2
\]

is equivalent to finding a minimizer of Eq. (3) that lies in the ball defined by \( ||x - x'||_2 \leq \rho \) for some \( \rho \). As such, minimizing this objective for an appropriate value of \( \lambda \) and \( T \) is a good strategy for finding adversarial perturbations of image \( x \) with small \( \ell_2 \) norm.

Writing \( C^{(l)}(x) \) for the classifier response of the \( l \)th layer of the neural network, we consider the related loss:

\[
(M_i(x') - T)^2 + \lambda' \sum_{l \in L} ||C^{(l)}(x') - C^{(l)}(x)||_2^2 + \lambda||x' - x||_2^2
\]

defined over a set of layers of the neural network \( L \).

The second term is the perceptual loss of [20], and minimizing this objective is equivalent to finding a minimizer of Eq. (4) subject to the requirement that \( x' \) lies in the ball defined by \( \sum_{l \in L} ||C^{(l)}(x') - C^{(l)}(x)||_2 \leq \rho' \) for some \( \rho' \).

To convert the adversarial perturbation to a saliency map, we first calculate the size of the adversarial perturbation in each pixel by computing the average squared difference over the channels. Second, in order to highlight areas with large changes, we apply a Gaussian blur with parameter \( \sigma \) to the differences to give our resultant saliency map.

We systematically evaluate the effect of altering the regularized layers for a range of tasks. We find that the method is relatively stable and Eq. (5) performs better than the unregularized Eq. (4) for weak localization, insertion and deletion and the pointing game. As shown in Fig. 2, as more layers are regularized the perturbation becomes more localized.

4. Perceptual Perturbations as Explanations

Before describing our experimental overview, we give a qualitative analysis of the perceptual perturbations (Fig. 4). The perturbations do a good job of localizing on a single object class, even in the presence of highly textured images (dragonfly on fern), and in images with multiple classes (baseball and people). Some error in localization seems to arise from supporting classes adjacent to the object - e.g. human legs behind the lawnmower are found to be salient.

Furthermore, a qualitative evaluation can be seen in Fig. 3. These images were selected to be challenging – we visualize a subset of those images where unregularized adversarial perturbations did not align with the object. Compared to other visual explanation techniques, our method highlights the interior textures of the target object in the image. This differs from gradient-based methods which capture finer edge details such as SmoothGrad [42] and to activation-based methods which highlight the entire object coarsely such as Grad-CAM [38]. This is perhaps clearest in the first image where we capture the interior texture of the socks rather than just its hard contours.

4.1. Sensitivity Studies

For all experiments we performed extensive sensitivity studies to evaluate the importance of regularizing over different layers (Fig. 5). Certain trends can be detected. Regularizing over most of the layers is effective for weak localization, insertion and deletion, and pointing games. However, the pointing game performs best with different layers, possibly as you only need to find a single point of an object, we find that regularizing only the top layer is optimal in our ablation study, even when testing at multiple resolutions.
4.2. Weak Localization

We evaluate perceptual perturbations as explanations using the weak localization protocol [12], and test our approach on the first 2000 ImageNet [37] validation images.

We construct a set of bounding boxes for the largest connected region using three simple strategies based on: thresholding the raw values, thresholding a fixed percent of the image, and thresholding scaled by the image mean following [12]. To match the previous thresholding strategies we normalize individual saliency maps to be in the range of \([0, 1]\) before applying the blur. For the first strategy, we use a value threshold where we grid search over the set of thresholds \(\alpha\) where \(0 < \alpha \leq 1\) at intervals of size 0.05. For the second strategy, we use a percentage threshold where we consider the \(\alpha\)% most salient pixels grid search over the same interval. For the third strategy, we use a threshold scaled by the per image mean where we grid search over

the set of thresholds \(\alpha\) where \(0 < \alpha \leq 10.50\) at intervals of size 0.05. We report the scores on all three strategies as well as the optimal strategy for each explanation method. For each threshold, we extract the largest connected component and draw a bounding box around it. The object is considered to be successfully localized when the Intersection over Union measure between this box and the ground truth is at least 0.5. Following Grad-CAM’s guided version [38], which makes use of image gradients, we consider a guided variant of our own method consisting of an element-wise multiplication between our perturbations and the normalized gradient of the \(C_i(x)\) with respect to the image \(x\).

We set \(T = -2\), \(\lambda' = 10000\), \(\lambda = 1\) in Eq. (5), and we run an ablation study for the first 1000 images. We select two sequential sets of ReLU layers to regularize over in a VGG19bn network [41] using the value-threshold strategy. For the un-guided variant of our method, we regularize from
Table 1. Results for Weak Object Localization (lower is better, see sec. 4). We have the lowest error for each thresholding strategy.

<table>
<thead>
<tr>
<th>Method</th>
<th>Value</th>
<th>Per. Mean</th>
<th>Best</th>
</tr>
</thead>
<tbody>
<tr>
<td>GuidedBP [45]</td>
<td>0.48</td>
<td>0.52</td>
<td>0.49</td>
</tr>
<tr>
<td>Grad-CAM [38]</td>
<td>0.49</td>
<td>0.51</td>
<td>0.48</td>
</tr>
<tr>
<td>Guided-Grad-CAM [38]</td>
<td>0.46</td>
<td>0.49</td>
<td>0.45</td>
</tr>
<tr>
<td>Excitation [56]</td>
<td>0.48</td>
<td>0.50</td>
<td>0.44</td>
</tr>
<tr>
<td>SmoothG [42]</td>
<td>0.47</td>
<td>0.49</td>
<td>0.47</td>
</tr>
<tr>
<td>IntegratedG [47]</td>
<td>0.44</td>
<td>0.51</td>
<td>0.48</td>
</tr>
<tr>
<td>Extremal [10]</td>
<td>0.55</td>
<td>0.52</td>
<td>0.54</td>
</tr>
<tr>
<td>RISE [33]</td>
<td>0.51</td>
<td>0.52</td>
<td>0.48</td>
</tr>
<tr>
<td>NormGrad [34]</td>
<td>0.49</td>
<td>0.52</td>
<td>0.46</td>
</tr>
<tr>
<td>sNormGrad [34]</td>
<td>0.49</td>
<td>0.52</td>
<td>0.47</td>
</tr>
<tr>
<td>Us NoPer</td>
<td>0.50</td>
<td>0.47</td>
<td>0.46</td>
</tr>
<tr>
<td>Us Unguided</td>
<td>0.44</td>
<td>0.44</td>
<td>0.43</td>
</tr>
<tr>
<td>Us Guided</td>
<td>0.41</td>
<td>0.43</td>
<td>0.41</td>
</tr>
</tbody>
</table>

Table 2. Results for deletion (first column) and insertion game (second column). The result shows our method performs better than other methods for the deletion metric (without blur) and is comparable to other methods for the insertion metric (with blur).

<table>
<thead>
<tr>
<th>Method</th>
<th>Deletion Score</th>
<th>Insertion Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gradient [30]</td>
<td>0.19</td>
<td>0.51</td>
</tr>
<tr>
<td>Deconv [55]</td>
<td>0.21</td>
<td>0.56</td>
</tr>
<tr>
<td>GuidedBP [45]</td>
<td>0.14</td>
<td>0.57</td>
</tr>
<tr>
<td>Excitation [56]</td>
<td>0.12</td>
<td>0.63</td>
</tr>
<tr>
<td>Grad-CAM [38]</td>
<td>0.11</td>
<td>0.64</td>
</tr>
<tr>
<td>Extremal [10]</td>
<td>0.16</td>
<td>0.62</td>
</tr>
<tr>
<td>RISE [33]</td>
<td>0.12</td>
<td>0.65</td>
</tr>
<tr>
<td>NormGrad [34]</td>
<td>0.09</td>
<td>0.58</td>
</tr>
<tr>
<td>sNormGrad [34]</td>
<td>0.10</td>
<td>0.59</td>
</tr>
<tr>
<td>blurDiff (σ = 2.5)</td>
<td>0.14</td>
<td>0.59</td>
</tr>
<tr>
<td>Us NoPer (σ = 0.0)</td>
<td>0.10</td>
<td>0.42</td>
</tr>
<tr>
<td>Us NoPer (σ = 2.5)</td>
<td>0.15</td>
<td>0.54</td>
</tr>
<tr>
<td>Us (σ = 0.0)</td>
<td>0.07</td>
<td>0.54</td>
</tr>
<tr>
<td>Us (σ = 1.0)</td>
<td>0.09</td>
<td>0.61</td>
</tr>
<tr>
<td>Us (σ = 2.5)</td>
<td>0.11</td>
<td>0.62</td>
</tr>
<tr>
<td>Us (σ = 5.0)</td>
<td>0.12</td>
<td>0.63</td>
</tr>
</tbody>
</table>

4.3. Insertion/Deletion Game

We compute the insertion and deletion metrics from [33]. For the deletion metric, we construct the deletion response curve by sequentially changing the most salient pixels from their original value to mid-gray and measuring the classifier response. The deletion metric is defined as the AUC of the deletion curve, a smaller AUC score (i.e., a sharper drop in classifier response) is considered indicative of a better explanation. The insertion metric is similar, however, rather than removing the pixels of largest saliency, it inserts the original values into a blurred version of the original image.
### 4.4. Pointing Game

Additionally, we evaluated on the pointing game introduced by [56] on the VOC dataset. In this game the task is to select a single point in the image which is in the object in question [56]. The game is subdivided into 2 subtasks, a standard set of images, and a more difficult subset in which the object occupies less than 25% of the image, and must also contain a distracting class (see [56]). We use the TorchRay [9] implementation of the game and comparison methods and [11] for the implementation of NormGrad.

The pointing game uses a modified VGG16 classifier that assigns multiple classes to a single region of the image, and thus we use Eq. (5) with $T = -10$, $\lambda' = 1000$, $\lambda = 1$ and the formulation from Eq. (2). In this game, the images vary in size, and the classifier returns a detector like response for a set of overlapping regions of the image. Given a choice of class, we suppress all candidate regions by ensuring that the maximal response of any region is close to the target value.

As with the previous games, we select the layers and blur using an ablation study. We test each range of contiguous layers from the first ReLU layer, to final ReLU layer, and we vary the blur between 0 and 100%. We perform the study on the first 500 bounding boxes, which corresponds to 457 standard bounding boxes of which 160 are in the difficult subset and 43 boxes excluded by the benchmark.

Results can be seen in Fig. 5 lower left. Due to the reduced dataset (457), we report the success rate over the standard set, rather than the average class success reported on the full dataset. For compactness, we display the best result over the $\sigma$s, and to measure consistency we present the percentage of $\sigma$s which yield a performance above 82% (SM B.2). Two layer sets achieve the largest score, ReLU 9 (layer 22) with $\sigma \in \{47, 48\}$ or ReLU 12, with $\sigma \in \{41, 42\}$. We select ReLU 9 with $\sigma = 48$. The first layer is less sensitive to the choice of blur with $\approx 70\%$ giving a result above 82% in contrast to the $\approx 44.6\%$ for the final layer (this setting also gives a lower performance overall with a score of 83.8% (66.4%).

The full results, conducted on all images/tests are presented in the left column of Table 3. We are competitive with the best methods, with a 2.9% performance difference with the best method in the standard setting. Further, in SM B we present a study with a limited set of $\sigma$s where we achieve qualitatively similar results, but only a 0.3% drop in standard performance and no drop in the difficult setting. We additionally compare to our framework with the same parameters but without the perceptual loss (with the blur which maximizes the ablation score - see SM B.1). We refer to this as the ‘no perceptual’. As we see in Table 3 our version with the perceptual loss outperforms it. We also display this approach in the ablation study (top left corner). The perceptual loss outperforms in almost all cases, with exceptions in the low layers. Further, the results with the perceptual are more consistently above 82% (SM B).

#### Table 3. Results for the Pointing Game

<table>
<thead>
<tr>
<th>Method</th>
<th>Orig. Image</th>
<th>Scaled Image</th>
</tr>
</thead>
<tbody>
<tr>
<td>Center</td>
<td>69.6 (42.4)</td>
<td>69.6 (42.4)</td>
</tr>
<tr>
<td>Gradient [49]</td>
<td>76.3 (56.9)</td>
<td>84.6 (70.0)</td>
</tr>
<tr>
<td>Deconv [55]</td>
<td>67.5 (44.2)</td>
<td>75.0 (53.5)</td>
</tr>
<tr>
<td>GuidedBP [45]</td>
<td>75.9 (53.0)</td>
<td>83.7 (67.1)</td>
</tr>
<tr>
<td>Excitation [56]</td>
<td>77.1 (56.6)</td>
<td>84.0 (67.5)</td>
</tr>
<tr>
<td>Grad-CAM [38]</td>
<td>86.6 (74.0)</td>
<td>89.1 (77.7)</td>
</tr>
<tr>
<td>Extremal [10]</td>
<td><strong>88.0 (77.3)</strong></td>
<td>86.4 (71.0)</td>
</tr>
<tr>
<td>RISE [33]</td>
<td>86.7 (75.4)</td>
<td>NA (NA)</td>
</tr>
<tr>
<td>NormGrad [34]</td>
<td>81.9 (64.9)</td>
<td>88.6 (75.6)</td>
</tr>
<tr>
<td>sNormGrad [34]</td>
<td>86.0 (72.7)</td>
<td><strong>90.1 (80.8)</strong></td>
</tr>
<tr>
<td>Us NoPer</td>
<td>81.2 (62.9)</td>
<td>86.0 (71.6)</td>
</tr>
<tr>
<td>Us</td>
<td>85.1 (69.0)</td>
<td>88.2 (76.5)</td>
</tr>
</tbody>
</table>

For efficiency, we create one perturbation per set of layers and vary $\sigma$
Finally, we also propose an additional pointing game experiment. As this game is defined on non-standard sized image, we consider the case where we increase the size of the images via a simple resizing, and test the performance on this set. This helps performance as the fully convolutional network then gives predictions for more areas that are then maximized over. We implement this as a pre-and post-processing step, where we resize the image, construct the saliency map, and resize the saliency map to the correct size. We perform a similar ablation study for this case, (Fig. 5 bottom right panel). The optimal layer in this case is ReLU 12 with $\sigma \in \{28.0, 30.0\}$, (selected 30).

The results can be seen in the right column of Table 3, where most methods work on this new dataset (the TorchRay RISE implementation requires a perfectly sized image and is NA). The performance of many of the methods is increased in this new approach, with many weaker methods substantially increasing (except for center - a baseline which chooses the center point). Further, the performance characteristics of the best performing methods also improves with resizing, albeit by lower margins, indicating that this resizing generally helps almost all methods. The best performing method is Selective NormGrad with a score of 90.1%, 2.1% above the previously highest score. Our Perceptual method is competitive with the best performing methods, achieving a score of 88.2%.

4.5. Sanity Checks

Finally, we apply the Sanity Checks proposed in [1] to our method. We randomize the weights on the final $k$ layers of the network, and observe how the saliency of our approach varies visually. We perform this experiment on VGG19bn, as most of the experiments in this paper were performed on this network, and we match the remaining parameters/layers to the insertion deletion study. When randomizing the layers we set the parameters to the value that would have been in an untrained network. We select the layers in the VGG19bn architecture that correspond to the VGG16 layers used by [34], namely each fully connected layer and the final convolution in each set with the exception of the first set for which we use the first convolution.

Fig. 7 shows the results on three common sanity check images. As the layers are progressively randomized, saliency spreads from the relevant objects towards other objects and highly textured regions in the image, and is mostly non-existent after the second convolution qualitatively matching the behavior in Selective NormGrad seen in [34].

5. Conclusion

We explored a novel regularization for adversarial perturbations based on the perceptual loss. This regularization is designed to block the exploitation of exploding gradients when generating adversarial perturbations forcing larger and more meaningful perturbations to be generated. The fact that they remain imperceptible to humans is another piece of the puzzle in understanding the interrelationship between adversarial perturbations, neural networks, and human vision. We believe that the imperceptible nature of our adversarial perturbations is due to both explanations discussed in the introduction for the existence of adversarial perturbations being partially correct. Even when regularizing over the layers of our network and preventing adversarial perturbations from exploiting exploding gradients and some of the inherent instability in deep networks, Szegedy et al.’s [48] argument still holds and one can obtain a new class label by slightly altering a large number of pixels.

We have shown how these perturbations can be interpreted as explanations and obtained state-of-the-art results on several standard explainability benchmarks.

Acknowledgments This work was supported by the Omidya Group and The Alan Turing Institute under the UK Engineering and Physical Sciences Research Council (EPSRC) grant no. EP/N510129/1 and Accenture Plc. Moreover, we acknowledge Pearl for computing resources and in particular the help of Tomas Lazauskas and Suleman Tariq.
References


