Abstract

Adversarial perturbation of images, in which a source image is deliberately modified with the intent of causing a classifier to misclassify the image, provides important insight into the robustness of image classifiers. In this work we develop two new methods for constructing adversarial perturbations, both of which are motivated by minimizing human ability to detect changes between the perturbed and source image. The first of these, the Edge-Aware method, reduces the magnitude of perturbations permitted in smooth regions of an image where changes are more easily detected. Our second method, the Color-Aware method, performs the perturbation in a color space which accurately captures human ability to distinguish differences in colors, thus reducing the perceived change. The Color-Aware and Edge-Aware methods can also be implemented simultaneously, resulting in image perturbations which account for both human color perception and sensitivity to changes in homogeneous regions. Though Edge-Aware and Color-Aware modifications exist for many image perturbations techniques, we focus on easily computed perturbations. We empirically demonstrate that the Color-Aware and Edge-Aware perturbations we consider effectively cause misclassification, are less distinguishable to human perception, and are as easy to compute as the most efficient image perturbation techniques. Code and demo available at https://github.com/rbassett3/Color-and-Edge-Aware-Perturbations.

1. Introduction

Adversarial perturbations have shown that state-of-the-art techniques for image classification are inherently unstable, because minute changes to an image can result in dramatic changes in the predicted class of the image. Many techniques have been introduced to generate adversarial perturbations, but a common theme is a formulation which encourages substantial change to the output of the classifier while restricting to only small changes of the image. In these formulations, metrics for quantifying change to the
image are often mathematically instead of perceptually motivated.

We address this problem by proposing two new techniques for generating adversarial perturbations. The first, our Edge-Aware method, is motivated by human ability to detect minor modifications against a smooth background. It uses a texture filter, such as a Sobel or Gabor filter, to limit perturbations in smooth regions. The result is that the Edge-Aware method constructs perturbations which preserve smoothly textured regions in the image. Though this limitation might be expected to make misclassification more difficult to achieve, we find that misclassification can still be caused relatively easily, even when generating perturbations which target a certain class for the perturbed image.

Our second contribution is the Color-Aware method for generating image perturbations. While the Edge-Aware method reduces detection by considering how a pixel differs from its neighbors, the Color-Aware method focuses on the pixel value itself. It is well-known that for RGB representations of a pixel, metrics like \( \ell_2 \) or \( \ell_\infty \) do not accurately capture human ability to perceive color difference. Therefore perturbations which are small with respect to these metrics may still be easily detected by an individual comparing the perturbed and source images. To overcome this issue we convert the image to a color space which does capture human perception in color difference, and construct the perturbation in this space. One concern with this approach is computational, because conversion from RGB to color spaces which attempt to accurately model human perception can involve complicated transformations. We mitigate this concern by using the CIE L*a*b* (CIELAB) color space, in which the \( \ell_2 \) distance between pixels captures perceived color distance. The tractability of this constraint, and the fact that we construct the perturbation directly in CIELAB color space, allows us to construct Color-Aware perturbations with minimal computational overhead.

The Color-Aware and Edge-Aware methods can be applied simultaneously to generate a Color-and-Edge-Aware perturbation, an example of which appears in Figure 1. Color-and-Edge-Aware perturbations reduce human ability to distinguish between the source and modified images by constraining both texture and color discrepancies. In the remainder of this paper, we demonstrate the effectiveness of Color and Edge-Aware perturbations. We show that they are among the most computationally efficient methods for generating adversarial perturbations, while still effectively causing misclassification of the perturbed image. We also provide empirical evidence which confirms that Color and Edge-Aware perturbations are more difficult for a human observer to detect.

Before proceeding we establish some notation. We assume that images have been scaled to take values in \([0, 1]\). Let \( w \), \( h \), and \( c \) be fixed positive integers which give the width, height, and number of color channels, respectively, of the images considered. Denote \([0, 1]^{h \times w \times c}\), the set of valid images, by \( \mathcal{X} \). Let \( \mathcal{C} := \{1, \ldots, C\} \) be a set of possible image classes. An image classification algorithm is a function \( F : \mathcal{X} \rightarrow \Delta \mathcal{C} \), where \( \Delta \mathcal{C} \) denotes the \( C \)-dimensional probability simplex. It is common for an image classifier to convert logits to probabilities using the softmax function, in which case \( F \) can written \( F = \text{softmax} \circ Z \), for some function \( Z : \mathcal{X} \rightarrow \mathbb{R}^C \). Throughout, we will denote a source image by \( x \), a perturbation by \( \delta \), and a perturbed image \( x + \delta \) by \( x' \). We denote by \( \| \cdot \|_{p,c} : [0, 1]^{h \times w \times c} \rightarrow \mathbb{R}^{h \times w} \) an \( \ell_p \) norm applied across the color dimension of an image. Otherwise, \( \| \cdot \|_p \rightarrow \mathbb{R} \) denotes the entrywise \( p \)-norm of a tensor. Lastly, we use \( \langle \cdot, \cdot \rangle \) to denote the Frobenius inner product.

2. Related Work

Neural networks are state-of-the-art tools for image classification, and have been used in a variety of application areas including computer vision [6], [17], natural language processing [4], and Markov decision processes [13]. Despite their success, Szegedy et al. [15] first noticed that neural networks are vulnerable to adversarial perturbations, in which noise is added to an input in order to cause misclassification without substantially changing the input. Adversarial perturbations are especially interesting in the context of image classification, in part because of the high-level of performance that artificial neural networks enjoy in that domain. In their paper, Szegedy et al. solved for a perturbed image \( x' \) of an input image \( x \) by solving the following optimization problem using bound-constrained L-BFGS.

\[
\min_{x' \in \mathbb{R}^{h \times w \times c}} \ L(F(x'), l) + \frac{\alpha}{2} \| x - x' \|_2^2
\]

subject to \( 0 \leq x' \leq 1 \).

The function \( L : \Delta^{C} \times \mathcal{C} \rightarrow \mathbb{R} \), and label \( l \in \mathcal{C} \) take two forms. In the untargeted setting, where any misclassification is acceptable, \( L \) is taken to be the negative cross entropy loss and \( l \) the true label of the image. In the targeted setting, \( L \) is taken to be the cross entropy loss and \( l \) the targeted label for the image. Large values of the parameter \( \alpha \) encourage the perturbation to be close to the source image; this value must be chosen separately.

A few features of the Szegedy et al. method are worth emphasizing. First, the optimization method is quasi-newton, and hence requires only first-order information about the objective function. This is important because modern software for neural networks emphasizes efficient gradient computation. Second order information, on the other hand, would be extremely burdensome to compute and cannot be assumed. Another important feature of the Szegedy et al. method is that projection onto the constraint set is easy, so that it can be computed quickly as part of
iterative first-order methods. Lastly, we note that, despite its simplicity, the L-BFGS method requires extra memory to store Hessian approximations, and can also require many function evaluations to compute the step length (often via backtracking).

In contrast to the L-BFGS method of Szegedy et al., the Fast Gradient Sign method (FGSM), prioritizes perturbations which can be easily computed [5]. The FGSM method proposes the following optimization problem

$$\min_{\delta \in \mathbb{R}^{h \times w \times c}} L(F(x + \delta), l)$$
subject to $\|\delta\|_\infty < \alpha$.

To reduce the computational burden associated with solving this problem, the authors instead optimize a linear approximation of the objective by taking the gradient of the input with respect to $\delta$.

$$\min_{\delta \in \mathbb{R}^{h \times w \times c}} \langle \nabla_x (L(F(x, l)), \delta) \rangle$$
subject to $\|\delta\|_\infty < \alpha$.

This has the closed-form solution

$$x' = x + \alpha \text{sign} (\nabla_x (L(F(x, l))))$$.

In the event that $x' \not\in [0, 1]^{h \times w \times c}$, it can easily be projected onto this set, making the result a valid image. FGSM can also be used as an iterative method, where the perturbed image $x'$ from one iteration is used as the input image $x$ in the next iteration.

The FGSM's simplicity is the key to its success. Though the linear approximation of FGSM is simpler than the quasineutron method of Szegedy et al, FGSM requires only the elementwise sign of the gradient yet has been shown to still generate effective perturbations.

There have been many other methods proposed to generate adversarial perturbations. Two of the most celebrated are the Carlini-Wagner $\ell_2$ perturbation [2] and DeepFool [7]. The Carlini-Wagner approach is designed to achieve very precise misclassification, and it includes a tuning parameter which specifies the misclassification confidence. In this sense, the Carlini-Wagner approach is well-suited to answer the question: “What is the minimal perturbation required to move this image onto the decision boundary between classes?” Though it effectively generates adversarial perturbations, the Carlini-Wagner method is complex relative to other methods for adversarially perturbing images, requiring multiple starting points and at least three additional univariate parameters depending on the descent method used. Because of the complex formulation and relatively high computational overhead, Carlini-Wagner perturbations have different motivations than our Color-Aware and Edge-Aware perturbations, in which we seek to efficiently and effectively create image perturbations which are undetectable by human observers. Doing so will accomplish two goals. The first is practical; perturbations which are easier to be compute can be more readily applied. Our second goal is theoretical, in that we seek the simplest technique that accomplishes the task of constructing effective and imperceivable image perturbations.

The motivation for DeepFool better aligns with the goals of this paper because of its emphasis on lightweight construction of perturbations. The DeepFool algorithm proceeds iteratively by stepping towards the decision boundary of the classifier. In order to make these iterates tractable, DeepFool linearly approximates the decision boundary at each iteration. DeepFool prioritizes efficient computation, and in this way is a compromise between Carlini-Wagner and FGSM. One drawback of DeepFool is that it only accommodates untargeted perturbations. Like DeepFool, we are motivated by efficient computation of perturbations, but our method will accommodate targeted perturbations. We also note that DeepFool, like many other perturbation methods, can be easily modified to include both our Edge-Aware and Color-Aware ideas by changing the norm it uses internally.

3. Proposed Approaches

We begin by describing our Color-Aware perturbation. The primary motivation for developing our Color-Aware perturbation is that the distance between colors, when represented as vectors in RGB space, does not correspond to the perceived difference from the perspective of a human observer. There have been many efforts to devise color systems which respect human perception, beginning with the Munsell Color System [9] in 1905. Since then, the International Commission on Illumination (with acronym CIE in French) developed a sequence of color spaces and color distances which attempt to quantify perceived color difference. The first of these was in 1931 with the CIELAB color space. This space was improved in 1976 with the addition of the CIELAB and CIELUV color spaces, which better models perceived color difference. The CIELAB and CIELUV spaces perform similarly with respect to their accuracy in perceived color difference [8], and we opt to use CIELAB. We note that in the CIELAB representation of colors, perceived distance between colors is measured using the Euclidean distance between them.

Conversion from RGB to CIELAB requires an intermediate conversion to CIEXYZ, which is a linear transformation.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = A \begin{bmatrix} R' \\ G \\ B \end{bmatrix}$$ (1)
The conversion functions where δx image when necessary, we will clarify the space in which a source with appropriate notational extensions to other color spaces. The (invertible) matrix distinguished from the background. The CIELAB distance, on the other hand, accurately captures perceived color difference. Left: CIELAB distance 3.04. Center: CIELAB distance 17.23. Right: CIELAB distance 76.94.

The (invertible) matrix $A \in \mathbb{R}^{3 \times 3}$ is specified in the CIEXYZ standard [12]. Conversion from XYZ to LAB space is nonlinear.

$$L^* = 116 f \left( \frac{Y}{Y_n} \right) - 16$$

$$a^* = 500 \left( f \left( \frac{X}{X_n} \right) - f \left( \frac{Y}{Y_n} \right) \right)$$

$$b^* = 200 \left( f \left( \frac{Y}{Y_n} \right) - f \left( \frac{Z}{Z_n} \right) \right)$$

where $\delta = \frac{6}{29}$ and

$$f(t) = \begin{cases} \sqrt{t} & \text{if } t > \delta^3 \\ \frac{t}{\delta^3} + \frac{1}{29} & \text{otherwise.} \end{cases}$$

The constants $X_n$, $Y_n$ and $Z_n$ depend on the illumination standard used, but are commonly taken as $X_n := 95.0489$, $Y_n := 100$, and $Z_n := 108.8840$ [11]. Denote by $C_{\text{RGB} \rightarrow \text{XYZ}}$ a conversion function from RGB to CIEXYZ, with appropriate notational extensions to other color spaces. When necessary, we will clarify the space in which a source image $x$, perturbed image $x'$, or perturbation $\delta$ reside with an appropriate superscript. We have

$$C_{\text{RGB} \rightarrow \text{LAB}} = C_{\text{XYZ} \rightarrow \text{LAB}} \circ C_{\text{RGB} \rightarrow \text{XYZ}}.$$

The conversion functions $C_{\text{XYZ} \rightarrow \text{LAB}}$ and $C_{\text{RGB} \rightarrow \text{XYZ}}$ are invertible, so $C_{\text{LAB} \rightarrow \text{RGB}}$ is defined as the appropriate composition of inverse conversions. We note that all conversion functions are continuous and piecewise differentiable.

CIELAB was further extended to color difference formulas CIEDE94 and CIEDE2000 in the corresponding years. One of the few other works to apply perceptual color spaces to adversarial perturbations is [16], which used a Carlini-Wagner approach to compute adversarial perturbations where the difference between the perturbed and source image was quantified using CIEDE2000. That work shares our Color-Aware motivation of constructing perturbations which account for human color perception, but our emphasis on efficient computation prompts us to use CIELAB instead, thus avoiding the complexity of a Carlini-Wagner formulation. Though the CIEDE94 and CIEDE2000 difference formulas were intended to correct some imprecisions in using CIELAB to measure perceived color difference, they require complicated nonconvex manipulations of the LAB coordinates and are not given by the $\ell_p$ distance in some color space. This increases the computational burden required to compute adversarial perturbations and motivates our use of CIELAB. To our knowledge there has only been one other effort using perceptual color spaces for image perturbations [1]. Like our work, the authors use the CIELAB distance, but their formulation differs critically from ours in its formulation, positing an intractable constraint which is mitigated by solving a penalized version instead. Our formulation only uses tractable constraints, mirroring the simplicity of FGSM in both its constraint set and the closed-form solution of its linear approximation.

We propose to adversarially perturb an image $x^{\text{RGB}}$ as follows. Convert $x^{\text{LAB}} := C_{\text{RGB} \rightarrow \text{LAB}}(x^{\text{RGB}})$ and solve the following.

$$\min_{\delta} L (F \circ C_{\text{LAB} \rightarrow \text{RGB}} (x^{\text{LAB}} + \delta), l)$$

subject to $\|\delta\|_{2,c} \leq \alpha$.

As in FGSM, we linearly approximate the objective function to yield the closed-form solution below.

$$x' = x^{\text{LAB}} + \alpha \frac{\nabla_{x} L (F \circ C_{\text{LAB} \rightarrow \text{RGB}} (x^{\text{LAB}} + \delta), l)}{\|\nabla_{x} L (F \circ C_{\text{LAB} \rightarrow \text{RGB}} (x^{\text{LAB}}), l)\|_{2,c}}$$

We note that division in (6) is pointwise and broadcast across the color dimension so that the dimensions are compatible. Finally, the perturbed image can be converted to RGB, $x'^{\text{RGB}} = C_{\text{LAB} \rightarrow \text{RGB}} (x'^{\text{LAB}})$.

We note that because $\delta$ is a perturbation in CIELAB space, the constraint in (7) represents perceived color difference. Also, the classifier $F$ is assumed to require RGB inputs, though we note that there is work attempting to train models directly on CIELAB representations of images [3].

Next we describe the Edge-Aware perturbation method. Let $W : \mathcal{X} \rightarrow [0, 1]^{h \times w}$ denote a pixel-wise edge detector, such as the Sobel or Gabor filter, where a value near one means an edge is detected. We construct Edge-Aware perturbations by weighting pixels in the perturbation constraint by their edge weights, thus reducing the magnitude of perturbation permitted in smooth regions. This can be applied to the FGSM directly, but we will introduce it in the context of our Color-Aware perturbation method. Letting
perturbation different objects (dog, grass, or dark background) have perturbations in different directions. Color-Aware method. Figure 3 gives a comparison of the perturbed image between a perturbed and source image, where the perturbation, we have rounded values less than $\frac{1}{4}$ and greater than $\frac{1}{6}$ to 0 and 1, respectively. Note that in the Color-Aware perturbation different objects (dog, grass, or dark background) have perturbations in different directions.

\[
w = W(x_{RGB}), \text{ where we propose solving the following}
\]

\[
\min_{\delta} L \left( F \circ C_{LAB \rightarrow RGB} \left( x_{LAB} + \delta \right), l \right)
\]

subject to $\|\delta\|_{2,c} \leq \alpha w$.

Again, by linearly approximating the objective function we arrive at the closed-form solution

\[
x'_{LAB} = x_{LAB} + \alpha w \frac{\nabla_x L \left( F \circ C_{LAB \rightarrow RGB} \left( x \right), l \right)}{\|\nabla_x L \left( F \circ C_{LAB \rightarrow RGB} \left( x \right), l \|_{2,c}}
\]

\[
x'_{RGB} = C_{LAB \rightarrow RGB} \left( x'_{LAB} \right).
\]

As in equation (6), the multiplication and division in (8) is pointwise and broadcast across the color dimension. We also note that in equations (6) and (8) it is possible that the 2-norm at a pixel is zero, in which case we do not make any perturbation at the pixel.

4. Experiments

In this section we empirically evaluate the performance of our Color and Edge-Aware perturbations. Throughout, we use the Inception v3 classifier [14] and the ILSVRC 2012 validation set [10] as the classifier to disrupt and the images to perturb. We will use a Sobel filter to construct the edge weights, though other edge filters would also be appropriate.

4.1. Evaluating Detectability of Perturbations

We begin by comparing perturbations computed using our Color-Aware method with those of FGSM, in order to show the qualitative difference between perturbation directions. For a small value of $\alpha$, we compute $x' - x$, the difference between a perturbed and source image, where the perturbed image $x'$ is constructed through both FGSM and our Color-Aware method. Figure 3 gives a comparison of the perturbations, where the Color-Aware version is rounded to the nearest vertex of the RGB cube to make it more visually comparable with the FGSM perturbation.

We see that our Color-Aware method clearly identifies color trends in this perturbation that FGSM does not. In the region containing the grass, red occurs in much lower quantities than the green and blue colors. Similar to our example in figure 2, perturbing the nondominant color planes results in large RGB change with small perceived color difference. The most prevalent colors in this region are the teal $(0, .5, .5)$ and coral $(1, .5, .5)$ which represent changes in the red color plane because of the $[0, 1]$ scaling. Less prevalent but still visible are lavender $(.5, .5, 1)$ and olive $(.5, .5, 0)$, which represent perturbations in the blue color plane, the other non-dominant color in the region. For the region containing the white samoyed dog, the white of the dog has a fairly even distribution of RGB colors, so there is more variety in the perturbations than in the grassy region. In FGSM, however, color trends are impossible to distinguish. The region containing the dog can be distinguished upon close inspection based on the texture of the perturbation, but not the distribution of its colors.

We also include a comparison of our Color-Aware and Color-and-Edge-Aware methods with FGSM and L-BFGS. In figure 4 we construct untargeted perturbations on an image containing a submarine. Close inspection reveals perturbation artefacts in the sky near the water’s surface and around the submarine’s broadcasting equipment for all methods except Color-and-Edge-Aware, with Color-Aware displaying less artefact than L-BFGS and FGSM. Table 1 contains classification confidence and quantifications of the perturbation’s size using various norms. The large $\ell_2$ norm of the Color-and-Edge-Aware perturbation relative to the other methods, combined with its small $\ell_1$ norm, suggests that there are fewer perturbed pixels but that the magnitude of these perturbations is larger. This aligns with the intu-
Figure 4: Comparison of perturbation methods in an untargeted attack. For all methods the perturbed image was classified as a breakwater. Details in text and table 1.

Figure 5 contains a similar example for a targeted attack, where the source image contains a tank and the targeted label is a mobile home. All perturbation methods successfully induced misclassification. The performance of the methods is generally similar to the untargeted setting, in that all methods except Color-and-Edge-Aware have easily detectable perturbations. Similar to the submarine example, L-BFGS appears to place large magnitude perturbations in certain regions, in this example at the front of the tank, but less perturbation overall. Both Color-Aware and FGSM make notable texture changes on the side of the tank and in the sky above. Table 3 summarizes the performance of the methods. Though the Color-and-Edge-Aware perturbation is larger in $\ell_2$ norm than the others, the location of these perturbations and its more accurate modeling of color perception makes it less discernible than the others from the perspective of a human observer.

### 4.2. Inducing Misclassification

In this section we demonstrate that Color-Aware and Color-and-Edge-Aware perturbations are effective at inducing misclassification. For our first experiment, we choose 100 images from the 2012 ILSVRC validation set which the Inception v3 network classifies correctly. The $\alpha$ value is chosen uniquely from a set of candidates for each perturbation but fixed for all images. Our selected $\alpha$ corresponds to the value that misclassified the highest proportion of the first 10 images. We perform both targeted and untargeted experiments, where for all images the target class was a coffee mug. The step length/penalty parameter was chosen in the untargeted experiment and was the same for the targeted experiments. Table 2 summarizes our results.

| Image          | Prob. | $||\delta||_1$ | $||\delta||_2$ |
|----------------|-------|---------------|---------------|
| Original       | .997  | 0             | 0             |
| L-BFGS         | .442  | 538.28        | 1.89          |
| FGSM           | 1.00  | 1558.36       | 3.82          |
| Color-Aware    | .980  | 1594.55       | 3.83          |
| Color-&-Edge-Aware | .980 | 987.29       | 3.77          |

Table 1: Numerical comparison of the untargeted attacks producing the perturbed images in figure 1. For the original image the probability listed is for the submarine class. For all others it is the breakwater class, which was the class with the highest probability for all methods.

| L-BFGS | FGSM     | C-Aware | C & E-Aware |
|--------|----------|---------|-------------|
| Untargeted | 92%     | 100%    | 100%        | 91%          |
| Targeted   | 100%    | 100%    | 100%        | 91%          |

Table 2: Misclassification percentages for 100 images with 5 iterations for untargeted and 10 iterations for targeted. Full details in text.

The results indicate that perturbations which are Color-Aware and/or Edge-Aware reliably induce misclassification. Moreover, all methods considered consistently cause misclassification given proper choice of $\alpha$ and sufficient number of iterations. Unsurprisingly, we see that achieving tar-
| Image                          | Prob. | \|\delta\|_1 | \|\delta\|_2 |
|-------------------------------|-------|-------------|-------------|
| Original                      | .852  | 0           | 0           |
| L-BFGS                        | .804  | 609.55      | 1.79        |
| FGSM                          | .995  | 1621.22     | 3.98        |
| Color-Aware                   | .980  | 1616.69     | 3.90        |
| Color and Edge-Aware          | .980  | 1185.87     | 4.45        |

Table 3: Numerical comparison of the targeted attacks producing the perturbed images in figure 3. For the original image the probability listed is for the tank class. For all others it is the mobile home class, which was the targeted label.

targeted misclassification is more difficult and requires more iterations to do so reliably. Our Color-and-Edge-Aware method achieves 91% misclassification rate on both the untargeted and targeted tasks whereas FGSM and Color-Aware achieves 100% misclassification, providing evidence that restricting perturbations to smoothly-textured regions has only a small impact on the ability of the method to induce misclassification. Only 3% of images were unable to be misclassified by both untargeted and targeted Color-and-Edge-Aware, which suggests that ability to induce misclassification using perturbations which are restricted to certain regions depends on the region and the target class.

For our second experiment, we compare how the misclassification confidence changes as a function of the number of iterations and magnitude of the perturbation as quantified by various norms. We use the source image of a submarine in figure 4 to perform an untargeted attack. Our results are given in figure 6, where the curves in each subfigure are parametrized by iteration number. These plots show that our contributions are competitive with L-BFGS and FGSM when measured by the \(\ell_p\) norms in an RGB representation, and that they outperform the competition when quantified using the CIELAB color distance.

From the perspective of per-iteration confidence, figure 6 provides evidence that FGSM and Color-Aware perturbations behave similarly with respect to misclassification confidence as a function of iterations. Color-and-Edge-Aware performs poorly in metrics which are sensitive to outliers (\(\ell_2\) and \(\ell_\infty\)) because of its tendency to place larger perturbations in regions they are not easily detected. L-BFGS also performs poorly in these metrics, suggesting that it places large perturbations in isolated regions. That fact that L-BFGS does not place these perturbations in regions where it is difficult to discern suggests that the perturbation artefacts may be easily detectable. When quantified using the \(\ell_1\) norm of the CIELAB distances, Color and Edge-Aware perturbations outperform FGSM and are competitive with the more complex L-BFGS method.

4.3. Computational Efficiency

Next we compare the computational burden of the methods considered, as assessed by their running times. Color-Aware and Color-and-Edge-Aware perturbations have the advantage of properly accounting for human perception of color and texture, but we hope to do so as simply as possible. The gold standard for generating simple image perturbations is the fast gradient sign method, because it only requires the elementwise sign of the gradient. Compared to FGSM, Color-Aware and Color-and-Edge-Aware impose additional structure to model human perception. First, the
computation of the gradient requires composition with the conversion function $F_{LAB\rightarrow RGB}$. Second, instead of the sign of the gradient Color-Aware methods normalize the gradient in the $\ell_2$ norm across the color dimension. Lastly, making a perturbation method Edge-Aware requires an additional application of an edge filter. Our experiments show that Color-Aware and Color-and-Edge-Aware perturbations are only marginally less efficient to construct than FGSM perturbations.

Because our emphasis is on generating effective image perturbations with minimal computational resources, we consider both CPU and GPU implementations. Table 4 gives the running times for each method applied to 100 ILSVRC images. Our experiments were carried out in a Linux operating system, with an 8-core Intel i9-9980 CPU at 2.4 GHz, 64 GB of memory, and a GeForce GTX 1650 GPU.

| Method             | CPU time (s) | GPU time (s) |
|--------------------|--------------|--------------|
| L-BFGS             | 3.56 ± 0.55  | 0.52 ± 0.04  |
| FGSM               | 1.71 ± 0.05  | 0.29 ± 0.04  |
| Color-Aware        | 1.81 ± 0.12  | 0.35 ± 0.04  |
| Color-&-Edge-Aware | 1.83 ± 0.03  | 0.36 ± 0.06  |

Table 4: Mean and standard deviation of the run times for each method applied to 100 images. Each method was run for 10 iterations.

Table 4 demonstrates that the time required to generate Color-Aware and Color-and-Edge-Aware perturbations is similar to FGSM and less than L-BFGS. Interestingly, we note that the computational burden associated with composing the model with the conversion function $F_{LAB\rightarrow RGB}$, evident in the difference between FGSM and Color-Aware, results in a smaller relative increase in the CPU implementation (5.8%) than the GPU implementation (20.7%). In both implementations, the additional time required to make the method edge-aware is minor, between 0.01 and 0.02 seconds.

5. Conclusion

We have presented two new methods for creating adversarial image perturbations which are less discernible by a human observer. The first, our Color-Aware perturbation method, accounts for human perception of color by performing the perturbation directly in CIELAB space, where a simple constraint guarantees the perceived color change to the perturbed image is small. Our second contribution is our Edge-Aware method, which uses a texture filter to restrict perturbations to regions where a human observer is less likely to detect them. Color-Aware and Edge-Aware methodology can be combined to generate Color-and-Edge-Aware perturbations, which address both issues simultaneously. We find that our contributions reliably induce misclassification, require similar computation time as the most efficient techniques for generating adversarial perturbations, and are more difficult to detect than methods of similar complexity, providing evidence that Color and Edge-Aware perturbations are a simple yet effective way to generate perturbations which properly account for human perception.

Acknowledgments

Both authors acknowledge support from the Office of Naval Research’s Science of Autonomy Program, award number N0001420WX01523.
References

[1] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. Synthesizing robust adversarial examples. In International conference on machine learning, pages 284–293, 2018. 4

[2] Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In 2017 IEEE symposium on security and privacy (sp), pages 39–57. IEEE, 2017. 3

[3] Javier Díaz-Cely, Carlos Arce-Lopera, Juan Cardona Mena, and Lina Quintero. The effect of color channel representations on the transferability of convolutional neural networks. In Science and Information Conference, pages 27–38. Springer, 2019. 4

[4] Yoav Goldberg. Neural network methods for natural language processing. Synthesis Lectures on Human Language Technologies, 10(1):1–309, 2017. 2

[5] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572, 2014. 3

[6] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012. 2

[7] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2574–2582, 2016. 3

[8] Dorothy I Morley, Ruth Munn, and Fred W Billmeyer Jr. Small and moderate colour differences: II the morley data. Journal of the Society of Dyers and Colourists, 91(7):229–242, 1975. 3

[9] Dorothy Nickerson. History of the munsell color system and its scientific application. JOSA, 30(12):575–586, 1940. 3

[10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 115(3):211–252, 2015. 1, 5

[11] Janos Schanda. Cie colorimetry. Colorimetry: Understanding the CIE system, pages 25–78, 2007. 4

[12] Thomas Smith and John Guild. The cie colorimetric standards and their use. Transactions of the optical society, 33(3):73, 1931. 4

[13] Richard S Sutton and Andrew G Barto. Reinforcement learning: An introduction. MIT press, 2018. 2

[14] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2818–2826, 2016. 1, 5

[15] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199, 2013. 2

[16] Zhengyu Zhao, Zhuoran Liu, and Martha Larson. Towards large yet imperceptible adversarial image perturbations with perceptual color distance. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 1039–1048, 2020. 4

[17] Zhong-Qiu Zhao, Peng Zheng, Shou-tao Xu, and Xindong Wu. Object detection with deep learning: A review. IEEE transactions on neural networks and learning systems, 30(11):3212–3232, 2019. 2