IMPROVING PERCEPTUAL QUALITY OF SPATIALLY TRANSFORMED ADVERSARIAL EXAMPLES

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF INFORMATICS
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

AYBERK AYDIN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
MODELLING AND SIMULATION

AUGUST 2022

Approval of the thesis:

IMPROVING PERCEPTUAL QUALITY OF SPATIALLY TRANSFORMED ADVERSARIAL EXAMPLES

submitted by **AYBERK AYDIN** in partial fulfillment of the requirements for the degree of **Master of Science in Modelling and Simulation Department, Middle East Technical University** by,

Prof. Dr. Deniz Zeyrek Bozşahin
Dean, Graduate School of **Informatics** _____

Assoc. Prof. Dr. Elif Sürer
Head of Department, **Modelling and Simulation** _____

Prof. Dr. Alptekin Temizel
Supervisor, **Modelling and Simulation, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. Elif Sürer
Modelling and Simulation, METU _____

Prof. Dr. Alptekin Temizel
Modelling and Simulation, METU _____

Assoc. Prof. Dr. Gökhan Koray Gültekin
Electrical&Electronics Eng., Ankara Yildirim Beyazit University _____

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Ayberk Aydin

Signature :

ABSTRACT

IMPROVING PERCEPTUAL QUALITY OF SPATIALLY TRANSFORMED ADVERSARIAL EXAMPLES

Aydin, Ayberk
M.S., Department of Modelling and Simulation
Supervisor: Prof. Dr. Alptekin Temizel

August 2022, 42 pages

Deep neural networks are known to be vulnerable to additive adversarial perturbations. The amount of these additive perturbations are generally quantified using $\mathcal{L}_p$ metrics over the difference between adversarial and benign examples. However, even when the measured perturbations are small, they tend to be noticeable by human observers since $\mathcal{L}_p$ distance metrics are not representative of human perception. Spatially transformed examples work by distorting pixel locations instead of applying an additive perturbation or altering the pixel values directly, which produces adversarial examples with improved visual quality. However, the perturbation made by spatial transformations produce visible non-smooth distortions on luminance channels and needs a smoothness regularization over the applied flow field in order to improve the visual quality. On the other hand, humans are less sensitive to changes in chrominance component of visual media and such as resolution loss or pixel shifts in a constrained neighborhood. Motivated by these observations, we propose a novel variation of spatially transformed adversarial examples that creates adversarial examples by applying spatial transformations to chrominance channels of perceptual colorspaces such as YC_bC_r and $CIELAB$ to generate adversarial examples with high perceptual quality. Moreover, we find that the visual quality of these examples could be further improved by limiting the magnitude of applied spatial transformations. In a targeted white-box attack setting, the proposed method is able to obtain competitive fooling rates and experimental evaluations show that the proposed method has favorable results in terms of approximate perceptual distance between benign and adversarial images.

Keywords: deep learning, adversarial examples, perceptual quality

ÖZ

UZAMSAL DÖNÜŞÜMLÜ ÇEKİŞMELİ ÖRNEKLERİN ALGISAL KALİTESİNİN İYİLEŞTİRİLMESİ

Aydin, Ayberk
Yüksek Lisans, Bölümü
Tez Yöneticisi: Prof. Dr. Alptekin Temizel

Ağustos 2022 , 42 sayfa

Derin yapay sinir ağlarının eklemeli çekişmeli bozulmalara karşı savunmasız olduğu bilinmektedir. Bu bozulmaların miktarı $\mathcal{L}_p$ metrikleri ile ölçülmektedir. Ancak, ölçülen bozulmaların miktarı az olsa da bu bozulmalar insan gözlemciler tarafından görülebilmektedir çünkü $\mathcal{L}_p$ uzaklık metrikleri insan görüsünü yansıtmamaktadır. Uzamsal dönüşümlü örnekler piksel değerlerini doğrudan değiştirmek yerine piksel konumlarında bozulmalar yaparak görsel kalitesi yüksek çekişmeli örnekler üretir. Ancak, uzaysal dönüşümler tarafından yapılmış bozulmalar da parlaklık kanalında insanlar tarafından görülebilen pürüzsüz olmayan bozulmalara sebep olduğundan, bu yöntem görsel kaliteyi artırmak için bir pürüzsüzlük düzenlemesine ihtiyaç duymaktadır. Diğer yandan, insan görüsü görsel medyalardaki renk bileşeninin değişimine parlaklık değişiminden çok daha az duyarlıdır. Ayrıca kısıtlandırılmış komşuluklarda çözünürlük kaybı ve piksel kaymaları güçlükle fark edilebilmektedir. Bu çokluortam sıkıştırma gözlemlerinden yola çıkarak uzaysal dönüşümlü çekişmeli örneklerin YC_bC_r ve $CIELAB$ gibi algısal renk uzaylarının renk bileşenlerine uzaysal dönüşüm yapan ve görsel kalitesi yüksek çekişmeli örnekler çıkaran yeni bir varyasyonu önerilmiştir. Buna ek olarak, uzaysal dönüşümün büyüklüğünü sınırlayarak görsel kalitenin daha da artırıldığı gözlemlenmiştir. Hedefli beyaz-kutu kurulumunda, önerilen yöntem yüksek bir güven puanı ile rekabetçi bir yanıltma oranı yakalamaktadır. Deneysel değerlendirilmeler, önerilen yöntemin, zararsız ve çekişmeli örnekler arasındaki algısal uzaklık cinsinden tercih edilir sonuçlar ortaya çıkardığını göstermektedir.

Anahtar Kelimeler: derin öğrenme, çekişmeli örnekler, algısal kalite

For the damaged coda

ACKNOWLEDGMENTS

This work has been funded by The Scientific and Technological Research Council of Turkey, ARDEB 1001 Research Projects Programme project no: 120E093

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

DNN	Deep Neural Network
ANN	Artifical Neural Network
CNN	Convolutional Neural Network
VIT	Vision Transformer
JPEG	Joint Photographic Experts Group
CIE	International Commission on Illumination
VGG	Visual Geometry Group
ACM	Association for Computing Machinery
AI	Artifical Intelligence
DL	Deep Learning
NIPS	Conference on Neural Information Processing Systems
StAdv	Spatially Transformed Adversarial Examples
GPU	Graphics Processing Unit
RAM	Random Access Memory
VRAM	Video RAM
FGSM	Fast Gradient Sign Method
PGD	Projected Gradient Descent
BIM	Basic Iterative Method
C&W	Carlini & Wagner
TV	Total Variation
CAPTCHA	Completely Automated Public Turing Test to Tell Computers and Humans Apart
YUV	YUV color space: Y stands for luminance component and U and V stands for chrominance components

YCbCr	YCbCr color space: Y stands for luminance component and Cb and Cr stands for chrominance components
CIELAB	L*a*b* color space: L* stands for luminance component and a* and b* stands for chrominance components
LPIPS	Learned Perceptual Image Patch Similarity
SSIM	Structural Similarity Index Measure
MS-SSIM	Multi Scale Structural Similarity Index Measure
DISTS	Deep Image Structure and Texture Similarity
BFGS	Broyden-Fletcher-Goldfarb-Shannon Algorithm
L-BFGS	Limited Memory BFGS
HVS	Human Visual System
CUDA	Compute Unified Device Architecture

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

In recent years, deep neural networks have shown impressive performance in many vision related tasks such as image classification [1, 2], object detection [3, 4], image segmentation [5, 6] and visual media generation [7]. However, they are found to be vulnerable to intentionally crafted small perturbations called adversarial perturbations [8]. These small perturbations added to the input image successfully change the output of a trained classifier by altering the logits large enough to change its decision to a preferred class [9]. While these perturbations are optimized in $\mathcal{L}_p$ spaces [10], they are visible to human observers, since small $\mathcal{L}_p$ does not always correspond to perturbations with less perceptual distortion [11, 12].

Since this intriguing property of neural networks has security implications in the production setting, it is often called *adversarial attacks*. For example, some websites use CAPTCHAs (Completely Automated Public Turing Tests to Tell Computers and Humans Apart) to avert automated request agents such as web scrapers and spiders from automatically reaching their content. A widely used CAPTCHA type works by asking users select the images containing specific objects such as "bicycle" or "crossroad". A human can effortlessly detect the probed images while web scrapers would need to use classifiers such as DNNs to classify each image and select the correct ones. Such image based CAPTCHAs distort the visual content so that automated CAPTCHA solvers would fail the test, while the content could still be recognized by humans. To do this, recent CAPTCHAs has extended this setup by adding adversarial perturbations to the provided images to prevent DNN classified based solvers from automatically selecting the correct images. In Figure 1.1, a recent CAPTHCA from Google is shown where the user is asked to select the images that contains "cars". Normally, a trained DNN could be used to classify images and to select the images whose output is the class "car" to pass the test. However, when these images are inspected, it could be seen that there is a noticeable adversarial noise added to the images to fool the DNN and to prevent the attacker from passing the test. When the amount of the additive noise gets high, it starts to become more difficult for humans to recognize the images, which is an undesirable side effect, and this negatively affects the usability of CAPTCHAs.

Multimedia compression standards have been developed to compress visual media such as images and videos to reduce the amount of data with minimum amount of distortion on the perceived output. One of the most fundamental findings about hu-

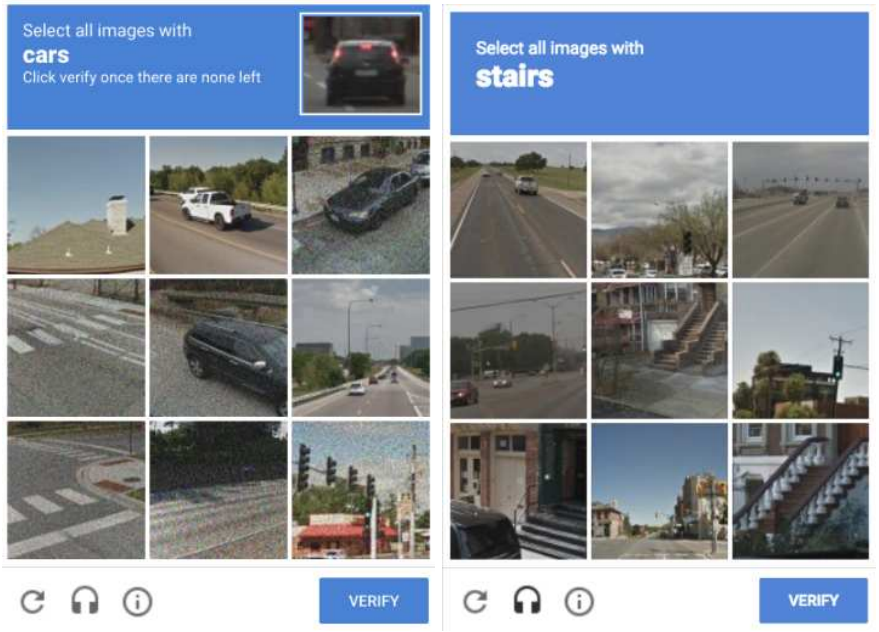
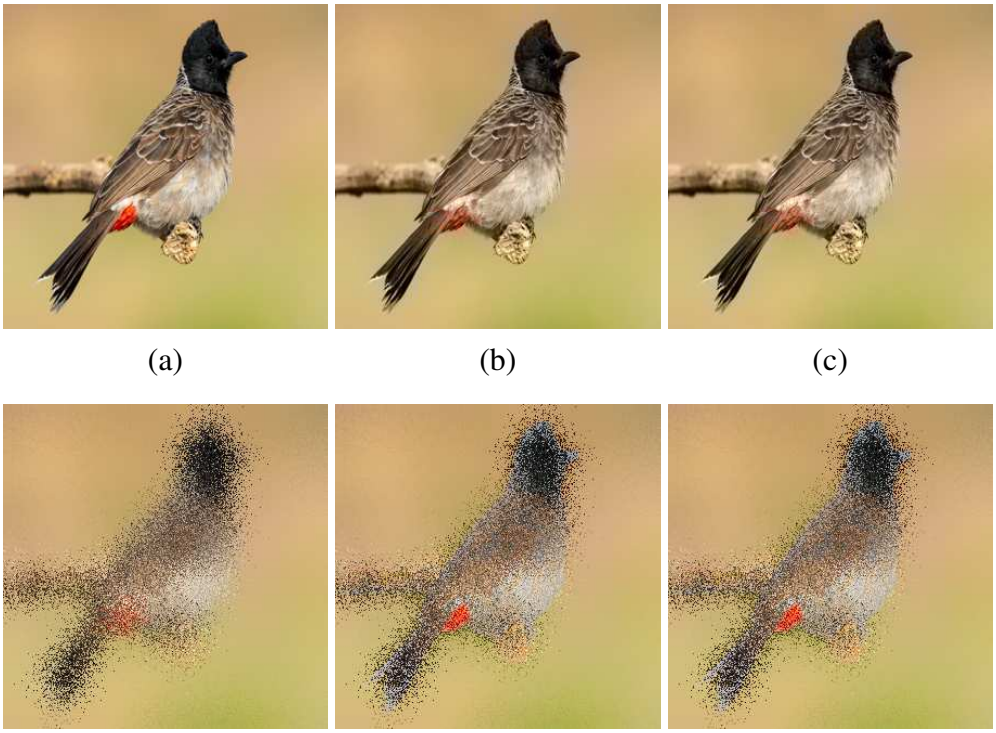


Figure 1.1: Two CAPTCHA examples from Google where adversarial examples are used to prevent web scrapers and spiders that are using automatic image classifiers such as DNNs by adversarially perturbing the CAPTCHA images.

man vision that most lossy visual multimedia compression methods utilize is that human vision is much less sensitive to the spatial information and resolution loss in chrominance (color) than the luminance (intensity) [13]. This observation is utilized in image compression as a technique known as "chroma subsampling". There are variants of chroma subsampling that only subsamples chrominance along horizontal axis (4:2:2) or both horizontal and vertical axes (4:2:0). Without further compression, (4:2:0) chroma subsampling reduces the size of an image effectively to half of its original size. Replacing the chroma components of the pixels in by neighboring chroma components does not yield visible artifacts. We employ these observations to derive a new type of adversarial attack based on spatial transformations in chroma channels of perceptual colorspaces. We apply spatial transformation only to the chroma components of input image while keeping the luminance component intact. Figure 1.2 shows the effect of a randomly initialized flow field applied to the luminance, chrominance and both set of channels. It is clear that spatial transformation in luminance channels causes visible distortions while chrominance only spatial transformations cause very subtle changes for human vision. This effect is much more highlighted when only the differences are observed after applying a flow field. Figure 1.3 shows the absolute pixel difference from the initial image when the same flow field is applied to RGB, C_bC_r and a*b* channels, respectively.

1.2 Contributions of the Study

The main findings of this thesis is published in ACM (Association for Computing Machinery) ADVM '21: Proceedings of the 1st International Workshop on Adversar-



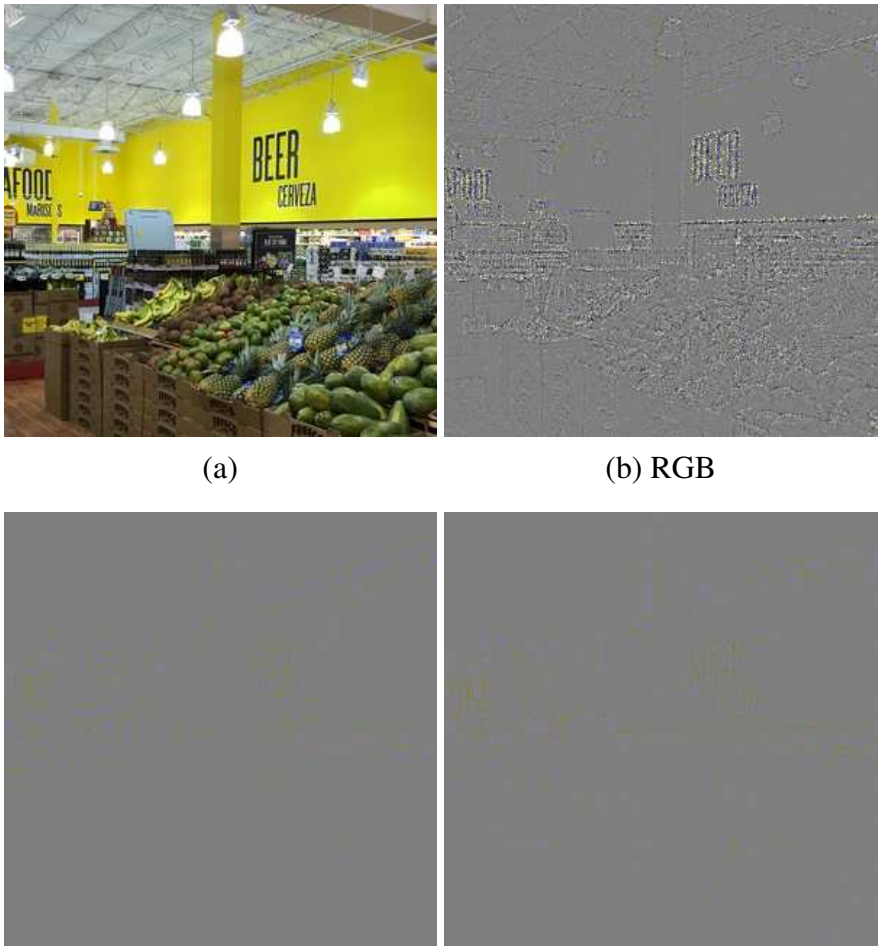
(a) (b) (c)

(d) (e) (f)

Figure 1.2: Effect of flow field applied to different channels, (a) original image, Images where flow field is applied to (b) C_bC_r, (c) a^*b^*, (d) RGB, (e) Y and (f) L channel. The magnitude of the flow is scaled up to emphasize the effect for illustration.

ial Learning for Multimedia as a workshop paper named Imperceptible Adversarial Examples by Spatial Chroma Shift [14] and the subject matter is further extended and elaborated in this thesis. The contributions of this work can be summarized as following;

- Utilization of the findings of human vision and ideas from image and video compression to make imperceptible changes on images without any explicit L_p norm restriction.
- A novel method to generate adversarial examples with little to no perceptual distortion by applying spatial transformations in chroma channels of perceptual colorspaces optimized by widely used gradient-based optimizers without requiring any regularization term in the loss function.
- Modification of the proposed method to restrict the flow magnitude to have subpixel spatial drifts to further improve perceptual quality with a performance trade-off.
- Colorfulness analysis of NIPS2017 Adversarial Challenge dataset and effect of the colorfulness of input image to the performance of proposed methods.



(a) (b) RGB

(c) a*b* (d) CbCr

Figure 1.3: Visual difference from random flow field application to different channels, (a) original image, Visualization of pixel differences where flow field is applied to (b) RGB, (c) C_bC_r, (d) a^*b^* channels. The magnitude of the flow is scaled up and contrast of the pixel differences is increased to increase the visibility for illustration.

1.3 Organization of the Thesis

This thesis is organized as follows. Chapter 1 provides an introduction on the thesis topic, explaining the motivation, defines the problem and briefly explains the methods and contributions of the thesis. Chapter 2 mentions the literature about the thesis topic, presents the types and classifications of adversarial attacks and methods for generating imperceptible types of adversarial attacks or methods to improve perceptual quality of adversarial examples. Chapter 3 explains the methodology of the method proposed in this thesis in a detailed manner. It starts with spatial transformations, then explains spatially transformed adversarial examples and colorspaces to build the foundation of this thesis. Then, it explains the method proposed in this thesis. Chapter 4 mentions the setup of the experiments and presents the experimental results as well as analysis of numerical results from the experiments with a brief

discussion. Chapter 5 provides a detailed discussion about the results presented in Chapter 4, explains and discusses the implications of the findings and mentions the failure cases, discussing the possible reasons and potential remedies. Chapter 6 draws conclusions on the thesis along with possible future studies of the new research questions with the thesis.

CHAPTER 2

RELATED WORK

In this chapter, related studies are given in detail. Firstly, the concept of adversarial examples and the types of adversarial examples are explained briefly. Then, spatially transformed adversarial examples are explained in detail.

2.1 Adversarial Examples

The concept of adversarial examples were introduced by Szegedy et al. [8]. They found that adding small calculated perturbances to the input image is able to change the decision of the target classifier (deep neural network) without affecting the decision human observers. Even though the perturbation is small, it can be easily noticed by human observers as "visual noise" instead of semantically meaningful patterns. A widely used example for adversarial examples is shown in Figure **??**

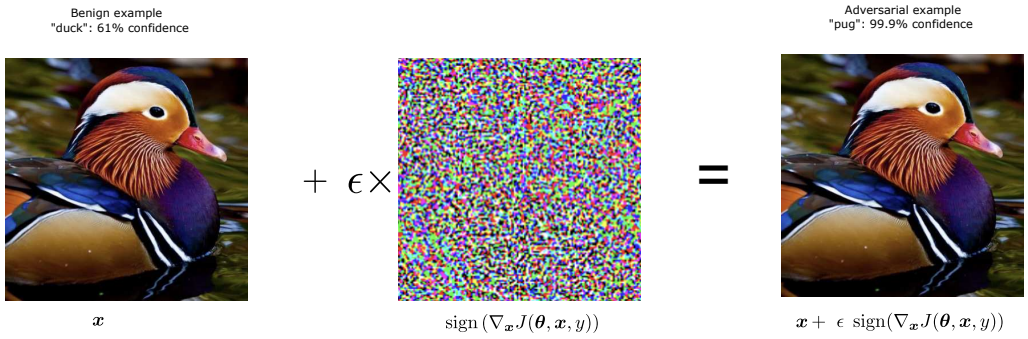
White-box and Black-box Attacks

Adversarial attacks can be classified depending on whether the attacker has access to the neural network parameters and gradients. White-box attacks has access to all the parameters and gradients of the network and adversarial examples are generated by direct optimization. Black-box attacks are generally done by generating adversarial example by attacking a proxy network with white-box transferable attack methods [15] and use generated examples against the target network.

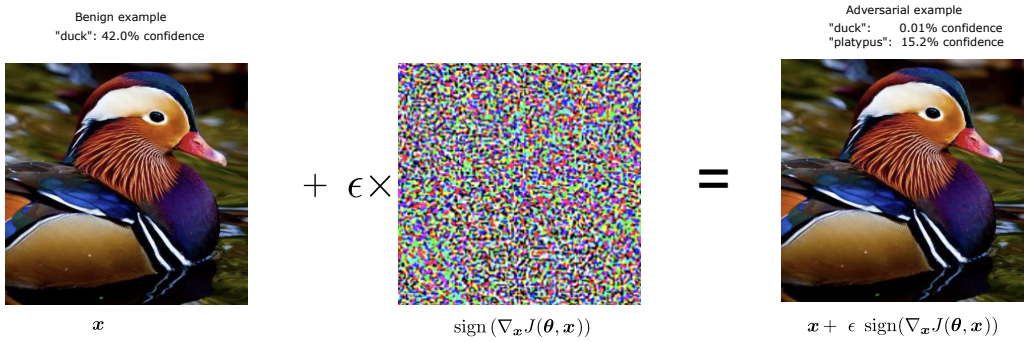
Targeted and Untargeted Attacks

Adversarial attacks are also classified as targeted or untargeted (evasion) attacks. In a targeted attack setup, the attack is considered successful if the network outputs the particular target class (which is different from a ground-truth label) when fed with the modified adversarial example. On the other hand, an untargeted attack is considered successful if the network outputs any class other than the true class label. This is generally accomplished by selecting a suitable cost function.

Adversarial attacks can be formally defined as a constrained optimization problem where the attacker tries to maximize a loss function by perturbing the input while the



(a)



(b)

Figure 2.1: Visual illustration of (a) targeted and (b) untargeted adversarial examples. In this example, FGSM is illustrated for its simplicity.

magnitude of the perturbation is constrained. Generally, the adversarial loss is the loss that is minimized throughout the training process. In its most general form, the optimization process can be formulated as in Equation 2.1. There is often a magnitude constraint to the perturbation and $\mathcal{L}_p$ norms are widely used to measure the magnitude of the added perturbation.

$$\underset{\|\delta\| \leq \epsilon}{\text{maximize}} \ell\left(h_\theta(x+\delta), y\right) \tag{2.1}$$

The equation simply indicates that the adversary is aiming to maximize a loss function of the adversarial image and the target label with the norm of the perturbation is upper bounded by ϵ. In an untargeted attack setup, the function is a single argument function of the adversarial image.

Since naively trained models are vulnerable to adversarial perturbations, the concept of adversarial robustness gained importance and several adversarial robustness methods and benchmarks have been introduced [16, 17, 18, 19, 20]. In adversarial training, the datapoints are sampled from both the original training set and on-the-

fly generated adversarial images. Although the training time significantly increases, since adversarial example generation needs at least one forward and backward pass on the model, this method has been shown to increase the adversarial robustness of the trained model. Also, the concept of universal adversarial perturbations is being investigated after DNNs have been found to be vulnerable to perturbations that is independent of the input image. [21]

2.2 Adversarial Example Generation Methods

There are several methods for adversarial example generation and the most popular ones are explained in this section. The variables used in this section are defined as follows;

- x: Benign image
- x_{adv}: Adversarial image
- y: Target class
- $J(x, y)$: Cost function with respect to the benign image and the target class
- $Z(x)_i$: Score of i^{th} class on input image x

2.2.1 Fast Gradient Sign Method

Goodfellow et al. found that local linearity of DNNs leads to l_∞ vulnerability [9] and proposed the Fast Gradient Sign Method (FGSM) method for generating adversarial examples with l_∞ norm constraint by using the sign of gradient of the loss with respect to the input. The formal equation of the adversarial image generation process is given in Equation 2.2

$$x_{adv} = x + \epsilon * \text{sign}(\bigtriangledown_x J(x, y)) \tag{2.2}$$

According to the authors, this method works even when ϵ is too small since the deep neural networks are actually linear in small local epsilon neighborhood of the input point. This method is typically used as a fast and easy to compute baseline in most studies.

After FGSM, many iterative methods have been proposed for adversarial example generation with $\mathcal{L}_p$ norm constraints. The most prominent and widely used iterative algorithms are Projected Gradient Descent (PDG) and Carlini & Wagner method (C&W).

2.2.2 Basic Iterative Method & Projected Gradient Descent

Basic Iterative Method (BIM) [22] can be simply thought as the iterative version of FGSM except that at each iteration, the adversarial example is clipped to the ϵ

neighborhood of the benign datapoint. This has been found to be a useful heuristic since it allows generating more successful examples in terms of fooling rate without requiring significant computational budget.

Projected Gradient Descent (PGD) [16] is very similar to BIM except that at each iteration projection operation instead of clipping is used to pull the adversarial example back to the $\mathcal{L}_p$ ball around the original datapoint if necessary so that the $\mathcal{L}_p$ constraint is satisfied throughout the adversarial example generation process. This method is found to provide better robustness for the models trained with adversarial training than FGSM when training. The iteration steps of PGD and BIM for adversarial example generation are formalized in Equations 2.3 and 2.4, respectively. This process is repeated until the attack is successful or the maximum number of iterations is reached.

$$v^{i+1} = \text{Clip}_{\epsilon}\left\{v^i + \alpha \,\text{sign}\left(\nabla_{x+v^i} J\left(x + v^i, y\right)\right)\right\} \tag{2.3}$$

$$v^{i+1} = \text{Project}_{\epsilon}\left\{v^i + \alpha \,\text{sign}\left(\nabla_{x+v^i} J\left(x + v^i, y\right)\right)\right\} \tag{2.4}$$

2.2.3 Carlini & Wagner Attack

Carlini & Wagner [10] attack is an iterative method for generating adversarial examples. It uses reformulation of the constrained optimization objective by defining a new variable w so that the constraints are naturally satisfied without requiring an explicit step for clipping or projecting the adversarial example in the $\mathcal{L}_p$ ball. The optimization process is formulated in Equation 2.5 and Equation 2.6 where κ denotes the target confidence for the adversarial example, which is the aimed score difference between the target class and the non-target class with the highest score.

$$\text{minimize} \left\| \frac{1}{2}(\tanh(w) + 1) - \boldsymbol{I} \right\|_2^2 + c \cdot f\left(\frac{1}{2}(\tanh(w) + 1)\right) \tag{2.5}$$

where

$$F(m) = \max\left(\max_{i \neq t}(Z(m)_i) - Z(m)_t, \kappa\right) \tag{2.6}$$

2.3 Perceptual Colorspaces

In most image processing tasks, standard RGB colorspace is used, where R, G, B denotes the Red, Blue, Green components of the image. However, it is incompatible with Human Visual System (HVS) so the need for HVS compatible (perceptual) colorspaces arises for the tasks concerning HVS such as lossy visual media compression where the media is compressed to reduce the size without affecting the perceived

output. In this thesis, standard RGB colorspace and the perceptual colorspaces YUV, YC_bC_r and CIELAB are used for experiments.

2.3.1 YUV and YCbCr

The YC_bC_r model defines a luminance component (Y) and two chrominance components C_b, C_r to specify color. YC_bC_r is a colorspace that is used in digital photography and visual media compression. In this space, luminance (brightness) and chrominance (color) is separated according to human visual perception. Y dimension of the space is the luminance information, or simply a grayscale representation of the image. C_b and C_r dimensions are the blue-difference and red-difference chroma components, respectively. The relation between RGB space and YC_bC_r space is modeled as Equation 2.7, which is a set of linear equations defined in ITU-T H.273 [23];

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ C_b &= 128 - (0.168736R) - (0.331264G) + (0.5B) \\ C_r &= 128 + (0.5R) - (0.418688G) - (0.081312B) \\ R &= Y + 1.402(C_r - 128) \\ G &= Y - 0.344136(C_b - 128) - 0.714136(C_r - 128) \\ B &= Y + 1.772C_b - 128 \end{aligned} \tag{2.7}$$

YUV is often considered as the analog counterpart of YC_bC_r. There is no significant difference between YUV and YC_bC_r except for the scaling of chrominance components to the range 0-255 for YC_bC_r so that they can be represented as unsigned integers while chrominance values can be negative in YUV colorspace. For that reason, these two terms are often used interchangeably.

2.3.2 CIEXYZ and CIELAB

CIE 1931 XYZ (CIEXYZ) colorspace was proposed by International Commission of Illumination (CIE) in 1931 after a series of human experiments on color perception. There are three components, namely X, Y and Z. While Y denotes the luma component, there is no independent representations of XZ components and these components together represent possible chrominance values for given Y value. CIELAB colorspace [24] defined by the International Commission on Illumination (CIE) has the following three components: L, a^* and b^*. L is perceptual lightness where $L = 0$ and $L* = 100$ define a black and a white pixel, respectively, regardless of the a^* and b^* values. a^* and b^* dimensions are the chroma components. They are designed to be perceptually uniform where a numerical change in pixel value corresponds to a similar change in human perception [25]. Both chroma components are in the range $[-127, 127]$. Unlike YC_bC_r, CIELAB space does not have a linear relationship with RGB space. In fact, conversion to an intermediary space CIEXYZ is needed to transform from RGB to CIELAB and there are different implementations of CIELAB

conversion. We used the RGB to CIELAB implementation from Kornia library [26], which assumes D65 illuminant and Observer 2.

2.4 Perceptual Distance Metrics and Similarity Measures

There is an ongoing research on finding difference metrics over 2D images that aligns with human visual perception, which is challenging due to the nature and lack of knowledge about the human vision. [27]. There are several studies proposing perceptual metrics with different methods. The perceptual metrics used in this thesis is briefly explained.

2.4.1 Structural Similarity Index Measure (SSIM)

Structural Similarity Index Measure is a method for measuring similarity between two images. Although it is a naive method for measuring difference between two images, it is considered to be a perceptual difference metric since it takes structural information into account. SSIM calculates 3 different comparison measures (luminance, contrast and structure) between two images x and y, which is shown in Equation 2.8 where l denotes luminance, c denotes contrast and s denotes structure measure.

$$\begin{aligned} l(x,y) &= \frac{2\mu_{:x}\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \\ c(x,y) &= \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_3} \\ s(x,y) &= \frac{\sigma_{xy} + c_2/2}{\sigma_x\sigma_y + c_2/2} \end{aligned} \tag{2.8}$$

with

- μ_x: mean of x
- μ_y: mean of y
- σ_x: standard deviation of x
- σ_y: standard deviation of y
- σ_{xy}: covariance of x and y
- L: $2^{\#bitsperpixel} - 1$
- c_1: $0.0001 \times L^2$
- c_2: $0.0009 \times L^2$

Using these three measures, SSIM is calculated according to Equation 2.9

$$SSIM(x,y) = l(x,y) \times c(x,y) \times s(x,y) \tag{2.9}$$

Although there are many variations of SSIM, original SSIM and Multi-Scale SSIM (MS-SSIM) are used in this thesis. The difference between SSIM and MS-SSIM is MS-SSIM is computed over multiple scales of the compared images through multiple scales of subsampling.

2.4.2 Learned Perceptual Image Patch Similarity (LPIPS)

Zhang et al. found that DNN representations of images are very effective for approximating perceptual similarity of two images, as supported by human experiments [28]. They proposed Learned Perceptual Image Patch Similarity to assess perceptual distance between images. It works by simply computing the Euclidean distance between deep representations of trained deep convolutional networks. They experimented with different DNN architectures and have not observed significant difference. In this thesis, ImageNet trained VGG-16 is used for LPIPS computation.

2.5 Perceptual Quality Preserving Adversarial Attacks

Utilizing perceptual colorspaces and metrics for imperceptible adversarial example generation is investigated in several studies. Aksoy et al. investigated additive noise based attacks on chrominance channels in YUV colorspace [29], which is the analog counterpart of YC_bC_r space. Despite Pestana et al. found that adversarial perturbations are more highlighted in luminance channels in terms of the magnitude [30], Aksoy et al. found that even suppressing the luminance perturbation, additive noise based attack on chrominance channels still successfully fool target networks, yet causes visible distortion. In our earlier work, we also explored spatial transformations to UV channels of YUV to generate imperceptible adversarial examples [14] and we extend this work by exploring YC_bC_r space as well as perceptually uniform CIELAB space and measuring structured similarity metrics such as SSIM [31] and MS-SSIM [32] between benign images and adversarially generated images. Karli et al. leveraged perceptual metric LPIPS [28] to improve the quality of adversarial examples. Since LPIPS is a differentiable metric, they used gradient based optimization to minimize LPIPS alongside the adversarial loss. Similarly, Zhao et al. [33] replaced CIEDE2000 perceptual distance metric [34] with $\mathcal{L}_p$ norm constraint in Carlini & Wagner attack to produce perceptually close adversarial examples. In addition, Functional Adversarial Attacks [35] modifies the input image by applying a parametric function of input pixels to generate adversarial examples. With this method, the perturbation is not perceived as a visual distortion by human observers since it does not modify the input by additive noise. However, it is often clearly visible as slight chrominance, luminance or contrast changes.

Croce et al. argued adding noise to smooth areas of an image causes visible artifacts and proposed "hiding" the perturbations at the locations with high spatial variations such as edges and corners [36]. As seen in Figure 1.3, perturbations generated with our method naturally occurs in the places with high variations since it is based on local spatial transforms. Also, since the differences made with our methods affect only the chrominance channels, visualizing these differences apart from luminance component

still yields barely noticeable changes while full RGB flow difference is significantly noticeable when visualized on its own. Similarly, Karli et al. proposed increasing the intensity of adversarial perturbation in the regions that have high spatial variance such as edges and corners and attenuating the amount of the overall perturbation by minimizing LPIPS distance between benign and adversarial image. Since LPIPS is differentiable, the minimization can be modeled as an optimization step or an extra term in the cost function.

Unlike these methods, the attack proposed in this paper does not rely on auxiliary losses or explicit perceptual distance terms in optimization process to produce examples with high perceptual quality. In addition, it does not require regularization, unlike spatial transformation based methods such as [37], due to its intrinsic imperceptibility. It should be noted that the existing spatial transformation based methods, as well as our work, does not utilize limited degree of freedom transformations such as rotation, translation or scaling that can be formulated as a 4×4 transformation matrix [38]. In that formulation, the flow field $f \in \mathbb{R}^{2\times H\times W}$ is calculated using the transformation matrix. Instead, we directly define and optimize flow field, where the number of parameters is equal to twice number of pixels in the input image since there is an x and y component for each pixel.

2.5.1 Spatially Transformed Adversarial Examples

Spatial transformations as a method for generating adversarial examples was first proposed in [37], where it is shown that small displacements applied to input pixels can successfully fool a target network. However, using this method, even small displacements could cause visible distortions when the adjacent pixels drift towards different directions. As a remedy to this problem, use of Total Variation (TV) regularization [39] was proposed. Application of TV regularization to the flow field pushes the neighboring displacement vectors to the same direction and, hence, produces smoother output. Similarly, Jordan et al. [11] combined spatial transformations with l_∞ bounded attacks to forge stronger attacks with better perceptual quality.

Spatial transformations aim to alter the geometry of the input image instead of changing the pixel values. To accomplish that, Spatially Transformed Adversarial Examples (stAdv) applies a flow field $f \in \mathbb{R}^{2\times H\times W}$ whose elements are flow vectors (or displacement vectors) f_i for each pixel in the adversarial image. Since the elements of displacement vectors are not integers, a need for interpolation to sample fractional positions arises. In this work, bilinear interpolation is used since it is computationally efficient. The application of flow field to the benign image is formulated in Equation 2.10 where $\mathbf{x}_{\text{adv}}^{(i)}$ denotes the value of i^{th} pixel in the adversarial image and u_{adv},v_{adv} denotes the position of that pixel in the adversarial image.

$$\mathbf{x}_{\text{adv}}^{(i)} = \sum_{g\in\mathcal{N}\left(u^{(i)},v^{(i)}\right\rangle} \mathbf{x}^{(q)} \left(1 - \left|u^{(i)} - u^{(q)}\right|\right) \left(1 - \left|v^{(i)} - v^{(q)}\right|\right) \tag{2.10}$$

Since bilinear interpolation is differentiable, application of the flow field is also a differentiable operation and can be optimized by gradient based optimization methods.

Carlini & Wagner loss in Equation 2.6 with a Total Variation regularization term $\mathcal{L}_{flow}$ for flow field f to reduce the high frequency pixel drift distortion is minimized for adversarial optimization. The TV loss term is shown in Equation 2.11. The optimization is made with L-BFGS [40] with linear backtracking, however the authors have stated that Adam [41] optimizer could be used as well.

$$\mathcal{L}_{\text{flow}}(f) = \sum_{p}^{\text{all pixels}} \sum_{q \in \mathcal{N}(p)} \sqrt{\|\Delta u^{(p)} - \Delta u^{(q)}\|_2^2 + \|\Delta v^{(p)} - \Delta v^{(q)}\|_2^2} \tag{2.11}$$

CHAPTER 3

METHODOLOGY

In this work, we address the problem of creating targeted adversarial examples without adversarial perturbation being perceptible by human vision. For this purpose, we use a modified version of Spatially Transformed Adversarial Examples [37] that perturbs the input image only in the channels that human vision is not sensitive to the spatial information loss in YC_bC_r and CIELAB colorspace representations of the input image.

3.1 Proposed Method

The proposed adversarial example generation method is as follows. Let $x \in \mathbb{R}^{3\times H\times W}$ be the 3-channel input image, where H, W are the height and the width of the image, respectively. First, we randomly initialize a flow field $f \in \mathbb{R}^{2\times H\times W}$ where a two-dimensional vector exists for each pixel location of the adversarial image x_{adv}. Then, we apply the flow field to the benign image as explained below to obtain the adversarial image. Then, we feed the adversarial image to the target network and backpropagate the loss gradient to the flow field. Since the flow field application is a differentiable process, it can be optimized by stochastic gradient descent and variants such as Adam [41] or L-BFGS[40]. The optimization process is repeated until the attack is successful or the maximum iteration count is reached. Visual illustration of the adversarial image generation methodology is shown in Figure 3.1. The pixel values of the output image are calculated by sampling the pixels from the input image from the positions according to the flow field f using Equation 3.1 and applying bilinear interpolation formula shown in Equation 2.10, where u^i and u^i_{adv} denotes the corresponding pixel locations of benign and adversarial image, and Δu^i and Δv^i denotes the values of the flow vector at that position of flow field f, respectively.

$$\begin{aligned} u^i &= u^i_{adv} + \Delta u^i, \\ v^i &= v^i_{adv} + \Delta v^i, \end{aligned} \tag{3.1}$$

3.1.1 Application of Flow Field

Flow field is applied to the benign image following the methodology in [37] also explained in Chapter 2. For each pixel in adversarial image i_{adv}, corresponding flow

Algorithm 1: Adversarial example generation by spatial transformation in chrominance channels in a perceptual colorspace.

Input: x
Output: x_{adv}
Data: target_class, model, κ, colorspace, max_iters, is_restricted,
$f \sim \mathcal{N}(0,\, \sigma^2)$;
$i \leftarrow 0$;
while $i < max\_iters$ **do**
  if $colorspace == YC_bC_r$ **then**
    $x_{color} \leftarrow to\_ycbcr(x)$;
  end
  if $colorspace == CIELAB$ **then**
    $x_{color} \leftarrow to\_lab(x)$;
  end
  $x_{luma}, x_{chroma} \leftarrow splitchannels(x_{color})$;
  if *is_restricted* **then**
    $f \leftarrow \tanh(f)$
  end
  $x_{chroma} \leftarrow apply\_flow(x_{chroma}, f)$;
  $x_{adv} \leftarrow concat(x_{luma}, x_{chroma})$;
  $x_{adv} \leftarrow to\_rgb(x_{adv})$;
  $adv\_scores = model(x_{adv})$;
  $loss \leftarrow loss\_fn(adv\_scores, target\_class, \kappa)$;
  if $loss \leq \kappa$ **then**
    return x_{adv};
  else
    $backprop(loss)$;
    $update(f)$;
    $i \leftarrow i + 1$;
  end
end

field vector value $p_{i,j}$ is added to the pixel location. Then, the corresponding pixel at the added location is sampled. Since the added location is not an integer, bilinear interpolation is used to sample from the fractional pixel locations. Bilinear interpolation also makes the method end-to-end differentiable, thus optimizable by gradient based optimizers.

3.1.2 Chrominance Restriction of Flow Field

Since applying a flow field to all channels or luminance channels of an image of perceptual colorspace yields visual distortions shown in Figure 1.2, the flow field is only applied to the chrominance channels where human vision is not very sensitive to the information loss [13] to make the adversarially perturbed images indistinguishable from their benign counterparts. Since widely used RGB colorspace is not designed to be a perceptual colorspace, even small spatial perturbations to any RGB channel

creates visually distinguishable changes. Hence, we first convert the benign image to a perceptual colorspace such as YC_bC_r where human vision is not sensitive to the spatial perturbations in, which is C_b and C_r in YC_bC_r, and a^* and b^* in CIELAB colorspace. Then, we apply the flow field only to the channels Cb and Cr in YC_bC_r, and A and B in CIELAB colorspace.

3.1.3 Subpixel Restriction of Flow Field

As mentioned in Chapter 1, chroma subsampling effectively causes the same chroma values to be used in the neighboring pixels by removing the local variation of chrominance. This method is widely used in visual lossy compression standards since the resolution loss in chrominance components of an image often does not cause any artifacts visible by a human observer. Accordingly, to exploit this fact, we can impose a restriction to the flow field to keep its values in the range $(-1, 1)$. We initialize a pre-flow field f_{pre} and calculate the applied flow field as $f = \tanh(f_{pre})$. This differentiable reparameterization [42] of flow field constraints the flow field magnitude to be smaller than 1 without inhibiting end-to-end differentiability so that chrominance value of each pixel of the adversarial image x_{adv} is only affected by the value of the pixel of the same location in x and its neighboring pixels.

3.2 Implementation Details

Flow field application is implemented in PyTorch [43], a scientific computation library for Python that has features facilitating automatic differentiation and Graphical Processing Unit (GPU) computation support which makes it suitable for deep learning applications. Initial flow field is randomly initialized from normal distribution with $\mu = 0$ and $\sigma = 0.01$. A common problem with stochastic gradient based optimizers is they are prone to be stuck in local minima. To mitigate this issue in our situation, a batch of randomly initialized flow fields is used instead of a singular one due to the fact that the gradient signal is only provided from the pixels of the direction of flow vectors. By using flow field batches, we effectively optimize many flow fields to maximize the performance. We used 32 batch size since it optimizes the utilization of our GPU, which is NVIDIA GTX 1080Ti with 8 GB of Video Random Access Memory (VRAM). For adversarial optimization of flow field, we used Adam optimizer [41] with learning rate $\gamma = 0.001, \beta_1 = 0.9, \beta_2 = 0.99, \epsilon = 10^{-8}$.

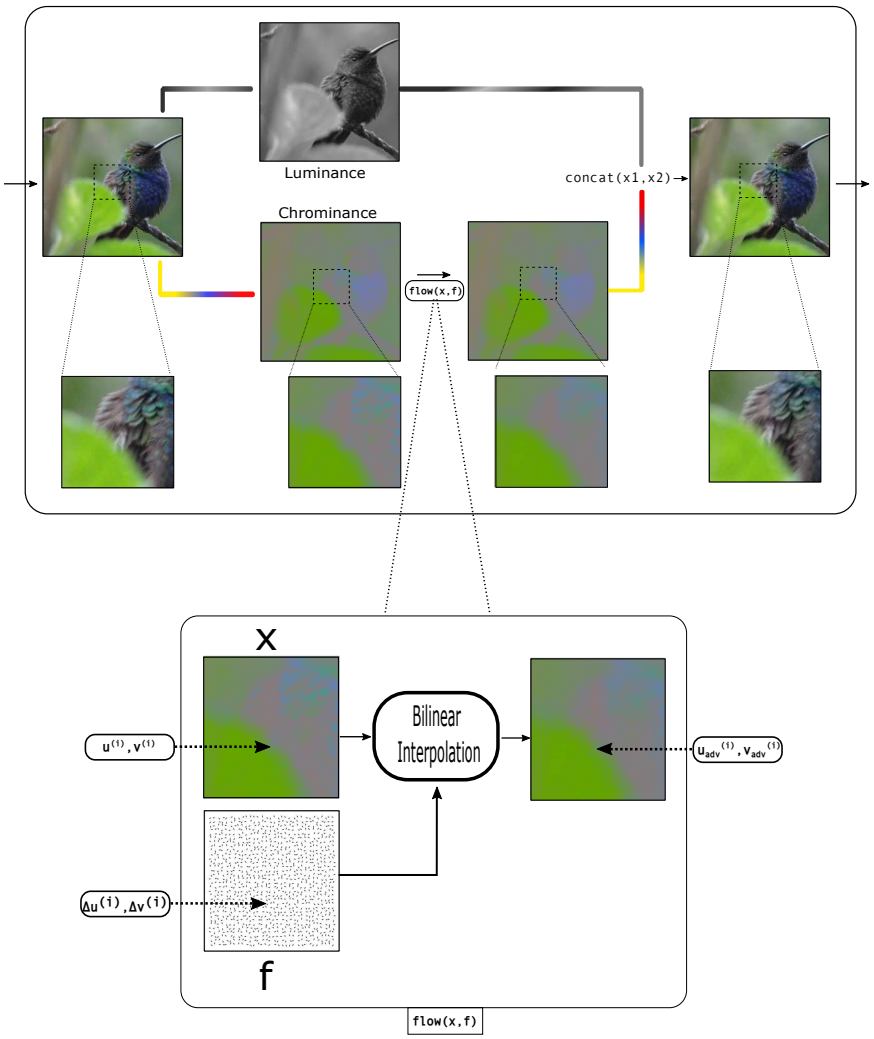


Figure 3.1: Visual illustration of the proposed adversarial example generation method. Luminance and chrominance channels are Y and C_bC_r when YC_bC_r colorspace and L and a^*b^* when CIELAB colorspace is used. Visual representation of flow field, subpixel restriction by $\tanh$ and conversion of concatenated image back to RGB colorspace is omitted for brevity.

CHAPTER 4

EXPERIMENTS

4.1 Dataset

We used the dataset and the provided model from NIPS 2017 Competition on Adversarial Attacks and Defenses [44] to evaluate our method. NIPS 2017 dataset is a collection of 1000 images curated by Google Brain with a resolution of 299×299 with their corresponding true and target classes from Imagenet [45] dataset. For the target network, we used Inception-v3 [46] architecture and Imagenet trained checkpoint provided with the dataset. Since this architecture accepts 299×299 images by default, no resize or crop is necessary for preprocessing.

4.2 Experimental Evaluation

We conducted our experiments in a white-box setup where the gradients are fully available. Experiments have been done in a targeted attack setting with the dataset provided targets. We optimized using Adam [41] with the default settings and used Carlini & Wagner loss [10] with a confidence margin of $\kappa \in \{0, 10\}$.

We compared the success rate of our attack in CIELAB and YC_bC_r against stAdv in both restricted and unrestricted settings. An attack is considered successful if the Carlini & Wagner loss is less than $-\kappa$. We did not use the smoothness regularization term in stAdv for a fair comparison.

Figure 4.1 shows the original images alongside with the adversarial images generated (with $\kappa = 10$) by attacking in a^*b^*, C_bC_r and RGB spaces. As can be observed from these images, perceptual distortions are much less pronounced for chrominance-only attacks. Attacking in RGB domain, which is the default approach in the literature, results in modification of the luminance channels, leading to much more visible artifacts.

Table 4.2 shows the attack success rates for attacks on different colorspaces. The results show that, adversarial images generated by attacks exclusively targeting the chrominance channels can fool the network with a high probability as well. On the other hand, they are less effective when restricted to operate in a subpixel-only setting. The fooling rate of a*b* attacks are slightly higher than C_bC_r attacks. We argue that this is due to many examples in the dataset being chroma subsampled in YC_bC_r space, as an indirect effect of image compression, restricting the search space for

Table 4.1: Average amount of distortion required to fool the target network with very high confidence ($\kappa = 10$) in not restricted and subpixel restricted settings.

	RGB	C_bC_r	a^*b^*
Not Restricted			
LPIPS	0.327	**0.019**	0.022
SSIM	0.321	**0.067**	0.070
MS-SSIM	0.164	0.017	**0.016**
Restricted to Subpixel			
LPIPS	0.222	**0.012**	0.014
SSIM	0.220	**0.050**	0.056
MS-SSIM	0.037	**0.011**	0.013

Table 4.2: Attack success rates with $\kappa = 0$ and $\kappa = 10$ in not restricted and subpixel restricted settings for RGB, a^*b^* and C_bC_r attacks.

	RGB	C_bC_r	a^*b^*
Not Restricted			
κ = 0	100%	95.0%	95.7%
κ = 10	100%	83.8%	87.3%
Restricted to Subpixel			
κ = 0	99.8%	86.1%	89.2%
κ = 10	99.7%	47.0%	53.2%

C_bC_r attacks.

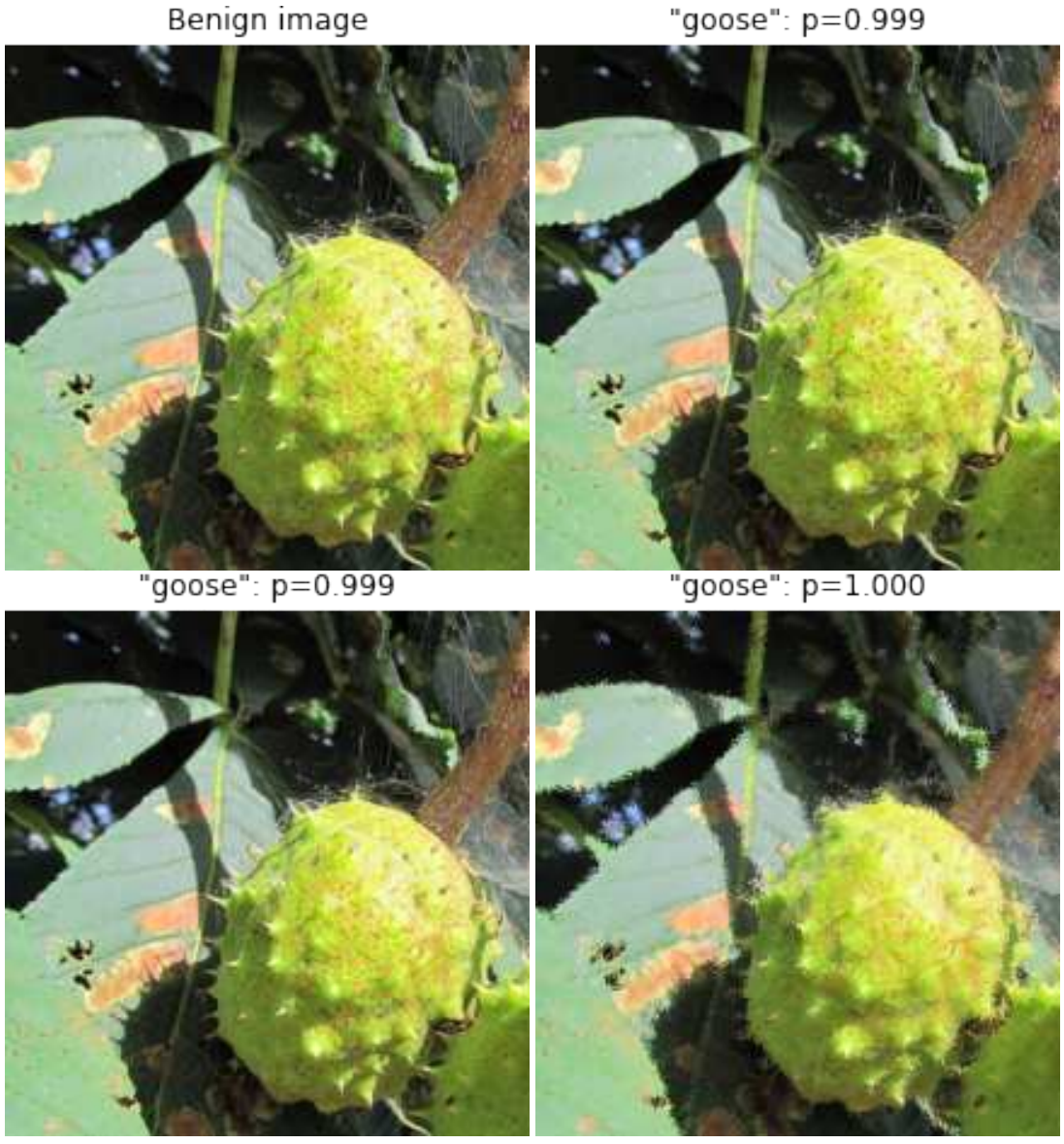
We measured the amount of distortion required to generate confident ($\kappa = 10$) adversarial examples with the following perceptual metrics: Learned Perceptual Image Patch Similarity (LPIPS) [28], Structured Similarity Index (SSIM) [31] and Multi-Scale SSIM (MS-SSIM) [32]. Table 4.1 shows the average results over the successful attacks for each perturbation mode in terms of these metrics. Since SSIM and MS-SSIM are similarity metrics, values of $1-$SSIM and $1-$MS-SSIM are provided. Hence, for all metrics, lower values are better. According to these results, colorspace restricted attacks have significantly better scores in terms of perceptual metrics compared to RGB attacks, implying that there is significantly less perceptual difference between benign and adversarial examples. While C_bC_r attacks generally produce better images in terms of perceptual quality metrics than a*b* attacks, the difference is relatively low.

(a)


Benign image
"weighing machine": p=0.998
"weighing machine": p=0.998
"weighing machine": p=0.999

(b)

(c)



(d)


Benign image
"analog clock": p=0.999
"analog clock": p=0.998
"analog clock": p=0.999

(e)


Benign image
"miniature poodle": p=0.999
"miniature poodle": p=0.999
"miniature poodle": p=1.000

(f)

Figure 4.1: Examples from the dataset and adversarial examples generated with their target class probabilities from target network Inception-v3. Benign image (top left), adversarial image generated by attacking to CbCr(top right), a*b*(bottom left) and RGB(bottom right) channels.

CHAPTER 5

DISCUSSION

As it can be seen in Figure 5.4, the input images that our method fails are generally grayscale or monochromatic images, which prevents chrominance spatial transforms from changing the pixel values due to the low magnitude of chrominance channel values. In addition, input images having a very limited local color variation negatively affect the performance by limiting the potential search space. We observed that there is a significant drop in the success rate with the setup confidence margin $\kappa = 10$ if the attack is restricted to subpixel changes in comparison to the unrestricted attacks. We argue that this performance drop is arising from the fact that the most examples are already JPEG compressed, which means chroma subsampling is applied to the benign examples, which restricts the subpixel restricted search space by dramatically reducing the local chrominance variation. This leads to the observation that chroma subsampling could be an effective defense method against our attack. Moreover, the search space is further restricted in JPEG compressed images as the quantization step of JPEG compression attenuates high frequency information, especially in the chrominance channels. Nonetheless, we observed adversarial examples generated by spatial transforms in chrominance channels of perceptual colorspaces obtain competitive fooling rates without making perceptible changes to the image. This observation provides further evidence for the hypothesis that representation of deep neural networks does not necessarily align with human vision [47].

Experimental results show that there are two main restrictions of the proposed method: out of gamut values in the chrominance channels emerging during optimization leading to visible artifacts and failing to generate adversarial images when the original image has limited colorfulness.

5.1 Out of Gamut Values

Modifying the chrominance channels in YC_bC_r and CIELAB spaces may lead to invalid values on individual RGB channels since some luminance and chrominance pairs do not correspond to a valid RGB value, as shown in Figure 5.1. This is also common in widely used chroma subsampling and mitigating this issue is an open research topic [48]. In our work, we clip the reconstructed RGB to the valid range and feed the target network with the clipped image at each iteration to prevent further change in the pixel values out of the gamut. Clipping also zeroes out the gradient and prevents further updates in gradient based optimization. However, we found that it still causes visible artifacts in the adversarial image, especially around the

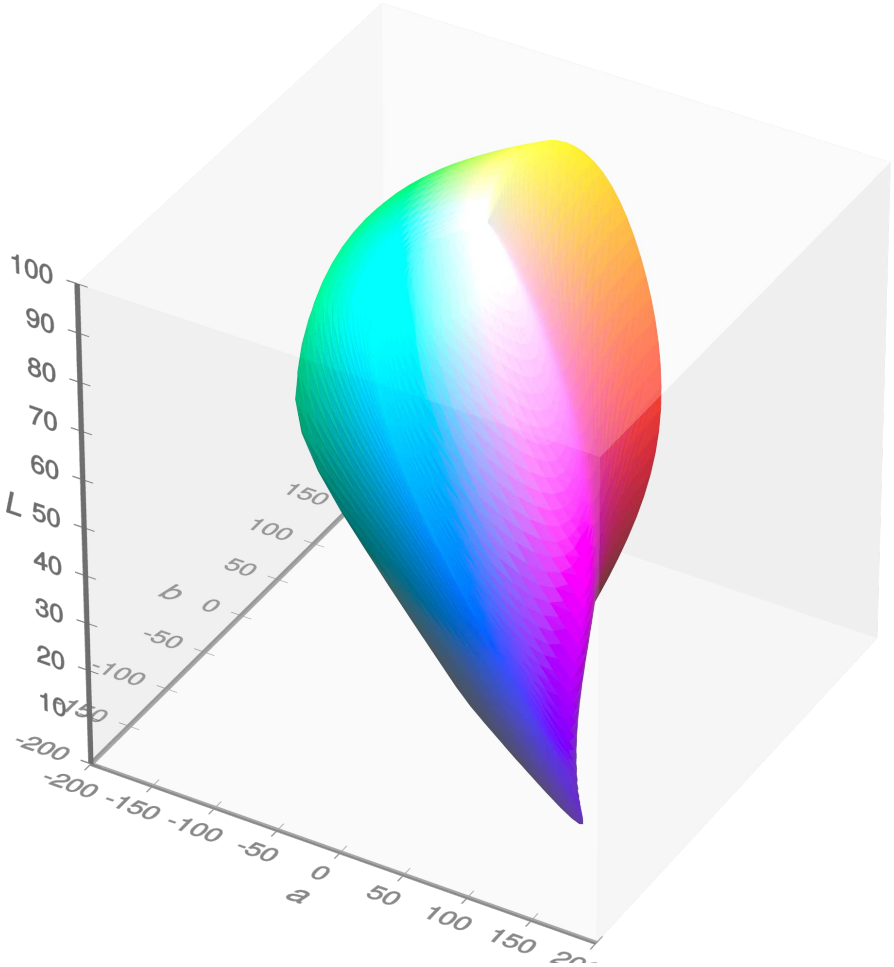


Figure 5.1: Visible gamut for CIELAB colorspace. Any value outside of this volume yields to an invalid RGB, such as negative pixel values.

borders between red and gray tones. Figure 5.5 shows two examples where spatial transformation in red-gray borders yield out of gamut pixels and clipping the values still causes visible artifacts since clipping in RGB space effectively changes the values of luminance channels.

5.2 Failed Attacks on Less Colorful Images

Results in Table 4.2, show that the attack success rate does not reach 100% when spatial transform attack is restricted to chrominance channels. This implies that the chrominance based attacks fail for a number of images in the dataset. Examples of such images are provided in Figure 5.4. We observed that these particular images are either monochromatic examples or have a uniform color pattern, for which spatial transformation in a neighborhood leads to insignificant changes.

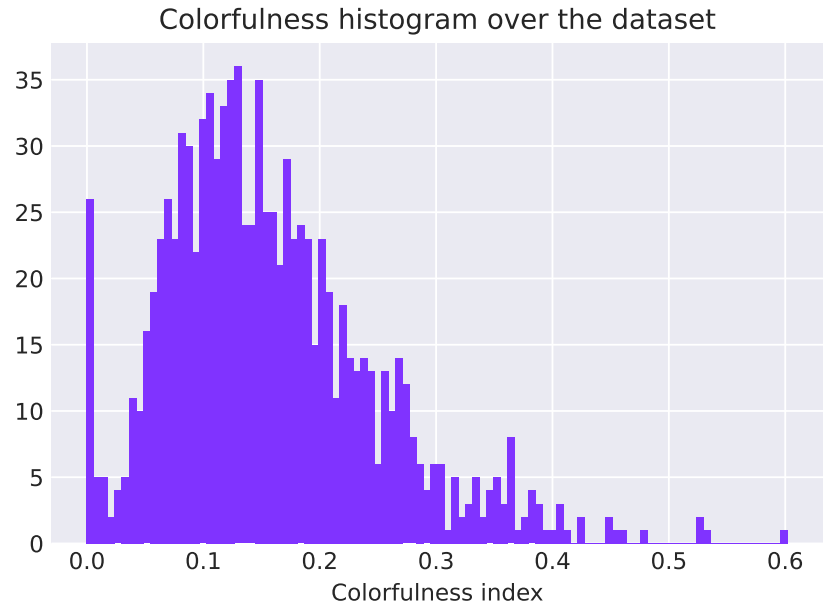


Figure 5.2: Colorfulness index histogram over NIPS2017 dataset.

To analyze the effect of colorfulness on the attack performance, we calculated the colorfulness index histogram of the images in the dataset (Figure 5.2) . We found that 3.2% of the dataset consists of grayscale images, for which our method would not be able to make any changes to the input image, inevitably resulting in a failed attack. Figure 5.3 shows the attack success rate using the subsets where colorfulness is lower-limited by filtering out examples having colorfulness index less than the x axis value. Although a*b* attacks are slightly more successful than C_bC_r in the low colorfulness regime $(<= 0.2)$, they have the same success rate of the attacks over higher colorfulness.

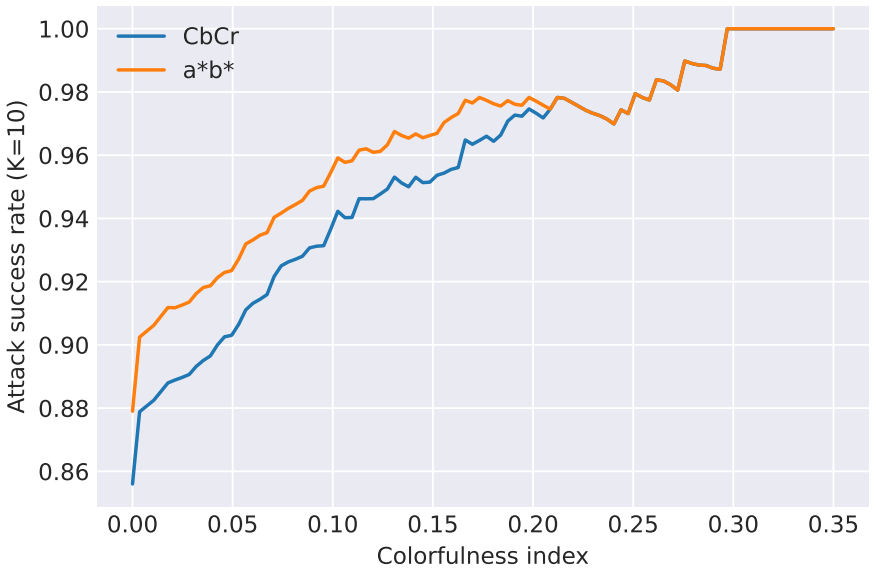


Figure 5.3: Attack success rate analysis with regards to colorfulness index with $\kappa = 10$ on $CbCr$ and a*b* channels. Images having colorfulness index less than the x axis value are excluded in calculation of the success rate. Note that both colorspaces attain very close success rates after around colorfulness index 0.2.

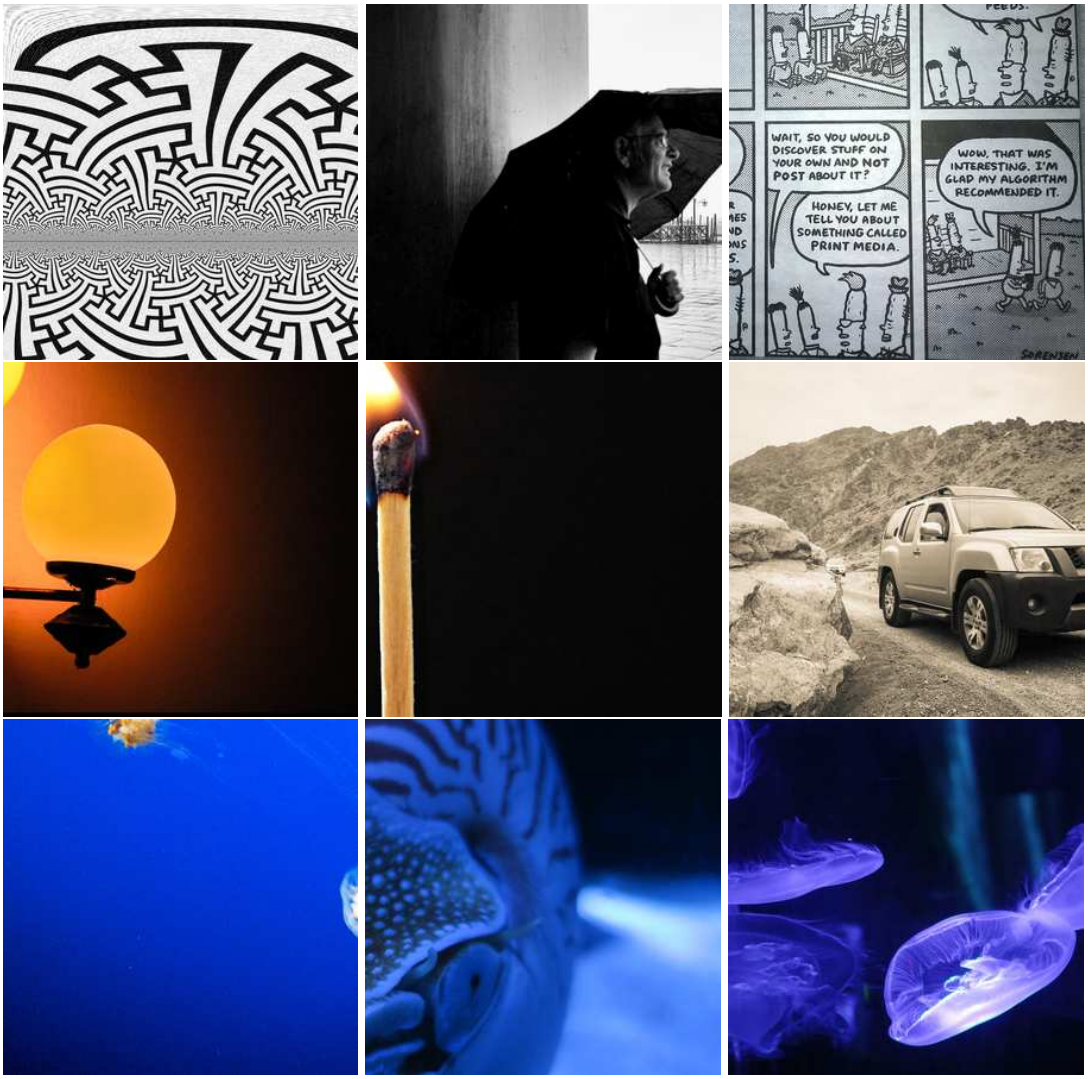


Figure 5.4: Examples from the dataset that our method fails to generate successful adversarial examples from in both YC_bC_r and CIELAB spaces, sorted from top bottom by colorfulness amount.

Figure 5.5: Examples of visible clipping artifacts of out-of-gamut pixels caused by spatial transform around red-gray borders. Flow magnitude has been scaled up to highlight the visible effects for illustration.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

Adopting the techniques used in multimedia compression and using the idea that pixel shifts in a constrained neighborhood are hard to notice, we designed a method that applies local spatial transformations to chrominance channels of perceptual colorspaces. The proposed method results in adversarial images having imperceptible distortions without requiring any regularization term for visual or perceptual quality. In addition to obtaining competitive fooling rates, restricting magnitude of the spatial transformations still yields successful attacks, when there is sufficient amount of local chrominance variation in the input image. As a limitation, this method may produce clipping artifacts which is visible by human observers when the spatial flow application produces out-of-gamut color values, which is also seen in applications of chroma subsampling for lossy visual media compression.

6.2 Future Work

In addition to the perceptual colorspaces investigated in this work, other perceptual colorspaces such as CIELUV, HSLuv and CIEXYZ [24] can also be utilized to create imperceptible adversarial examples. Out of gamut values at borders with red pixels may result in visible artifacts during the adversarial image generation and preventing such out-of-gamut values would result in better quality adversarial images. To accomplish this, sophisticated projection methods of out-of-gamut pixels towards the boundaries of possible colors can be utilized instead of naively clipping the pixels into the viable range, which produces visible artifacts since clipping pixel values cause changes in luminance component of pixel values. While our method does not require optimizing using a visual quality metric, it can be utilized along with our method to obtain a better visual quality. As having imperceptible adversarial examples has implications in security and privacy in Artificial Intelligence (AI), data poisoning attacks using imperceptible adversarial examples is a promising direction for AI security and privacy research [10, 49]. Since adversarial robustness research is gaining interest in computer vision research, comparison of our method with the methods of generating imperceptible adversarial examples in current literature against adversarially trained networks remains an open research topic.

Recently, there is a trend in computer vision research towards self-attention based

transformer architectures [50] for visual classification or detection tasks [51]. However, their adversarial robustness and behavior on adversarial settings are not yet well understood. Since we only have investigated Convolutional Neural Networks(CNNs), exploring the properties of self attention based vision architectures against our method is on open research area.

REFERENCES

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *ComputerScience*, 2015.

[2] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

[3] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

[5] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

[6] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018.

[7] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[8] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," *arXiv preprint arXiv:1312.6199*, 2013.

[9] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[10] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 ieee symposium on security and privacy (sp)*, pp. 39–57, Ieee, 2017.

[11] M. Jordan, N. Manoj, S. Goel, and A. G. Dimakis, "Quantifying perceptual distortion of adversarial examples," *arXiv preprint arXiv:1902.08265*, 2019.

[12] L. Engstrom, B. Tran, D. Tsipras, L. Schmidt, and A. Madry, "A rotation and a translation suffice: Fooling cnns with simple transformations," 2018.

[13] M. Vorobyev, "Ecology and evolution of primate colour vision," *Clinical and Experimental Optometry*, vol. 87, no. 4-5, pp. 230–238, 2004.

[14] A. Aydin, D. Sen, B. T. Karli, O. Hanoglu, and A. Temizel, *Imperceptible Adversarial Examples by Spatial Chroma-Shift*, p. 8–14. New York, NY, USA: Association for Computing Machinery, 2021.

[15] F. Tramèr, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "The space of transferable adversarial examples," *arXiv preprint arXiv:1704.03453*, 2017.

[16] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *arXiv preprint arXiv:1706.06083*, 2017.

[17] A. Shafahi, M. Najibi, M. A. Ghiasi, Z. Xu, J. Dickerson, C. Studer, L. S. Davis, G. Taylor, and T. Goldstein, "Adversarial training for free!," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[18] A. Shafahi, W. R. Huang, C. Studer, S. Feizi, and T. Goldstein, "Are adversarial examples inevitable?," *arXiv preprint arXiv:1809.02104*, 2018.

[19] F. Tramèr, A. Kurakin, N. Papernot, I. Goodfellow, D. Boneh, and P. McDaniel, "Ensemble adversarial training: Attacks and defenses," *arXiv preprint arXiv:1705.07204*, 2017.

[20] A. Ross and F. Doshi-Velez, "Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018.

[21] D. Sen, B. T. Karli, and A. Temizel, "Training universal adversarial perturbations with alternating loss functions," in *The AAAI-22 Workshop on Adversarial Machine Learning and Beyond*, 2021.

[22] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Artificial intelligence safety and security*, pp. 99–112, Chapman and Hall/CRC, 2018.

[23] E. Hamilton, "Jpeg file interchange format," 2004.

[24] J. Schanda, *Colorimetry: understanding the CIE system*. John Wiley & Sons, 2007.

[25] M. Mahy, B. Van Mellaert, L. Van Eycken, and A. Oosterlinck, "The influence of uniform color spaces on color image processing: A comparative study of cielab, cieluv, and atd," *Journal of Imaging Technology*, vol. 17, pp. 232—-243, 1991.

[26] E. Riba, D. Mishkin, D. Ponsa, E. Rublee, and G. Bradski, "Kornia: an open source differentiable computer vision library for pytorch," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3674–3683, 2020.

[27] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, "Comparison of full-reference image quality models for optimization of image processing systems," *International Journal of Computer Vision*, vol. 129, no. 4, pp. 1258–1281, 2021.

[28] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

[29] B. Aksoy and A. Temizel, "Attack type agnostic perceptual enhancement of adversarial images," in *International Workshop on Adversarial Machine Learning And Security (AMLAS), IEEE World Congress on Computational Intelligence (IEEE WCCI)*, 2019.

[30] C. Pestana, N. Akhtar, W. Liu, D. Glance, and A. Mian, "Adversarial perturbations prevail in the Y-Channel of the YCbCr color space," Feb. 2020.

[31] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[32] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2, pp. 1398–1402, Ieee, 2003.

[33] Z. Zhao, Z. Liu, and M. Larson, "Towards large yet imperceptible adversarial image perturbations with perceptual color distance," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[34] M. R. Luo, G. Cui, and B. Rigg, "The development of the cie 2000 colour-difference formula: Ciede2000," *Color Research & Application*, vol. 26, no. 5, pp. 340–350, 2001.

[35] C. Laidlaw and S. Feizi, "Functional adversarial attacks," *Advances in Neural Information Processing Systems*, vol. 32, pp. 10408–10418, 2019.

[36] F. Croce and M. Hein, "Sparse and imperceivable adversarial attacks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4724–4732, 2019.

[37] C. Xiao, J.-Y. Zhu, B. Li, W. He, M. Liu, and D. Song, "Spatially transformed adversarial examples," *arXiv preprint arXiv:1801.02612*, 2018.

[38] M. Jaderberg, K. Simonyan, A. Zisserman, *et al.*, "Spatial transformer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[39] V. V. Estrela, H. A. Magalhães, and O. Saotome, "Total variation applications in computer vision," in *Handbook of Research on Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processing*, pp. 41–64, IGI Global, 2016.

[40] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1, pp. 503–528, 1989.

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[42] A. Mordvintsev, N. Pezzotti, L. Schubert, and C. Olah, "Differentiable image parameterizations," *Distill*, vol. 3, no. 7, p. e12, 2018.

[43] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*, pp. 8024–8035, Curran Associates, Inc., 2019.

[44] A. Kurakin, I. Goodfellow, S. Bengio, Y. Dong, F. Liao, M. Liang, T. Pang, J. Zhu, X. Hu, C. Xie, *et al.*, "Adversarial attacks and defences competition," in *The NIPS'17 Competition: Building Intelligent Systems*, pp. 195–231, Springer, 2018.

[45] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255, Ieee, 2009.

[46] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818–2826, 2016.

[47] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness," *arXiv preprint arXiv:1811.12231*, 2018.

[48] G. Chan, "Toward better chroma subsampling," *SMPTE motion imaging journal*, vol. 117, no. 4, pp. 39–45, 2008.

[49] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Adversarial examples are not bugs, they are features," *Advances in neural information processing systems*, vol. 32, 2019.

[50] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[51] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.