Masters Thesis

Stealthy Targeted Adversarial Patch Attacks through Perceptibility-Aware Optimization

Submitted in fulfillment of the degree requirements of the

MSc in Informatik at Saarland University

Author:
Subrat Kishore Dutta
Matriculation: 7028082

Supervisors:
Prof. Dr. Mario Fritz
Dr. Xiao Zhang

14 February, 2024

Erklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Saarbrücken, January, 2024

_____

Subrat Kishore Dutta

Acknowledgments

This thesis is a humble stride in the pursuit of knowledge. My masters studies, have been shaped by profound challenges and deeply gratifying endeavors. I am extremely thankful to the help I have received from many individuals throughout this journey. I wish to convey my heartfelt gratitude to all of them who have influenced me through their unwavering support and invaluable guidance.

At the very start, I would like to express my deepest gratitude and appreciation to my supervisor, Dr. Xiao Zhang. Dr. Zhang has been instrumental in shaping my aspirations to pursue a career in research in adversarial machine learning. His constant presence and patience throughout, have inspired me in some of the most uncertain periods of this work. I have greatly cherished the ease with which I could engage in detailed discussions, whenever needed throughout this experience. His contributions, from the initial ideation to the final drafting of this thesis, have been pivotal in bringing this research to its conclusion. I would also like to express my sincere appreciation to Prof. Dr. Mario Fritz for his valuable feedback throughout my thesis. His insights have been incredibly helpful, drawing from his vast expertise and knowledge. Despite his busy schedule, his oversight of my project has been instrumental in shaping the direction of my research, and I am truly grateful for his guidance.

I am deeply appreciative of Universität des Saarlandes for maintaining a high standard of education and providing exposure to renowned researchers in the field of computer science. It has been a privilege to be part of such an esteemed institution. Additionally, I would like to express my sincere gratitude to CISPA for cultivating a collaborative and inclusive research environment, which has been instrumental in the successful completion of my thesis. Working within this supportive community has been both enriching and rewarding.

Finally, on a personal level, I owe a special thanks to my dear girlfriend Dr. Ruchita Somani. The past three years have been a blend of emotional and mental health struggles, along with many happy moments. Ruchita with her unconditional love, patience and support made sure that I feel accompanied during this entire time. I am thankful to my parents Dr. Dandeswar Dutta and Janu Lahon and my sister Supriya, for their endless love, support and reassurance throughout my academic journey. Their keen interest in my research has been a source of validation and motivation as I pursued my academic career.

Abstract

Adversarial patch attacks, where the adversary is only allowed to modify a small localized area of the input image, have recently attracted a lot of attention due to their attack efficacy in a constrained local environment. However, due to this area constraint, existing methods either are not successful in producing visually imperceptible patches or cannot achieve satisfactory performance under targeted attack scenarios. We argue that current attack methods are not optimized for human imperceptibility as a result cannot bypass state-of-the-art patch defense techniques. To bridge this gap, we propose a novel adversarial patch attack based on perceptibility-aware optimization schemes, achieving a strong targeted attack performance while maintaining the invisibility of the attached patch. In particular, our method first searches for a proper location for patch placement by leveraging class localization and sensitivity maps, balancing the susceptibility of the patch location to both victim model prediction and human perception, then employs a perceptibility-regularized adversarial loss and a gradient update rule that prioritizes color constancy to optimize the perturbations. Extensive experiments on image benchmarks and across architectures demonstrate that our method consistently achieves competitive attack success rates compared to existing methods but with a significantly improved level of imperceptibility. Besides being completely invisible to human observers, our attack is also stealthy enough to render several state-of-the-art patch defenses ineffective.

Contents

List of Tables

List of Figures

Chapter 1
Introduction

Artificial intelligence has seen remarkable advancements in recent years, with the development of deep neural networks followed by transformers serving as a pivotal breakthrough(57). Deep learning is a rapidly evolving field that utilizes deep neural networks to model and solve complex problems. It enables computers to learn from large amounts of data by automatically discovering intricate patterns and representations across multiple layers of abstraction. Contrary to traditional machine learning methods, which often require manual feature engineering, deep learning architectures such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs) can autonomously learn hierarchical features from raw data where at each layer the architecture extract more finer features than the ones encountered in the previous layer(66). This capabilities has led to breakthroughs in areas such as computer vision, natural language processing, and reinforcement learning.

From a theoretical standpoint, deep learning seeks to approximate the underlying true data distribution, which is inherently represented by the training dataset Consequently, the performance and generalization capabilities of deep learning models are heavily influenced by the quality and quantity of the available training data. A sufficiently large and diverse dataset that accurately captures the characteristics of the true distribution enables these models to learn more expressive representations and achieve high predictive accuracy. However, in practice, constructing datasets that precisely reflect the underlying data distribution is a challenging task. As a result, the ability of these models to generalize effectively to unseen data remains a subject of ongoing scrutiny. Furthermore, due to the inherent black-box nature of deep learning models, concerns regarding their interpretability, robustness, and vulnerability to adversarial attacks have been widely recognized. These challenges underscore the necessity for continued research efforts aimed at enhancing the security, transparency, and reliability of deep learning systems in real-world applications.

Deep neural networks are notoriously susceptible to adversarial examples as shown in Figure 1.1, which are inputs crafted with small, carefully designed perturbations that intentionally deceive the model into making incorrect predictions (51). Most existing works considers ℓ_p-norm bounded perturbations, where any pixel of the entire input image can be modified by a small amount described by the perturbation budget and

proposed different attacks to generate such perturbations (6; 14; 25; 36; 37). Imposing a ℓ_p-norm constraint as the perturbation budget, it restricts the perturbation size and ensures the generated perturbations remain visually invisible to humans. The attempt in this line of attacks is made such that the resultant adversarial sample closely replicates its original benign form thus achieving imperceptibility. These attacks have been highly successful in the digital space and have been the corner stone for a wide range of machine learning robustness research specifically through adversarial training(36; 56). Despite the high attack efficacy of these global attacks, their transfer to the physical world or real world scenarios is highly challenging. The perturbation are curated assuming a highly constrained environment which is extremely challenging to mimic precisely in the real world as a result of variations in perspective, imaging noise, and other inherent transformations in natural settings (1).

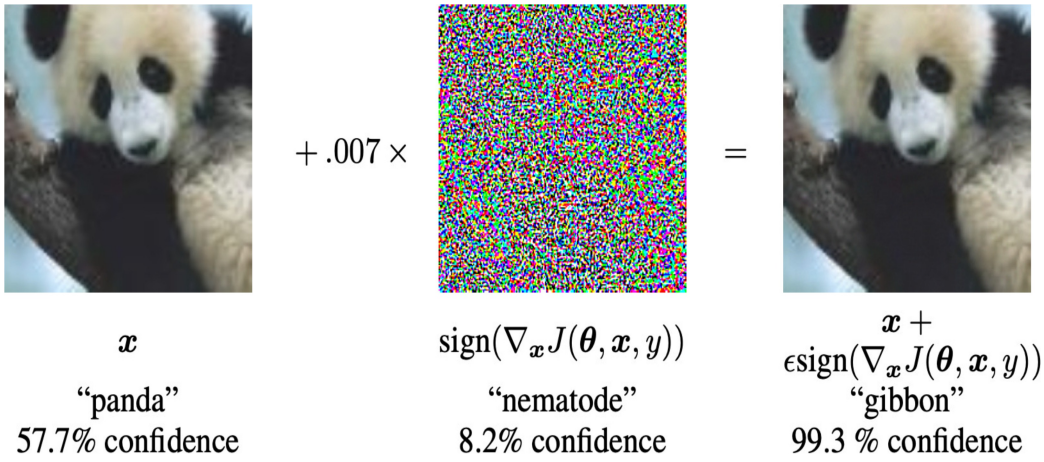

Figure 1.1: An adversarial example represented in Goodfellow at al. (14). A clean image is subtly perturbed to create an adversarial image that is misclassified by the model, while remaining imperceptible to the human eye.

A different but related line of research focused on adversarial patch attacks (4; 22), where the adversary is only allowed to modify a localized region of the input image but without constraints on how much each pixel can be perturbed. These attacks stand in contrast to global ℓ_p-norm bounded perturbations, which impose constraints on the overall perturbation size across the entire image. Adversarial patch attacks are motivated by real-world scenarios where inputs to machine learning models are typically processed automatically, without manual inspection or validation. Furthermore, defenses developed before the formalization of adversarial patch attacks primarily relied on adversarial training with bounded perturbation budgets. Consequently, models trained under such frameworks were not equipped to counter large, localized perturbations introduced by adversarial patches, thereby exposing a significant vulnerability from a defensive standpoint.

Addressing these considerations, adversarial patches developed through these methods are often extremely visually salient, characterized by highly textured and colorful appearances. Notably, (22) emphasized that in targeted attack scenarios, the generated

patches frequently exhibit symbolic representations of the target class chosen by the adversary, as the visual features of the patch align closely with those of the intended target class. This salient nature also enhanced the attacking efficiency that lead to easier transfer into physical-world attacks, posing more threats to real-world machine learning systems deployed for security-critical applications.

Despite being more practical, these adversarial patches can easily raise suspicion from the perspective of human perception, as they are highly salient in appearance and cannot maintain homogeneity with the host image. To make the attacks inconspicuous to humans, multiple studies have developed methods to generate adversarial patches that look like realistic images such that placing them in the host image does not raise any suspicion. Maintaining context homogeneity along with high realism is central to this approach (12; 47; 59). While many attempts have been made to conceal the purpose, the presence of the adversarial patch still remains visibly discernible in these attack strategies. Despite their prowess in physical attack capabilities, it is frequently observed that the perturbations necessary to sustain high attack efficacy, particularly in targeted settings, are often large, rendering the patch conspicuous.

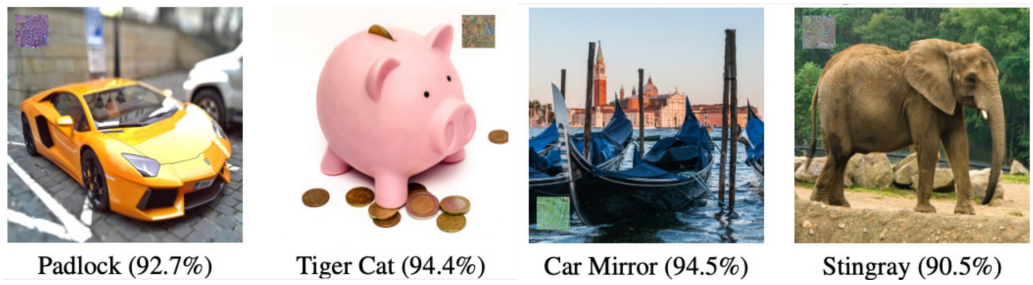

Figure 1.2: An example of an adversarial patch represented in Karmon at al. (22). An adversarial patch is applied to the clean image, causing the model to misclassify it with high confidence.

Defenses developed to counter adversarial patch attacks typically leverage the high visual saliency and contiguous nature of the patch region, achieving notable success in detecting such adversarial manipulations (7; 13; 21; 31; 53; 65). Consequently, from an adversary's perspective, the primary objective is to develop an optimization method aimed at reducing the saliency of the generated patch to evade detection. A line of research focused on developing such methods to generate adversarial patches that are entirely invisible to humans (2; 28; 41). However, these works either focus solely on untargeted settings or suffer from a significant attack performance drop when considering the more challenging targeted scenarios. Although high imperceptibility can be achieved for untargeted attacks, performing targeted attacks is notably more difficult, leading to limited studies in the field. Admittedly, achieving strong attack capabilities while being imperceptible is challenging for adversarial patch attacks due to the limited patch size and the need to avoid saliency, which is often crucial for achieving high attack success rates. This raises a natural question of whether targeted goals are at all achievable with visually imperceptible adversarial patches?

Furthermore, these defense mechanisms generally employ preprocessing operations

applied to incoming samples to mitigate the effects of adversarial patches. However, a significant limitation of most of these methods lies in their reliance on complex, multi-stage processes that often include at least one computationally intensive phase, frequently involving an autoencoder. While this approach can enhance effectiveness, it renders these defenses less practical for real-world applications, especially in scenarios requiring low-latency or resource-constrained environments. Consequently, their utility is largely confined to digital spaces, where computational resources are less restrictive and latency requirements are less stringent. In addition to that, in applications like face recognition, systems often processes digital format of input instead of real time detection in order to identify individuals. This makes the exploration of the digital landscape important specifically in terms of adversarial patch attacks where imperceptibility and inconspicuous nature of the patch generated is of paramount importance.

1.1 Motivation, Research Objectives and Problem Setup

Adversarial patch attacks present a significant threat to machine learning models, particularly in security-sensitive applications, but their high visual saliency makes them vulnerable to detection. While existing imperceptible techniques attempt to reduce patch visibility, they often come at the cost of reduced attack effectiveness, especially in targeted scenarios. The key challenge lies in balancing strong attack performance with imperceptibility, as minimizing saliency can weaken the adversarial impact. Current defense mechanisms capitalize on the patch's highly salient visual attributes, relying on complex, multi-stage processes that, while effective, are resource-intensive and impractical for real-time applications—though still viable in digital environments. This underscores the need for a general optimized adversarial patch generation method that enhances both attack success and imperceptibility to human vision, and by extension, to defense methods that rely on similar principles, with the potential to transition into real-world applications.

In this work, we intend to answer the aforementioned questions affirmatively by aiming to develop a general perceptibility-aware framework for the generation of adversarial patches that are imperceptible to human vision while achieving high targeted attack success rates in the digital domain. The proposed framework is designed such that the underlying principles can be seamlessly adapted to the real-world applications with minimal modifications if necessary.

Imperceptibility in adversarial attacks has traditionally been approached through bounded perturbations, typically enforced using the ℓ_p-norm constraint. While this approach has demonstrated partial success, it is often insufficient in the context of localized attacks, such as adversarial patch attacks, where achieving targeted success frequently necessitates a higher magnitude of perturbation. This increased perturbation typically results in highly textured patches that are easily detectable by human observers.

We hypothesize that by aligning the perturbation update process with human perceptual mechanisms—paying particular attention to factors influencing human visual perception—it is possible to generate adversarial patches that remain inconspicuous even when high levels of perturbation are applied. Furthermore, given that many existing defense mechanisms are designed based on human visual perception, adversarial patches optimized using our proposed perceptually guided approach have the potential to evade such defenses effectively.

The proposed framework leverages advances in attack methods along with insights from

human perception to refine the adversarial optimization process, thereby ensuring that the generated patches exhibit both high attack efficacy and imperceptibility. This work contributes to the advancement of adversarial attack methodologies by addressing the trade-off between perturbation magnitude and perceptual stealthiness, with potential applications in both digital and physical domains. The goal of this work is to develop a novel perceptually-aware optimization framework for generating adversarial patches that are both effective and imperceptible.

1.2 Key Research Contributions

We propose a novel method based on a series of perceptibility-aware optimization schemes, realizing the targeted attacker's goals with imperceptible adversarial patches (see Figure 4.1 for the overall pipeline). Most prior work did not consider the sensitivity of the human visual system with respect to patch placement and perturbation generation, which we argue plays a vital role in contributing to the success of our design. Specifically, our method first locates a patch region in the host image that optimally balances the class localization and sensitivity scores, enabling a strategic advantage in both attack capabilities and imperceptibility (Section 4.2). After the localization step, our method iteratively updates the patch by restricting the changes in base color and regularizing the adversarial loss using a human perception-based distance, which reduces the saliency of the resulting patch and further promotes imperceptibility (Section 4.3). By conducting extensive experiments on benchmark datasets for image classification and recognition tasks, we demonstrate that compared with state-of-the-art patch attacks, our method achieves comparable or even higher attack success rates but with a significantly improved degree of invisibility across various targeted attack scenarios (Sections 7). Moreover, we show that the adversarial patches generated by our method can successfully bypass various patch defenses, confirming the stealthiness of our attack (Section 7.3). The high stealthiness and strong capabilities of the adversarial patches generated by our method call for an urgent need to develop better defense strategies to detect and mitigate such stealthy attacks.

1.3 Outline

Our work is structured into seven chapters to give a comprehensive analysis of adversarial patch attacks with specific focus on conducting imperceptible patch attacks. In Chapter 2, we delve into the existing literature on adversarial patch attacks, examining their strengths and weaknesses. We then explore techniques for improving the imperceptibility of these attacks, drawing insights from both the adversarial patch and general adversarial example domains. Finally, we review current defense mechanisms specifically designed to counter adversarial patch attacks. Chapter 3 introduces the preliminary concepts and knowledge that forms the foundation of this work and will be referred to as and when required during the due course of this thesis. This chapter provides a foundational overview of deep neural networks, adversarial patch attacks, and the challenges they pose. It delves into defense mechanisms against these attacks and explores the critical concept of imperceptibility, along with the metrics used to evaluate it. In chapter 4 we describe our methodology where we give details on implementation and strategic design considerations of the perceptibility-aware perturbation optimization method that we propose. Chapter 5 establishes the different experimental

setups that are used to evaluate the proposed method's efficacy both in terms of attack ability and imperceptibility. This is followed by chapter 6 which shows the initial results obtained on Stanford Dogs Dataset as a proof of concept. Chapter 7 gives detailed and comprehensive evaluation of the method on ImageNet including method performance and its comparison to other state-of-the-art methods. This is followed by performance against existing defense methods, concluding with analysis on transferability aspects into the real world. In chapter 8 we delve into a more realistic case of face recognition and evaluated our method on VGG Face dataset. We showcase factors affecting our attack method through ablation studies presented in chapter 9. Chapter 10 presents the findings of this research and discusses their implications. It also offers key insights into the methodology, its implication from both an attack and a defense perspective and limitations suggesting potential avenues for future research. We Conclude with Chapter 11 where we highlight the central contributions and takeaway of this work in to the world of adversarial machine learning.

Chapter 2
Related Work

In this section, we attempt to provide a bridge between the existing works of adversarial patch attacks and the perspective of imperceptibility in adversarial attacks and how the latter aids the former. We provide an overview of recent advancements in adversarial defense methods and offer a detailed account of the current state of research in this area.

2.1 Adversarial Patch Attacks

The foundational ideas about adversarial patch attacks conducted in the form of printable sticker was initial done by Brown et al. (4) where they introduces a method for creating universal, robust, and targeted adversarial patches capable of attacking image classifiers in real-world scenarios. These patches are ideally expected to be universal, that is working across different scenes; robust, remaining effective under various transformations; and targeted, forcing classifiers to predict a specific target class. The central idea that lead to the popularity of this form of attack is its promise of print-ability which would allow for the attack to be transferred to the physical world. the study proved their efficacy by misleading classifiers into predicting the chosen target class, with successes even in the physical world. This work was closely followed by Karmon et al. (22), who followed a similar methodology specifically targeting the digital space. They studied both universal as well as instance specific attack methods where patch for each input instance is created separately. Designing specific experiments they were also able to successfully challenge the initial claim by (4) that: adversarial patches are effective because its saliency which captures the entirety of the network's attention. They further highlighted that often types the patches generated closely resembles miniature and symbolic representations of the target class. Hayes (15) introduced a defense method against the existing adversarial method while describing an innovative idea of spreading the formed adversarial patch to sparsify the formed patch to evade the proposed defense method.

Apart from targeting classification task numerous studies also have targeted object detectors with similar methodology which involves visually susceptible adversarial patches. Liu et al. (32) introduced DPATCH that is highly effective, reducing detection performance quantified by mAP scores of Faster R-CNN and YOLO from 75.10% and 65.7%

respectively to below 1%. Unlike adversarial patch, DPATCH is optimized to reduce performance for both object classification as well as regression task for the bounding boxes. It offers several advantages, including the ability to perform untargeted and targeted attacks, location-independent effectiveness, and strong transferability across different detectors and datasets. DPAttack by Wu et al. (63) introduces a novel approach using diffused patches, such as asteroid-shaped or grid-shaped patterns, that alter only a small number of pixels while effectively fooling object detectors. Huang et al (19) focused on reducing the perceptibility concerns associated with patch attacks concerning object detectors by highlighting that perceptible perturbation as unnecessary for effectiveness of the attack. Through RPAttack, it introduces a novel approach that creates minimal yet highly efficient perturbations. It employs a patch selection and refining scheme to identify the most critical pixels for the attack while gradually eliminating inconsequential perturbations. Additionally, it balances the gradients of different detectors to ensure stable ensemble attacks.

2.2 Imperceptibility in Adversarial Patch Attacks

We classify imperceptibility in adversarial patch attacks into two categories: Context Homogeneity and Host Constancy. Context Homogeneity refers to approaches that leverage inherent human knowledge about the environment in which the attack is executed. Here, the attacker's objective is to maintain contextual homogeneity between the applied patch and its surrounding environment, ensuring the patch remains inconspicuous to observers. Here, the visibility of the perturbation itself is not the primary concern. In contrast, Host Constancy focuses on achieving complete invisibility with respect to the host image by designing perturbations on the adversarial patch that seamlessly integrate with the original image, making the patch appear as a natural part of the original content.

2.2.1 Attacks Prioritizing Context Homogeneity

Although patch visibility is not a matter of central importance in these initial works, the latter works aim to make the generated adversarial patches inconspicuous such that they raise minimized suspicion to human observers. For instance, Sharif et al. (46) proposed to incorporate eyeglasses featuring a unique texture to launch an attack on face recognition systems. Eykholt et al. (12) proposed to conceal black and white adversarial patches on traffic signs to attack traffic sign classifiers and more broadly autonomous driving systems. Liu et al. (30) leveraged PS-GAN, a specific type of generative adversarial network (GAN), to produce adversarial stickers with high visual fidelity. The study also emphasized the importance of context homogeneity, ensuring that the applied patch seamlessly blends with the benign image, rendering the perturbation inconspicuous to observers. Observing the highly perturbed appearance of the formed patches, Wang et al. (59) proposed VRAP, a novel adversarial patch generation algorithm that produces visually realistic patches capable of fooling deep neural networks in both digital and physical environments. Although the aforementioned methods can lower suspicion to certain extents, the adversarial patch usually remains visibly discernible when placed in the host image or relies on prominent semantic features of another object class to fool the victim model. Zolfi et al. (69) designed adversarial patch attacks for detection tasks by leveraging the idea of patch blending to make the patterns as unnoticeable as possible.

2.2.2 Attacks Prioritizing Host Constancy

Another line of research focused on rendering the patch entirely invisible, exemplified by the work of Bai et al. (2). In particular, Bai et al. (2) proposed to use multiple Generative Adversarial Networks (GANs) to generate adversarial patches at different scales, resulting in patches that closely resemble the original image. However, the computational complexity of training multiple GANs for each image is impractical. Qian et al. (41) exploited the model's perceptual sensitivity to determine the location of perturbations, but this method does not confine the attack region, potentially leading to widespread perturbations, which contradicts the conventional goal of patch attacks. Both these works considered the scenario of untargeted attacks but refrained from extending their work to targeted attacks. In addition, GDPA by Li et al. (28) utilized a generator to create both dynamic and static patch patterns, determining their locations within the input image and leveraging the idea of using a soft mask to place the patches such that the invisibility of the patches can be enhanced. Although these works attempted to encompass the targeted attack settings, the trade-off between visibility and attack efficacy is heavily skewed, as reduced visibility stemming from reduced perturbation led to a significant drop in performance. This seems to suggest an inherent requirement of large perturbation to conduct targeted patch attacks.

2.3 Imperceptibility in Adversarial Examples

In a broader context of adversarial attacks, imperceptibility refers to ensuring the adversarial example closely resembles the original image to the extent that it remains undetectable by humans. The human visual system is relatively insensitive to changes in color values in regions of higher variation. Out of these high-variance locations the high-textured regions receive even lesser attention from human perception relative to the perturbations done on the edges of the image, because of the prior knowledge about the structures of edges (29). Following this notion, Luo et al. (35) realized that perturbations made in locations of high variance are less visible than perturbations made in areas with low variance. Croce et al. (8) further added that perturbations on the horizontal and vertical edges are more noticeable, thereby restricting perturbations from those locations could lead to better imperceptibility. Changing only the saturation and brightness of a pixel can be used to mimic different quantized levels of the same base color. Quantization errors can be hidden effectively in high-textured locations, thus perturbations that only alter the saturation and brightness of a pixel and not the base color can achieve a better camouflaging effect (8; 11; 29). Moreover, prior work has been done on restricting the added noise by incorporating a ϵ-budget on the perturbation or using its ℓ_p norm as a regularization term on the final objective function (12; 14; 36; 37; 51). While having an ϵ budget enhances imperceptibility, in the context of adversarial patches where the space for the attack is restricted, conducting targeted attacks with limited perturbation is highly non-trivial. In addition, despite ℓ_p norm-based approaches working well in limiting the perturbation values, they are agnostic to human perception and treat every pixel equally. Luo et al. (35) proposed a distance metric taking into account human perception which we hypothesize can be a better regularization term for imperceptibility. In this work, we argue that the human perception-oriented strategies can facilitate storing large perturbation values which we hypothesize are instrumental in conducting imperceptible targeted adversarial patch attacks. We thus propose our novel pipeline to generate adversarial patches that are highly effective in attacking image classifiers while being

convincingly stealthy.

2.4 Defense against Adversarial Patch Attacks

The growing body of research on defense against adversarial patch attacks has predominantly focused on the saliency of such attacks, leveraging the distinct visual and statistical characteristics of adversarial patches to develop defense mechanisms. These current attack methods generate localized perturbations that exhibit unique properties compared to benign image regions, such as higher entropy, distinctive and highly texture patterns, and distributional discrepancies. Consequently, many defense strategies capitalize on these attributes for patch detection, segmentation, and removal. A variety of approaches have been proposed which are often multi step processes, including entropy-based methods that identify anomalous high-variance regions (53), saliency maps constructed through guided backpropagation methods followed by simple image preprocessing operations (15), autoencoder based architecture for segmentation and detection (31; 65), generative models that reconstruct perturbed areas (7), and diffusion-based frameworks that leverage learned priors to restore original content(13; 21).

Among the diversity in methodologies, a common underlying assumption across existing defenses is that adversarial patches possess visually or statistically detectable signatures that can be effectively exploited to mitigate their impact. These defense methods, grounded in the assumption of adversarial patch saliency, predominantly focus on features that are readily identifiable by the human visual system. They are closely aligned with human perceptual mechanisms for detecting anomalies, such as texture inconsistencies and unnatural patterns. Consequently, although the initial consideration to conduct adversarial attacks eliminates human manual inspection, these defense methods inadvertently encode similar perceptual heuristics. They leverage characteristics such as unnatural textures and irregular patterns, which are inherently conspicuous to the human visual system, to identify and mitigate adversarial patches. In contrast, some methods have also approached patch detection through adversarial purification, aiming to refine or cleanse the inputs to reduce the effectiveness of adversarial manipulations without relying solely on perceptible cues.

2.4.1 Defense via High-Saliency Region Detection

Hayes (15) approached adversarial patch defense as an inpainting problem, addressing it at two levels: non-blind and blind inpainting. In the non-blind setting, where the perturbation's location is known, the corrupted region is reconstructed using inpainting methods like the Telea algorithm (54). In the blind setting, where the location is unknown, a saliency map generated via guided backpropagation highlights high-influence regions, which are further refined using morphological operations to isolate adversarial patches. Jujutsu by Chen et al. (7) is a two-stage defense framework designed to detect and mitigate adversarial patch attacks on deep neural networks (DNNs). The first stage, similar to (15), focuses on attack detection by leveraging saliency maps to identify suspicious localized features that exert a dominant influence on the model's predictions. The study acknowledges that salient regions can also correspond to non adversarial features, hence to enhance detection accuracy and reduce false positives, Jujutsu applies a pre-processing step to the salient map generated. This is followed by guided transplantation of the suspected adversarial features to hold-out input to further validate the presence of an attack

through its success on the augmented input. Once the perturbed location is determined generative adversarial networks (GANs) are used to reconstruct the corrupted regions within the image, effectively restoring clean content and enabling correct classification by the DNN. Tarchoun et al. (53) along the same line approached patch detection through an information-theoretic approach. The method introduces two key techniques: entropy analysis and autoencoder-based patch completion. First, Jedi utilizes entropy analysis to identify potential adversarial patch regions which is then refined by an autoencoder that enhances the localization accuracy along with completion of the proposed patch. This is followed by inpainting.

Liu et al. (31) proposed the Segment and Complete (SAC) defense framework to protect object detectors against adversarial patch attacks. The framework employs a U-Net-based patch segmenter, trained with pixel-level annotations to generate patch masks for adversarial localization. To improve robustness of the patch segmenter, a self-adversarial training algorithm is introduced. Finally, a shape completion algorithm guarantees complete patch removal if the predicted mask is within a specified Hamming distance from the ground truth. PatchZero by Xu et al. (65) is a general defense pipeline designed to counter white-box adversarial patch attacks without requiring retraining of downstream classifiers or detectors. The method leverages the general notion discussed, that from the observations the adversarial patches are highly textured and visually distinct from natural images. Moving along the same direction as Liu et al.(31), PatchZero detects adversarial regions at the pixel level by utilizing a patch detector that predicts a pixel-wise adversarial binary mask. The method then mitigates the adversarial effect by masking out the patch region from the perturbed input located by the generated mask and repainting it with mean pixel values. Additionally, it incorporates a two-stage adversarial training scheme to enhance robustness against stronger adaptive attacks.

2.4.2 Defense via Adversarial Purification

Several preprocessing-based defense methods have been proposed for global attacks which aims to retrieve the original benign sample by removing the added perturbation using generative models. Pouya et al. (43) utilized GANs to defend against adversarial perturbation by modeling the distribution that represents the benign samples. With the advent of more capable generative model which have able to achieve state-of-the-art performance in generative tasks (9; 18), recently diffusion models have been utilized for the purification task as well(38; 48; 50; 58; 58; 64). Guided diffusion model for purification (GDMP) by Wang et al. (58) integrates purification into the diffusion-denoising process of Denoised Diffusion Probabilistic Model (DDPM) to mitigate adversarial perturbations. DiffPure (38) incrementally introduces Gaussian noise during the forward diffusion process and subsequently removes it through the reverse generation phase, effectively purifying the adversarial perturbations in the process. Xiao et al. (64) proposed DensePure, which denoises adversarial samples by generating multiple reversed samples through repeated reverse diffusion runs, followed by majority voting using the target model to recover the target class. These methods however, are specifically trained to handle ℓ_p-norm bounded adversarial perturbations and hence are not ideally designed to located adversarial patches and hence their mitigation.

The works by Kang et al. and Fu et al. (13; 21) are some of the approaches that are developed recently which extends the use of diffusion models in defending against adversarial patch attacks. DIFFender introduced by Kang et al. (21), is a novel defense framework that leverages the capabilities of a text-guided diffusion model to counter

adversarial patch attacks. At its core is the discovery of the Adversarial Anomaly Perception (AAP) phenomenon, which enables the diffusion model to detect and localize adversarial patches by analyzing distributional discrepancies. DIFFender combines patch localization and restoration tasks within a unified diffusion model framework. Additionally, it employs vision-language pre-training and a few-shot prompt-tuning algorithm to adapt the pre-trained diffusion model to defense tasks without requiring extensive retraining. Fu et al. (13) introduced DiffPAD that forms a framework designed to address adversarial patch attacks by leveraging the capabilities of diffusion models for adversarial patch decontamination. The framework begins with super-resolution restoration on suspected downsampled input images, followed by a binarization and dynamic thresholding scheme combined with a sliding window approach to effectively localize adversarial patches. Once the patch region is localized, DiffPAD applies inpainting techniques to restore the original image.

Chapter 3
Preliminaries

3.1 Introduction to Deep Neural Network for classification

Deep Neural Networks (DNNs) have significantly advanced classification tasks across various domains, including image recognition, speech processing, and natural language understanding (27). These models, inspired by biological neural networks, consist of multiple layers that learn hierarchical feature representations. Among the most widely used architectures for classification are Convolutional Neural Networks (CNNs), Transformers, and Ensemble models, each designed to handle specific challenges in pattern recognition and decision-making.

3.1.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are specialized deep learning models that excel in analyzing structured grid-like data, such as images (24). They utilize convolutional layers to extract spatial features by applying learnable filters, thereby reducing the need for manual feature engineering. CNNs achieve translation invariance through pooling layers and maintain computational efficiency while preserving important spatial information. Popular CNN architectures such as AlexNet (24), VGGNet (49), ResNet (16), and EfficientNet (52) have demonstrated remarkable performance in large-scale image classification tasks.

3.1.2 Transformers

Transformers, initially introduced for sequence modeling in natural language processing (57), have been adapted to computer vision tasks, leading to the development of Vision Transformers (ViTs) (10). Unlike CNNs, which rely on local receptive fields, transformers utilize self-attention mechanisms to capture long-range dependencies in data, allowing them to model global relationships between image regions. Recent architectures, such as

Swin Transformer (34), have further optimized this approach by incorporating hierarchical feature extraction and computational efficiency. These models have demonstrated state-of-the-art performance in several benchmark datasets, surpassing traditional CNNs in certain scenarios.

3.1.3 Ensemble models

Ensemble models combine multiple classifiers to enhance overall performance, robustness, and generalization. Techniques such as bagging (3), boosting (44), and stacking (62) leverage the diversity of individual models to reduce variance and improve prediction reliability. In image classification, ensembles of CNNs and transformers have been shown to outperform individual models, particularly in adversarial settings where robustness is critical (56).

3.2 Adversarial Attacks

Adversarial attacks exploit the vulnerabilities of machine learning models by introducing small, often imperceptible perturbations that cause incorrect predictions (51). These attacks pose significant security risks in applications such as biometric authentication, autonomous driving, and medical diagnosis. Depending on the attacker's level of access to the target model, adversarial attacks are classified into different scenarios:

3.2.1 Different Attack Scenarios

Black-Box

In black-box attacks, the attacker has no access to the model's internal parameters, gradients, or architecture. Instead, they rely on querying the model and observing its outputs to craft adversarial examples (39). Common black-box attack techniques include transferability-based attacks (33), query-based methods (20), and surrogate model training (14).

White-Box

White-box attacks assume full knowledge of the target model, including its parameters and gradients. This allows attackers to generate highly optimized adversarial perturbations. Well-known white-box attacks include the Fast Gradient Sign Method (FGSM) (14), Projected Gradient Descent (PGD) (36), and Carlini & Wagner (C&W) attack (6).

3.2.2 Targeted and untargeted setting

Depending on the specificity of adversary's goal in terms of model's prediction we have two scenarios:

Targeted Attacks

The goal is to force the model to misclassify an input as a specific incorrect label (25). This is particularly dangerous in security-sensitive applications, such as facial recognition systems.

Untargeted Attacks

The objective is to cause any form of misclassification without specifying the incorrect output class. These attacks degrade model performance and are generally easier to execute (51).

3.3 Adversarial Patch Attacks

Unlike traditional adversarial attacks, adversarial patch attacks introduce localized, structured perturbations known as patches that mislead classifiers without requiring access to the entire image (4). These attacks are particularly potent in real-world scenarios, as they remain effective under transformations such as resizing, rotation, and occlusion (55). Adversarial patches have been successfully deployed against object detection models, facial recognition systems, and autonomous vehicles.

3.4 Discriminative Task

Discriminative tasks involve classifying input data into predefined categories based on learned patterns. Two critical applications affected by adversarial attacks are image classification and face recognition.

3.4.1 Image classification

Image classification assigns labels to images based on learned features. Deep learning models, particularly CNNs and transformers, have achieved high accuracy in this domain (16). However, adversarial attacks significantly threaten their reliability, as small perturbations can lead to incorrect predictions (51).

3.4.2 Face Recognition

Face recognition systems identify or verify individuals based on facial features. Adversarial attacks on these systems can lead to unauthorized access or identity impersonation (46). Adversarial patches, in particular, have been shown to bypass facial recognition by strategically modifying facial regions (Komkov & Petiushko, 2021).

3.5 ℓ_p-norm bounds for Imperceptibility

Imperceptibility of adversarial perturbations is typically quantified using ℓp-norm constraints. ℓ_0-norm is measures as the number of modified pixels, favoring sparse perturbations (6). ℓ_2-norm is measures as the Euclidean distance between original and adversarial images, ensuring smooth modifications (51). ℓ_∞-norm captures the maximum pixel-wise perturbation, constraining the worst-case distortion(36). These form of norm-constrained optimization ensures the overall perturbation added to the whole of image which although induces some level of imperceptibility but these restrictions do not take into consideration the sensitivity of human visual system. For most part of adversarial machine learning literature, imperceptibility has always been argued as a notion that can be primarily be achieved using such bounds to the perturbation budget.

Chapter 4
Methodology

In this section, we explain the details of our proposed attack pipeline (Figure 4.1). We address the challenge of balancing attack effectiveness and imperceptibility from two complementary perspectives. First, we focus on strategically positioning the adversarial patch in regions of the target image that provide a dual advantage: maximizing the attack success rate while minimizing visual detectability. This involves first identifying the optimal location of patch placement that inherently facilitate the adversarial objective without drawing human attention. To achieve this objective, our method considers both the attack capabilities of the proposed attack method and its imperceptibility to the human visual system (Section 4.2). Second, we aim to develop optimization update strategies that incorporate principles of the human visual system, ensuring that perturbations are applied in a manner that aligns with perceptual characteristics that facilities imperceptibility. To achieve this, the perturbations are optimized by minimizing a regularized targeted adversarial loss using proposed color constant gradient updates (Section 4.3). Every stage of our proposed method is meticulously crafted to optimize the visual imperceptibility of the generated adversarial patch, enabling the accommodation of high-magnitude perturbations that remain inconspicuous to the human eye while effectively achieving the intended adversarial objectives.

4.1 Problem Formulation

We consider targeted, white-box settings for adversarial patch attacks. Assume the attacker has the full knowledge of a victim model f_θ with model parameters θ. Let an RGB image $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^{W \times H \times C}$ be a correctly classified benign sample, $y \in \mathcal{Y}$ be the ground-truth class label of $\boldsymbol{x}$, and y_{targ} be the class label that the attacker aims to target for. The adversarial input $\hat{\boldsymbol{x}}$ is generated by placing an adversarial patch of width w and height h on the benign sample x at a certain localized region indexed by (i, j) such that $f_\theta(\hat{\boldsymbol{x}}) = y_{\text{targ}}$. More rigorously, $\hat{\boldsymbol{x}}$ is defined as:

$$\hat{\boldsymbol{x}} = (1 - \boldsymbol{m}) \odot \boldsymbol{x} + \boldsymbol{m} \odot \boldsymbol{\delta}, \tag{4.1}$$

where $\boldsymbol{m} \in \{0,1\}^{W \times H \times C}$ is a location mask such that $m_{k,l,c} = 1$ if $i \leq k \leq i + w$, $j \leq l \leq j + h$ and $c \in C$, and 0 otherwise, $\boldsymbol{\delta} \in \mathbb{R}^{W \times H \times C}$, and $\odot$ stands for the element-wise multiplication. For the ease of the presentation, we use the following simplified notation to denote an adversarial input and the attached adversarial patch throughout the paper:

$$\hat{\boldsymbol{x}} = \boldsymbol{x} +_{i,j} \boldsymbol{\delta}, \tag{4.2}$$

where $+_{i,j}\boldsymbol{\delta}$ denote placing the patch $\boldsymbol{\delta}$ at location (i, j) of the benign sample.

4.2 Optimization of Patch Placement

The central idea of this optimization step is to place the patch at a location that is highly vulnerable to adversarial perturbations and can host high perturbations such that the targeted goals can be achieved with adversarial patches without being visually salient. From a technical perspective, our objective is to target specific regions within the image where small perturbations can induce significant changes in the model's output. In other words, we aim to identify locations where the gradient of the loss function with respect to the pixel values is high. Previous studies have shown that models contain visually sensitive zones that play a significant role in their predictions (5; 67), making them more susceptible to attacks in these regions (30; 41). These regions offer a strategic advantage by enabling successful adversarial attacks with minimal perturbations. Although, it can be argued that these vulnerable locations might require less perturbation for an attack to be successful—potentially making the adversarial patch less noticeable and hence desirable, these regions are frequently observed to be highly sensitive to human observers as well from our observations. Often times it was observed that these highly attackable sensitive locations coincide with areas that are also highly sensitive to human perception. This consequently limits their capacity to accommodate large perturbations. As a result, even minor perturbations in these regions can produce noticeable visual artifacts, making them more susceptible to detection by the human visual system.

Therefore, we aim to integrate the sensitivity of the human visual system into the optimization process to achieve a balanced trade-off between attack efficacy and imperceptibility. By incorporating perceptual sensitivity, our goal is to identify an optimal patch location that effectively exploits model vulnerabilities while remaining inconspicuous to human observers. To account for human perception, we aim to develop a sensitivity map that identifies pixel-level regions within an image that exhibit low sensitivity to human vision (Figure 4.4). Variance in an image is a measure of the dispersion of pixel intensity values within a local region of the image, indicating the level of texture complexity and contrast. Higher variance regions typically contain more detailed textures and rapid intensity changes, while lower variance regions are smoother and more uniform in appearance (Figure 4.2). Contrary to highly sensitive locations that are characterized by relatively low variance values, regions with high variance, particularly those with intricate textures, can accommodate significantly large perturbations while remaining imperceptible to human scrutiny (29). These areas have been favored for global adversarial attacks (8; 35).

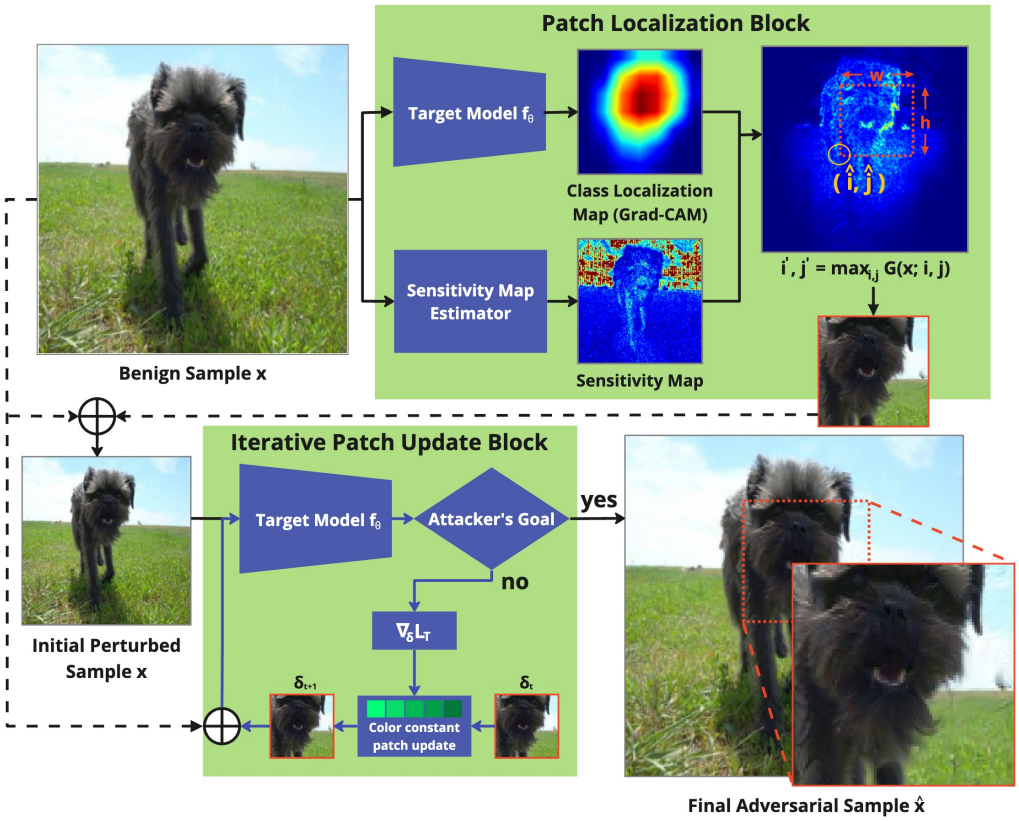

Figure 4.1: The overall pipeline of our method for conducting targeted attacks with imperceptible adversarial patches, consisting of both patch localization and iterative patch update blocks.

However, in adversarial patch attacks, the attack region is restricted within a localized region, thereby limiting the attack's area and hence effectiveness. Consequently, a reliance solely on high perturbation levels on pixels with high perturbation affinity, without strategic placement to enhance attack ability, may not yield a desirable result. Therefore extrapolating the knowledge pertaining to imperceptibility gained from global adversarial attacks into adversarial patch attack, our patch localization step is designed to arrive at an equilibrium that balances the vulnerability of the location and the capability to accommodate large perturbations without being visually salient. More specifically, we propose a notion of perturbation priority index $G(\boldsymbol{x}; i, j)$ for any possible location (i, j) with respect to the victim model f and $(\boldsymbol{x}, y)$, where the optimal location (i', j') is determined based on $G(\boldsymbol{x}; i, j)$:

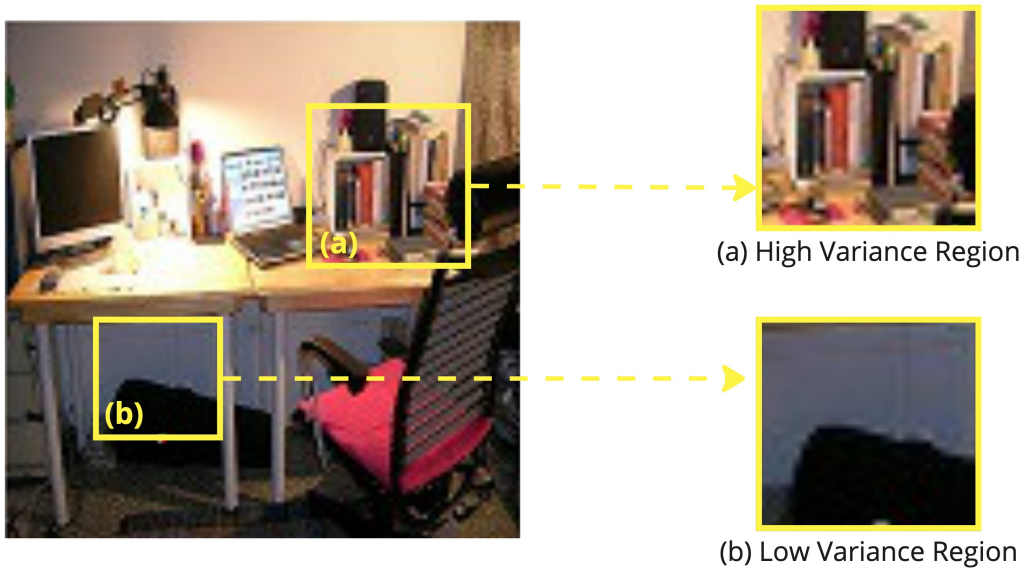

Figure 4.2: Illustration of high and low variance regions within an image. High variance regions can accommodate larger perturbations while remaining less perceptible, whereas low variance regions are more sensitive to visual changes.

$$(i', j') = \arg\max_{i,j} G(\boldsymbol{x}; i, j), \text{ where } G(\boldsymbol{x}; i, j) = \sum_{k=0}^{w} \sum_{l=0}^{h} \frac{J_y(\boldsymbol{x}; i+k, j+l)}{\text{Sens}(\boldsymbol{x}; i+k, j+l)}, \tag{4.3}$$

where h and w represent the height and width of the patch, $J_y(\cdot, \cdot)$ denotes the class localization map capturing the model susceptibility with respect to the ground-truth label class y, and $\text{Sens}(\cdot, \cdot)$ is the sensitivity map capturing the perturbation sensitivity to human visual system. The perturbation priority metric aims to strike a balance between two aspects, seeking the most optimal location (i, j), the window from which has the highest value for $G(\boldsymbol{x}; i, j)$ such that it facilitates attack capabilities, while also showing a high affinity for accommodating large perturbations.

4.2.1 Estimating Model Sensitivity through Class Localization Map

To obtain the class localization map $J_y(\cdot, \cdot)$, we employ Grad-CAM (45). Since we consider white-box settings, we can directly employ the parameters of the victim model f_θ to obtain model-specific attention maps for the given input image $\boldsymbol{x}$. The computation process involves computing the gradient of the last fully connected layer's output denoted as $g_\theta(\boldsymbol{x}, y)$, where y is the ground-truth class of the benign input $\boldsymbol{x}$. Let $\mathbf{A}^k$ be the k-th feature map of the model's last convolution layer and α_k^y be its weight that characterizes the importance of k-th feature map in predicting class label y. To be more specific, α_k^y is calculated by taking a global average pool over its calculated gradient as follows:

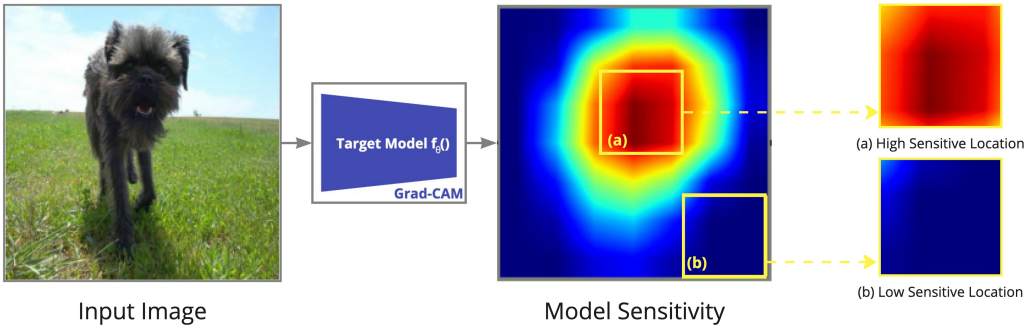

Figure 4.3: Illustration of the Model Sensitivity/Class Activation Map generated using Grad-CAM. The optimization process aims to identify regions that are high sensitivity to adversarial perturbations.

$$\alpha_k^y = \frac{1}{u \times v} \sum_{i=0}^{u} \sum_{j=0}^{v} \frac{\partial g_\theta(\boldsymbol{x}, y)}{\partial A_{ij}^k}, \tag{4.4}$$

where u and v are the height and width of the feature map A^k. The final class localization map is calculated as the weighted sum of all feature maps followed by a ReLU function given by:

$$J_y(\boldsymbol{x}; i, j) = \text{ReLU}(\sum_k \alpha_k^y \cdot A_{ij}^k), \text{for any pixel location } (i, j). \tag{4.5}$$

4.2.2 Estimating Human Perception Sensitivity through Sensitivity Map

Following prior works (8; 35), we aim to position the patch in regions of high variance, while avoiding placement on object edges that are aligned with the coordinate axes. To ensure both factors when defining the sensitivity map $\text{Sens}(\cdot, \cdot)$, we calculate the mean standard deviation of the pixel across the color channels along the horizontal and vertical axes, considering adjacent pixels, denoted as σ_{ij}^x and σ_{ij}^y respectively. Finally, the value of the sensitivity map at (i, j) is computed as the reciprocal of the standard deviation given by:

$$\text{Sens}(\boldsymbol{x}; i, j) = \frac{1}{\sigma_{ij} + \lambda}, \text{ where } \sigma_{ij} = \sqrt{\min(\sigma_{ij}^x, \sigma_{ij}^y)}, \tag{4.6}$$

where $\lambda > 0$ is a small value chosen to prevent division by zero. The sensitivity map induces a human perspective to the perturbation priority measure $G(\boldsymbol{x})$ in terms of perturbation sensitivity, such that those locations that cannot host large perturbations have a higher sensitivity than their counterparts. In the following, we use $J_y(\boldsymbol{x})$ and $\text{Sens}(\boldsymbol{x})$ for the class localization and sensitivity maps of $\boldsymbol{x}$.

Algorithm 1 Imperceptible Adversarial Patch Attack

Input: benign example $(\boldsymbol{x}, y)$, target class y_{targ}, victim model f_θ, and parameters s, T, η, w, h
  $J_y(\boldsymbol{x}) \leftarrow$ compute the class localization map of $\boldsymbol{x}$ based on Equation 4.5
  $\text{Sens}(\boldsymbol{x}) \leftarrow$ compute the sensitivity map of $\boldsymbol{x}$ based on Equation 4.6
  $(i', j') \leftarrow$ find the optimal patch location based on Equation 4.3
  $\boldsymbol{m} \leftarrow$ define the mask indexed by (i', j') with patch size $w \times h$
  Initialize $\boldsymbol{\delta}_0 \leftarrow \boldsymbol{x}$
  for $t = 0, 1, \ldots T-1$ **do**
    if prediction confidence $f_\theta(y_{\text{targ}}|\hat{\boldsymbol{x}}) \geq s$ **then**
      return $\hat{\boldsymbol{x}}$
    else
      $\mathcal{L}_T \leftarrow$ define the total adversarial loss function based on Equation 4.10
      $\boldsymbol{\delta}_{t+1} \leftarrow \boldsymbol{\delta}_t - \eta \cdot \overline{\nabla_{\boldsymbol{\delta}}}\, \mathcal{L}_T(\boldsymbol{\delta}_t; \theta, \boldsymbol{x}, y) \odot \big(\boldsymbol{\delta}_t \oslash \text{Sens}(\boldsymbol{x})\big)$
      $\boldsymbol{\delta}_{t+1} \leftarrow \text{clip}(\boldsymbol{\delta}_{t+1}, 0, 1)$
    $\hat{\boldsymbol{x}} \leftarrow \boldsymbol{x} +_{i',j'} \boldsymbol{\delta}_{t+1}$
Output: $\hat{\boldsymbol{x}}$

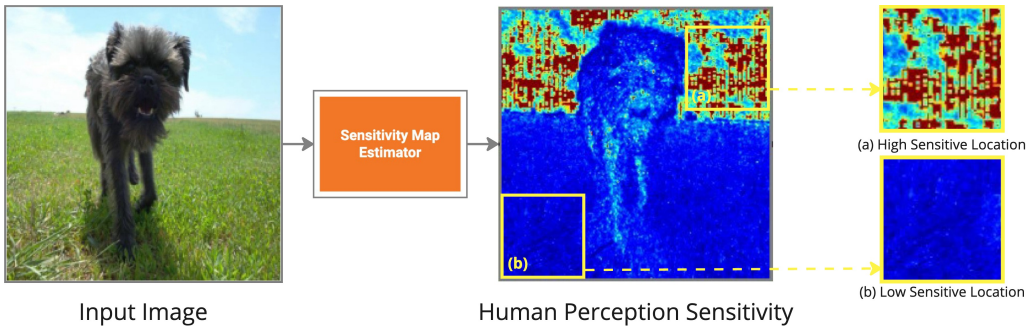

Figure 4.4: Illustration of the generated Human Sensitivity Map. The optimization process aims to identify regions that are less visually susceptible to large perturbations. During the subsequent perturbation optimization stage, updates are applied ensuring that significant perturbations are introduced in areas identified as less sensitive to human perception.

4.3 Optimization of Perturbation Update

During the update procedure, we aim to ensure two key aspects: Firsly, Considering the human perception, we intend to deposit high perturbation in locations where the affinity towards high perturbation is more and keep it restricted to locations that are deemed sensitive from the sensitivity map developed from Section 4.2.2. Secondly, the update rule should preserve the original pixel's gray scale value, as the human visual system is highly sensitive to abrupt changes in base color, which can make the perturbations more noticeable. Instead, we encourage modifications that adjust the brightness and saturation of the base color. This approach ensures that pixel alterations remain consistent with

the gray scale values of the surrounding pixels, thereby enhancing visual coherence and reducing perceptible saliency.

We start with initializing the patch with the original pixel values at the optimal location (i', j') based on Equation 4.3. We introduce a two-fold solution to integrate the human visual system into the perturbation crafting process. First, we introduce a regularization term to the adversarial loss to learn perturbations that are less salient to the human eye. Second, we utilize an update rule that considers human indifference to gray-level quantization as its basis to update the perturbation to gain visual advantage.

In particular, we utilize the following distance metric, introduced in (35), as the regularization term to penalize the visual distortion of $\hat{\boldsymbol{x}}$ with reference to $\boldsymbol{x}$:

$$D(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{h \times w} \sum_{k=i'}^{i'+w} \sum_{l=j'}^{j'+h} \mathrm{Sens}(\boldsymbol{x}; k, l) \cdot |x_{kl} - \hat{x}_{kl}|, \tag{4.7}$$

where $\mathrm{Sens}(\boldsymbol{x}; k, l)$ is defined according to Equation 4.4. $D(\boldsymbol{x}, \hat{\boldsymbol{x}})$ incorporates the sensitivity of the human visual system in measuring the difference between the original and the adversarial example.

For representational convenience the equation can be also written with the following abstraction as follows:

$$D(\boldsymbol{x}, \hat{\boldsymbol{x}}) = \frac{1}{h \times w} \sum_{k=i'}^{i'+w} \sum_{l=j'}^{j'+h} d(k, l), \tag{4.8}$$

where $d(k, l)$ is the distance value measured at pixel (k, l) given by the following:

$$d(k, l) = \mathrm{Sens}(\boldsymbol{x}; k, l) \cdot |x_{kl} - \hat{x}_{kl}|, \tag{4.9}$$

For a given pixel (k, l), the distance metric $d(k, l)$ is defined as the product of the human sensitivity value at that location and the magnitude of the adversarial perturbation applied to it. This metric yields a higher value when both the perturbation magnitude and the sensitivity of the pixel are significant, indicating a greater likelihood of the perturbation being perceptible to the human visual system.

We argue that employing such a distance metric as a regularization term in the final loss function will encourage producing large perturbations at locations, where the sensitivity of the human vision is limited while suppressing the perturbations in locations that are highly sensitive. Accordingly, we can achieve large perturbations favoring the attack while maintaining the overall insensitivity in appearance. The central part of the loss function remains consistent with existing attacks, comprising two cross-entropy loss terms with respect to the target class y_{targ} and the ground-truth class y, respectively. Specifically, the final adversarial loss objective that we aim to minimize is given by:

$$\mathcal{L}_T(\boldsymbol{\delta}; \theta, \boldsymbol{x}, y) = w_1 \cdot \mathcal{L}_{\mathrm{CE}}(\hat{\boldsymbol{x}}, y_{\mathrm{targ}}; \theta) - w_2 \cdot \mathcal{L}_{\mathrm{CE}}(\hat{\boldsymbol{x}}, y; \theta) + w_3 \cdot D(\boldsymbol{x}, \hat{\boldsymbol{x}}), \tag{4.10}$$

where w_1, w_2, and w_3 are weight parameters that regulate the contribution for each term and can be adjusted based on the preference of the attack.

Moreover, the loss function is accompanied by an update rule through which we aim to achieve two main objectives. Motivated by the fact that humans are highly indifferent to

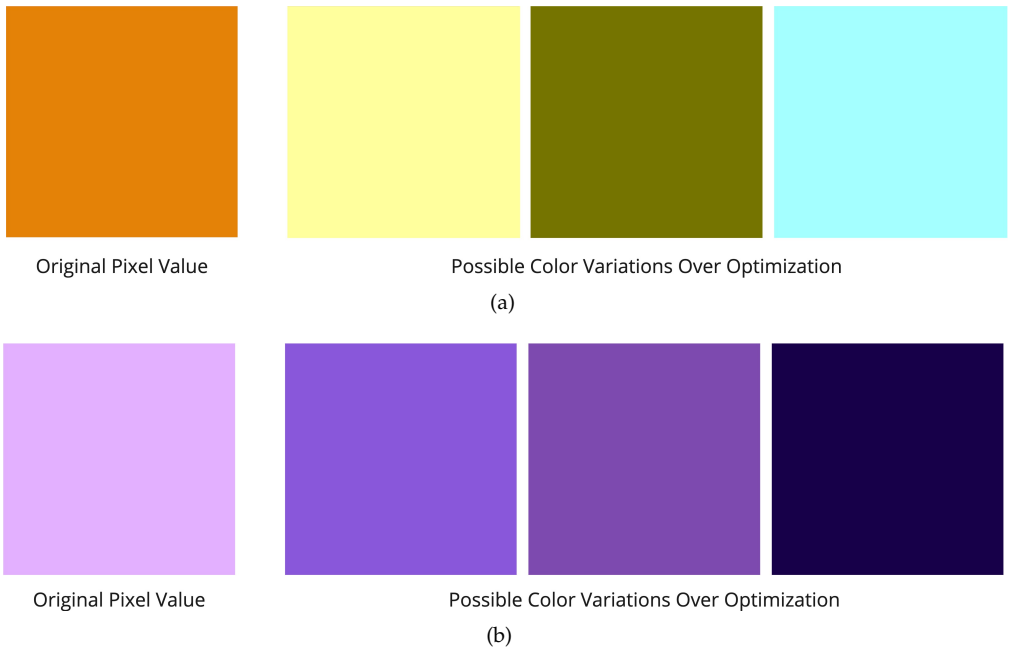

Figure 4.5: Comparison of Possible Color Variations Achievable through (a) Adam Optimization Update Rule and (b) Proposed Update Rule.

changes in brightness and saturation levels of the same base color, which is analogous to the behavior with lower quantization levels, we update the perturbation such that the base color of the pixel does not alter. To ensure and implement this requirement, it is essential to apply changes of equal magnitude across all color channels of the pixel under consideration. As a result, we opted not to employ the update rules provided by conventional optimizers such as Adam. These standard update rules rely on gradients computed individually for each channel, which may differ in magnitude, leading to unequal updates across channels. Consequently, this discrepancy have the capability to make the altered pixel more salient, as the final color on which these optimization processes can converge can be far off from the base color, as illustrated in Figure 4.5 (a), compared to our proposed method which results in a pixel value that still maintains the base color of the pixel 4.5 (b). Next, we aim to maximize the utility of gradient magnitude information of the loss function with respect to the input, while adhering to color constraints. Such a strategy is designed to minimize the number of iterations required for the attack as the update magnitude can be still adapt to the gradient magnitude which is much more informed of the loss landscape. To be more specific, we propose the following gradient update rule:

$$\boldsymbol{\delta}_{t+1} = \boldsymbol{\delta}_t - \eta \cdot \overline{\nabla_{\boldsymbol{\delta}}}\, \mathcal{L}_T(\boldsymbol{\delta}_t; \theta, \boldsymbol{x}, y) \odot \Big(\boldsymbol{\delta}_t \oslash \mathrm{Sens}(\boldsymbol{x})\Big), \text{ for } t = 0, 1, \ldots, T-1, \tag{4.11}$$

where $\odot$ (resp., $\oslash$) stands for element-wise multiplication (resp., division), $\overline{\nabla_{\boldsymbol{\delta}}}$ denotes the averaged gradient of the loss function $\mathcal{L}_T$ over the three color channels, and η is the step size. Averaging over the three channels ensures that each pixel channel is updated by the same amount, thereby ensuring that the base color is not changed. We note that Croce et al. (8) proposed a similar update rule, but their method does not enforce any

color constraint. our proposed regularized loss and update rule contributes significantly to realizing targeted goals with imperceptible adversarial patches.

Chapter 5
Experiments and Results

This chapter provides a comprehensive account of the experimental setup and design, detailing the datasets, attack scenarios, and evaluation metrics employed to assess the proposed method. The evaluation is conducted across multiple dimensions through both qualitative and quantitative analyses. We First give a brief on the datasets that are utilized in the course of this study. We then describe the experimental setup that we consider for the proof of concept and moving ahead for the broader study, giving details on the architectures, and other design considerations. We then present our results and compare our method's attack efficacy and imperceptibility with that of other baseline methodologies. This is followed by a detailed ablation studies which help us understand the effect of the hyperparameters on the attack along with its nature. Finally, considering evasion of the existing defense methods as one of our central goals, we demonstrate the attack abilities of our method against some state-of-the-art adversarial patch defense methods.

5.1 Dataset Utilized Throughout the Work

To establish a proof of concept for our proposed method, we conducted an initial validation using the Stanford Dogs dataset (23). This dataset, a subset of ImageNet, comprises 120 classes, each corresponding to a distinct dog breed with a total of 20,580 samples. Few samples of the Stanford Dogs dataset are presented in Figure 5.1. For our evaluation, we carefully selected one image per class, ensuring that each chosen sample was correctly classified by the victim model under consideration. Given our objective of extending the validation to the broader ImageNet dataset, leveraging the Stanford Dogs dataset allowed us to gain meaningful insights into the method's effectiveness, while also highlighting the strengths and potential limitations of the design choices made throughout development.

For the core part of our study, where we assess the efficacy of our method in terms of both attack success and imperceptibility in a classification task, we conducted evaluations using the ILSVRC 2012 validation set (42). This dataset comprises 1,000 classes, with 1.28 million training images, 50,000 validation images, and 10,000 test images. Few samples

of the ILSVRC 2012 validation set are presented in Figure 5.2. For our evaluation, we selected a subset of the validation set, ensuring that it contained a correctly classified image per class based on the predictions of the victim model. We utilized this dataset as the primary benchmark for a comprehensive evaluation of our method. It served as the foundation for all comparisons against existing attack methodologies and was also employed in experiments assessing the method's evasive capabilities against various defense mechanisms.

Figure 5.1: Example images from the Stanford Dogs dataset, showcasing a variety of dog breeds included in our evaluation

Figure 5.2: Example images from the ImageNet dataset, showcasing samples from a variety of classes included in our evaluation

In addition to the image classification task, we conducted additional evaluation for the

face recognition task. VGG Face is a large-scale face recognition dataset developed by at the University of Oxford by Parkhi et al. (40). It contains 2.6 million images of 2,622 identities, collected from the internet. We used a subset of the test set of this dataset, similar to Li et al. (28). The dataset consisted of a total of 470 images across 10 classes. A sample from each of the classes considered from the VGG Face dataset are presented in Figure 5.3.

Figure 5.3: Example images from the VGG Face dataset, showcasing samples from a variety of classes included in our evaluation

5.2 Experimental Setup

In this section, we highlight the experimental design that is used in this work to evaluate the proposed method of creating imperceptible adversarial patches. We structured our experimental studies into two distinct phases. The first phase focused on developing a proof of concept for our proposed method, utilizing the Stanford Dogs dataset, as discussed in Section 5.1. For each class we considered the performance for both attack ability and imperceptibility of a samples that are correctly classified by the victim model. For this phase, we selected ResNet-50 as the victim model, a deep convolutional neural network with 25.6 million parameters, pretrained on ImageNet (16).

Given that the Stanford Dogs dataset is a subset of ImageNet, we employed the pre-trained ResNet-50 model without modifying its final layers. This ensured that the model retained its original classification capabilities, allowing us to assess our method's effectiveness without introducing additional biases from retraining or fine-tuning. This setup provided a controlled environment for evaluating our approach before proceeding to more extensive experiments in the subsequent phase.

After validating our initial concepts, we proceeded with a comprehensive analysis and evaluation of our method. This phase focused on assessing both attack efficacy and imperceptibility on a larger and more diverse dataset. As discussed in Section 5.1, we selected a subset of the ILSVRC 2012 validation set, comprising 1,000 correctly classified images, with one representative image from each of the 1,000 classes provided they are correctly classified by the victim model.

To evaluate the cross-architecture performance of our method, we conducted experiments across multiple deep learning architectures. Specifically, we considered four different

models, spanning both convolutional neural networks (CNNs) and the more recently developed transformer-based architectures. This selection allowed us to analyze the generalizability and robustness of our method across fundamentally different neural network designs, ensuring a more thorough and reliable evaluation. From the convolutional neural network (CNN) family, we selected ResNet-50 and VGG16 as representative architectures. ResNet-50, a widely used deep residual network, comprises 25.6 million parameters (16), while VGG16, known for its deep yet uniform structure, consists of 138 million parameters (49). These models were chosen due to their strong feature extraction capabilities and their historical significance in image classification benchmarks. From the transformer-based architectures, we considered the Swin Transformer Tiny and the Swin Transformer Base models. The Swin Transformer Tiny has 28 million parameters, while the Swin Transformer Base contains 88 million parameters (34). These models leverage a hierarchical vision transformer (ViT) approach, introducing shifted window attention mechanisms that improve computational efficiency and performance on vision tasks. The underlying concept of Vision Transformers (ViT) was originally introduced by Dosovitskiy et al. (10), demonstrating that pure transformer models can outperform CNNs on image classification tasks when trained on large datasets. By incorporating both CNN-based and transformer-based architectures, we aimed to evaluate the generalizability and robustness of our method across fundamentally different network designs. This comprehensive selection allowed us to analyze our method's performance under diverse feature extraction paradigms, ensuring a more rigorous and insightful assessment. It is to note that all of our considered architectures are pretrained on ImageNet for this particular task and for the attack scenario we considered a white-box setting where all the model parameters are known to the attacker.

Along with evaluating our method's performance we compared it to state-of-the-art attack methods such as Google Patch by Brown et al. (4), LaVAN by Karmon et al. (22), GDPA by Li et al. (28) and Masked Projected Gradient Descent (MPGD), which is an extension of the standard PGD attack introduced by Madry et al. (36). For GDPA, we consider a balanced scenario between attack efficacy and imperceptibility by setting their visibility parameter α to 0.4 in our experiments. In addition, we evaluate the effectiveness of our attack against existing defense methods designed specifically against adversarial patch attacks. The defense methods considered includes methods described in both Section 2.4.1 (7; 15; 31; 53) and Section 2.4.2 (13; 21).

In order to simulate a realistic scenario in a digital space we opt for the face recognition task. Following a similar set up as Li et al. (28), we used the test set of the VGG Face dataset, as we discussed earlier in Section 5.1. The dataset is consisting of a total of 470 images across 10 classes. We test the task across all the architecture that we considered for the image classification task with slight alteration to fit to the task. We utilize the pre-trained version of these model which are trained on ImageNet and re-trained on the train set of the VGG Face dataset for the mentioned classes that contained a total of 3178 images spanning the 10 classes. The retraining procedure follows the same specifications as used by Li et al. (28). In particular, We finetuned each of the pretrained architecture with an Adam Optimizer with a starting learning rate of 10^{-4} followed by a drop of 0.1 every 10 epochs. We configure the batch size to be 64. To prevent overfitting, we monitor the validation set accuracy for hyperparameter tuning and model selection.

In all experiments, images from both tasks were resized to a standardized dimension of 224 $\times$ 224 before applying the attack to fit to the requirements of the architectures considered. For both ImageNet and VGG Face datasets, we iteratively updated the adversarial patch until one of the following conditions was met:

- The confidence score of the target class reached 0.9, indicating a high-confidence misclassification.
- A maximum of $1,000$ iterations was reached, ensuring computational efficiency.

Depending on the attack's success, we reinitialized the step size and restarted the optimization process, with a maximum of three reinitializations.

All experiments were conducted on a single NVIDIA A100 GPU with 80GB of memory, leveraging its high computational power for efficient optimization and evaluation. We implemented our method using PyTorch, a widely used deep learning framework, ensuring reproducibility and ease of experimentation.

5.3 Evaluation metrics

We aim to evaluate the performance of our proposed method across two key aspects: attack efficacy, which measures the method's ability to successfully mislead the target model away from the original class in the untargeted attack scenario and into the target class in the targeted attack scenario, and imperceptibility, which ensures that the generated adversarial patch remains as unnoticeable as possible to human observers. By analyzing these two dimensions, we seek to demonstrate both the strength of our attack and its ability to remain visually inconspicuous.

For attack efficacy we evaluate the effectiveness of different attack methods based on attack success rate, denoted as ASR, which characterizes the ratio of instances that can be successfully attacked using the evaluated method. Let $\mathcal{A}$ be the evaluated attack, f_θ be the victim model, and $\mathcal{S}$ be a test set of correctly classified images. The ASR of $\mathcal{A}$ with respect to f_θ and $\mathcal{S}$ is defined as:

$$\mathrm{ASR}(\mathcal{A}; f_\theta, \mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{\boldsymbol{x} \in \mathcal{S}} \mathbb{1}\big(f_\theta(\hat{\boldsymbol{x}}) = y_{\mathrm{targ}}\big), \tag{5.1}$$

where $|\mathcal{S}|$ denotes the cardinality of $\mathcal{S}$, and $\hat{\boldsymbol{x}}$ is the adversarial example generated by $\mathcal{A}$ for $\boldsymbol{x}$.

To comprehensively assess the imperceptibility of adversarial patches, we evaluate them across multiple similarity metrics, ensuring a thorough understanding of their visual impact. Our evaluation includes both traditional statistical methods and learned similarity measures derived from convolutional neural networks (CNNs). Traditional statistical methods provide a low-level pixel-wise analysis, capturing differences in color, texture, and structural properties, while CNN-based similarity measures leverage deep feature representations to assess perceptual similarity from a human visual perspective. By incorporating both approaches, we obtain a more robust and holistic evaluation of imperceptibility, ensuring that the adversarial perturbations remain visually inconspicuous while maintaining their attack efficacy. In an ideal scenario, an adversarial sample should be visually identical to its corresponding benign sample, ensuring no perceptible differences to the human eye. This imperceptibility must be maintained at both a global level, where the entire image should retain its original structure, color distribution, and overall coherence, and at a local level, where the specific region modified by the adversarial patch should seamlessly blend with its surroundings without introducing noticeable artifacts or inconsistencies.

To achieve this, we assess global similarity by ensuring that the overall composition, texture, and spatial integrity of the image remain unchanged, preventing any distortions that might reveal the presence of perturbations or draw unwanted attention. Simultaneously, local similarity is examined to verify that the adversarial patch does not disrupt the fine-grained details, edges, or contextual consistency within the affected area. The patch should integrate naturally with the surrounding pixels, avoiding any structural changes to the original image as this leads to unnatural appearance that could be detected either by human observers or automated detection mechanisms.

The traditional statistical method-oriented measures involve the Structural Similar Index Measure (SSIM) (61),the Universal Image Quality index (UIQ) (60) and the Signal to Reconstruction Error ratio (SRE) (26), while the learned similarity measures involves the CLIPScore (17), and the Learned Perceptual Image Patch Similarity (LPIPS) metric (68).

SSIM by Wang et al. (61) is a perceptual metric used to assess the similarity between two images. SSIM considers luminance, contrast, and structural similarity to provide a more human-perceptual assessment of image quality with a global view of the sample. The range of the metric is from -1 to 1, where 1 represent perfect similarity between the samples, 0 means no similarity and -1 represents that they are structurally different. The Structural Similarity Index Measure (SSIM) between two images x and y is defined as:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{5.2}$$

where μ_x and μ_y represent the mean intensities of images x and y, while σ_x^2 and σ_y^2 denote their respective variances, capturing contrast information. The term σ_{xy} corresponds to the covariance between x and y, reflecting structural similarity. The constants C_1 and C_2 are small positive values introduced to prevent division by zero and ensure numerical stability.

UIQ by Wang et al. (60) measures the similarity between two images by evaluating losses in correlation, luminance, and contrast. UIQ provides a more comprehensive assessment by considering structural distortions, yielding a single index within -1 and 1. 1 means the two images are identical, 0 indicates that there is no correlation and -1 represents there is strong structural distortion among the two samples. This metric is particularly useful in assessing the similarity between the two samples, as it provides a more perceptually relevant evaluation of image degradation compared to traditional pixel-wise measures.

The Universal Image Quality Index (UIQ) between two images x and y is defined as:

$$UIQ(x, y) = \frac{4\sigma_{xy}\mu_x\mu_y}{(\sigma_x^2 + \sigma_y^2)(\mu_x^2 + \mu_y^2)} \tag{5.3}$$

where μ_x and μ_y represent the mean intensities of images x and y, respectively. The terms σ_x^2 and σ_y^2 denote their respective variances, while σ_{xy} represents the covariance between x and y. This formulation ensures that UIQ simultaneously accounts for structural correlation, contrast, and luminance degradation between the images.

SRE by Lanaras et el. (26), evaluates the quality of reconstructed images by comparing the original signal strength to the reconstruction error. It better compares errors across images with varying brightness, unlike Peak Signal to Noise Ratio (PSNR) which uses a fixed peak value. It measures how accurately an image has been reconstructed after any alteration. In our case, we attempt to measure the similarity between the adversarially

modified image and that of the benign image. A higher SRE value indicates better similarity between the two samples, meaning the adversarial image closely resembles the original benign image.

The Signal to Reconstruction Error Ratio (SRE) is defined as:

$$SRE = 10 \log_{10} \left(\frac{\mu_x^2}{\frac{1}{n} \|\hat{x} - x\|_2^2} \right) \tag{5.4}$$

where μ_x is the mean of the original image x, x_i represents the original image, $\hat{x}_i$ represents the reconstructed image, and n is the total number of pixels.

From the learned similarity measures we evaluate the CLIPScore and LPIPS that quantify perceptual similarity using pre-trained DNNs, capturing nuanced visual features. CLIPScore is a metric used to evaluate the similarity between two images using the CLIP (Contrastive Language-Image Pretraining) model, that we utilize to measure the similarity between the adversarial and the benign image. The CLIP model embeds both images into a shared multi-dimensional space, where the similarity between them is measured by the cosine similarity between the derived embeddings.

LPIPS by Zhang et al. (68) is a perceptual image similarity metric that measures the distance between images based on deep neural network features rather than pixel-wise comparisons. It computes the similarity by comparing the activations from a pretrained convolutional network (e.g., VGG or AlexNet) at different layers. LPIPS focuses on perceptual differences, capturing high-level structural, semantic, and textural information, making it more aligned with human perception than traditional metrics. In our experiments, we consider VGG as the pretrained convolutional network to measure the LPIPS measure.

In summary, our evaluation framework rigorously assesses both the attack efficacy and imperceptibility of the adversarial patches to ensure a comprehensive analysis of our method's performance. By incorporating a combination of traditional statistical similarity measures that evaluate specific features at statistical level and learned perceptual metrics that extract more abstract feature cues imitating human perception, we establish a robust evaluation strategy that captures both low-level pixel differences and high-level perceptual discrepancies.

Chapter 6
Proof of Concept: Experimental Results on the Stanford Dogs

6.1 Experimental Details

As outlined in Section 5.2, we conducted extensive experimentation and evaluation on the Stanford Dogs Dataset during the development phase of our proposed methodology. Our primary objective was to utilize a dataset that would allow for rapid validation of our approach while maintaining sufficient diversity to rigorously assess the robustness of the method. The Stanford Dogs Dataset, consisting of various dog breeds, provided an ideal testbed due to its rich diversity in natural image content and structural variations.

To systematically analyze the impact of the target class on the attack performance, encompassing both attackability and evasive capabilities, we selected five distinct target classes: iPod, Baseball, Toaster, Goldfish, and English Setter. The rationale behind this selection was to introduce a balanced mix of both natural and artificial elements, thereby enabling a more comprehensive evaluation of the adversarial method's attackability in relation to the contextual disparity between the target class and the host environment. We kept the size of the attack region to be 84×84 which covered about 14% of the total image.

Since the Stanford Dogs Dataset primarily consists of images depicting various dog breeds, it inherently represents natural elements significantly more than man-made environments. Consequently, our selection of target classes was designed to span both the spectrum, ensuring that the adversarial attack's efficacy and efficiency, which is the ease of conducting the attack, could be tested in settings where the target class are aligned with the natural elements of the host image and also when the target class represents total misalignment with host environment. This deliberate selection allowed us to investigate the influence of image semantics and contextual background on adversarial robustness and its ease, thereby yielding deeper insights into the method's generalization capabilities.

Since the Stanford Dogs Dataset primarily comprises images of various dog breeds, it predominantly represents natural environments rather than man-made settings. To ensure a comprehensive evaluation of our adversarial attack's efficacy and efficiency, which is the ease of conducting the attack, we strategically selected target classes that span both ends of the spectrum—those that naturally align with the host image's context and those that exhibit complete semantic misalignment. This deliberate choice enabled us to assess the impact of image semantics and contextual background on adversarial robustness and attack feasibility, thereby offering deeper insights into the method's generalization capabilities across diverse visual domains. Ideally, our proposed method should demonstrate stable attack performance, remaining invariant to the choice of the target class. However, we hypothesize that the effectiveness of the attack is influenced by the degree of semantic alignment between the target class and the original benign sample. Specifically, when the target class shares greater visual or contextual similarity with the benign sample, the adversarial perturbation is expected to be more seamlessly integrated, making the attack both more effective and less detectable. Conversely, when the target class exhibits significant dissimilarity from the benign sample, the attack may require stronger perturbations, potentially affecting both its success rate and imperceptibility. This hypothesis underscores the importance of understanding the interplay between adversarial feasibility and semantic coherence, offering insights into the fundamental properties that govern adversarial vulnerability in deep learning models.

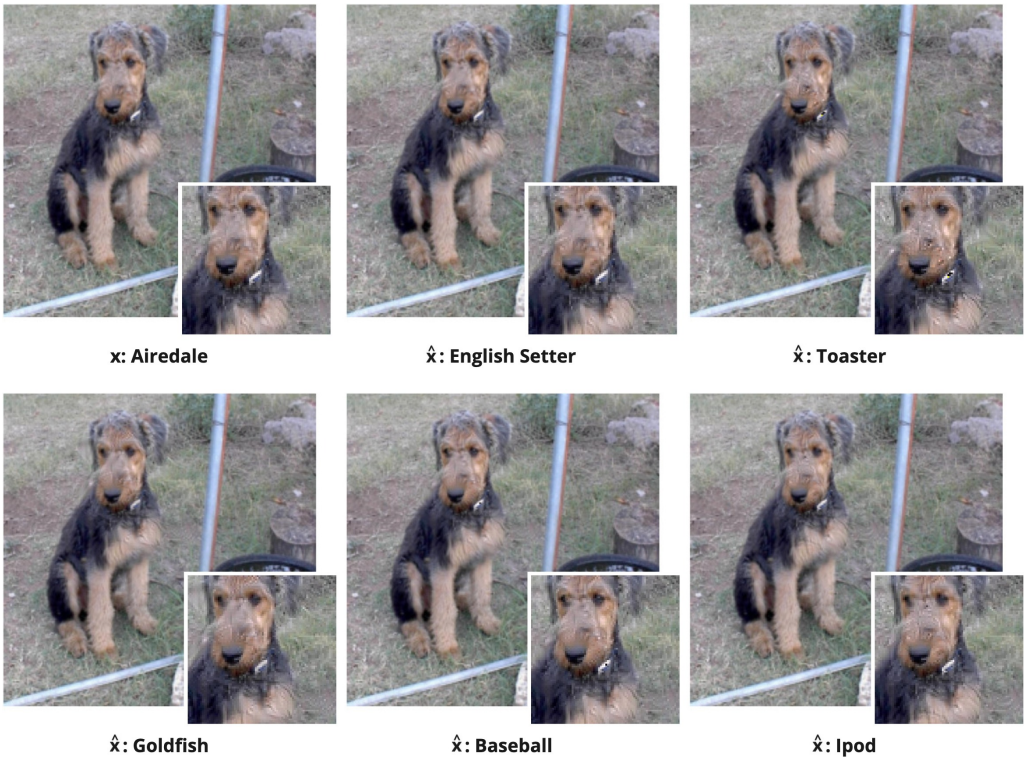

Figure 6.1: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to different target class on the Stanford Dogs Dataset. x represent the benign sample's original class and $\hat{x}$ represent the target class corresponding to the presented adversarial samples with the generated adversarial patch. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Table 6.1: Detailed evaluation of attack efficacy through ASR (%) and imperceptibility for different target class within Stanford Dogs Dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS. CosSim represents the semantic alignment between the target class and the host class. Empirical evidence supports that target classes closer to the original classes leads to better imperceptibility

$\mathbf{y}_{targ}$	**CosSim**	**ASR(%)**	**Scale**	**Imperceptibility metric**				
				SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Ipod	0.42	98.3	Local	0.93	0.91	28.0	86.0	0.130
			Global	0.99	0.98	37.3	95.8	0.020
Baseball	0.47	99.2	Local	0.95	0.93	28.3	86.2	0.120
			Global	0.99	0.99	37.8	96.0	0.019
Toaster	0.46	100	Local	0.93	0.91	27.9	86.0	0.130
			Global	0.99	0.98	37.3	96.0	0.020
Goldfish	0.55	100	Local	0.91	0.90	27.5	83.7	0.140
			Global	0.98	0.98	36.8	95.4	0.020
English Setter	0.72	100	Local	0.97	0.96	30.6	92.3	0.100
			Global	1.00	0.99	40.0	98.0	0.016

6.2 Results and Discussions

We summarize the results demonstrating the performance of our attack methodology in Table 6.1. The outcomes validate that our proposed method has the potential to produce strong attack performance, as quantified by the attack success rate (ASR), while maintaining a high level of imperceptibility, as evidenced by the imperceptibility metrics considered. Furthermore, the performance highlights the method's robustness, with consistent stability observed across the different target classes, reflected in an attack success rate of (99.5 ± 0.67) %. To illustrate the imperceptibility of the adversarial patches generated by our method, we visualize the final adversarial samples $\hat{x}$ corresponding to each class and compare them with the original benign sample x in the Figure 6.1. The visual results further solidifies the effectiveness of our approach in achieving imperceptibility while being able to achieve high attack success rates in the targeted attack scenario.

Based on the conditions outlined in Section 5.2, we evaluated all the successful attack instances for each target class. We assessed the average target class prediction confidence for the adversarial samples generated. The results for these evaluations are as follows: an average prediction confidence of 90.3% for the target class "English Setter", 89.4% for "Toaster", 87.6% for "Goldfish", 83.2% for "iPod", and 86.0% for "Baseball". These results demonstrate that our methodology consistently achieves high confidence levels in its attacks across a wide range of target classes, further underscoring the strength and versatility of our approach.

In addition to the primary results, we conducted a supplementary experiment to further assess the semantic alignment between the target class samples and the host samples. Specifically, we selected five samples from each of the five target classes discussed earlier and computed the similarity of their CLIP embeddings with those of 20 randomly selected samples for each of the 10 randomly chosen classes within the Stanford Dogs dataset.

As hypothesized, although the attack performance remained stable across different target classes, we observed a notable correlation between the performance of the attack and the semantic alignment between the target and host classes. This relationship is empirically supported by the findings: for example, the cosine similarity value between the CLIP embeddings of the "English Setter" target class, which is a class within the 120 classes of Stanford Dogs dataset, and the 20 randomly chosen classes was calculated to be 0.72. To ensure the validity of the results, it was confirmed that the 20 classes considered did not include the "English Setter", as its inclusion could lead to overestimated similarity values. Notably, the "English Setter" class exhibited superior attack performance, as indicated by its higher attack success rate and lower imperceptibility, compared to other target classes like "iPod", which had a cosine similarity value of only 0.42. This comparison underscores the influence of semantic alignment on the efficacy of the attack, suggesting that target classes with higher alignment to the host classes yield better performance in terms of attack success and imperceptibility. The detailed alignment scores are summarized in Table 6.1.

Chapter 7
Extensive Evaluation and Comparison: Experimental Results on ImageNet

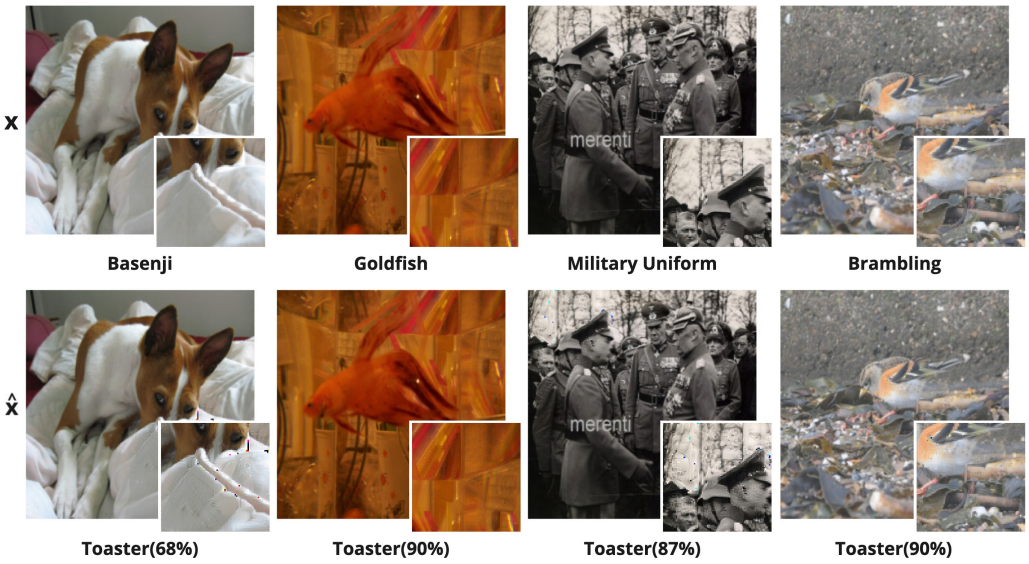

Figure 7.1: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class on the ImageNet Dataset with **VGG16** as the victim model. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Table 7.1: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **VGG16** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LIPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	**100**	Local	0.002	0.000	11.93	32.50	0.760
		Global	0.830	0.820	18.73	73.10	0.190
LaVAN	93.6	Local	0.002	0.000	11.13	33.20	0.790
		Global	0.820	0.810	20.30	76.32	0.230
GDPA	89.2	Local	0.310	0.300	19.90	56.25	0.610
		Global	0.890	0.880	28.00	84.00	0.130
MPGD ($l_\infty, \epsilon = 16/255$)	96.5	Local	0.810	0.800	26.44	73.91	0.320
		Global	0.940	0.920	32.80	94.00	0.090
Ours	**99.1**	Local	**0.900**	**0.860**	**28.94**	**72.70**	**0.230**
		Global	**0.985**	**0.960**	**36.42**	**95.10**	**0.060**

7.1 Experimental Details

After successfully validating the effectiveness of our methodology on the Stanford Dogs dataset, we extended our evaluation to a more diverse and comprehensive dataset to gain deeper insights into its performance across a broader range of scenarios. For this purpose, we selected ImageNet (42), a large-scale dataset known for its extensive diversity in object categories, varying backgrounds, and complex real-world variations. This dataset provides a more challenging benchmark, allowing us to assess the generalizability and robustness of our method beyond a domain-specific setting like Stanford Dogs.

In addition to evaluating our approach, we conducted a comparative analysis against existing state-of-the-art adversarial patch attacks, which serve as baseline methods for our study. Specifically, we compared our method with Google Patch by Brown et al. (4), LaVAN by Karmon et al. (22), GDPA by Li et al. (28), and MPGD. For MPGD, we determined the patch placement location using the optimal placement strategy outlined in our proposed methodology to ensure a fair and effective comparison. The perturbation budget was constrained using an l_∞ norm bound of $\epsilon = 16/255$. This setting provided an optimal balance between attack effectiveness and imperceptibility. The comparison is based on the two aspects of the attack as discussed earlier: Attack ability and imperceptibility.

To maintain consistency with our previous experiments on the Stanford Dogs dataset, we set the adversarial patch size to 84×84 pixels, covering approximately 14% of the total image area. This ensured that differences in attack performance were primarily attributed to methodological variations rather than differences in patch size or placement strategy. We conducted the attack in a white-box setting for each of the four architectures discussed in Section 5.2. Going with the existing literature we considered "Toaster" as the target class for the attack.

7.2 Results and Discussions

Table 7.1 presents a comprehensive evaluation of our proposed method, assessing its effectiveness in terms of both attack success rate (ASR) and patch imperceptibility, using VGG16 as the victim model. Additionally, the table provides a comparative analysis against the baseline adversarial patch methods considered in this study. Our proposed approach demonstrates a high attack success rate, achieving an average ASR of 99.1% across the host classes with a average target class prediction confidence of 79%. This performance is competitive with the baseline methods, with only Google Patch surpassing it by achieving a perfect 100% ASR. In terms of imperceptibility, however, our proposed method achieved the best performance, as evidenced by the imperceptibility metrics considered. Notably, methods that prioritize imperceptibility, such as MPGD and GDPA (with an alpha value of 0.4), exhibit a considerable decline in attack performance. Through empirical analysis, we attribute this reduction to two primary factors: (i) the inherently restricted nature of perturbations in MPGD, which limits the attack's effectiveness, and (ii) the diminished influence of adversarial perturbations in GDPA, where the alpha blending technique reduces the perturbation's relative contribution to the original pixel values. These findings underscore the trade-off between imperceptibility and attack success in the existing adversarial patch designs. Some of the adversarial samples generated in this study are shown in Figure 7.1, providing a visual representation of the imperceptibility of the adversarial patches produced by our method.

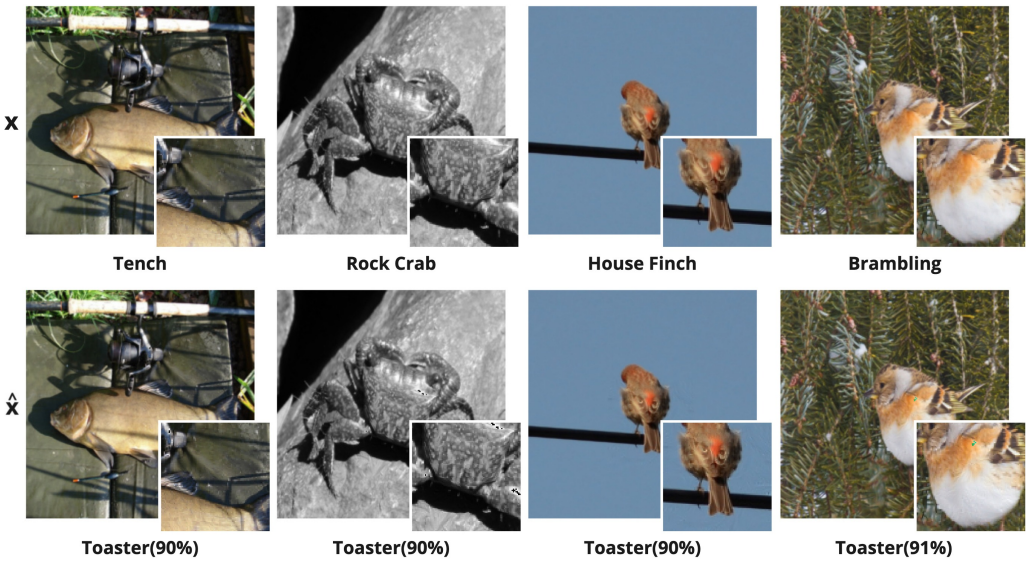

Figure 7.2: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class on the ImageNet Dataset with **ResNet-50** as the victim model. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Table 7.2 presents the performance of the attack for the second set of experiments which utilized the ResNet-50 as the victim model. Our method achieves a 99.5% average attack

Table 7.2: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **ResNet-50** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LIPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	99.1	Local	0.010	0.000	14.20	33.00	0.740
		Global	0.820	0.810	22.90	74.10	0.180
LaVAN	**100**	Local	0.010	0.000	14.20	33.30	0.780
		Global	0.820	0.810	23.40	76.10	0.180
GDPA	93.7	Local	0.350	0.330	19.80	65.20	0.570
		Global	0.920	0.910	28.40	87.10	0.090
MPGD ($l_\infty, \epsilon = 16/255$)	97.8	Local	0.790	0.780	25.30	76.20	0.240
		Global	0.950	0.930	33.60	93.30	0.050
Ours	**99.5**	Local	**0.940**	**0.910**	**28.34**	**84.54**	**0.120**
		Global	**0.990**	**0.970**	**37.23**	**96.52**	**0.020**

success rate across host classes, with an average target class prediction confidence of 87.6%. This outperforms all the state-of-the-art baseline approaches considered, with LaVAN being the only method to outperform it by reaching a 100% attack success rate. For imperceptibility, similar to the performance on VGG16, our proposed method outperformed every other method, evident by the imperceptibility values obtained. The observation previously with respect to methods prioritizing imperceptibility to some level and their impact on attack success rate remains consistent. Figure 8.3 presents a selection of adversarial samples generated in this study, illustrating the high imperceptibility of the adversarial patches crafted using our method.

The performance with the Swin Transformer Tiny as the victim model are summarized in Table 7.3. We achieve an average attack success rate 99.6% with an average target class prediction confidence of 87.0% which is comparable to the other baselines considered. Google Patch achieved the best attack performance with a 100% attack success rate. However as far as imperceptibility metrics are concerned, consistent with observations from VGG16 and ResNet-50 our method was unchallenged. The impact of imperceptibility-focused methods on attack success rate remains consistent with previous observations. Figure 7.3 showcases adversarial samples from this portion, highlighting the strong imperceptibility of the patches generated by our method.

With the Swin Transformer Base as the victim model we summarize our results in Table 7.4. A significant variation in terms of attack success rate was observed for MPGD among other methods where the performance dropped to 70.5% in comparison to what previously was observed for other victim models. Our method achieves an average attack success rate of 99.6% and an average target class prediction confidence of 83.4%, demonstrating performance comparable or better than the baseline methods considered.LaVAN demonstrated the highest attack performance, achieving a 100% ASR. However, in terms of imperceptibility metrics, our method outperformed all others, aligning with previous observations. The trade-off between imperceptibility and attack success rate for other imperceptibility prioritizing methods remained consistent across evaluations. Figure 7.4 presents adversarial samples from this set of experiments, emphasizing the subtlety of

the generated patches.

With the Swin Transformer Base as the victim model we summarize our results in Table 7.4. A significant variation in terms of attack success rate was observed for MPGD among other methods where the performance dropped to 70.5% in comparison to what previously was observed for other victim models. Our method achieves an average attack success rate of 99.6% and an average target class prediction confidence of 83.4%, demonstrating performance comparable or better than the baseline methods considered.LaVAN demonstrated the highest attack performance, achieving a 100% ASR. However, in terms of imperceptibility metrics, our method outperformed all others, aligning with previous observations. The trade-off between imperceptibility and attack success rate for other imperceptibility prioritizing methods remained consistent across evaluations. Figure 7.4 presents adversarial samples from this set of experiments, emphasizing the subtlety of the generated patches.

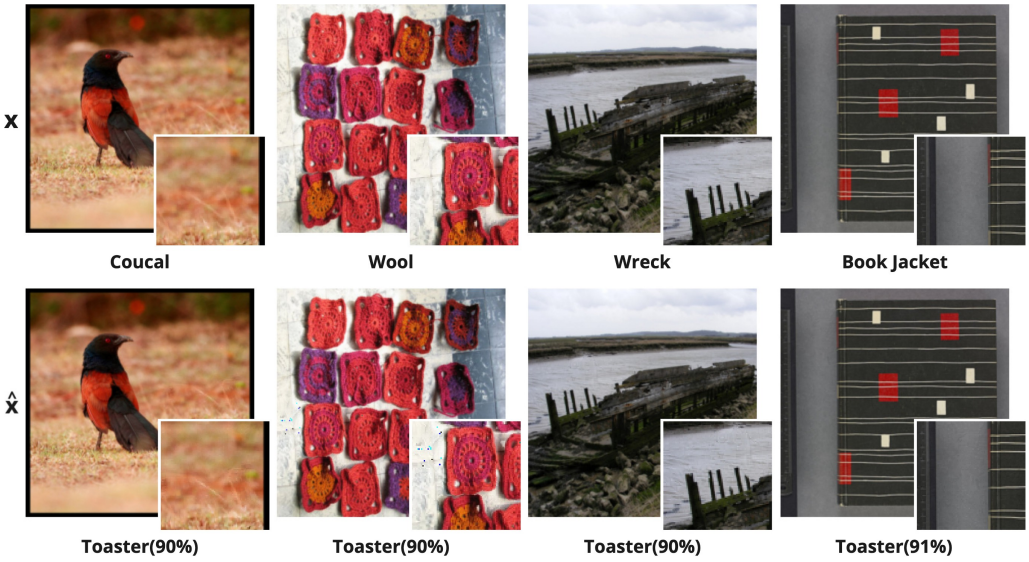

Figure 7.3: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class on the ImageNet Dataset with **Swin Transformer Tiny** as the victim model. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Table 7.3: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Tiny** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LIPIPS.

Method	ASR(%)	Scale	Imperceptibility metric				
			SSIM (↑)	UIQ (↑)	SRE (↑)	CLIP (↑)	LPIPS (↓)
Google Patch	**99.8**	Local	0.002	0.000	11.80	32.80	0.770
		Global	0.830	0.820	18.94	73.90	0.150
LaVAN	99.7	Local	0.005	0.000	14.13	33.10	0.780
		Global	0.820	0.810	23.30	76.32	0.170
GDPA	83.7	Local	0.390	0.360	20.20	63.65	0.540
		Global	0.900	0.890	28.21	85.75	0.100
MPGD ($l_\infty, \epsilon = 16/255$)	98.8	Local	0.800	0.790	25.50	80.54	0.190
		Global	0.940	0.920	33.11	95.80	0.050
Ours	**99.6**	Local	**0.980**	**0.940**	**31.74**	**90.41**	**0.060**
		Global	**0.996**	**0.980**	**40.67**	**98.61**	**0.008**

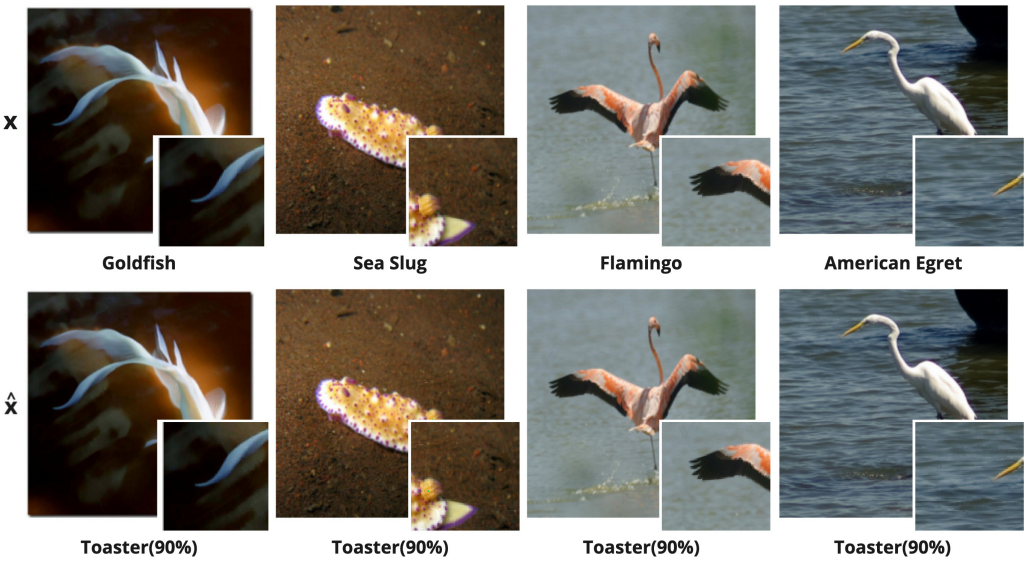

Figure 7.4: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class on the ImageNet Dataset with **Swin Transformer Base** as the victim model. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Across all the four victim models— VGG16, ResNet-50, Swin Transformer Tiny, and Swin Transformer Base— our proposed method consistently demonstrated high attack success rates and superior imperceptibility. The attack success rate remained above 99% across the models, with slight variations in the average target class prediction confidence.

Table 7.4: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LIPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	97.9	Local	0.003	0.000	10.74	32.90	0.770
		Global	0.830	0.820	17.61	73.20	0.170
LaVAN	**100**	Local	0.004	0.000	13.10	33.19	0.780
		Global	0.820	0.810	23.30	76.35	0.180
GDPA	85.1	Local	0.360	0.345	20.40	61.25	0.540
		Global	0.880	0.870	28.00	85.10	0.110
MPGD ($l_\infty, \epsilon = 16/255$)	70.5	Local	0.800	0.800	25.30	74.30	0.200
		Global	0.940	0.920	33.00	92.10	0.050
Ours	**99.4**	Local	**0.970**	**0.910**	**31.30**	**89.33**	**0.070**
		Global	**0.994**	**0.970**	**40.10**	**98.43**	**0.010**

From the performance metrics we observe that ease of attack is significantly limited for architectures like VGG16 and ResNet-50 where it requires more perturbation updates resulting in reduced performance in imperceptibility while the opposite was observed for recently developed models like Swin Transformer Tiny and Swin Transformer Base. We attribute this observation into three key factors:

Network Architecture and Local Receptive Fields:

Convolutional neural networks (CNNs) like VGG16 and ResNet-50 rely on local receptive fields and strong inductive biases. This localized processing makes them more resistant to small, subtle perturbations, requiring stronger and more noticeable modifications to deceive them. As a result, achieving a high attack success rate on these models demands more aggressive perturbation updates, which negatively impacts imperceptibility.

Feature Processing in Transformer-Based Models

Transformer-based models like Swin Transformers leverage hierarchical feature processing and self-attention mechanisms. Unlike CNNs, they do not rely on fixed spatial hierarchies, allowing them to capture global dependencies more effectively. This makes them more susceptible to adversarial perturbations, as subtle changes in one region of the image can influence feature representations across the entire model.

Perturbation Propagation and Effectiveness

Perturbation propagation differs significantly between these architectures. CNNs primarily process features locally, meaning adversarial noise remains concentrated in specific regions, often requiring stronger perturbations to be effective. In contrast, Swin Transformers' self-attention mechanism enables perturbations to diffuse across multiple spatial locations, making them more efficient at fooling the model with fewer updates. This results in better imperceptibility while maintaining high attack success rates.

7.3 Evading State-Of-The-Art Defense Methods

We evaluated our attack method against state-of-the-art defense mechanisms specifically designed to counter adversarial patch attacks. As outlined in Section **??**, we considered two primary categories of defense strategies: approaches that focus on detecting adversarial patches by identifying high-saliency regions: Jedi by Tarchoun et al. (53), Jujutsu by Chen et al. (7), SAC by Liu et al. (31), and DW by Hayes(15) and methods that employ adversarial purification techniques to mitigate the impact of adversarial perturbations: DIFFENDER by Kang et al. (21), and DiffPAD by Fu et al. (13). Operating under a white-box setting, we assume that the attacker has access to the underlying model parameters to launch the patch attack. We focus on ImageNet in this task and employ ResNet50 as the target model. Since patch defenses typically include a patch detection module as an initial step, we generate the adversarial samples as a priori on the target model for the target class "toaster". That essentially means, the perturbations were made only considering the target model and not the defense block in the pipeline. We define an attack as successful if the adversarial sample, after going through the defense module, induces the target model to correctly classify it as the target class or as defined by the defense method. Table 7.6 demonstrates the effectiveness of different patch attacks in the presence of defenses. In all the scenarios, our generated adversarial samples can bypass the defense module effectively with high attack success rates. In stark contrast, all the other attacks are ineffective against at least one of the evaluated defenses. For instance, our method can achieve 100% attack success rate against SAC, whereas the best performance attained by all the remaining patch attacks is as low as 11.6%.

A common observation across all defense methods was that, due to the lack of saliency in the adversarial patch, some methods mistakenly identified non-adversarial features of the image as the primary region of interest. This issue was prevalent in both categories of defense strategies discussed earlier—those that focus on high-saliency regions for patch detection and those that employ adversarial purification techniques.

This phenomenon highlights a fundamental limitation of existing defense mechanisms: they lack a deeper understanding of the intrinsic properties of adversarial patches and the underlying reasons for their effectiveness in inducing misclassification. Instead of comprehending the adversarial perturbation's role in manipulating the model's decision boundaries, these defenses rely on heuristic cues such as the presence of highly textured regions or abrupt visual artifacts. This limitation is evident in both image-processing-based defense techniques, which, for instance, searches for high variance locations to counter adversarial effects, and learning-based approaches, which train models to recognize and mitigate adversarial patterns. In both cases, the reliance on superficial visual features rather than a deeper semantic understanding of adversarial behavior limits the robustness of these defenses. Consequently, adversarial patches that effectively blend into natural image features remain undetected, reducing the overall efficacy of these defense mechanisms.

7.4 Adaptation into the Real-World Scenarios

Although our central goal is to generate a general method for creating imperceptible patches, which is validated by the results we obtained, we also evaluated our method on more realistic scenarios.

Table 7.5: Comparisons of ASR (%) between different attack methods against various patch defenses.

Method	Jedi	Jujutsu	SAC	DW	DIFFender	DiffPAD
Google Patch	46.8	0.0	2.7	1.4	35.5	33.2
LAVAN	50.9	0.3	3.8	54.0	53.2	39.8
GDPA	67.1	94.0	7.4	1.3	57.0	52.1
MPGD ($l_\infty, \epsilon = 16/255$)	68.2	95.1	11.6	79.0	95.7	92.1
Ours	**78.6**	**99.8**	**100**	**89.8**	**99.8**	**98.6**

Table 7.6: Transferability represented by ASR(%) on ImageNet. The first row represents the substitute model and the first column represents the target models.

	ResNet-50	VGG16	Swin T(Tiny)	Swin T(Base)
ResNet-50	**100**	46.2	43.4	43.6
VGG16	63.0	**100**	58.2	56.8
Swin T(Tiny)	16.7	15.3	**100**	21.1
Swin T(Base)	13.2	9.90	12.7	**100**
ResNet-18	60.7	55.9	53.2	55.4
ResNet-34	49.4	44.8	45.2	43.4
VGG11	**70.1**	**72.4**	**68.4**	**68.2**
VGG13	65.5	70.5	61.6	63.1

7.4.1 Evaluation of Attack Transferability in a Black-Box Setting

In this section, we assess the transferability of adversarial perturbations across different neural network architectures in a black-box setting. Specifically, we examine whether adversarial patches crafted for one model can effectively mislead other architectures without direct access to their parameters. We considered the untargeted setting for this set of experiments.

To conduct this evaluation, we first analyze the transferability within the four primary architectures used for attack generation: ResNet-50, VGG-16, Swin-Tiny, and Swin-Base. We then extend our evaluation to additional related architectures—ResNet-18, ResNet-34, VGG-11, and VGG-13—to measure how well perturbations generalize across similar model families. It is important to note that these additional models are used solely for evaluation purposes; adversarial perturbations were not explicitly crafted for them.The experimental setup remains consistent with previous evaluations, ensuring a fair comparison. Specifically, we maintain a fixed set of 1,000 test samples across all experiments. Our results indicate a reasonable degree of transferability, with VGG-11 and VGG-13 exhibiting the highest susceptibility to adversarial perturbations generated by different training setups. This observation suggests that certain model architectures may be inherently more vulnerable to transferred attacks. Although our study does not employ a dedicated methodology tailored for maximizing transferability, the observed results provide valuable insights into the potential of adversarial patch attacks to generalize across architectures.

7.4.2 Evaluation of Physical-World Attack Using Proposed Method

To assess the real-world applicability of our proposed attack method, we extended our experiments to a physical attack scenario. Given that printability constraints play a crucial role in transferring adversarial patches from a digital to a physical setting, we adopted a reference sticker-based approach similar to the methodology introduced by Liu et al. (30). Specifically, we curated a dataset named **SaarSticker**, comprising 100 images of stickers commonly found near traffic signals in Saarbrücken. A subset of these images was used as reference backgrounds onto which perturbations were optimized. Representative samples from this dataset are shown in Figure 7.6.

For training these patches, we largely adhered to the experimental configuration used in our instance-based untargeted attack scenario, with minor modifications to better reflect real-world physical conditions. To enhance the robustness of our patches against real-world variations, we incorporated a set of standard augmentation techniques. These included geometric transformations such as random cropping, flipping, rotation, and perspective distortion, as well as color adjustments involving changes in brightness, contrast, saturation, and hue, along with grayscale conversions. These augmentations were applied to the input images after the adversarial patch was overlaid, mimicking the variations encountered in real-world settings such as lighting conditions, viewing angles, and environmental disturbances.

For our evaluation, we selected the Swin Transformer Tiny as the victim model in a white-box setting. Once the attack was successfully crafted in the digital environment, we printed ten randomly chosen benign images alongside their corresponding adversarial patches. The printed patches were then physically placed on the respective images, and we manually assessed whether the attack objective was met. Our evaluation demonstrated a satisfactory physical attack success rate of 60%, confirming the feasibility of our method in real-world scenarios. Figure **??** illustrates the benign sample, the adversarial sample in digital space and the adversarial sample in the physical space.

Figure 7.5: Examples from the proposed SaarStricker dataset showcasing a variety of stickers present across the traffic signals of Saarbrücken city.

Figure 7.6: x represent the benign sample, $\hat{x}_{dig}$ represent the adversarial sample in the digital space and $\hat{x}_{phy}$ represents the printed adversarial sample.

Chapter 8
Extensive Evaluation and Comparison: Experimental Results on the VGG Face

8.1 Experimental Details

To evaluate the proposed method in a real-world application, we conducted experiments on a face recognition task. Specifically, we utilized a subset of the VGG Face dataset comprising 10 classes, each corresponding to a distinct celebrity: "A. J. Buckley", "A. R. Rahman", "Aamir Khan", "Aaron Staton", "Aaron Tveit", "Aaron Yoo", "Abbie Cornish", "Abel Ferrara", "Abigail Breslin", and "Abigail Spencer". Like the previous two datasets, we set the adversarial patch size to 84 × 84 pixels, covering approximately 14% of the total image area. The comparison to the baseline settings and their configurations are kept same as described in the Section 7.

Our experimental setup consisted of three sets of trials, with each set comprising four attack runs against four victim models. The primary variable across these sets was the designated target class. The target classes chosen for the three sets of experiments were "A. J. Buckley", "Aamir Khan", and "Aaron Staton", which were randomly selected at the outset of the study.

8.2 Results and Discussions

Using VGG16 as the victim model, we achieved a comparable attack success rate across all target classes. Notably, we obtained a 100% attack success rate for the class "A. J. Buckley", which is either equivalent to or superior to existing methods. This was followed by a 98.8% success rate for "Aamir Khan", which, while not outperforming all other methods, remains on par with them. Additionally, for "Aaron Staton", we achieved a 99.53% success rate, which is comparable to the performance of LaVAN and Google Patch, while surpassing other considered methods. In terms of imperceptibility, our

approach achieves state-of-the-art performance across all evaluated methods.

With ResNet-50 as the target model, following the previous trends our method demonstrated consistently high attack success rates across all classes. We achieved 98.8% success on "A. J. Buckley", which performed comparably with existing techniques. An attack success rate of 93.0% in "Aamir Khan" was outperformed by other state-of-the-art competitive methods except for MPGD which scored 70.74%, and same was observed for "Aaron Staton", where we reached 91.80%, exceeded by LaVAN and GDPA while surpassing others. Google Patch observed a drop to 80.3% with MPGD marking the lowest attack success rate at 52.50%. Compared to other approaches our approach excels in imperceptibility, setting a new benchmark across all evaluated methods.

When using Swin Transformer Tiny as the victim model, our attack achieved impressive results as we recorded a 99.3% success rate for "A. J. Buckley", maintaining superiority to other methods only getting surpassed by LaVAN. For "Aamir Khan", we similarly attained 99.3%, being on par with the performance of other leading techniques. In the case of "Aaron Staton", our method reached 98.6% which again was comparable to the other existing approaches while surpassing the imperceptibility prioritizing approaches. Furthermore, our method continued to set the standard for imperceptibility, outperforming all other evaluated methods.

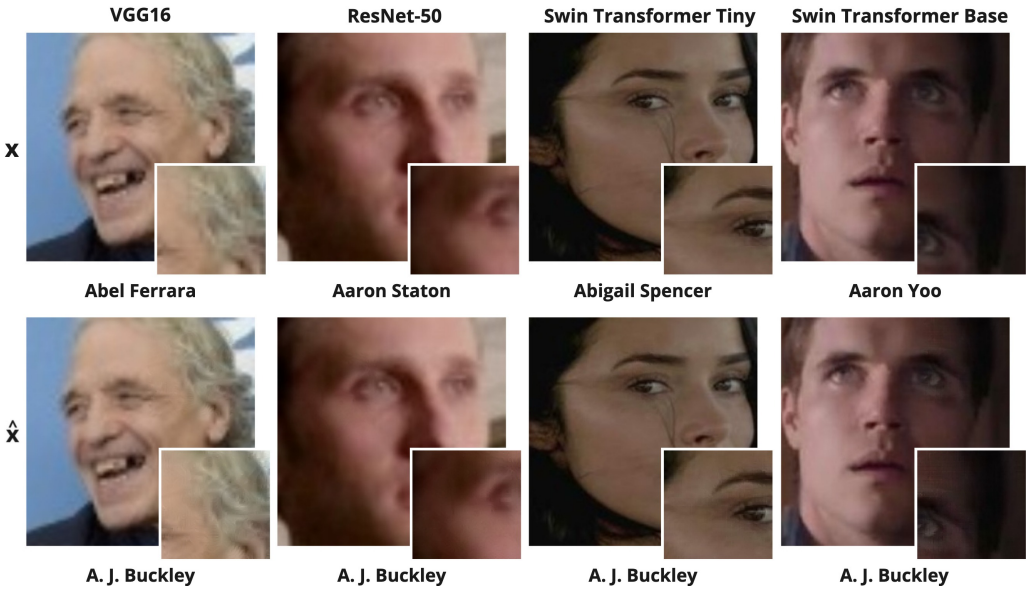

Figure 8.1: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class "A. J. Buckley" on the VGG Face Dataset. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

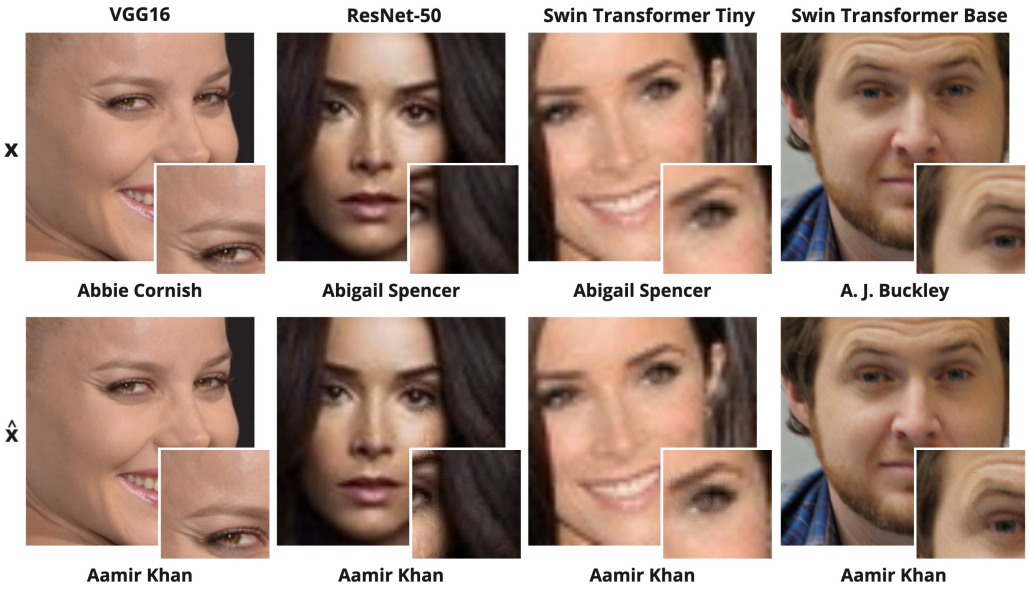

Figure 8.2: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class "Aamir Khan" on the VGG Face Dataset. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

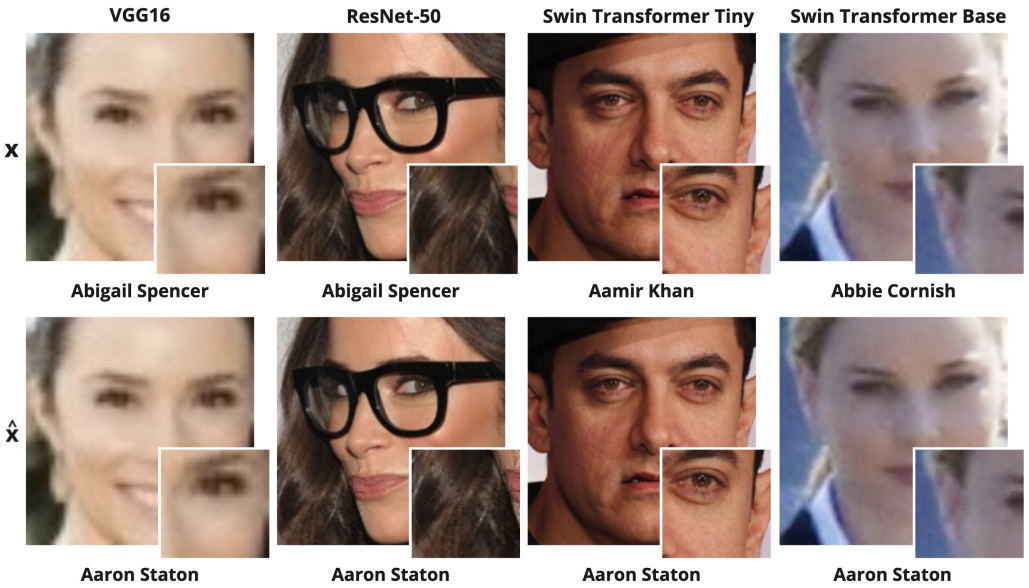

Figure 8.3: Visualizations of the original images and their adversarial counterparts produced by our method corresponding to the target class "Aaron Staton" on the VGG Face Dataset. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Table 8.1: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **VGG16** as the victim model on the VGG Face dataset for the Target class **"A. J. Buckley"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	**100**	Local	0.000	0.000	11.95	36.82	0.890
		Global	0.812	0.820	19.46	68.22	0.270
LaVAN	**100**	Local	0.006	0.000	15.85	36.55	0.865
		Global	0.820	0.825	24.18	71.84	0.220
GDPA	96.12	Local	0.240	0.220	21.00	57.96	0.660
		Global	0.870	0.865	29.00	75.66	0.151
MPGD ($l_\infty, \epsilon = 16/255$)	88.9	Local	0.620	0.533	28.30	65.30	0.400
		Global	0.960	0.935	36.70	86.70	0.087
Ours	**100**	Local	**0.930**	**0.880**	**31.81**	**66.50**	**0.207**
		Global	**0.990**	**0.980**	**40.11**	**88.57**	**0.039**

Employing Swin Transformer Base as the victim model, our attack demonstrated consistently strong performance. We achieved 99.0% success on "A. J. Buckley", surpassing most of the existing methods except for LaVAN. For "Aamir Khan", our approach observed a attack success rate of 97.0%, remaining competitive with leading techniques. Similarly, we obtained 98.6% success on Aaron Staton, exceeding most baselines except for LaVAN. Following previous observations, even for this set of experiments, our attack set a new benchmark in imperceptibility, outperforming all evaluated approaches.

Consistent with observations from ImageNet, existing methods that prioritize imperceptibility often experience a trade-off, where improved invisibility comes at the cost of reduced attack success rates. However, our method successfully maintains a balance in this trade-off, maintaining high imperceptibility while achieving strong and stable attack success rates. The stability in terms of attackability also further validates our method.

Similar to the set of experiments conducted on ImageNet where we observed an ease of attack on transformer-based architectures compared to CNN-based architectures, the same characteristics were observed. The attack success rates for the transformer based architectures were elevated compared to the later, where on an average the attack success rates for Swin Transformer Tiny and Swin Transformer Base was 98.63% comapred to the average value of 97% attack success rate for VGG16 and ResNet-50 together, with ResNet-50 being the most robust architecture.

Table 8.2: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **VGG16** as the victim model on the VGG Face dataset for the Target class **"Aamir Khan"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	**99.9**	Local	0.000	0.000	11.76	36.43	0.860
		Global	0.810	0.820	19.36	68.22	0.270
LaVAN	99.5	Local	0.005	0.000	15.64	36.52	0.850
		Global	0.820	0.825	24.06	71.56	0.220
GDPA	99.50	Local	0.220	0.190	21.46	55.50	0.685
		Global	0.850	0.840	55.50	63.41	0.190
MPGD ($l_\infty, \epsilon = 16/255$)	86.85	Local	0.650	0.550	27.80	65.20	0.420
		Global	0.950	0.930	36.10	86.60	0.090
Ours	**98.8**	Local	**0.924**	**0.870**	**31.94**	**68.24**	**0.200**
		Global	**0.990**	**0.980**	**40.08**	**88.70**	**0.039**

Table 8.3: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **VGG16** as the victim model on the VGG Face dataset for the Target class **"Aaron Staton"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	**100**	Local	0.000	0.000	10.76	36.65	0.860
		Global	0.810	0.820	18.27	68.70	0.290
LaVAN	**100**	Local	0.003	0.000	11.89	36.45	0.870
		Global	0.820	0.824	20.30	71.67	0.260
GDPA	91.50	Local	0.476	0.465	22.85	60.48	0.53
		Global	0.900	0.890	29.45	76.00	0.125
MPGD ($l_\infty, \epsilon = 16/255$)	84.95	Local	0.680	0.564	27.30	65.10	0.440
		Global	0.940	0.924	35.88	85.10	0.094
Ours	**99.53**	Local	**0.904**	**0.850**	**31.40**	**65.80**	**0.217**
		Global	**0.985**	**0.980**	**39.61**	**87.72**	**0.042**

Table 8.4: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **ResNet-50** as the victim model on the VGG Face dataset for the Target class **"A. J. Buckley"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	98.0	Local	0.010	0.000	17.52	38.81	0.730
		Global	0.830	0.820	24.25	63.13	0.210
LaVAN	**100**	Local	0.007	0.000	16.80	36.81	0.840
		Global	0.840	0.826	25.12	71.64	0.200
GDPA	99.5	Local	0.310	0.250	22.00	53.00	0.660
		Global	0.880	0.860	29.00	59.00	0.170
MPGD ($l_\infty, \epsilon = 16/255$)	78.1	Local	0.620	0.560	26.99	61.78	0.380
		Global	0.950	0.930	35.56	85.42	0.080
Ours	**98.8**	Local	**0.920**	**0.880**	**32.11**	**69.40**	**0.170**
		Global	**0.990**	**0.980**	**40.66**	**90.55**	**0.030**

Table 8.5: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **ResNet-50** as the victim model on the VGG Face dataset for the Target class **"Aamir Khan"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	99.5	Local	0.001	0.000	16.47	38.80	0.800
		Global	0.830	0.820	21.89	63.13	0.270
LaVAN	**100**	Local	0.007	0.000	16.89	36.82	0.830
		Global	0.840	0.826	25.30	71.51	0.210
GDPA	99.70	Local	0.280	0.230	21.99	56.73	0.600
		Global	0.870	0.850	56.73	59.32	0.200
MPGD ($l_\infty, \epsilon = 16/255$)	70.74	Local	0.610	0.550	26.60	59.87	0.390
		Global	0.940	0.930	35.30	84.63	0.080
Ours	**93.0**	Local	**0.890**	**0.830**	**30.88**	**65.75**	**0.226**
		Global	**0.980**	**0.970**	**39.37**	**87.30**	**0.040**

Table 8.6: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **ResNet-50** as the victim model on the VGG Face dataset for the Target class **"Aaron Staton"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	80.3	Local	0.010	0.000	17.52	38.81	0.730
		Global	0.830	0.820	24.25	63.13	0.210
LaVAN	**97.0**	Local	0.010	0.000	17.45	41.54	0.750
		Global	0.830	0.820	22.32	62.68	0.240
GDPA	98.00	Local	0.330	0.280	22.10	55.68	0.60
		Global	0.880	0.850	29.12	57.54	0.200
MPGD ($l_\infty, \epsilon = 16/255$)	52.50	Local	0.610	0.550	26.83	60.25	0.380
		Global	0.940	0.930	35.25	83.42	0.080
Ours	**91.80**	Local	**0.890**	**0.840**	**30.89**	**65.70**	**0.216**
		Global	**0.980**	**0.970**	**39.33**	**88.32**	**0.040**

Table 8.7: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Tiny** as the victim model on the VGG Face dataset for the Target class **"A. J. Buckley"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	98.9	Local	0.040	0.000	10.12	36.10	0.820
		Global	0.830	0.830	16.87	66.88	0.260
LaVAN	**100**	Local	0.007	0.000	16.49	36.50	0.850
		Global	0.840	0.825	24.75	71.87	0.210
GDPA	92.9	Local	0.330	0.270	21.85	62.10	0.570
		Global	0.880	0.870	29.30	71.76	0.140
MPGD ($l_\infty, \epsilon = 16/255$)	95.5	Local	0.630	0.540	27.65	62.48	0.380
		Global	0.950	0.930	35.72	86.66	0.070
Ours	**99.3**	Local	**0.860**	**0.800**	**29.22**	**63.28**	**0.275**
		Global	**0.980**	**0.970**	**38.00**	**87.83**	**0.048**

Table 8.8: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Tiny** as the victim model on the VGG Face dataset for the Target class **"Aamir Khan"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	99.2	Local	0.000	0.000	10.11	37.23	0.780
		Global	0.830	0.820	17.22	67.50	0.230
LaVAN	**100**	Local	0.006	0.000	16.31	36.57	0.850
		Global	0.840	0.825	24.71	71.73	0.210
GDPA	100	Local	0.340	0.300	19.85	60.84	0.600
		Global	0.910	0.910	29.82	80.01	0.100
MPGD ($l_\infty, \epsilon = 16/255$)	94.87	Local	0.640	0.550	27.68	62.69	0.370
		Global	0.950	0.930	35.80	86.97	0.070
Ours	**99.3**	Local	**0.870**	**0.820**	**29.80**	**63.00**	**0.240**
		Global	**0.980**	**0.970**	**38.60**	**88.20**	**0.043**

Table 8.9: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Tiny** as the victim model on the VGG Face dataset for the Target class **"Aaron Staton"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	99.3	Local	0.000	0.000	12.60	38.84	0.820
		Global	0.830	0.820	17.48	63.13	0.290
LaVAN	**100**	Local	0.007	0.000	16.45	36.67	0.850
		Global	0.840	0.825	24.82	72.06	0.210
GDPA	92.4	Local	0.310	0.260	20.19	54.86	0.65
		Global	0.860	0.840	27.21	60.54	0.220
MPGD ($l_\infty, \epsilon = 16/255$)	96.2	Local	0.640	0.550	27.76	61.90	0.360
		Global	0.950	0.930	35.85	87.30	0.070
Ours	**98.60**	Local	**0.860**	**0.800**	**29.64**	**62.00**	**0.260**
		Global	**0.980**	**0.970**	**38.34**	**87.90**	**0.046**

Table 8.10: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the VGG Face dataset for the Target class **"A. J. Buckley"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	98.2	Local	0.000	0.000	11.23	36.51	0.835
		Global	0.830	0.820	18.23	67.65	0.240
LaVAN	**100**	Local	0.005	0.000	15.47	36.52	0.850
		Global	0.840	0.825	23.80	71.82	0.220
GDPA	77.24	Local	0.410	0.360	21.59	58.14	0.56
		Global	0.910	0.900	29.66	72.23	0.110
MPGD ($l_\infty, \epsilon = 16/255$)	97.9	Local	0.600	0.520	27.45	61.22	0.390
		Global	0.940	0.920	35.57	85.00	0.080
Ours	**99.0**	Local	**0.860**	**0.780**	**29.8**	**63.00**	**0.300**
		Global	**0.980**	**0.960**	**38.10**	**86.00**	**0.055**

Table 8.11: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the VGG Face dataset for the Target class **"Aamir Khan"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
Google Patch	97.2	Local	0.000	0.000	10.78	36.85	0.900
		Global	0.830	0.820	18.10	69.36	0.260
LaVAN	**99.3**	Local	0.004	0.000	15.00	36.49	0.850
		Global	0.840	0.824	23.40	71.68	0.220
GDPA	55.1	Local	0.160	0.140	18.36	65.87	0.700
		Global	0.920	0.920	30.10	84.10	0.090
MPGD ($l_\infty, \epsilon = 16/255$)	80.81	Local	0.610	0.530	27.35	61.22	0.390
		Global	0.940	0.930	35.47	85.28	0.080
Ours	**97.0**	Local	**0.840**	**0.760**	**29.63**	**61.00**	**0.300**
		Global	**0.970**	**0.960**	**37.84**	**86.00**	**0.055**

Table 8.12: Detailed evaluation and comparison of attack efficacy through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the VGG Face dataset for the Target class **"Aaron Staton"**. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Method	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE** (↑)	**CLIP** (↑)	**LPIPS** (↓)
Google Patch	98.3	Local	0.000	0.000	11.86	35.58	0.920
		Global	0.830	0.820	17.78	68.63	0.280
LaVAN	**99.8**	Local	0.006	0.000	15.73	36.70	0.850
		Global	0.840	0.825	24.12	71.96	0.210
GDPA	84.9	Local	0.290	0.260	19.76	57.25	0.620
		Global	0.910	0.910	29.72	78.20	0.100
MPGD ($l_\infty, \epsilon = 16/255$)	94.9	Local	0.630	0.550	27.75	61.50	0.370
		Global	0.950	0.930	35.84	86.54	0.070
Ours	**98.6**	Local	**0.880**	**0.810**	**30.50**	**63.50**	**0.250**
		Global	**0.980**	**0.970**	**38.84**	**88.40**	**0.045**

Chapter 9
Ablation Studies

We conducted a series of ablation studies to systematically analyze the key components of our method and their individual contributions to overall performance. Specifically, we examine elements that exhibit strong correlations with our findings. First, we investigate the effect of patch size, followed by an analysis of the number of update iterations and the regularization coefficient associated with the distance term in the loss function, as defined in Equation (4.10).

Additionally, we assess the impact of the update rule employed for perturbation optimization by comparing our approach with a widely used update strategy from the Adam Optimizer. Lastly, we evaluate a commonly held assumption about adversarial patches, namely that they predominantly attract the classifier's attention, thereby inducing misclassification. To test this, we measured the proportion of generated adversarial samples in which the highest attention region, as identified by GradCAM, overlaps with the attack location. This assessment is particularly significant, as the attention map's localization of adversarial perturbations can serve as an indicator for their detection. For all our experiments we used ImageNet as the dataset and Swin Transformer Base as the victim model.

9.1 Effect of Patch Size on the Attack

From the existing literature, it is clear that patch size significantly impacts attack efficacy as increasing the patch size leads to stronger attack. We wanted to evaluate the same for our method along with its impact on imperceptibility. We hypothesized that increasing the attack area will lead to stronger attack performance as well as stronger imperceptibility performance as perturbations have more area to disperse into while not being salient. The results validated our theory as with the increase in the patch size increased the attack efficacy as a 99.4% ASR was achieved with a patch size that covered 14% of the total image compared to 22.1% ASR for 2% coverage. It is worth noting that for patch coverage equal to or exceeding 8%, our success rates attain 92.5% or higher. The detailed results are summarized in table 9.1. A similar trend is observed where the imperceptibility of the adversarial sample achieves more desirable values as the patch

coverage increases, as visualized in 9.1.

9.2 Effect of Number of Update Iterations on the Attack

As the number of updates to the patch increases, its appearance diverges further from the original state, even if the updates do not result in highly salient features. Based on the update rule employed in our methodology, we argue that while the perturbations remain less salient, the original appearance of the location may undergo significant alterations with a higher number of iterations. However, it is important to note that more iterations typically lead to higher attack success rates. In this set of experiments, we fixed the patch size at 6% to emphasize the impact of both attack efficacy and imperceptibility. The detailed results are summarized in table 9.2, We observe an increase in attack success rates as the number of iterations grows. While there is a slight decrease in imperceptibility, the reduction is not drastic, suggesting that the method maintains stability even with higher iteration counts.

9.3 Effect of Distance Term Regularization Coefficient on the Attack

We aimed to investigate the effect of the regularization coefficient w_3 associated with the human-oriented distance metric (Equation 4.7), which is included in the total loss function (Equation 4.10). We hypothesized that increasing the value of w_3 would improve imperceptibility at the cost of slightly reducing attack performance. Our observations, as shown in Table 9.3, support this hypothesis to some extent. Initially, as w_3 increases, the attack success rate decreases slightly while imperceptibility improves. However, as we continue to increase w_3 , the trend reverses. We attribute this behavior to the destabilizing effect of a large w_3 value on the overall loss function. This causes the loss to become dominated by the regularization term, requiring more iterations for the attack to succeed. Although this results in successful attacks, it leads to a reduction in imperceptibility performance.

9.4 Effect of Update Rule on the Attack

As outlined in the methodology, the update rule proposed in our work allows for longer update iterations without any constraints on the perturbation magnitude, while still maintaining high levels of imperceptibility. To conclusively demonstrate its contribution, we compared our proposed update rule with the widely-used update rule from the Adam Optimizer, which is commonly employed in most studies. The Adam Optimizer update rule is highly optimized to minimize the overall loss with a strong emphasis on attack success, leading us to hypothesize that it would result in a higher attack success rate. However, due to the inherent nature of Adam's updates, which modify each color channel separately, the updates can alter the base color of the pixel. In contrast, our method preserves the base color of the pixel. The results, as shown in Table 9.4, validate our approach. While following Adam Optimizer's update rule led to a slightly higher success rate, our method outperformed Adam Optimizer in terms of imperceptibility. Figure 9.3 illustrates the adversarial patches generated by both approaches.

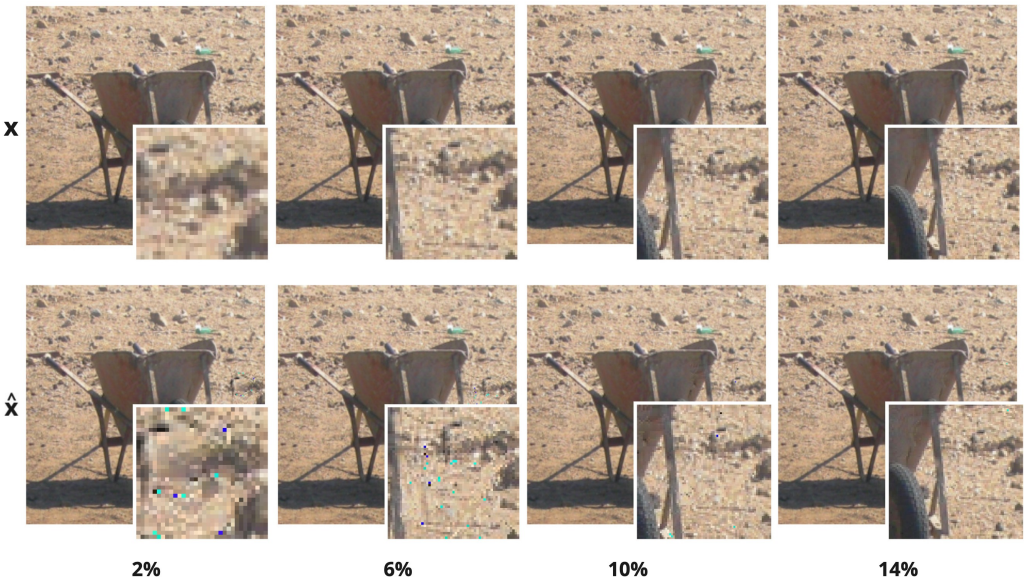

Figure 9.1: Visualizations of the impact of the patch sizes on attack imperceptibility. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

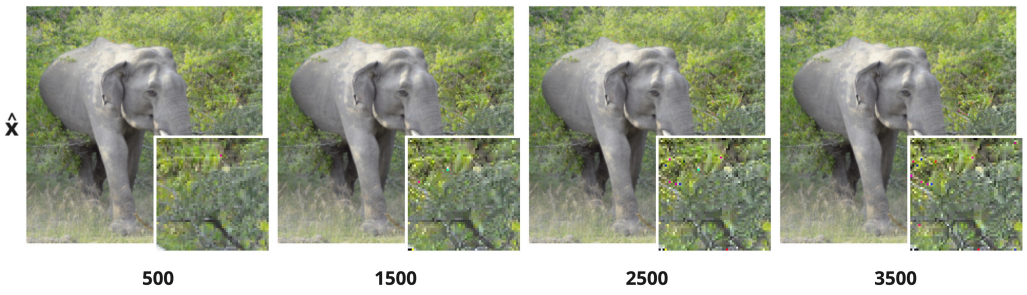

Figure 9.2: Visualizations of the impact of the number of update iteration on attack imperceptibility. $\hat{x}$ represent the adversarial samples with the generated adversarial patch. The smaller images at the right-bottom corner correspond to the optimal location (i', j'). x axis represents the number of update iteration.

Table 9.1: Impact of **patch size** on attack performance represented through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Patch Size(%)	ASR(%)	Scale	Imperceptibility metric				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
2	22.1	Local	0.640	0.530	21.07	70.00	0.413
		Global	0.992	0.985	38.10	98.20	0.014
4	46.8	Local	0.784	0.683	23.32	70.77	0.308
		Global	0.991	0.972	37.68	98.11	0.014
6	70.0	Local	0.854	0.756	25.02	74.74	0.024
		Global	0.991	0.970	37.86	98.18	0.013
8	92.5	Local	0.896	0.810	26.77	78.05	0.183
		Global	0.991	0.970	38.14	98.15	0.012
10	98.1	Local	0.920	0.840	27.90	80.91	0.152
		Global	0.992	0.970	38.31	97.97	0.011
12	99.0	Local	0.934	0.860	28.90	83.43	0.126
		Global	0.992	0.965	38.46	98.03	0.011
14	**99.4**	Local	**0.970**	**0.910**	**31.30**	**89.33**	**0.070**
		Global	**0.994**	**0.970**	**40.10**	**98.43**	**0.010**

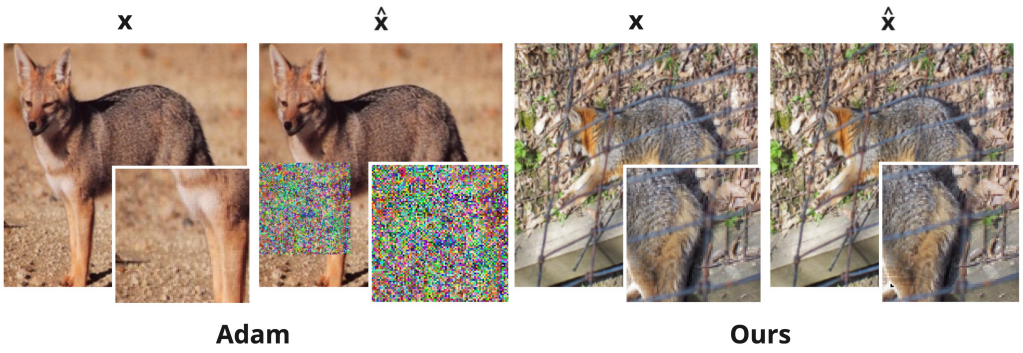

Figure 9.3: Visualizations of adversarial patch generated by update rule from Adam optimizer vs ours. x represent the benign sample's original class and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. The smaller images at the right-bottom corner correspond to the optimal location (i', j').

Table 9.2: Impact of **number of update iterations** on attack performance, represented through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS. Patch size is kept fixed at 6%

No. Iters	ASR(%)	Scale	Imperceptibility metric				
			SSIM (↑)	**UIQ (↑)**	**SRE (↑)**	**CLIP (↑)**	**LPIPS (↓)**
500	86.0	Local	**0.870**	**0.770**	**25.88**	**75.87**	**0.223**
		Global	**0.992**	**0.972**	**38.48**	**98.23**	**0.015**
1000	70.0	Local	0.854	0.756	25.02	74.74	0.024
		Global	0.991	0.970	37.86	98.18	0.013
1500	96.2	Local	0.850	0.755	24.91	73.81	0.024
		Global	0.991	0.969	37.70	98.10	0.016
2000	97.3	Local	0.843	0.749	24.77	73.29	0.246
		Global	0.990	0.968	37.59	98.04	0.017
2500	98.0	Local	0.840	0.746	24.67	72.87	0.249
		Global	0.990	0.967	37.50	98.02	0.017
3000	98.5	Local	0.836	0.743	24.48	72.78	0.252
		Global	0.990	0.966	37.41	97.98	0.017
3500	**98.6**	Local	0.834	0.741	24.65	72.49	0.254
		Global	0.990	0.969	37.40	97.92	0.017

9.5 GradCAM analysis of attention overlap with patch location.

Initially stated by Brown et al. (4), the adversarial patches are significantly salient and consequently draws all the attention of the classifiers and hence results in misclassification. This was investigated by Karmon et al. (22) where they found it inconsistent with their work. Hence considering the patches generated by our method which are not salient we wanted to explore whether the explanation by Brown et al. holds true for our work. We evaluated the attention maps generated by GradCAM corresponding to the adversarial samples generated by our method. We argued that if the location with maximum attention on the attention map lies within the attack area then the argument by Brown et al. holds true. Hence we measured the total proportion of adversarial samples for which this overlap holds for each of the victim models considered. From the results as presented on Table 9.5 we can say that on an average for almost 71% of the adversarial samples the highest attention point determined by GradCAM do not lie on attack surface. Visualisation of this behavior is represented in Figure 9.4. Considering many defense methods analyses the attention maps of these models corresponding to the adversarial samples for detection, this insights is primarily important as it solidifies the stealthy nature of our method.

Table 9.3: Impact of distance term regularization coefficient $\mathbf{w}_3$ on attack performance represented through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

$\mathbf{w}_3$	ASR(%)	Scale	Imperceptibility metric				
			SSIM (↑)	UIQ (↑)	SRE (↑)	CLIP (↑)	LPIPS (↓)
0	99.0	Local	0.943	0.873	29.76	85.54	0.111
		Global	0.992	0.964	38.59	97.95	0.017
1	98.9	Local	0.944	0.874	29.79	85.65	0.110
		Global	0.992	0.965	38.62	97.97	0.017
4	98.9	Local	0.945	0.875	29.83	85.66	0.109
		Global	0.992	0.965	38.67	97.99	0.017
7	98.8	Local	0.946	0.876	29.84	85.68	0.108
		Global	0.992	0.966	38.69	98.01	0.016
10	99.1	Local	0.945	0.875	29.83	85.71	0.108
		Global	0.992	0.965	38.66	97.98	0.017
13	99.0	Local	0.944	0.874	29.78	85.56	0.110
		Global	0.992	0.965	38.60	97.98	0.017

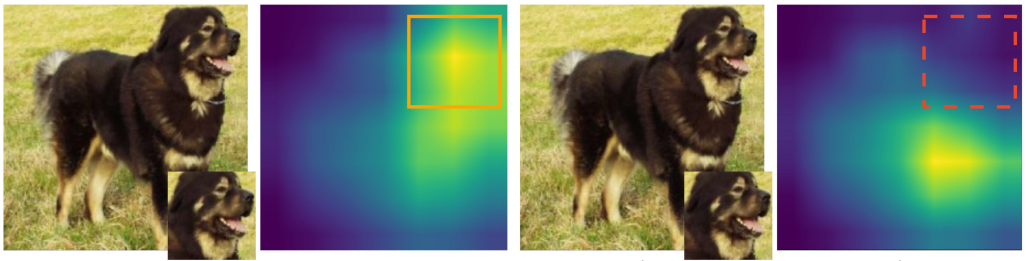

Figure 9.4: Visualization of the shift in high attention location from the original benign sample compared to that of the adversarial sample. x represent the benign sample and $\hat{x}$ represent the adversarial samples with the generated adversarial patch corresponding to the target class. x_{at} and $\hat{x}_{at}$ represents the attention map generated corresponding to the original benign sample and the adversarial sample. the red square on $\hat{x}_{at}$ represent the attack location.

Table 9.4: Impact of the **update rule** on attack performance represented through ASR (%) and imperceptibility with **Swin Transformer Base** as the victim model on the ImageNet dataset. For SSIM, UIQ, SRE and CLIP scores, the higher (↑) the better, while the lower (↓) the better for LPIPS.

Update Rule	**ASR(%)**	**Scale**	**Imperceptibility metric**				
			SSIM (↑)	**UIQ (↑)**	**SRE** (↑)	**CLIP** (↑)	**LPIPS** (↓)
Adam	100	Local	0.130	0.157	17.13	36.15	0.662
		Global	0.867	0.848	25.98	80.94	0.130
Ours	**99.4**	Local	**0.970**	**0.910**	**31.30**	**89.33**	**0.070**
		Global	**0.994**	**0.970**	**40.10**	**98.43**	**0.010**

Table 9.5: Assessment of whether the GradCAM's highest attention location overlaps with the adversarial patch location.

Model	**NoPatchLoc(%)**
VGG16	72.15
ResNet-50	53.70
Swin Transformer Tiny	73.11
Swin Transformer Base	81.30
Average	**70.7**

Chapter 10
Discussion and Future Directions

Through this work we made an attempt to demonstrate that adversarial patch attacks can be designed to achieve both high attack effectiveness and strong imperceptibility. Our approach ensures that the generated adversarial patches are not only visually inconspicuous to human observers but also remain undetectable by state-of-the-art defense mechanisms designed to counter such attacks.

To achieve this, we proposed a general attack pipeline that prioritizes high targeted attack success rates while maintaining a high degree of imperceptibility. This design choice ensures that the underlying principles of our method can be effectively translated into real-world applications with slight modification for adaptations. Through extensive evaluations, we validated the effectiveness and stability of our proposed approach across multiple datasets and a diverse range of classifier architectures. The results consistently demonstrated the robustness of our method in maintaining a high attack success rate without compromising stealth.

Furthermore, we evaluated the ability of our attack to bypass state-of-the-art adversarial defense mechanisms. The results of these experiments provide compelling evidence that our approach successfully circumvents advanced defense strategies, further reinforcing the effectiveness of our proposed methodology.

These findings also highlight the pressing need for the development of more robust defense mechanisms that go beyond reliance on human-perceptible visual cues. Traditional defenses often focus on detecting perturbations based on features that are noticeable to the human eye. However, our results suggest that such an approach may be fundamentally insufficient, as machine perception does not always align perfectly with human vision. Hence, features that look visually benign can be in reality be adversarially designed. Adversarial attacks, including our method, exploit these discrepancies by introducing perturbations that may be imperceptible to human observers but significantly impact model predictions. This misalignment underscores the importance of designing defense mechanisms that understand and address the underlying feature representations that contribute to model misclassification. Instead of solely depending on human-intuitive visual patterns, future defenses must integrate a deeper understanding of the internal decision-making processes of machine learning models. Failure to do so leaves exploitable loopholes that adversaries can readily manipulate, ultimately

compromising the security and robustness of deployed models.

A key factor contributing to the success of our method is its ability to subtly manipulate the model's attention, guiding it toward targeted misclassification without explicitly drawing focus to the adversarial patch itself. Unlike conventional adversarial patches, which often attract excessive attention and can be more easily detected, our approach ensures that the perturbations remain inconspicuous while still exerting a significant influence on the classifier's decision-making process. This unique characteristic makes our method not only highly effective but also exceptionally stealthy, setting it apart from existing adversarial attack strategies.

While our study demonstrates the effectiveness of the proposed adversarial patch attack, we acknowledge certain limitations and drawbacks that should be addressed in future research. Identifying and overcoming these challenges will be crucial in further refining adversarial attack methodologies.

One key limitation of our approach is its dependence on relatively larger patch sizes, covering approximately 14% of the total image area. Although the attack success rate remains high even when using smaller patches, this often comes at the cost of reduced imperceptibility. This trade-off suggests that further optimization techniques are required to enhance the stability of the attack, ensuring that imperceptibility is maintained across varying patch sizes. Developing more sophisticated optimization strategies could help mitigate this limitation, making the attack less dependent on patch size while preserving both effectiveness and stealth.

Another limitation arises from the nature of our update rule. While it effectively preserves the base color of pixels by modifying only their brightness and saturation, it remains independent of the surrounding pixel intensities and brightness levels. Consequently, in certain cases, the update rule may generate isolated pixels with extreme brightness or darkness that contrast starkly with their neighboring pixels. This inconsistency in pixel appearance can impact the overall imperceptibility of the adversarial patch, making it more detectable under close examination. To address this, future research should explore methods that incorporate contextual awareness into the update process, ensuring that modifications are not only effective but also blend seamlessly with the surrounding image regions.

By addressing these limitations, future advancements in adversarial patch attacks can achieve greater robustness, making them more adaptable while maintaining high imperceptibility. These improvements will be essential in understanding and mitigating adversarial vulnerabilities in machine learning models.

The growing sophistication of adversarial attacks, as demonstrated in our study, underscores the urgent need for the development of defense mechanisms that go beyond surface-level perturbation detection and instead focus on understanding the underlying reasons behind model vulnerabilities. An effective approach would involve designing robust defense strategies that analyze the internal feature representations contributing to misclassification, thereby addressing the root cause of adversarial susceptibility rather than merely reacting to observable perturbations. Such defenses should not only detect and neutralize adversarial manipulations but also ensure that the classifier's normal performance remains unaffected. Striking this balance is crucial, as overly rigid defense mechanisms can degrade model accuracy on benign inputs, reducing their practical usability. By integrating a deeper understanding of decision-making processes in neural networks and ensuring adaptability across varying adversarial strategies, future defense techniques can provide long-term robustness against evolving attack methodologies while maintaining reliable classification performance in real-world applications.

Chapter 11
Conclusion

In this master's thesis, we developed a general methodology for generating imperceptible adversarial patches that achieve high targeted attack success rates. Instead of approaching the imperceptibility problem from a ℓ_p-norm bounded perspective, our proposed perturbation curation optimization scheme balances targeted attack success while producing updates that is agnostic to human perception, enabling effective misclassifications of state-of-the-art classifiers while remaining visually inconspicuous. Through extensive evaluations across multiple datasets and classifier architectures along with their comparison to the existing attack methods, we demonstrated the robustness and stability of our method, achieving targeted misclassification without drawing model's attention to the patch.

Our findings emphasize the need for more advanced defense mechanisms that move beyond detecting only human-perceptible perturbations. Since machine perception differs significantly from human vision, defenses relying solely on visual cues remain vulnerable. Our experiments demonstrated this by successfully bypassing state-of-the-art defense methods. Therefore, future defense strategies should focus on understanding the underlying feature representations that drive model misclassification, ensuring both robustness against attacks and reliable performance on benign inputs.

Despite its effectiveness, our approach has certain limitations, including its dependence on larger patch sizes and the potential for isolated pixel artifacts. Addressing these through improved optimization techniques and contextual awareness in perturbation updates will further enhance attack imperceptibility and adaptability. This work contributes to the understanding of adversarial robustness and highlights key areas for future research in both attack development and defense strategies.

Bibliography

[1] Anish Athalye, Logan Engstrom, Andrew Ilyas, and Kevin Kwok. 2018. Synthesizing Robust Adversarial Examples. (2018). https://arxiv.org/abs/1707.07397

[2] Tao Bai, Jinqi Luo, and Jun Zhao. 2021. Inconspicuous adversarial patches for fooling image-recognition systems on mobile devices. *IEEE Internet of Things Journal* 9, 12 (2021), 9515–9524.

[3] Leo Breiman. 1996. Bagging predictors. *Machine learning* 24 (1996), 123–140.

[4] Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. 2017. Adversarial patch. *arXiv preprint arXiv:1712.09665* (2017).

[5] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, and others. 2015. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *Proceedings of the IEEE international conference on computer vision*. 2956–2964.

[6] Nicholas Carlini and David Wagner. 2017. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*. Ieee, 39–57.

[7] Zitao Chen, Pritam Dash, and Karthik Pattabiraman. 2022. Jujutsu: A Two-stage Defense against Adversarial Patch Attacks on Deep Neural Networks. (2022).

[8] Francesco Croce and Matthias Hein. 2019. Sparse and imperceivable adversarial attacks. In *Proceedings of the IEEE/CVF international conference on computer vision*. 4724–4732.

[9] Prafulla Dhariwal and Alexander Nichol. 2021. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems* 34 (2021), 8780–8794.

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2020. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv preprint arXiv:2010.11929* (2020).

[11] Michael P Eckert and Andrew P Bradley. 1998. Perceptual quality metrics applied to still image compression. *Signal processing* 70, 3 (1998), 177–200.

[12] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. 2018. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1625–1634.

[13] Jia Fu, Xiao Zhang, Sepideh Pashami, Fatemeh Rahimian, and Anders Holst. 2024. DiffPAD: Denoising Diffusion-based Adversarial Patch Decontamination. *arXiv preprint arXiv:2410.24006* (2024).

[14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[15] Jamie Hayes. 2018. On visible adversarial perturbations & digital watermarking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 1597–1604.

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.

[17] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718* (2021).

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

[19] Hao Huang, Yongtao Wang, Zhaoyu Chen, Zhi Tang, Wenqiang Zhang, and Kai-Kuang Ma. 2021. Rpattack: Refined patch attack on general object detectors. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1–6.

[20] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. 2019. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems* 32 (2019).

[21] Caixin Kang, Yinpeng Dong, Zhengyi Wang, Shouwei Ruan, Hang Su, and Xingxing Wei. 2023. Diffender: Diffusion-based adversarial defense against patch attacks in the physical world. *arXiv preprint arXiv:2306.09124* (2023).

[22] Danny Karmon, Daniel Zoran, and Yoav Goldberg. 2018. Lavan: Localized and visible adversarial noise. In *International Conference on Machine Learning*. PMLR, 2507–2515.

[23] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. 2011. Novel Dataset for Fine-Grained Image Categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*. Colorado Springs, CO.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 25 (2012).

[25] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. 2016. Adversarial machine learning at scale. *arXiv preprint arXiv:1611.01236* (2016).

[26] Charis Lanaras, José Bioucas-Dias, Silvano Galliani, Emmanuel Baltsavias, and Konrad Schindler. 2018. Super-resolution of Sentinel-2 images: Learning a globally applicable deep neural network. *ISPRS Journal of Photogrammetry and Remote Sensing* 146 (2018), 305–319.

[27] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436–444.

[28] Xiang Li and Shihao Ji. 2021. Generative dynamic patch attack. *arXiv preprint arXiv:2111.04266* (2021).

[29] Anmin Liu, Weisi Lin, Manoranjan Paul, Chenwei Deng, and Fan Zhang. 2010. Just noticeable difference for images with decomposition model for separating edge and textured regions. *IEEE Transactions on Circuits and Systems for Video Technology* 20, 11 (2010), 1648–1652.

[30] Aishan Liu, Xianglong Liu, Jiaxin Fan, Yuqing Ma, Anlan Zhang, Huiyuan Xie, and Dacheng Tao. 2019. Perceptual-sensitive gan for generating adversarial patches. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 1028–1035.

[31] Jiang Liu, Alexander Levine, Chun Pong Lau, Rama Chellappa, and Soheil Feizi. 2022. Segment and complete: Defending object detectors against adversarial patch attacks with robust patch detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 14973–14982.

[32] Xin Liu, Huanrui Yang, Ziwei Liu, Linghao Song, Hai Li, and Yiran Chen. 2018. Dpatch: An adversarial patch attack on object detectors. *arXiv preprint arXiv:1806.02299* (2018).

[33] Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. 2016. Delving into transferable adversarial examples and black-box attacks. *arXiv preprint arXiv:1611.02770* (2016).

[34] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 10012–10022.

[35] Bo Luo, Yannan Liu, Lingxiao Wei, and Qiang Xu. 2018. Towards imperceptible and robust adversarial example attacks against neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[36] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2017. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083* (2017).

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

[38] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. 2022. Diffusion Models for Adversarial Purification. (2022). `https://arxiv.org/abs/2205.07460`

[39] Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. 2017. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*. 506–519.

[40] Omkar M. Parkhi, Andrea Vedaldi, and Andrew Zisserman. 2015. Deep Face Recognition. In *British Machine Vision Conference*.

[41] Yaguan Qian, Jiamin Wang, Bin Wang, Shaoning Zeng, Zhaoquan Gu, Shouling Ji, and Wassim Swaileh. 2020. Visually imperceptible adversarial patch attacks on digital images. *arXiv preprint arXiv:2012.00909* (2020).

[42] 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. 2015. ImageNet Large Scale Visual Recognition Challenge. (2015).

[43] Pouya Samangouei, Maya Kabkab, and Rama Chellappa. 2018. Defense-GAN: Protecting Classifiers Against Adversarial Attacks Using Generative Models. (2018). https://arxiv.org/abs/1805.06605

[44] Robert E Schapire. 1990. The strength of weak learnability. *Machine learning* 5 (1990), 197–227.

[45] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*. 618–626.

[46] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. 2016. Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition. In *Proceedings of the 2016 acm sigsac conference on computer and communications security*. 1528–1540.

[47] Mahmood Sharif, Sruti Bhagavatula, Lujo Bauer, and Michael K Reiter. 2019. A general framework for adversarial examples with objectives. *ACM Transactions on Privacy and Security (TOPS)* 22, 3 (2019), 1–30.

[48] Changhao Shi, Chester Holtz, and Gal Mishne. 2021. Online adversarial purification based on self-supervision. *arXiv preprint arXiv:2101.09387* (2021).

[49] Karen Simonyan and Andrew Zisserman. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556* (2014).

[50] Yang Song, Taesup Kim, Sebastian Nowozin, Stefano Ermon, and Nate Kushman. 2017. Pixeldefend: Leveraging generative models to understand and defend against adversarial examples. *arXiv preprint arXiv:1710.10766* (2017).

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

[52] Mingxing Tan and Quoc Le. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*. PMLR, 6105–6114.

[53] Bilel Tarchoun, Anouar Ben Khalifa, Mohamed Ali Mahjoub, Nael Abu-Ghazaleh, and Ihsen Alouani. 2023. Jedi: entropy-based localization and removal of adversarial patches. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4087–4095.

[54] Alexandru Telea. 2004. An image inpainting technique based on the fast marching method. *Journal of graphics tools* 9, 1 (2004), 23–34.

[55] Simen Thys, Wiebe Van Ranst, and Toon Goedemé. 2019. Fooling automated surveillance cameras: adversarial patches to attack person detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*. 0–0.

[56] Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Ian Goodfellow, Dan Boneh, and Patrick McDaniel. 2017. Ensemble adversarial training: Attacks and defenses. *arXiv preprint arXiv:1705.07204* (2017).

[57] A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems* (2017).

[58] Jinyi Wang, Zhaoyang Lyu, Dahua Lin, Bo Dai, and Hongfei Fu. 2022. Guided diffusion model for adversarial purification. *arXiv preprint arXiv:2205.14969* (2022).

[59] Xiaosen Wang and Kunyu Wang. 2023. Generating Visually Realistic Adversarial Patch. *arXiv preprint arXiv:2312.03030* (2023).

[60] Zhou Wang and Alan C Bovik. 2002. A universal image quality index. *IEEE signal processing letters* 9, 3 (2002), 81–84.

[61] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* 13, 4 (2004), 600–612.

[62] David H Wolpert. 1992. Stacked generalization. *Neural networks* 5, 2 (1992), 241–259.

[63] Shudeng Wu, Tao Dai, and Shu-Tao Xia. 2020. Dpattack: Diffused patch attacks against universal object detection. *arXiv preprint arXiv:2010.11679* (2020).

[64] Chaowei Xiao, Zhongzhu Chen, Kun Jin, Jiongxiao Wang, Weili Nie, Mingyan Liu, Anima Anandkumar, Bo Li, and Dawn Song. 2023. Densepure: Understanding diffusion models for adversarial robustness. In *The Eleventh International Conference on Learning Representations*.

[65] Ke Xu, Yao Xiao, Zhaoheng Zheng, Kaijie Cai, and Ram Nevatia. 2023. Patchzero: Defending against adversarial patch attacks by detecting and zeroing the patch. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 4632–4641.

[66] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. 2015. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579* (2015).

[67] Matthew D Zeiler and Rob Fergus. 2014. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*. Springer, 818–833.

[68] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 586–595.

[69] Alon Zolfi, Moshe Kravchik, Yuval Elovici, and Asaf Shabtai. 2021. The translucent patch: A physical and universal attack on object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15232–15241.