Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

Marius B. Larsen

A Survey on Attacks and Defenses Targeting Region Proposals in Object Detectors

Master's thesis in MTDT
Supervisor: Jingyue Li
Co-supervisor: Nektaria Kaloudi
June 2022

NTNU
Kunnskap for en bedre verden

Marius B. Larsen

A Survey on Attacks and Defenses Targeting Region Proposals in Object Detectors

Master's thesis in MTDT
Supervisor: Jingyue Li
Co-supervisor: Nektaria Kaloudi
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Computer Science

NTNU
Kunnskap for en bedre verden

A Survey on Attacks and Defenses Targeting Region Proposals in Object Detectors

Marius Larsen

2022/06/17

Abstract

Object detectors have established their dominance in several safety-critical systems. Meanwhile, there are an increasing number of research proving they are prone to carefully manufactured perturbations, known as adversarial examples. These examples can disrupt the detectors to misclassify, mislocate or even skip important objects in the input. Any of these disruptions can cause fatal consequences in safety-critical systems, thereby being a highly important topic to research as the usage of object detectors inevitably keeps increasing.

Recently, these attacks have extended to targeting the region proposals modules in object detectors, namely the bounding box regressor. This introduces an important issue, as this module is a commonality across the most widely used detectors. This thesis will look into this issue with the following research questions to answer: What is the state of today's attacks against bounding box regression? How well have these attacks been evaluated? And what defenses are there to detect, mitigate or make the models robust against these attack and their evaluations?

These research questions are answered through a literature review, where the most recent attacks are discussed and compared to find the state-of-the-art attacks. Furthermore, defensive strategies that seek to mitigate these attacks are presented and evaluated. To further landscape this field, and help classify new attacks with the same attributes, a novel taxonomy is proposed.

The research results in this thesis show that the state of today's attacks that targets the bounding box regression achieves great attack performance, while their evaluations indicate that several of these attacks generalizes to transfer across detectors. This important feature of the attacks makes the regressor an important attack surface shared across the most common object detectors. Lastly, there were only three of 16 attacks that have been mitigated, indicating a significant research gap between the state of attacks and defenses for robust region proposals.

Sammendrag

Objekt detektorer har etablert sin dominans i flere sikkerhetskritiske systemer, samtidig er det et økende antall studier som beviser at de er sårbare mot nøye beregnede forstyrrelser. Disse forstyrrelsene kan føre til feilklassifisering, feilplassering eller at viktige objekter unngår deteksjon. Enhver av disse feilene kan forårsake fatale konsekvenser i sikkerhetskritiske systemer, det er dermed et viktig tema å forske på da bruken av objekt detektorer uunngåelig kommer til å fortsette i et økende tempo.

Nylig har disse angrepene blitt utvidet til å fokusere på modulene som foreslår objekt regioner. Ettersom at denne modulen har fellestrekk som er delt blant de mest brukte detektorene, er dette en svært viktig utfordring.

Denne oppgaven vil se nærmere på denne problemstillingen ved å svare på følgende forskningsspørsmål: Hva er tilstanden til dagens angrep mot modulen for region forslag? Hvor godt er disse angrepene blitt evaluert? Og hvilke forsvar eksisterer for minimere effekten til disse angrepen, eller for å gjøre detektorene robuste mot de?

Disse forskningsspørsmålene besvares gjennom et litteraturstudie, der de nyeste angrepene blir diskutert og sammenlignet for å finne de mest lovende angrepene. Videre er defensive strategier som prøver å dempe disse angrepene presentert og vurdert. Deretter foreslås det en taksonomi for å ytterligere beskrive landskapet for dette feltet, og for å bidra til å klassifisere nye angrep med de samme egenskapene.

Forskningsresultatene i denne oppgaven viser at dagens tilstand for angrep mot regions forslags modulen oppnår god angrep ytelse, samt viser deres evalueringer at flere av disse angrepene generaliserer effektivt for overføring på tvers av detektorer. Dette gjør at regions forslags modulen blir en effektiv angrep overflate, som også deles blant de vanligste objekt detektorene. Til slutt var det bare tre av 16 angrep som er blitt dempet, noe som indikerer et betydelig mellomrom i forsknings om tilstanden til angrep og forsvar for robuste regions forslag.

Contents

Figures

Tables

Acronyms

AB Adversarial Border. 24, 29, 30, 49, 50, 61

AE Adversarial Example. 6–10, 20, 23, 24, 26–28, 30–34, 39, 40, 43, 54–56, 62–64, 66

AP Average Precision. 76–78

APS Adversarial Pan-Sharpening. 27, 29, 31, 49, 50, 61

CAP Contextual Adversarial Perturbation. 25, 26, 29, 49–52, 61

CNN Convolutional Neural Network. 3, 74–76, 78

DAG Dense Adversary Generation. 6, 29, 36, 43, 61

DR Detection as Regression. 54, 59–61

FCIS Fully Convolutional Instance-Aware Semantic Segmentation Method. 37

FGSM Fast Gradient Signed Method. 6, 7, 20, 55, 56, 61

FP False Positive. 22

FPN Feature Pyramid Network. 29

FRCNN Faster R-CNN. 4, 49

FSSD Feature Fusion Single Shot Multibox Detector. 56

G-UAP Generic Universal Adversarial Perturbation. 26, 29–31, 49–53, 61

GAN Generative Adversarial Network. 21

IFGS Iterative Fast Gradient Sign. 7, 41

IoU Intersection over Union. 33, 60, 77, 78

mAP mean Average Precision. vi, 7, 29, 30, 36, 37, 43–45, 47, 48, 50–53, 55, 57, 60, 64, 75–78

MI-FGSM Momentum Iterative Fast Gradient Sign Method. 33, 39, 43–46, 49–53

NMS Non-max suppression. 59, 60, 78

OR Objectness Regularization. 54, 58, 59, 61

PGD Projected Gradient Descent. vi, 6, 38–40, 43, 44, 46, 49–53, 55, 56, 61

PS Probability Score. vi, 44, 45

R-AP Robust Adversarial Perturbation. 29, 33, 36–38, 49–53, 57, 60, 61

ResNet Residual Network. 3, 20, 31, 36, 37, 45, 54

RFB Receptive Field Block-based Detector. 56

RFCN Region Fully Convolutional Network. 29–31, 37, 53

RoI Region of Interest. 25, 46, 78, 80

RPN Region Proposal Network. 1, 4, 5, 20, 21, 25–32, 34–40, 42–44, 47, 61, 63, 76, 78, 80

SAA Sparse Adversarial Attack. 27–31, 49, 50, 59, 61

SSD Single Shot MultiBox Detector. 29, 36, 54, 56, 60, 75

SSM Single Shot Module. 40, 41, 44, 46, 49–53

T-SAT Two-Stage Adversarial Training. 54, 55, 61

TOG Targeted Adversarial Objectness Gradient. 59, 61

TOR Task Oriented Restriction. vi, 54, 56, 61

TP True Positive. 22

UEA Unified and Efficient Adversary. 36, 61

UPC Universal Physical Camouflage Attack. 42, 44, 45, 49, 50

YOLACT You Only Look At Coefficient. 39, 40, 44, 46, 49–53, 61

YOLO You Only Look Once. 28–30, 36, 41, 46, 54–56, 59, 60, 75, 76, 80

Chapter 1

Introduction

While it has been discovered that neural networks are vulnerable to adversarial examples, a large part of the studies focuses on how to fool classifiers. Fooling classifiers were quickly proved to be incredibly effective, both in white-box and black-box settings. Later work studies how to extend these attacks against object detectors. This is deemed a harder task since the object detectors can have multiple proposals for each object. Still, the early work only considered corrupting the classification of each object, not the regions that contain said objects. Since then, studies have built a strong case for proving that targeting the classification alone can be improved by simultaneously targeting the bounding box regression layer in both one-stage and two-stage detectors. This leads to an important field for the future, where the bounding box regression is a common bottleneck across the one- and two-stage detectors, leading to more powerful attacks and better transferability. This highlights the importance of further investigating the state of today's attacks that targets the Region Proposal Network (RPN) or the regressor of the detector. And to research defenses that aim to improve robust region proposals.

1.1 Research Questions and Contributions

Throughout this survey, we seek to uncover the latest attacks showing the highest potential in attack performance in white-box settings and when transferred to black-box settings. Furthermore, the experimental results are to be evaluated to find how well these attacks have been tested, and to indicate if the performance is trustworthy. Lastly, we look into defense strategies that may help to mitigate the consequences of these attacks.

Summarized, the research questions this survey will aim to answer are:

- **RQ1**: What is the state of today's attacks against bounding box regression?
- **RQ2**: How well have the attacks been evaluated?
- **RQ3**: What defenses are there to detect, mitigate or make the models robust against these attack and their evaluations?

Data were collected to answer the above research questions. This was done through an initial manual database search, followed by a snowball process to extract all relevant papers following the references and citations of the papers found in the initial search. Furthermore, a taxonomy is proposed to extract and cluster the attacks found in the data collection phase.

While answering RQ1, we find that the state-of-art attacks targeting the bounding box regression achieve high attack performance in white-box settings. Furthermore, answering RQ2 discovers that several of these attacks provides extensive evaluations which prove the attacks generalize well and can transfer across detectors based on different model architectures, backbone networks, training data and tasks. Lastly, there were only a few of the attacks which have been tried mitigated, thus RQ3 brings an important encouragement to further research these types of attacks to close the gap between their performance and the robustness of object detectors.

The foremost contribution of this survey is providing a foundation to increase awareness of the vulnerabilities connected to the bounding box regression in object detectors. These insights can contribute to helping future research investigate these vulnerabilities, and hopefully uncover the risks and possibilities for making the bounding box regression more robust.

1.2 Structure of the Thesis

Thus, the structure of the thesis is as follows: The relevant theoretical background is presented in chapter 2. Then, previous work conducted which can relate to this survey is presented in chapter 3. Section 4 describes in detail the method of how the survey was conducted, including all steps and the process of extracting the data. In chapter 5, the proposed novel taxonomy will be presented, along with the results of the data collection which is discussed to answer the research questions. The validity of this survey and proposed future work in the field are discussed in chapter 6 and finally, the thesis is concluded in chapter 7.

Chapter 2

Background

Many topics for the background were covered through my specialization project and are referred to in Appendix A. Topics that need an extension for this thesis are further elaborated through this chapter.

2.1 Adversarial Examples

Extending the background from section A.5, the adversarial examples generally aims to find the perturbation to optimize the objective function Equation 2.1, as described by Szegedy *et al.* [1].

$$\begin{aligned} &\min_{x'} ||x' - x||_p, \\ s.t. \quad & f(x') = \hat{y}, \end{aligned} \tag{2.1}$$

where $||\cdot||_p$ denotes the distance metric described in subsection A.11.1, and where x is the benign example, x' is the adversarial example and $\hat{y}$ is a class label different from the ground truth class.

2.2 Backbone Networks

For processing the input, the object detector utilizes backbone networks to extract the features of the input. The backbones are realized through a Convolutional Neural Network (CNN), and among the most common backbones are Residual Network (ResNet) [2], VGGNet [3], MobileNets [4] and DarkNet [5]. ResNet includes ResNet-50, ResNet-101 and ResNet-152, which will be denoted through this survey as rn50, rn101 and rn152, respectively. VGGNets VGG16 is denoted V16, Mobilenets is denoted mn and DarkNet-53 as dn53.

2.3 Object Detectors

For a detailed description of object detectors, see Appendices A.1 to A.4.

2.3.1 Region Proposal Network

Region Proposal Network (RPN) was introduced to generate high-quality region proposals to realize the Faster R-CNN (FRCNN) detector [6]. The RPN is a fully convolutional network that utilizes a novel approach of anchor boxes to produce these proposals. All anchor boxes are associated with multiple scales and aspect ratios to enhance the detector's ability to detect smaller objects. Each region proposal has a binary prediction, named object score, of whether the proposed region contains an object or not.

Faster R-CNN Regression Layer

Faster R-CNN utilizes the bounding box regression introduced in [7] to improve the localization performance. This regression layer of the Faster R-CNN outputs four values: d_x, d_y, d_w and d_h to refine the RPNs proposal coordinates (p_x, p_y, p_w, p_h). To do so, Faster R-CNN computes the bounding box coordinates following these equations from [7]:

$$g_x = p_w \times d_x + p_x \tag{2.2}$$

$$g_y = p_h \times d_y + p_y \tag{2.3}$$

$$g_w = p_w \times e^{d_w} \tag{2.4}$$

$$g_h = p_h \times e^{d_h} \tag{2.5}$$

Where g_x, g_y, g_w and g_h denotes the ground truth x- and y-coordinates, and the width and height of the box.

2.4 Attack formulation on Faster R-CNN

When targeting Faster R-CNN, there are two attack surfaces: The regressor and the classifier. When targeting the regressor, the attack aims to corrupt the regression of the bounding box, achieving mislocated and/or out-of-proportion boxes. Attacks on the classifiers aim to change the label or minimize the confidence of the predicted class.

The combined loss function for targeting the Faster R-CNN is denoted as:

$$L = L_{cls}^{FasterR-CNN} + L_{reg}^{FasterR-CNN} \tag{2.6}$$

where $L_{cls}^{FasterR-CNN}$ is the loss function minimized to reduce the confidence of the true label. $L_{reg}^{FasterR-CNN}$ is the loss function minimized to corrupt the regression of bounding boxes from the region proposals of the RPN. Note that this is the loss function of the Fast R-CNN [8], which is merged with RPN to implement the Faster R-CNN detector.

2.5 Attack formulation on RPN

The RPN has two attack surfaces, one for the binary classification and one for the bounding box regression. Firstly, the classification outputs two values for each region, the confidence of the region containing an object and the confidence of the region being a part of the background. A region proposal that the RPN predicts contains an object with a confidence higher than a given threshold, is said to be a *positive proposal*. Thus, a loss function for this classification task can be designed to make the RPN output only negative proposals, such that there is no object left to predict, as seen in Equation 2.7.

$$L_{cls}^{rpn} = \frac{1}{M}\sum_{i=1}^{N} z_i \log(s_i) \tag{2.7}$$

Where M is the number of positive proposals and N is the total number of proposals. $z_i \in \{0,1\}$, where $z_i = 1$ if the i-th proposal is positive, else 0 for negative proposals. s_i is the confidence score of the given proposal.

Furthermore, the region contains a localization, presented by center coordinates, width and height. The total loss function for attacking RPN can be further extended to target this regression task. This is done by adding an "offset" which disturbs the regression such that the RPN outputs out of proportion bounding boxes which also can have wrong central coordinates, as seen in Equation 2.8.

$$\begin{aligned} L_{reg}^{rpn} = exp(-\frac{1}{N}\sum_{j=1}^{N} z_j(|\Delta x_j - \Delta\overline{x}_j| + |\Delta y_j - \Delta\overline{y}_j| \\ +|\Delta w_j - \Delta\overline{w}_j| + |\Delta h_j - \Delta\overline{h}_j|), \end{aligned} \tag{2.8}$$

Where

- $\Delta x_j, \Delta y_j, \Delta w_j, \Delta h_j$ denotes the predicted offset in terms of object center and bounding box size.
- $\Delta\overline{x}_j, \Delta\overline{y}_j, \Delta\overline{w}_j, \Delta\overline{h}_j$ denotes the true offset between the anchor box and the ground truth.

2.6 Datasets

A large amount of data is needed to train a machine learning model. To simplify this task, there have been collectively gathered large datasets which are freely available. Two of the most used datasets for training object detection models are MS COCO[1] and Pascal VOC[2], from here on called COCO and VOC, respectively.

[1] https://cocodataset.org/

[2] http://host.robots.ox.ac.uk/pascal/VOC/

Excluding background, VOC provides 20 object classes, while COCO provides 80 object classes. Thus COCO is the more complex dataset, which makes it more expensive to train an object detector to achieve high accuracy on the dataset.

2.7 Transferability

Transferability measures how well an attack can be transferred to attack another target than the one used in the training of the attack. This feature is an important and dangerous attribute, as it can make attacks meant for white-box environments be able to attack in a black-box fashion. There are different features that can describe how the attack transfer, given the commonalities and differences between the target and the source. These are cross-model, -task, -network and -data.

Cross-Model Transferability describes how an attack trained on a source-detector can target another object detector [9]. This leads to challenges such as attacks that can try to be transferred between one- and two-stage detectors.

Cross-Network Transferability describes how an attack trained on a source-detector can target an equal detector with a different backbone network [10].

Cross-Data Transferability describes how an attack trained on a dataset can target an equal detector trained on different data [10]. This describes how the attack performs trained on data the target has never seen.

Cross-Task Transferability describes how an attack trained on a detector can attack an instance segmentation model or vice versa [10]. There are some commonalities between the tasks, but the attacks can differ a lot in implementation.

2.8 Adversarial Examples

Xie *et al.* [10] proposed Dense Adversary Generation (DAG) to generate Adversarial Examples (AEs). DAG aimed to make the target propose dense proposals, to further make the target misclassify the proposals with the highest confidence.

2.8.1 Projected Gradient Descent

PGD is a first-order optimization method, bounded by the L_∞ norm, which seeks to find a perturbation that maximizes the loss while keeping the perturbation distance within the restriction given by the L_∞. To achieve this, PGD utilizes random starts for each iteration, making it more robust against saddle point problems. Madry *et al.* utilized PGD to generate AEs to attack image classifiers [11].

2.8.2 Fast Gradient Signed Method

Goodfellow *et al.* [12] introduced Fast Gradient Signed Method (FGSM), a simple and effective method of generating AEs for classifiers. Fast Gradient Signed Method (FGSM) perturbate the target image in such a way that the prediction is pushed across the decision border by following the steepest descent, which is given by the gradient.

2.8.3 Iterative Fast Gradient Sign

Goodfellow and Bengio [13] introduced the Iterative Fast Gradient Sign (IFGS) method, an iterative version of FGSM. IFGS applies the gradient step multiple times with a smaller step size while clipping the intermediate pixel values to ensure they are within the $L_\infty \epsilon$-neighborhood of the targeted image.

2.9 Performance Metrics of Machine Learning Tasks

A description of common performance metrics can be seen in section A.6. These performance metrics are often used to evaluate various machine learning tasks. Moreover, they have been used to evaluate the performance of attacks and defenses by measuring the drop in performance for object detectors when predicting on AEs. The popular mAP metric is both represented by a percentage or by a decimal number between 0 and 1, where 1 indicates 100%. Throughout the survey, the mAP representation has been normalized to use the decimal description of the value.

2.10 Adversarial Training

Adversarial training is one of the effective defenses against adversarial attacks on classifiers [11, 12] and was later generalized to object detection. Adversarial training achieves robustness by solving a min-max problem, Equation 2.9, where the inner maximization generates adversarial examples against the model parameters, θ. These AEs are then used for training the model, thus solving the outer minimization problem with respect to θ.

$$\underset{\theta}{min}\{\max_{x' \in [x-\epsilon, x+\epsilon]} L(f_\theta(x'), y_{true})\} \tag{2.9}$$

Chapter 3

Related Work

There have been several studies conducted in the field of Adversarial Input Attacks over the last decade. And multiple surveys are done to compare and validate the approaches of generating and defending against AEs. To the best of my knowledge, none of these surveys have a particular focus on attacks that target the bounding box regression of the object detectors. Nor have they been clustered to differentiate the nuances within this approach.

Nonetheless, there are several studies and surveys which seek to landscape the field of Adversarial Input Attacks. These are not direct competitors to this thesis, but they provide taxonomies for attacks and defenses for classifiers and object detectors and share some attributes with the taxonomy proposed in this thesis. Some of these studies are discussed and described in this chapter, to provide an overview of the related work existing in the field.

3.1 Existing Literature Reviews and Surveys

Kong *et al.* [14] conducts a survey, where the attacks against machine learning are classified into image-, text- and malware-based adversarial attacks. The image-based attacks are classified through the main categories of *white-box, black-box* and *physical* attacks. The survey further classifies the attacks within the categories based on their access permission, if they are targeted or nontargeted, the application domain and which metrics or strategies are used. Of all the attacks covered by Kong *et al.* [14], 13 are attacks that can target image classifications. The survey adds contribution by helping researchers quickly enter the field of adversarial attacks and by reviewing high-quality relevant articles published since 2010.

3.2 Existing Classification on Adversarial Attacks and Defenses

Pitropakis *et al.* [15] proposes a taxonomy for adversarial attacks against machine learning within the domains such as spam filters, intrusion and classifiers. During the survey, a comparative analysis of classifiers is conducted where attack knowledge is among the attributes. The taxonomy is separated into two distinct phases: Preparation phase and Manifestation phase. The preparation phase is evaluated by Attacker Knowledge (Black-Box, White-Box and Gray-Box), Algorithm (Clustering, Classification and Hybrid algorithms) and Game Theory (Yes/No) attributes. And the manifestation phase is evaluated across the attributes of Attack Specificity (Targeted vs. Indiscriminate), Attack Mode (Colluding vs. Non-colluding) and Attack Type (Poisoning vs. Evasion). Pitropakis *et al.* [15] conducts a literature study and classifies a total of 21 attacks against classifiers according to the proposed taxonomy, where none of which is covered by this survey.

Serban *et al.* [16] conducts a comprehensive survey of attacks and defenses within the field of AEs. By only limiting the survey to object recognition, the scope of the survey is broad and includes a large representative set of offensive and defensive papers. The taxonomy outlines the basic threat models of AEs, and introduces the following attributes for classifying attacks: Attacker Goal (Untargeted vs. Targeted attacks), Attacker Knowledge (Black-Box, White-Box and Gray-Box), Attack Strategies (Noise-based perturbations vs. Geometric transformations). Furthermore, Serban *et al.* [16] introduces taxonomy of defenses, where a defender can be either be *reactive* to new attacks, or *proactive* to try to anticipate new attacks. The defenses are further classified by their Defense Strategies (Guards vs. Defense by design). With the taxonomy, Serban *et al.* [16] classifies a total of 28 attacks and 45 defenses, where none of the attacks nor defenses overlaps with those in this thesis.

Liu *et al.* [17] uses a taxonomy where the attacks are classified by their overall goals. The four classes are *Poisoning*, *Evasion*, *Impersonate* and *Inversion Attack*. Furthermore, the attacks are given attributes within the influence of classifiers (Causative vs. Exploratory attacks), the security violation (Integrity attacks, Availability attacks or Privacy Violation attacks) and the attack specificity (Targeted vs. Indiscriminate attacks). Liu *et al.* [17] also compare 12 different defenses which are clustered by the techniques: Reject on Negative Impact, Adversarial training, defense distillation, ensemble method, differential privacy and homomorphic encryption. Furthermore, the survey illustrates the lifecycle of machine learning and describes how the four classes target different parts of the pipeline. Lastly, the four classes of defensive techniques are described how to fit the lifecycle to handle the security threats. Liu *et al.* [17] provides contribution by identifying and discussing the trends of security threats and defensive techniques of machine learning. Given the reviewed literature, they conclude that new security threats are constantly emerging, while the security assessment is still in its initial stage. Lastly, they argue that increasing security also increases the overhead, thereby reducing the generalization performance of the machine learning algorithms. Thus requiring to jointly optimize the three aspects of the machine learning model to make it feasible in real-world applications.

Chapter 4

Method

The methodology prepared to perform this survey will be described in this chapter. The research motivation and questions create a foundation of the methodology.

4.1 Research Motivation

The property of transferability [9] is a dangerous property within the domain of Adversarial Example attacks. When hiding the internal configuration and training data is no longer enough to mitigate and prevent adversarial input attacks, the consequences of deploying autonomous vehicles and machine learning in safety-critical tasks like medical diagnostics can be far beyond acceptable. The risk of deploying a vulnerable system in the real world is a motivation to research the field of Adversarial Example (AE) alone, and one of the important areas within this field is how an attacker can achieve successful attacks against a target with little to no information.

Throughout the pre-study leading up to this survey, several offensive papers were found that focus on targeting the regressor, indicating a mature field with potential. However, there were no surveys found which focused on this class of attacks. Thereby introducing an interesting angle to research what this attack surface can imply for the attacks and defenses. Besides, the bounding box regression has commonalities across object detectors, which allows for the attacker to target a broader surface when generating the AEs.

There are some very strong and promising defense strategies applicable to classifiers today. To extend these defenses to object detectors, a way would be to apply these strategies within the bounding boxes. Thus, if one could achieve a guarantee that at least all objects within the input would receive a bounding box, this robustness could be combined with other defense techniques for classifiers.

This work will hopefully be the first step towards such a robust detector.

4.2 Research Questions

This thesis targets at answering the research questions below. This will map the state of today's attacks with a focus on the bounding box regression in object detectors. With the purpose of answering what the state-of-the-art attacks and defenses are, how rigorously they are evaluated and what makes them unique. Hopefully, this can provide essential information as second-hand research and help new research progress, both within the scope of attack and defense. Furthermore, any research gap between the state of attack and defense can be accounted for if there are attacks with no proven defense to mitigate them.

- **RQ1**: What is the state of today's attacks against Bounding Box regression?
- **RQ2**: How well have the attacks been evaluated?
- **RQ3**: What defenses are there to detect, mitigate or make the models robust against these attack and their evaluations?

4.3 Research Design

The method described in [18] was used to perform a systematic search and selection process to collect papers needed to answer the research questions. The overall methodology followed in this thesis is visualized by the flowchart in Figure 4.1.

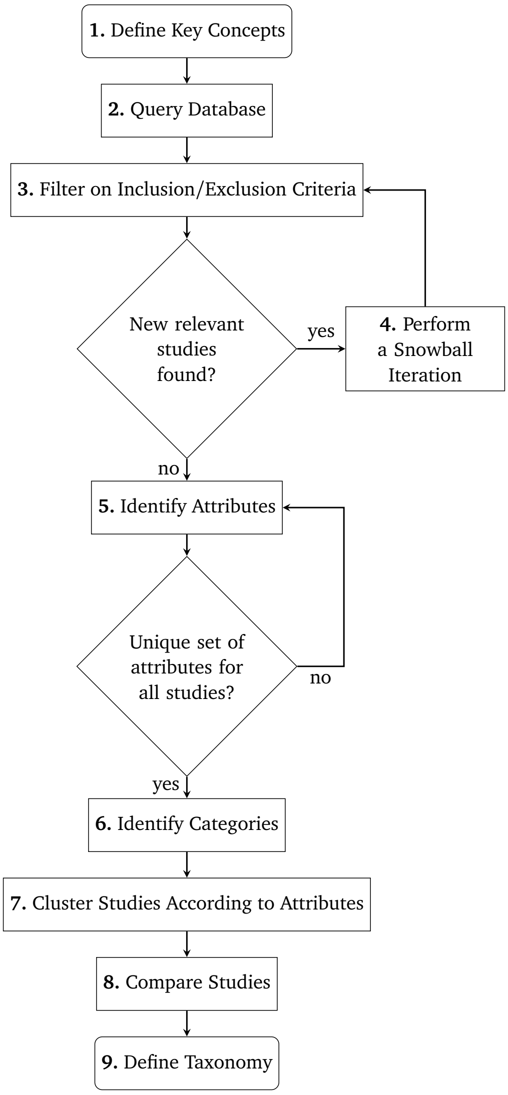

1. Define Key Concepts
2. Query Database
3. Filter on Inclusion/Exclusion Criteria
New relevant studies found?
yes
4. Perform a Snowball Iteration
no
5. Identify Attributes
Unique set of attributes for all studies?
no
yes
6. Identify Categories
7. Cluster Studies According to Attributes
8. Compare Studies
9. Define Taxonomy

Figure 4.1: Flowchart of the methodology used in this survey, with step numbers. Steps 1.-4. concludes the Data Collection, while steps 5.-9. concludes the Data Analysis.

4.3.1 Data Collection

The search and selection process described by Molléri *et al.* [18] were followed to ensure that all relevant papers are found, as shown by steps 1.-4. in Figure 4.1. Firstly, a set of key concepts is defined (step 1.) and described in Table 4.1. These key concepts are combined to construct a search query, such that all selected papers is relevant according to RQ1 and RQ3. The search query will then be used to structure an initial base set of relevant papers from the paper database `oria.no` (step 2.-3.).

Table 4.1: Key Concepts for the study

Concept	Description
Region Proposal	The target of the attacks within this survey, and used as the main attack surface.
Bounding Box	Output of region proposal, a successful attack will disrupt the bounding boxes.
Object Detector	The region proposals and bounding boxes are only relevant for object detectors, which also distinguishes the attacks from attacks aimed for classifiers.
Adversarial Example	The survey will look into attacks providing adversarial images for detection tasks in images and/or videos.

And to further help to discover all relevant studies, a snowball procedure was performed by starting with the base set. This included backward- and forward snowballing until no new relevant studies were to be found (step 4.). Inclusion criteria were declared to filter all results during the snowballing process, as shown in Table 4.2. This meant step 3. and 4. was repeated until no more relevant papers were found.

Throughout the snowballing process, many of the papers being evaluated could be excluded by the title due to the terminology used in the title. When the title stated the paper included an attack/defense regarding classifiers it could be early excluded as it does not match the relevance criteria *Focus on attack or defense within object detection*. The same goes for anchor-free object detection, which was not relevant for this study as it does not fulfill the relevance criteria of targeting bounding box regression.

The key concepts described in Table 4.1 were combined for the following query string:

> ("region proposal" OR "bounding box") AND "object detector" AND "adversarial example"

which was used to perform a manual search on `oria.no`. To reduce the list of papers to a base set with only relevant papers, some inclusion criteria were elicited, and summarized in Table 4.2.

Table 4.2: Relevance and Quality Criteria, used to filter relevant papers for this survey

Relevance Criteria	**Quality Criteria**
English as language	The paper is peer-reviewed
Focus on attack or defense within object detection	The paper provides empiric data for evaluation
First published within the last 5 years, i.e in 2017 or later	
The domain is within image and video	
Have to target the Bounding Box regression of the detector, not only the class label loss within each bounding box	

For papers that could not be excluded by the title, an abstract review was conducted. The abstract quickly gave information if the papers focused on the classification part of detection, and could thus be excluded since they did not provide any attack or defense with a focus on bounding box regression or mislocation.

When there was doubt about the relevance of the paper after the title and abstract review, the paper was reviewed in depth. Here, the important parts for examination were: Methodology, implementation, discussed loss functions and the conclusion. There were also a handful of studies which has staked out the background of all these papers. They were easily identified and most date to before 2017, which was the lower limit of relevance. Thus, they can be excluded as they are only used to build up the background of the papers.

4.3.2 Data Analysis

When all the relevant papers are collected, steps 5.-9. in Figure 4.1 will be conducted to analyze the data, as described byMolléri *et al.* [18]. Firstly, commonalities will be defined as attributes (step 5.) and categories identified to cluster the relevant papers (step 6). When all attributes and categories are identified, the studies can be clustered into these categories (step 7.). Then, the studies can be compared within each category, as well as across the categories to help identify research gaps and propose future work (step 8.). In addition, to answering the research questions, a proposed taxonomy will be included as a product of this survey (step 9.).

Furthermore, data will be extracted from each relevant paper in this survey. The selected data is based on the commonalities of the attacks and defenses and chosen to answer RQ1 and RQ3, such that the overall state of the attacks and defenses could be compared and summarized. The relevant details of each study are

- *what* the paper is proposing

- *how* they formulate the approach to achieve this
- *where* the attack is deployed

Furthermore, to help answer RQ2, all conducted experiments that are done to evaluate the performance of the attacks, including potential tested defensive techniques, are collected and compared.

Chapter 5

Results

To answer the research questions, relevant papers are needed and are found through the method explained in chapter 4. The results will be presented in this chapter. This includes the initial search, traces from the snowball process, identification of categories, brief summaries of the relevant studies and comparisons within, and across all categories. Lastly, the proposed taxonomy will be presented.

5.1 Data Collection

While following the data collection steps from Figure 4.1, the search query described in subsection 4.3.1 provided 62 hits on 29. March 2022 when performing steps 1. and 2. These 62 papers were closely examined up against the inclusion criteria described in Table 4.2 during step 3. The final base set consisted of 6 papers, where 4 papers focused on attacks and 2 on defense measures. The base set is summarized and enumerated in Table 5.1.

Table 5.1: The resulting base set after querying `Oria.no` with the query-string presented in subsection 4.3.1

#	Paper	Attack	Defense	First Published
P1	[19]	✓	✗	2019
P2	[20]	✓	✗	2019
P3	[21]	✓	✗	2021
P4	[22]	✓	✗	2018
P5	[23]	✗	✓	2020
P6	[24]	✗	✓	2021

The base set was diverse, both in publication year and authors. P3 and P4 shared three authors, which was naturally as P3 was an extension of P4. This was the only occurrence of low diversity within the base set. Thus, the base set was deemed to not be biased, and was used for further mining of relevant studies through a snowballing process. Nonetheless, the reference list of the two related papers, P3 and P4, showed to be diverse, as they resulted in different relevant studies in the backward snowballing, as seen in Table 5.5.

With the base set and inclusion criteria in place, the six papers' references and citations were ready to be evaluated through a snowball process. This was done by repeating steps 3. and 4. twice, when no more relevant studies were found while performing the second iterations. For the forward snowballing, Google Scholar was utilized, due to it being a powerful tool to follow the citations of the papers.

Table 5.2: The results of first iteration forward snowballing.

#	Paper	Attack	Defense	First Published
P7	[25]	✓	✗	2020
P8	[26]	✓	✗	2021
P9	[27]	✓	✗	2021
P10	[28]	✓	✗	2018
P11	[29]	✓	✗	2018
P12	[30]	✗	✓	2019
P13	[31]	✓	✗	2019
P14	[32]	✓	✗	2020
P15	[33]	✗	✓	2020
P16	[34]	✓	✗	2019
P17	[35]	✓	✗	2021
P18	[36]	✓	✗	2021
P19	[37]	✗	✓	2021

As seen in Table 5.2, the resulting papers from the forward snowballing were fairly newly published, which was of expectation due to the dates published for the base set. The three defense papers have references to attacks in the base set and might provide defenses against said attacks.

Then, a backward snowballing of the base set was conducted, which only resulted in two new relevant papers, Table 5.3. This provides evidence for this angle of attack being new, and only researched in the last couple of years.

Table 5.3: The results of first iteration backward snowballing.

#	Paper	Attack	Defense	First Published
P20	[38]	✓	✗	2019
P21	[39]	✓	✗	2020

Only a few of the papers in the first iteration had been cited after being published, as shown in Table 5.4, and none provided new relevant papers. And neither did the second backward snowball iteration on the papers in Table 5.3 and Table 5.2. Thus, the final set consisted of the 21 papers (16 attacks and 5 defenses), and their citation matrix can be seen in Table 5.5.

Table 5.4: Second iteration forward snowball results pr. 5. April 2022

Paper	# Citations
P7	1
P8	0
P9	0
P10	98
P11	32
P12	58
P13	51
P14	11
P15	7
P16	4
P17	0
P18	0
P19	1
P20	15
P21	2

Table 5.5: Citation matrix of all relevant papers from the snowballing process. The left column represents the papers whose references are listed in the row. "X" marks that the paper is cited. "-" marks that the paper couldn't have been due to the publication date. The base set (P1-P6) is shaded in gray. All papers are denoted only with their number, without the prefix "P".

Ref.	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1			-	x	-	-	-	-	-			x	-	-	-	-	-	-	-		-
2	-		-		-	-	-	-	-				-	-	-	-	-	-	-		-
3	x			x		-		-	-					x			-	-	-		
4	-	-	-		-	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-
5			-			-	-	-	-								-	-	-	x	-
6				x						x		x					-				x
7	x		-	x		-		-	-								-	-	-		-
8	x																-	-			
9	x					-		-		x							-	-			
10	-	-	-	-	-	-	-	-	-		-	-	-	-	-	-	-	-	-	-	-
11	-	-	-	x	-	-	-	-	-			-	-	-	-	-	-	-	-	-	-
12	-	-	-	x	-	-	-	-	-	x	x		-	-	-	-	-	-	-		-
13	-		-	x	-	-	-	-	-	x				-	-	-	-	-	-		-
14			-	x	-	-	-	-	-	x	x	x	x		-		-	-	-		-
15			-	x	-	-	-	-	-	x	x						-	-	-		-
16			-	x	-	-	-	-	-					-	-		-	-	-		-
17				x		-		-	-	x		x						-	-		
18				x				-													
19	-			x		-		-	-	x		x						-			
20	-	-	-		-	-	-	-	-			-	-	-	-	-	-	-	-		-
21			-			-		-	-	x			x				-	-	-		

The citation matrix is presented in Table 5.5, where each row represents the paper in the left columns references. The base set is represented by faded cells.

Since most of the papers are relatively new, as seen in Table 5.1, an assumption would be that most of the papers will be found from backward-snowballing. This was not the case for the snowballing process, this can be seen in Table 5.5 as the base-set contains a lot of "-", meaning most papers of the first iteration of snowballing the base-set came from forward-snowballing. This, together with the timeline, Table 5.6, indicates that this is a new field, where most research has happened in the last couple of years.

Table 5.6: Timeline of which years the papers were first published.

Year of publish	Papers	Count
2017	-	0
2018	P4, P10, P11	3
2019	P1, P2, P12, P13, P16, P20	6
2020	P5, P7, P14, P15, P21	5
2021	P3, P6, P8, P9, P17, P18, P19	7
2022	-	0

When conducting a snowball process, one of the strengths of the final set of selected papers is that they are intervened through citations. This means that finding any one of the papers from the final set will lead to finding the entire final set after snowballing. This indicates that the final set covers relevant papers that are not found in the initial query, and is thereby found regardless of the keywords used by the authors and in the papers search step.

The citation matrix, Table 5.5, shows that this is almost the case for the final set in this survey, with the exception of the papers P2, P5 and P20. P20 was only found through the citations of P5, while P2 and P5 were only found through the initial query. This reflects the importance of the combination of a careful initial query and a snowball process when finding and extracting relevant work in the searching process.

5.2 Classification of Attacks

While analyzing the data, the attributes defined in step 5. of Figure 4.1 will be identified for each paper. The identified attributes are *Attack Knowledge, Target(s), Transferability, Loss function, Environment, Attack Style* and *Optimization Scheme*.

Attack Knowledge describes what information the attacker assumes it can retrieve from the target. The possible values for this survey are *Black-Box* and *White-Box*. If an attack assumes to have no information about the target, the attack is said to be a Black-Box attack. While if the attacker assumes to have information about the internal configurations (e.g. the weights of the model) or the training data, the attack is said to be White-Box. This is an important property to evaluate how effective the attack can be in the real world, where the information of object detectors most likely will be concealed by the enterprises. Thus also important to evaluate the state of the attacks today to answer RQ1.

Target(s) describes which object detector, or sub-modules, that are targeted for each attack. This is needed to see a trend of which detector or sub-modules are successfully attacked and which approaches they are weak to. This can be the different layers of the model, for instance, the classification layer of a Faster R-CNN model, or the RPN itself.

Transferability is an important attribute to give a sense of the generalization of the attack. The possible values are cross-model, cross-network, cross-data and cross-task, as described in section 2.7. An attack that can be transferred to attack different models than the one which is targeted has a high level of generalization. This implies that the attack will be easier to deploy due to the lack of needing information about the target. This also helps to answer RQ2 if extensive experiments on transferring the attack are conducted, it indicates a more rigorous evaluation of the attack. For example, Li *et al.* [21] reports cross-network transfer for Trans-RPN where an AE designed to attack a Faster R-CNN detector with a ResNet-50 backbone network achieves to attack another Faster R-CNN detector trained on the same data with a ResNet-101 backbone.

Loss Function describes the attack surface of the target. More precisely, the loss functions that the attacks seek to compromise while generating the AE. This attribute helps to emphasize which parts of the target are vulnerable and explains how each attack would expect to work if they manage to perform a successful attack. These loss functions can include the binary classification loss of the RPN or the regression loss of the targeted object detector. For instance, Shi *et al.* [35] targets the classification and regression loss of RPN and the regression loss of Faster R-CNN simultaneously when generating AEs for attacking two-stage detectors.

Environment describes if the attack is deployed *Digital* or *Physical*. A digital attack generates the AE and directly feeds the digital AE to the targeted detector. A physical attack includes a perturbation or AE whose attack performance is preserved when printed out physically and placed in the real world. The early stages of AEs focused on deploying the generated perturbation digitally, e.g. the FGSM attack by Szegedy *et al.* [1]. Digital deployed attacks proved to be very efficient, but not sufficient to deploy an attack in the real world. To deploy and attack digitally, one would need to be able to change the captured frame with an adversarial. Later studies have proven perturbation can be deployed in the physical world, e.g. as shown in [31]. Physically deployable perturbations are a dangerous feature of AEs and need to be investigated to discuss the state of these attacks.

Attack Style describes the style of the perturbation, and includes the attributes *Patch-based* and *Noise*. Patch-based attacks apply the perturbation to a restricted area of the target image, these restricted areas can have different forms, where the most common is a rectangular patch, e.g. as shown in [28, 39]. Noise attacks apply the perturbation to the entire image, e.g. as shown in [21]. Patch-based attacks have been more researched in the previous last years, making it important to distinguish the trend of the attack styles.

Optimization Scheme describes how the attack solves the loss function and generates the perturbations. This can be done through *gradients* (e.g. the boosted gradient descent in [25]), $l_p norms$ (e.g. the l_2-norm restriction in [20]) or by utilizing Generative Adversarial Networks (GANs) (e.g. as done in [36]). Gradient-based attacks can be slow, needing several iterations to achieve devastating performance. With computers becoming faster, the computational-time decreases and may be of less importance in the future.

With these defined attributes, the relevant attack studies are clustered with respect to their *Attack Knowledge, Target(s), Transferability, Loss function, Environment, Attack Style and Optimization Scheme* and categorized within the categories presented in Table 5.7. All categories shares some attributes, which systematically identify each study, such that each of them has a unique set of attributes.

Table 5.7: The different categories of which the studies are divided into.

Category	**Description**	**Common Attack Output**
Background Evasion Attack (e.g. [39])	The attack aims to reduce the objectness and discard positive proposals. Such that all objects are proposed as a part of the background.	Evasion of all objects
Offset-Push Attack (e.g. [21])	The attack aims to disturb the RPN by maximizing the offset between the proposed bounding box and the ground truth.	Large Bounding Boxes/Segmentation's with low confidence
Total Loss Attack (e.g. [19])	The attack includes classification and bounding box regression loss in their total loss functions (and RPN loss if targeting two-stage detectors), and maximizes the loss. Thus the attacks try to make the predictions as wrong as possible, without any clear targets.	Object fabrication, evasion and misclassification
Region of Interest Attack (e.g. [28])	The attack focuses all the interest in one specific region. Effectively disabling all other proposals from the detector.	One very confident proposal on the targeted area. May also break the detector in a more untargeted manner

To further describe the common attack outputs presented in Table 5.7, a deeper look into how the attacks target the True and False Positive proposals of the detectors can distinguish the attack outputs.

During object detection, the detector will propose several predictions of regions where there might be an object. These predictions are filtered on how confident the detector is that there actually is an object in the region. The proposals which pass this filtering are said to be positive proposals, while discarded proposals are negative proposals.

For the positive proposals, if the prediction is correct that there actually is an object in the region, the proposal is a True Positive (TP). On the other hand, if the detector is wrong and there is no object in the positive region, the prediction is a False Positive (FP). In other words, False Positives (FPs) are erroneous proposals where the detector detects an object which is not there.

Furthermore, if there is an object in the frame, and the detector fails to provide a positive proposal for this region, it is said to be a False Negative. These False Negatives are by nature adversarial, as it includes an object that has been evaded by the detector. Thus, the False Negatives are not considered when describing the common attack outputs. The True Negatives are all regions that do not enclose any pixels of an object, which means there are way too many True Negatives in object detection to provide any information. Thus, True Negatives are not useful in object detection and are also ignored.

Given these definitions, we can look into how the attacks try to affect the detector's rate and form of the TP and FP predictions. A commonality for all the cluster classes is that they try to evade the TPs. Further distinguishing of the cluster classes can be elicited by inspecting whether or not they try to misclassify or distort the TPs, and if they try to remove any FP or increase the number of FP by fabricating them. This is presented in Table 5.8, where the categories have a unique distribution of attacking goals.

Table 5.8: Further distinction between cluster categories, with focus on their attack goals on the True Positives (TP) and False Positive (FP).

Category	**TP**			**FP**	
	Evade	**Misclassify**	**Distort**	**Evade**	**Fabricate**
Background Evasion Attack	✓	✗	✗	✓	✗
Offset-Push Attack	✓	✗	✓	✗	✗
Total Loss Attack	✓	✓	✗	✗	✓
Region of Interest Attack	✓	✗	✗	✗	✓

- *Background Evasion Attacks* seeks to evade all proposals, leaving the detector to believe the entire scene is background.
- *Offset-Push Attacks* aims to evade the true positives or distort the height and width of the bounding box or by shifting the center.
- *Total Loss Attacks* seeks to attack the true positives by misclassifying and evading the proposals, in addition, to increasing the false positives by fabrication.
- *Region of Interest Attacks* seeks to evade the objects and increase false positives with the goal of making the fabricated proposal the only region of interest for the targeted object detector.

5.3 Proposed Taxonomy of Attacks

With the attributes identified, a taxonomy can be defined, according to step 9. in Figure 4.1. Some of the attributes affect the same phases during the lifecycle of an AE attack. Besides, while the different attacks have their own method of generating AEs, there are some shared commonalities in how they proceed to generate and deploy their AEs. The proposed taxonomy groups these commonalities into three separate phases, which together show the entire framework of how the different Adversarial Input Attacks are conducted. These phases are summarized as follows:

- *Generation*
- *Deployment*
- *Transfer*

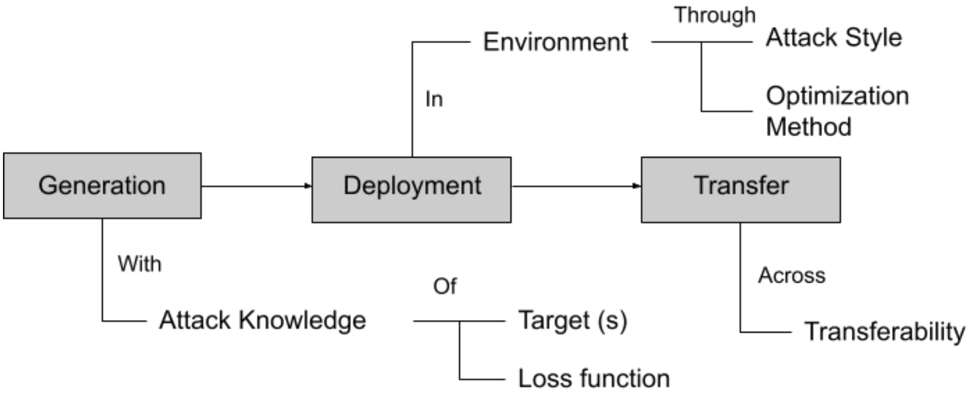


Figure 5.1: The connection between the phases and their attributes of the proposed taxonomy

The *Generation*-phase envelops all the necessary steps for an attack needed before the attack can be deployed, here the attributes *Attack Knowledge, Target(s) and Loss Function* of the studies in this survey is important. These attributes describe and distinguish how the different attacks proceed to generate the AEs.

The *Deployment*-phase utilizes the attributes *Environment, Attack Style and Optimization Method* to realize and deploy the attack. Some attacks are only applicable in a digital setting, where the attacker needs to interfere with the pipeline between when the targeted image is taken, and when it is inputted into the object detector module. This makes it less realizable in the real world and requires full access to the targeted system, which would lead to far more simpler and effective attacks being more suitable, as one can replace the taken image with just an empty frame instead of perturbating it. Nonetheless, this is an important part of the research of the state of adversarial examples, and can result in enormous attack performance. Attacks applicable in the physical environment can be deployed in the real world, thus interfering with the target before the image is even taken.

The *Transfer*-phase includes the *Transferability* attribute by describing if the AE generated to attack model A is applicable to attack model B. This is given that model A differs from model B with respect to the detector model itself, its backbone network, training data or the performed task. This feature can make an attack meant for white-box settings being able to attack a different target in a black-box fashion. Furthermore, it describes the state of generalization between attacks and object detectors.

The connection between the three phases, including their attributes, are visualized in Figure 5.1, which reassembles a general framework for AE attacks.

5.4 Background Evasion Attacks

An attack that can make the object detector perceive the entire presented scene as a background would make all objects in the given scene evade objection. Such an attack would lead to fatal consequences in tasks such as autonomous vehicles and medical diagnosis. As presented in Table 5.8, *Background Evasion Attacks* has one goal: *Evade all true and false positive proposals*. And given that the attacks expect to evade all positive proposals, there are no true positives left to misclassify or distort.

Attacking Object Detectors Without Changing the Target Object

Huang *et al.* [20] proposes an Adversarial Border (AB) algorithm that generates an AE without alternating any pixels within the targeted object. Instead, the attack places the adversarial pixels around the border.

Adversarial Border (AB) is generated with regard to Faster R-CNN Regression Loss. As seen in Equation 2.4 and Equation 2.5 from section 2.3.1, g_w and g_h are heavily affected by d_w and d_h respectively due to the exponential function. This is a result of the design of Faster R-CNN, which ensures d_w and d_h can be learned effectively. Thus, small alterations to these values will lead to huge impacts in the final regression.

The proposed AB targets the regression layer such that it outputs overly large bounding boxes for the targeted object. This will in turn lead the classification confidence to be drastically reduced. More precisely, the size of the output bounding box will increase with a factor of $e^v \times e^v$. Meaning if the regression layer is misled to output $d_w = v$ and $d_h = v$, where $v > 0$, the proposed bounding box will be enlarged by an exponential large factor. The study used experiments which set $v = 1$, resulting in a 738% enlargement of the proposed bounding box (where $e^2 = 7.38$).

Contextual Adversarial Attacks For Object Detection

Zhang *et al.* [32] proposes Contextual Adversarial Perturbation (CAP), a novel attack that optimizes previous methods of targeting both the classifier and the RPN of object detectors by damaging the contextual information of target objects by utilizing a background loss. The proposed attack does not rely on ground-truth information, making it more generalized and subject to attack Weakly-Supervised Object Detectors (WSOD).

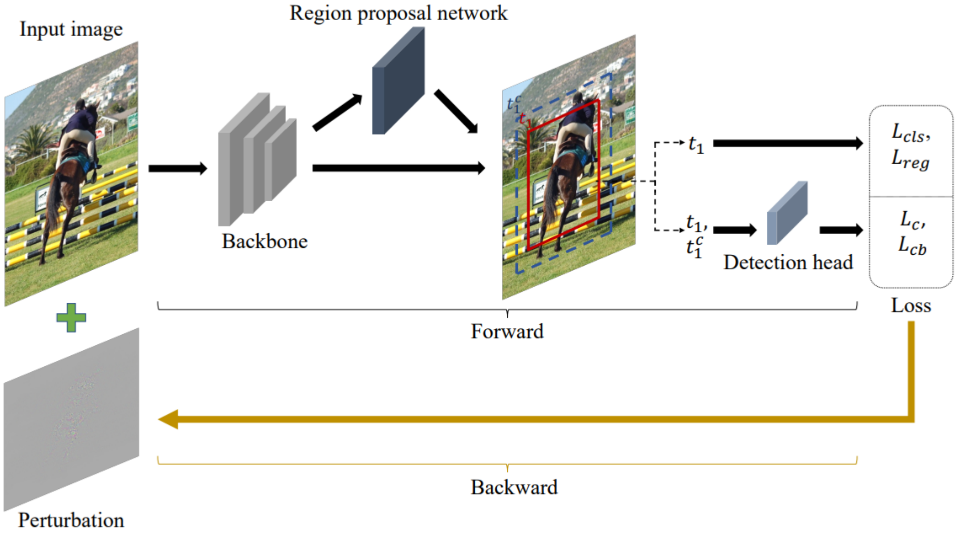


Figure 5.2: Overview of the CAP attack, obtained from [32]

Contextual Adversarial Perturbation (CAP) effectively damages the local and contextual information of the positive proposal. This is done by combining classification loss (L_{cls}, Equation 2.7), regression loss (L_{reg}, Equation 2.8) and the novel context loss (L_c, Equation 5.1). L_c aims to corrupt the contextual information of each positive Region of Interest (RoI), thus CAP attack the contextual information for a total of M RoIs by optimizing the context loss L_c:

$$L_c = \frac{1}{M}\sum_{j=1}^{M} z_j e_j^2, \tag{5.1}$$

where M denotes the number of attacked RoI, e_j is the highest classification score for the region and $z_j \in \{0, 1\}$ indicates false or true proposal, respectively.

And for the contextual background loss L_{cb}, CAP optimizes:

$$L_{cb} = \frac{1}{M}\sum_{j=1}^{M} -z_j \tilde{e}_j^2, \tag{5.2}$$

where $\tilde{e}_j$ denote the highest background score of the contextual region.

The total object function that CAP optimizes to generate AEs is then summarized as:

$$\min_{X}\{L_{cls}(X;\mathcal{D})+L_{reg}(X;\mathcal{D})+L_{c}(X;\mathcal{D}) \\ +L_{cb}(X;\mathcal{D})\}, s.t. PSNR(X)\leq\epsilon \tag{5.3}$$

where X is the input image, $\mathcal{D}$ is the target object detector, PSNR is the Peak Signal-to-Noise Ratio limited by the threshold ϵ. As seen in the overview of CAP, Figure 5.2, the attack first calculates the losses by the positive proposals from the RPN, and the contextual regions. Then CAP generates the perturbation through back propagation, applies it to the target image and inputs it to the next iteration.

CAP is also generalized to attack WSOD, which does not utilize RPNs. This part of the study is not discussed in this thesis, as it no longer falls under the inclusion criteria of targeting bounding box regression.

G-UAP: Generic Universal Adversarial Perturbation that Fools RPN-based Detectors

Wu *et al.* [34] proposes Generic Universal Adversarial Perturbation (G-UAP), an attack to generate universal perturbations. Generic Universal Adversarial Perturbation (G-UAP) aims to directly mislead the RPN, by making it mistake foreground object for background.

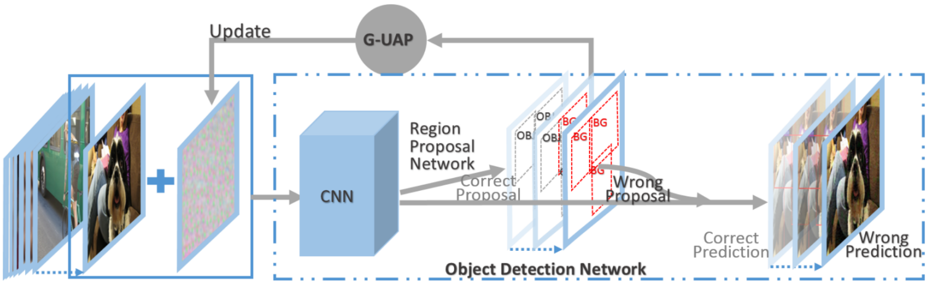


Figure 5.3: Overview of the attack framework for G-UAP, obtained from [34]

G-UAP optimizes Equation 5.4 to generate perturbation that fools the RPN to mistake the foreground as the background.

$$l_{cls\_obj}(x_i+\delta)\approx l_{cls\_obj}(x_i)+\mathcal{J}_{l_{cls\_obj}}(x_i)\delta, \\ l_{cls\_bg}(x_i+\delta)\approx l_{cls\_bg}(x_i)+\mathcal{J}_{l_{cls\_bg}}(x_i)\delta \tag{5.4}$$

Where $\mathcal{J}_{l_{cls\_obj}}(x_i)$ is an Jacobian Matrix, $\mathcal{J}_{l_{cls\_obj}}(x_i+\delta)$ is the probability scores of foreground that the attack tries to get to 0 and $\mathcal{J}_{l_{cls\_bg}}(x_i+\delta)$ is the probability scores of the background that the attack tries to get to 1. This would result in the

RPN incorrectly give all regions a label of background, leaving no targets for the classifier to label. This is done when Equation 5.4 is optimized to output:

$$l_{cls\_obj}(x_i + \delta) \approx 0 \quad \text{and} \quad l_{cls\_bg}(x_i + \delta) \approx 1$$

Generating Adversarial Remote Sensing Images via Pan-Sharpening Technique

Yuan and Wei [36] proposes an Adversarial Pan-Sharpening (APS) method that utilizes a generative network to generate a pan-sharpened image. Adversarial Pan-Sharpening (APS) includes a shape loss and label loss to generate AEs. The method applies adversarial noise to a pan-sharpened image, which disturbs the prediction of the RPN in the targeted model. The label loss (L^{rpn}_{cls}, Equation 2.7) decreases the confidence of positive proposals. In addition, the shape loss (L^{rpn}_{reg}, Equation 2.8) disrupts the bounding box regression of the RPN, as described in section 2.5.

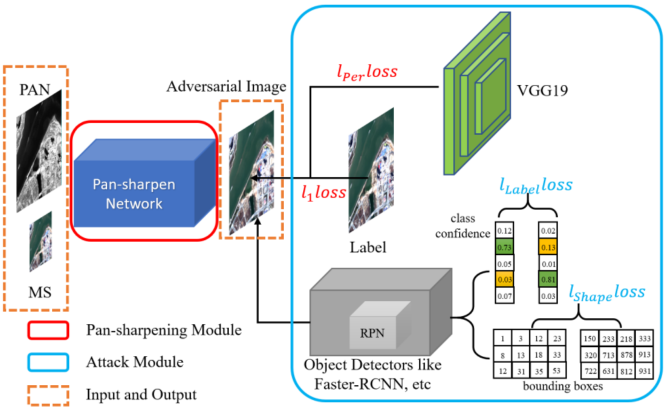


Figure 5.4: Overview of the Adversarial Pan-Sharpening (APS) method, obtained from [36].

An overview of the proposed attack method APS is shown in Figure 5.4. L_{l_1} and L_{per} loss is used to generate the pan-sharpened image. While the attack module generates an AE through the loss functions L^{rpn}_{reg} and L^{rpn}_{cls} to fool the RPN.

By combining these loss functions, the attack generates AEs by optimizing the objective function:

$$\min\{L_{l_1} + \alpha L^{rpn}_{cls} + \beta L^{rpn}_{reg} + \delta L_{per}\}, \tag{5.5}$$

where α, β and δ are relative weights for the different losses.

Sparse Adversarial Attack to Object Detection

Bao [39] proposes Sparse Adversarial Attack (SAA) to perform a pixel-efficient evasion attack on object detectors. SAA aims to make all objects in the image evade detection through a sparsely distributed patch attack.

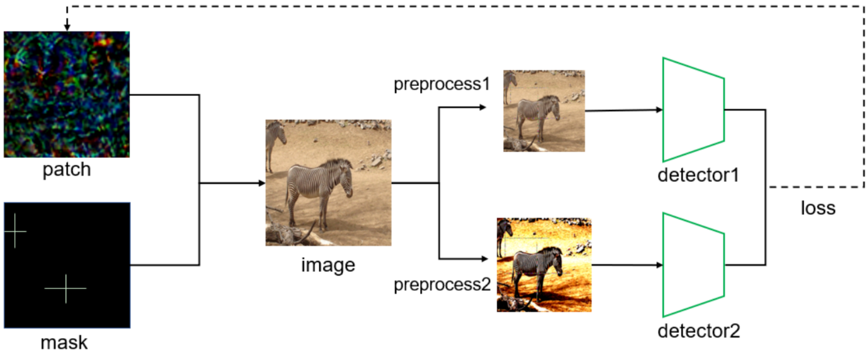


Figure 5.5: Overview of the SAA, obtained from [39]

Figure 5.5 shows the overview of the SAA attack, where the overall goal is to make all objects evade detections by making the detectors believe the objects are a part of the background, thus making the detector perceive all proposed regions as a negative prediction.

To make the object evade detection, SAA constructs a loss function based on the definition of foreground and background of You Only Look Once (YOLO)v4 [40] and Faster R-CNN. YOLOv4 utilizes a confidence branch to distinguish the two, and bounding boxes with confidence below a given threshold are discarded as background. The evasion loss for YOLOv4 is defined as:

$$Loss_{YOLO} = \max_{c \in C, b \in B} (conf(c, b)), \tag{5.6}$$

where C denotes all object categories and B the set of all predicted bounding boxes. YOLOv4 maximizes the object confidence, $conf(\cdot)$, of an image. To attack the YOLO detector, SAA seeks to generate AEs by minimizing $Loss_{YOLO}$.

For the evasion loss of Faster R-CNN, SAA increases the softmax output probability of background, while decreasing that of foreground object categories.

$$LOSS_{FRCNN} = \alpha_1 \cdot Loss_1 + \alpha_2 \cdot Loss_2, \tag{5.7}$$

where α_1 and α_2 are hyperparameters. $Loss_1$ and $Loss_2$ in Equation 5.7 are defined as

$$Loss_1 = \max_{c \in C, b \in B} conf(c, b) \tag{5.8}$$

$$Loss_2 = \frac{1}{N} \sum_{b \in B} \max_{c \in C} (conf(c, b)) \tag{5.9}$$

This combination of loss functions is utilized to handle the enormous amount of region proposals from the output of the RPN, where $Loss_2$ aims to erase as many bounding boxes as possible, while $Loss_1$ attacks the bounding box with the highest object probability, given by the confidence that the bounding box containing an object from the classes in C.

SAA ensembles the two loss functions of YOLOv4 and Faster R-CNN for the final loss function:

$$Loss = Loss_{YOLO} + Loss_{FRCNN} \tag{5.10}$$

5.4.1 Attack Evaluation

The CAP attack is benchmarked against DAG [10] and Robust Adversarial Perturbation (R-AP) [21], where CAP outperforms DAG and R-AP on almost all targeted labels. Furthermore, of the conducted experiments, CAP achieves several 0.0 mAP scores.

Wu *et al.* [34] conducts experiments for cross-model transfer, where they test against another RPN-based detector: Region Fully Convolutional Network (RFCN) [41] and an one-stage detector: Single Shot MultiBox Detector (SSD) [42]. The attack transferred to the RFCN detector, but experiments show that the attack does not transfer well to detectors that do not rely on RPN for region proposals. The experiments also indicate that the attack can transfer between Faster R-CNN with different backbones.

Yuan and Wei [36] conducts experiments on six state-of-the-art object detectors and evaluates across five different pan-sharpening evaluation metrics. The evaluation metrics show that the APS method achieves 0.79 and 0.73 drop of accuracy on Faster R-CNN and Feature Pyramid Network (FPN) [43] detectors, respectively.

To further examine how the attack compares, the reported mAPs are presented in Table 5.9. Henceforth, Faster R-CNN will also be denoted as FR in tables for simplicity. The APS attack [36] achieves the highest mAP drop and reduction in percentage when comparing the adversarial result to the mAP achieved on the benign dataset without perturbations. The CAP attack [32] achieves similar adversarial results and has a comparable drop in mAP to the APS attack

Table 5.9: Reported mAP@0.5 evaluation for the white-box results of the *Background Evasion Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop in mAP. Bold entries indicates highest attack performance in the table.

Study	**Target Detector**	**Dataset**	**Benign Result**	**Adversarial Result**	**mAP Drop (Drop in %)**
CAP [32]	FR-rn101	VOC 2007	0.79	0.02	0.77 (97.5%)
		COCO 2014	0.55	0.01	0.54 (98.2%)
G-UAP [34]	FR-V16	VOC 0712	0.76	0.34	0.42 (55.3%)
		VOC 2007	0.71	0.31	0.40 (56.3%)
	FR-rn101	VOC 2007	0.76	0.50	0.26 (34.2%)
APS [36]	FR-V16	GaoFen-1 [44]	0.87	0.01	**0.86(98.9%)**

The AB attack from [20] and SAA from [39] did not report any mAP for their experiments but presented their results with Attack Rate and an Evasion Score, respectively. This makes it difficult to cross-examine these attacks against the three others from the same category. Neither did they provide a base result for the clean images, this was rather incorporated into the evaluation metric.

Huang *et al.* [20] defines the successful attack rate as $AttackRate = 1 - Det_{adv}/Det_{org}$, where Det_{adv} denotes the number of detected stop signs with an AB and Det_{org} denotes the total number of detected targeted stop signs without any border. The experiments gave an Attack Rate of 0.912, as shown in Table 5.10, which means 91.2% of the stop signs with an AB were successfully evaded from detection.

Bao [39] defines their evaluation metric as a bit more complex. As a *Background Evasion Attack*, they aim to evade all objects, thus their Evasion Score calculates a score based on the number of Bounding Box predictions. This means for a single image, the Evasion Score is 2 if there are no Bounding Box predictions in either the benign or the adversarial example. On the other hand, the evasion score is 0 if the number of predicted Bounding Boxes is larger than or equal to the number from the benign example. Bao [39] evaluated the evasion score on 1000 images, giving the best case scenario an Evasion Score of 2000. Table 5.10 shows that the SAA on YOLOv4 leads to high performance with a score of 1610.03. Bao [39] does not declare any amount of total bounding box prediction on the benign dataset, so any percentage of evaded objects can't be further elicited. Anyhow, the attack achieves substantially worse on the Faster R-CNN model, with an Evasion Score of 1174.21. As the two-stage model Faster R-CNN is known to be better at detecting smaller objects, and one reason for this large drop in SAA's performance might be that there were a large amount of smaller objects in the evaluation dataset, thus making it easier to disturb the one-stage model YOLOv4 in the evaluation.

Table 5.10: Reported evaluation for the white-box results of the *Background Evasion Attacks* which did not provide mAP evaluation

Study	Target Detector	Dataset	Evaluation Metric	Adversarial Result
AB [20]	FR-V16	COCO	Attack Rate	0.912
SAA [39]	YOLOv4	COCO 2017	Evasion Score	1610.03
	FR-rn50			1174.21

The studies [20], [34], [36] and [39] conducted experiments to evaluate their attacks against black-box targets, this was done by transferring the attack to another model, a model trained on a different dataset or with another backbone network.

Wu *et al.* [34] conducted experiments for cross-model and cross-data transferability for the G-UAP attack, as shown in Table 5.11. The first lines show that the attack achieves a 39.2% mAP reduction when transferring the AE to another two-stage model, RFCN, which also utilizes an RPN. When transferring to a one-stage model on the other hand, the reduction is quite smaller. Likely due to the one-stage model having no RPN in the base network. Thus the attack surface, the objectness label loss in the RPN, loses its effect when the target is not dependent on it.

Lastly, and quite remarkably, Wu *et al.* [34] reports evaluation showing that several of the cross-data attacks outperforms the white-box G-UAP attacks. This applies to the attacks on both VGG16 and ResNet-101 trained on VOC 2007, where the AE trained on the same models, but trained with VOC 0712 outperforms the white-box attacks. Thus indicating that the G-UAP attack is indifferent to the difference between the training data for the attack and the target detector.

Table 5.11: Reported mAP@0.5 evaluation for the black-box results of the *Background Evasion Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop of mAP. Bold entries indicates highest attack performance in the table.

Study	**Source Detector**	**Target Detector**	**Benign Result**	**Adversarial Result**	**mAP Drop (Reduction in %)**
G-UAP [34]	FR-rn101	RFCN	0.74	0.45	0.29 (39.2%)
	FR-V16	SSD	0.78	0.68	0.10 (12.8%)
	FR-V16 VOC0712	FR-V16 VOC07	0.71	0.28	0.43 **(60.6%)**
	FR-rn101 VOC0712	FR-rn101 VOC07	0.76	0.47	0.29 (38.2%)
APS [36]	FR-rn50	FR-V16	0.87	0.38	**0.49**(56.3%)

Bao [39] reported an evasion score of 355.69 when transferring the SAA attack to two black-box models. Given that no transfer enhancement was implemented in the attack, this is a strong foundation showing the promise of the attack in black-box settings.

5.5 Offset-Push Attack

Section 2.5 describes how the regressor in RPN can be disrupted by adding a large offset. The *Offset-Push Attacks* mainly targets this regression loss, L_{reg}^{rpn}, of the RPN. The goal of pushing the offset is to disrupt the regressor to output out of proportion bounding boxes, which may have shifted the center such that the bounding box no longer contains the original object, or contains both the object and a large part of the background. This will firstly result in erroneous predictions, but also predictions with lower confidence such that they may are evaded in the final prediction.

TransRPN: Towards the Transferable Adversarial Perturbations using Region Proposal Networks and Beyond

Li *et al.* [21] extends a previous work of Li *et al.* [22] and proposes TransRPN which targets the RPN as the common bottleneck of the existing object detectors. The attack is done by attacking the intermediate features of the RPN, thus disrupting it.

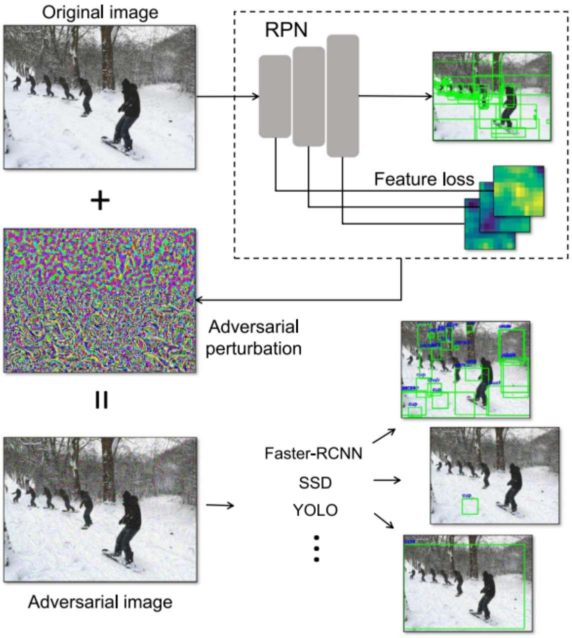


Figure 5.6: TransRPN framework, obtained from [21]

The attack utilizes the confidence loss (L_{cls}^{rpn}, Equation 2.7) and shape loss (L_{reg}^{rpn}, Equation 2.8) of the RPN to generate AEs. TransRPN also includes a feature loss (L_f, Equation 5.11).

$$L_f = \frac{1}{k}\sum_{i=1}^{k} \frac{f_i^T \cdot f_i'}{||f_i|| \cdot ||f_i'||} \tag{5.11}$$

Which is the cosine distance between the k features of the benign image f_i and the k features of perturbated image f_i', where f_i^T denotes the transpose of f_i. By minimizing L_f, the attack seeks to increase the error between the two sets of features and thus achieve higher attack performance.

The shape loss attacks the bounding box regression by explicitly disturbing the shape regression, such that the proposed boxes are diverging from the ground truth boxes as described in section 2.5. Hence, minimizing Equation 2.8 encourages pushing the predicted offsets away from the true offsets, leading to mislocalization of the bounding box.

Li *et al.* [21] focuses on the transferability of the attack between RPNs. And through their experiments, they observe that solely attacking the feature loss, L_f, achieved higher transferability than including L_{cls}^{rpn} and L_{reg}^{rpn}. Furthermore, following the DIM Method described in Xie *et al.* [45], Li *et al.* [21] adopts the

Momentum Iterative Fast Gradient Sign Method (MI-FGSM) method [46] for optimizing the TransRPN loss function. This was due to Xie *et al.* [45] observing that a mixture of optimization schemes can improve transferability for attacks against classifiers, and Li *et al.* [21] wished to observe the effect this had on transferability among object detectors.

Robust Adversarial Perturbation on Deep Proposal-based Models

Li *et al.* [22] proposes a Robust Adversarial Perturbation (R-AP) method to attack object detectors and instance segmentation models. R-AP generates AEs by an objective function combining label and shape loss. This objective function is further optimized by using an iterative gradient-based method [13]. R-AP aims to disturb the bounding box shape regression, by increasing the offset, such that the shape regression can't successfully adjust the anchor boxes to the ground truth bounding box.

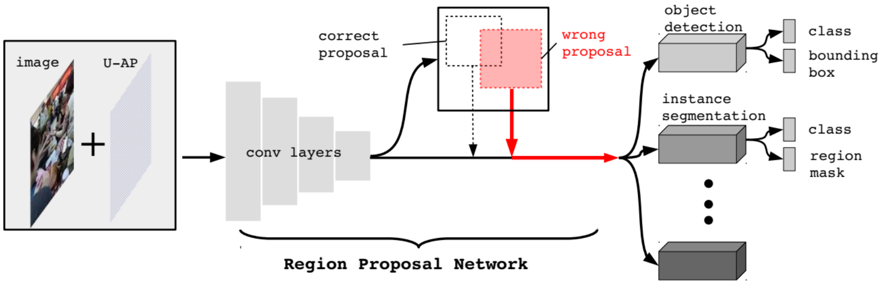


Figure 5.7: R-AP framework, obtained from [22]

R-AP proposes a novel variant of the L_{reg}^{rpn} in Equation 2.8, by substituting the ground truth coordinates of the offset by large offsets.

The novel shape loss is thus defined as:

$$\begin{gathered} L_{shape} = \sum_{j=1}^{m} z_j\big((\Delta x_j - \tau_x)^2 \\ +(\Delta y_j - \tau_y)^2 + (\Delta w_j - \tau_w)^2 + (\Delta h_j - \tau_h)^2\big) \end{gathered} \tag{5.12}$$

Where m is the total number of proposals, $\tau_x, \tau_y, \tau_w and \tau_h$ denotes the large offsets and $\Delta x_j, \Delta y_j, \Delta w_j and \Delta h_j$ are the predicted offsets. $z_j \in [0,1]$, where $z_j = 1$ if the j-th proposed box either has an Intersection over Union (IoU) above a given threshold τ_1 or if the confidence score of the proposal is above a given threshold τ_2, and $z_j = 0$ otherwise. By minimizing Equation 5.12, the predicted offsets will be forced to approach the large substituted offsets $(\tau_x, \tau_y, \tau_w, \tau_h)$, such that the bounding box will be incorrect.

Adversarial Attacks on Object Detectors with Limited Perturbations

Shi *et al.* [35] proposes DTTACK, a framework to attack both one- and two-stage detectors. DTTACK utilizes a salient map to target the more salient part of the targets to increase the probability of a successful attack.

One-Stage Detectors

To attack one-stage detectors, DTTACK introduces L_{BB} as a penalty for correctly detected bounding boxes with high confidence:

$$L_{BB} = \sum_{i \in \{j | g_j > t\}} g_i \tag{5.13}$$

where g_i is the score of every bounding box, for all proposed bounding boxes with confidence larger than the threshold t.

Two-Stage Detectors

For two-stage detectors, DTTACK targets the RPN, classification and regressor. DTTACK uses a pre-defined offset vector used to corrupt the shape of the original proposals from the RPN. Furthermore, DTTACK selects a dense set of proposals and seeks to distort and push the regression of these proposals with a distortion offset vector. This leads the detector to shift the regressed bounding boxes away from the center and distort the aspect ratio.

The final loss function for the DTTACK attack is defined as:

$$\mathcal{L}_{TS} = L_{rpn} + \lambda_1 L_{cls} + \lambda_2 L_{reg} \tag{5.14}$$

Where L_{rpn} is a combination of L_{cls}^{rpn} and L_{reg}^{rpn} from Equation 2.7 and Equation 2.8, respectively. λ_1 and λ_2 are hyperparameters used to tune the weight of the terms in the loss function.

Fooling Detection Alone is Not Enough: First Adversarial Attack against Multiple Object Tracking

Jia *et al.* [38] focuses on object tracking, a vital part of the vision system of autonomous vehicles. They discover that attacks that target object detection alone needs to successfully attack at least 60 frames consecutively to fool a Multiple Object Tracking (MOT) process. The proposed attack in [38], called tracker hijacking, effectively attacks the MOT process using AEs on object detectors.

The tracker hijacking attack generates an adversarial patch, with the goals to

(1) erase the bounding box of the target object, and
(2) fabricate a bounding box with a similar shape as ground truth but shift the location a little towards an attacker-specified direction.

By achieving goal (1), the tracker belonging to the targeted object will be erased. Furthermore, if goal (2) is achieved, the Kalman filter will be disturbed and predict an adversarial movement that is not actually happened. Thus making the tracker hijacking attack capable of mimicking the car moving, which can be used to fake the target changing lanes or leaving the road.

Tracker hijacking utilizes two loss functions, where $\mathcal{L}_1$ is used to minimize the target class probability at the given location to erase the target bounding box (and set $\lambda = 0$ to remove the last term), thus aiming to achieve goal (1).

$\mathcal{L}_2$ controls the fabrication of the adversarial bounding box at the given center coordinates (cx_t, cy_t) with a width and height of (w_t, h_t), where the adversarial bounding box tries to hijack the tracker and thereby achieving goal (2).

Combined, the tracker hijacking aims to optimize Equation 5.15.

$$\min_{\Delta \in patch} \mathcal{L}_1(x_t + \Delta) + \lambda \mathcal{L}_2(x_t + \Delta), \tag{5.15}$$

Where $\mathcal{L}_1$ and $\mathcal{L}_2$ in Equation 5.15 are defined as

$$\begin{aligned} \mathcal{L}_1 &= \sum_{i=0}^{B} \mathbb{1}_i^{obj}[C_i^2 - CrossEntropy(p_i, class_t)] \\ \mathcal{L}_2 &= \sum_{i=0}^{B} \mathbb{1}_i^{obj}\{[(cx_i - cx_t)^2 + (cy_i - cy_t)^2] + \\ &\quad [(\sqrt{w_i} - \sqrt{w_t})^2 + (\sqrt{h_i} - \sqrt{h_t})^2] + \\ &\quad (1 - C_i)^2 + CrossEntropy(p_i, class_t)\} \end{aligned} \tag{5.16}$$

Where $\sum_{i=0}^{B} \mathbb{1}_i^{obj}$ identifies all bounding boxes, B, with their confidence score C_i. The location and shape of the targeted bounding box is given by (x_i, y_i) and (w_i, h_i), respectively. The attack seeks to push the targeted bounding box towards the center location given by (x_t, y_t) and the shape (w_t, h_t). On top of this, $CrossEntropy(p, c)$, see Equation 5.17, is used to calculate the probability of the detector correctly identifying the targeted class, which is used to make the detector mislabel the target object as background. If the attack succeeds by doing this, the detector will fail to track the targeted object, and instead, track the new fabricated object.

$$CrossEntropy(p, c) = -\sum_{x \in \mathcal{X}} p(x) \log(c(x)) \tag{5.17}$$

Jia *et al.* [38] utilizes Adam optimizer [47] to minimize the objective function in Equation 5.15, by iteratively perturbating the pixels within the patch location.

5.5.1 Attack Evaluation

Li *et al.* [21] studies four different types of RPNs, and validates the proposed method on each type of RPN on the COCO dataset with nine object detectors and two instance segmentation methods. This extensive validation indicates the strong transferability of the proposed attack. Li *et al.* [21] conducts experiments and benchmarks the TransRPN attack against DAG [10] and Unified and Efficient Adversary (UEA) [48] attacks. Where TransRPN outperforms both on Faster-RCNN. UEA is only slightly better than TransRPN on SSD300. TransRPN achieves great cross-model transferability for attacks transferred from Faster R-CNN to SSD and YOLOv2 and YOLOv3, by reducing mAP from 0.42, 0.37 and 0.40 to 0.16, 0.04 and 0.05, respectively. This shows the effect of targeting the RPN, which has share properties with the one-stage models, such as SSD and YOLO, leading to improved transferability between one- and two-stage detectors. Furthermore, the experiments discover that more simple backbone networks, such as VGG16, achieve greater transferability and are more robust against the tested defense methods. Li *et al.* [21] also studies the robustness of TransRPN under two scenarios: Adversarial defense and image compression. Where the adversarial defense is a strategy to mitigate the effect of perturbations. They tested TransRPN against the Random Resizing and Padding (RRP) method proposed in [10]. The defense measure only slightly improved the detection performance, leaving the attack effective against this defense. The experiments on image compression show that the detection rate only slightly improved with reduces image quality when under TransRPN attack.

Shi *et al.* [35] conducts experiments and uses DAG, DPatch and R-AP as benchmarks. DTTACK outperforms all three attacks it compares to. Following [22] and [28], [35] focus on Misclassification Rate (MC) and Invisible Rate (L) when generating patches. Furthermore, [35] generates patches with and without limitations, where both versions outperform DAG [10], DPatch [28] and R-AP [22]. The experiments show that making the patches less visible was a harder task than evading the targeted objects.

TransRPN [21] and R-AP [22] has their performance evaluated and reported by mAP, see Table 5.12, where TransRPN achieves to reduce all targets mAP down to 0.00 in white-box settings, thus achieving a 100% mAP drop. Furthermore, Li *et al.* [21] reports the highest drop in mAP of 0.63, this is when attacking a Faster R-CNN detector with the ResNet-152 backbone network.

Li *et al.* [22] also reports strong white-box attack performance for the R-AP attack, with mAP drops of 0.54 and 0.48 on Faster R-CNN detectors with VGG16 and ResNet-152 backbones, respectively.

Table 5.12: Reported mAP@0.5 evaluation for the white-box results of the *Offset-Push Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop of mAP. Bold entries indicates highest attack performance in the table.

Study	Target Detector	Dataset	Benign Result	Adversarial Result	mAP Drop (Reduction in %)
TransRPN [21]	FR-V16	COCO 2014	0.47	0.0	0.47 **(100%)**
	FR-r152		0.63	0.0	**0.63 (100%)**
R-AP [22]	FR-V16	COCO 2014	0.59	0.05	0.54 (91.5%)
	FR-r152		0.65	0.17	0.48 (73.8%)

Shi *et al.* [35] and Jia *et al.* [38] reported their experiments evaluation in Success Rate, defined by the ration of the successfully attacked number of images in the evaluation set, see Table 5.13. Both attacks achieved great attack performance, with a success rate of 97.89% and 98.3%, respectively. Shi *et al.* [35] reported performance with respect to two metrics: Misclassification rate and invisible rate. Their experiments discovers that making the objects evade detection was a harder task than misclassification alone. This is seen in Table 5.13, where the limited perturbation added achieved an 88.51% Success Rate for evasion, as opposed to a 97.89% Success Rate for misclassification.

Table 5.13: Reported evaluation for the white-box results of the *Offset-Push Attacks* who report their experimental results by Success Rate.

Study	Target Detector	Dataset	Success Rate
DTTACK [35]	YOLOv3	COCO 2014	97.89%
			88.51%
Tracker Hijacking [38]	FR-V16	Berkeley Deep Drive [49]	98.3%

Of the four papers, only Li *et al.* [21] and Li *et al.* [22] experimented with black-box settings when transferring the attack, see Table 5.14. Both attacks were tested for cross-task transferability, and achieved a substantial mAP drop when transferred to instance segmentation models. Furthermore, the R-AP attack [22] was transferred across backbones, and the experiments revealed that the attack had poor cross-network transferability, only decreasing the mAP from 0.59 to 0.54 when transferring from Faster R-CNN with a ResNet-101 backbone to a VGG16 backbone. When transferring to RFCN [41], another RPN-dependent detector, and the instance segmentation models Fully Convolutional Instance-Aware Semantic Segmentation Method (FCIS) [50] and Mask R-CNN [51], Li *et al.* [22] discovered that the accumulated perturbation, denoted P, achieved best cross-model and cross-task transferability. However, the attack is not reliable in transfer settings, as the attack achieved a marginal reduction of 21.7% for cross-model attack, and a 24.6% and 26.7% drop in mAP when performing cross-task attacks against FCIS and Mask R-CNN, respectively.

Table 5.14: Reported mAP@0.5 evaluation for the black-box results of the *Offset-Push Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop in mAP. P denotes an accumulated perturbation, from V16, mn, rn50, rn101 and rn152. Bold entries indicates highest attack performance in the table.

Study	**Source Detector**	**Target Detector**	**Benign Result**	**Adversarial Result**	**mAP Drop (Reduction in %)**
TransRPN [21]	FR-V16	SSD	0.42	0.16	0.26 (61.9%)
		YOLOv2	0.37	0.04	0.33 (89.2%)
		YOLOv3	0.40	0.05	0.35 (87.5%)
		Mask-RCNN	0.54	0.02	**0.52 (96.3%)**
		YOLACT	0.49	0.07	0.42 (85.7%)
R-AP [22]	FR-rn101	FR-V16	0.59	0.54	0.05 (8.5%)
	P	RFCN	0.60	0.47	0.13 (21.7%)
		FCIS	0.61	0.46	0.15 (24.6%)
		Mask-RCNN	0.60	0.44	0.16 (26.7%)

5.6 Total Loss Attack

Total Loss Attacks target the entirety of the model, this includes both the regression and classification of RPN, as well as the classification and regression layer of the detector. Such attacks have a large range of results, where the focus is on maximizing the attack output, with no clear target for the attack. Thus making this type of attack results in a variety of attack results, where objects may be evaded or fabricated, and the attack can lead to misclassification of objects within the image.

The total loss function toward Faster R-CNN, Equation 5.18, can be denoted as a combination of the Faster R-CNN loss and the RPN losses from Equation 2.6 and section 2.5, respectively.

$$L = L_{cls}^{FastR-CNN} + L_{reg}^{FastR-CNN} + L_{cls}^{RPN} + L_{reg}^{RPN} \tag{5.18}$$

Adversarial attacks on Faster R-CNN object detector

Wang *et al.* [19] uses PGD [11] to attack Faster R-CNN by targeting the total loss of the object detector. Wang *et al.* [19] conducts a thorough analysis of how the attack performs when targeting separate parts of the loss function, as well as the entire loss function which includes the RPN loss. The experiments reveal that the most effective attack is done by targeting the total loss, including RPN loss, of the target model, which gives the total loss as seen in Equation 5.18.

MI-FGSM on Faster R-CNN Object Detector

Liu *et al.* [25] uses Momentum Iterative Fast Gradient Sign Method (MI-FGSM) to stabilize the optimization and escape from poor local maxima. Compared to PGD [11], which starts from a random value, MI-FGSM is more stable and powerful in both white- and black-box environments.

MI-FGSM utilizes the total loss function from Wang *et al.* [19], as seen in Equation 5.18, which was proved to be the most efficient to generate AEs in white-box settings.

Equation 5.18 is inserted into the attacks loss function, and then the momentum is used to update the direction, thus achieving the property of avoiding occurrences of undesirable local maximums. Liu *et al.* [25] claims this achieves the highest attack performance.

Adversarial Attacks on Faster R-CNN: Design and Ablation Study

Liu *et al.* [26] applies PGD to solve the optimization problem of the attack. The optimization problem includes the objective function of the attack, defined by the model's classification results, coordinates of predicted bounding boxes, region proposals and feature maps. By applying PGD, the constraints of the adversarial perturbations are dealt with.

The ablation study shows that the best attack results come from attacking both the Fast R-CNN submodule and the backbone network, which includes the RPN.

The final loss function of the attack, L_{adv}, is given by the loss function of both the Fast R-CNN submodule and the RPN loss function, as shown in Equation 5.18. Furthermore, Liu *et al.* [26] adds a novel term, L_{bac}, which targets the extracted feature maps of the backbone network. This term is added to the total loss function Equation 5.18, giving the final loss function for the attack as

$$\begin{aligned} L_{adv} = \lambda_1 L_{cls}^{RPN} + \lambda_2 L_{reg}^{RPN} + \lambda_3 L_{cls}^{FastR-CNN} \\ + \lambda_4 L_{reg}^{FastR-CNN} + \lambda_{bac} L_{bac} \end{aligned} \tag{5.19}$$

The ablation study performed for the attack shows that attacking both Fast R-CNN and the backbone network submodule can reduce the detection accuracy for large objects further than only targeting the Faster R-CNN.

Adversarial attacks on YOLACT instance segmentation

Zhang *et al.* [27] extends PGD to attack the instance segmentation model You Only Look At Coefficient (YOLACT) [52].

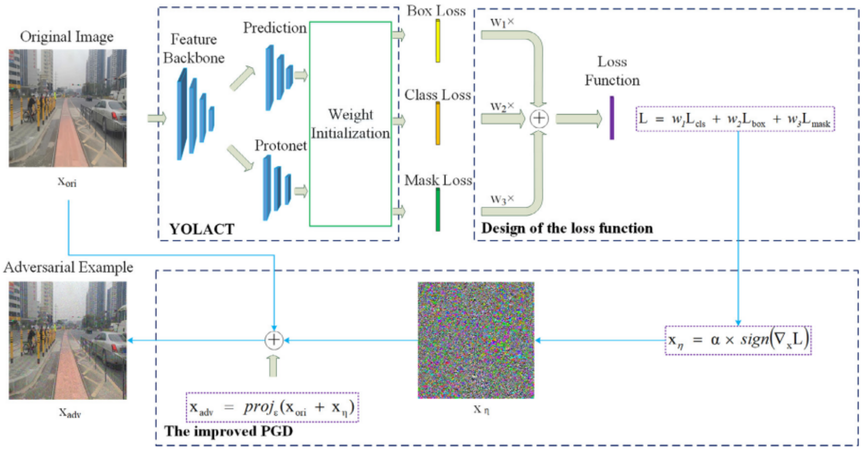


Figure 5.8: Overview, obtained from [27]

Through their testing, they observe that the bounding box regression loss is the most effective loss of the three parts of the total loss function. But all three are slightly worse alone than combined to the total loss. Furthermore, the attack was tested on different network architectures and proved great cross-network transferability.

The total loss function used to generate AEs is the sum of the box regression loss L_{box}, classification loss L_{cls} and the mask loss L_{mask}, where both L_{box} and L_{cls} are defined as in [42], and L_{mask} from [52]. The total loss of the attack is then defined as

$$\begin{aligned} L = w_1 L_{box}(x + r, \theta, l, g) + w_2 L_{cls}(x + r, \theta, c) \\ + w_3 L_{mask}(x + r, \theta, m) \end{aligned} \tag{5.20}$$

where w denotes the different weights of the separate loss functions, r is the generated perturbation, θ is the parameters of the YOLACT model, l is the predicted boxes, g is the ground truth boxes and c is the object confidence.

Zhang *et al.* [27] then uses an improved PGD to maximize Equation 5.20 and generate the AE.

Exploring the Vulnerability of Single Shot Module in Object Detectors via Imperceptible Background Patches

Li *et al.* [29] explores the vulnerability of the Single Shot Module (SSM) commonly used in recent object detectors. For two-stage detectors, like Faster R-CNN, this is the RPN. For one-stage detectors, the SSM is the detectors as a whole. The attack adds a small perturbation to patches within the background of the scene, thus not altering the targeted object directly.

The attacks have similarities to [21], only this extends further to both one- and two-stage detectors, where Li *et al.* [21] only targets the RPN, thus only the two-stage detectors. Furthermore, this attack also impairs the false positives, on top of the true positives, effectively increasing the objectness of the background meanwhile decreasing the objectness of true objects.

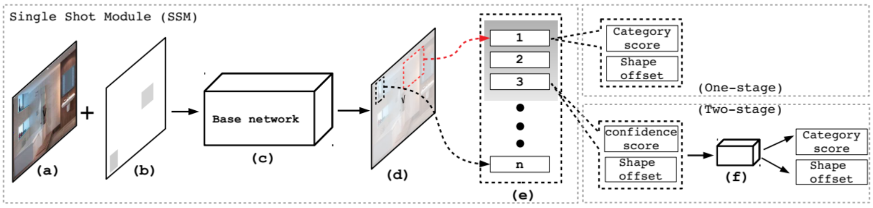


Figure 2. *Overview. (a) Original image. (b) Background patches generated by our method. (c) Base-network, which is the RPN for two-stage object detectors or the single-stage object detector itself. (d) Output of SSM, where the red box and black box denote a false positive and a true positive, respectively. Our attack can disrupt the top ranked results by decreasing true positives and increasing false positives. (e) denotes the top ranked results, which are the object proposals for two-stage object detectors or the detections for single-stage object detectors. (f) Sub-network of two-stage object detectors for class labels prediction and shape refinements.*

Figure 5.9: Overview of the Single Shot Module (SSM), obtained from [29]

An overview of a SSM is presented in Figure 5.9. The goal of targeting the SSM is to corrupt it, such that it can't provide any correct object proposals or detections. This is done by letting the output ranking of the proposals give false positives a higher ranking, such that they are pushed ahead of the true positives.

The attack minimizes the combination of

- True Positive Class loss, correct label loss of the true positives
- True Positive Shape (TPS) loss, which is the correctness of the shape offset regression of the true positives.
- False Positive Class loss, the non-background class scores of false positives arising from the background.

The three above loss functions are combined as terms in a sum, and minimized by IFGS.

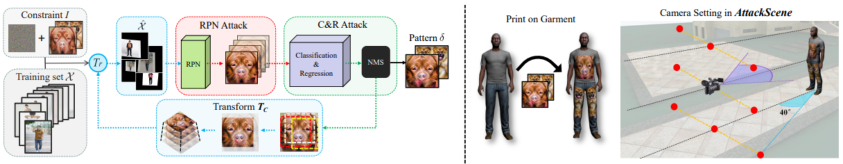
The TPS loss, here given as L_{shape}, is designed to increase the offset, such that the predicted localization is pushed away from the ground truth. This is done similar to how [21] pushes the predicted offset away from a large constant, as described in section 5.5. Instead of a large constant, Li *et al.* [29] pushes the predicted offset away from the ground truth.

The attack is a white-box attack, where the gradients are considered in optimizing the attack. The attack does show some transferability strength between detectors with similar architectures. Nonetheless, the attack barely transfers between one- and two-stage detectors. There is almost no effect when transferring the attack from YOLOv2 or YOLOv3 to Faster R-CNN models, and vice versa. Li *et al.* [29] performs an ablation study to compare the different combinations of the three loss terms, where the three terms combined outperform all other combinations of them.

Universal Physical Camouflage Attacks on Object Detectors

Huang *et al.* [31] proposes Universal Physical Camouflage Attack (UPC), which crafts a camouflage by jointly fooling the RPN, classifier and the regression layer.

UPC generates a universal pattern, which can attack all instances that belongs to the same category, e.g. person or cars. To make this attack work in the physical world, Huang *et al.* [31] proposes to model the deformable characteristics and external physical environments. These steps are summarized in the pipeline shown in Figure 5.10.



(a) Training in *Digital Space*

(b) Attacking in *Physical Space*

Figure 2. The overall pipeline of UPC. (a) training the camouflage patterns in digital space; (b) attacking the target in physical space.

Figure 5.10: Overview of the UPC attack pipeline, obtained from [31]

When targeting the Faster R-CNN detector, the attack aims to reduce the number of valid proposals from the RPN. Then, corrupt the classifier and regressor to output incorrect predictions.

For the RPN-attack, UPC aims to minimize the loss function for RPN, L_{rpn}. Thus, the goal is to generate adversarial patterns for RPN such that the foreground proposals are severely reduced and the proposed candidate boxes are corrupted.

This loss function is added as a term in the total object function, together with terms for the classification and regression layer loss, L_{cls} and L_{reg} respectively. Where UPC selects a dense set of region proposals and aims to corrupt the regression by adding a distortion offset, similar as described in [35].

The attack is conducted in two stages.

- **First stage** focuses on only attacking the RPN to reduce the number of valid proposals.
- **Second stage** attacks the Faster R-CNN classification and bounding box regression tasks.

5.6.1 Attack Evaluation

Wang *et al.* [19] achieves three success cases, which leads to a minimum of one of the class label or bounding boxes being incorrectly proposed. The success cases are summarized in Table 5.15.

Table 5.15: Success cases which includes either or both of successfully attack the bounding box shape/location or the predicted class label.

Success Case	Bounding Box	Class Label
SC1	✓	✗
SC2	✗	✓
SC3	✗	✗

SC2 and SC3 in Table 5.15 are when the attack successfully mislocates the bounding box of the target detector.

Furthermore, the attack does not achieve a 100% success rate, thus leading to some failure cases. The failure cases are where the target manages to correctly detect objects in the AE, both with correctly placed bounding box and correct class label.

Wang *et al.* [19] conducts experiments, where the cross-network transferability of the attack is proven. AEs was generated based on a detector with the backbones VGG-07, VGG-0712, RN-07 and RN-0712. Each of the AEs was detected by the four different detectors with a high success rate. The AEs generated by VGG was also proved to be effective against ResNet, leading to an effective black-box attack.

The attack outperforms the state-of-the-art attack DAG [10], which is reasoned by adding the total loss of the target, including the RPN Regression Loss. Rather than only the classification score in DAG. Wang *et al.* [19] also reports stronger transferability than DAG, due to the extended loss function.

Wang *et al.* [19] failed to find any transferable attack that failed, no matter what the detection model is. These results strengthen the discussed property of AEs which targets the bounding box regression of the targeted model as a commonality across detection architectures.

The experiments [19] show that AEs generated on the total loss, which includes the RPN regression and classification loss, hinder the RPN from generating proposals with high confidence. Nonetheless, the experiments also discover that using only the RPN bounding-box regression loss alone to generate AEs achieves a much lower success rate than with other terms.

Wang *et al.* [19] used DAG as a benchmark, where the proposed attack outperforms DAG in both black- and white-box settings. This was also achieved with fewer iterations.

Liu *et al.* [25] conducts experiments of the MI-FGSM, and benchmarks against the PGD attack. The experiments show that PGD reduces the mAP value of Faster R-CNN with VGG16 backbone to 0.23, while MI-FGSM reduces the mAP further to 0.17, with the same parameters.

The MI-FGSM attack is also tested across different backbones, transferring the attack to a black-box Faster R-CNN with a different backbone. MI-FGSM achieves great transferability which infers that the attack is effective in black-box environments.

Zhang *et al.* [27] conducted experiments for the proposed attack and benchmarked it against FGSM, CI-FGSM and AI-FGSM on image classification tasks. The proposed attack achieved the highest mAP-drop of all benchmarked attacks. On COCO 2017, under white-box attacks, the proposed method achieves 0.01 box mAP and 0.02 mask mAP on YOLACT with ResNet101 backbone.

Huang *et al.* [31] conducts experiments and manages to generate a perturbation that drastically lowers the quality of the proposals from the Faster R-CNN detector. The UPC also outperforms ShapeShifter [53], ERP [54] and AdvPat [55] during the conducted experiments.

Furthermore, Huang *et al.* [31] conducted experiments to test the cross-training transferability and cross-network transferability of the UPC attack. The experiments show that UPC achieves great cross-training strength, but achieves only a slight drop in precision in cross-network transfer attacks.

Five of the six *Total Loss Attacks* presented their evaluation metrics in mAP, as shown in Table 5.16. All show great promise in attack efficiency, especially PGD [19], MI-FGSM [25] and YOLACT [27] which all reported above 90% mAP reduction. Improved PGD [26] and SSM [29] also achieved sufficient attack performance, but less robust, with all evaluations above 30% reduction.

Table 5.16: Reported mAP@0.5 evaluation for the white-box results of the *Total Loss Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop of mAP. Bold entries indicates highest attack performance in the table.

Study	**Target Detector**	**Dataset**	**Benign Result**	**Adversarial Result**	**mAP Drop (Reduction in %)**
PGD [19]	FR-V16	VOC 2007	0.71	0.01	0.70 (98.6%)
	FR-rn101		0.80	0.03	**0.77** (96.3%)
MI-FGSM [25]	FR-V16	VOC 2007	0.70	0.02	0.68 (97.1%)
	FR-rn101		0.75	0.00	0.75 **(100%)**
Improved PGD [26]	FR-rn50	COCO 2017	0.59	0.37	0.22 (37.3%)
YOLACT [27]	FR-rn101	COCO 2014	0.45	0.01	0.44 (97.8%)
	FR-dn53		0.44	0.00	0.44 **(100%)**
	FR-rn50		0.43	0.01	0.42 (97.7%)
SSM [29]	FR-V16	COCO 2014	0.62	0.42	0.20 (32.2%)
	FR-rn152		0.70	0.37	0.33 (47.1%)
	SSD-V16		0.48	0.25	0.23 (47.9%)
	YOLOv3		0.49	0.33	0.16 (32.7%)

The last Total Loss Attack, UPC [31], did not report their evaluation in mAP. Huang *et al.* [31] provided a custom probability metric, Probability Score (PS). The PS represents the probability of whether the detector can hit the true category. Huang *et al.* [31] discovered through their experiments that the joint attack paradigm, which inclusively targets the RPN, achieves stronger attacking strength than when only targeting the classification layer.

Table 5.17: Reported PS on Standard and Adversarial datasets for the UPC attack, where PS represents the probability that the targeted detector will label the object with the correct class. A lower PS indicates stronger attack performance.

Study	**Target Detector**	**Dataset**	**Standard PS**	**Adversarial PS**	**PS Drop (Reduction in %)**
UPC [31]	FR-V16	VOC 0712	0.95	0.04	0.91 **(95.8%)**
	FR-rn101		0.99	0.06	**0.93** (93.9%)

Table 5.18 presents the reported black-box mAP result from [19], [25], [27] and [29]. Where all papers conducted experiments to find the cross-network transferability of their attacks. MI-FGSM in [25] achieved a 100% reduction of mAP when transferring the attack across Faster R-CNN models with different backbone networks. While the rest had difficulties with a more robust transferring of their attacks, where [29] could not transfer the attack across models, nor backbone networks, leaving it improper in black-box settings. Wang *et al.* [19] achieved medium transferability, with a reduction of 35.0% and 22.5% when transferring between ResNet-101 and VGG16 backbones for Faster R-CNN. Zhang *et al.* [27] did extensive testing between several backbones, where almost all transfers achieved above 70% reduction, making the attack invariant to backbones in the target.

Furthermore, the experiments in [19] reveals that the attack achieves high cross-data transferability, with a reduction of 94.7% and 75.0% when transferring the attack from VOC 2007 to VOC 0712 on Faster R-CNN VGG16 and ResNet-101, respectively.

Table 5.18: Reported mAP@0.5 evaluation for the black-box results of the *Total Loss Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop of mAP. Bold entries indicates highest attack performance in the table.

Study	Source	Target	Benign Result	Adversarial Result	mAP Drop (Reduction in %)
PGD [19]	FR-V16	FR-rn101	0.80	0.42	0.38 (47.5%)
	FR-rn101	FR-V16	0.71	0.55	0.16 (22.5%)
	FR-V16 VOC 2007	FR-V16 VOC 0712	0.75	0.04	0.71 (94.7%)
	FR-rn101 VOC 2007	FR-rn101 VOC 0712	0.80	0.20	0.60 (75.0%)
MI-FGSM [25]	FR-V16	FR-rn101	0.75	0.00	**0.75 (100%)**
	FR-rn101	FR-V16	0.70	0.00	0.70 **(100%)**
YOLACT [27]	FR-dn53	FR-rn50	0.43	0.10	0.30 (69.8%)
	FR-rn101			0.03	0.40 (93.0%)
	FR-dn53	FR-rn101	0.45	0.05	0.40 (88.9%)
	FR-rn50			0.12	0.33 (73.3%)
	FR-rn101	FR-dn53	0.44	0.15	0.29 (65.9%)
	FR-rn50			0.13	0.31 (70.5%)
SSM [29]	FR-rn152	FR-V16	0.62	0.62	0.00 (0%)
	SSD-V16			0.60	0.02 (0.3%)
	FR-rn152	FR-rn101	0.66	0.60	0.06 (9.1%)
	YOLOv3	SSD-V16	0.48	0.48	0 (0%)

5.7 Region of Interest Attacks

Region of Interest (RoI) attacks seeks to evade foreground objects by targeting the proposed RoIs. This is done by fabricating false positives which seek to disrupt the RoI by being the only regions that are interesting for further proposals. If the attack succeeds, it will fabricate a strong false positive, whilst evading all actual foreground objects.

DPatch: An Adversarial Patch Attack on Object Detectors

Liu *et al.* [28] proposes the DPatch attack as an extension of the Adversarial Patch attack of Brown *et al.* [56] to attack object detectors, with a focus on YOLO and Faster R-CNN detectors. The original Adversarial Patch showed great attack strength against classifiers and was able to attack them in the real world with a printed physical patch. Liu *et al.* [28] found that to make the patch applicable against object detectors, the bounding box regression and the object classification need to be attacked simultaneously.

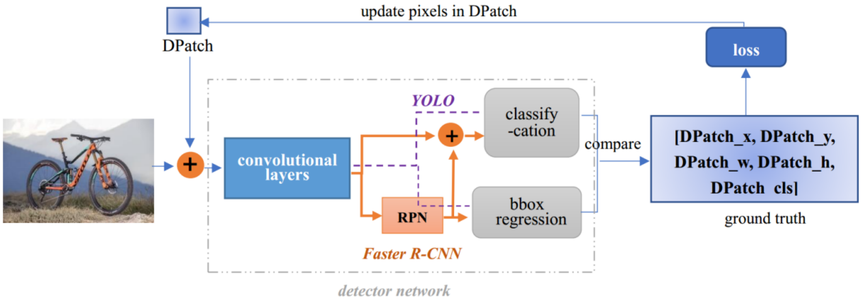


Figure 5.11: Overview of the DPatch attack, obtained from [28]

Liu *et al.* [28] proposes two separate loss functions, extended from Google's Adversarial Patch [56], to perform targeted and untargeted attacks.

DPatch's untargeted and targeted attacks prove to degrade the mAP of Faster R-CNN and YOLO from 0.75 and 0.66 down to below 0.01, respectively. Furthermore, the study shows that a patch achieves cross-model transferability, as a patch trained on YOLO can successfully transfer to attack Faster R-CNN and vice versa.

The attack framework of DPatch is shown in Figure 5.11, where a random patch is applied to the targeted image at the first iteration. Then, depending on the target model, the patch is iteratively updated by targeting the RPN of Faster R-CNN or the classification and regression layer of YOLO.

5.7.1 Attack Evaluation

Concerns about the DPatch attack [28] have been raised, as it shows some weaknesses not disclosed in the paper, and not by the peer reviewers. See Appendix B for further information about the concerns.

DPatch shows very promisingly white-box attack performance from the reported evaluation of Faster R-CNN and YOLOv2 detectors, as shown in Table 5.19. Liu *et al.* [28] performs experiments on both targeted and untargeted attacks, where both attack settings achieve to reduce the mAP of the targeted detectors close to 0.00.

Table 5.19: Reported mAP@0.5 evaluation for the white-box results of the *RoI Attack* DPatch. Lower Adversarial Result means better attack performance, and results in a larger drop of mAP. Bold entries indicates highest attack performance in the table.

Study	Target Detector	Dataset	Benign Result	Adversarial Result	mAP Drop (Reduction in %)
DPatch [28] (Targeted)	FR-rn101	VOC 2007	0.75	0.01	**0.74** (98.7%)
	YOLOv2		0.66	0.02	0.64 (97.0%)
DPatch [28] (Untargeted)	FR-rn101	VOC 2007	0.75	0.03	0.72 (96.0%)
	YOLOv2		0.66	0.00	0.66 **(100.0%)**

Furthermore, [28] performs several black-box attacks with DPatch, where they test for both cross-model and cross-data transferability, as shown in Table 5.20. The experiments validate that a YOLO-trained DPatch can attack Faster R-CNN detectors and vice versa, which is a dangerous and powerful attribute of the attack. Furthermore, the experiments for cross-data transferability indicate that the attack has some more difficulties transferring to another dataset than what the patch is trained on. Nonetheless, a DPatch trained on the COCO dataset in a cross-data setting achieved to drop the mAP of the detectors trained on VOC dataset to 0.28 and 0.24 for YOLO and Faster R-CNN, respectively.

Table 5.20: Reported mAP@0.5 evaluation for the black-box results of the *RoI Attacks*. Lower Adversarial Result means better attack performance, and results in a larger drop in mAP. Bold entries indicates highest attack performance in the table.

Study	**Source**	**Target**	**Benign Result**	**Adversarial Result**	**mAP Drop (Reduction in %)**
DPatch [28]	FR-rn101	YOLOv2	0.66	0.00	0.66 **(100%)**
	YOLOv2	FR-rn101	0.75	0.02	**0.73** (97.3%)
	YOLOv2 COCO	YOLOv2 VOC	0.66	0.28	0.38 (57.6%)
	FR-rn101 COCO	FR-rn101 VOC	0.75	0.24	0.51 (68.0%)

5.8 Attacks Summary

The 16 attacks included in this survey have been clustered into the 4 different categories *Background Evasion*, *Offset-Push*, *Total Loss* and *RoI Attacks*. The identified attributes are explained and described in subsection 4.3.2, and are uniquely distributed for all the attacks, as shown in the two tables Table 5.21 and Table 5.22.

As described in Table 5.21, several attacks can be implemented in black-box settings. Nonetheless, all attacks are white-box based, which implies all attacks can only be executed in black-box settings through a transferring of the attack. How the attack can transfer is denoted by the Transferability attribute, summarized in its respective column in Table 5.21. The Transferability-attribute is given to an attack if they have reported transfer-attack experiments with sufficient attack performance.

For the *Target(s)* attribute, & is used to denote that the attack targets the mentioned modules simultaneously, while ∨ denotes that the attack targets either of the modules.

Table 5.21: Generation and Transfer Phase Attributes for each study, with their respective category from Table 5.7. The attributes are as described in subsection 4.3.2, and the different optimization schemes are introduces through their representative papers discussions. "-" denotes that the paper has not provided enough information to declare which transferability attribute to assign the attack.

		Generation Phase			**Transfer Phase**
Category	**Ref.**	**Attack Knowledge**	**Target(s)**	**Loss Function**	**Transferability**
Background Evasion Attack	AB [20]	White & Black Box	FRCNN	L_{reg}^{FRCNN}	Cross-Model
	CAP [32]	White & Black Box	RPN	L_{reg}^{rpn} & L_{cls}^{rpn}	Cross-Data
	G-UAP [34]	White Box	RPN	L_{cls}^{rpn}	None
	APS [36]	White & Black Box	RPN	L_{reg}^{rpn} & L_{cls}^{rpn}	Cross-Network
	SAA [39]	White & Black Box	FRCNN & YOLOv4	L_{reg}^{FRCNN}	-
Offset-Push Attack	TransRPN [21]	White & Black Box	RPN	L_{reg}^{rpn} & L_{cls}^{rpn}	Cross-Model, -Network and -Task
	R-AP [22]	White Box	RPN	L_{reg}^{rpn} & L_{cls}^{rpn}	None
	DTTACK [35]	White Box	(RPN & FRCNN) $\lor$ YOLOv3	L_{reg}^{rpn} & L_{cls}^{rpn} & L_{reg}^{FRCNN}	None
	Tracker Hijacking [38]	White Box	YOLOv3	L_{reg}^{YOLO} & L_{conf}^{YOLO}	None
Total Loss Attack	PGD [19]	White & Black Box	RPN & FRCNN	L_{reg}^{rpn} & L_{cls}^{rpn} & L_{reg}^{FRCNN}	Cross-Data and -Network
	MI-FGSM [25]	White & Black Box	RPN & FRCNN	L_{reg}^{rpn} & L_{cls}^{rpn} & L_{reg}^{FRCNN}	Cross-Network
	Improved PGD [26]	White Box	RPN & FRCNN	L_{reg}^{rpn} & L_{cls}^{rpn} & L_{reg}^{FRCNN}	None
	YOLACT [27]	White & Black Box	YOLACT	L_{reg}^{YOLACT}	Cross-Network
	SSM [29]	White Box	RPN $\lor$ SSD $\lor$ YOLO	L_{reg}^{rpn} & L_{cls}^{rpn} & L_{reg}^{YOLO} & L_{reg}^{SSD}	None
	UPC [31]	White & Black Box	RPN & FRCNN	L_{reg}^{rpn} & L_{cls}^{rpn} & L_{reg}^{FRCNN}	Cross-Data, -Model and -Network
Region of Interest Attack	DPatch [28]	White & Black Box	FRCNN $\lor$ YOLOv2	L_{reg}^{FRCNN} & L_{reg}^{YOLO}	Cross-Data and -Model

Table 5.22: Deployment Phase Attributes for each study, with their respective category from Table 5.7. The attributes are as described in subsection 4.3.2

		Deployment Phase		
Category	**Ref.**	**Environment**	**Attack Style**	**Optimization Scheme**
Background Evasion Attack	AB [20]	Digital & Physical	Patch-based	l_0
	CAP [32]	Digital	Noise	Gradient (IFGS [13])
	G-UAP [34]	Digital	Noise	Gradient (GD-UAP [57])
	APS [36]	Digital	Noise	GAN
	SAA [39]	Digital	Patch-based	l_0
Offset-Push Attack	TransRPN [21]	Digital	Noise	Gradient (MI-FGSM [46])
	R-AP [22]	Digital	Noise	Gradient (IFGS [13])
	DTTACK [35]	Digital	Patch-based	l_2
	Tracker Hijacking [38]	Digital	Patch-based	Gradient (Adam [47])
Total Loss Attack	PGD [19]	Digital	Noise	Gradient (PGD [11])
	MI-FGSM [25]	Digital	Noise	Gradient (MI-FGSM [46])
	Improved PGD [26]	Digital	Noise	Gradient (PGD [11])
	YOLACT [27]	Digital	Noise	Gradient (PGD [11])
	SSM [29]	Digital	Patch-based	Gradient (IFGS [13])
	UPC [31]	Digital & Physical	Patch-based	Gradient (IFGS [13])
Region of Interest Attack	DPatch [28]	Digital	Patch-based	Gradient (IFGS [13])

5.8.1 Cross-Category Evaluation

While the attacks have been compared within each category, it's important to compare across categories to discover any trends in the strength of the categories. To do so, common evaluation metrics have to be used. Most of the papers provide their experimental results in mAP, which will be used to cross-category evaluate the attacks. This shows the importance of a common agreement on which empiric evaluation metric should be included in all papers, to make it feasible to compare attacks against each other.

Furthermore, given the importance of which datasets are used for training the model and the attack, only the two most common datasets are considered, namely COCO and VOC. All the conducted experiments of the attacks are tested against different models with different configurations and training data, the provided results of the experiments may not be fair to evaluate against each other, given their difference in performance under normal circumstances. Hence, the evaluation is clustered on the targeted models, to provide an overview of how well the different attacks manage to transfer from their source to target, meanwhile also providing a useful overview of how the most common detectors react to attacks.

Given that four of the studies ([31], [35], [38] and [39]) did not provide their evaluations i mAP and Yuan and Wei [36] evaluated APS on a different dataset from VOC and COCO, their attacks are excluded from the following cross-category evaluation. This was due to their evaluations being infeasible to compare against the remaining attacks. Thus, the following evaluation only considers 11 of the 16 attacks.

Cross-Category White-Box Evaluation

Table 5.23 summarizes the results of all white-box experiments conducted for the attacks that provide their empirical data in mAP. Table 5.23 describes the attacks whose experiments were reported in mAP@0.5. For simplicity of the table, COCO 2014 and COCO 2017 dataset is denoted as C14 and C17, respectively. And VOC datasets are denoted as V07 and V0712 for VOC 2007 and VOC 0712, respectively.

Table 5.23: White-Box cross-category evaluation for the studies which reported experimental results in mAP@0.5 scores. All scores are given as drop in mAP, and relative mAP percentage reduction. U: Untargeted, T: Targeted

Attack	Target Detector										
	FR-V16-C14	FR-V16-V07	FR-V16-V0712	FR-rn50-C14	FR-rn50-C17	FR-rn101-C14	FR-rn101-V07	FR-rn101-V0712	FR-rn101-C17	FR-rn152-C14	FR-dn53-C17
PGD [19]		0.70 (98.6%)	0.73 (97.3%)				0.72 (94.7%)	0.77 (96.3%)			
TransRPN [21]	0.47 (100%)			0.58 (100%)		0.62 (100%)				0.63 (100%)	
R-AP [22]	0.54 (91.4%)			0.49 (82.4%)		0.47 (73.5%)				0.48 (73.3%)	
MI-FGSM [25]		0.70 (99.8%)					0.75 (99.5%)				
Improved PGD [26]					0.22 (37.4%)						
YOLACT [27]					0.42 (98.3%)				0.44 (97.5%)		0.44 (99.8%)
DPatch (U) [28]							0.72 (96.1%)				
DPatch (T) [28]							0.74 (98.7%)				
SSM [29]	0.21 (32.9%)			0.25 (38.5%)		0.30 (45.2%)				0.33 (47.4%)	
CAP [32]		0.74 (100%)					0.77 (98.0%)				
G-UAP [34]		0.40 (56.0%)	0.42 (55.5%)				0.25 (33.6%)	0.27 (33.8%)			

Table 5.23: (*Continuation*) White-Box cross-category evaluation for the studies which reported experimental results in mAP@0.5 scores. All scores given as drop in mAP, and relative mAP percentage reduction. U: Untargeted, T: Targeted

Attack	Target Detector										
	FR-mn-C14	FR-mn-V07	SSD-V16-C14	SSD-rn50-C14	YOLOv2-V16-V07	YOLOv2-mn-C14	YOLOv3-mn-C14	RFB-V16-C14	RFB-rn50-C14	FSSD-V16-C14	FSSD-rn50-C14
PGD [19]											
TransRPN [21]											
R-AP [22]	0.36 (76.6%)										
MI-FGSM [25]											
Improved PGD [26]											
YOLACT [27]											
DPatch (U) [28]					0.66 (100%)						
DPatch (T) [28]					0.64 (97.2%)						
SSM [29]	0.20 (42.3%)		0.24 (49.3%)	0.19 (40.1%)		0.24 (52.1%)	0.16 (32.0%)	0.22 (46.2%)	0.23 (46.6%)	0.21 (38.0%)	0.22 (43.8%)
CAP [32]		0.47 (93.4%)									
G-UAP [34]											

Of these white-box evaluations in Table 5.23 TransRPN [21] and MI-FGSM

[25] from *Offset-Push*, YOLACT [27] from *Total Loss*, untargeted DPatch [28] from *Region of Interest* and CAP [32] from *Background Evasion* all achieve some very close to or actual 100% reduction in mAP. Thus reporting the highest mAP reduction of the white-box attacks. The distribution of highest performing attacks is relatively even among the four categories, thus not showing any clear trends from white-box evaluation alone.

Cross-Category Black-Box Evaluation

For the black-box evaluation of the attacks, the reported cross-data, -model, -network and -task performance is summarized in Table 5.24. Given that many of the attacks report high mAP reduction of more than 80%, attacks failing to reduce any performance by more than 40% is deemed insufficient for that style of transfer.

Table 5.24: Black-Box cross-category evaluation for the studies which reported experimental results in mAP@0.5 scores, denoted as mAP drop and relative percentage reduction. U: Untargeted.

Attack	**Source Detector**	**Target Detector**										
		FR-V16-C14	FR-V16-V07	FR-V16-V0712	FR-rn50-C14	FR-rn50-C17	FR-rn101-C14	FR-rn101-V07	FR-rn101-V0712	FR-rn101-C17	FR-rn152-C14	FR-dn53-C17
PGD [19]	FR-V16-V07			0.71 (94.4%)				0.29 (38.8%)	0.27 (34.4%)			
	FR-V16-V0712		0.63 (88.7%)					0.34 (44.7%)	0.38 (47.1%)			
	FR-rn101-V07		0.19 (27.1%)	0.17 (22.1%)					0.60 (75.4%)			
	FR-rn101-V0712		0.20 (28.3%)	0.21 (27.4%)				0.63 (83.2%)				
TransRPN [21]	FR-V16-C14				0.56 (96.6%)		0.59 (95.2%)				0.59 (93.7%)	
	FR-rn50-C14	0.41 (87.2%)					0.60 (96.8%)				0.60 (95.2%)	
	FR-rn101-C14	0.39 (83.0%)			0.56 (96.6%)						0.61 (96.8%)	
	FR-rn152-C14	0.39 (83.0%)			0.56 (96.6%)		0.60 (96.8%)					
R-AP [22]	FR-V16-C14 (p_1)				0.12 (19.5%)		0.11 (17.0%)				0.09 (14.4%)	
	FR-mn-C14 (p_2)	0.02 (4.1%)			0.03 (4.7%)		0.03 (4.6%)				0.03 (3.9%)	
	FR-rn50-C14 (p_3)	0.05 (9.1%)					0.11 (16.9%)				0.09 (13.7%)	
	FR-rn101-C14 (p_4)	0.04 (7.4%)			0.10 (16.0%)						0.09 (13.6%)	
	FR-152-C14 (p_5)	0.04 (7.1%)			0.10 (16.3%)		0.10 (15.6%)					
	P= $\sum_{i=1}^{5} p_i$	0.22 (36.7%)			0.28 (47.4%)		0.26 (40.3%)				0.23 (36.1%)	
MI-FGSM [25]	FR-V16-V07							0.75 (99.4%)				
	FR-rn101-V07		0.70 (99.5%)									
YOLACT [27]	rn50-C17									0.33 (72.8%)		0.30 (69.5%)
	rn101-C17					0.33 (75.8%)						0.29 (65.4%)
	dn53-C17					0.40 (93.0%)				0.39 (87.1%)		
DPatch (U) [28]	FR-rn101-V07											
	YOLOv2-V16-V07							0.73 (97.7%)				
	FR-rn101-C14							0.47 (62.7%)				
	YOLOv2-V16-C14											
SSM [29]	FR-V16-C14				0.03 (4.8%)		0.03 (4.5%)				0.02 (3.1%)	
	FR-rn50-C14	0.02 (3.4%)					0.04 (6.1%)				0.02 (3.4%)	
	FR-rn101-C14	0.00 (0.0%)			0.04 (6.2%)						0.04 (5.3%)	
	FR-rn152-C14	0.01 (0.8%)			0.05 (7.3%)		0.07 (9.8%)					
	SSD-v16-C14	0.02 (3.4%)			0.01 (0.8%)		0.01 (1.4%)				0.01 (0.7%)	
	SSD-rn50-C14	0.01 (1.6%)			0.00 (0.0%)		0.01 (0.9%)				0.00 (0.0%)	
	YOLOv2-mn-C14	0.01 (1.3%)			0.00 (0.0%)		0.00 (0.0%)				0.00 (0.0%)	
	YOLOv3-mn-C14	0.01 (1.1%)			0.00 (0.0%)		0.01 (2.0%)				0.00 (0.0%)	
G-UAP [34]	FR-V16-V07			0.42 (55.0%)				0.19 (24.7%)	0.16 (20.6%)			
	FR-V16-V0712		0.43 (60.0%)					0.20 (26.0%)	0.18 (21.9%)			
	FR-rn101-V07		0.36 (51.0%)	0.31 (41.5%)					0.23 (28.7%)			
	FR-rn101-V0712		0.36 (50.4%)	0.32 (41.6%)				0.29 (38.0%)				

Table 5.24: (*Continuation*) Black-Box cross-category evaluation for the studies which reported experimental results in mAP@0.5 scores, denoted as mAP drop and relative percentage reduction. U: Untargeted.

Attack	**Source Detector**	**Target Detector**									
		FR-mn-C14	SSD-V16-C14	SSD-V16-V07	SSD-rn50-C14	SSD-mn-C14	YOLOv2-V16-V07	YOLOv2-mn-C14	YOLOv3-mn-C14	RFCN-rn101-C14	RFCN-rn101-V07
PGD [19]	FR-V16-V07										
	FR-V16-V0712										
	FR-rn101-V07										
	FR-rn101-V0712										
TransRPN [21]	FR-V16-C14		0.35 (85.4%)		0.26 (61.9%)	0.23 (67.6%)		0.33 (89.2%)	0.35 (87.5%)		
	FR-rn50-C14		0.20 (48.8%)		0.11 (26.2%)	0.11 (32.4%)		0.21 (56.8%)	0.23 (57.5%)		
	FR-rn101-C14		0.20 (48.8%)		0.11 (26.2%)	0.11 (32.4%)		0.20 (54.1%)	0.21 (52.5%)		
	FR-rn152-C14		0.21 (51.2%)		0.11 (26.2%)	0.11 (32.4%)		0.22 (59.5%)	0.23 (57.5%)		
R-AP [22]	FR-V16-C14 (p_1)	0.12 (26.1%)								0.06 (9.3%)	
	FR-mn-C14 (p_2)									0.03 (4.3%)	
	FR-rn50-C14 (p_3)	0.08 (16.1%)								0.06 (10.6%)	
	FR-rn101-C14 (p_4)	0.06 (13.0%)								0.08 (13.5%)	
	FR-152-C14 (p_5)	0.05 (11.3%)								0.06 (9.3%)	
	P= $\sum_{i=1}^{5} p_i$	0.21 (43.9%)								0.13 (21.8%)	
MI-FGSM [25]	FR-V16-V07										
	FR-rn101-V07										
YOLACT [27]	rn50-C17										
	rn101-C17										
	dn53-C17										
DPatch (U) [28]	FR-rn101-V07						0.66 (100%)				
	YOLOv2-V16-V07										
	FR-rn101-C14										
	YOLOv2-V16-C14						0.41 (63.0%)				
SSM [29]	FR-V16-C14		0.02 (3.3%)		0.00 (0.0%)			0.02 (4.5%)	0.00 (0.0%)		
	FR-rn50-C14		0.01 (1.0%)		0.00 (0.0%)			0.01 (2.4%)	0.01 (1.0%)		
	FR-rn101-C14		0.01 (1.2%)		0.00 (0.0%)			-0.01 (-1.1%)	0.01 (1.4%)		
	FR-rn152-C14		0.01 (1.4%)		0.00 (0.0%)			0.02 (3.9%)	0.00 (0.0%)		
	SSD-v16-C14				0.01 (2.6%)			0.00 (0.0%)	0.01 (1.4%)		
	SSD-rn50-C14		0.01 (1.7%)					0.00 (0.0%)	0.00 (0.0%)		
	YOLOv2-mn-C14		0.01 (1.0%)		0.00 (0.0%)				0.03 (6.9%)		
	YOLOv3-mn-C14		0.01 (1.0%)		0.00 (0.0%)			0.07 (14.4%)			
G-UAP [34]	FR-V16-V07			0.08 (9.9%)							0.23 (31.7%)
	FR-V16-V0712			0.10 (13.0%)							0.24 (33.1%)
	FR-rn101-V07			0.07 (8.7%)							0.26 (34.7%)
	FR-rn101-V0712			0.07 (8.5%)							0.28 (38.5%)

Firstly, one can observe from Table 5.24 that G-UAP [34], R-AP [22] and SSM [29] struggle to generalize the attack, as the mAP reduction for some of the transfer attacks is not sufficient and far away from their white-box performance. However, PGD [19] manages to transfer the attack cross-data, seemingly invariant to the data used in the detector and attack.

Both G-UAP [34] and PGD [19] only successfully transfer the attack cross-data and -network to the Faster R-CNN target with VGG16 backbone network. Notably, R-AP [22] achieves an acceptable reduction in mAP when combining all perturbations from the five different detectors. SSM [29] has almost no impact when transferring the attack, and even provides one unique case where the mAP is increased instead of reduced during the attack.

Total Loss reported 3 out of 4 successfully transfer attacks, while *Background Evasion* and *Offset-Push* only had 1 out of 2 successfully transfer attacks. The conducted *Region of Interest* transfer attacks shows the most successful cross-model transfer attacks of all categories, while the cross-data attacks achieve a lower mAP reduction. Thus, *Total Loss* seems to have the highest chance of transferring the attacks across datasets, networks and models. Nonetheless, none of the *Total Loss Attacks* reported a successful cross-task transfer attack.

An interesting observation is that the experiments conducted in TransRPN [21] show that attacks based on less complex backbones, here VGG16, achieve higher attack performance against detectors, defense strategies and achieve better transferability across networks and models. Zhang *et al.* [27] reports similar findings, where AEs generated on Faster R-CNN detectors with DarkNet-53 backbone network is more valid to attack ResNet backbones than the other way around. Li *et al.* [21] also reports experiments that indicate that the VGG16 backbone outperforms ResNet backbones when transferring the attack from Faster R-CNN to the one-stage detectors SSD and YOLO, as shown in Table 5.24.

5.9 Classification of Defenses

To discuss and answer RQ3, the five selected defensive papers for this survey are considered. Two main classes of defensive strategies are denoted as *proactive* and *reactive* strategies.

Proactive strategies seek to preemptively make the detectors robust such that the AEs can be input to the main detector with neglected effect. The most common proactive strategy is adversarial training, which adds AEs to the training data, as described in section 2.10.

The reactive defenses focus on adding an intermediate module that intercepts the input image to conduct the security measure before the image is input to the main detector. The reactive strategies thus seek to disturb the perturbations and thus prevent the AEs to be input to the main detector as intended by the attacker. Two common strategies for this are adversarial detection and denoising.

Of the five papers, two are proactive and propose methods of adversarial training, while the last three were reactive strategies, as shown in Table 5.25.

Table 5.25: The five defenses discussed through this chapter.

	Defense methods	
Study	**Reactive**	**Proactive**
T-SAT [23]		✓
OR [24]	✓	
TOR [30]		✓
DR [33]	✓	
FUSE [37]	✓	

5.10 Proactive Defenses

Proactive defense mechanisms are implemented in the training and design phase of the detector, thus seeking to create a robust detector. The proactive defenses seek to reduce the overhead when the robust detector first is deployed, as there is no intermediate between the input and the detector.

Despite reducing the overhead in run-time, a common weakness of proactive defenses is their need for information about current attacks. While adversarial training has shown great potential, new attacks may bypass them as the detector has not seen the perturbations of these new attacks before. Thus creating a constant need for re-training to maintain the performance of the defenses against new attacks.

Towards Practical Robustness Improvement for Object Detection in Safety-Critical Scenarios

Hu and Zhong [23] proposes a Two-Stage Adversarial Training (T-SAT) algorithm to increase the robustness of state-of-the-art object detectors practically. Their focus is on adversarial training of a YOLOv3 [5] detector on the COCO dataset. This is barely done before, as most adversarial training has been focused on both smaller networks and datasets.

Furthermore, the proposed adversarial training method focus on attacks that can either make the target misclassify the main objects, or ignore the main objects in the image. This is a cause of the adversarial training utilizing the PGD [11] attack to generate the AEs used in training, which is mainly for classification attacks.

Generally, adversarial training tries to solve the min-max problem described in section 2.10. With this in mind, the proposed adversarial training algorithms consist of two main parts:

1. **Adversarial Example Generation**
   - Where the method tries to solve the inner maximization problem of Equation 2.9 with the PGD method.
2. **Training**
   - Where the method tries to solve the outer minimization problem of Equation 2.9 on the AEs generated in the previous step.

As an extra measure against overfitting, the method freezes the convolution layers and trains the output layers for part of the training, and re-train from the weights of the frozen layers and trains the whole model with a smaller learning rate.

The proposed T-SAT method is tested against the one proposed by Zhang and Wang [30], where T-SAT achieves a higher mAP than the method proposed by Zhang and Wang [30].

The robust model is trained on AEs generated by PGD, [23] also studies the robustness of their new model against other white-box noise attacks, namely C&W [58] and FGSM [12]. Where C&W is an iterative attack, like PGD, and FGSM in a one-step gradient attack. The robust model shows little to no robustness against the C&W attack, which may imply that the defense can't transfer to other iterative attacks. Another interesting result of the conducted experiments is that a robust model trained on "weak" PGD examples is more robust against FGSM than a model trained on stronger PGD examples. This indicates that the generated AEs used in the training step needs to replicate the attack it is trying to mitigate, which indicates that there is a lack of generalization of the defense.

The defense theoretically would expect to defend against several of the attacks examined in this study, as there are multiple Background Evasion, Offset-Push and Total Loss attacks which use PGD to solve their objective function, as shown in Table 5.22, and more having the attributes of Digital and Noise while solving their objective function iterative. Nonetheless, the initial experiments in [23] show that there is still further work to be done to have the robust model more generalized and able to defend against these attacks.

Towards Adversarially Robust Object Detection

Zhang and Wang [30] proposes TOR, a practical approach for achieving adversarial robustness. This is done by

1. Categorization and analysis of different attacks, to reveal the underlying mechanism.
2. Analyzes the different impacts of the task losses of the attacks.
3. Generalize adversarial training from classification to detection.

Zhang and Wang [30] discovers cross-task transference when targeting the attack to the classification or localization tasks of the detector. This was discovered through experiments, where they isolate the impact of class and localization loss and observe that only using the localization loss affects the pure classification task. Likewise, only using the class-loss decreases the accuracy of the localization task of an object detector.

Zhang and Wang [30] proposes the following definition of adversarial training:

$$\underset{\theta}{min}[\max_{\bar{x}\in\mathcal{S}_{cls}\cup\mathcal{S}_{loc}} \mathcal{L}(f_\theta(\bar{x}),\{y_k,b_k\})] \tag{5.21}$$

where the loss function, $\mathcal{L}(\cdot)$, is defined as the combination of classification and localization loss, as shown in Equation 5.22.

$$\min_\theta loss_{cls}(f_\theta(x),\{y_k,b_k\}) + loss_{loc}(f_\theta(x),\{y_k,b_k\}), \tag{5.22}$$

The inner maximization of Equation 5.21 is approximately solved by FGSM. The min-max problem in Equation 5.21 differs from the standard min-max described in section 2.10 by splitting into task-oriented domains S_{cls} and S_{loc}. These task-oriented domain constraints maximize either the classification loss or the localization loss, then the AE used in the adversarial training is the one that maximizes the overall loss. By doing so, the proposed method isolates the effect of the classification- and localization-loss, without suffering from the inferences between the two tasks.

The proposed robust model was tested across SSD [42], Receptive Field Block-based Detector (RFB) [59], Feature Fusion Single Shot Multibox Detector (FSSD) [60] and YOLO [5, 61] detectors with various backbone networks. The robust model achieved a substantially higher mAP than the standard models against R-AP attacks, see Table 5.26. Anyhow, one of the challenges of adversarially trained models becomes clear, the mAP of the clean images is reduced by 0.26 on average, and the reported clean mAP drops from 0.72 on the standard model, to 0.46 for the robust model.

Table 5.26: The evaluation of experiments conducted by Zhang and Wang [30] against the R-AP attack [22]. STD denotes the performance of the standard detector

	R-AP	
Architecture	**STD**	**Robust model from [30]**
SSD-V16	0.07	0.45
RFB-rn50	0.09	0.49
FSSD-dn53	0.08	0.47
YOLO-dn53	0.08	0.44

Zhang and Wang [30] compares the proposed min-max problem, shown in Equation 5.21, against the standard definition, shown in Equation 2.9. Where the adversarial training with the proposed min-max contributes to a better mAP than the standard definition. Implying the interference between the two tasks has a negative impact on the robustness gained through adversarial training.

5.11 Reactive Defenses

Reactive defenses seek to intercept the pipeline of the system and conduct security measures for all inputs before it is input to the detector. These security measures include adversarial detection and denoising. For adversarial detection, the target system consists of a detector with the sole role of classifying the input as benign or adversarial. Denoising seeks to remove parts of the potential perturbations, effectively disabling the perturbation and increasing the likelihood of the main detector providing the correct prediction.

These reactive defenses have their positives and negatives, where the main positive is reducing the need to re-training the main detector as needed in several of the proactive defenses, which adds overhead in the design and training phase. The reactive defenses can also be added as a layer of defense on top of a robust detector that has been secured with proactive measures, leaving an even more rigorous defense. The main negative of reactive defenses is that they only react in response to known attacks, which makes them susceptible to new attacks.

The reactive defense can be an isolated module of the pipeline, and improving or changing this module does not need to affect the main detector. The reaction to the input does add overhead in run-time, which can turn the defense method infeasible in real-time demanding scenarios.

Improving Adversarial Robustness of Detector via Objectness Regularization

Bao *et al.* [24] argues that adversarial training is only efficient against attacks with perturbations of small l_2 and l_∞ norm. This is not sufficient against patch-based attacks, as they often have a very large l_∞. Thus, they propose an Objectness Regularization (OR) method to defend against patches that seek to evade objects from detection. OR also achieve a proper trade-off between robustness and clean image detection performance.

The objectness score of bounding boxes represents the confidence that the bounding box contains a foreground object. In addition, objectness can be represented by the sum of all foreground object probabilities.

The impact of evasion attacks is defined as:

1. The patch reduces the objectness of the image
2. The patch compresses the objectness to a small range, such that the difference between foreground and background become smaller.

Thus, to defend against this, OR aims to increase the overall objectness of the image, such that attacks struggle to reduce the objectness of the objects. This is done by applying the objective function, shown in Equation 5.23, before the final classification and regression. This is further visualized in Figure 5.12.

$$f'_{obj}(x) = r * S(f_{obj}(x)) + b \tag{5.23}$$

Where $f_{obj}(x)$ denotes the objectness features of an image x, and $S(\cdot)$ denotes the boosting function chosen as the sigmoid function, described in Equation 5.24, which maps the objectness features from 0 to 1. r and b are regularization parameters, which is necessary for balancing the objectness regularization to avoid increasing false predictions in normal circumstances. This is done by having both r and b tightly connected to the objectness threshold τ, which is used to remove redundant and false bounding boxes during the post process of the detector. r controls the range of the regularization, and b controls the lowest prediction objectness.

$$S(t) = \frac{1}{1 + e^{-t}} \tag{5.24}$$

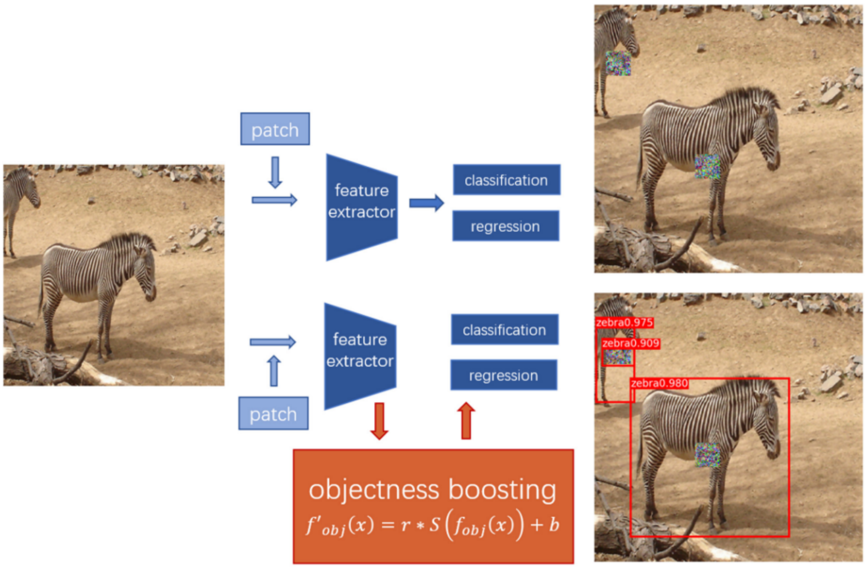


Figure 5.12: Overview of OR, obtained from [24]

Bao *et al.* [24] conducts experiments to test the OR method against Targeted Adversarial Objectness Gradient (TOG) [62] and SAA [39]. In their experiments, the strongest implemented SAA achieved a Vanishing Rate of 91.0% and 78.3% on YOLOv3 and YOLOv4, respectively. When applying OR to both models, the robust YOLOv3 [5] and YOLOv4 [40] achieved a superior Vanishing Rate of 42.2% and 37.0%, respectively.

Another observation from the data represented in [24] is that for weaker SAA patches, the OR method had a very small impact in comparison to the strong attacks. For a patch of size 2500 pixels and 100 attack iterations, the vanishing rate only decreased by 14.1% and 2.9% when applying OR to YOLOv3 and YOLOv4, respectively. Thus indicating that the OR method does not generalize well to weaker attacks.

Detection as Regression: Certified Object Detection by Median Smoothing

Chiang *et al.* [33] presents Detection as Regression (DR) as a reduction from object detection to a regression problem, then enables certified regression. The regression problem envelops the proposal, classification and Non-max suppression (NMS) stages of the detection task. Furthermore, the certified regression is enabled by the proposal of median smoothing.

Chiang *et al.* [33] argues that the defense proposed by Zhang and Wang [30] would fail against stronger, more sophisticated attackers and that their proposed DR can guarantee robustness against all possible attackers within the threat model.

DR proposes a set of 16 bounding boxes, containing all combination of a lower bound, $(\underline{x}_1, \underline{y}_1, \underline{x}_2, \underline{y}_2)$, and upper bound, $(\bar{x}_1, \bar{y}_1, \bar{x}_2, \bar{y}_2)$, bounding box set. Then, the worst case bounding box, defined as the bounding box which achieves the lowest IoU with the ground-truth box, is selected. As long as the worst-case IoU is above the NMS threshold, τ, the bounding box is considered certifiably correct.

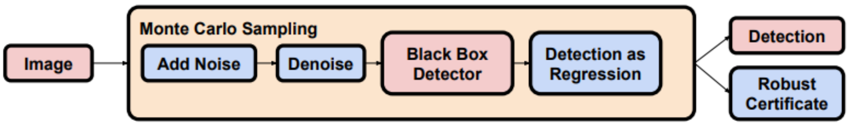


Figure 5.13: Overview of the proposed method, obtained from [33]

DR is implemented as a certifiably-robust wrapper, as shown in Figure 5.13, of different black-box detectors. The wrapper estimates the bounding boxes through an ensemble of proposals and majority voting. The median smoothing is utilized to minimize the effect of outliers. The robust wrapper outputs both the detection and a robust certificate. The certificate annotates bounding boxes which can be distorted or evaded with a perturbation with $||\delta||_2 < 0.36$, where δ is bounded by the l_2-norm.

Robust Object Detection Fusion Against Deception

Chow and Liu [37] proposes a deception-resilient detection fusion approach they name FUSE. FUSE utilizes a fusion framework, using the detection outputs of objectness fusion, bounding box fusion and classification fusion.

The bounding box fusion in FUSE combines all bounding boxes of multiple detectors: The victim detector and the verification detectors. This is done such that false proposals can be discarded and positive proposals are kept for the classification fusion to ensure the correct label are assigned.

To ensure a robust model, FUSE aims to ensemble a diverse set of detectors with different backbones. To further ensure diversity, FUSE minimizes the correlation between the selected detectors.

Furthermore, FUSE aims to maintain the performance on the clean images, an improvement of the mAP drop on benign examples discussed in [30].

Chow and Liu [37] conducts experiments with an ensemble of 11 detectors, which consists of Faster R-CNN, YOLOv3 and SSD detectors with different backbones.

Furthermore, Chow and Liu [37] conducts experiments against R-AP [22], where the R-AP attack achieves to reduce the mAP of the victim Faster R-CNN detector from 0.67 to 0.05. The adversarial training method proposed by Zhang and Wang [30] was also tested, where the robust model achieves a mAP of 0.36 on the same benign set (reduction of more than half from the standard detector). The robust model achieves a mAP of 0.36 on the R-AP attack. FUSE manages to maintain the performance on the benign set, with a mAP of 0.81. FUSE also mitigates the R-AP attack by achieving 0.77 mAP during the attack.

Furthermore, FUSE is tested against the vanishing TOG attack [62], where FUSE mitigates the attack fully. Chow and Liu [37] also claims to be effective against other digital evasion patch attacks, thus inclining to be effective against DPatch [28] and SAA [39].

5.12 Defense Summary

Five different defense approaches have been presented with methods strategies as adversarial training, objectness regularization, detection fusion and detection certification.

The defenses performed experiments on different attacks, where only three of the attacks in this survey are mitigated: DPatch [28], SAA [39] and R-AP [22].

The defenses also claim to have an effect against attacks beyond those that are mitigated in the papers. This theoretical mitigation potential is mapped to the attacks in this survey, and the proposed possible mitigations are summarized in Table 5.27. These proposed possible mitigations are derived by mapping the properties that the defenses claim to be effective against and the attributes of the attacks described in Table 5.21 and Table 5.22.

Table 5.27: Summarized mitigation of attacks of the five defenses discussed through this chapter. Theoretical effect denotes attacks which is of interest to conduct experiments on, as the defense has described features which may make them robust against these attacks. Bold entries in Mitigated Attacks are the relevant attacks discussed in this survey.

Study	Mitigated Attacks	Theoretical Mitigation Potential
T-SAT [23]	C&W [58], FGSM [12]	PGD-based attacks (PGD [19], Improved PGD [26] and YOLACT) [27]
OR [24]	**SAA [39]**	Vanishing Patches (DTTACK [35], Tracker Hijacking [38], AB [20], DPatch [28])
TOR [30]	**R-AP [22]**, DAG [10]	Attacks which targets both localization and classification (*Total Loss Attacks*)
DR [33]	DAG [10]	l_2-bounded attacks (DTTACK [35])
FUSE [37]	**R-AP [22], DPatch [28]**, DAG [10], TOG [62], UEA [48], Patch of [63]	Digital evasion patches (SAA [39], AB [20])

Collectively, the defenses discussed in this survey have a theoretical potential to mitigate most of the discussed attacks, though many of them are still theoretical and has not been verified through experiments. The remaining attacks are CAP [32], G-UAP [34], APS [36] and TransRPN [21]. From Table 5.21 and Table 5.22 one can observe that these four attacks share some important features which it does not seem to exist any effective mitigation against, including they target RPN solely. This shows a clear trend: *There are no defenses focusing on mitigating attacks which targets the RPN.*

Chapter 6

Discussion

This chapter will discuss the novelty and contributions of this survey compared to the related work. Then the potential usage of the results will be discussed along with proposed future work for academia and industry. Lastly, any known threats to the validity of this thesis will be disclosed.

6.1 Comparison with related work

A novel taxonomy has been proposed in through this survey, which has been used to classify 16 different attacks within the four separate categories. Furthermore, these attacks have been examined against existing defenses. This taxonomy differs from the related work described in section 3.2, first of all by targeting object detectors, and also by focusing on the region proposals of the detectors. Lastly, the taxonomy covers the generalization of the attacks, used to describe how well they transfer across targets. This attribute has not been found in any other taxonomies, still being a very important aspect of attacks to further research and hopefully one day mitigate.

Of the related work, Pitropakis *et al.* [15] proposed a taxonomy for classifiers, where the Attacker Knowledge is the only shared attribute with this survey. The taxonomy targeted general attacks on machine learning and was only regarding image classifiers for computer vision tasks. This makes several of the attributes irrelevant for AE attacks against object detectors.

Serban *et al.* [16] introduces taxonomy for AE attacks against object detectors, and only shares the Attacker Knowledge attribute with the proposed taxonomy in this thesis. The Attack Strategy attributes also has similarities, where Serban *et al.* [16] includes noise-based and geometric-based attacks. For the taxonomy of defenses, the reactive defenses discussed in this thesis have similarities to the guards discussed by Serban *et al.* [16], where the reactive defenses are placed early in the pipeline with the task of responding to potential attacks. While Serban *et al.* [16] provides a more comprehensive categorization of defenses, the defense strategies discussed in this survey is categorized by the more abstract classes *reactive* and *proactive*. None of the defenses in this thesis is covered by the literature study in [16].

The classification given by Liu *et al.* [17] does not share any attributes for the taxonomy of attacks. For the defenses, similarities are found in both taxonomies discussing the adversarial training strategy, and defensive methods which require ensembles of detectors.

Lastly, the literature study and survey of Kong *et al.* [14] focuses only on how to attack classifiers but shares the discussion of the attacker's knowledge and physical attacks. Nonetheless, the survey does not extend to object detectors.

While the related work proposes taxonomies with some similar attributes, the taxonomy proposed during this survey adds a novel focus on the generalization of the attacks. As discussed throughout the survey, the bounding box regression is a common module across most of the detectors that are used today. We see from the results that attacks that target this module can achieve great transferability across networks, models, training data and even tasks. Thus leaving the localization module an important attack surface that needs to be investigated.

6.2 Implication to Academia and Industry

The usage of object detectors in machine learning tasks is increasing. And there are many pre-trained and open-source detectors available, which makes it important to gain knowledge about the security risks of the different detectors before choosing a detector for the task. Here, this survey provides an examination of how different detectors behave under different attack settings, thus providing necessary information for the industry when selecting detectors.

The state of today's attacks targeting the bounding box regression (RQ1) shows that they are able to achieve great attack performance while jointly targeting the regressor and classifier. Furthermore, their evaluations (RQ2) indicate that targeting this common feature of region proposals makes the attacks transfer well to black-box targets. Lastly, none of the discussed defenses (RQ3) have proved to mitigate attacks that target the RPN.

Of the four categories introduced in this survey, the *Background Evasion* category proves to be efficient both in white- and black-box settings, while not having any clear defense strategies to mitigate their effect. This is further strengthened by the results of Table 5.27, where the attacks which miss theoretical mitigation all share the commonality of targeting the RPN. This commonality is shared by the *Background Evasion* attacks, hence there is an identified research gap in how to mitigate attacks that targets the RPN.

The proposed taxonomy is provided to further mitigate this research gap and aims to be the first step toward robust bounding box regression for object detectors. Overall, the taxonomy describes how an AE can be generated to attack different object detectors, based on various information. Next, it describes how the attack can be deployed to realize the attack. Lastly, to uncover the strength in transferring the attacks, the taxonomy describes the potential of the attacks being transferred to attack other targets than used in the generation. This can be utilized in future research to categorize attacks and help to map out how to defend against attacks with a particular focus on the bounding box regression.

In this survey, some defense techniques that show potential in mitigating the discussed attacks have been proposed. This is still just theoretical and has not been verified through experiments. Experiments of implementing these defenses against the discussed attacks are left for future work.

6.3 Threat to Validity

The work done in this thesis contains some threats to validity that will be disclosed, and which can be improved through future work. Firstly, all the evaluations used in this survey are obtained from their respective studies. This means that the different attacks and defenses have been evaluated on different detectors with different internal configurations. The difference in the detectors may have resulted in unfairness when comparing some of the attacks. To help mitigate this error, the attacks were compared by both the mAP reduction and the relative drop of mAP between their benign results and the results on the AEs.

Furthermore, even though the results of the attacks are discussed to show transferability, their performance in black-box settings when transferred is not evaluated against other state-of-the-art black-box attacks. Ideally, this would be included in this thesis to justify whether or not targeting the region proposals can outperform the state-of-the-art black-box attacks.

Chapter 7

Conclusion and Future Work

This chapter will summarize the contribution and findings of the discussed research questions, and conclude the thesis. Lastly, any work I intend for the future to further improve the contributions of this survey are presented.

7.1 Conclusion

In this thesis, the results of a literature review have been presented and used to discuss the state of today's attacks and defenses targeting the bounding box regression in object detection according to the research questions. The search and selection process of the literature review resulted in a total of 21 relevant studies, where 16 studies focused on attacks and the last five focused on defense.

The main findings of RQ1 show that these attacks are capable of disrupting the object detectors' capability to provide true region proposals. Thereby, the bounding box regressor serves to be an important attack surface for adversarial input attacks.

The evaluations collected from each of the selected studies for RQ2 proved that some of these attacks could transfer across different internal settings in the object detectors, without loss of attack performance. As the regressor shares commonalities between different object detector architectures, it can be exploited to improve the generalization of the attacks.

The findings of RQ3 show that there were only a few defensive studies related to the bounding box regression, and only three of the 16 attacks have been mitigated. This indicates that there is a significant gap between the research of robust region proposals and exploiting them in attacks.

After the literature review, the important commonalities of the selected attacks were identified as the attributes *Attack Knowledge, Target(s), Transferability, Loss function, Environment, Attack Style* and *Optimization Scheme*. Then, the attacks were clustered into four abstract categories depending on their attack output. These categories were *Background Evasion Attacks, Offset-Push Attacks, Total Loss Attacks* and *Region of Interest Attacks*.

To define a taxonomy, three phases of an AE attack were elicited: *Generation*, *Deployment* and *Transfer*. Each of these phases is connected to separate parts of the attributes while providing important insights into the different ways AEs are generated, how they are deployed to attack their target, and lastly how they can generalize to transfer across different object detectors.

7.2 Future Work

For future work, I would like to implement all the discussed attacks, to conduct experiments in both white- and black-box settings while evaluating all the attacks. This would be done by using the same detectors for all targets and attacks, providing a fair baseline for evaluation. The evaluation would be done by a common evaluation metric which would let me compare all the attacks more fairly, and thus improve the shortcomings of different detectors used in the experiments, as discussed in section 6.3.

And to further validate the generalization of the discussed attacks, I would want to implement the outperforming transfer attacks in this survey and benchmark them against other state-of-the-art black-box attacks. This would preferably be done as an ablation study of different transfer settings, such that the attacks can be evaluated across all combinations of cross-data, -model, -network and -task transfer to further research how targeting the region proposal as a common vulnerability among object detectors affects transferability.

Bibliography

[1] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow and R. Fergus, *Intriguing properties of neural networks*, 2014. arXiv: 1312.6199 [cs.CV].

[2] K. He, X. Zhang, S. Ren and J. Sun, 'Deep residual learning for image recognition,' in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[3] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2014. DOI: 10.48550/ARXIV.1409.1556. [Online]. Available: https://arxiv.org/abs/1409.1556.

[4] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto and H. Adam, *Mobilenets: Efficient convolutional neural networks for mobile vision applications*, 2017. DOI: 10.48550/ARXIV.1704.04861. [Online]. Available: https://arxiv.org/abs/1704.04861.

[5] J. Redmon and A. Farhadi, *Yolov3: An incremental improvement*, 2018. arXiv: 1804.02767 [cs.CV].

[6] S. Ren, K. He, R. Girshick and J. Sun, 'Faster r-cnn: Towards real-time object detection with region proposal networks,' in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama and R. Garnett, Eds., vol. 28, Curran Associates, Inc., 2015. [Online]. Available: https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf.

[7] R. Girshick, J. Donahue, T. Darrell and J. Malik, 'Rich feature hierarchies for accurate object detection and semantic segmentation,' in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[8] R. Girshick, 'Fast r-cnn,' in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[9] N. Papernot, P. McDaniel and I. Goodfellow, *Transferability in machine learning: From phenomena to black-box attacks using adversarial samples*, 2016. arXiv: 1605.07277 [cs.CR].

[10] C. Xie, J. Wang, Z. Zhang, Z. Ren and A. Yuille, 'Mitigating adversarial effects through randomization,' 2017. DOI: `10.48550/ARXIV.1711.01991`. [Online]. Available: `https://arxiv.org/abs/1711.01991`.

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

[12] I. J. Goodfellow, J. Shlens and C. Szegedy, *Explaining and harnessing adversarial examples*, 2014. arXiv: `1412.6572 [stat.ML]`.

[13] A. K. I. J. Goodfellow and S. Bengio, 'Adversarial examples the physical world,' in *International Conference on Learning Representations (ICRL)*, vol. 2, 2017, p. 4.

[14] Z. Kong, J. Xue, Y. Wang, L. Huang, Z. Niu and F. Li, 'A survey on adversarial attack in the age of artificial intelligence,' *Wireless Communications and Mobile Computing*, vol. 2021, 2021.

[15] N. Pitropakis, E. Panaousis, T. Giannetsos, E. Anastasiadis and G. Loukas, 'A taxonomy and survey of attacks against machine learning,' *Computer Science Review*, vol. 34, p. 100 199, 2019, ISSN: 1574-0137. DOI: `https://doi.org/10.1016/j.cosrev.2019.100199`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S1574013718303289`.

[16] A. Serban, E. Poll and J. Visser, 'Adversarial examples on object recognition: A comprehensive survey,' *ACM Computing Surveys (CSUR)*, vol. 53, no. 3, pp. 1–38, 2020.

[17] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu and V. C. M. Leung, 'A survey on security threats and defensive techniques of machine learning: A data driven view,' *IEEE Access*, vol. 6, pp. 12 103–12 117, 2018. DOI: `10.1109/ACCESS.2018.2805680`.

[18] J. S. Molléri, K. Petersen and E. Mendes, 'Survey guidelines in software engineering: An annotated review,' in *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, ser. ESEM '16, Ciudad Real, Spain: Association for Computing Machinery, 2016, ISBN: 9781450344272. DOI: `10.1145/2961111.2962619`. [Online]. Available: `https://doi.org/10.1145/2961111.2962619`.

[19] Y. Wang, K. Wang, Z. Zhu and F.-Y. Wang, 'Adversarial attacks on faster r-cnn object detector,' *Neurocomputing*, vol. 382, pp. 87–95, 2020, ISSN: 0925-2312. DOI: `https://doi.org/10.1016/j.neucom.2019.11.051`. [Online]. Available: `https://www.sciencedirect.com/science/article/pii/S0925231219316534`.

[20] Y. Huang, A. W.-K. Kong and K.-Y. Lam, 'Attacking object detectors without changing the target object,' in *PRICAI 2019: Trends in Artificial Intelligence*, A. C. Nayak and A. Sharma, Eds., Cham: Springer International Publishing, 2019, pp. 3–15.

[21] Y. Li, M.-C. Chang, P. Sun, H. Qi, J. Dong and S. Lyu, 'Transrpn: Towards the transferable adversarial perturbations using region proposal networks and beyond,' *Computer Vision and Image Understanding*, vol. 213, p. 103 302, 2021, ISSN: 1077-3142. DOI: https://doi.org/10.1016/j.cviu.2021.103302. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314221001466.

[22] Y. Li, D. Tian, M.-C. Chang, X. Bian and S. Lyu, 'Robust adversarial perturbation on deep proposal-based models,' 2018. DOI: 10.48550/ARXIV.1809.05962. [Online]. Available: https://arxiv.org/abs/1809.05962.

[23] Z. Hu and Z. Zhong, 'Towards practical robustness improvement for object detection in safety-critical scenarios,' in *Deployable Machine Learning for Security Defense*, G. Wang, A. Ciptadi and A. Ahmadzadeh, Eds., Cham: Springer International Publishing, 2020, pp. 66–83, ISBN: 978-3-030-59621-7.

[24] J. Bao, J. Chen, H. Ma, H. Ma, C. Yu and Y. Huang, 'Improving adversarial robustness of detector via objectness regularization,' in *Pattern Recognition and Computer Vision*, Cham: Springer International Publishing, 2021, pp. 252–262, ISBN: 978-3-030-88013-2.

[25] Z. Liu, W. Peng, J. Zhou, Z. Wu, J. Zhang and Y. Zhang, 'Mi-fgsm on faster r-cnn object detector,' in *2020 The 4th International Conference on Video and Image Processing*, ser. ICVIP 2020, Xi'an, China: Association for Computing Machinery, 2020, pp. 27–32, ISBN: 9781450389075. DOI: 10.1145/3447450.3447455. [Online]. Available: https://doi.org/10.1145/3447450.3447455.

[26] J. Liu, Y. Wang, Y. Yin, Y. Hu, H. Chen and X. Gong, 'Adversarial attacks on faster r-cnn: Design and ablation study,' in *2021 China Automation Congress (CAC)*, 2021, pp. 7395–7400. DOI: 10.1109/CAC53003.2021.9728435.

[27] Z. Zhang, S. Huang, X. Liu, B. Zhang and D. Dong, 'Adversarial attacks on yolact instance segmentation,' *Computers Security*, vol. 116, p. 102 682, 2022, ISSN: 0167-4048. DOI: https://doi.org/10.1016/j.cose.2022.102682. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404822000803.

[28] X. Liu, H. Yang, Z. Liu, L. Song, H. Li and Y. Chen, 'Dpatch: An adversarial patch attack on object detectors,' *arXiv: Computer Vision and Pattern Recognition*, 2019.

[29] Y. Li, X. Bian, M.-C. Chang and S. Lyu, 'Exploring the vulnerability of single shot module in object detectors via imperceptible background patches,' *arXiv preprint arXiv:1809.05966*, 2018.

[30] H. Zhang and J. Wang, 'Towards adversarially robust object detection,' in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019.

[31] L. Huang, C. Gao, Y. Zhou, C. Xie, A. L. Yuille, C. Zou and N. Liu, 'Universal physical camouflage attacks on object detectors,' in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 720–729.

[32] H. Zhang, W. Zhou and H. Li, 'Contextual adversarial attacks for object detection,' in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, 2020, pp. 1–6. DOI: 10.1109/ICME46284.2020.9102805.

[33] P.-y. Chiang, M. Curry, A. Abdelkader, A. Kumar, J. Dickerson and T. Goldstein, 'Detection as regression: Certified object detection with median smoothing,' *Advances in Neural Information Processing Systems*, vol. 33, pp. 1275–1286, 2020.

[34] X. Wu, L. Huang and C. Gao, 'G-uap: Generic universal adversarial perturbation that fools rpn-based detectors,' in *Proceedings of The Eleventh Asian Conference on Machine Learning*, W. S. Lee and T. Suzuki, Eds., ser. Proceedings of Machine Learning Research, vol. 101, PMLR, 17th–19th Nov. 2019, pp. 1204–1217. [Online]. Available: https://proceedings.mlr.press/v101/wu19a.html.

[35] Z. Shi, W. Yang, Z. Xu, Z. Chen, Y. Li, H. Zhu and L. Huang, 'Adversarial attacks on object detectors with limited perturbations,' in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 1375–1379. DOI: 10.1109/ICASSP39728.2021.9414125.

[36] M. Yuan and X. Wei, 'Generating adversarial remote sensing images via pansharpening technique,' in *Proceedings of the 1st International Workshop on Adversarial Learning for Multimedia*. New York, NY, USA: Association for Computing Machinery, 2021, pp. 15–20, ISBN: 9781450386722. [Online]. Available: https://doi.org/10.1145/3475724.3483602.

[37] K.-H. Chow and L. Liu, 'Robust object detection fusion against deception,' in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery Data Mining*, ser. KDD '21, Virtual Event, Singapore: Association for Computing Machinery, 2021, pp. 2703–2713, ISBN: 9781450383325. DOI: 10.1145/3447548.3467121. [Online]. Available: https://doi.org/10.1145/3447548.3467121.

[38] Y. Jia, Y. Lu, J. Shen, Q. A. Chen, Z. Zhong and T. Wei, 'Fooling detection alone is not enough: First adversarial attack against multiple object tracking,' in *International Conference on Learning Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=rJl31TNYPr.

[39] J. Bao, 'Sparse adversarial attack to object detection,' *ArXiv*, vol. abs/2012.13692, 2020. [Online]. Available: https://arxiv.org/abs/2012.13692.

[40] A. Bochkovskiy, C.-Y. Wang and H.-Y. M. Liao, *Yolov4: Optimal speed and accuracy of object detection*, 2020. arXiv: 2004.10934 [cs.CV].

[41] J. Dai, Y. Li, K. He and J. Sun, 'R-fcn: Object detection via region-based fully convolutional networks,' *Advances in neural information processing systems*, vol. 29, 2016.

[42] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu and A. C. Berg, 'Ssd: Single shot multibox detector,' in *Computer Vision – ECCV 2016*, B. Leibe, J. Matas, N. Sebe and M. Welling, Eds., Cham: Springer International Publishing, 2016, pp. 21–37.

[43] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan and S. Belongie, 'Feature pyramid networks for object detection,' in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017.

[44] S. Zhou, *Dataset of gf-1 satellite images (2017-2018)*, Feb. 2019. DOI: `10.11888/Geogra.tpdc.271227`. [Online]. Available: `http://dx.doi.org10.11888/Geogra.tpdc.271227`.

[45] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren and A. L. Yuille, 'Improving transferability of adversarial examples with input diversity,' in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2730–2739.

[46] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu and J. Li, 'Boosting adversarial attacks with momentum,' in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9185–9193.

[47] D. P. Kingma and J. Ba, 'Adam: A method for stochastic optimization,' *CoRR*, vol. abs/1412.6980, 2015.

[48] X. Wei, S. Liang, N. Chen and X. Cao, 'Transferable adversarial attacks for image and video object detection,' 2018. DOI: `10.48550/ARXIV.1811.12641`. [Online]. Available: `https://arxiv.org/abs/1811.12641`.

[49] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan and T. Darrell, 'Bdd100k: A diverse driving dataset for heterogeneous multitask learning,' in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2636–2645.

[50] Y. Li, H. Qi, J. Dai, X. Ji and Y. Wei, 'Fully convolutional instance-aware semantic segmentation,' in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2359–2367.

[51] K. He, G. Gkioxari, P. Dollár and R. Girshick, 'Mask r-cnn,' in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[52] D. Bolya, C. Zhou, F. Xiao and Y. J. Lee, 'Yolact: Real-time instance segmentation,' in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 9156–9165. DOI: `10.1109/ICCV.2019.00925`.

[53] S.-T. Chen, C. Cornelius, J. Martin and D. H. P. Chau, 'Shapeshifter: Robust physical adversarial attack on faster r-cnn object detector,' in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, 2018, pp. 52–68.

[54] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno and D. Song, 'Robust physical-world attacks on deep learning visual classification,' in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1625–1634.

[55] S. Thys, W. Van Ranst and T. Goedemé, 'Fooling automated surveillance cameras: Adversarial patches to attack person detection,' in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 0–0.

[56] T. B. Brown, D. Mané, A. Roy, M. Abadi and J. Gilmer, 'Adversarial patch,' *arXiv preprint arXiv:1712.09665*, 2017.

[57] K. R. Mopuri, A. Ganeshan and R. V. Babu, 'Generalizable data-free objective for crafting universal adversarial perturbations,' *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 10, pp. 2452–2465, 2018.

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

[59] S. Liu, D. Huang *et al.*, 'Receptive field block net for accurate and fast object detection,' in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 385–400.

[60] Z. Li and F. Zhou, *Fssd: Feature fusion single shot multibox detector*, 2017. DOI: `10.48550/ARXIV.1712.00960`. [Online]. Available: `https://arxiv.org/abs/1712.00960`.

[61] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, 'You only look once: Unified, real-time object detection,' in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[62] K.-H. Chow, L. Liu, M. E. Gursoy, S. Truex, W. Wei and Y. Wu, 'Tog: Targeted adversarial objectness gradient attacks on real-time object detection systems,' *arXiv preprint arXiv:2004.04320*, 2020.

[63] S. Thys, W. Van Ranst and T. Goedemé, *Fooling automated surveillance cameras: Adversarial patches to attack person detection*, 2019. DOI: `10.48550/ARXIV.1904.08653`. [Online]. Available: `https://arxiv.org/abs/1904.08653`.

[64] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez and J. Garcia-Rodriguez, *A review on deep learning techniques applied to semantic segmentation*, 2017. arXiv: `1704.06857 [cs.CV]`.

[65] Z. Qin, F. Yu, C. Liu and X. Chen, *How convolutional neural network see the world - a survey of convolutional neural network visualization methods*, 2018. arXiv: `1804.11191 [cs.CV]`.

[66] D. Wang, C. Li, S. Wen, Q.-L. Han, S. Nepal, X. Zhang and Y. Xiang, *Daedalus: Breaking non-maximum suppression in object detection via adversarial examples*, 2020. arXiv: 1902.02067 [cs.CV].

[67] J. Redmon and A. Farhadi, *Yolo9000: Better, faster, stronger*, 2016. arXiv: 1612.08242 [cs.CV].

[68] L. Engstrom, B. Tran, D. Tsipras, L. Schmidt and A. Madry, *Exploring the landscape of spatial robustness*, 2019. arXiv: 1712.02779 [cs.LG].

[69] N. Carlini and D. Wagner, *Towards evaluating the robustness of neural networks*, 2017. arXiv: 1608.04644 [cs.CR].

[70] T. B. Brown, D. Mané, A. Roy, M. Abadi and J. Gilmer, *Adversarial patch*, 2018. arXiv: 1712.09665 [cs.CV].

[71] M. Lee and Z. Kolter, *On physical adversarial patches for object detection*, 2019. arXiv: 1906.11897 [cs.CV].

[72] Y. Wang, H. Lv, X. Kuang, G. Zhao, Y.-a. Tan, Q. Zhang and J. Hu, 'Towards a physical-world adversarial patch for blinding object detection models,' *Information Sciences*, vol. 556, pp. 459–471, 2021, ISSN: 0020-0255. DOI: https://doi.org/10.1016/j.ins.2020.08.087. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0020025520308586.

[73] Q. Zhang, Y. Zhao, Y. Wang, T. Baker, J. Zhang and H. Jingjing, 'Towards cross-task universal perturbation against black-box object detectors in autonomous driving,' *Computer Networks*, vol. 180, p. 107 388, Jul. 2020. DOI: 10.1016/j.comnet.2020.107388.

[74] M. Lu, Q. Li, L. Chen and H. Li, 'Scale-adaptive adversarial patch attack for remote sensing image aircraft detection,' *Remote Sensing*, vol. 13, no. 20, 2021, ISSN: 2072-4292. DOI: 10.3390/rs13204078. [Online]. Available: https://www.mdpi.com/2072-4292/13/20/4078.

Appendix A

Background

A.1 Convolutional Neural Networks

Convolutional Neural Networks (CNNs), has become the model architecture of choice for any task including analyzing images [64]. Furthermore, CNN is used to realize most modern object detectors and has become the state of the art in terms of classification and segmentation tasks.

CNNs utilizes features in images to extract feature maps, which are further analyzed to make predictions [65].

A.2 Image Classification

Image classification has become the fundamental problem for computer vision tasks. The task of image classification is given an image, the classification models are to provide a list of the predicted classes of the object dominating the image and their respective confidences.

A.3 Bounding boxes

Taking computer vision a step further from classification alone, certain tasks depends on detecting and localizing multiple objects within an image or a single frame from a video stream.

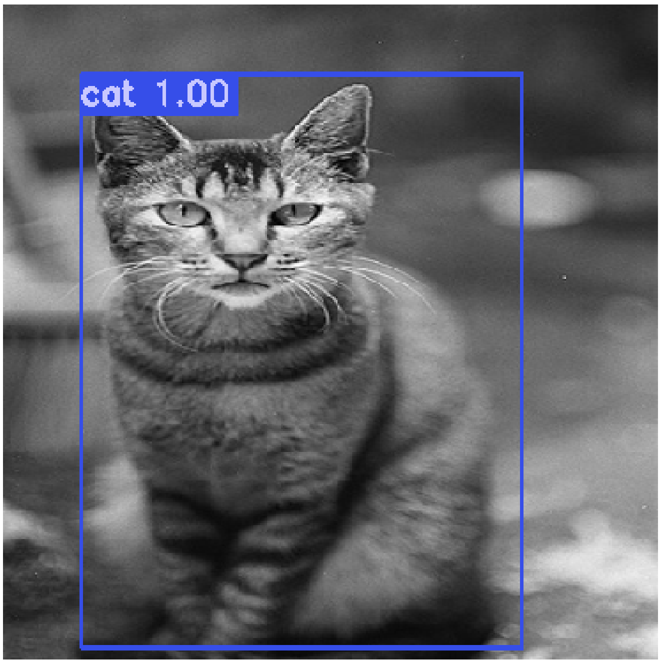


Figure A.1: A cat with a predicted bounding box detected by a Faster RCNN object detector.

A.4 Object Detectors

Object detection combines the concept of bounding boxes and classification, where we introduce localization of the object to be classified in the image or video. The objects localization can be described through the bounding boxes proposed by the object detectors.

Throughout the years, several object detectors have been developed to perform computer vision tasks. In particular, You Only Look Once (YOLO), Single Shot MultiBox Detector (SSD) [42] and Faster R-CNN are leading object detection models in computer vision tasks [66].

Object detector models can be further divided into one- and two-stage models. Where Faster R-CNN (faster region-based CNN)[6] is an example of a model with two-step architecture. Faster R-CNN extracts possible object regions and performs classifications of the object within the proposed region in two separate steps. While YOLO, a one-stage model, extracts the proposed bounding boxes and performs classification at the same time.

The one-stage models trades off the accuracy of smaller objects for higher speed, giving them the property to achieve real-time object detection. Two-stage models aim for higher mean Average Precision (mAP) scores while reducing the detection speed, making it generally less viable for tasks requiring a high update rate, such as autonomous driving. Faster R-CNN is several times faster than its predecessor, Fast R-CNN, making it near real-time detector, as described in [6].

A.4.1 YOLO

"You Only Look Once" is the core principle of YOLO, making it an one-stage object detector. YOLO both propose regions for objects and classify the regions in the same step, achieving real-time detection speed with a cost of lower accuracy.

There have been several iterations of the object detectors, where YOLOv1 [61] directly returns the bounding boxes and classification at the output layer. YOLOv2 [67] removes the fully connected layer by adding batch-normalization. This iteration improved the accuracy of the object detector. Detecting small objects was still a problem with YOLO, this was improved in YOLOv3 [5] where multi-scale predictions were used. Lastly, several optimizations were proposed for YOLOv4 [40] to achieve a state-of-the-art mAP and speed.

A.4.2 Faster R-CNN

Faster R-CNN is a two-stage object detector. The first stage of the detector is to propose regions where objects exist. The second stage classifies said regions. Two realize the first stage of Faster R-CNN, a RPN is used to output a feature map which again is used as input to a fully connected neural network for object classification.

A.5 Adversarial Examples

CNNs and object detector has been known to be prone to imperceptible non-random perturbations to input images, leading to misclassification [1]. These perturbed examples were termed adversarial examples.

Adversarial examples and why they are effective were further explored in [12]. Goodfellow *et al.* [12] argues that the reasoning for why adversarial examples are effective is due to the linearity in neural networks. This was a new argument from the focus on nonlinearity and overfitting which was the previous hypothesis. This is further backed up by the generalization of the attacks which makes them transferable across datasets and models.

A.6 Performance Metrics

When building an object detector, there is a need to measure how accurate the predictions of the classification model are. Some basic measures which are widely used are precision, recall and F1. Later, the more advanced metrics AP and mAP have become a more all-around metric heavily used in the field.

Common for these three measures is that they utilize true/false negatives and positives.

Precision is in place to measure whether the model avoids mistakes while classifying a specific class.

$$Precision = \frac{TP}{TP + FP}$$

$$\text{where } TP = \text{True Positive and } FP = \text{False Positive}$$

Recall measures how well the model finds all the positives. If the model avoids mistakes of classifying a given class as other classes, the model has a high recall.

$$Recall = \frac{TP}{TP + FN}$$

$$\text{where } TP = \text{True Positive and } FN = \text{False Negative}$$

F1 takes in account both precision and recall.

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

In some scenarios, a model with high recall at the cost of precision is beneficial. More precisely, when the cost of false negatives are extremely higher than false positives. E.g. for autonomous vehicles, false negative on a pedestrian crossing the road can't be allowed to happen. Thus, focusing on high recall would be more beneficial.

For a better comparison of prediction models, a precision-recall curve can be analyzed to evaluate the trade-off between precision and recall. This introduces another performance metric, Average Precision (AP). Which can be further used to calculate mAP. AP and mAP has become the most popular metrics for evaluating prediction models, and have proved to be great at evaluating models against each other on the same datasets. The metrics gained heavy popularity through the usage of the metrics in the challenges COCO [1] and PASCAL VOC[2].

Firstly, let's look at Intersection over Union (IoU), see Figure A.2 for visualization. IoU is a metric to evaluate the intersection of the area of the ground truth and the area of the predicted mask. Where IoU will be a number between 0 and 1, and closer to 1 means a more accurate prediction.

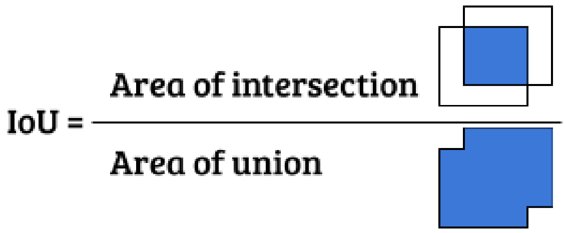


Figure A.2: Intersection over Union illustration

[1] https://cocodataset.org/

[2] http://host.robots.ox.ac.uk/pascal/VOC/

By defining a threshold, here α, for the intersection to determine whether to mark the predicted bounding box to be correct or not, we can calculate the IoU.

$$AP@\alpha = \int_0^1 p(r)dr, \tag{A.1}$$

where α is the threshold, and $p(r)$ is the Precision-Recall curve

For each class, AP can be calculated. For COCO, this means at one specific α, one can calculate 80 different APs, one for each of the 80 classes. To evaluate a model given all classes in the dataset, all AP values can be averaged to get the resulting mAP.

$$mAP@\alpha = \frac{1}{n}\sum_{i=1}^{n} AP_i, \tag{A.2}$$

sum over all n classes, and α is the IoU-threshold

A.7 Non-max suppression

Non-max suppression (NMS) is an algorithm to solve the problem of multiple overlapping proposals of bounding boxes of the same classes. In short, NMS iterates over all classes and compares IoUs to discard the proposals of lowest confidence when the IoU is above a predefined threshold. This makes the less confident proposals regarded as false positives.

A.8 Region of Interest Pooling

When processing an image, it's first preprocessed in a trained CNN to generate a feature map. This feature map will then be inputted into a Region Proposal Network (RPN) to get proposals of Region of Interest (RoI). These RoIs describes where the RPN predicts there exists object, and are then inputted into the RoI pooling

RoI pooling was needed for object detectors based on fully connected CNNs, since they expect a fixed-sized feature map. RoI pooling handles the output of the RPN which will be of different shapes.

A.9 Attack types

The different attacks can be divided into one of three attack types: Black-box, white-box and gray-box attack. The difference is what kind of information about the target that is available to the attacker. If the attack has full access to the target model, including weights, dataset, input and output, the attack is classified as a white-box attack.

Furthermore, if only the input and output, optionally with the scores of the output, the attack can be classified as black-box attack. Effective black-box attacks can have severe consequences in safety-critical tasks like autonomous vehicles, as the internal structures of the object detection models will less likely be publicly available. If an attacker can successfully attack given no information about the target, keeping the configuration and output of the object detector private no longer prove as any security measure.

A.10 Transferability

While the most effective attacks often came from white-box attacks, attacks often transfer to other models which the attack was not initially attended to attack [1, 9, 12]. This proves some serious concerns, as the attacker no longer may need to have any information about their target. Transferability can be thought of as the measure of how well an attack against a target can be performed on a separate target with no further configurations.

The transferability can further be specified into cross-model and cross-dataset, where the latter focuses on transferability of an attack meant for model A trained on dataset X can be used for a model similar to A, only trained on dataset Y.

A.11 Attack Constraints

As discussed, the perturbations are meant to be imperceptible for the human eye. To achieve this, attacks can implement certain constraints on how to perturbate the images. These constraints can seek to minimize changes in pixels, or how the images are to be rotated or transformed to generate an adversarial example [68].

A.11.1 Distance Metrics

The most common constraint of perturbations are the ℓ_p-norms ℓ_0, ℓ_2 and ℓ_∞ [69]. These are distance metrics that can be used to analyze how perceptible the perturbations are. The ℓ_p-norms are a mathematical definition, and not a perfect measurement for how perceptible the perturbation will be for the human eye.

ℓ_p-norms are defined by

$$||x - x'||_p \tag{A.3}$$

where the p-norm is defined as

$$||x||_p = (\sum_{i=\mathcal{I}} |x_i|)^{\frac{1}{p}} \tag{A.4}$$

These norms specify how close the adversarial examples are to the benign examples in terms of pixels changed.

- ℓ_0 seeks to minimize the number of pixels perturbated, with no limits on how much they change.
- ℓ_2 minimizes the Euclidean distance between the adversarial and the benign example, as Equation A.3 becomes the Euclidean distance with $p = 2$. Meaning the ℓ_2 can stay minimized if there are many pixels with small changes.
- Lastly, ℓ_∞ only measures the maximum change of *any* pixel, with no regard to how many pixels changed up to this maximum.

A.11.2 Patches

Patch-based adversarial examples utilized small patches that are heavily perturbated, but limited to a small area of the image. These types of attacks quickly proved to have great attack effects on image-level classifiers, and they also contained the property to be able to attack physical real-world objects. To achieve a physical attack, the patch could be printed out and attached to the physical world, resulting in the detector failing to detect objects properly [70–72].

A.12 DPatch

Liu *et al.* [28] introduces DPatch, an adversarial patch-attack with a high attack effect with a limited-sized, location-independent patch. The attack focuses on two state-of-the-art object detectors, YOLOv2 and Faster R-CNN. DPatch has the possibility to perform both targeted and untargeted attacks, leading to devastating attack results. A DPatch trained on YOLO was proved to be effective on a Faster R-CNN based detection model, and vice-versa, meaning the attack contains great transferability potential. Due to the location-independent property, [28] claims the attack is practical for a real-world implementation of the attack. This was further explored in [71, 73].

The targeted attack aims to make the patch the only RoI, with a given label, thus breaking the object detector and making it fail to detect the other objects in the frame. Thus the targeted attack is with the purpose of evading objects in the frame. The untargeted attack on the other hand, seek to break the RPN in such a way that the object detector proposes objects that are not in the frame, simultaneously removing detection of actual objects. Thereby having the untargeted attack both be with the goal of evasion and fabrication.

Appendix B

Issues of DPatch

Throughout a pre-study[1], examples has been provided which doubts the effect of the DPatch attack. The attack seems to achieve manufacturing a fake object with the target label, but suppression of real objects was not found.

An interesting remark is that the original paper [28] was published by Advancement of Artificial Intelligence (AAAI)[2] 2019, which brings a certain legitimacy to the paper and the attack. The findings in these experiments, which confirms the problems described in [71], shows that the publishing can be misleading and that further validation should be in place to verify the published papers.

As discussed in [74] and [71], the colors of the patch is not clipped to match the targeted images pixel values. Thus, there is a mismatch between the colors of the patch and the target image, making it less suitable for physical attacks and less robust in digital attacks.

This leaves future work to be done to investigate the attack further, with a goal of pinpointing the problem of the attack and further improve it such that it can perform more reliable. And we can for now conclude that the DPatch attack is not robust.

[1] https://github.com/mariusblarsen/dpatch-experiments/blob/a03848019642c63b312c175994225fb20e84a6ac/A%20study%20of%20the%20robustness%20of%20DPatch%20attack.pdf

[2] https://www.aaai.org/

NTNU

Kunnskap for en bedre verden