National Technical University of Athens

School of Electrical and Computer Engineering

Division of Computer Science

Vulnerabilities and robustness of Convolutional Neural Networks against Adversarial Attacks in the spatial and spectral domain

Diploma Thesis

of

FOTINI DELIGIANNAKI

Supervisor: Andreas-Georgios Stafylopatis, NTUA Professor
Georgios Siolas, NTUA Senior Researcher

Artificial Intelligence and Learning Systems Laboratory

Athens, February 2022

National Technical University of Athens
School of Electrical and Computer Engineering
Division of Computer Science
Artificial Intelligence and Learning Systems Laboratory

Vulnerabilities and robustness of Convolutional Neural Networks against Adversarial Attacks in the spatial and spectral domain

Diploma Thesis

of

FOTINI DELIGIANNAKI

Supervisor: Andreas-Georgios Stafylopatis, NTUA Professor
Georgios Siolas, NTUA Senior Researcher

Approved by the examination committee on 24 February 2022.

(Signature)	*(Signature)*	*(Signature)*
.....	.....	.....
Andreas-Georgios Stafylopatis	Georgios Stamou	Stefanos Kollias
Professor	Professor	Professor
NTUA	NTUA	NTUA

Athens, February 2022

(Signature)

.....

Fotini N. Deligiannaki
Electrical and Computer Engineering Graduate, NTUA

Abstract

The constant rise in the capabilities of Artificial Intelligence has led to its application in numerous domains even when safety is a critical component. In the area of computer vision, Convolutional Neural Networks (CNNs) achieve impressive results in image classification, segmentation and object detection. It has been proven though that CNNs are easily manipulated and fooled by very small and carefully crafted corruptions, imperceptible to the human eye. These corruptions known as adversarial attacks have raised the question of the robustness of modern CNNs to images deviating from the training data distribution and pose an important threat to their reliability. A variety of attack as well as defence and detection methods have been proposed but to this date models are still vulnerable.

The purpose of this thesis is to examine the success rate of common adversarial attack algorithms as well as the defence method of adversarial training in image classification tasks. Specifically, we start by using common CNN architectures trained on the CIFAR-10 and 350 Bird Species datasets as victim models. We implement two attacks, namely the white-box C&W and PGD methods and manage to fool our models into misclassifying perturbed images with a success rate of up to 100%. In order to then investigate ways to defend our models we use adversarial training with the TRADES algorithm and significantly drop attack success rates, but also show the existing trade-off between accuracy and robustness. Lastly, since current detection methods propose a strong distinction between the spectral representation of adversarial examples and benign images, we explore the characteristics of adversarial attacks as well as training methods in the Fourier domain. Through this analysis we observe that perturbations are influenced by a number of factors related to the dataset, training algorithm and model architecture and aspire to bring forward the Fourier domain properties that differentiate robust from non-robust models and their vulnerabilities.

Keywords— robustness, adversarial machine learning, convolutional neural networks, image classification, Fourier transform

Περίληψη

Η συνεχής εξέλιξη των δυνατοτήτων της Τεχνητής Νοημοσύνης έχει οδηγήσει στην ευρεία εφαρμογή της ακόμη και σε πεδία όπου η ανάγκη ασφαλούς λειτουργίας της είναι κρίσιμη. Στον τομέα της όρασης υπολογιστών τα Συνελικτικά Νευρωνικά Δίκτυα (ΣΝΔ) επιτυγχάνουν εντυπωσιακά αποτελέσματα, ωστόσο έχει αποδειχθεί ότι αυτά είναι επιρρεπή σε μικρές και στοχευμένες αλλοιώσεις των εικόνων, ανεπαίσθητες στο ανθρώπινο μάτι. Αυτές οι αλλοιώσεις, κοινώς γνωστές ως ανταγωνιστικές επιθέσεις (adversarial attacks) έχουν οδηγήσει σε ερωτήματα σχετικά με την ανθεκτικότητα των ΣΝΔ απέναντι σε εικόνες αποκλίνουσες της κατανομής των δεδομένων εκπαίδευσης και αποτελούν σημαντική απειλή για την αξιοπιστία τους.

Σκοπός της παρούσας διατριβής είναι να εξεταστεί η επιτυχία πολλαπλών αλγορίθμων επίθεσης καθώς και η άμυνα με την μέθοδο της ανταγωνιστικής εκπαίδευσης σε προβλήματα ταξινόμησης εικόνων. Συγκεκριμένα, ξεκινάμε χρησιμοποιώντας αρχιτεκτονικές ΣΝΔ εκπαιδευμένες στα σύνολα δεδομένων CIFAR-10 και 350 Bird Species ως μοντέλα-θύματα. Υλοποιούμε τις επιθέσεις λευκού-κουτιού (white-box) C&W και PGD και καταφέρνουμε να οδηγήσουμε τα μοντέλα σε λάθος ταξινόμηση των αλλοιωμένων εικόνων με ποσοστό επιτυχίας έως και 100%. Προκειμένου να διερευνήσουμε στη συνέχεια τρόπους υπεράσπισης των ΣΝΔ, χρησιμοποιούμε ανταγωνιστική εκπαίδευση (adversarial training) με τον αλγόριθμο TRADES μειώνοντας σημαντικά τα ποσοστά επιτυχίας των επιθέσεων, αλλά δείχνουμε επίσης το "trade-off" μεταξύ ακρίβειας και ανθεκτικότητας. Τέλος, δεδομένου ότι πολλές μέθοδοι ανίχνευσης τονίζουν ότι υπάρχει ισχυρή διάκριση μεταξύ της φασματικής αναπαράστασης των ανταγωνιστικών παραδειγμάτων (adversarial examples) και των καλοηθών εικόνων, διερευνούμε τα χαρακτηριστικά των εχθρικών επιθέσεων καθώς και τις μεθόδους εκπαίδευσης στο πεδίο Fourier. Μέσω αυτής της ανάλυσης παρατηρούμε ότι οι αλλοιώσεις επηρεάζονται από διάφορους παράγοντες όπως το σύνολο δεδομένων, τον αλγόριθμο εκπαίδευσης και την αρχιτεκτονική του μοντέλου και φιλοδοξούμε να κατανοήσουμε τις ιδιότητες του πεδίου Fourier που διαφοροποιούν τα ανθεκτικά από τα μη ανθεκτικά μοντέλα και τα τρωτά τους σημεία.

Λέξεις-κλειδιά — ισχυρά νευρωνικά δίκτυα, ανταγωνιστική μηχανική μάθηση, συνελικτικά νευρωνικά δίκτυα, ταξινόμηση εικόνων, μετασχηματισμός Fourier

Ευχαριστίες

Η διπλωματική μου εργασία είναι το αποτέλεσμα πολλού κόπου, αναζήτησης, δημιουργικότητας και πειραματισμού, και όλα αυτά τα οφείλω πρώτα απ΄όλα στον επιβλέποντα μου κ. Ανδρέα-Γεώργιο Σταφυλοπάτη. Ήταν μεγάλη μου χαρά που μου έδωσε την ευκαιρία να την εκπονήσω στο εργαστήριο Τεχνητής Νοημοσύνης και Συστημάτων Μάθησης και εκτιμώ ιδιαίτερα τις γνώσεις που πήρα μέσα από τα μαθήματα που διδάσκει. Επίσης, θέλω να ευχαριστήσω θερμά τον κ. Γιώργο Σιόλα ο οποίος από την αρχή μέχρι το τέλος της διπλωματικής με στήριζε με την προθυμία του, την εμπιστοσύνη που μου έδειχνε και τις ουσιαστικές ερευνητικές συμβουλές του. Τέλος, το αποτέλεσμα αυτών των μηνών δεν θα μπορούσε να είναι τόσο πλούσιο χωρίς τις υποδομές που μου παρείχε το ινστιτούτο Max Planck for Empirical Aesthetics για να κάνω τα πειράματα που έπρεπε και τους ευχαριστώ πολύ για την υποστήριξη, και ιδιαίτερα τους Dr. Nori Jacoby και Dr. Peter Harrison που μου έδωσαν αυτήν την δυνατότητα συνεργασίας.

Τα χρόνια μου στο Εθνικό Μετσόβιο Πολυτεχνείο υπήρξαν ξεχωριστά για πολλούς λόγους, πρωταρχικά για τον πλούτο γνώσης που πήρα από τα μαθήματα και τους καθηγητές μου. Γνώρισα φίλους που έδωσαν και δίνουν κάτι ξεχωριστό στη ζωή μου, και τους ευχαριστώ πολύ για τις αξέχαστες στιγμές που περάσαμε μαζί τα τελευταία χρόνια. Εύχομαι να συνεχίσει το ταξίδι μας παρέα και πάντα να μαθαίνουμε και να εξελλισόμαστε. Τέλος, όσο μακριά έφτασα το κατάφερα με την βοήθεια της οικογένειας μου, των γονιών μου και των αδερφών μου και εύχομαι να παίρνουν τόση δύναμη από εμένα, όση παίρνω και εγώ από αυτούς.

Στα αδέρφια μου, Νεκτάριο, Αναστασία, Μόνικα και Θεόδωρο
Στους γονείς μου, Βαρβάρα και Νικόλαο

Contents

List of Figures

List of Tables

Acronyms

ΒΝΔ Βαθιά Νευρωνικά Δίκτυα. 1

ΣΝΔ Συνελικτικά Νευρωνικά Δίκτυα. 7

ΤΝ Τεχνητή Νοημοσύνη. 1

AA Adversarial Attack. 15, 31

AI Artificial Intelligence. 15

BIM Basic Iterative Method. 28

CNN Convolutional Neural Network. 20, 28, 34, 61

CPU Central Processing Unit. 45

CV Computer Vision. 15

DE Differential Evolution. 24

DFT Discrete Fourier Transform. 42

DL Deep Learning. 15

DNN Deep Neural Network. 30

FFT Fast Fourier Transform. 42

FGSM Fast Gradient Sign Method. 22, 39

GPU Graphics Processing Unit. 45

MIAs Membership Inference Attacks. 15

ML Machine Learning. 19

P-RGF Prior-guided Random Gradient-Free method. 5

PGD Projected Gradient Descend. 28

SGD Stochastic Gradient Descend. 48

TRADES TRadeoff-inspired Adversarial DEfense via Surrogate-loss minimization. 20

Κεφάλαιο 1

Εκτεταμένη περίληψη στα Ελληνικά

1.1 Εισαγωγή

Η επίδραση της Τεχνητής Νοημοσύνης (ΤΝ) στην εξέλιξη της κοινωνίας και της τεχνολογίας είναι πλέον εμφανής σε πολυάριθμους τομείς, μερικοί εκ των οποίων είναι η αυτόνομη πλοήγηση [YWY18], η ανακατασκευή τρισδιάστατων εικόνων [ZL21] και βιολογία [Jum+21]. Αυτή η πολυδιάστατη εφαρμογή της ΤΝ σε τομείς στους οποίους η ασφαλείς και αξιόπιστη λειτουργία της είναι κρίσιμης σημασίας, αξίζει να ερευνήσει κανείς τρόπους με τους οποίους μπορεί να χρησιμοποιηθεί με κακόβουλη πρόθεση.

Μέχρι σήμερα έχουν εμφανιστεί πολλαπλές επιθέσεις που θέτουν ως στόχο συστήματα Μηχανικής Μάθησης (Machine Learning - ML) και Βαθιάς Μάθησης (Deep Learning - DL) και έχουν απρόβλεπτες συνέπειες. Στον τομέα της υπολογιστικής όρασης, όσον αφορά προβλήματα από ταξινόμηση μέχρι και ανίχνευση αντικειμένων, μια σημαντική μέθοδος εξαπάτησης των Βαθιών Νευρωνικών Δικτύων (ΒΝΔ) είναι οι ανταγωνιστικές επιθέσεις, που αποσκοπούν στην αλλοίωση της εισόδου με κατάλληλα σχεδιασμένο τρόπο ώστε αυτά να παράξουν λανθασμένη έξοδο (ή σε άλλες περιπτώσεις την επιθυμητή έξοδο του επιτιθέμενου). Αυτές οι αλλοιώσεις είναι δύσκολο να ανιχνευθούν από το ανθρώπινο μάτι (και τα συστήματα) καθώς είναι ιδιαίτερα μικρής ακτίνας απόσταση μακριά από την αρχική είσοδο. Επιπλέον, απειλές μπορούν να συμβούν κατά την εκπαίδευση των μοντέλων όπως οι Trojan [Yun+20] επιθέσεις που εισάγουν προμελετημένο θόρυβο στα δεδομένα εκπαίδευσης για να δημιουργήσουν ανακριβείς συσχετίσεις εισόδου και εξόδου. Τέλος, μέσω επιθέσεων συμπεράσματος μέλους (Membership Inference Attacks) [Che+20a] μπορεί ένας κακοήθης

χρήστης να συλλέξει πληροφορίες για τα δεδομένα που χρησιμοποιήθηκαν στην εκπαίδευσης του μοντέλου, και ως συνέπεια να συγκεντρώσει πιθανόν ευαίσθητα προσωπικά στοιχεία μέσα σε αυτές. Τα παραπάνω φαινόμενα χρήζουν απαραίτητη και ουσιώδη ανάγκη την έρευνα της ανθεκτικότητα, αξιοπιστίας και ασφάλειας των συστημάτων ΤΝ. Στην παρούσα εργασία μας ενδιαφέρουν οι ανταγωνιστικές επιθέσεις που στοχεύουν τα Συνελικτικά Νευρωνικά Δίκτυα (ΣΝΔ) [Pin+20] σε προβλήματα ταξινόμησης.

Οι παραπάνω απειλές έχουν φέρει στο φως συνθήκες λειτουργίας των ΒΝΔ στις οποίες έχουν ελλιπή στιβαρότητα (robustness) και παράγουν αναξιόπιστα αποτελέσματα. Παρόλα αυτά έχουν προταθεί μια σειρά από τρόπους αντιμετώπισης ή ανίχνευσης τέτοιων κακόβουλων επιθέσεων. Όσον αφορά τις ανταγωνιστικές επιθέσεις, έρευνες στοχεύουν κατά κύριο λόγο στο να δημιουργήσουν πιο στιβαρά μοντέλα με μεθόδους όπως η ανταγωνιστική μάθηση [Mad+19] και πιο πρόσφατα τους ορθογώνιους ταξινομητές [XLY21]. Οι πρώτοι όπως θα δούμε αργότερα αλλάζουν το πρόβλημα που καλείται ο ταξινομητής να βελτιστοποιήσει με το να συνθέτουν και να τον εκπαιδεύουν σε ανταγωνιστικά παραδείγματα, οι δεύτεροι με το να ωθούν κατηγορηματικά τον ταξινομητή στο να προβάλει σε κοντινές αποστάσεις τα σημεία της ίδιας κατηγορίας, αλλά προβάλλοντας μακριά τις κατηγορίες μεταξύ τους. Άλλες μέθοδοι έχουν προσπαθήσει να παράξουν ασαφείς κλίσεις (obfuscated gradients) στα ΒΝΔ ώστε ναν μην μπορεί κανείς να προσεγγίσει εύκολα την κλίση για να επιτεθεί, είτε προσπαθούν να ανιχνεύσουν τις κακόβουλες εισόδους, συχνά όμως αποδεικνύεται πως προσαρμοσμένες μέθοδοι επίθεσης μπορούν να ξεπεράσουν τέτοιες άμυνες [A-ΩΩ18]. Με την σειρά μας, σε αυτήν την εργασία υιοθετούμε την ανταγωνιστική μάθηση για να παρατηρήσουμε πόση δύναμη έχουν οι επιθέσεις ενάντια σε ένα αμυνόμενο ΒΝΔ.

Εμβαθύνοντας κανείς στο ερώτημα της ύπαρξης των ανταγωνιστικών παραδειγμάτων, αξίζει να αναφέρουμε τα χαρακτηριστικά τους στο πεδίο της συχνότητας πάνω στα οποία βασίζονται και πολλαπλές μέθοδοι άμυνας ή ανίχνευσης τους. Συγκεκριμένα φαίνεται να έχουν σημαντικά διαφορετική φασματική κατανομή σε σχέση με καλοήθεις εικόνες [Har+21] [Lor+21] και συνήθως αλλοιώνουν τις μέσες και υψηλές συχνότητες στις οποίες η ανθρώπινη αντίληψη μένει αμετάβλητη. Ένα παράδειγμα των φασμάτων της αρχικής και αλλοιωμένης εικόνας φαίνονται στο Σχήμα 1.2. Η παρατήρηση αυτή παρότι είναι από τη μία αναμενόμενη, ανοίγει συζήτηση σχετικά με το ποιά χαρακτηριστικά εκμεταλεύονται τα ΒΝΔ και γιατί επηρεάζονται από μικρές αλλαγές υψηλών συχνοτήτων. Η ιδέα των στιβαρών (robust) και μη-στιβαρών χαρακτηριστικών (non-robust) όπως εισήγαγαν οι [Ily+19] προσπαθεί να ξεχωρίσει αυτά τα χαρακτηριστικά που κάνουν ένα μοντέλο ευάλωτο σε μικρές αλλαγές στην είσοδο και αυτά που

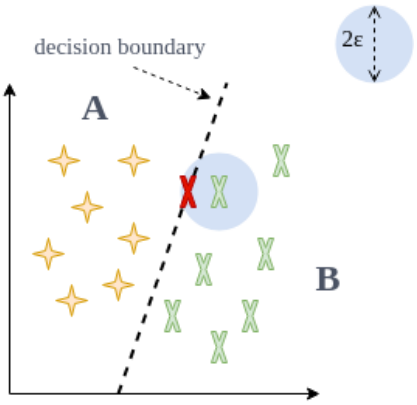
είναι ανθεκτικά και έχουν σημασιολογική και χρήσιμη πληροφορία.

1.2 Ανταγωνιστικές Επιθέσεις σε ταξινομητές

Όπως αναφέραμε προηγουμένως, οι ανταγωνιστικές επιθέσεις είναι ένα ευρέως διαδεδομένο πρόβλημα που από τη μια πλευρά παρατηρείται σε ποικίλες εφαρμογές ML και DL μοντέλων, αλλά από την άλλη πλευρά είναι δύσκολα ανιχνεύσιμο από έναν παρατηρητή. Στο πρόβλημα της ταξινόμησης εικόνων (το οποίο διευρύνεται και σε διαφορετικού είδους εισόδους) η βασική πρόθεση του κακόβουλου χειριστή είναι να ελαχιστοποιήσει την αλλοίωση που χρειάζεται μια εικόνα ώστε να βρεθεί στο όριο απόφασης μεταξύ της πραγματικής κατηγορίας και των γειτονικών αυτής (όπως απεικονίζεται στο Σχήμα 1.1). Η αλλοίωση υπολογίζεται με μια μετρική απόστασης (distance norm) l_p όπως ορίζεται στην Σχέση 4.3 και συνεπώς υπάρχουν διαφορετικού τύπου επιτιθέντες ανάλογα με την μετρική που τους περιορίζει.

Οι επιθέσεις αυτές συναντώνται κυρίως σε μοντέλα ταξινόμησης, όπου ο κακόβουλος χρήστης αλλοιώνει την έισοδο έτσι ώστε το μοντέλο να την τοποθετήσει σε λάθος κατηγορία. Ανάλογα με το είδος της πληροφορίας που έχει ο επιτιθέμενος αλλά και την μέθοδο που χρησιμοποιεί οι επιθέσεις μπορούν να ταξινομηθούν σε λευκού- (white-) και μαύρου- (black-) κουτιού (box), επιθέσεις βάση κλίσης (gradient-based), επιθέσεις βάση ερωτημάτων (query-based), καθολικές και απεριόριστες. Παρακάτω παραθέτουμε τις βασικές αυτές κατηγορίες επιθέσεων και τα χαρακτηριστικά τους στο πρόβλημα της ταξιμόνησης εικόνων με το οποίο καταπιάνεται αυτή η εργασία.

Επιθέσεις λευκού-κουτιού (white-box) Σε αυτήν την περίπτωση ο επιτιθέμενος μπορεί να έχει πρόσβαση σε όλες τις παραμέτρους του ΒΝΔ-στόχου, όπως τα βάρη των διαφορετικών νευρώνων και την αρχιτεκτονική του. Οι πρώτες απόπειρες επίθεσης απορρέουν από τη μέθοδο ταχείας κλίσης (Fast Gradient Sign Method – FGSM) [GSS15] και χρησιμοποιούν την κλίση (gradient) της συνάρτησης απώλειας (loss function) $\nabla_x \mathcal{L}_\theta(x)$ (μοντέλου με παραμέτρους θ) με την οποία γίνεται τυπικά η εκπαίδευση. Συγκεκριμένα, εφόσον κατά την εκπαίδευση λύνεται το πρόβλημα της ελαχιστοποίησης της συνάρτησης απώλειας $\mathcal{L}_\theta(x, y)$ για ζεύγη (x, y) εισόδου-κατηγορίας του συνόλου εκπαίδευσης, στόχος είναι να βρεθεί αλλοίωση δ που να είναι σε αποσταση ϵ από ένα αρχικό σημείο x και να μεγιστοποιεί την συνάρτηση απώλειας (είτε στην περίπτωση που η επίθεση είναι στοχευμένη σε μια διαφορετική κατηγορία $t \neq y$



Σχήμα 1.1: Απεικόνιση της ιδέας πίσω από τις ανταγωνιστικές επιθέσεις - στόχος είναι η αναζήτηση σημείων του συνόλου δεδομένων που να απέχουν το πολύ απόσταση ϵ από το μαθημένο όριο απόφασης (decision boundary) και έπειτα η (ελάχιστη) αλλοίωση τους προκειμένου να περάσουν από το όριο απόφασης σε γειτονική κατηγορία.

να ελαχιστοποιείται το μέγεθος $\mathcal{L}_\theta(x+\delta, t))$. Στόχος δηλαδή είναι η αλλοιωμένη εικόνα να κινηθεί προς την κατεύθυνση του ορίου μεταξύ της πραγματικής κατηγορίας και όλων των υπολοίπων (ή της επιθυμητής κατηγορίας t). Ενώ η μέθοδος FGSM υπολογίζει μοναδικά το πρόσιμο αυτής της κατεύθυνσης, οι επόμενες μέθοδοι όπως η επαναληπτική μέθοδος PGD [Mad+19]και η επίθεση των Carlini και Wagner (C&W) [ND17] υπολογίζουν το δ σε πολλές επαναλήψεις και πετυχαίνουν σημαντικά μικρότερες και επιτυχείς αλλοιωσεις. Η επίθεση C&W αποτελεί μια από τις πιο δυνατές απειλές και διαφέρει ως προς τον τρόπο της μαθηματικής διατύπωσης της αντικειμενικής συνάρτησης προς ελαχιστοποίηση (ο αναγνώστης μπορεί να διατρέξει στο Κεφάλαιο 4 για την ακριβή διατύπωση) και πετυχαίνει μικρότερες αλλοιώσεις από την μέθοδο PGD η οποία τρέχει πολλαπλές επαναλήψεις της FGSM.

Επιθέσεις μαύρου-κουτιού (black-box) Το πιο ρεαλιστικό σενάριο ενός συστήματος ΤΝ είναι να μην μπορεί κανείς να έχει πρόσβαση στην εσωτερική αρχιτεκτονική, τα βάρη και τη μέθοδο εκπαίδευσης αυτού. Για αυτήν την περίπτωση έχουν αναπτυχθεί οι μέθοδοι μαύρου κουτιού οι οποίες κάτα συνέπεια δεν έχουν άμεση πρόσβαση στην κλίση της συνάρτησης απώλειας. Ωστόσο έχει αποδειχθεί ότι μπορεί κανείς να εκμετταλευτεί τις εξόδους που παράγει το ΒΝΔ (είτε αυτές είναι απλά η τελική απόφαση, είτε οι πιθανότητες μια εισόδου να ανήκει σε κάθεμια από τις κατηγορίες, γνωστά ως logits) για να παράξει επιτυχή ανταγωνιστικά παραδείγματα αλλά και για να προσεγγίσει την κλίση $\nabla_x \mathcal{L}_\theta(x)$. Συγκεκριμένα, η επίθεση ορίου απόφασης (boundary attack)

[BRB18] ανήκει σε αυτήν την κατηγορία και χρησιμοποιεί μόνο την απόφαση του δικτύου για να αναζητήση ευριστικά μια ανταγωνιστική εικόνα. Είναι επαναληπτική μέθοδος που ξεκινάει με ένα σημείο που ήδη ανήκει σε μια άλλη κατηγορία από την πραγματική, και σε κάθε βήμα επιλέγει μια τυχαία αλλοίωση τέτοια ώστε να μειώνει την απόσταση το σημείου από το αρχικό και ταυτόχρονα να συνεχίζει να ανήκει σε διαφορετική κατηγορία (τα βήματα αναλύονται στον Αλγόριθμο 1).

Άλλες επιτυχιμένες μέθοδοι όπως η προγενέστερα καθοδηγούμενη τυχαία μέθοδος χωρίς κλίση ορισμένη ως Prior-guided Random Gradient-Free method (P-RGF) [Che+20b], βασίζονται τυπικά σε λιγότερα ερωτήματα πάνω στο μοντέλο (queries) τα οποία χρησιμεύουν στο να ανακατασκευαστεί μια προσέγγιση της κλίσης, η οποία στην P-RGF με τη σειρά της βασίζεται και στην κλίση ενός υποκατάστατου (surrogate) νευρωνικού δικτύου (απ' όπου προκύπτει ο όρος prior-based). Οι επιθέσεις αυτές χρήζουν έναν πιο ρεαλιστικό κίνδυνο που παρά την περιορισμένη πληροφορία παράγουν ανεπαίσθητες αλλά εξίσου επιτυχείς επιθέσεις.

Καθολικές και απεριόριστες επιθέσεις (universal, unrestricted) Σε προηγούμενες εργασίες έχει παρατηρηθεί ότι ένας επιπρόσθετος κίνδυνος των ανταγωνιστικών επιθέσεων είναι το γεγονός ότι μπορούν να εξαπατήσουν πολλά ΒΝΔ πέραν του στόχου-θύματος της επίθεσης. Πάνω σε αυτήν την ιδέα πατάνε οι καθολικές επιθέσεις, που υλοποιούν (επαναληπτικές) μεθόδους για να την εύρεση μιας αλλοίωσης που να μπορεί να γενικευτεί ως προς την εικόνα που αλλοιώνεται αλλά και το μοντέλο-θύμα. Συνεπώς, όπως φαίνεται στην εργασία [Moo+17] οι συγγραφείς κατασκευάζουν μια αλλοίωση που αφού προστεθεί σε οποιαδήποτε εικόνα επιτυγχάνει να μπερδέψει τον ταξιμομητή με μεγάλο ποσοστό επιτυχίας, και παρατηρούν ότι αυτό συμβαίνει και αν δοθεί ως είσοδος σε διαφορετικούς ταξινομητές.

Τέλος, αξίζει να αναφερθούν οι απεριόριστες επιθέσεις οι οποίες διαφωνούν με τον ευρύ ορισμό των ανταγωνιστικών επιθέσεων υπό την έννοια του ότι δεν είναι ανεπαίσθητες, ωστόσο οδηγούν ταξινομητές σε λάθος συμπεράσματα στα οποία ο άνθρωπος είναι ανθεκτικός. Τέτοιες επιθέσεις, σύμφωνα με τους συγγραφείς του [HP18], επιτυγχάνουν να ξεγελάσουν τα ΒΝΔ με την αλλοίωση των χρωματικών καναλιών των εικόνων, αλλά εκμετταλεύονται και τα σημασιολογικά χαρακτηριστικά εικόνων ή τον παραμετρικό χώρο σημασιολογικών ιδιοτήτων Γεννητικών Ανταγωνιστικών Δικτύων (Generative Adversarial Network – GAN) [Goo+14] για να παράξουν εικόνες με την ίδια σημασιολογία αλλά αλλοιωμένα (ή πρόσθετα) χαρακτηριστικά [Jos+19].

1.3 Στιβαρότητα των Βαθιών Νευρωνικών Δικτύων

Ήδη έχουμε αναφέρει πως οι ανταγωνιστικές επιθέσεις πηγάζουν από την περιορισμένη στιβαρότητα της εξόδου των ΒΝΔ σε μικρές αλλαγές της εισόδου. Η μέθοδος της ανταγωνιστικής μάθησης (adversarial learning) αποσκοπεί στο να εκπαιδεύει πιο στιβαρά μοντέλα με το να προσπαθεί να ταξινομήσει σωστά ανταγωνιστικές εικόνες που κατασκευάζει κατά την διάρκεια της εκπαίδευσης. Με άλλα λόγια, υλοποιείται ενός είδους επαύξηση δεδομένων (data augmentation) στην οποία οφείλει η έξοδος του μοντέλο να είναι αμετάβλητη, αλλά έναντι των κοινών μετασχηματισμών χρησιμοποιούνται δυναμικά-υπολογισμένα ανταγωνιστικά παραδείγματα. Ως προς το πρόβλημα που προσπαθεί η μέθοδος αυτή να βελτιστοποιήσει, παραθέτουμε την μαθηματική έκφραση στην σχέση 3.2 όπου φαίνεται ότι αντικείμενο της βελτιστοποίησης είναι να ελαχιστοποιήσει την μέγιστη τιμή της συνάρτησης απώλειας που έχει το μοντέλο από ανταγωνιστικά παραδείγματα.

Στα πειράματα μας θελήσαμε να χρησιμοποιήσουμε μια εναλλακτική αλλά κοντινή προσέγγιση, την μέθοδο TRADES [Zha+19] η οποία μοντελοποιεί το φαινόμενο του "trade-off" μεταξύ της στιβαρότητας και της ακρίβειας ενός ταξινομητή. Αυτό εκφράζει την παρατήρηση πως τα πιο στιβαρά μοντέλα τείνουν να έχουν απώλεια στην ακρίβεια τους (όπως αυτή αποδίδεται κλασσικά, δηλαδή ως το ποσοστό των δεδομένων ελέγχου που ταξινομούνται ορθά) καθώς η προσπάθειά τους να κατηγοριοποιήσουν σωστά τα αλλοιωμένα σημεία μπορεί να έχει αρνητική επίδραση στα σημεία που βρίσκονται κοντά στα όρια μεταξύ των κατηγοριών. Σύμφωνα λοιπόν με την παρούσα μέθοδο, αυτά τα σημεία που βρίσκονται σε απόσταση το πολύ ϵ από το όριο απόφασης περιγράφουν το οριακό σφάλμα (boundary error), τα σημεία που κατηγοριοποιούνται λάθος από τον ταξινομητή το φυσικό σφάλμα (νατυραλ ερρορ) και τέλος το στιβαρό σφάλμα (robust error) αποτελεί το άθροισμα των προηγούμενων δύο σφαλμάτων. Συνεπώς, γίνεται προφανές το ότι αν θελήσουμε να ελαχιστοποιήσουμε το $\mathcal{R}_B$ είτε θα αυξηθεί το $\mathcal{R}_N$ και θα μειωθεί το $\mathcal{R}_{Rob}$, είτε το αντίθετο. Η παράμετρος λ του αλγόριθμου αυτού (περιγράφεται στον Αλγόριθμο 2) που εκφράζει το "trade-off" εμφάνισε ιδιαίτερο ενδιαφέρον στα πειράματα μας στο πεδίο της συχνότητας.

Πέραν της δυνατότητας των στιβαρών ΒΝΔ να αμυνθούν απέναντι στις ανταγωνιστικές επιθέσεις, προηγούμενες εργασίες δείχνουν ότι υπάρχει σημαντική διαφορά στους χάρτες χαρακτηριστικών (feature maps) που μαθαίνουν τα ΣΝΔ και ότι τα αποτελέσματα τους ερμηνεύουν καλύτερα την συσχέτιση

εισόδου-εξόδου. Για παράδειγμα, σύμφωνα με τα αποτελέσματα των [Tsi+19] στην Εικόνα 3.2 βλέπουμε την απεικόνιση της κλίσης της συνάρτησης απώλειας ως προς τα εικονοστοιχεία (pixel) της εισόδου, δηλαδή το μέγεθος $\nabla_x \mathcal{L}$. Αυτό εξ' ορισμού απεικονίζει τα εικονοστοιχεία που έχουν μεγαλύτερη επιρροή στην τελική απόφαση του ταξινομητή και είναι εμφανές ότι οι στιβαρώς εκπαιδευμένοι ταξινομητές (δύο τελευταίες σειρές) έχουν με διαφορά καλύτερες ερμηνείες από την κλίση σε σχέση με τους τυπικά εκπαιδευμένους (δεύτερη σειρά) που δεν συνάδουν με την ανθρώπινη αντίληψη. Ακόμη, στην Εικόνα 3.1 οι συγγραφείς έκαναν παρεμβολή από μία κατηγορία σε μία διαφορετική (στην πάνω σειρά από πίθηκο σε σκύλο και τελικά γάτα) και δείχνουν ότι τα χαρακτηριστικά αλλάζουν προς μια κατεύθυνση ευθυγραμμισμένη με την σημασία της δεύτερης κατηγορίας.

1.4 Συνελικτικά Νευρωνικά Δίκτυα

Στις περισσότερες προηγούμενες εργασίες στις οποίες έχουμε κάνει αναφορά όπως και στην παρούσα εργασία καταπιανόμαστε με το γενικό πρόβλημα της ταξινόμησης εικόνων χρησιμοποιώντας Συνελικτικά Νευρωνικά Δίκτυα (ΣΝΔ). Τα δίκτυα αυτά που αποτελούν υποκατηγορία των ΒΝΔ καταφέρνουν αξιομνημόνευτα αποτελέσματα σε διαφόρων ειδών σύνολα δεδομένων και εδώ και πολλά χρόνια αποτελούν την βάση για τις περισσότερες προσεγγίσεις στον τομέα της όρασης υπολογιστών. Τα δίκτυα αυτά είναι εμπνευσμένα από τον τρόπο λειτουργίας του οπτικού συστήματος των ζωντανών οργανισμών που λειτουργούν ιεραρχικά με μία σειρά από επίπεδα νευρώνων και σύνθεσης χαρακτηριστικών, το οποίο αποδεικνύεται συνεχώς ακόμη και σε νέες διεξοδικές συμπεριφορικές έρευνες [Lan+21].

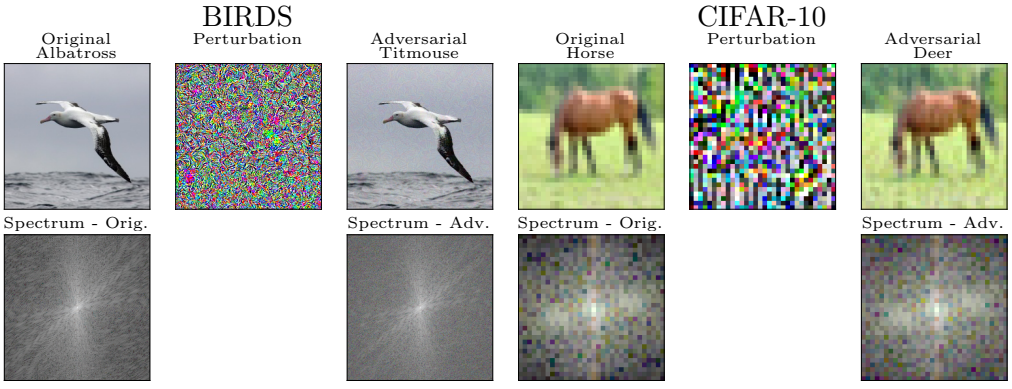
Σε αναλογία με το βιολογικό οπτικό σύστημα, τα ΣΝΔ έχουν ως χαρακτηριστικό τα χωρικά δισδιάστατα φίλτρα (spatial filters) που εφαρμόζονται πάνω στην δισδιάστατη εικόνα-είσοδο και στοχεύουν στο να μάθουν διαφόρων ειδών χαρακτηριστικά ανάλογα με τα δεδομένα εκπαίδευσης. Ο τρόπος που διατρέχουν την εικόνα βασίζεται στην πράξη της συνέλιξης, όπου το προς μάθηση φίλτρο διασχίζει ανα οριζόντια σειρά (μπορεί και να διασχίζει ανα πολλαπλές σειρές) την εικόνα και πολλαπλασιάζονται τα αντίστοιχα εικονοστοιχεία μεταξύ τους για να παράξουν μια δισδιάστατη έξοδο που ονομάζεται χάρτης χαρακτηριστικών. Όπως δείχνει και η Εικόνα 4.2, μπορούν να υπάρχουν πολλαπλά επίπεδα φίλτρων σε ένα ΣΝΔ (τα πρόσφατα ΣΝΔ έχουν δεκάδες επίπεδα και χιλιάδες νευρώνες στο σύνολο τους) όπου ο χάρτης χαρακτηριστικών του $i-$οστού επιπέδου αποτελεί την είσοδο στο φίλτρο του $(i+1)-$οστού επιπέδου. Ομοίως

μπορούν να υπάρχουν πολλαπλά φίλτρα σε ένα επίπεδο τα οποία μαθαίνουν ποικίλα χαρακτηριστικά. Ο λόγος που καθιστά χρήσιμα τα πολλαπλά επίπεδα είναι η δυνατότητα να μαθαίνονται έτσι όλο και πιο σύνθετα χαρακτηριστικά, τα οποία συντίθενται από τα προηγούμενα επίπεδα φίλτρων, συναρτήσει του βάθους που βρίσκονται τα φίλτρα.

Πέραν των συνελικτικών επιπέδων, τα ΣΝΔ αποτελούνται από επίπεδα pooling που στοχεύουν στο να μειώσουν την χωρική διάσταση των χαρτών χαρακτηριστικών αλλά και να βοηθήσουν στην γενίκευση και την μείωση της υπερεκπαίδευσης. Ακόμη, στο πρόβλημα της ταξινόμησης το ΣΝΔ μπορεί να διαχωριστεί στο κομμάτι της εξαγωγής χαρακτηριστικών (που πραγματοποιείται από τα πολλαπλά συνελικτικά επίπεδα) και τον ταξινομητή, ο οποίος ουσιαστικά παίρνει ως είσοδο τα εξαγμένα χαρακτηριστικά και τα περνάει από ένα ή παραπάνω επίπεδα νευρώνων μέχρι την έξοδο που αποτυπώνει την πιθανότητα της εισόδου να ανήκει σε κάθε μία από τις κατηγορίες Για μια υψηλού-επιπέδου παρουσίαση της αρχιτεκτονικής παραπέμπουμε τον αναγνώστη και πάλι στην Εικόνα 4.2.

1.5 Πειραματικά αποτελέσματα

Όπως έχουμε αναφέρει περιληπτικά στις προηγούμενες παραγράφους, στην εργασία αυτήν υλοποιήσαμε τόσο επιθέσεις σε ΣΝΔ, όσο και μια μέθοδος άμυνας και παρατηρήσαμε την συμπεριφορά των ανταγωνιστικών παραδειγμάτων στο πεδίο της συχνότητας σε κάθε μια από αυτές τις περιπτώσεις.



Σχήμα 1.2: Ανταγωνιστικά παραδείγματα για καθένα από τα σύνολα δεδομένων και το συχνοτικό τους φάσμα (δεύτερη σειρά). Αριστερά έχουμε στην πρώτη στήλη την αρχική εικόνα, στη δεύτερη την αλλοίωση που υπολόγισε η μέθοδος PGD^{∞} και στην τρίτη την αλλοιωμένη εικόνα. Ομοίως και για τις δεξιότερες τρείς εικόνες.

Επιθέσεις σε Συνελικτικά Νευρωνικά Δίκτυα Για την εξέταση των ανταγωνιστικών επιθέσεων εκπαιδεύσαμε ποικίλα ΣΝΔ πάνω στα σύνολα δεδομένων CIFAR-10 [KH+09] και 350 Birds species [Ger21] (θα αναφερόμαστε σε αυτό ως BIRDS). Συγκεκριμένα, χρησιμοποιήσαμε και εκπαιδεύσαμε από την αρχή μια απλή αρχιτεκτονική ΣΝΔ για το CIFAR-10 βασισμένη στα πειράματα των [ND17] και ένα ResNet-34 με αρχικά προεκπαιδευμένα βάρη στο σύνολο δεδομένων ImageNet [Rus+15] για το BIRDS σύνολο (στο Παράρτημα A υπάρχουν και τα αποτελέσματα επιθέσεων στα μοντέλα GoogleNet και EfficientNet_B0). Τα προεκπαιδευμένα βάρη έκανα μεγάλη διαφορά στην τελική ακρίβεια του μοντέλου που ξεπερνάει το 99% και στην ταχύτητα της εκπαίδευσης (χρειάστηκαν λιγότερες από 10 επαναλήψεις (epochs) του συνόλου δεδομένων).

Οι επιθέσεις που υλοποιήσαμε είναι λευκού-κουτιού και βασισμένες στην κλίση της απώλειας, συγκεκριμένα η C&W και PGD επίθεση. Συγκρίναμε και μια υλοποίηση ανοιχτού κώδικα της οριακής επίθεσης (από την βιβλιοθήκη foolbox [RBB18] της Python) η οποία είναι μέθοδος μαύρου-κουτιού ώστε να έχουμε μια ευρύτερη εικόνα των απειλών. Οι επιτιθέμενοι κινήθηκαν τόσο με l_2 ευκλείδιες αποστάσεις από την αρχική εικόνα, όσο και με l_∞ αποστάσεις, οι οποίες περιορίζουν-υπολογίζουν την μέγιστη (επιτρεπτή) αλλοίωση ενός εικονοστοιχείου. Στους Πίνακες 5.3 και 5.4 φαίνονται τα αποτελέσματα μας για το CIFAR-10 και αντίστοιχα στους Πίνακες 5.5 για το BIRDS. Οι μετρικές που χρησιμοποιήσαμε είναι το ποσοστό των εικόνων για τις οποίες βρέθηκε έγκυρη (δηλαδή δυνατή να αλλάξει την κατηγορία της εικόνας) αλλοίωση καθώς και την Ευκλείδια και l_∞ απόσταση.

Πέραν της υψηλης επιτυχίας των επιθέσεων μας ενάντια σε ΣΝΔ ταξινομητές, παρατηρούμε την συχνοτική κατανομή των αλλοιώσεων για τιε διαφορετικές μεθόδους και τα σύνολα δεδομένων. Σημειώνουμε σαν γενική παρατήρηση βασισμένη σε προηγούμενες εργασίες [Yin+20] [JB17], ότι τα ΒΝΔ κωδικοποιούν τόσο υψηλές όσο και χαμηλές συχνότητες για να μάθουν το δοσμένο σύνολο δεδομένων (μάλιστα γενικεύουν καλύτερα αν χρησιμοποιήσουν χαρακτηριστικά εικόνων μη αντιληπτά στους ανθρώπους που τείνουν να είναι υψηλών συχνοτήτων), και επιπροσθέτως ότι ανάλογα με τις συχνότητες που μαθαίνουν, γίνονται ευαίσθητα σε αλλοιώσεις αυτών. Παρατηρούμε μέσα από την Εικόνα 5.3 ότι στην περίπτωση του CIFAR-10 υπάρχει ευρεία κατανομή στις συνιστώσες συχνότητας που μεταβάλλονται, σε σύγκριση με αυτές του BIRDS της Εικόνας 5.4 που είναι με διαφορά πιο έντονες στις χαμηλές συχνότητες. Επιπλέον, οι επιτιθέμενοι με l_2 και l_∞ νόρμα απόστασης παρουσιάζουν διαφορές μεταξύ τους στην συχνότητα το οποίο είναι αναμενόμενο καθώς στην δεύτερη περίπτωση είναι προσδιορισμένη η (μέγιστη) αλλοίωση που επιτρέπεται

σε επίπεδο εικονοστοιχείου και όχι της συνολικής εικόνας όπως στην πρώτη περίπτωση.

Η Ανταγωνιστική Μάθηση ως μηχανισμός άμυνας Αφού αποδείξαμε ότι μπορεί κανείς να ξεγελάσει με 100% επιτυχία τα εκπαιδευμένα ΣΝΔ (παρότι έχουν πολύ υψηλή ακρίβεια στην ταξινόμηση) εξετάσαμε την ικανότητα της ανταγωνιστικής μάθησης με τη μέθοδο TRADES να μειώσει τον βαθμό επιτυχίας των επιτιθέμενων. Όπως αναφέραμε εν συντομία παραπάνω, στην διατύπωση της συνάρτησης απώλειας του TRADES μπορεί κανείς να διαβαθμίσει την σημασία της ακρίβειας ενάντι της στιβαρότητας στην εκπαίδευση. Από τα πειράματα που τρέξαμε αυτή η διαβάθμιση που γίνεται με την παράμετρο λ έχει μεγάλη επιρροή τόσο στα αποτελέσματα των επιθέσεων (Πίνακες 5.6 και 5.7) όσο και στο συχνοτικό αποτύπωμα τους όπως φαίνεται στις Εικόνες B.1 (στην περίπτωση του CIFAR-10) και 5.4 (στην περίπτωση του BIRDS, πλην της πρώτης σειράς).

Συγκεκριμένα, βλέπουμε ότι οι αλλοιώσεις των επιθέσεων συγκεντρώνονται στις χαμηλές συχνότητες σε αναλογία με την στιβαρότητα του μοντέλου (σημειώνουμε ότι όσο μειώνεται η τιμή λ τόσο πιο στιβαρό γίνεται ένα μοντέλο). Αυτό δείχνει πως τα μοντέλα γίνονται ανθεκτικά σε μεταβολλες των υψηλών συχνοτήτων, δηλαδή ότι δεν τις χρησιμοποιούν σημαντικά για γενίκευση στο σύνολο δεδομένων. Ταυτόχρονα, οι χαμηλές συχνότητες παίζουν μεγαλύτερο ρόλο στην κατηγοριοποίηση της εισόδου και μεταβολλές τους οδηγούν σε λάθος ταξινόμηση. Αξίζει να αναφέρουμε εδώ ότι οι χαμηλές και μέσες συχνότητες παρουσιάζουν χαρακτηριστικά σε μία εικόνα που είναι διακρίσιμα στο ανθρώπινο μάτι και ως αναλογία αντιστοιχούν σε μεγάλο βαθμό στα "στιβαρά" χαρακτηριστικά (robust features) όπως χρώματα, σχήματα και περιγράμματα που πρέπει να μάθει ένα μοντέλο και περιέχουν την κύρια σημασιολογία μιας εικόνας.

Επιπλέον, παρότι πετυχαίνουμε να μειώσουμε το ποσοστό επιτυχίας των επιθέσεων (δηλαδή το ποσοστό των εικόνων για τις οποίες μπορεί μια επίθεση να υπολογίσει κακόβουλες αλλοιώσεις) δεν φτάνουμε ποτέ κοντά σε 0% επιτυχία, και επιπλέον σε πολλές περιπτώσεις η εκπαίδευση είναι ασταθής ως προς την απώλεια (ένα φαινόμενο που το συναντήσαμε πολύ έντονα στην περίπτωση του EfficientNet_B0). Αυτό δείχνει ότι οι παράμετροι της ανταγωνιστικής εκπαίδευσης την καθιστούν πολύ ευαίσθητη και πρέπει να επιλέγονται με προσοχή και έπειτα από πολλές δοκιμές. Επίσης, καθορίζουν το πόσο δυνατές επιθέσεις θα χρησιμοποιηθούν κατά την εκπαίδευση (αφού όπως είπαμε παραπάνω υπολογίζονται ανταγωνιστικά παραδείγματα για κάθες δεδομένο εκπαίδευσης) και μια πιθανότητα είναι η επίθεση που χρησιμοποιήσαμε σύμφωνα με τους [Zha+19]

να μην είναι επαρκής. Όλα τα παραπάνω είναι μια κατεύθυνση για μελλοντική έρευνα και πρακτικές λεπτομέρειες που όμως θα οδηγήσουν σε δυνατά και στιβαρά ΒΝΔ.

1.6 Σχολιασμός και μελλοντικές κατευθύνσεις

Στην παρούσα εργασία εντρυφήσαμε στο πρόβλημα των ανταγωνιστικών επιθέσεων οι οποίες αποτελούν σημαντική παραβίαση της εμπιστευτικότητας και αξιοπιστίας των ΒΝΔ. Μελετήσαμε δύο ευρέως διαδεδομένες μεθόδους επίθεσης λευκού-κουτιού, υποθέτοντας ότι έχουμε πλήρη πρόσβαση σε όλες τις παραμέτρους των μοντέλων προς επίθεση. Το υποθετικό μας σύστημα ασχολείται με την ταξινόμηση εικόνων από τα σύνολα δεδομένων CIFAR-10 και 350 Birds Species χρησιμοποιώντας γνωστές αρχιτεκτονικές ΣΝΔ. Οι επιθέσεις που υλοποιήσαμε είναι οι C&W και PGD ενώ τις συγκρίναμε και με μια επίθεση μαύρου-κουτιού υλοποιημένη στην βιβλιοθήκη ανοιχτού κώδικα foolbox όπου δεν χρησιμοποιήθηκε καμία εσωτερική γνώση για τα μοντέλα πέρεαν της τελικής απόφασης. Όλες οι μέθοδοι ήταν ικανές να υπολογίσουν πολύ μικρές αλλοιώσεις που μεταβάλουν την τελική ταξινόμηση της εικόνας-στόχου σε ποσοστό μεγαλύτερο του 80% από όλες τις εικόνες που επιλέχθηκαν τυχαία (από 200 εώς 300 το πλήθος ανά επίθεση).

Έπειτα ερευνήσαμε την μέθοδο ανταγωνιστικής μάθησης TRADES που δυναμικά υπολογίζει κακόβουλες (ανταγωνιστικές) αλλοιώσεις κατά την εκπαίδευση και τη χρησιμοποιήσαμε για να εκπαιδεύσουμε στιβαρά μοντέλα, ανθεκτικά στις παραπάνω επιθέσεις. Η μέθοδος αυτή, που παρέχει την δυνατότητα διαστάθμισης ανάμεσα στην στιβαρότητα και ακρίβεια ενός ΒΝΔ, βοήθησε στην μείωση της επιτυχίας των επιθέσεων αλλά και στην παρατήρηση της διαστάθμισης αυτής μέσω πολλαπλών πειραμάτων που τρέξαμε με διαφορετικούς συνδυασμούς παραμέτρων. Παρόλα αυτά για κάποια ΣΝΔ απεδείχθη ασταθής μέθοδος εκπαίδευσης και ιδιαίτερα ευαίσθητη στην επιλογή της επίθεσης πάνω στην οποία γίνεται η μάθηση.

Για να κατανοήσουμε καλύτερα ποια συστατικά στα δεδομένα, την εκπαίδευση αλλά και την μέθοδο επίθεσης καθιστούν τα ΣΝΔ αδύναμα απέναντι σε διακριτικές στοχευμένες αλλοιώσεις της εισόδου κάναμε ανάλυση αυτών στο πεδίο Fourier. Προβάλλοντας τον δισδιάστατο μετασχηματισμό Φουριερ όλων των πετυχημένων εικόνων αθροιστικά, παρατηρήσαμε ότι στην περίπτωση των αρχικών μη-στιβαρών μοντέλων η κατανομή της αλλοίωσης είναι φανερή τόσο στις χαμηλές όσο και στις μέσες και υψηλές (σε κάποιο βαθμό) συχνότητες,

σαφώς ανάλογα με το σύνολο δεδομένων, καθώς καθένα έχει διαφορετικό μέγεθος πληροφορίας κατανεμημένο στις συνιστώσες συχνότητας (με το CIFAR-10 να έχει πιο ευρεία κατανομή απ' ότι το 350 Birds species που συγκεντρώνει πληροφορία κυρίως σε χαμηλές συνιστώσες όπως βλέπουμε στο φασματογράφημα των συνόλων δεδομένων στην Εικόνα 5.2).

Η απεικόνιση που παίρνουμε για τις επιθέσεις στα στιβαρά μοντέλα που εκπαιδεύσαμαε με την μέθοδο TRADES διαφέρει κατά πολύ και για τα δύο σύνολα δεδομένων. Παρατηρούμε μια μετατόπιση της αλλοίωσης στις χαμηλότερες συχνότητες σε σχέση με την κατανομή που είδαμε στην προηγούμενη περίπτωση, φαινόμενο το οποίο συμβαίνει σε όλες τις επιθέσεις και τα ΣΝΔ. Το στοιχείο αυτό ανοίγει την συζήτηση σχετικά με την φύση των στιβαρών χαρακτηριστικών των δεδομένων που μαθαίνει ένα ανθεκτικό ΒΝΔ και που όπως είναι αναμενόμενο ζουν στις χαμηλές συχνότητες, σε αντιπαράθεση με τα μη-στιβαρά χαρακτηριστικά που χρησιμοποιούν τα κοινά ΒΝΔ για γενίκευση. Στο σημείο αυτό υπάρχει ανάγκη για περαιτέρω συστηματικά πειράματα με διαφορετικές αρχιτεκτονικές (τελευταίας τεχνολογίας – state of the art), μοντέρνες δυνατές επιθέσεις και ρεαλιστικά σύνολα δεδομένων.

Άλλες μελλοντικές κατευθύνσεις είναι η επέκταση της ανταγωνιστικής μάθησης με πιο δυνατές δυναμικές επιθέσεις έναντι της επίθεσης PGD που χρησιμοποιείται αποκλειστικά. Όσον αφορά την ανάλυση των συχνοτήτων των ανταγωνιστικών αλλοιώσεων, απαιτείται ένας συστηματικός τρόπος για να ξεχωρίσουν τα χαρακτηριστικά των στιβαρών και μη-στιβαρών στοιχείων που περιέχουν οι εικόνες στο πεδίο Fourier. Συγκεκριμένα, μπορεί να δοκιμάσει κανείς να εκπαιδεύσει μοντέλα σε εισόδους με αποκομένες κάποιες συχνότητες και να συγκρίνει τις επιθέσεις σε αυτά ανάλογα με τις συχνότητες που αφαιρούνται, αλλά και την δυνατότητα γενίκευσης τους στις αρχικές εισόδους. Για παράδειγμα έχει αποδειχθεί ότι ενώ οι χαμηλές συχνότητες μεμονωμένα πετυχαίνουν σημαντική ακρίβεια στα ΣΝΔ, οι υψηλές βοηθούν να ενισχυθεί αυτή και να πάρει την μέγιστη τιμή της παρότι σημασιολογικά δεν προσφέρουν πληροφορία στον άνθρωπο [Yin+20]. Δεν είναι γνωστό όμως ακόμα αν αυτό το γεγονός μπορεί να αποφευχθεί, αν είναι χρήσιμο να εκπαιδεύει κανείς ένα "ensemble" μοντέλων που θα είναι ανθεκτικά σε διαφορετικές αλλοιώσεις-επιθέσεις ή αν ο τρόπος αξιολόγησης της ικανότητας γενίκευσης των ΣΝΔ (με την μετρική της ακρίβειας) είναι αντιπροσωπευτικός.

Ευελπιστούμε να ωθήσουμε τους ενδιαφερόμενους αναγνώστες να σκεφτούν εις βάθος τους παράγοντες που επηρεάζουν την ασφάλεια και την αξιοπιστία των ΒΝΔ . Παράλληλα, είμαστε αισιόδοξοι ότι η επιστημονική κοινότητα στα επόμενα χρόνια θα υιοθετήσει πρακτικές ανάπτυξης μεθόδων στα πλαίσιο της Τεχνητής Νοημοσύνης (ΤΝ) που από την μία συνάδουν καλύτερα με την

ανθρώπινη αντίληψη (που στους περισσότερους τομείς είναι παράδειγμα προς μίμηση για την TN), αλλά από την άλλη μπορούν να λειτουργήσουν άρρηκτα σε ακραίες συνθήκες και να συνεχίσουν να βελτιώνουν με ασφάλεια την καθημερινότητα μας.

Chapter 2

Introduction

It is widely known that machines are able to solve a variety of challenging problems in almost all known areas such as Artificial Intelligence (AI), biology and economics, even up to the extent of surpassing human-level performance [He+15b] [Dav+16]. These radical technological advances are implemented and used across many fields, either for scientific research or in our everyday lives and needs. This extensive use of technology - although crucial - can potentially be used maliciously.

In the past few years Machine Learning (ML) researchers have shed light on possible ways one can leverage ML systems to cause undesired outcomes. Since Deep Learning (DL) and ML have continuously demonstrated powerful results in problems such as protein structure prediction [Jum+21], 3D image reconstruction [ZL21], self driving cars [YWY18] and realistic music generation [Dha+20] it is of great importance to understand how these techniques could fail or be harmful.

Focusing on the area of Computer Vision (CV), there exist various malicious threats to almost all used architectures and tasks like object detection, classification and face recognition. The well-studied Adversarial Attack (AA) threat can manipulate the input space of a classifier and create new adversarial inputs from benign ones that are misclassified by the classifier, without an observer being able to perceive this manipulation in the input. Another form of threat that can occur while training a model are back-door or Trojan attacks [Yun+20], where the goal is to inject some trigger patterns to the training data so that the model learns some false association to the labels. Membership Inference Attacks MIAs [Che+20a] and model inversion are two of the most serious privacy threats since they aim at gaining knowledge of the training data, meaning that they could potentially reveal sensitive infor-

mation contained in the data. These attacks, to name but a few, show how crucial research in these problems can be but will hopefully also highlight ways to create more secure and robust technological frameworks. We highly recommend the interested readers to refer to [Akh+21] for an extensive review of such diverse threats.

2.1 Adversarial attacks

Adversarial attacks are arguably very concerning in terms of their proven high effectiveness in various ML and DL settings but also for revealing an intrinsic vulnerability in the way many models work. In other words, beyond the practical security issues that arise with adversarial examples, which mainly apply in malware detection [SCJ19] and ad-blocking [Tra+19], they bring forward issues in the learning and generalization ability of NN models. As discussed earlier, the main goal of an adversarial attacker is to fool a target model, i.e. to create an input (e.g. an image) that doesn't yield the expected output label. By definition, these attacks produce mainly norm-bounded perturbations, i.e. that lie in an ϵ-ball centered at the input, which are extremely low-magnitude and imperceptible to humans.

In [GSS15] the authors first presented this phenomenon and proposed an initial explanation, namely the fact that models are too linear in combination with their high-dimensional input space. In other words, they stated that many infinitesimal perturbations to the input can add up to large changes in the output. This weakness was also connected to the nature of the features that they learn and optimize over. Specifically, researchers noticed that models tend to learn features not meaningful or visible to humans [Yin+20], such as strangely biased representations of the input towards e.g. textures instead of meaningful features like shapes [Gei+18], and choose to utilize non-robust features – that are nevertheless useful for generalization - that can easily be manipulated towards fooling a model [Ily+19]. Non-robust features are defined here as input features that once altered infinitesimally might be negatively correlated to the true label of the input.

Depending on the system information that an attacker has access to, AAs can be characterized as black-box and white-box. Simply put, in the former attackers have access solely to the output of the target model while in the later they have access to the entire architecture and parameters. Black-box attacks are generally either transfer-based, i.e. the attacker creates adversarial examples by attacking a white-box model and then uses them to attack

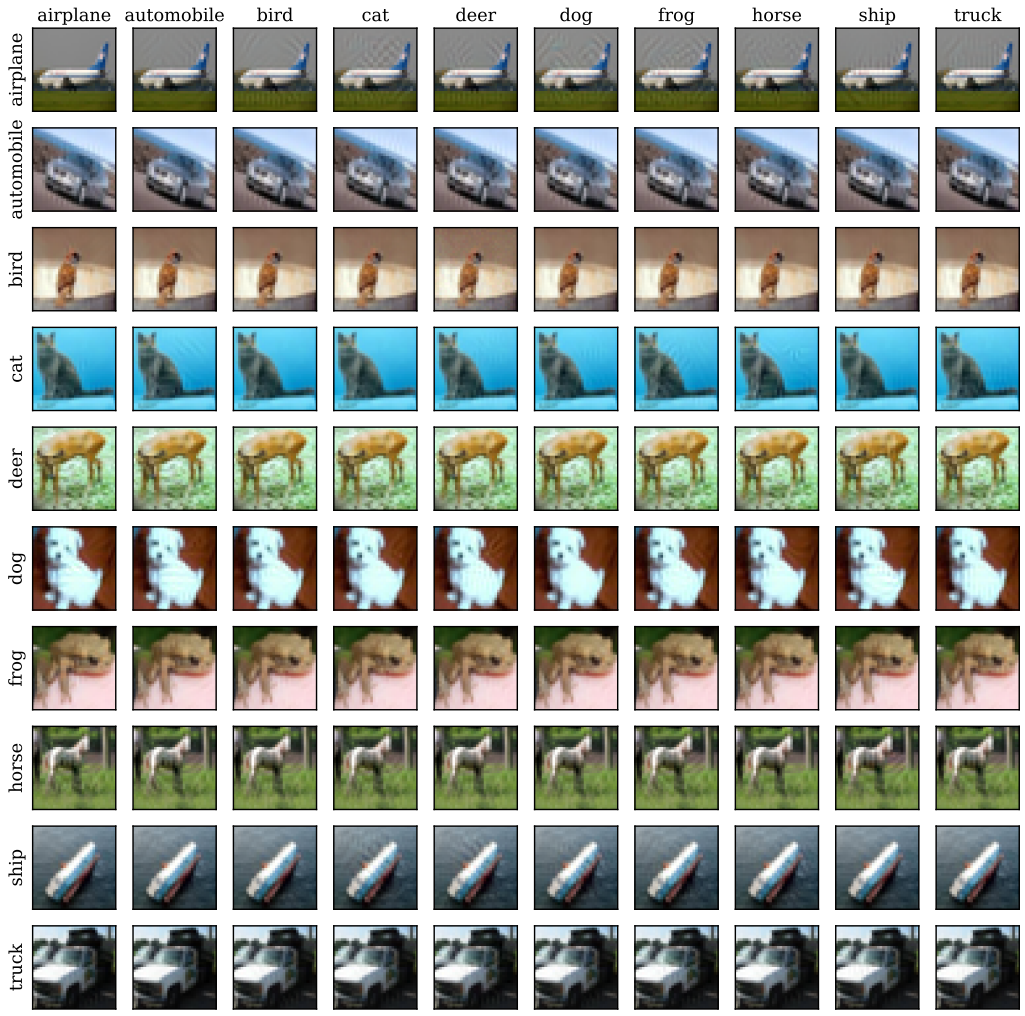


Figure 2.1: Adversarial examples for all 10 classes of the CIFAR-10 dataset, generated by the C&W attack. Each row represents the true class of the image and the columns represent the target class.

the black-box target model, or query-based, meaning that through multiple queries to the target model adversarial examples are constructed by utilizing the output in a specific way. At the same time, white-box attacks are gradient-based, i.e. they use gradient ascent over the model's loss surface to find inputs that can fool it. On a final note it is worth mentioning that although most AAs are in general norm-bounded, an extension to non-bounded attacks exists known as unrestricted attacks where feature manipulation such as color-shift and change in texture is applied. Universal attacks that fool

various target models at once are also possible and surprisingly easy to construct [Moo+17].

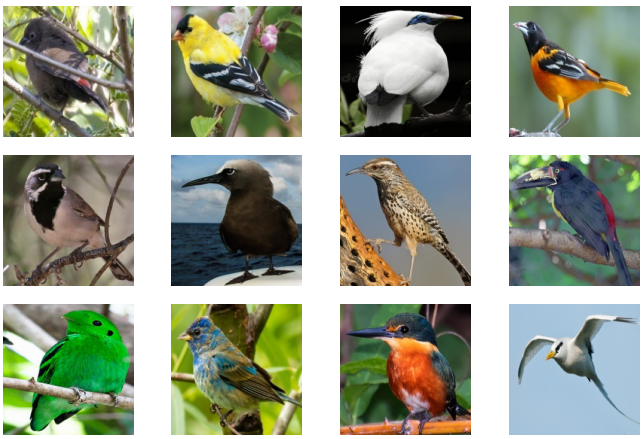


Figure 2.2: Sample images from the 350 Birds species [Ger21] dataset.

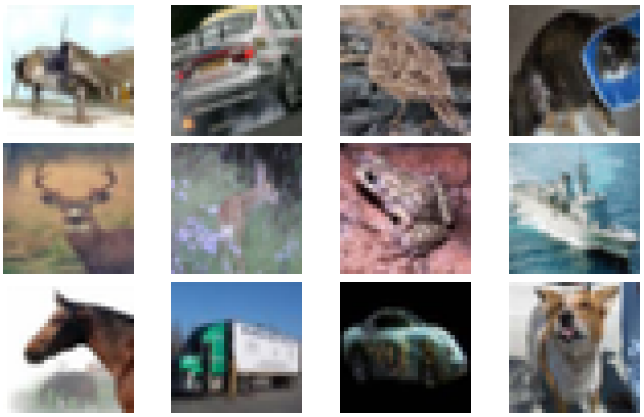


Figure 2.3: Sample images from the CIFAR-10 [KH+09] dataset.

2.2 (Adversarial) Robustness

By characterizing a ML model as robust, it is thought to achieve high accuracy even in diverse settings where the test data are significantly (but not semantically) different from the train data, e.g. including additive noise, distributional shifts, adversarial examples etc. In practice though a model can be robust to a restricted subset of data corruptions depending on the training data distributions, as proven in [Yin+20].

In this section we briefly want to mention various methods for achieving robustness to adversarial attacks. The most obvious and empirically most successful choice of method is adversarial training [GSS15], where the goal is to train a model explicitly on adversarial examples, thus redefining the optimization problem from empirical error minimization to minimizing the maximum error of adversarial inputs. Many extensions of this method use a regularization term between benign and adversarial inputs, by incorporating both the natural and robust error in the optimization function [Zha+19] [Bai+21].

Adversarial training can be though of as a type of data augmentation, although other augmentation methods can also improve robustness, like AutoAugment [Cub+19] which is essentially a mixture of such methods. To improve robustness over white-box attacks, gradient masking can be used to add noise and complexity to the loss surface of a model, although this type of defense has been surpassed by adaptive attacks as seen in [ACW18]. A more general approach to measuring robustness is embodied through certified robustness, where one aims at providing a bounded area in the input space in which models are proven to be robust to. Lastly, we should also note that instead of making a model robust, researchers have also been interested in constructing mechanisms to detect adversarial examples as a means of more robust ML systems [Akh+21].

2.3 Contribution

In this work we took a closer look to the aforementioned Adversarial Attacks problem in the scope of image classification (though many similar threats also exist for speech-to-text problems, generative models etc.). Our setting consists of an adversary who is trying to fool a target CNN image classifier and a dataset that is used for training and inference. We experimented on the CIFAR-10 [KH+09] and 350 Bird species [Ger21] datasets (referred to as BIRDS) with three different attack methods, two of them being white-box attacks and one black-box. An example of our produced corruptions can be seen in Figure 2.1.

We aim at unveiling the origin of such malicious perturbations by adopting a Fourier-domain perspective on adversarial examples, similar to approaches in [Yin+20], [Har+21] and [JB17]. Analyzing images in the frequency domain has been widely used as a tool with applications in image filtering, edge detection and compression. The effect of high and low fre-

quencies in human visual perception shows contradicting characteristics to image perception by CNNs [Yin+20] [JB17] and therefore is a fruitful direction of research. To this end, we analyze the frequency components that attacks modify and compare them with respect to the frequency distribution of dataset images, the attack method and most importantly the training method. We chose adversarial training with TRadeoff-inspired Adversarial DEfense via Surrogate-loss minimization (TRADES) [Zha+19] as a comparative robust method to natural training and found interesting properties related to robustness. Our final objective is to observe whether robust and non-robust features of datasets are visible in the Fourier domain.

2.4 Thesis structure

In the introductory chapter we presented the objectives of our research and stated the problem of adversarial attacks at a high level. In Chapter 3 we will discuss previous approaches to the problem of adversarial attacks and robustness in general but also their frequency characteristics that are currently understood and the origins of our methodology. Chapter 4 provides the essential theoretical background needed to understand and potentially reproduce our method. This includes a definition of the attack algorithms used, the TRADES adversarial training method, the Fourier analysis in our experiments as well as the core ideas of CNNs for image classification. In Chapter 5 we have a detailed description of our methodology and hypothesis that connects to the previous related works and background theory. We also examine the effectiveness of our experimental setup followed by our main findings and Chapter 6 summarizes the impact and future directions of our work.

Chapter 3

Related Work

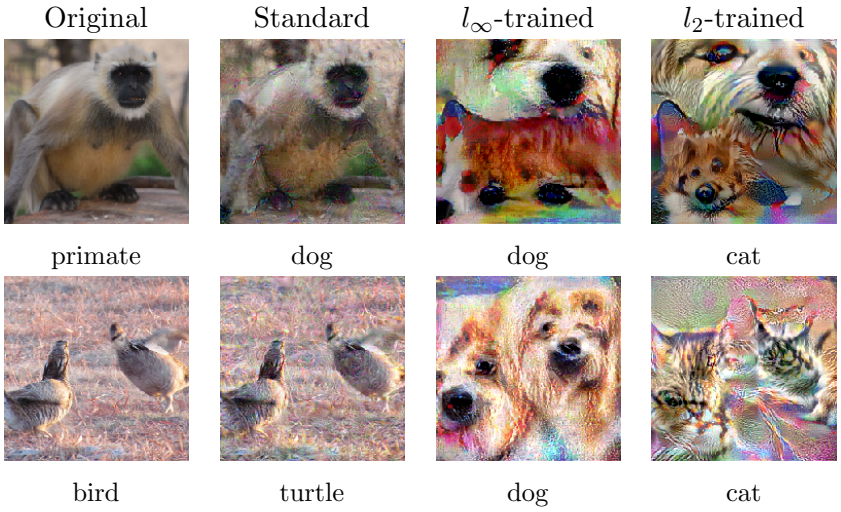
After introducing our primary goals in the previous chapter, we now aim at addressing the building blocks of our ideas. To approach the problem of adversarial attacks more broadly we present the basic intuition behind many notable attack methods in Section 4.3 that span with respect to the CNN's architecture information that is utilized, the tools and the objective of the attack. Also, in Sections 3.3 and 3.2 we discuss the fundamental principles behind robustness and the notion of robust and non-robust features, which characterise the learned image representations of CNNs. Lastly, we present previous works that have observed the frequency space of adversarial examples in order to understand their nature and ways to detect them.

3.1 Adversarial attacks on classifiers

Adversarial attacks where first introduced roughly in 2014 by [GSS15] and have since played an enormous role in deep learning research. Here we present a brief overview of various attack approaches in a somewhat historical temperament. The first generation of attacks where mainly white-box attacks but in the recent years scientists also managed to develop impressive black-box attacks that correspond to more realistic adversarial settings.

3.1.1 White-box and first-generation attacks

FGSM The authors in [GSS15] pioneered adversarial attacks by introducing a simple one-step attack known as the Fast Gradient Sign Method (FGSM) – although they implemented a previous approach in [Sze+14b] called the L-BFGS attack that worked almost identically but with much



Source: [Tsi+19]

Figure 3.1: Inter-class interpolation visualized in adversarial examples of large ϵ-bounded perturbations for standard and adversarially trained models. While there are no clear target-class features represented in the case of the standard model, they appear strongly in the adversarially trained models.

weaker results -and producing attacks with up to 87.15% success rate and confidence 96.6% with an l_∞ distortion as small as $\epsilon = 0.1$. Their algorithm uses the loss gradient to find the direction that maximizes the loss of the true label and thus moving the sample away from its original class (or towards the class boundaries). A full explanation and formalization of this attack can be found in Chapter 4 and thus we will not go into further details in this section.

JSMA Instead of using a gradient-based algorithm, in [WX18] the authors use saliency maps and the forward derivative ∇F, which is derived by:

$$\nabla F = \frac{\partial F(X)}{\partial x} = \left[\frac{\partial F_j(X)}{\partial x_i}\right]_{j \in 1..M, i \in 1..N}$$

where F is the classifier's probability output, M the number of classes and N the dimensionality of input X, to find the input pixels that will mostly move the output probabilities towards an incorrect (or target) class. In their experiments they run a number of iterations in which the two pixels with maximum saliency map values (in other words maximum negative effect on the true class of the input) are found and modified by a fixed amount θ.

This attack produces adversarial examples for the MNIST dataset and a DNN classifier with 97.05% misclassification rate and 4.03% distortion.

Pixel attack In this attack the authors designed an algorithm that uses Differential Evolution (DE) instead of loss/output gradients and after a number of so-called generations finds the pixel (or 3-5 pixels) that yields the perturbed image with the highest target (or lowest true) class confidence. The one-pixel attack creates adversarial examples with a success rate of up to 71.66% with 75.02% confidence for targeted attacks on CIFAR-10 and 16.04% with top-1 confidence of 22.09% on ImageNet. The authors use the average RGB distortion as their metric and present a minimum distortion of 123/255 for one channel. This attack can also be classified as a black-box attack since it doesn't require any model or weight information due to use of the DE algorithm.

PGD To extend the single-step FGSM attack, in [KGB17] an iterative attack that uses FGSM as an inner step with l_∞ but also l_2 constrained adversaries was introduced. For a more real-life experimental setup, the authors measured the attack's success rate on photos of printed adversarial images instead of just the source ones, and resulted in an error rate of maximum 37.4% in the case of photos with a perturbation of l_∞-norm $\epsilon = 2/255$, compared to an error rate of 71.6% for the source adversarial images under the same perturbation size. This attack is significantly more successful that its single-step counterpart (that e.g. yields error rates of 45.5% and 64.7% respectively for $\epsilon = 2/255$).

C&W In [ND17] a different optimization-based attack was designed that was motivated by a defense mechanism called defensive distillation . Apart from bypassing this defense, the authors created one of the most effective attacks (for l_∞, l_0 and l_2 constrained adversaries) so far by redefining the optimization task of the adversary and using a well-suited objective function. We will dive into this attack in Chapter 4, but we add here that the success rate of this attack is 100% for all ImageNet, MNIST and CIFAR-10 datasets and with perturbations nearly 50% smaller than ones created by JSMA, FGSM and DeepFool.

DeepFool The DeepFool attack [MFF16] is another iterative gradient-based attack that in each iteration moves the current input towards the

nearest decision boundary (meaning that this attack is untargeted) until the output label of the classifier is incorrect. They chose to formulate the target classifier as a linear affine classifier and yield smaller perturbations than previous attacks. More specifically, they created an attack for binary and multiclass classifiers, with l_∞, l_0 and l_2 measured perturbations. The authors also use a robustness metric ρ for comparison of their results, defined (here for the l_2 attacker) as:

$$\rho_{adv}(f) = \mathbb{E}_x \frac{\Delta(x; f)}{\|x\|_2}$$

for classifier f, input x, expectation over the input space $\mathbb{E}_x$ and minimum perturbation $\Delta(x; f)$.

Trust Region Based attack Moving now towards more advanced ideas that extend the previous optimization based approaches, an attack worth mentioning is the Trust Region Based adversarial attack [Yao+18]. Here the authors aimed at surpassing an important limitation that previous attacks had, namely the fixed iterative step when searching for adversarial points. Instead they used trust regions for approaching the optimal point, since this optimization method computes an adaptive search space (and thus step size) for the next point in each iteration based on some criteria and threshold. They present competitive results tested on multiple target architectures and produce up to 50% smaller perturbations compared to C&W, DeepFool and FGSM attacks.

3.1.2 Black-box attacks

White-box attacks where the starting point of this research area and revealed many weak points of modern DL models, yet they describe a superficially strong adversary. On the other hand, black-box attacks which we will discuss in the next paragraphs assume that models are protected in the simplest sense of only being available as black-box models to the potential attackers. This more realistic approach brought forward another batch of effective attacks that can be characterised as query-based and transfer-based.

Decision Based attack For query-based attacks, attackers leverage the top-N output of the models and run multiple queries in order to gain information and approximate an optimal adversarial example. The decision

boundary attack [BRB18] is implemented accordingly and has this advantage of requiring only the model predictions, uses a relaxed definition of the adversary's objective (e.g. the adversary might be interested in producing incorrect top-5 predictions instead of top-1) and is a lot faster than it's white-box counterparts while producing as much as 50% smaller perturbations. Interested readers can continue on Chapter 4 for more technical details on this attack.

Transfer Based attack As previously mentioned, there exist transfer-based attacks, which in essence use the information of attacks run on surrogate models to attack the target black-box one. In another variant, transfer-based attacks aim at estimating the gradient of the target model, which is typically done with the help of targeted queries, and use it to run white-box attacks on the model. To better illustrate this approach, we will briefly explain the prior-guided random gradient-free method (P-RGF). In this attack, the authors designed a gradient estimation method that takes into account a transfer gradient from a surrogate model trained on the same dataset and derive a theoretically-proven optimal parameter λ for this estimation with fewer queries. In other words, they incorporate a gradient prior to the algorithm, although stating that also data-dependent priors, which utilize the input's structure and subsample the input-space dimensions, can also be introduced for boosting the attack's performance. Their attacks reach 100% success rate for a VGG target model and more than 80% for specific defensive models.

3.1.3 Universal and unrestricted attacks

Lastly, researches played around with the idea of adversarial examples on a more abstract level and built upon universal and unrestricted attacks instead of plain ϵ-bounded and image-targeted attacks.

Universal attacks This type of attack is two-fold, meaning that on the one hand they aim at finding a single perturbation that can attack multiple images, which in mathematical terms is expressed in [Moo+17] by:

$$\mathbb{P}_{x\sim D}\left[C(x+w) \neq C(x)\right] \geq 1-\delta \text{ and } \|w\|_p \leq \xi \tag{3.1}$$

where D denotes the input space distribution and ξ the maximum perturbation magnitude, or they find adversarial examples that can fool multiple models (which also holds in [Moo+17]). The algorithm for producing

these perturbations runs iterates through all data points and for each point finds the minimal perturbation that pushes it towards the class boundary (if $C(x+w) \neq C(x)$ doesn't hold) and aggregates it with the previous perturbation. Since the perturbation is ξ bounded, a projection step (similarly to the PGD method) is needed and all above steps are repeated until 3.1 holds. In mathematical terms:

$$\Delta w_i \leftarrow \arg\min_r \|r\|_2 \text{ s.t. } C(x_i + w + r) \neq C(x_i)$$

for data point x_i, and:

$$w \leftarrow \mathcal{P}_\xi(w + \Delta w_i)$$

Unrestricted attacks In this setting we assume an unrestricted adversary that can produce however-large perturbations. These attacks modify high-level semantic characteristics such as colour or image attributes [HP18] and are resilient to many common defense mechanisms (like adversarial training). In [HP18] a method of only modifying colour saturation and hue is proposed and is based on the representation of RGB images in the HSV space. With the Value in HSV being kept unchanged, they iteratively modify Hue and Saturation to find adversarial examples that have the same meaning to humans. With this simple yet effective attack, they manage to drop an adversarially trained model's accuracy to 8.4%. In [Jos+19] the authors create semantic adversarial examples by experimenting with Generative Adversarial Networks [Goo+14] that can tune image attributes and parametric generative transformations. The generated images, although meaningful, manipulate the attribute space (e.g. by adding glasses or changing the hair color of input faces) in a way that fools a target classifier and drop the classifier's accuracy down to 1%.

3.2 (Non) Robust features learned by CNNs

Many approaches have been explored in order to understand the nature of features that ML models and especially Convolutional Neural Networks (CNNs) learn. As discussed above through these various adversarial attacks, there are many forms of model manipulation that show the complex and unusual interpretations that models learn (for examples, why are many models not invariant to subtle pixel noise or colour shifts?). To this end, a number of researchers have attempted to quantify the information that is used and have indeed found some interesting and unexpected results.

In [Gei+18] the authors found that naturally trained CNNs are heavily biased towards texture features, something that isn't true in the case of human perception and also contradicts the established intuition that CNNs learn increasingly complex shape representations. To induce more robust shape biases instead of textures, they styled images of ImageNet [Rus+15] with random textures using style-transfer methods and hence reducing the generalization ability of textures in training. This reduced the accuracy of models trained on the stylized ImageNet compared to non-stylized images, but shows a greater shape-bias and good generalization to ImageNet.

Authors in [Ily+19] managed to design a theoretical framework to define robust and non-robust features in terms of how they relate to the true label after a small distortion of the input. They experiment with the Basic Iterative Method attack [KGB17] – also known as Projected Gradient Descend (PGD) attack - and construct two datasets, one robust D_R and one non-robust D_{NR} that exhibit only robust and non-robust features respectively. Concisely, D_{NR} contains adversarial examples along with the attack's target (hence wrong) label, meaning that due to the way they are created the correlation of the label to the image is based on non-robust (easily manipulated) features. On the other hand, D_R was created by utilizing a robust (adversarially trained) classifier with mapping g_R from input to the logits layer and constructing a sample x' that approximates:

$$\arg\min_{z\in[0,1]^d} \|g_R(z) - g_R(x)\|$$

with PGD, i.e. by mapping x' to the robust classifier's logits layer as close as possible. Through extensive experiments they showed that a model trained on D_{NR} generalizes well to the initial dataset (although the labels seem wrong to humans) and thus it brings forth the usability of non-robust features for training. Specifically they result in up to 63.3% and 87.9% accuracy for CIFAR-10 [KH+09] and Restricted ImageNet [Rus+15] respectively. They also note that training on D_R yields non-trivial accuracy to the natural dataset of 48.27%.

To enhance these relations of (non) robust features to AAs, [Tsi+19] examine the meaning of adversarial perturbations and feature representations in robust and standard models. They show that representations of robust models align better with human perception and obtain cleaner inter-class interpolations. The authors show this by visualizing the gradients of the loss with respect to input pixels and observing a stronger relation of these features to human perceptual information in adversarially trained models than the

ones from naturally trained models. Creating adversarial examples for robust models reveals another useful fact about the corresponding perturbations, namely that they exhibit salient feature characteristics of the target class. Figures 3.1 and 3.2 showcase both of these facts.

(a) MNIST (b) CIFAR-10 (c) ImageNet

Source: [Tsi+19]

Figure 3.2: Visualization of the loss gradient with respect to input pixels for images from the CIFAR-10 [KH+09], MNIST [Yan+98] and ImageNet [Rus+15] datasets. Note that gradients show which areas in the images mostly influence the model's prediction. From top to bottom, the first row shows the input images, the second row shows gradients for a naturally trained model, and the remaining rows present gradients from adversarially trained models with l_2 and l_∞ adversaries respectively. It is evident that training robust models yields more representative gradients while natural models seem to attract random-looking features.

3.3 Adversarial training and robustness

Adversarial training [GSS15] was first introduced as a regularization (and augmentation) method, where the objective was to minimize:

$$L(\theta; x, y) = \alpha L(\theta; x, y) + (1 - \alpha) L(\theta; x + \text{sign}(\nabla_x L(\theta; x, y)), y)$$

which in other words is a linear combination of natural training error and error introduced by adversarial examples (created with the Fast Gradient Sign Method [GSS15]). In [Mad+19] authors reformulated this definition to a

more generic saddle-point optimization problem, so as to use it to train robust models. Specifically, they formulate the optimization problem as follows:

$$\min_{\theta} \mathbb{E}_{(X,Y)\sim D} \left[\max_{X' \in \mathbb{B}(X,\epsilon)} \mathcal{L}(\theta; X', Y) \right] \tag{3.2}$$

where D denotes the data distribution and $\mathbb{B}(X, \epsilon)$ the ϵ bounded neighborhood of adversarial examples near X, to incorporate strong adversarial examples to the training and minimize the worst-case error that they introduce. This optimization problem presents a more abstract way of computing X', in contradiction to the previous restrictive method.

In [Zha+19] another variant of adversarial training known as TRADES was introduced, that defines robust error as the summation of natural and boundary error (which stems from data points that lie at a maximum distance ϵ from the class boundary). The authors speculate on the trade-off between robustness and accuracy while designing an intuitive objective function for adversarial training that incorporates both natural and boundary errors. We will skip the definitions at this point but readers can find a related section in Chapter 4.

Regarding methods for robust Deep Neural Network (DNN) models for classification, adversarial training is the most popular and effective choice. That said, we briefly want to mention a different state of the art method that yields higher robust accuracy under various attacks, one of which is AutoAttack [CH20], an ensemble of very powerful and controversial attacks. It's intuition is related to TRADES in the sense that both consider the class boundaries as an important factor and a key to better understanding the problem. This method's objective is to control for the intra-class compactness and inter-class diversity of feature representations. This is achieved with the following two objective functions:

$$\min_f \mathbb{E}_{(x,y)} \|f(x) - w_y\|_2^2$$

where C is the number of classes, w_y corresponds to the weight of the y-th node in the classification layer and $f(x)$ is the logit output of the classifier for sample x that belongs to class j, and:

$$\max_{W} \min_{1 \le i < j \le C} \|w_i - w_j\|_2^2$$

where $W^{C \times M}$ holds the weights of the classification layer. It is evident that by optimizing these functions the authors push intra-class samples close to their kernel w_j and push inter-class kernels away from each other.

3.4 Fourier perspective of adversarial examples

In this section we want to bridge the knowledge introduced above with the Fourier perspective of robustness and adversarial attacks discussed in [Yin+20]. The main insight that we get from this work is that models can only be robust to a subset of corruptions that are predominantly defined by the training data. The authors reveal a correlation between frequency component corruptions and test error with different training methods. They provide evidence that AAs tend to corrupt high and mid frequencies when the attacked model uses standard training, however these statistics change with adversarial training and data augmentation methods, where adversarial examples potentially modify high as well as low frequencies. They further corroborate to the idea that by biasing models towards higher/lower frequencies (through data augmentation or input frequency filtering), they become more robust to high/low frequency corruptions, while sacrificing accuracy and vulnerability to other complementary corruptions. We attempt to put more context to these findings through our experiments on the 350 Bird Species dataset [Ger21] and find some interesting alignments. Specifically, the authors of [Yin+20] run experiments to measure the error rate of models to all frequency basis perturbations while also training models on filtered inputs and re-calculating their robustness.

In addition to [Yin+20], in [JB17] one gains a better understanding of CNNs' generalization characteristics through some extensive experiments on datasets constructed with two frequency filtering techniques, a random and radial filter (plus the initial unfiltered dataset). Specifically the authors aim at comparing the generalization gap between models trained on these training sets when tested on their three corresponding test sets. Although none of the methods yield good generalization to all three test sets, the radially filtered train set gives better results in comparison to the randomly and unfiltered train sets.

The authors of [Har+21] and [Lor+21] consider the robustness properties of analysing features and inputs in the frequency domain, in order to detect adversarial attacks. In both defense mechanisms the authors observe intriguing patterns in adversarial examples generated from various algorithms, with respect to their frequency characteristics. Thus they design a binary classifier that effectively learns to distinguish between the frequency spectrum of benign and adversarial images (although they leave out a very important ex-

tension, namely testing the detector on adaptive attacks [ACW18][CW17]). They investigate the detection ability of both frequency magnitude and phase and construct a black-box detector that utilizes the input alone, but also a white-box detector that utilizes the Fourier transformations of a mixture of feature maps from different CNN layers. As expected, the white-box detector achieves significantly better detection rate, in the order of 50% higher than the black-box detector. These attempts where very crucial for our intuition to further analyze the frequency representations of adversarial attacks.

Chapter 4

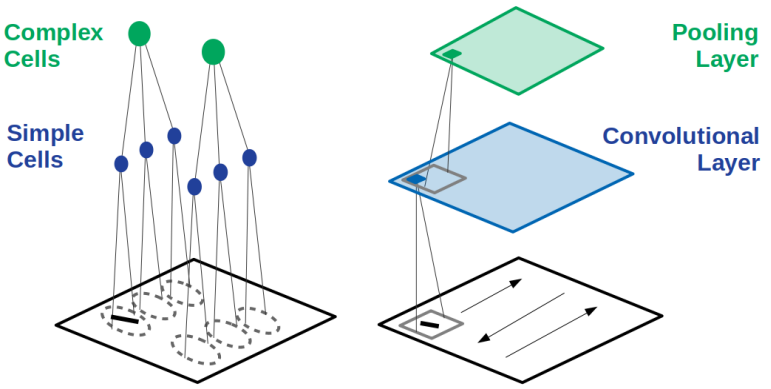
Theoretical background

In this chapter we will analyze the theoretical tools utilized in our method. Specifically, we discuss the fundamental principles of a Convolutional Neural Network (CNN), the mathematical formulation of adversarial attack algorithms as well as the details regarding adversarial training with TRADES, a highly successful method that helps to understand the trade-offs between accuracy and robustness. For a more detailed analysis we refer the reader directly to the corresponding sources.

4.1 Notation

We denote by $x \in \mathbb{R}^N$ the input space and by $F(x)$ or $f(x)$ the output of the full (convolutional) neural network model of parameter ϑ, which is given by applying the softmax function to the logits layer $Z(x)$ (also known as the penultimate layer), i.e. $F(x) = \text{softmax}(Z(x))$.

The classification output is denoted by $C(x) = \arg\max_i \{F(x)\}$ and yields the prediction of the model for input x. By $\mathcal{L}(\vartheta; x, y)$ we represent the multi-class cross-entropy loss of sample x. Lastly, $\tilde{X}$ describes an adversarial example and $\mathbb{B}(X, \epsilon)$ is the ϵ bounded (for any distance norm) neighbourhood around point X.



Source: [Lin21]

Figure 4.1: The relation between visual system components and the basic structure of a convolutional layer.

4.2 Convolutional Neural Networks

Architecture

Convolutional Neural Networks have defined the way images are understood for over a decade, and they show impressive results in tasks such as image classification, object recognition and face recognition. Their effectiveness is thought to be strongly connected to how the biological visual system is structured.

On a high level CNNs are stacked feature-extraction layers, which consist of a convolutional layer, a pooling layer and an activation function. This structure was inspired by [Fuk80] where a neural network is described as layers of S-cells and C-cells on top of each other, the former recognising a specific simple pattern in multiple regions of an image (thus extracting patterns in a spatially invariant way) while the later are connected to the previous S-cells and recognise more complex patterns. In Figure 4.1 this relation becomes more evident.

By construction, CNNs extract information from images by computing the discrete convolution between a filter and the input in an iterative manner. Simply put, a small two-dimensional filter is convolved, i.e. passed over the

entire image one region at a time and multiplied element-wise, with the input which yields values know as the feature map of the input. The convolution of a filter $f^{n\times m}$ with input $I^{N\times M}$ is given by:

$$O(n', m') = \sum_{i=0}^{n-1}\sum_{j=0}^{m-1} f(i,j)I(n'+i, m'+j), 0 \geq n' < N-n, 0 \geq m' < M-m$$

In essence, if the input $X \in \mathbb{R}^{H\times W}$ is convolved with a the filter-kernel $k \in \mathbb{R}^K$ then the output is of shape $\frac{H-K+2P}{S}+1$ where P is the zero padding that we is added to the image and S the stride with which the filter is applied. After this operation, pooling (typically average or max pooling) is employed for down-sampling and for regularization of the feature maps, e.g. a 2×2 max pooling downsamples the image be a factor of two. Lastly, an activation function is applied in order to introduce non-linearities to the way these feature maps are combined for the final result. Depending on the architecture, the number of filters in each convolutional layer, the pooling method and the number of convolution layers can vary. The most widely used activation function is the rectified linear unit (ReLU), which is defined as:

$$\text{ReLU}(x) = \max(0, x)$$

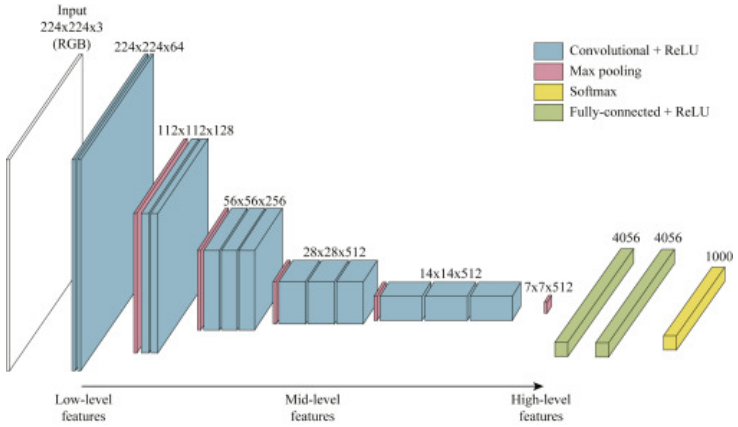
although other functions like the hyperbolic tangent tanh or the sigmoid function can be used.

CNNs for Image Classification

For the purpose of this work, CNNs are used to classify images in a supervised setting, i.e. a labeled training set is given which represents the classes that the model ought to recognise. This type of problem deploys a classification layer on top of the CNN, which essentially takes the learned features and passes them through one or more fully-connected layers with as many output nodes as the distinct classes. The final output is passed through a softmax layer, where the softmax function is applied:

$$\text{softmax}(x) = \frac{e^{Z_i(x)}}{\sum_{j=1}^{M} Z_j(x)}$$

and represents a vector with probabilities of the input belonging to each class. The learning process takes place at the calculation of the loss function and its minimization (a method known as Empirical Risk Minimization



Source: [Pin+20]

Figure 4.2: Simple CNN architecture which takes as input $224 \times 224 \times 3$ RGB images and outputs a probability vector of size 1000.

[Vap91]) through running stochastic gradient descent [HS51] over the network parameters. It is important to note here that the trainable parameters of a CNN are its filters and the weights of the fully-connected layer(s). In other words, the model aims at learning meaningful and representative filters that can describe the features of the training data set. A widely adopted loss function is the cross-entropy loss defined as follows:

$$\mathcal{L}(x, y) = -\sum_{c=1}^{M} y_c \log f_c(x) \tag{4.1}$$

where M is the number of classes, $y_{x,c}$ is an indicator that equals 1 if input x belongs to class c and 0 otherwise, and $f_c(x)$ is the model's softmax output probability of x belonging to class c with $f(x) \in \mathbb{R}^M$.

Residual Networks

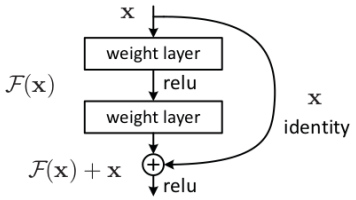


Figure 4.3: A residual block's structure.

In [He+15a] the authors introduced a CNN model that performed better than all previous architectures (e.g. VGG [SZ15]) by using residual connections. The authors observed an important drawback in the existing models which was their inability to represent the identity function. Effectively, models were constrained on the number of

convolutional layers that they could use because of exploding/vanishing gradients [YPP94]. Through residual connections, i.e. connections mapping the input of the convolutional block directly to the output, they overcame this implication and managed to create even deeper and more accurate networks. As seen in Figure 4.3 they add a mapping from the input to the output which can describe the identity mapping if proven optimal. We used this architecture in our experiments with BIRDS and on preliminary experiments with CIFAR-10.

4.3 Attack methods

There are various attack methods one can explore, from black-box attacks (where the attacker has only minimum knowledge of the underlying target model, e.g. only the outputs of the softmax layer are known) to white-box (the attacker has knowledge about the architecture and the weights of the target model). Although we implemented white-box attacks for our experiments – which represent the strongest possible adversary – we want to underline the high success rate of black-box attacks that exist in the literature such as [BRB18], where the authors proposed an attack algorithm solely based on the final decision of the model.

An intuitive way of understanding adversarial examples is by formally defining the optimization problem that they approximately solve, defined in [ND17] as follows:

$$\begin{aligned} &\text{minimize } D(x, x+\delta) \\ &\text{such that } C(x+\delta) \neq C(x) \\ &x+\delta \in [0,1]^n \end{aligned} \tag{4.2}$$

where D is an appropriate distance metric. By solving the above problem we obtain the smallest δ-change that results to an image $x+d$ that is classified differently by the model while still remaining valid. In our setting we use the l_p norm distance metric, which is defined as:

$$D(x, x') = \|x - x'\|_p$$

and is based on the p-norm:

$$\|x\|_p = \left(\sum_{i=0}^{N} |x|^p \right)^{\frac{1}{p}} \tag{4.3}$$

Specifically, our attacks introduce adversaries that are leveraging the euclidean distance measured by the l_2 norm, and the l_∞ norm that measures the maximum change over the N elements in $x - x'$.

For l_∞ as well as l_2 bounded attacks we implemented the Basic Iterative Method extension [KGB17] – also known as Projected Gradient Descent (PGD) attack - of the Fast Gradient Sign Method (FGSM) attack [GSS15]. For l_2 bounded adversaries we use PGD and the Carlini & Wagner attack (C&W) [ND17] that has proven to be one of the most effective white-box attacks. In our experiments we also used the Boundary attack [BRB18] as implemented in foolbox [RBB18], an open-source Python library for adversarial attacks.

Fast Gradient Sign Method (FGSM)

The authors of [GSS15] were motivated by the observation that high dimensional spaces such as the input space of images can translate small perturbations to significant impacts in the output of DNNs (they note that this is a problem of DNNs being too linear). Thus, they add a small perturbation step in the direction that maximizes the loss of the true label (or minimizes the loss of the target label in the case of targeted attacks):

$$\tilde{x} = x + \epsilon \cdot \text{sign}\left(\nabla_x \mathcal{L}\left(\vartheta; x, y\right)\right)$$

Note that this is a single-step attack and as such is not as effective as multiple-step or iterative attacks. The loss function used in both FGSM and PGD attack is the cross-entropy loss as defined in 4.1

Basic Iterative Method (PGD)

As an extension to the previous attack, this attack is more effective since it applies n iterations of η-step FGSM, while after each step it clips the image to lay in the ϵ bounded area around the initial image. In other words the attacker decides on the size of the distortion beforehand and uses a smaller distortion-step in each iteration . Formally defined:

$$\tilde{x}_{i+1} = x + \mathcal{P}_\epsilon(\eta \cdot \text{sign}\left(\nabla_{\tilde{x}_i} \mathcal{L}\left(\vartheta; \tilde{x}_i, y\right)\right))$$

We should mention that for the purposes of our work we wanted to maximize the success rate of this attack -which is far worse than both subsequent

attacks- and found that we had better results when running PGD for a couple more tries (specifically for failed cases). Since our goal was not to prove the attack effectiveness but rather to measure it's impact we consider this as a plausible approach.

Carlini & Wagner attack

In [ND17] the authors tried solving a more explicit optimization problem that follows definition 4.2 but can be solved using existing optimization algorithms. To define an objective function that both minimizes the loss for target class (or maximizes the loss of the true class in non-targeted attacks) but also constrains the output image to be valid (with each pixel lying in $[0, 1]$), they introduced a change of variable as follows:

$$x + \delta \in [0, 1]^n \iff \delta_i = \frac{1}{2}(\tanh w + 1) - x_i$$

The optimization problem now optimizes over parameter and yields a valid image. The authors showed that (amongst others) the objective function defined as:

$$f(\tilde{x}) = \max\{\max\{Z(\tilde{x})_i : i \neq t\} - Z(\tilde{x})_t, -\kappa\}$$

creates the most successful and small perturbations. The final minimization problem C&W attack optimizes is the following:

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

and is optimized by running gradient descent. The constant c is found empirically by running binary search from an initially small value and for a specific number of iterations (typically 10). The intuition behind it is that if for the current value of c no adversarial example is found, then we increase c by a factor of 10 so that the algorithm searches a wider area around the input, i.e. c controls for the relative importance of optimizing the distance $\|\tilde{x} - x\|_2^2$ over the objective function $f(\tilde{x})$.

Boundary attack

The difference between all previous attacks and the Boundary Attack is that the later is a black-box attack whereas the former were white-box attacks.

Along with its high effectiveness, these are the two main reasons why we decided to experiment with this attack.

In principle, the attack starts from an adversarial point and in each iteration samples a perturbation η^k from a proposal distribution P, such that it minimizes the distance between the adversarial point and the input, and yields a new adversarial point. Although the authors note that the meaning of "adversarial" is not restrictive to misclassification, we use it with the same meaning as before. The boundary attack is described in Algorithm 1.

Input: x, $\tilde{x}^0$ s.t. $\tilde{x}^0$ is adversarial
$k \leftarrow 0$;
while $k <$ *total steps* **do**
  sample $\eta^k \sim P(\widetilde{x}^{k-1})$;
  if $\widetilde{x}^{k-1} + \eta^k$ *is adversarial* **then**
    $\widetilde{x}^k \leftarrow \widetilde{x}^{k-1} + \eta^k$;
  else
    $\widetilde{x}^k \leftarrow \widetilde{x}^{k-1}$;
  end
  $k \leftarrow k + 1$;
end
return $\tilde{x}^k$

Algorithm 1: Boundary Attack

4.4 Adversarial training with TRADES

As we already discussed in the previous chapters, adversarial training [GSS15] [Mad+19] yields robust models against various attacks and will be used as a comparative method of finding robust features in our experiments and evaluating the effectiveness of our attacks. To this end we implemented TRADES [Zha+19] as our training method, which encapsulates a natural error R_B induced by the samples that are misclassified and a boundary error R_B, that corresponds to the number of samples that lie no further that distance ϵ from their class boundary, to the robust error R_{Rob} that describes the number of existing adversarial examples, i.e. $R_{Rob}(F) = R_N(F) + R_B(F)$. We skip the mathematical error definitions since they are merely a theoretical tool in [Zha+19], which brings us to the main takeaway that we utilize, namely the objective function of the training task which TRADES minimizes. Simply put TRADES minimizes the value of R_{Rob} which is formulated as:

$$\min_{\theta} \mathbb{E} \left\{ \mathcal{L}(\theta; F(X)Y) + \max_{\tilde{X} \in \mathbb{B}(X,\epsilon)} \mathcal{L}(\theta; F(\tilde{X})F(X)/\lambda) \right\} \tag{4.4}$$

where the hyperparameter λ denotes the relative weight of minimizing the natural versus the boundary error. In the above formulation, adversarial example $\tilde{X}$ is calculated by running the PGD^p attack. It is important to note here that, in contrast to previous adversarial training methods such as [Mad+19], the calculation of the adversarial examples as seen in Algorithm 2 is dependent on the difference between $f(X)$ and $f(\tilde{X})$ instead of the label Y and $f(\tilde{X})$. This models the notion of the boundary error as a tighter approximation than previous methods, since in order to maximize the second loss term the algorithm "picks" samples near the decision boundary of f (as an example, in the case of a binary classifier, the boundary error takes into account samples for which $f(X) \cdot f(\tilde{X}) < 0$). In Algorithm 2 all the training steps of TRADES are presented.

Input: step sizes η_1 and η_2, hyperparameter λ, batch x of size m, number of iterations K, neural network f of parameter θ
initialize $f(\theta)$;
while f *not converged* **do**
  $\tilde{x} \leftarrow x + 0.001 \cdot \mathcal{N}(0, I)$; `/* 𝒩 the standard Gaussian distribution */`
  for $k = 1, ..., K$ **do**
    $\tilde{x} \leftarrow \tilde{x} + \mathcal{P}_\epsilon \left[\eta_1 \cdot \text{sign}\left(\nabla_{\tilde{x}} \mathcal{L}\left(f(\tilde{x}), f(x)\right)\right)\right]$;
  end
  $\theta \leftarrow \theta - \eta_2 \nabla_\theta \left[\mathcal{L}((f_\theta(x), y)) + \mathcal{L}(f_\theta(x), f_\theta(\tilde{x}))/\lambda\right]/m$;
end
return $f(\theta)$

Algorithm 2: TRADES

4.5 Discrete Fourier Transform

To observe the difference between features in the frequency domain we use the widely used Fast Fourier Transform (FFT) algorithm to calculate the 2-dimensional Discrete Fourier Transform. For a discrete signal X with N equidistant samples DFT decomposes it to its constituent frequency components. In the case of a 2-dimensional signal (an image in our case) $X \in \mathbb{R}^{N \times M}$ the DFT is defined as:

$$\mathcal{F}(X)(k,l) = \sum_{n=0}^{N}\sum_{m=0}^{M} X(n,m)\cdot e^{-2\pi i(\frac{nk}{N}+\frac{ml}{M})}$$

In practise the Fourier transform of an image is shifted so that the zero-component is in the center of the image (an example is shown in Figure 5.2). We use the 2-dimensional DFT either to simply observe the characteristics of adversarial perturbations, or as a preliminary direction to filter an image in the frequency domain, by using low, high and band pass filtering to preserve the low, high or intermediate frequencies respectively. Here we used two different types of filter kernels, a Gaussian and a box kernel. We employed a Gaussian kernel for the low pass filtering due to the amount of distortion that the box kernel introduces, as seen in Figure C.2. In the case of high pass filtering though the values of the corresponding frequency components are relatively small, so we simply employed the box kernel. The box kernel with threshold t is described as follows:

$$\mathcal{H}_t(X) = \begin{cases} 1, & \text{if } X \in [-t,t] \\ 0, & \text{else} \end{cases}$$

and the Gaussian kernel as:

$$\mathcal{H}_t(X) = \frac{1}{t\sqrt{2\pi}} e^{-\frac{X^2}{2t^2}}$$

The above formulas describe a low pass filter but one can easily derive the high pass formula simply by exchanging the first case to $X \notin [-t,t]$ for the box kernel, and taking $1-\mathcal{H}_t(X)$ for the Gaussian kernel. The band pass filter simply uses both a low on top of a high pass filter. Since image filtering is essentially a convolution operation on a 2-dimensional space and thus is equivalent to multiplication in the frequency domain, the resulting frequency filtered components are the product of the initial Fourier transform of X and filter $\mathcal{H}(x)$:

$$f(x) * h(x) \leftrightarrow \mathcal{F}(X)\cdot\mathcal{H}(X)$$

and thus:

$$\mathcal{F}(X_H) = \mathcal{F}(X)\cdot\mathcal{H}(X)$$

Chapter 5

Method

After introducing the core concepts of adversarial attacks and adversarially robust models, we now concentrate on implementation and experimental details regarding our attacker, target models and defence. In a nutshell, we developed two popular attacks, namely the PGD and C&W methods and achieved high success rates comparable to the original works. We also experimented with the open source boundary attack implementation contained in the foolbox [RBB18] python library. These attacks are run on common convolutional network architectures trained on the CIFAR-10 and BIRDS dataset. While the former dataset has been used in a variety of methods, we also experimented with the later which contains different resolution and frequency characteristics. The aforementioned topics are discussed in Section 5.1. Despite our attack's success rates on normal training scenarios, we are interested in their performance under defended models which we present in Section 5.2. Finally, we pose the question of which frequency components are modified by different attack and training methods in Section 5.3 and discuss our findings and overall observations in Section 5.4.

Hardware and Software setup

All computations on attacks and model training are done on two 16-core Intel Xeon Central Processing Unit (CPU), accelerated with an NVIDIA Titan RTX Graphics Processing Unit (GPU). Our implementation and code is written in the Python 3.7.11 programming language and the key machine learning tools that we used are PyTorch 1.10.1 and torchvision 0.11.2 [Pas+19]. Since

all our implementations are open-source[1], interested readers can use our code as a reference and reproduce all our experiments.

Datasets and pre-processing

CIFAR-10 The CIFAR-10 [KH+09] dataset contains 60000 images split into 50000 training and 10000 test images (1000 per class) of dimensions 32x32x3 in RGB. Sample images can be seen in Figure 2.3. Images are classified to 10 classes, namely airplane, automobile, bird, cat, deer, dog, frog, horse, ship and truck. This dataset yields useful comparative results since it's a typical target for adversarial attacks in the research community although it certainly does not efficiently represent a real life dataset.

350 Bird species - BIRDS The 350 Bird species [Ger21] Kaggle dataset[2] contains 45980 training, 1575 test (5 per species) and 1575 validation (5 per species) RGB images of shape 224x224x3 in JPG format. They are classified to 315 distinct bird species similar to samples that we present in Figure 2.2. Due to its high resolution but reasonable size and class complexity (with respect to many inter-class similarities) we decided to run experiments on it and observe its characteristics in the frequency space. We will refer to this dataset as BIRDS.

To train our models we use data augmentation and normalization techniques. Specifically, we normalize the images to get mean $\mu = 0$ and a standard deviation $\sigma = 1$ and use padding, random cropping and flipping for augmentation. Note that we don't use augmentation for our attacks. If frequency filtering is applied, it is done before the augmentation and after normalizing the data.

5.1 Attacks on CIFAR-10 and BIRDS

5.1.1 Target models and training

Throughout our experiments and after trying different CNN architectures (such as various ResNet [He+15a] and Wide ResNet [ZK17] models) we fo-

[1]Code for all implementations and experiments is hosted in this github repository: https://github.com/fotinidelig/foolproofNN

[2]We use version 47 which is available here

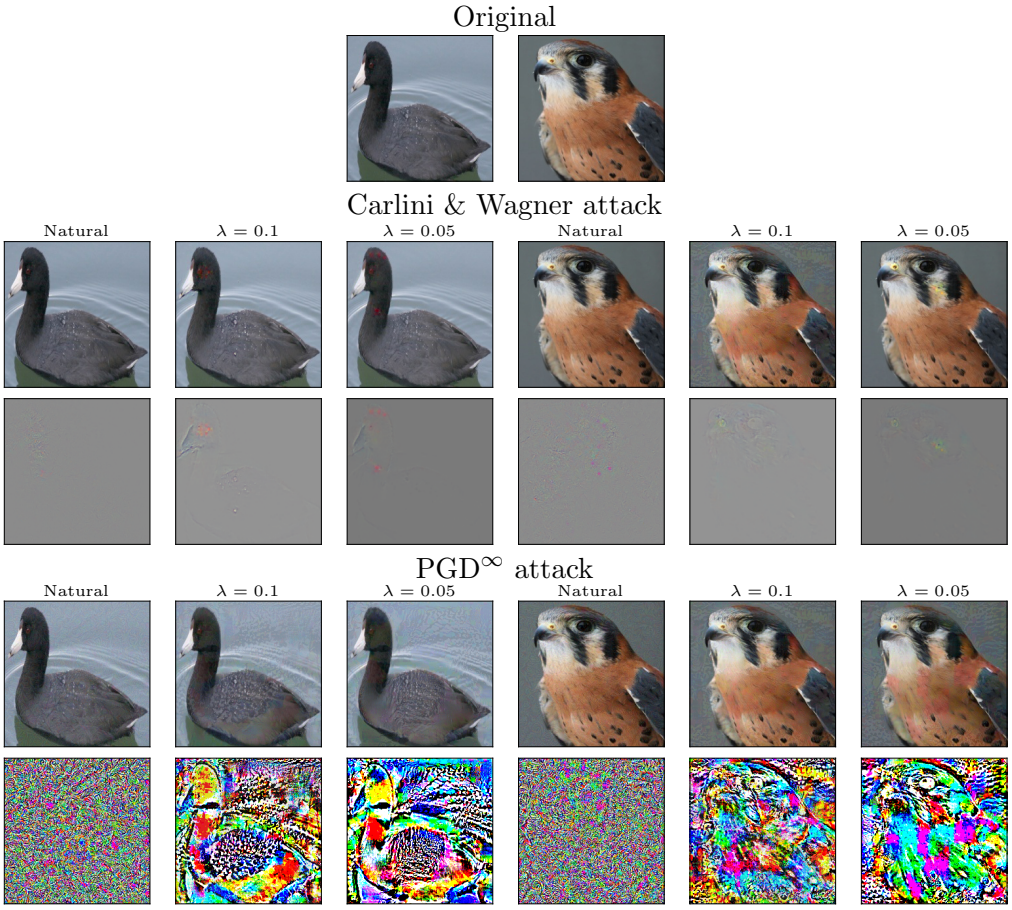


Figure 5.1: Adversarial examples and their corresponding perturbations on two sample images from BIRDS[Ger21]. We performed the PGD^{∞} [KGB17] and C&W [ND17] attacks on the naturally and adversarially trained ResNet34 [He+15a] models ($\lambda = 0.1$ and $\lambda = 0.05$) as seen on each column. The difference in the perturbations for different training methods is clear since for the robust models the image's features are evidently distorted in a meaningful way.

cused on one target model for each dataset that achieved the highest accuracy and training efficiency. To train our models we used Stochastic Gradient Descend (SGD) [HS51] with Nesterov momentum [Nes83] as our optimizer, weight decay and exponential learning rate decay (see Table 5.1 for specific values used in our experiments). Since we use SGD, we feed our data in mini-batches of size 128. Our loss function is the cross-entropy loss defined in 4.1, as is typically used in classification tasks.

For the CIFAR-10 dataset, following [ND17] for evaluating our attacks, we implemented the same CNN architecture (we will refer to it as CWCIFAR10) which is described in Table 5.2. We trained it from scratch and achieved top-1 accuracy of up to 82% (and more than 79%) in 60-70 epochs. Regarding the BIRDS dataset we used a ResNet34 (as well as a GoogleNet [Sze+14a] and EfficientNet_B0 [TL20] which can be found in Appendix B) and used the pretrained[3] weights from ImageNet to initialize our model. We achieve 99.30% and 99.17% top-1 accuracy respectively. With this configuration the model converges in less than 10 training epochs.

Parameter	Value
optimizer	SGD
lr	0.01
lr-decay	0.95
momentum	0.9
batch size	128
weight-decay	5e-4

Table 5.1: Training parameters used throughout our experiments

Layer	Kernel size
Conv + ReLU	$3 \times 3 \times 64$
Conv + ReLU	$3 \times 3 \times 64$
Max Pooling	2×2
Conv + ReLU	$3 \times 3 \times 128$
Conv + ReLU	$3 \times 3 \times 128$
Max Pooling	2×2
Fully Connected + ReLU	256
Fully Connected + ReLU	10
Softmax	10

Table 5.2: CWCIFAR10 model architecture details

5.1.2 Attacker setup

To attack our previous models we used a white-box adversary with the PGD and C&W attack but we also experimented with the black-box boundary

[3]Specifically, we used the implementations for the ResNet34, GoogleNet and EfficientNet_B0 provided by torchvision here.

attack. Based on the theoretical formulation of these methods as discussed Section 4.3, we tested multiple hyperparameters with the prospect of creating strong attacks. Specifically, for the PGD attack and both l_∞ and l_2 bounded adversaries we tuned the number of iterations, the step size η and perturbation size of ϵ ($\epsilon = 0.03$ for l_∞ adversaries, $\epsilon = 0.5, 1.5$ for l CIFAR-10 and BIRDS attacks respectively). For the C&W algorithm we assumed an l_2 adversary we only tested the number of iterations while fixing the learning rate to 0.01 and confidence to 0.01. We used 300 images drawn from the respective test sets for each experiment.

5.1.3 Results

The metric for our evaluation is the fooling/success rate, i.e. the percentage of samples for which an adversarial example is found:

$$\text{SR} = \frac{\sum_{x \in D} \mathbb{1}\left\{\exists \delta \in S \text{ s.t. } C(x+\delta) \neq C(x)\right\}}{N}$$

with S the space of valid perturbations, and the distance $\|x - \tilde{x}\|_{p \in \infty,2}$ of adversarial example $\tilde{x}$ from the benign image x aggregated over all samples. After carefully hand-picking the parameters we achieved near 100% accuracy for all attacks and datasets as seen in tables 5.3, 5.4 and 5.5. We want to note that in practise, some of these attacks where computationally very expensive (with the C&W attack being significantly more time- and resource-consuming) and thus the attack iterations and step sizes should be chosen wisely in order to get feasible results.

Attack	**SR**	l_2 **distance**	**Iterations**	**Step**
∗Boundary	100%	0.260	-	-
C&W^2	100%	0.234	800	-
C&W^2	100%	0.232	400	-
∗C&W^2	100%	0.227	200	-
∗PGD^2	89%	0.491	100	0.1
PGD^2	91.75%	0.494	200	0.1
PGD^2	93.38%	0.490	500	0.1
PGD^2	92.12%	0.489	200	0.07
PGD^2	92.75%	0.494	500	0.07

Table 5.3: Untargeted attack results for the CIFAR-10 dataset with an l_2 adversary

Attack	SR	l_2 distance	l_∞ distance	Iterations	Step
PGD$^\infty$	97.62%	1.169	0.03	30	0.007
$*$PGD$^\infty$	98.12%	1.228	0.03	100	0.007
PGD$^\infty$	95.12%	1.046	0.03	30	0.004
PGD$^\infty$	97.50%	1.250	0.03	100	0.004

Table 5.4: PGD untargeted attack results for the CIFAR-10 dataset with an l_∞ adversary

Attack	SR	l_2 distance	l_∞ distance	Iterations	Step
$*$Boundary	100%	0.601	-	-	-
C&W^2	100%	0.481	-	800	-
C&W^2	100%	0.505	-	400	-
$*$C&W^2	100%	0.487	-	200	-
PGD^2	56%	0.5	-	500	0.1
PGD^2	62.33%	0.5	-	500	0.07
$*$PGD^2	99%	1.5	-	100	0.1
PGD^2	100%	1.5	-	500	0.1
PGD^2	99%	1.5	-	500	0.07
$*$PGD$^\infty$	100%	7.607	0.03	100	0.007
PGD$^\infty$	100%	7.454	0.03	100	0.004

Table 5.5: Untargeted attack results for the BIRDS dataset and ResNet34 model with both l_2 and l_∞ adversaries

5.2 Defending with adversarial training

After ensuring the success of our attacks, we proceed to test a practical defence mechanism by training our models with the TRADES algorithm. In essence, this includes implementing a different loss function than the one used in our previous natural training method, which in each minibatch iteration calculates adversarial examples and pushes the model towards correctly classifying those as well as benign samples.

5.2.1 Experimental setup

Our training configurations mainly stay the same, with the difference that we don't use any data augmentation, although incorporating perturbed images can be though as an augmentation method. Adversarial examples are computed with the PGD algorithm, with the difference that we don't use the cross-entropy loss to compute the gradients but rather the Kullback-Leibler divergence loss defined as:

$$\mathcal{L}(x, \tilde{x}) = \sum_{c=1}^{M} f_c(x) \cdot (\log f_c(x) - \log f_c(\tilde{x}))$$

where x and $\tilde{x}$ are the benign and adversarial samples respectively, M the number of classes, $y \in \mathbb{R}^M$ the one-hot label and $f \in \mathbb{R}^M$ the softmax output probabilities. The choice of loss function was equally inspired by the implementation details[4] provided by the authors of [Zha+19] and the loss function definitions provided by the PyTorch tools[5].

This defence contains a number of tunable parameters for both the training loss computation and the adversarial examples search. In general adversarial training is largely more computationally aggressive than normal training and thus we had to limit our parameter space and mainly experimented with the lambda and norm parameter. Our choice of the parameter λ values stems from the effectiveness of TRADES to defend our models and the maximum desired loss in accuracy. For completeness we have included the parameters used for the inner maximization of problem 4.4 in Appendix B. Lastly, we use the PGD^∞ attack for training but some results on training with the PGD^2 attack can also be found in the Appendix B.

5.2.2 Results

In order to quantify the effect of TRADES over the success rate of our attack, we first run attacks on the non-robustly trained target models and then run the same attacks on our adversarially trained models. We have denoted the exact attack parameters used here in Tables 5.3, 5.4 and 5.5 by an asterisk (*) next to the attack. It is evident from tables 5.6 and 5.7 that the bounded attacks (PGD attacks) significantly under-perform in this setting, while the

[4] An official implementation can be found here

[5] The definitions of the cross-entropy and Kullback-Leibler divergence losses in PyTorch: CrossEntropyLoss and KLDivLoss.

non-bounded attacks (C&W and boundary attacks) compute adversarial examples with significantly larger perturbations. This suggests that perturbations of adversarially trained models are potentially more easily picked up by an observer or an attack detector (empirically we noticed many of them can be observer by the human eye). Moreover, by observing how λ affects both accuracy and robustness, we can spot the trade-off between these two properties and the final choice of λ should reflect their relative importance within a specific task. As a reminder, from problem 4.4 one observes that the significance of minimizing the boundary loss R_B (i.e. the second loss term) increases as λ decreases, which as a consequence trains increasingly robust models.

λ	**Accuracy**	**Attack**	**SR**	l_2 **distance**
(∞)	82.73%	C&W^2	100%	0.227
		Boundary	100%	0.260
		PGD$^\infty$	98.12%	1.228
		PGD^2	89%	0.491
0.1	70.62%	C&W^2	100%	0.874
		Boundary	100%	0.990
		PGD$^\infty$	34.50%	1.41
		PGD^2	23.00%	0.49
1	77.4%	C&W^2	100%	0.549
		Boundary	100%	0.771
		PGD$^\infty$	66%	1.373
		PGD^2	33.50%	0.478
2	78.97%	C&W^2	100%	0.554
		Boundary	100%	0.640
		PGD$^\infty$	71.50%	1.356
		PGD^2	48.50%	0.485
5	78.39%	C&W^2	100%	0.402
		Boundary	100%	0.530
		PGD$^\infty$	84%	1.307
		PGD^2	57.50%	0.474

Table 5.6: TRADES results on the CWCIFAR10 model for different λ values. The trade-off between robustness (observed in the reduced success rate of our attacks as λ decreases) and accuracy is evident.

λ	**Accuracy**	**Attack**	**SR**	l_2 **distance**
(∞)	99.30%	C&W^2	100%	0.481
		Boundary	100%	0.260
		PGD$^\infty$	100%	7.607
		PGD^2	100%	1.5
0.05	98.83%	C&W^2	100%	5.231
		Boundary	99.50%	5.189
		PGD$^\infty$	63.50%	10.079
		PGD^2	6%	1.453
0.1	98.67%	C&W^2	100%	1.812
		Boundary	99.50%	2.594
		PGD$^\infty$	93.50%	9.223
		PGD^2	16.50%	1.49

Table 5.7: TRADES results on the ResNet34 model and BIRDS dataset for different λ values. The λ values that reduced the attack success rate differ where fine-tuned to this specific model.

5.3 Fourier analysis of adversarial examples

We have previously discussed the existence of various attack detection approaches that take into account an image's Fourier transformation and seek to understand whether this image is benign or corrupted [Har+21] [Lor+21]. Many previous works emphasize that adversarial examples are on the most part mid to high frequency distortions and they posses intrinsic frequency properties, different from benign images. With respect to properties of robust and adversarially trained DNNs, it has been proven that they are resilient against high frequency modulations, yet remain vulnerable to low frequency ones [Yin+20]. With our analysis we aim at approaching these two claims, i.e. the general perturbation characteristics that are concentrated to mid-high- frequencies and the apparent sensitivity of robust models to mainly low frequency perturbations.

5.3.1 Analysis method

To measure how a computed perturbation is distributed across frequency components, we first compute its 2D Fourier transform, and then iterate over each component (in the 2D plane a frequency component corresponds

to the (i, j)-pixel value) to calculate the percentage of the total perturbation it represents. In other words,

$$\Delta_{i,j}(\tilde{x}, x) = \frac{|\mathcal{F}_{i,j}(\tilde{x}) - \mathcal{F}_{i,j}(x)|}{\|\mathcal{F}(\tilde{x}) - \mathcal{F}(x)\|_1}, i \in [1, W] \text{ and } j \in [1, H]$$

where $\mathcal{F}(x) \in \mathbb{R}^{H \times W}$ the Fourier transform of x (we only use the real-valued amplitude), aggregated over the image's channels (3 channels for RGB color images).

Since we run our attacks on multiple samples (typically 200 or 300), we take the median:

$$\Delta^N_{i,j} = \text{Med}(\{\Delta_{i,j}(\tilde{x}_k, x_k), x_k \in D^N\})$$

of all benign-adversarial image pairs over the samples set D^N. This yields a 2D perturbation frequency distribution Δ^N which we visualize to observe the differences in our attacks.

5.3.2 Comparison of attacks and training methods in Fourier space

In this section we will present our attack and defence results with respect to the Fourier analysis of the computed perturbations. As a starting point, and in order to be able to assess these findings we looked at the overall frequency characteristics of our datasets. For this purpose we visualized the mean value for each frequency component over all training images in both datasets as seen in Figure 5.2. It is evident that samples obtained from CIFAR-10 have a wider frequency distribution over both low and high components, whereas in the case of the BIRDS dataset the frequency energy is mainly concentrated in the lower frequencies. Figure A.1 shows an attack on the same model architecture trained on both datasets and the connection to the dataset's characteristics is more clear. This serves a first intuition as to why adversarial attacks produce entirely different frequency profiles for our datasets. As seen in Appendix C Figure C.1 where we show different filtering thresholds applied to images from our datasets, lower frequencies mainly describe slow alterations in the images such as color alterations, whilst high frequencies are responsible for the very subtle details.

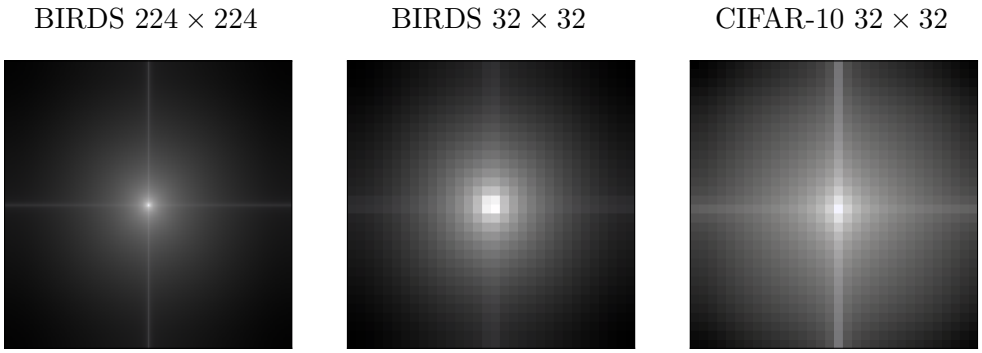


Figure 5.2: Visualization of the mean 2D Fourier transformation amplitudes over all images for both BIRDS and CIFAR-10 datasets. The middle image is the down-sampled BIRDS representation from 224 × 224 to 32 × 32 for better comparison with CIFAR-10.

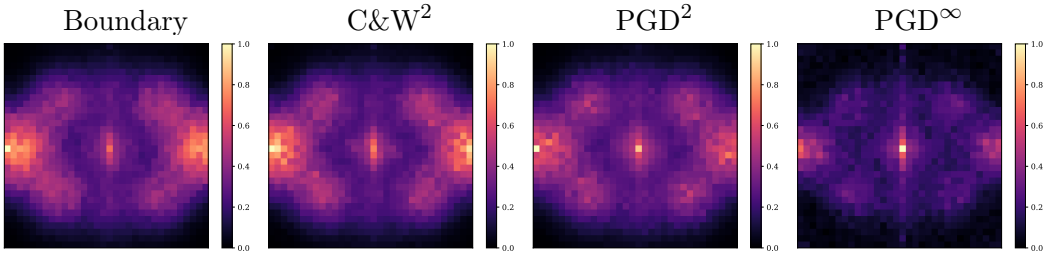


Figure 5.3: Visualization of Δ^N for all attack methods on the CWCIFAR10 model with natural training.

Natural training

CIFAR-10 In Figure 5.3 we present the frequency representations for each attack algorithm for the natural CWCIFAR10 model. These suggest that adversarial examples are manipulating both low and high frequencies but merely as much mid frequencies. This is a contradicting fact to previous beliefs [Yin+20], though it might be an intrinsic characteristic of this specific CNN architecture or our training accuracy. One can also observe that the PGD$^\infty$ attack slightly deviates from the profile of the other l_2 restricted attacks. This is a natural consequence of the fact that the perturbation of each pixel in the case of an PGD$^\infty$ adversary is constant.

BIRDS In the case of the BIRDS dataset, the frequency representations in Figure 5.4 differ significantly. The perturbations are more concentrated

in lower to mid frequencies and amplitudes increase as we move towards the zeroth Fourier component, which is expected due to the energy concentration of BIRDS images in lower components as described in the previous paragraphs. In other words, and according to [Yin+20] the model counts on these low- and mid-frequencies to generalize and thus is more vulnerable to modulations in this region (while CIFAR-10 images have weaker concentration of large amplitude frequencies in a restricted space and thus our model is vulnerable to a wider frequency region).

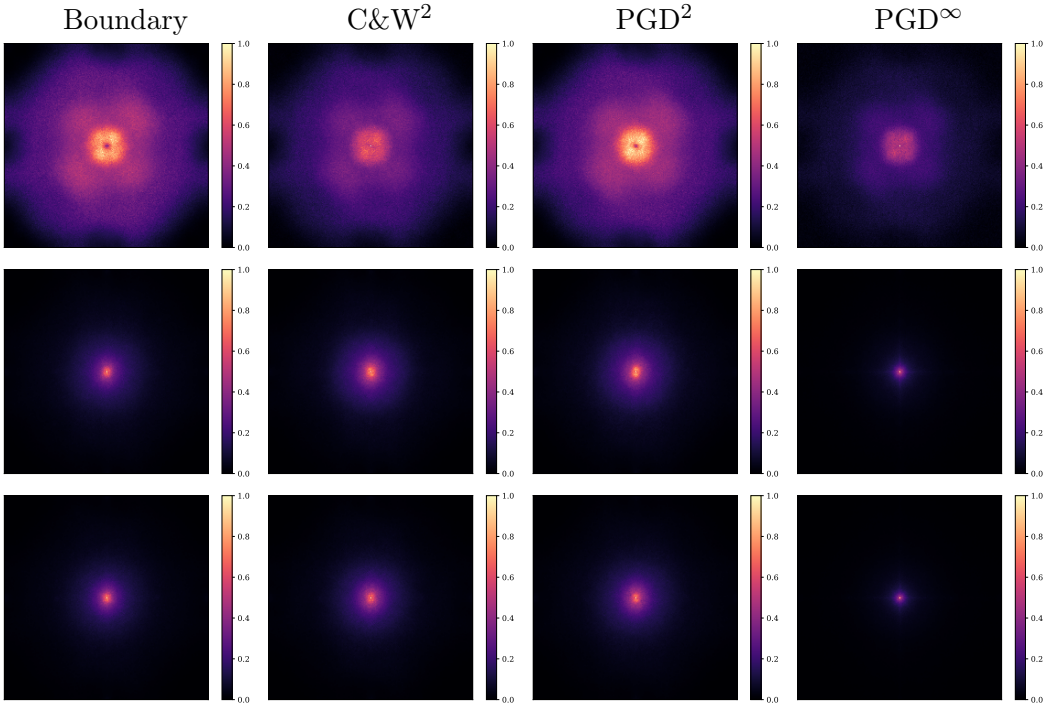


Figure 5.4: Visualization of Δ^N for all attack methods on the ResNet34 model for BIRDS images. The first row represents the results of natural training, the second and third ones of training with TRADES$_{\lambda=0.1}$ TRADES$_{\lambda=0.05}$ respectively.

Adversarial training

In general and throughout all of our experiments we found that adversarial training yields models with significantly altered Δ^N distributions. Specifically it showed a clear improvement in the robustness of models to high frequency pixel manipulations, with increasing robustness leading to increasingly sustained perturbations towards lower frequencies.

CIFAR-10 Although in the previous paragraphs we emphasized that adversarial attacks for CIFAR-10 images are distributed over a large frequency region (shown in Figure 5.3), this region is entirely shifted towards low frequencies as seen in Figure B.1 in Appendix B. This occurs with all adversaries and parameter combinations that we run.

BIRDS In the case of images from BIRDS, adversarial attacks manipulated a large low- to mid-frequency neighbourhood before. We observe a decrease in this region when models are trained robustly, with the highest perturbation amplitudes being gathered in much fewer and lower Fourier components.

5.4 Discussion

In the previous section we presented two claims suggested from related works, specifically proposing that adversarial attacks corrupt mid to high frequencies and that adversarial training boosts model's robustness against higher frequency modifications. We found the former to be deeply tied to the dataset (and to some extend model) characteristics and the latter to be reproducible by our experiments.

After extensively testing our attacks and evaluating the effects of adversarial training, we want to discuss some catholic observations with respect to different attack methods as well as model generalization properties. So far we have seen few differences between different attack's perturbations, with the most notable being the effect of the utilized norm of an attacker (e.g. a PGD^2 vs PGD^∞ adversary). We suspect that the key characteristic that defines perturbations is the dataset itself and as a consequence its learnable features. This has been previously suggested with the appearance of universal attacks [Moo+17] [Ily+19]. Some preliminary results we have on different model architectures for test images of BIRDS and CIFAR-10 (Figure A.2 and A.1) also suggest that some architectures represent models with different frequency vulnerabilities. Furthermore, the intuition that classes that appear close to humans (e.g. truck and car, cat and dog) are also closer in the feature manifold of models also seems very plausible (as seen in Figure 5.5).

Regarding the separation between robust and non-robust features, a common baseline is that adversarial attacks mainly modify non-robust features [Ily+19]. We find that it's hard to distinguish them in the frequency domain since attacks corrupt a large frequency region, still robust models experience corruptions mainly in the lower Fourier components. The assumption that

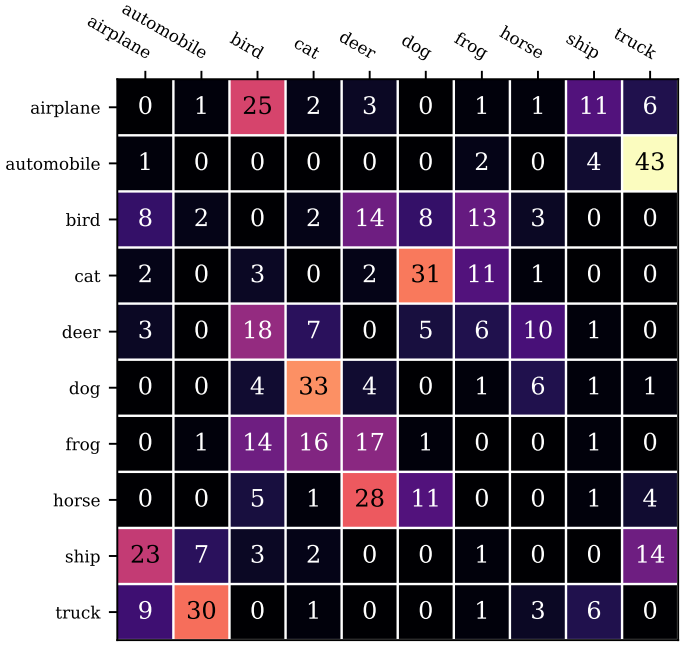


Figure 5.5: Here we run untargeted C&W attacks on 50 CIFAR-10 samples belonging in each class (a total of 500 samples) and present the number of adversarial samples from each true class that were classified falsely in different classes. The vertical axis represents the true class whereas the horizontal states the classification class of the perturbed images. In almost all classes the class with the most adversarial examples would be perceived as a relatively similar class to the true one by humans (e.g. 60% of cat images where modified to be classified as dogs).

lower frequencies are related to robust features is not too far-fetched. But rather than focusing on this distinction, we want to raise the same idea as discussed in [Yin+20], namely that models can only be resilient to a subset of corruptions that are on the most part introduced in the training augmentation. Adversarial training, i.e. data augmentation with corrupted images in low as well as higher frequencies thus yield models robust to these Fourier components.

Chapter 6

Conclusion and future directions

In this thesis we investigated the threat of adversarial attacks in image classification Convolutional Neural Network (CNN) models. We experimented with the CIFAR-10 and 350 Birds Species datasets with a variety of victim models and attack methods. Many questions remain unresolved regarding the ability of models to become robust to such corruptions and the characteristics that these attacks utilize, with respect to the dataset, model architecture and training (i.e. either natural-standard or adversarial) properties.

6.1 Conclusion

In summary, we demonstrated the effectiveness of adversarial attacks by developing two widely used white-box attacks, namely the C&W [ND17] and PGD [KGB17] attacks and experimented with their parameter space to achieve up to a 100% fooling rate. These attacks produced very subtle corruptions, dependent on the attack configurations and the dataset. Since white-box attacks are on the one hand the optimal way to measure robustness, but on the other hand pose a non-realistic adversary, we also compared them with the black-box Boundary attack and found that it is a strong and effective of an attack. Interestingly, after implementing the TRADES adversarial training algorithm [Zha+19] we clearly observed a lower fooling rate in all attacks, as well as a trade-off between the classification accuracy and the ability of models to become robust to adversarial examples.

Furthermore, since it has been previously suggested that adversarial examples contain fundamental differences in their frequency space profile in

comparison to benign images, we aimed at exploring these effects and tendencies by computing the Fourier transform of attack perturbations, denoted as Δ^N. By visualizing Δ^N (across N adversarial examples) we found that corruptions are sensitive to the l_p distance metric chosen by the adversary, the dataset's frequency properties, as well as the target model architecture. Although our initial goal was to work towards clarifying the debate between robust and non-robust features of models, we found this to be a delicate property not entirely observable in the frequency domain that needs even more extensive research and experimenting. Nevertheless the differences in the distribution of perturbations across the frequency domain is an encouraging observation and justifies the success of frequency-related detection mechanisms such as [CH20] and [Har+21].

6.2 Future work

The area of adversarial machine learning has made impressive progress in recent years, nevertheless some questions remain open which we want to discuss with respect to our future research intentions.

Since adversarial training yields the most effective defense method, we propose that TRADES as well as other similar methods should utilize even stronger (and more efficient) attacks than the PGD attack (e.g. the boundary attack) in order to increase their robustness towards a wider variety of attacks. This topic would need both theoretical and experimental proofs since so far only the PGD method has been proven to solve the saddle point problem of equation 3.2 effectively [Mad+19]. This comes hand-in-hand with creating robust models in other machine learning tasks beyond image classification, such as self-supervised learning and Natural Language Processing models.

We are also interested in investigating the effect of training models on filtered images, with respect to their generalization ability as well as robustness properties, with the prospect of further manipulating the features learned by CNNs. This could lead to architectural changes of common ML and DL models in order to boost their ability to learn semantically meaningful features. Furthermore, accuracy as the benchmarking metric and optimization goal of models should be reconsidered, since it doesn't capture the quality of the learned data representations or its robustness (and hence we experience the trade-off between robustness and accuracy). Research also shows that bridging the gap between human and machine vision can reveal inter-

esting properties on both ends that can lead towards better interpretable AI systems. As an example, authors in [Pet+19] proved that by using human (and thus noisy) labels on CIFAR-10 images rather than the dataset labels classifiers are trained in a robust manner, while [Lan+21] proved that classical CNN architectures perform better in an alignment to human attention than modern attention-based models (e.g. [Fuk+19]). Thus we believe that utilizing such experimental methods or even a human visual understanding prior (which was also proposed in [Ily+19]) in computer vision models is a fruitful future direction.

On a final note, we are very positive about the increasingly popular shift of the AI community towards more safe and reliable tools and hope to systematically incorporate reliability and robustness against diverse attacks to the AI development pipeline.

Bibliography

[ACW18] Anish Athalye, Nicholas Carlini, and David Wagner. *Obfuscated Gradients Give a False Sense of Security: Circumventing Defenses to Adversarial Examples*. 2018. arXiv: 1802.00420 [cs.LG].

[Akh+21] Naveed Akhtar, Ajmal Mian, Navid Kardan, and Mubarak Shah. *Advances in adversarial attacks and defenses in computer vision: A survey*. 2021. arXiv: 2108.00401 [cs.CV].

[Bai+21] Tao Bai, Jinqi Luo, Jun Zhao, Bihan Wen, and Qian Wang. *Recent Advances in Adversarial Training for Adversarial Robustness*. 2021. arXiv: 2102.01356 [cs.LG].

[BRB18] Wieland Brendel, Jonas Rauber, and Matthias Bethge. *Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models*. 2018. arXiv: 1712.04248 [stat.ML].

[CH20] Francesco Croce and Matthias Hein. *Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks*. 2020. arXiv: 2003.01690 [cs.LG].

[Che+20a] Dingfan Chen, Ning Yu, Yang Zhang, and Mario Fritz. "GAN-Leaks: A Taxonomy of Membership Inference Attacks against Generative Models". In: *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security* (2020). DOI: 10.1145/3372297.3417238.

[Che+20b] Shuyu Cheng, Yinpeng Dong, Tianyu Pang, Hang Su, and Jun Zhu. *Improving Black-box Adversarial Attacks with a Transfer-based Prior*. 2020. arXiv: 1906.06919 [cs.LG].

[Cub+19] Ekin D. Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V. Le. *AutoAugment: Learning Augmentation Policies from Data*. 2019. arXiv: 1805.09501 [cs.CV].

[CW17] Nicholas Carlini and David Wagner. *Adversarial Examples Are Not Easily Detected: Bypassing Ten Detection Methods*. 2017. arXiv: 1705.07263 [cs.LG].

[Dav+16] Silver David, Huang Aja, Maddison Chris, and Arthur Guez. "Mastering the game of Go with deep neural networks and tree search". In: *Nature* 529 (2016), pp. 484–489. DOI: 10.1038/nature16961.

[Dha+20] Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. *Jukebox: A Generative Model for Music*. 2020. arXiv: 2005.00341 [eess.AS].

[Fuk+19] Hiroshi Fukui, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. *Attention Branch Network: Learning of Attention Mechanism for Visual Explanation*. 2019. arXiv: 1812.10025 [cs.CV].

[Fuk80] Kunihiko Fukushima. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". In: *iological Cybernetics volume* 36 (1980), pp. 192–202. DOI: 10.1007/BF00344251.

[Gei+18] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. "ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness". In: *CoRR* (2018). arXiv: 1811.12231 [cs.CV].

[Ger21] Gerry. *325 bird species - classification*. 2021. URL: https://www.kaggle.com/gpiosenka/100-bird-species.

[Goo+14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative Adversarial Networks*. 2014. arXiv: 1406.2661 [stat.ML].

[GSS15] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. *Explaining and Harnessing Adversarial Examples*. 2015. arXiv: 1412.6572 [stat.ML].

[Har+21] Paula Harder, Franz-Josef Pfreundt, Margret Keuper, and Janis Keuper. *SpectralDefense: Detecting Adversarial Attacks on CNNs in the Fourier Domain*. 2021. arXiv: 2103.03000 [cs.CV].

[He+15a] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. *Deep Residual Learning for Image Recognition*. 2015. arXiv: 1512.03385 [cs.CV].

[He+15b] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification". In: *CoRR* abs/1502.01852 (2015). arXiv: 1502.01852.

[HP18] Hossein Hosseini and Radha Poovendran. *Semantic Adversarial Examples*. 2018. arXiv: 1804.00499 [cs.CV].

[HS51] Robbins Herbert and Monro Sutton. "A stochastic approximation method". In: *The annals of mathematical statistics* (1951), pp. 400–407.

[Ily+19] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. *Adversarial Examples Are Not Bugs, They Are Features*. 2019. arXiv: 1905.02175 [stat.ML].

[JB17] Jason Jo and Yoshua Bengio. *Measuring the tendency of CNNs to Learn Surface Statistical Regularities*. 2017. arXiv: 1711.11561 [cs.LG].

[Jos+19] Ameya Joshi, Amitangshu Mukherjee, Soumik Sarkar, and Chinmay Hegde. *Semantic Adversarial Attacks: Parametric Transformations That Fool Deep Classifiers*. 2019. arXiv: 1904.08489 [cs.CV].

[Jum+21] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A. A. Kohl, Andrew J. Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler1, Trevor Back1, Stig Petersen1, David Reiman1, Ellen Clancy1, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W. Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. "Highly accurate protein structure prediction with AlphaFold". In: *Nature* 596 (2021), pp. 583–589. DOI: 10.1038/s41586-021-03819-2.

[KGB17] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. *Adversarial examples in the physical world*. 2017. arXiv: 1607.02533 [cs.CV].

[KH+09] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images". In: (2009).

[Lan+21] Thomas A. Langlois, H. Charles Zhao, Erin Grant, Ishita Dasgupta, Thomas L. Griffiths, and Nori Jacoby. *Passive Attention in Artificial Neural Networks Predicts Human Visual Selectivity*. 2021. arXiv: 2107.07013 [cs.CV].

[Lin21] Grace W. Lindsay. "Convolutional Neural Networks as a Model of the Visual System: Past, Present, and Future". In: *Journal of cognitive neuroscience* 33.10 (2021), pp. 2017–2031. DOI: 10.1162/jocn_a_01544.

[Lor+21] Peter Lorenz, Paula Harder, Dominik Strassel, Margret Keuper, and Janis Keuper. *Detecting AutoAttack Perturbations in the Frequency Domain*. 2021. arXiv: 2111.08785 [cs.CV].

[Mad+19] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. *Towards Deep Learning Models Resistant to Adversarial Attacks*. 2019. arXiv: 1706.06083 [stat.ML].

[MFF16] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. *DeepFool: a simple and accurate method to fool deep neural networks*. 2016. arXiv: 1511.04599 [cs.LG].

[Moo+17] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. *Universal adversarial perturbations*. 2017. arXiv: 1610.08401 [cs.CV].

[ND17] Carlini Nicholas and Wagner David. "Towards Evaluating the Robustness of Neural Networks". In: *2017 IEEE Symposium on Security and Privacy (SP)*. 2017, pp. 39–57. DOI: 10.1109/SP.2017.49.

[Nes83] Yurii E Nesterov. "A method for solving the convex programming problem with convergence rate O (1/kˆ 2)". In: *Dokl. akad. nauk Sssr*. Vol. 269. 1983, pp. 543–547.

[Pas+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[Pet+19] Joshua C. Peterson, Ruairidh M. Battleday, Thomas L. Griffiths, and Olga Russakovsky. *Human uncertainty makes classification more robust*. 2019. arXiv: 1908.07086 [cs.CV].

[Pin+20] Walter Hugo Lopez Pinaya, Sandra Vieira, Rafael Garcia-Dias, and Andrea Mechelli. "Chapter 10 - Convolutional neural networks". In: (2020). Ed. by Andrea Mechelli and Sandra Vieira, pp. 173–191. DOI: https://doi.org/10.1016/B978-0-12-815739-8.00010-9.

[RBB18] Jonas Rauber, Wieland Brendel, and Matthias Bethge. *Foolbox: A Python toolbox to benchmark the robustness of machine learning models*. 2018. arXiv: 1707.04131 [cs.LG].

[Rus+15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. *ImageNet Large Scale Visual Recognition Challenge*. 2015. arXiv: 1409.0575 [cs.CV].

[RWK20] Leslie Rice, Eric Wong, and J. Zico Kolter. *Overfitting in adversarially robust deep learning*. 2020. arXiv: 2002.11569 [cs.LG].

[SCJ19] Octavian Suciu, Scott E. Coull, and Jeffrey Johns. *Exploring Adversarial Examples in Malware Detection*. 2019. arXiv: 1810.08280 [cs.LG].

[SZ15] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image Recognition*. 2015. arXiv: 1409.1556 [cs.CV].

[Sze+14a] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. *Going Deeper with Convolutions*. 2014. arXiv: 1409.4842 [cs.CV].

[Sze+14b] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. *Intriguing properties of neural networks*. 2014. arXiv: 1312.6199 [cs.CV].

[TL20] Mingxing Tan and Quoc V. Le. *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. 2020. arXiv: 1905.11946 [cs.LG].

[Tra+19] Florian Tramèr, Pascal Dupré, Gili Rusak, Giancarlo Pellegrino, and Dan Boneh. "AdVersarial: Perceptual Ad Blocking meets Adversarial Machine Learning". In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security* (Nov. 2019). DOI: 10.1145/3319535.3354222. URL: http://dx.doi.org/10.1145/3319535.3354222.

[Tsi+19] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. *Robustness May Be at Odds with Accuracy*. 2019. arXiv: 1805.12152 [stat.ML].

[Vap91] V. Vapnik. "Principles of Risk Minimization for Learning Theory". In: *Proceedings of the 4th International Conference on Neural Information Processing Systems*. Morgan Kaufmann Publishers Inc., 1991, pp. 831–838. ISBN: 1558602224. URL: papers.nips.cc/paper/506-principles-of-risk-minimization-for-learning-theory.pdf.

[WX18] Rey Wiyatno and Anqi Xu. *Maximal Jacobian-based Saliency Map Attack*. 2018. arXiv: 1808.07945 [cs.LG].

[XLY21] Cong Xu, Xiang Li, and Min Yang. *An Orthogonal Classifier for Improving the Adversarial Robustness of Neural Networks*. 2021. arXiv: 2105.09109 [cs.CV].

[Yan+98] Lecun Yann, Bottou Leon, Bengio Yoshua, and Haffner Patrick. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: 10.1109/5.726791.

[Yao+18] Zhewei Yao, Amir Gholami, Peng Xu, Kurt Keutzer, and Michael Mahoney. *Trust Region Based Adversarial Attack on Neural Networks*. 2018. arXiv: 1812.06371 [cs.LG].

[Yin+20] Dong Yin, Raphael Gontijo Lopes, Jonathon Shlens, Ekin D. Cubuk, and Justin Gilmer. *A Fourier Perspective on Model Robustness in Computer Vision*. 2020. arXiv: 1906.08988 [cs.LG].

[YPP94] Bengio Yoshua, Simard Patrice, and Frasconi Paolo. "Learning long-term dependencies with gradient descent is difficult". In: *IEEE Transactions on Neural Networks* 5.2 (1994), pp. 157–166. DOI: 10.1109/72.279181.

[Yun+20] Liu Yuntao, Mondal Ankit, Chakraborty Abhishek, Zuzak Michael, Jacobsen Nina, Xing Daniel, and Srivastava Ankur. "A Survey on Neural Trojans". In: *2020 21st International Symposium on Quality Electronic Design (ISQED)*. 2020, pp. 33–39. DOI: 10.1109/ISQED48828.2020.9137011.

[YWY18] Dingyi You, Haiyan Wang, and Kaiming Yang. "State-of-the-art and trends of autonomous driving technology". In: *2018 IEEE International Symposium on Innovation and Entrepreneurship (TEMS-ISIE)*. 2018, pp. 1–8. DOI: 10.1109/TEMS-ISIE.2018.8478449.

[Zha+19] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. *Theoretically Principled Trade-off between Robustness and Accuracy*. 2019. arXiv: 1901.08573 [cs.LG].

[ZK17] Sergey Zagoruyko and Nikos Komodakis. *Wide Residual Networks*. 2017. arXiv: 1605.07146 [cs.CV].

[ZL21] Nikola Zubić and Pietro Liò. *An Effective Loss Function for Generating 3D Models from Single 2D Image without Rendering*. 2021. arXiv: 2103.03390 [cs.CV].

Appendix A

Additional attacks run on CIFAR-10 images

So far we have shown our attack results when running untargeted attacks, which as we discussed previously seek to find adversarial examples that move away from the true class and towards the closest classes in a bounded neighbourhood. We also experimented with targeted attacks where the adversary defines what the output class for the adversarial example should be. We run our attacks with 200 samples and targeting all classes for each sample. As expected this setup is difficult in the case of a large number of classes so we present results only for the CIFAR-10 dataset in tables A.1 and A.2.

Attack	**SR**	**l_2 distance**	**Iterations**	**Step**
C&W^2	99%	0.41	800	-
C&W^2	99%	0.396	400	-
C&W^2	100%	0.227	200	-
PGD^2	69.40%	0.494	200	0.1
PGD^2	68.90%	0.49	500	0.1
PGD^2	67%	0.489	200	0.07
PGD^2	66.40%	0.494	500	0.07

Table A.1: Targeted attack results for the CIFAR-10 data set with an l_2 adversary

Attack	SR	l_2 distance	l_∞ distance	Iterations	Step
PGD$^\infty$	96.30%	1.1.45	0.03	30	0.007
PGD$^\infty$	97%	1.47	0.03	100	0.007
PGD$^\infty$	97.70%	1.45	0.03	30	0.004
PGD$^\infty$	97%	1.49	0.03	100	0.004

Table A.2: PGD targeted attack results for the CIFAR-10 data set with an l_∞ adversary

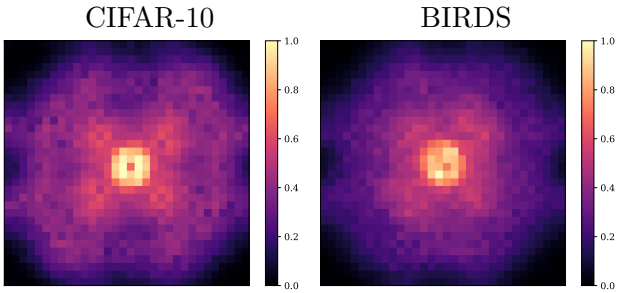


Figure A.1: Visualization of Δ^N for the boundary attack on the ResNet34 architecture trained on CIFAR-10 and BIRDS in 32x32 resolution (both cases achieving 99.50% SR on 200 test). We observe the model's but also the dataset's "fingerprints" in that the distortions have similar effects but span on different frequency components.

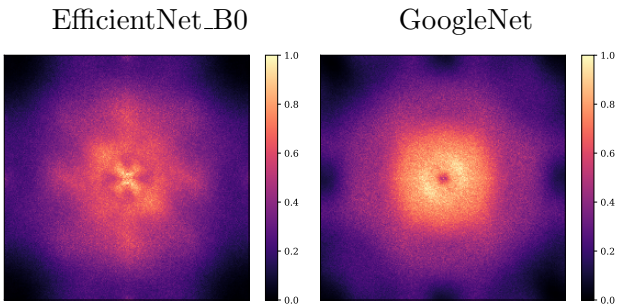


Figure A.2: Visualization of Δ^N for the boundary attack on EfficientNet_B0 and GoogleNet (with 100% SR on 200 test samples), which we only trained normally with 99.11% and 99.17% accuracy respectively. They exhibit differences with respect to their vulnerabilities, meaning the distortion distribution in different Fourier.

Appendix B

Adversarial training setup and frequency analysis results

In Section 5.2 we described the general adversarial training setup and results without discussing the parameters used for maximizing the second loss term in problem 4.4. In Table B.1 we present the inner PGD attack parameters that were used. In the case of the EfficientNet_B0 architecture training a robust model was significantly more difficult and with many parameter combinations we experienced a drop in accuracy on one hand, without a drop in the success rate of attacks on the other. Thus we didn't incorporate it in our final experiments. It is worth mentioning that this could be connected to the observation that adversarial robust generalization is far worse that standard generalization as discussed in [RWK20]

Model	**Iterations**	η_1	ϵ	**Norm**	η_2	**Epochs**
CWCIFAR10	30	0.007	0.03	l_∞	0.01	45
ResNet34	40	0.007	0.02	l_∞	0.01	8

Table B.1: Parameters chosen for training our models with TRADES, specifically the parameters of the inner perturbation calculation for each training sample

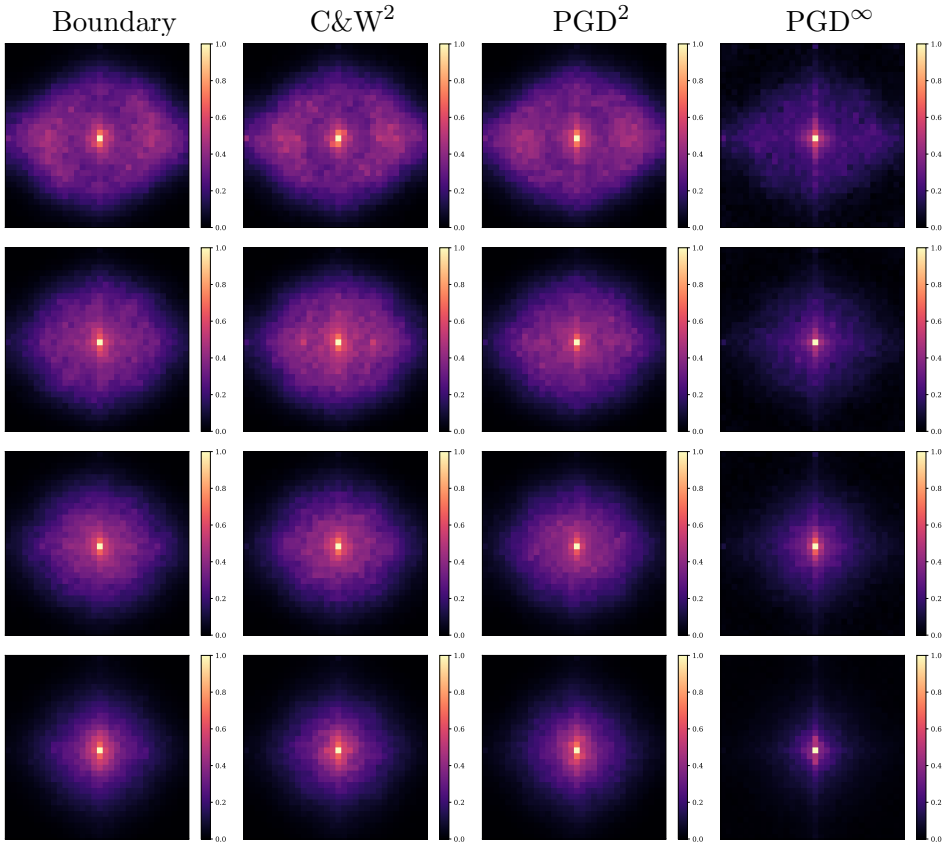


Figure B.1: Visualization of Δ^N for all attack methods on the CWCIFAR10 model when training with TRADES and λ values of (from top row to bottom) 5, 2, 1 and 0.1.

Appendix C

Image filtering in the frequency domain

In this section of the appendix we apply a range of low, high and band pass filtering with different thresholds on a sample image from both datasets. The filtering method is described in Section 4.5 and as discussed, the difference between a Gaussian and a box filter is quite clear. This visualization helps to understand the features that our models will generalize over and will be most vulnerable to, as well as what it means to modify specific frequency components. For example, since BIRDS images have high amplitudes and information gathered in the lower frequencies (indicating that color properties play a significant role) it is safe to assume that a model learns their correlation to different classes and as such is vulnerable to their manipulation. This fact is of course noticeable in our perturbation frequency visualizations. Our experiments with different filtering methods and thresholds can be seen as an introduction to exploring training CNNs on filtered images and possibly gaining more knowledge as to what types of features they learn and are vulnerable to.

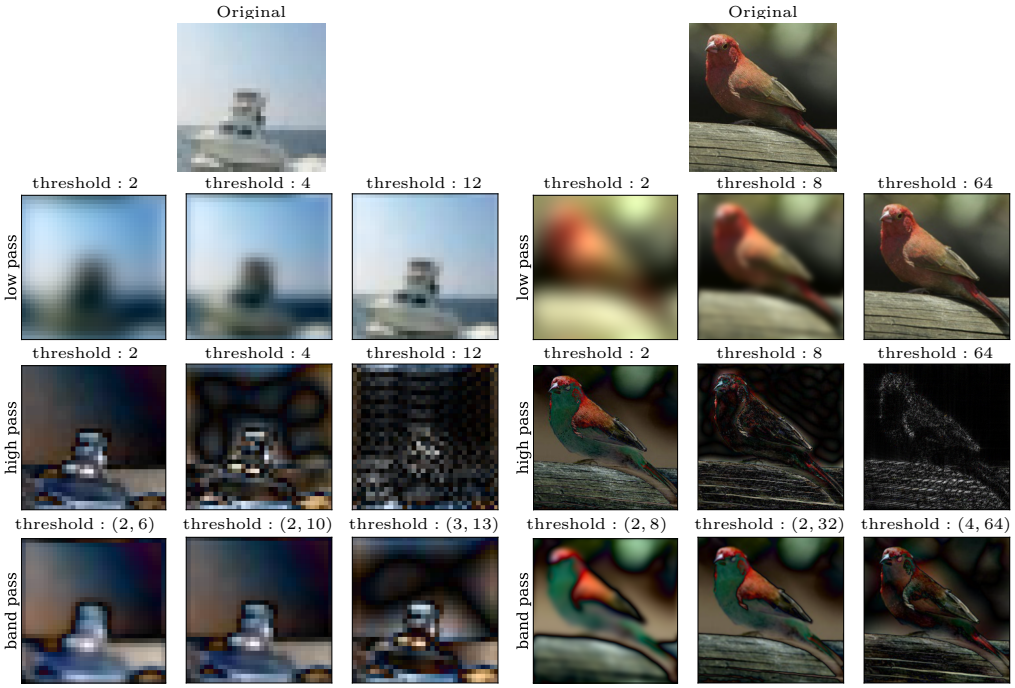


Figure C.1: Low (Gaussian), high and band pass filtering applied on a CIFAR-10 (left) and a BIRDS (right) sample image.

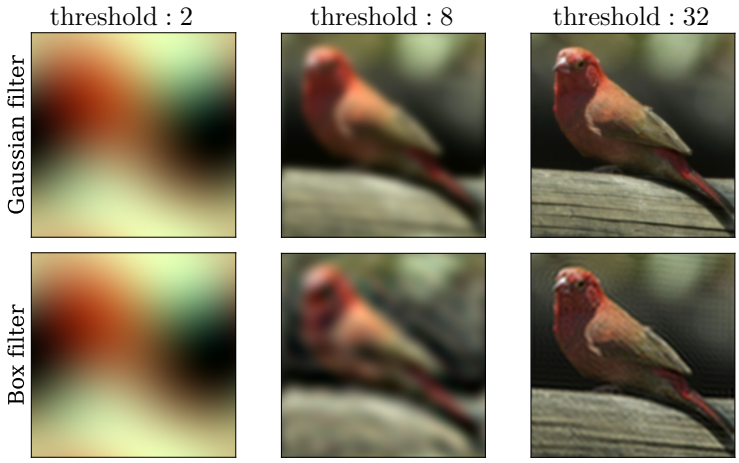


Figure C.2: A comparison between a low pass Gaussian and box filter. The quality of the reconstruction after applying Gaussian filtering in much superior and doesn't produce perceivable artifacts.