Deep Protection

Universal Defense against Adversarial Examples

Sourabh Kulkarni, Pradeep Ambati

Overview

- Deep learning models have been widely successful and are being employed in several computer vision (CV) applications.
- This widespread use has left the CV application spaces open to **adversarial attacks** – images that have been slightly perturbed to cause misclassification.
- There exists several attacks types, and defenses against them are attack specific and hence not very practical in real world.
- We re-imagine adversarial defense problem as an image restoration problem - restore non-adversarial prior from the adversarial image.
- We use **Deep Image Prior (DIP)** – an image restoration technique, and modify it to protect against adversarial images
- We successfully demonstrate protection of deep networks (**Deep Protection)** against 8 different adversarial attack types and gain several insights in the process.

Background

• **DIP**: Uses structure of a generative deep convnet to regenerate target image from noise. Applications include denoising, in-filling etc.

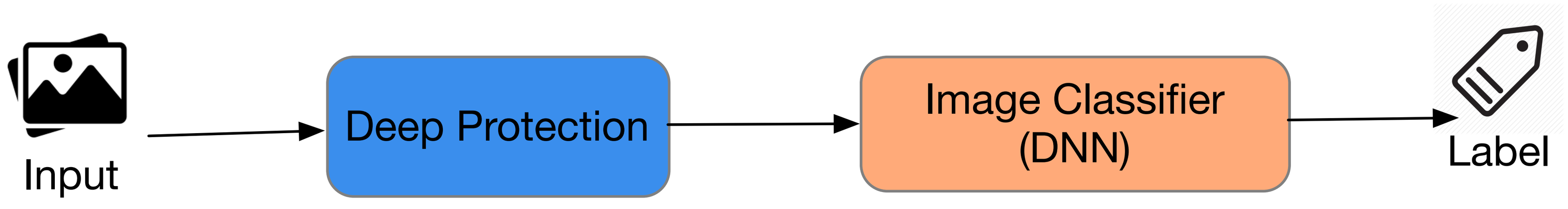
• **Adversarial Attacks**: Various techniques that slightly perturb image in pixel space to cause departure in feature/class space of a target deep network model. Several approaches exist, but basic premise stays the same.

• **Defense Strategies**: Existing strategies usually target only one attack type, e.g., training models with adversarial samples, filtering images, rotations and other linear operations.

Approach

• **Key Idea**: Modify *DIP* and use it to obtain the non-adversarial prior of an adversarial image. (Treat adversarial image as a corrupted version of the original and run restoration on it)

• **Deep Protection Process**:
- ➢ Initialization: random weights, random inputs
- ➢ Loss: *L2 MSE* with target image
- ➢ Hyperparamter tuning: chosen config – *ADAM* optimizer and *swish* activation
- ➢ Termination: Automatic (Image specific, complex) and Static (Fixed iteration, simple)
- ➢ Classification: Off-the-shelf classifier (ResNet18)

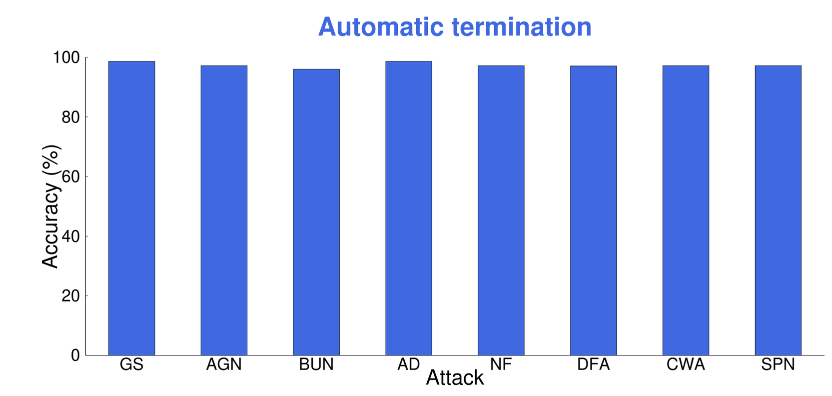


Experimental Stuff

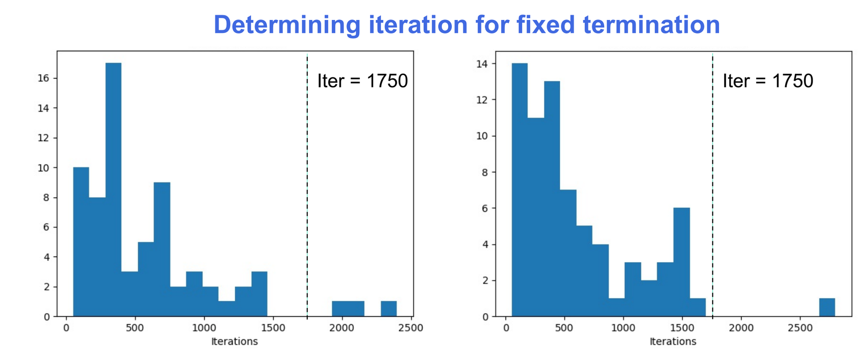
- **Main experiment** - Check effectiveness of Deep Protection:
  - ➢ 100 ImageNet examples (from 25 categories)
  - ➢ Apply 8 attacks per image using foolbox (800 inputs in total)
  - ➢ *Deep Protection* in two modes – *automatic* and *static*
  - ➢ Classify the obtained priors using resnet18
  - ➢ Test it in the real world
- **Exploring DIP application space** – *Text extraction from maps*:
  - ➢ *Key idea* – text is harder to draw than terrain/roads
  - ➢ Use *Deep Image Residual* (what is NOT generated by DIP yet)
  - ➢ With some processing, at correct iteration, all text can be extracted from maps

Results

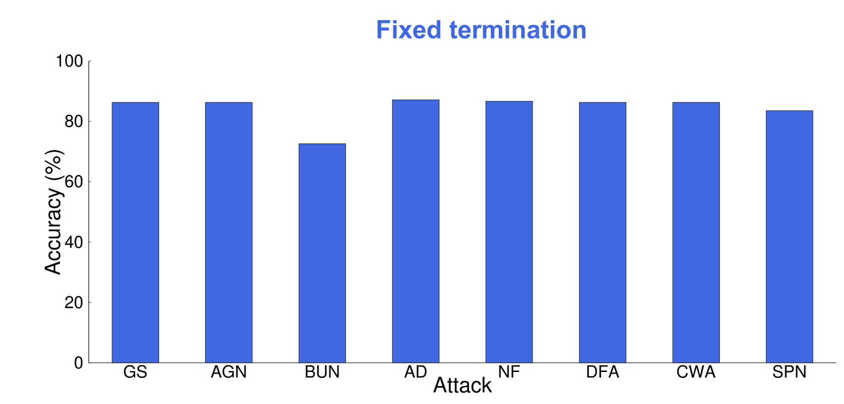
Automatic termination case:
- Iterations stop when 'true' class in top 5
- Represents Ideal accuracy
- *Avg accuracy: 97.4%*



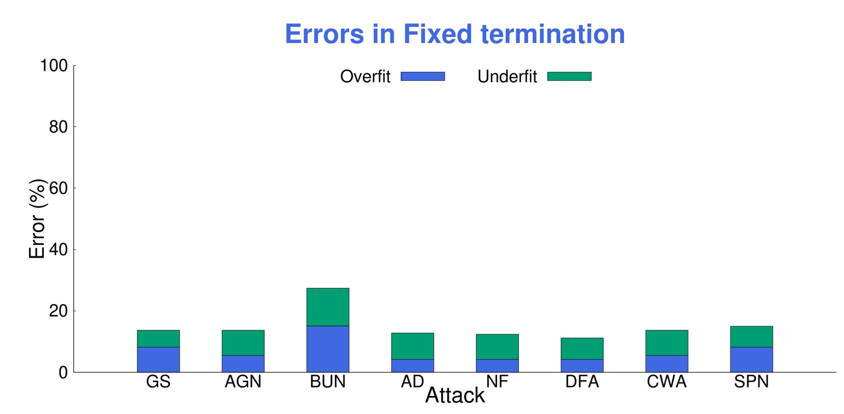
Determining iteration for fixed termination:
- Observed histogram of terminations in automatic case
- Majority terminations <1500 iterations; 1750 chosen



Fixed termination case:
- Process runs on all images for 1750 iterations
- *Avg accuracy: 84.4%*

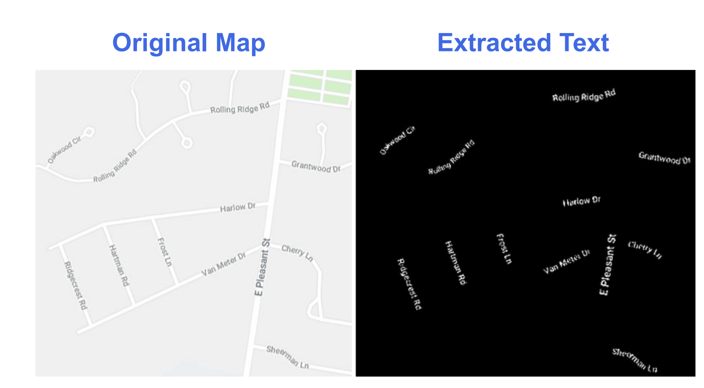


Errors in fixed termination:
- Overfit: Process ran too long; adversarial noise started to reappear
- Underfit: Process stopped too soon; reconstruction not sufficient for recognition



Real-world example:
- Tested on commercially available service (clarifai.com)

Text extraction from maps:
- Successfully extracted text from maps
- Termination fine tuned by observation



Key Insights

• **Universality**: In both modes, *Deep Protection* is quite effective on all 8 attacks, though need to work on automatic termination strategies to maximize accuracy

• **Attack type invariance**: No matter how the image is perturbed, the process of restoring image is the same – could indicate that this process is robust to any future attacks

• **Market ready**: No training required, can be deployed in-field right away

• **No Free Lunch**: Deep Protection is time consuming, though automatic termination could help speed it up

Going Ahead

• Work on strategies for automatic termination of *Deep Protection* process

• Design a binary classifier to detect if *Deep Protection* process is required for an input

• Explore other datasets (e.g., will this work on medical images?)

• Explore other applications of *DIP* technique

Acknowledgements: We thank Prof. Learned-Miller and Aruni Roy Choudhary for their guidance.