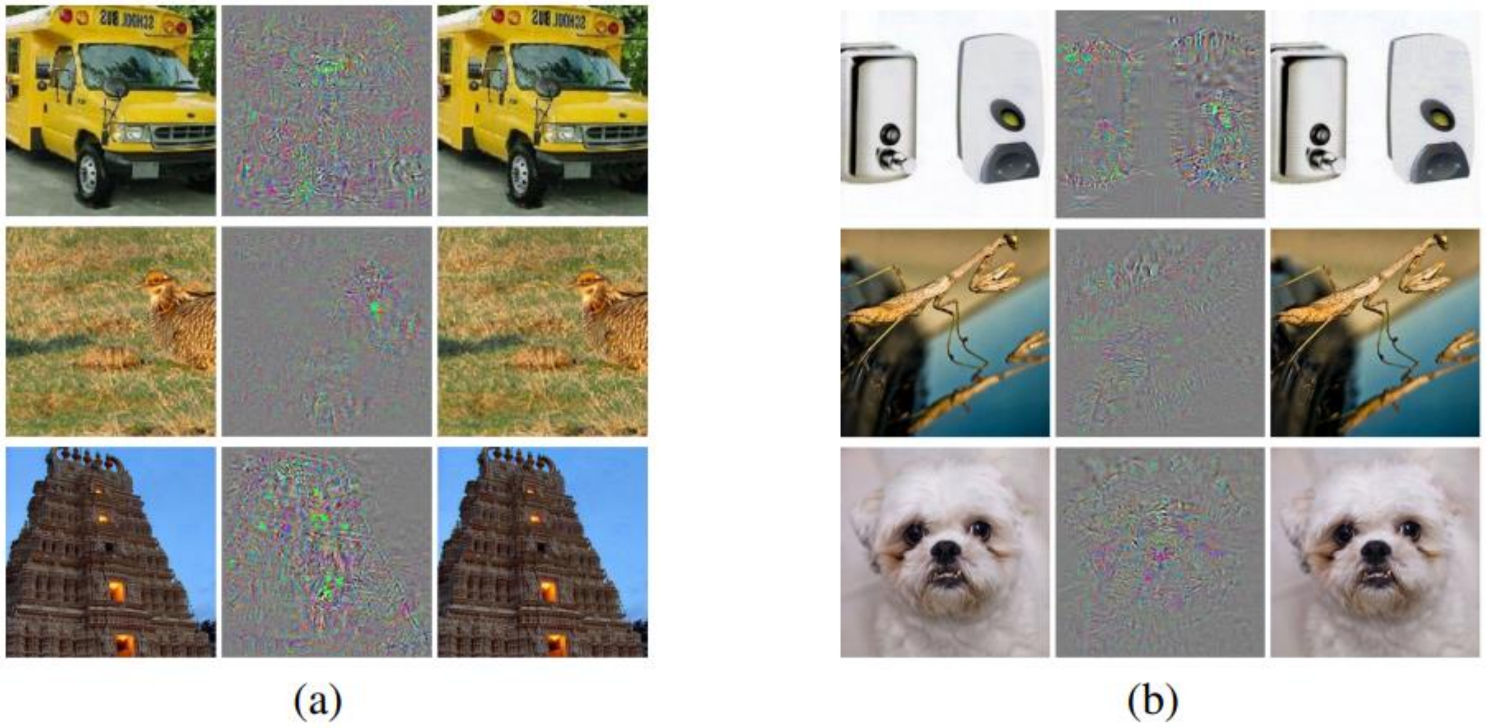
# Deep Neural Networks Are Easily Fooled

Anthony Dickson

# Examples

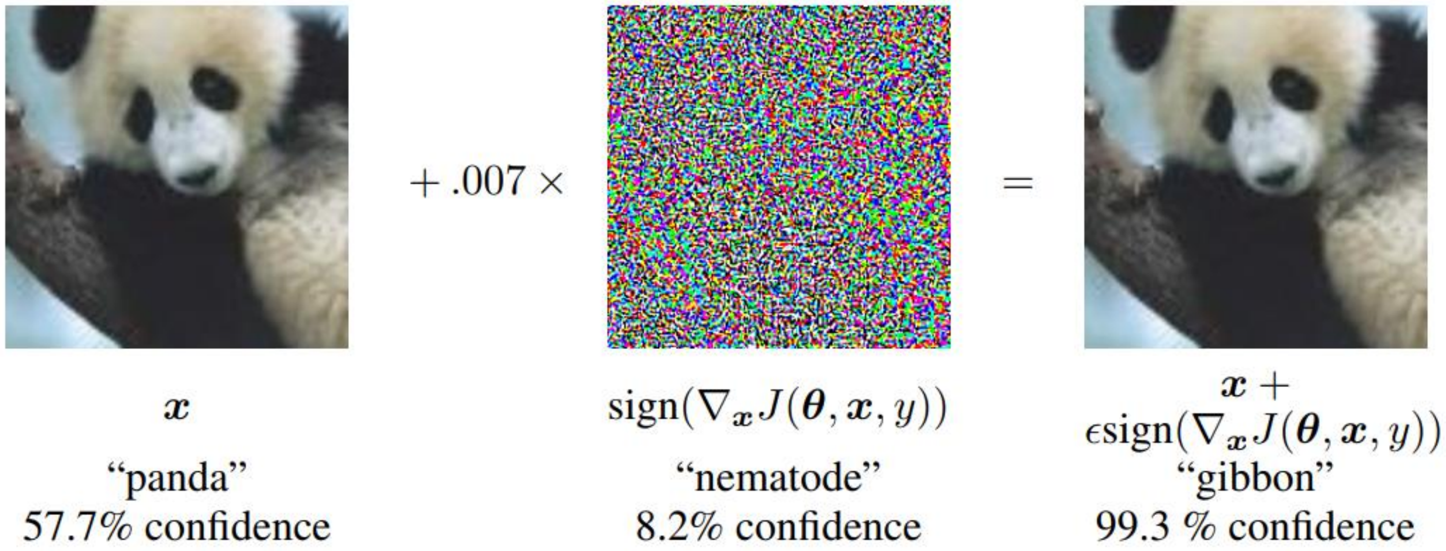
# Can you spot the ostrich?



- Left columns are the original image, right columns are all ‘pictures of ostriches’, centre columns are the difference of the two images magnified 10x.
- The adversarial examples (right columns) are indistinguishable from the original images!

Figure from: Szegedy, Christian, et al. "Intriguing properties of neural networks." arXiv preprint arXiv:1312.6199 (2013).

# Are you sure about that?




- Neural networks are often very confident when they are fooled.

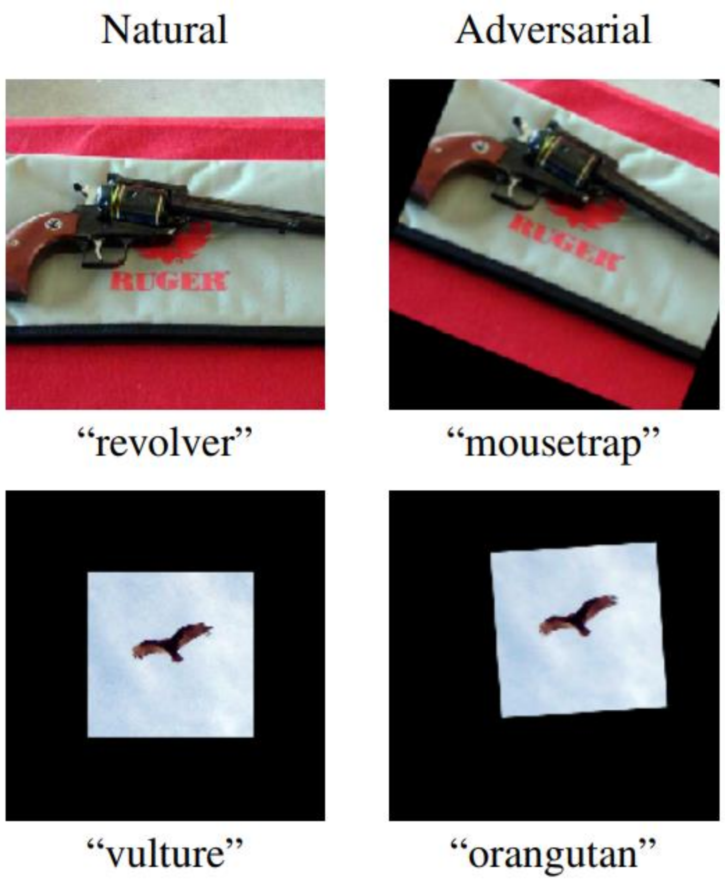
Figure from: Goodfellow, Ian J., Jonathon Shlens, and Christian Szegedy. "Explaining and harnessing adversarial examples." *arXiv preprint arXiv:1412.6572* (2014).

# Size does not matter


- A single pixel attack can be enough to fool a deep neural network…

Figure adapted from: Su, Jiawei, Danilo Vasconcellos Vargas, and Kouichi Sakurai. "One pixel attack for fooling deep neural networks." IEEE Transactions on Evolutionary Computation (2019).

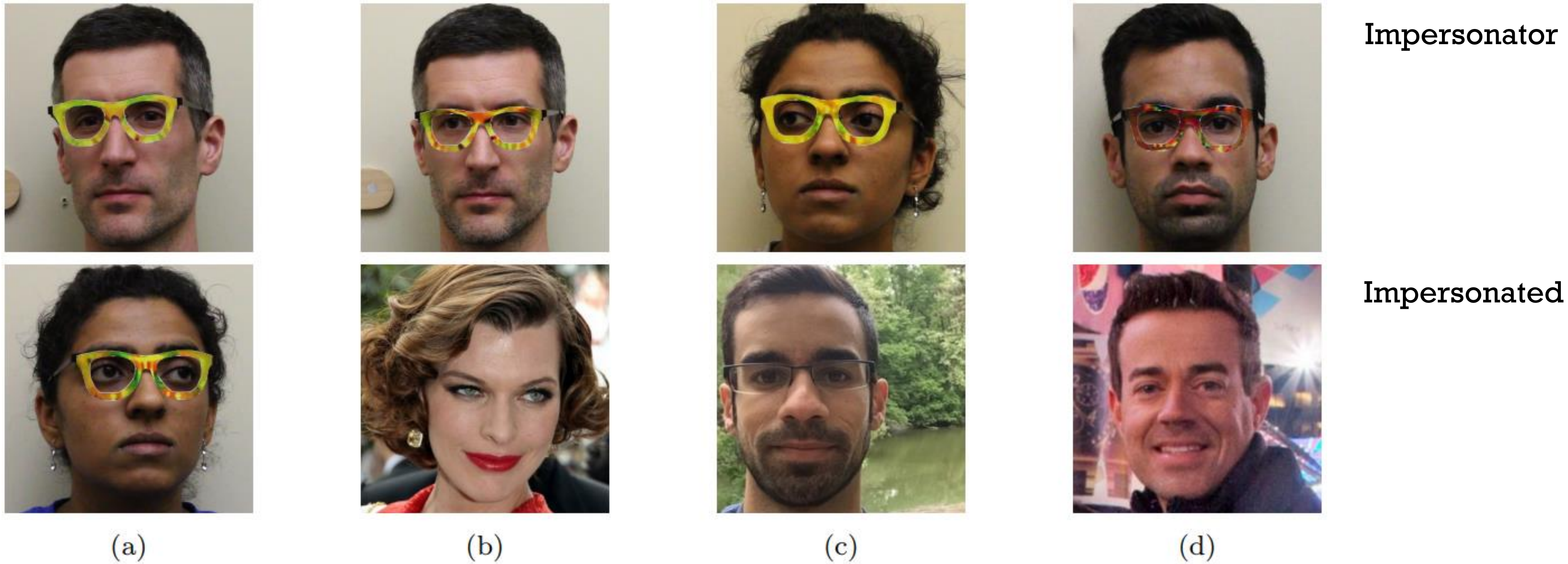
# Simple Transformations




- A simple rotation and translation can be all it takes to fool these convolutional neural networks.

Figure from: Engstrom, Logan, et al. "A rotation and a translation suffice: Fooling cnns with simple transformations." *arXiv preprint arXiv:1712.02779*(2017).

# These are not the people you are looking for…




- You would not want security systems to make important decisions solely based on the output of these networks if they can be fooled so easily.

Figure from: Sharif, Mahmood, et al. "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition." *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016.

# What is Going On?

# Adversarial Attacks in Computer Vision Tasks

- A pattern of noise is added to an image.
- The target/victim neural network assigns an incorrect label for the given image with a high level of confidence.
- The adversarial example is indistinguishable from the original image.

# Not Just Images

- Adversarial attacks are widespread throughout these deep learning models.
- Adversarial attacks can work for other types of data.
- Attacks are not limited to convolutional neural networks.
- There has been work showing successful attacks on natural language processing and speech recognition.

# A Landmark Paper

- Christian  Szegedy et al.'s 2013 work "Intriguing properties of neural networks'' was one of the first to touch on the subject.
- They created adversarial examples by finding the smallest pattern of noise that causes the target network to classify the given input with a certain label.
- They showed that adversarial examples *generalise* across different models and across models trained on different data.

# Adversarial Attacks Are Effective and Pervasive

- Deep neural networks are easily fooled with adversarial examples.
- When they are fooled, they have a high level of confidence.
- Adversarial attacks seem to be possible for any type of neural network or dataset.
- Adversarial examples generalise.

# Robust Neural Networks

# Adversarial Training

- Train on adversarial examples.
- Provides limited defence.
- Adversarially trained networks are less vulnerable to attacks but are still fooled with a high level of confidence.

# Robust Optimisation

- Incorporate an adversary into the optimisation process.
- Adversarial loss can be used as a regulariser.
  - $\underbrace{\eta = \epsilon \operatorname{sign}\left(\nabla_x J(\theta, x, y)\right)}_{\textit{Adversarial Example}}$
  - $\tilde{J}(\theta, x, y) = \underbrace{\alpha J(\theta, x, y)}_{\textit{Normal Loss}} + \underbrace{(1 - \alpha) J(\theta, x + \eta, y)}_{\textit{Adversarial Loss}}$
- Some methods opt for a minimax game theory approach.
  - $\underbrace{\min_{\theta}}_{\textit{Model}} \boldsymbol{E}_{(x,y)\sim D}\left[\underbrace{\max_{\eta \in S} L(\theta, x + \eta, y)}_{\textit{Adversary}}\right]$
- Currently the most effective defence against adversarial attacks.
- Some of the methods can be a bit slow.

# A Feature, Not a Bug?

*Ilyas et al.*, 2019

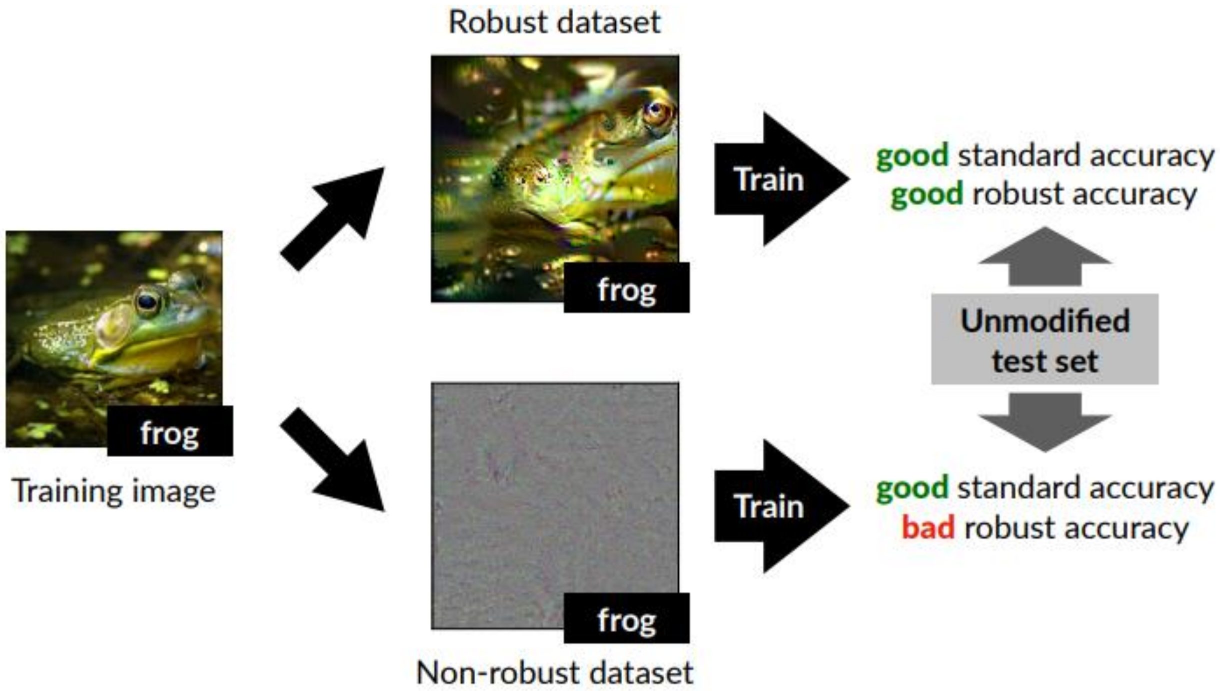
# Adversarial Examples Are Not Bugs, They Are Features

Previously: Adversarial examples are possibly due to:

- Linearity of deep neural networks (e.g. ReLU activation)
- High-dimensional geometry of data and complex decision boundaries.

This paper: Adversarial examples are the result of the supervised learning paradigm.

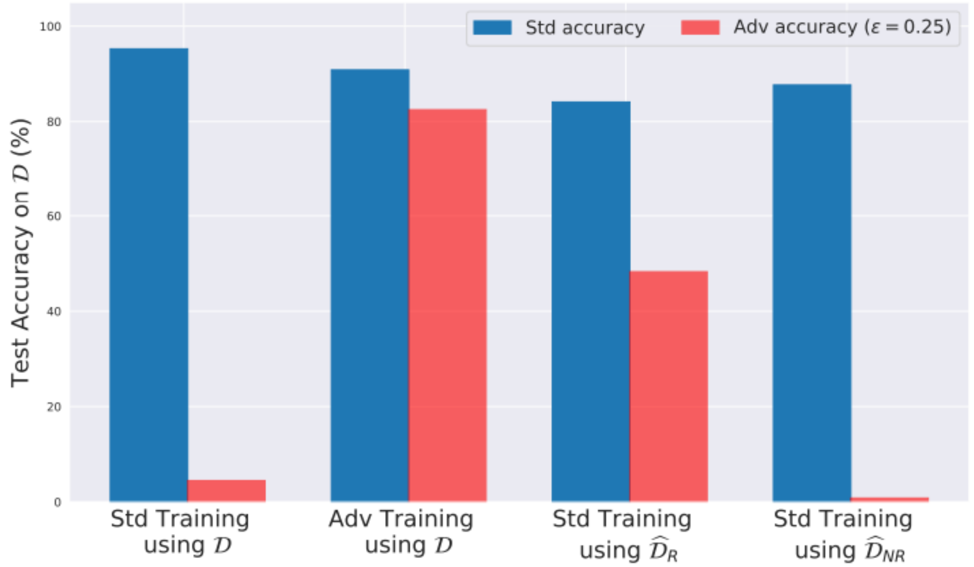
# Robust and Non-robust Features




- Authors propose the existence of robust and non-robust features.
- Robust features are resistant against adversarial perturbations to some degree, non-robust features are not.
- Hypothesis: models trained with supervised learning rely on both robust and non-robust features.

Figure adapted from: Ilyas, Andrew, et al. "Adversarial Examples Are Not Bugs, They Are Features." *arXiv preprint arXiv:1905.02175* (2019).

# Training on Robust and Non-Robust Features




- Train separate models on the robust and non-robust features.
- On clean test set there is little difference between models trained on robust or non-robust features.
- Models trained on robust features are indeed more resilient against attacks.

Figure from: Ilyas, Andrew, et al. "Adversarial Examples Are Not Bugs, They Are Features." *arXiv preprint arXiv:1905.02175* (2019).

# Summary

- Deep neural networks are vulnerable to adversarial attacks.
- These attacks generalise between different models.
- We can build neural networks that are more robust against these attacks by:
  - Adding a pool of adversarial examples to the training data
  - Training models with adversarial loss as a regulariser
  - Training models with a loss function that incorporates the adversary
  - Training models with a dataset of robust features.
- Adversarial examples may actually be a feature of supervised learning, rather than a bug.


That's all Folks!