Safety verification for deep neural networks

Kovács Ádám, Homework for SWVV

slides based on paper CAV 2017, and ICST2018 keynote by Marta Kwiatkowska, images from paper CAV 2017

https://arxiv.org/abs/1610.06940

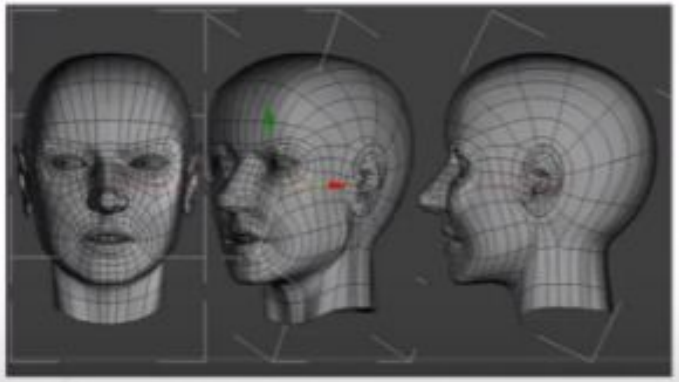
BME AUT
Department of Automation and Applied Informatics

The rise of deep learning

- Timeline:
  - 1940s - First proposed
  - 1998 - Convolutional nets
  - 2006 - Deep nets trained
  - 2015 - Vision breakthrough
  - 2016 - Reinf. learning (Win at Go)
- Reasons
  - Lots of data
  - Good architectures
  - The rise of GPUs
  - Flexible, easy to use models

Lots of interest from tech companies

DeepFace
Closing the Gap to Human-Level Performance in Face Verification

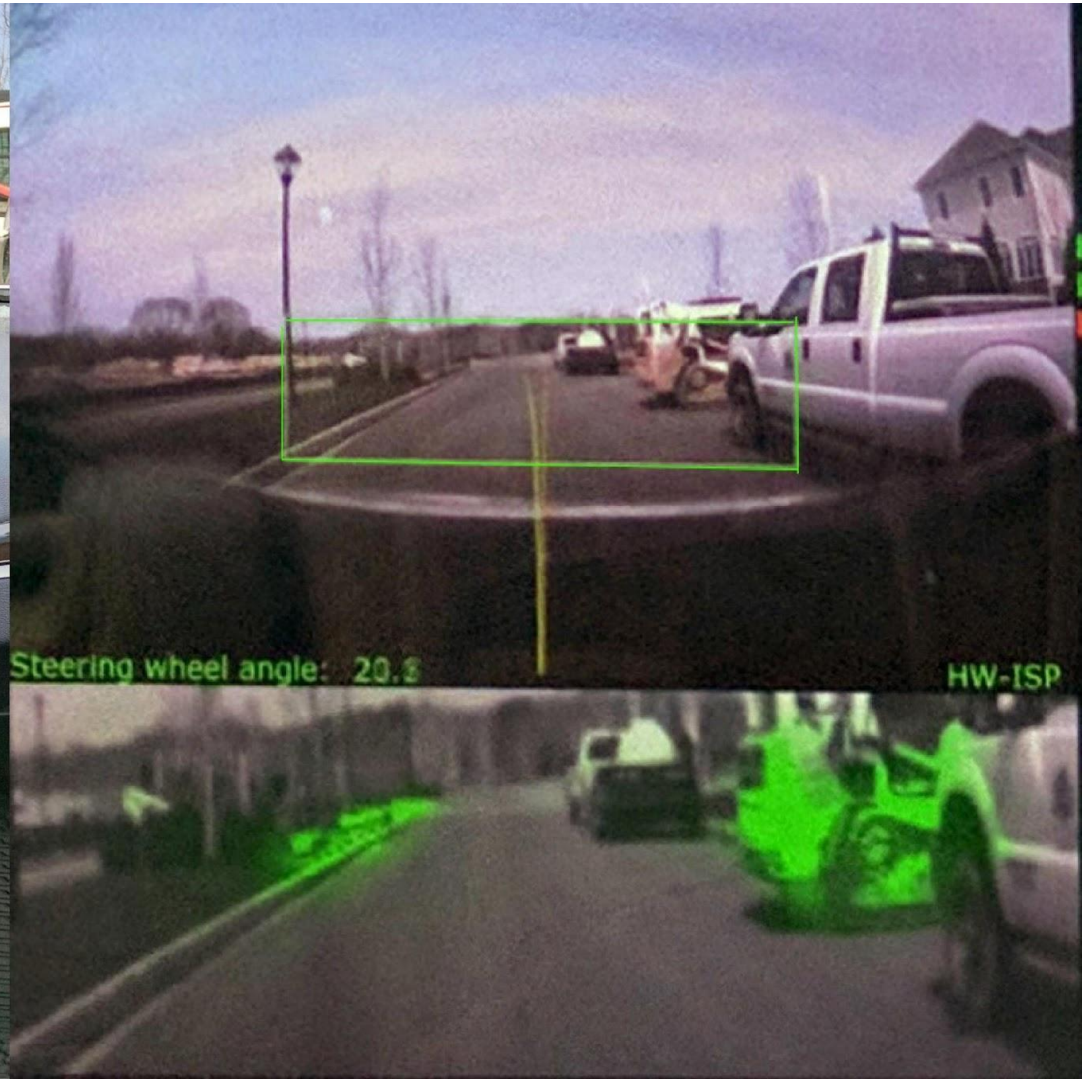


Yaniv Taigman
Ming Yang
Marc'Aurelio Ranzato
Lior Wolf
- 2014

97.35% accuracy
Trained on the largest facial dataset – 4M facial images belonging to more than 4,000 identities.

Google Translate—here shown on a mobile phone—will use deep learning to improve its translations between texts.

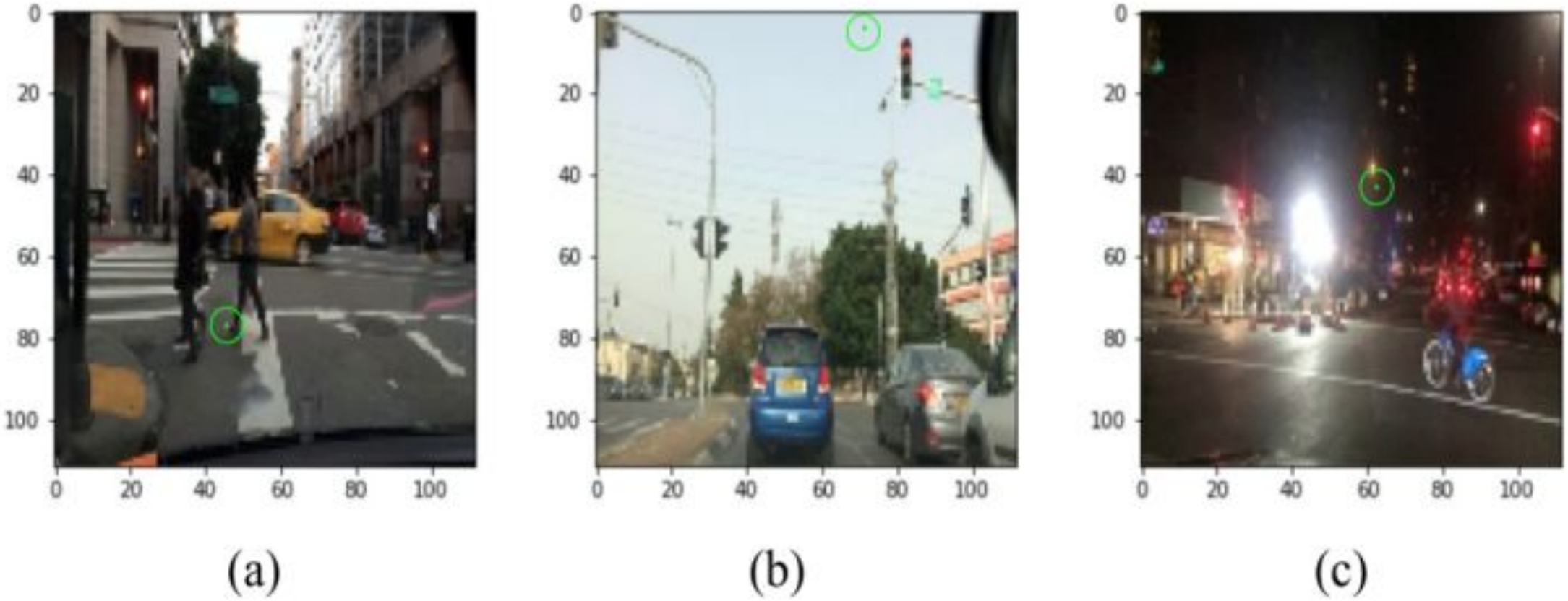
Automotive industry



Self-driving cars

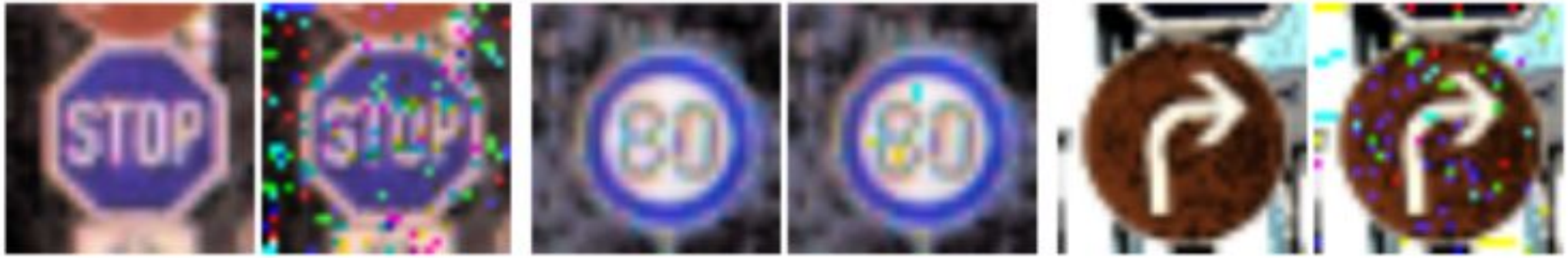
- PilotNet by NVIDIA
  - > End-to-end controller for self-driving cars
  - > Deep learning techniques
  - > Keep and change the lane
  - > Trained on data from human driven cars
- Traffic sign recognition
  - > Object recognition
- Neural nets are often treated as a black box
  - > How can we guarantee their decision?

Nexar traffic sign benchmark



TACAS 2018, https://arxiv.org/abs/1710.07859

German traffic sign benchmark



stop | 30m speed limit | 80m speed limit | 30m speed limit | go right | go straight

Safety of self-driving cars

Self-Driving Uber Car Kills Pedestrian in Arizona, Where Robots Roam

Leer en español

By DAISUKE WAKABAYASHI MARCH 19, 2018

Tesla Says Crashed Vehicle Had Been on Autopilot Before Fatal Accident

By GREGORY SCHMIDT MARCH 31, 2018

RELATED COVERAGE

Tesla Looked Like the Fi
Ask if It Has One. MARC

Fatal Tesla Crash Raises New Questions About Autopilot System

U.S. Safety Agency Criticizes Tesla Crash Data Release

Deep neural networks can be fooled

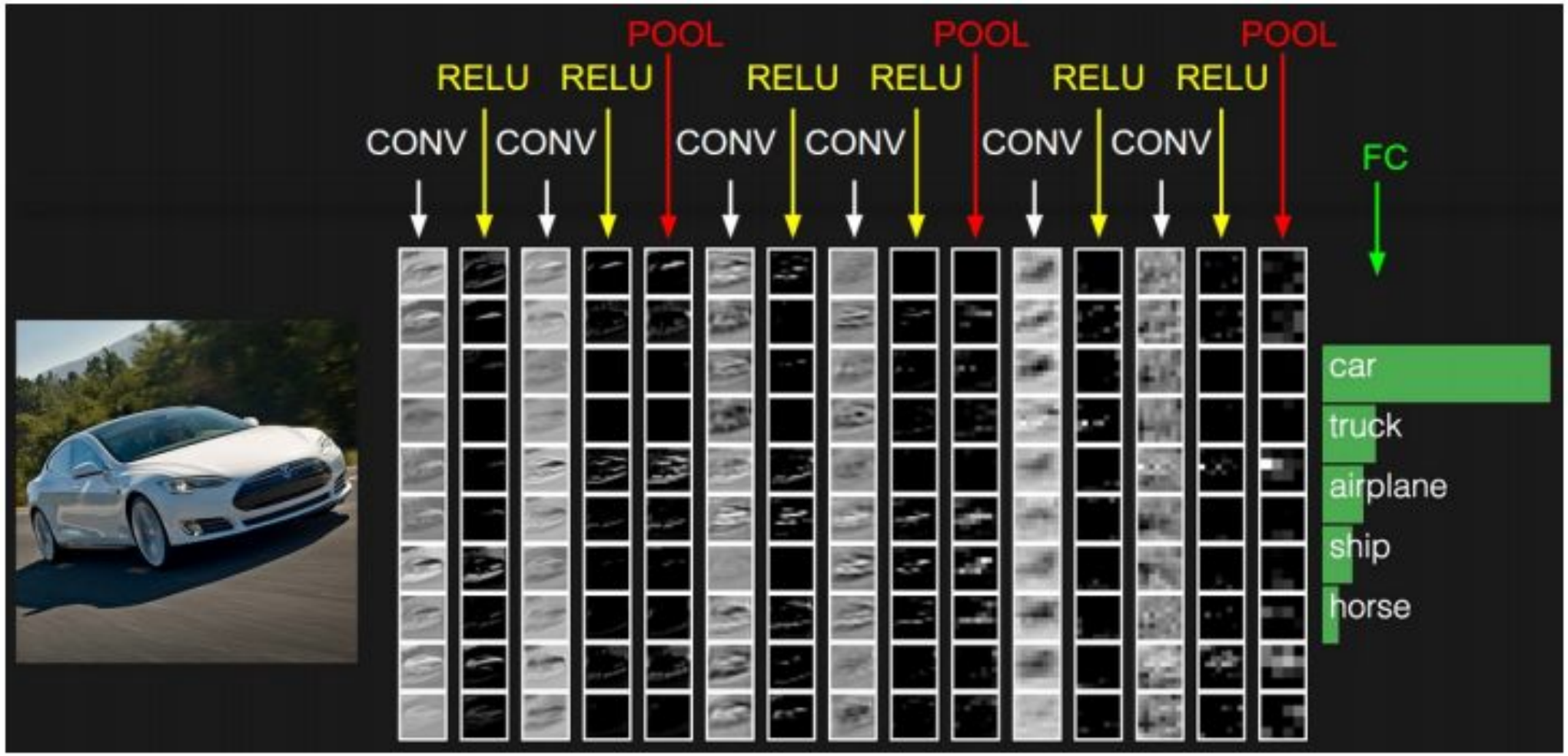
- They can be easily unstable with **adversarial perturbations**
- Often a very small change to the image is enough *(Szegedy et al 2014, Biggio et al 2013)*
- Artifical white noise
- Practical attacks
- The paper claims they are transferable

Physical attacks need to be considered.

The methodology

- To ensure robustness
- First step towards methodology to ensure safety of classification decisions
  - Visible and human-recognisable perturbations: change of camera angle, snow, sign imperfections…
  - They should not result in class change
  - The paper focuses on individual decisions
- Towards an automated verification framework

Deep feed-forward neural network



The problem setting

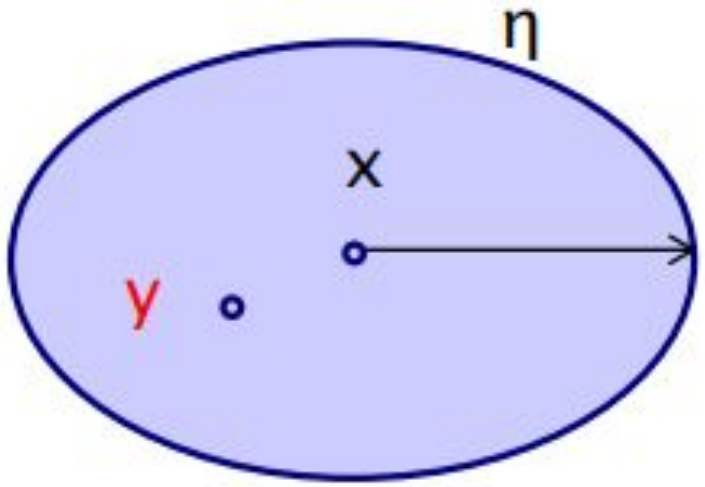
- vector spaces D_{L0}, D_{L1}, …, D_{Ln}, one for each layer
- $f : D_{L0} \rightarrow \{c_1,\ldots c_k\}$ classifier function modelling human perception ability
- The network f' approximates f from M training examples
  - > built from activation functions $\varphi 0$, $\varphi 1$, …, φn, one for each layer
  - > For an image x its activation layer k is
    - $\alpha_{x,k} = \varphi_k(\varphi_{k-1}(\ldots\varphi_1(x)))$

Verification for neural networks

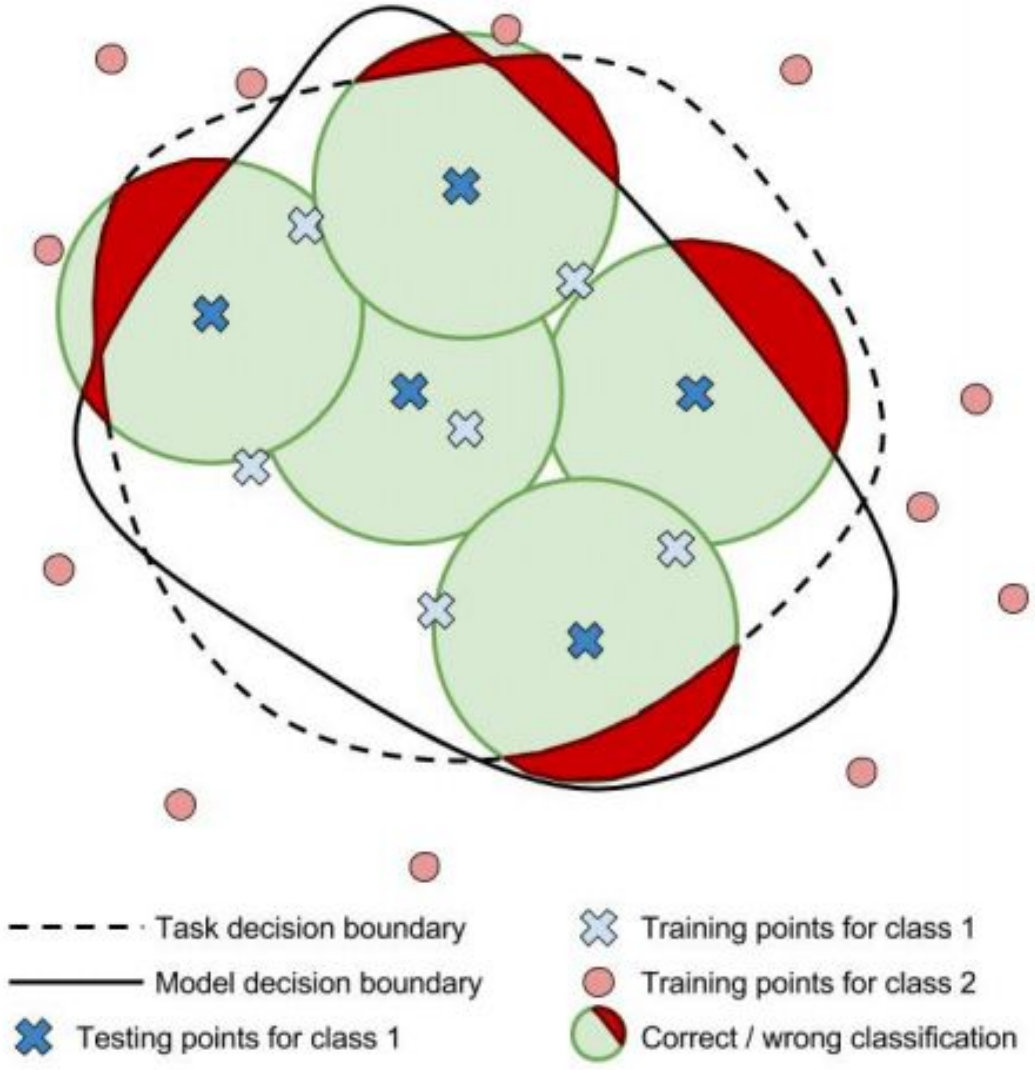
- They are little studied
- Pulina and Tachela 2010
  - Encode the network using constraints
  - SMT solving, does not scale
- Reluplex [Barret et al 2017]
  - Similar but for ReLU activation
  - SMT solver
  - Hard to scale

Safety of classification decisions

- Assuring safety is a complex process
- The paper focuses on safety at a point
  - > Consider region supporting decision at point x(image)
- Questions
  - > What diameter for the region?
  - > What is an acceptable pertrubation?
  - > Introduce the concept of manipulation



Training vs testing vs verification



Verification framework

- Take a set of manipulations and a region η
  - > Exhaustively search the region for misclassifications
- Challenges
  - > High dimensionality, nonlinearity, huge scale
- The approach
  - > Propagate the network layer by layer, i.e. for each activation in the layer there is a region η
  - > SMT search based using Z3

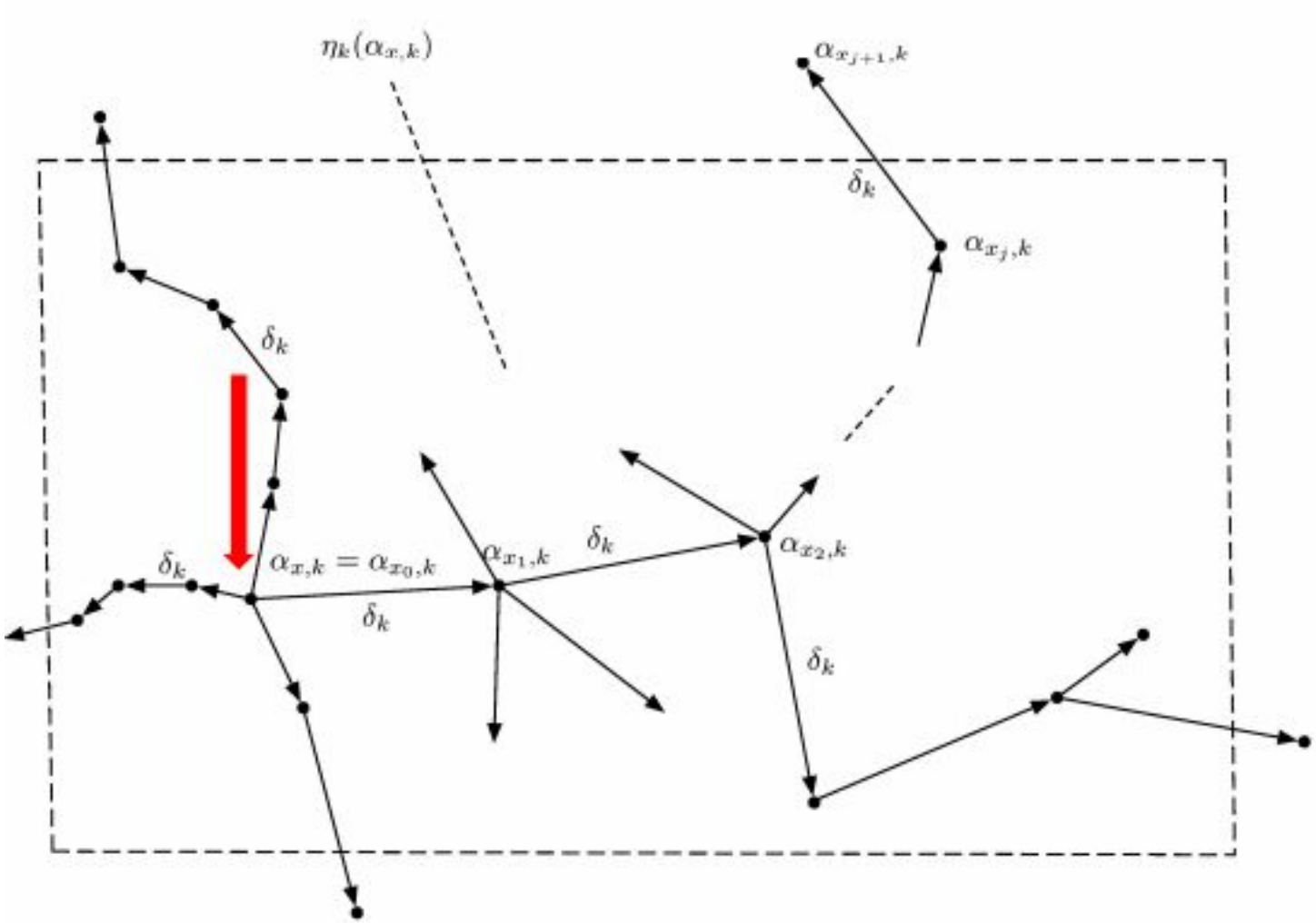
Manipulations

- Consider a family of operators that pertrub activations in layer k
  - To imitate scratches, weather conditions, camera angle, etc..
- Intuitively, safety of network N at a point x with the region $\eta_k(\alpha_x,k)$ set of manipulations Δ_k means that perturbing activation $\alpha_{x,k}$ by manipulations from Δ_k will not result in a class change

Region coverage

- Exhaustive search of the region for adversarial manipulations
  - > If found, fine-tune the network
  - > Else, declare region safe with the specified manipulations
- Method
  - > Discretize the region
  - > Cover the region with ‚ladders'
  - > If there are no missclassification in the tree of ladders, the region is safe
  - > The search is exhaustive because of minimality of manipulations

The ladders


$\eta_k(\alpha_{x,k})$
$\alpha_{x_{j+1},k}$
δ_k
$\alpha_{x_j,k}$
δ_k
δ_k
$\alpha_{x,k} = \alpha_{x_0,k}$
$\alpha_{x_1,k}$
δ_k
$\alpha_{x_2,k}$
δ_k
δ_k

Case study: Nexar

- Using a method called Monte Carlo Tree Search the authors were able to reduce the accuracy of the network to 0%
- On average, each input took less than a second to manipulate (.304 seconds)
- On average each image was vulnerable to 3 pixel changes

Challenges for verification of NNs

- Fascinating application domain, huge challenges
- Many aspects to make it difficult
  - No source code (only weights)
  - Variety of activation functions
  - High dimensionality of input space
  - Lack of interpretability
- The goal of the paper is to provide
  - Scalable and efficient methods
  - With provable guarantees

Thank you for your attention!