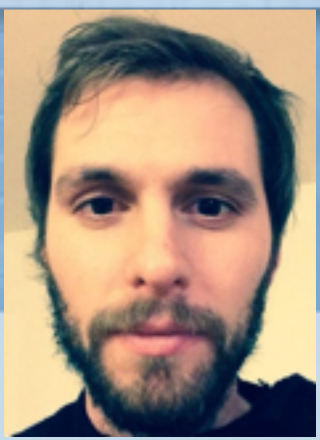
# The Mathematics of Deep Learning

ICCV Tutorial, Santiago de Chile, December 12, 2015

Joan Bruna (Berkeley), Raja Giryes (Duke), Guillermo Sapiro (Duke), Rene Vidal (Johns Hopkins)



# Motivations and Goals of the Tutorial

- **Motivation:** Deep networks have led to dramatic improvements in performance for many tasks, but the mathematical reasons for this success remain unclear.

- **Goal:** Review very recent work that aims at understanding the mathematical reasons for the success of deep networks.

- **What we will do:** Study theoretical questions such as
  - What properties of images are being captured/exploited by DNNs?
  - Can we ensure that the learned representations are globally optimal?
  - Can we ensure that the learned representations are stable?

- **What we will not do:** Show X% improvement in performance for a particular application.

# Tutorial Schedule

- 14:00-14.30: Introduction
- 14:30-15.15: Global Optimality in Deep Learning (René Vidal)
- 15:15-16.00: Coffee Break
- 16:00-16:45: Scattering Convolutional Networks (Joan Bruna)
- 16:45-17:30: Stability of Deep Networks (Raja Giryes)
- 17.30-18:00: Questions and Discussion

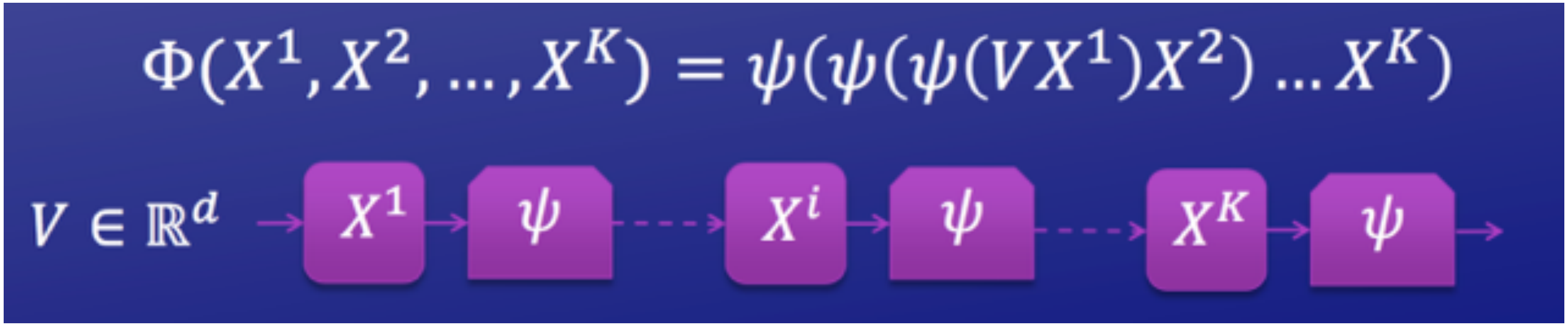
# Disclaimer

What do we mean by 'Deep Learning' in this tutorial?

# Disclaimer

What do we mean by 'Deep Learning' in this tutorial?

- A class of signal representations that are hierarchical:

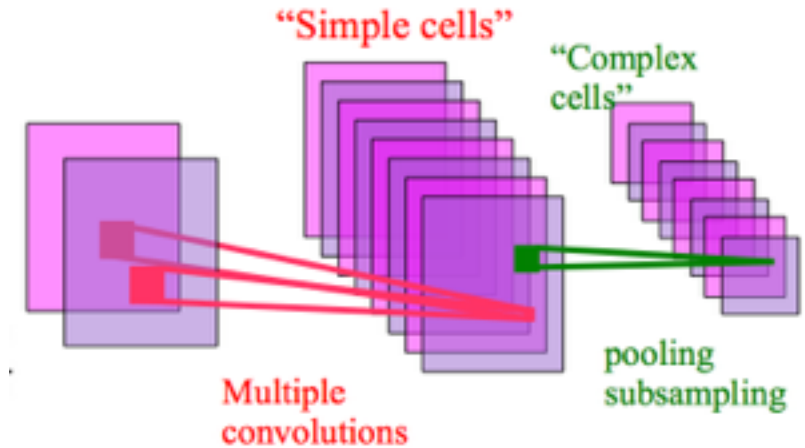



- The optimization procedure by which these representations are learnt from data end-to-end.

*figure from Raja Giryes*

# Early Hierarchical Feature Models for Vision

- Hubel & Wiesel [60s] Simple & Complex cells architecture:




- Fukushima's Neocognitron [70s]

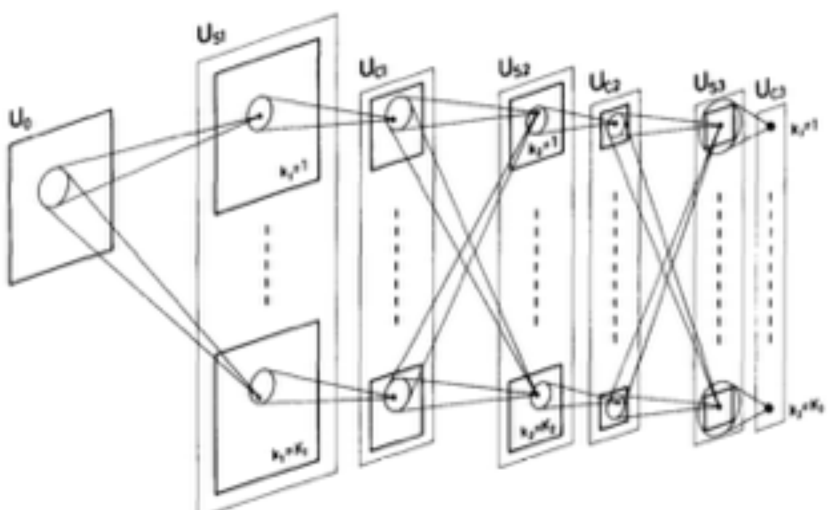
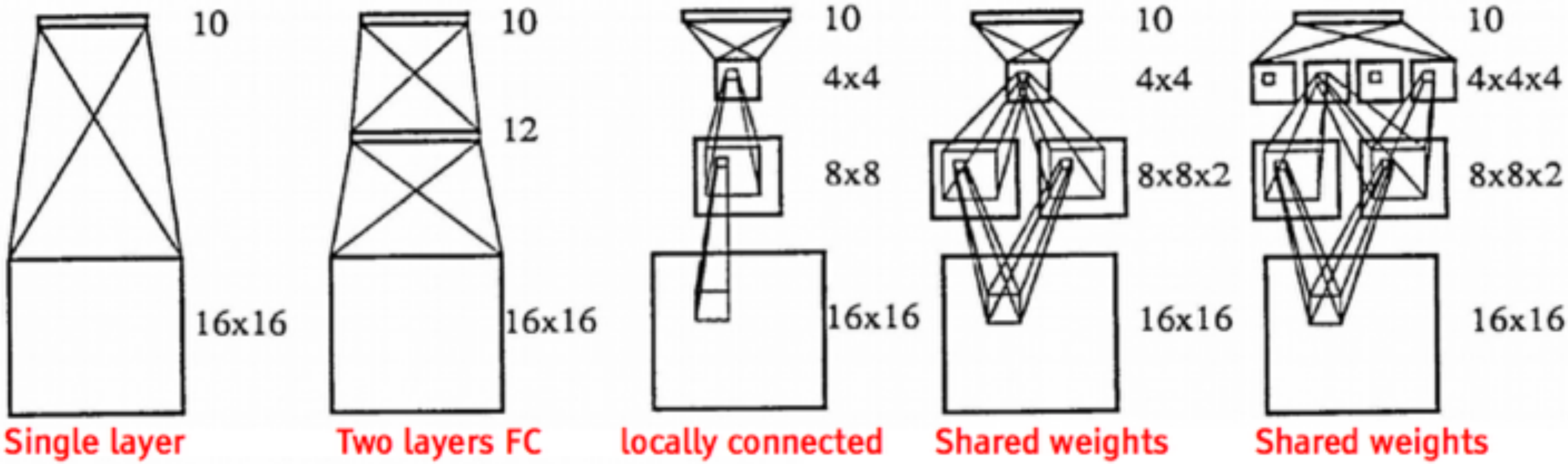



Fig. 2. Schematic diagram illustrating the interconnections between layers in the neocognitron

*figures from Yann LeCun's CVPR'15 plenary*

# Early Hierarchical Feature Models for Vision

- Yann LeCun's Early ConvNets [80s]:

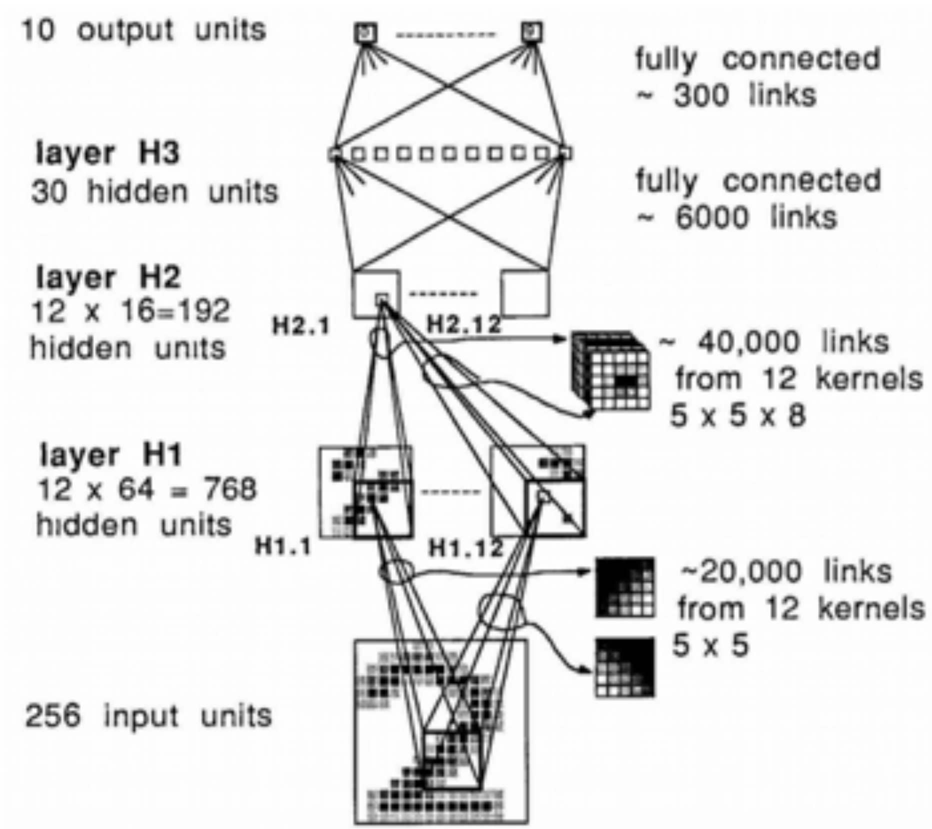


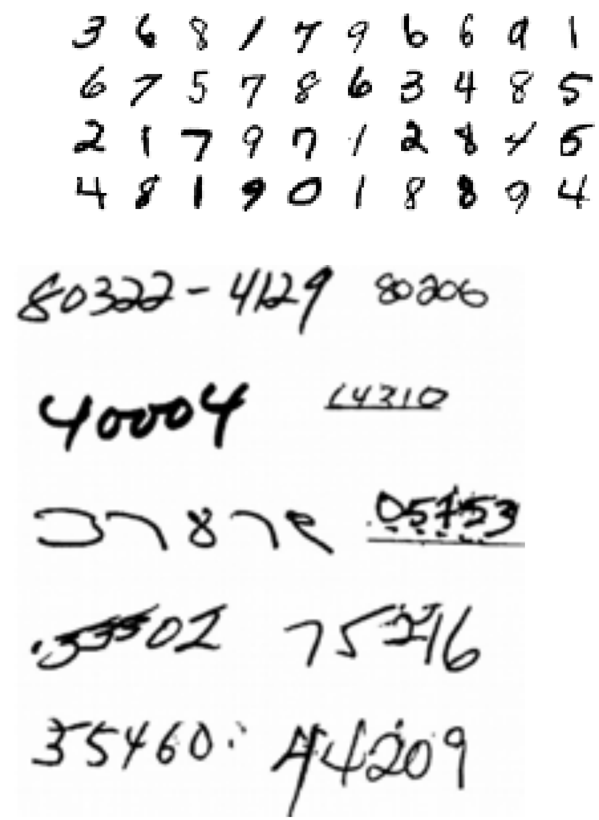

  – Used for character recognition
  – Trained with back propagation.

*figures from Yann LeCun's CVPR'15 plenary*

# Deep Learning pre-2012

- Despite its very competitive performance, deep learning architectures were not widespread before 2012.
  - State-of-the-art in handwritten pattern recognition [LeCun et al. '89, Ciresan et al, '07, etc]



*figures from Yann LeCun's CVPR'15 plenary*

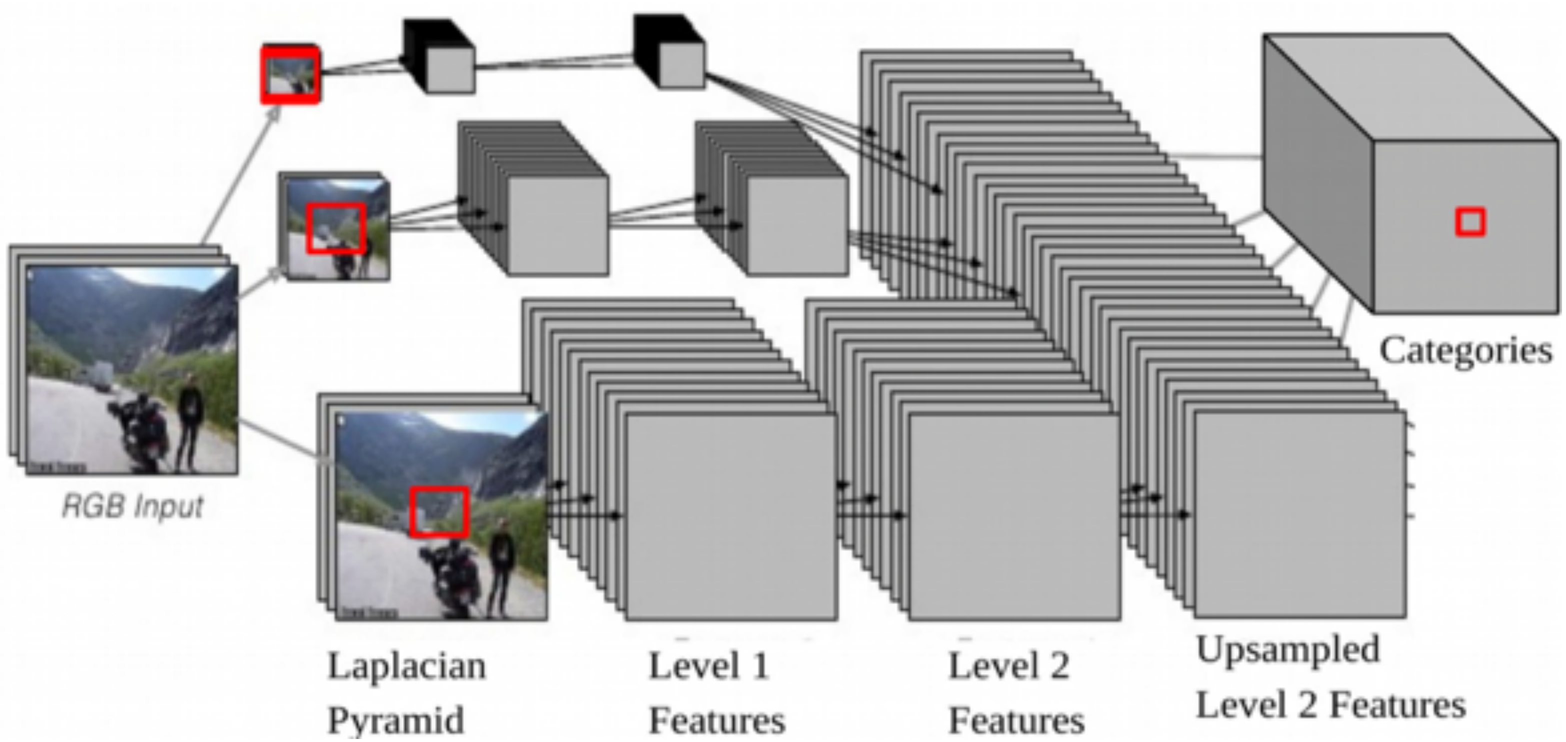
# Deep Learning pre-2012

- Despite its very competitive performance, deep learning architectures were not widespread before 2012.
  - Face detection [Vaillant et al'93,'94 ; Osadchy et al, '03, '04, '07]

*(Yann's Family)*

# Deep Learning pre-2012

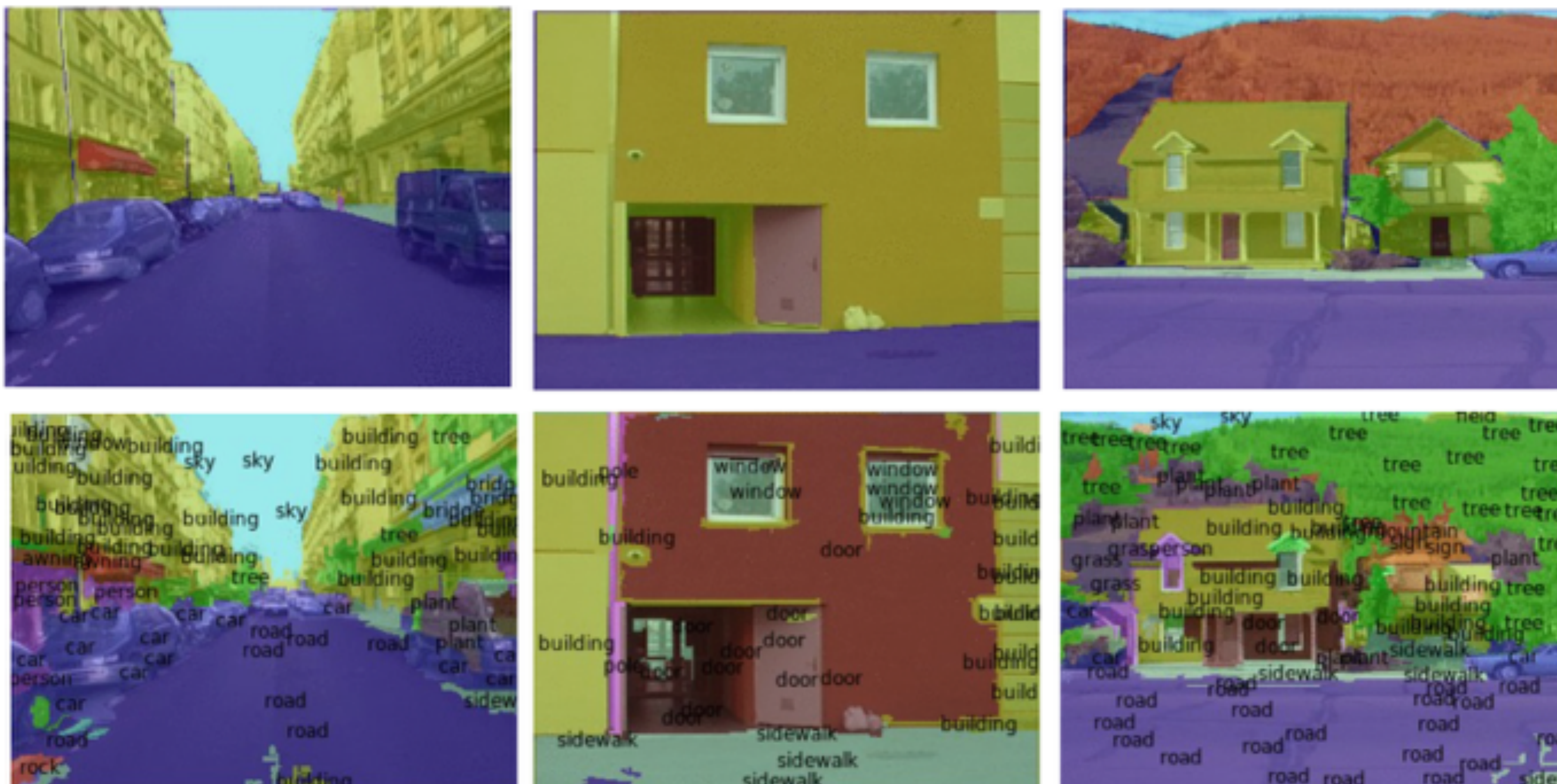
- Despite its very competitive performance, deep learning architectures were not widespread before 2012.
  - Scene Parsing [Farabet et al, '12,'13]




*figures from Yann LeCun's CVPR'15 plenary*

# Deep Learning pre-2012

- Despite its very competitive performance, deep learning architectures were not widespread before 2012.
  - Scene Parsing [Farabet et al, '12,'13]




*figures from Yann LeCun's CVPR'15 plenary*

# Deep Learning pre-2012

- Despite its very competitive performance, deep learning architectures were not widespread before 2012.
  - Too many parameters to learn from few labeled examples.
  - "I know my features are better for this task".
  - Non-convex optimization? No, thanks.
  - Black-box model, no interpretability.

# Deep Learning Golden age in Vision

- 2012-2014 Imagenet results:

CNN
non-CNN

| 2012 Teams | %error |
|---|---|
| Supervision (Toronto) | 15.3 |
| ISI (Tokyo) | 26.1 |
| VGG (Oxford) | 26.9 |
| XRCE/INRIA | 27.0 |
| UvA (Amsterdam) | 29.6 |
| INRIA/LEAR | 33.4 |

| 2013 Teams | %error |
|---|---|
| Clarifai (NYU spinoff) | 11.7 |
| NUS (singapore) | 12.9 |
| Zeiler-Fergus (NYU) | 13.5 |
| A. Howard | 13.5 |
| OverFeat (NYU) | 14.1 |
| UvA (Amsterdam) | 14.2 |
| Adobe | 15.2 |
| VGG (Oxford) | 15.2 |
| VGG (Oxford) | 23.0 |

| 2014 Teams | %error |
|---|---|
| GoogLeNet | 6.6 |
| VGG (Oxford) | 7.3 |
| MSRA | 8.0 |
| A. Howard | 8.1 |
| DeeperVision | 9.5 |
| NUS-BST | 9.7 |
| TTIC-ECP | 10.2 |
| XYZ | 11.2 |
| UvA | 12.1 |

- 2015 results: MSRA under **3.5%** error. (using a CNN with 150 layers!)

*figures from Yann LeCun's CVPR'15 plenary*

# Puzzling Questions

- What made this result possible?
  - Larger training sets (1.2 million, high-resolution training samples, 1000 object categories)
  - Better Hardware (GPU)
  - Better Learning Regularization (Dropout)

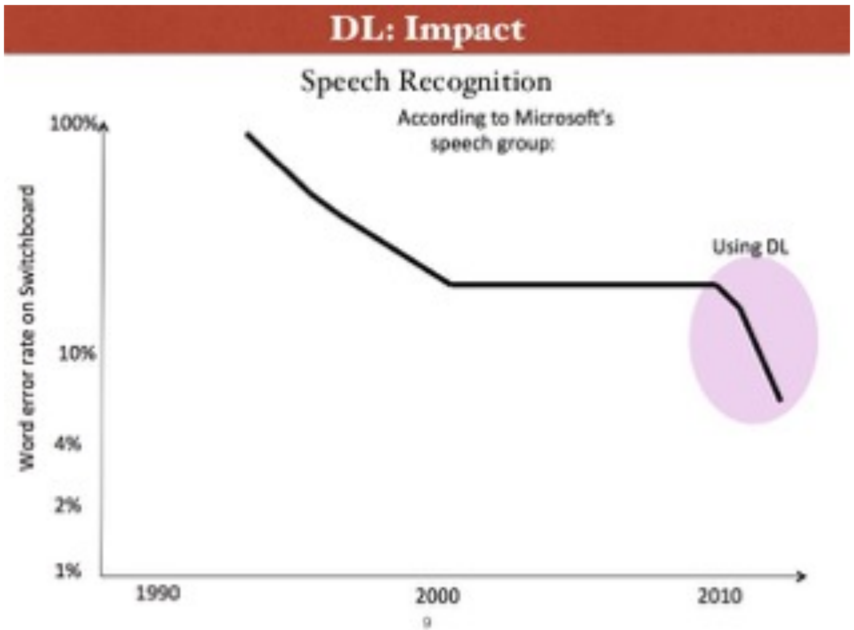
- Is this just for a particular dataset?

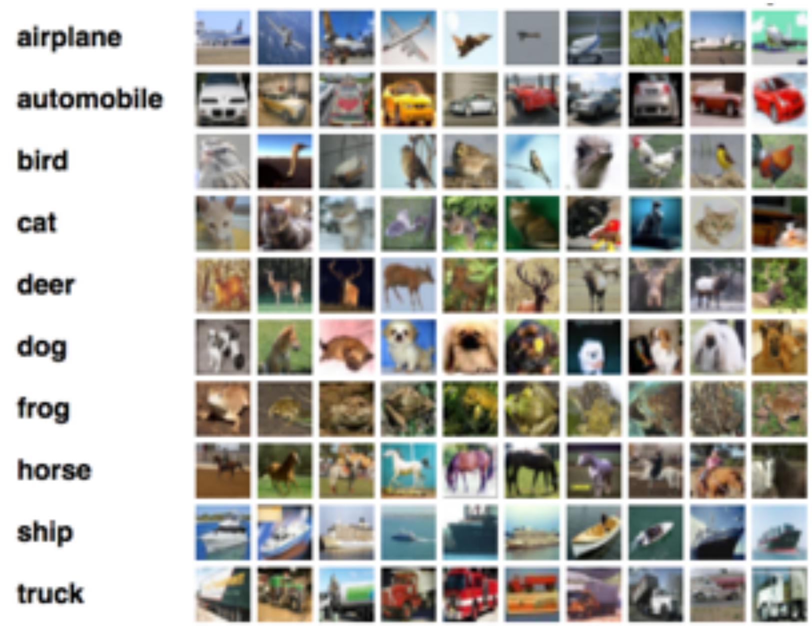
- Is this just for a particular task?

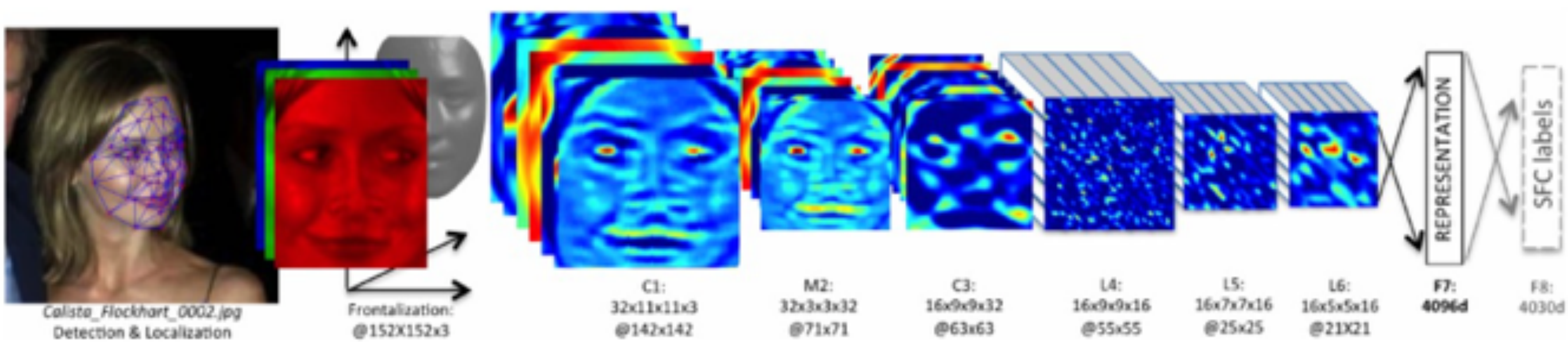
- Why are these architectures so efficient?

# Is it just for a particular dataset?

- No. Nowadays CNNs hold the state-of-the-art on virtually any object classification task.




*figures from Yann LeCun's NIPS'15 tutorial*

# Is it just for a particular task?

- No. CNN architectures also obtain state-of-the-art performance on many other tasks:

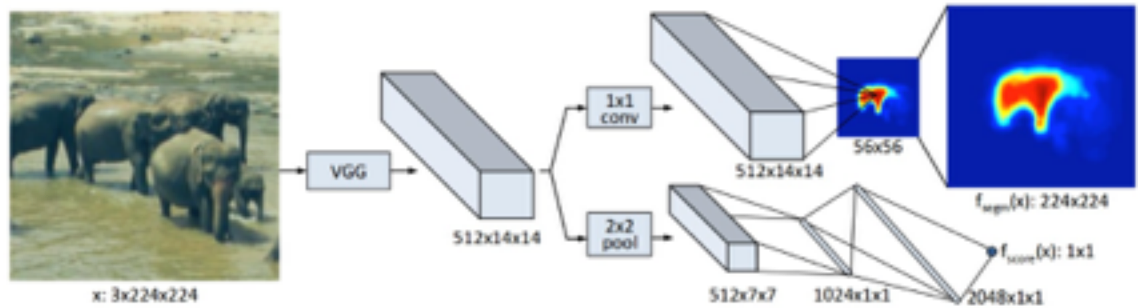
Object Localization
[R-CNN, HyperColumns, Overfeat, etc.]

Pose estimation [Thomson et al, CVPR'15]

*figures from Yann LeCun's CVPR'15 plenary*

# Is it just for a particular task?

- No. CNN architectures also obtain state-of-the-art performance on other tasks:

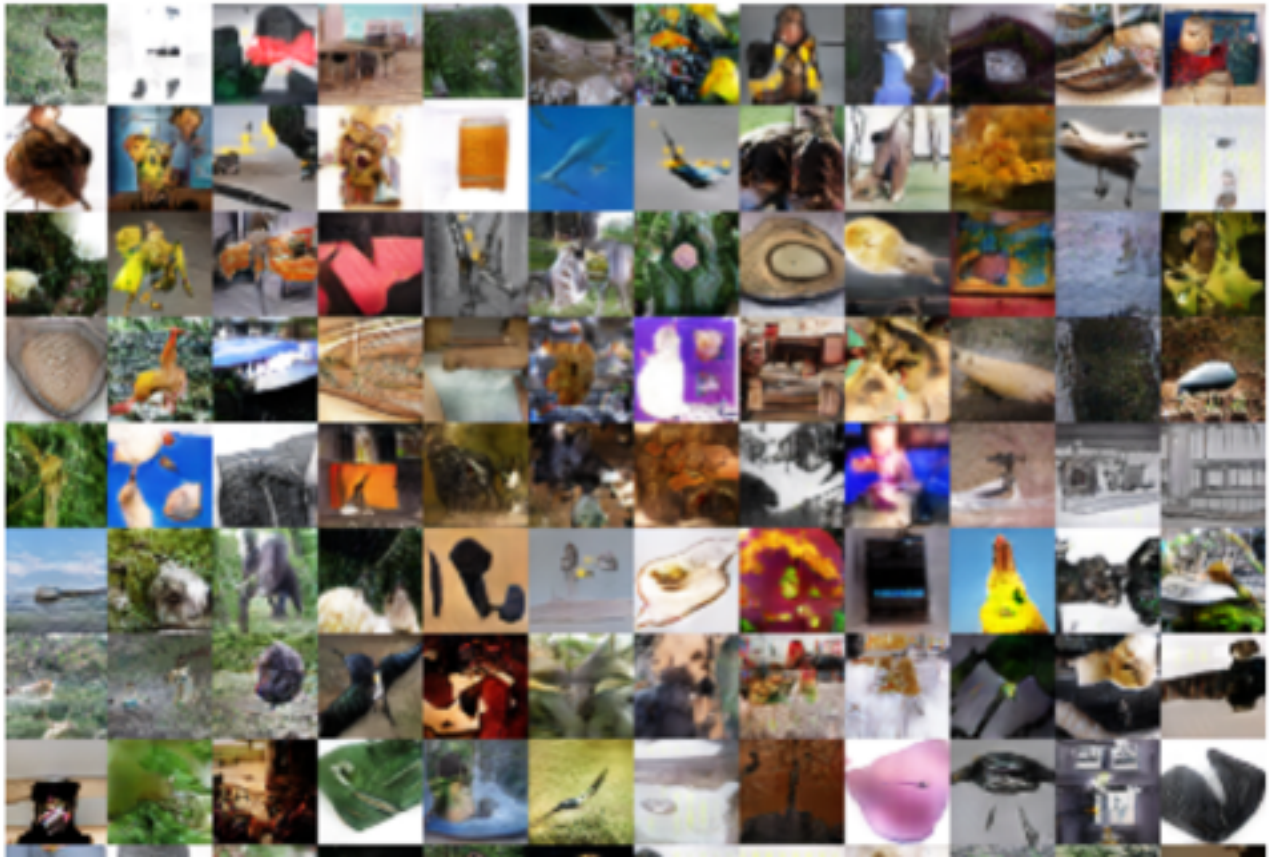



- Semantic Segmentation [Pinhero, Collobert, Dollar, ICCV'15]

*figures from Yann LeCun's CVPR'15 plenary*

# Is it just for a particular task?

- No. CNN architectures also obtain state-of-the-art performance on other tasks:



  - Generative Models for Natural Images [Radford, Metz & Chintala,'15]

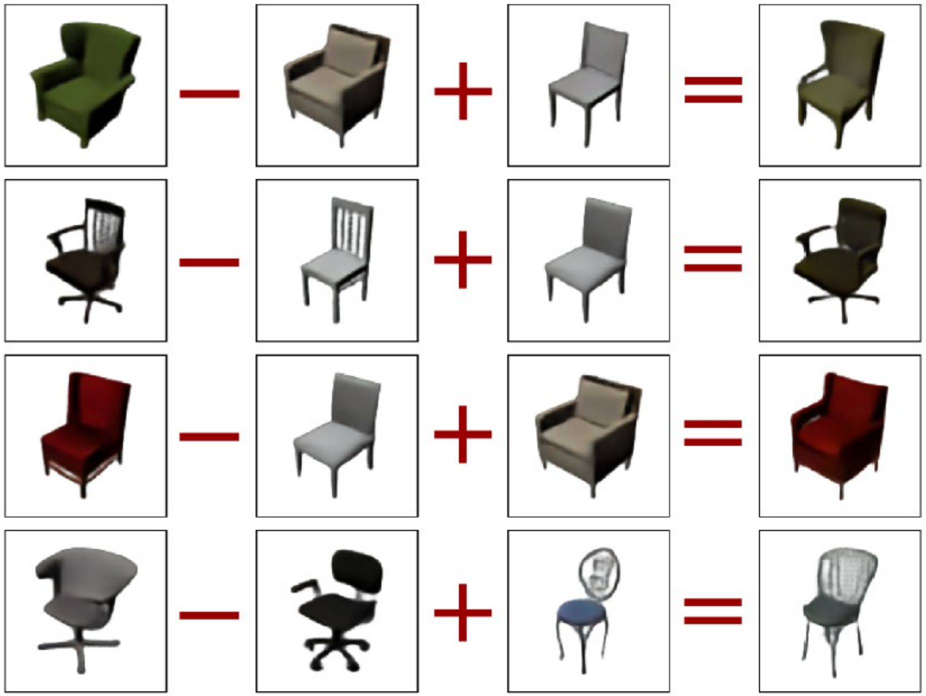
# Is it just for a particular task?

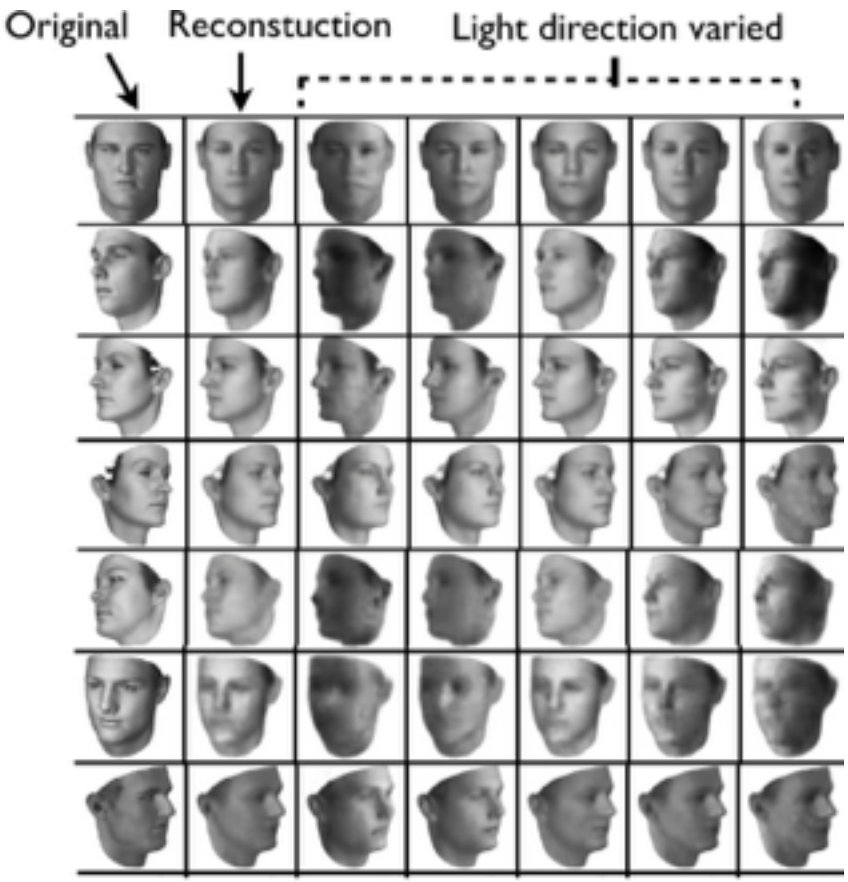
- No. CNN architectures also obtain state-of-the-art performance on other tasks:

  - Generative Models for Natural Images [Radford, Metz & Chintala,'15]

# Is it just for a particular task?

- No. CNN architectures also obtain state-of-the-art performance on other tasks:





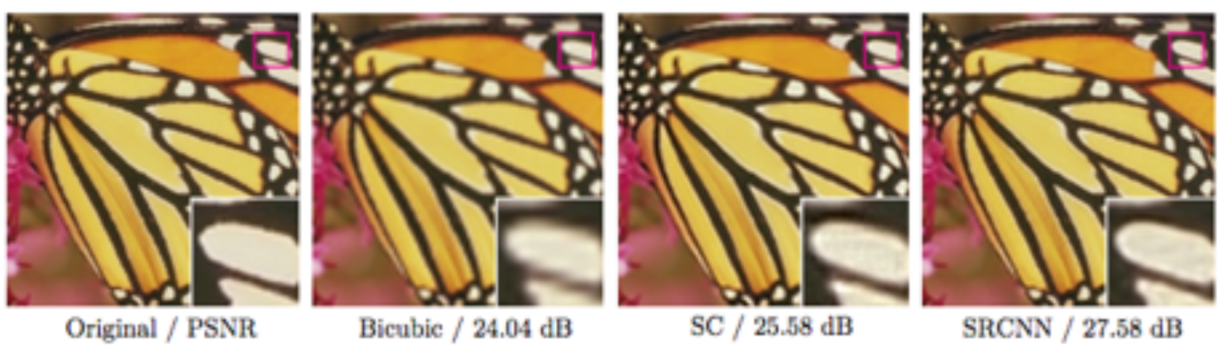

- Related work [Kulkarni et al'15, Dosovitsky et al '14]

# Is it just for a particular task?

- No. CNN architectures also obtain state-of-the-art performance on other tasks:


  - Image Captioning [Vinyals et al'14, Karpathy et al '14, etc]

  - Optical Flow estimation [Zontar '15]




  - Im

- Convolutional Deep Learning models thus appear to capture high level image properties more efficiently than previous models.
    - Highly Expressive Representations capturing complex geometrical and statistical patterns.
    - Excellent generalization: "beating" the curse of dimensionality

- Convolutional Deep Learning models thus appear to capture high level image properties more efficiently than previous models.

- Which architectural choices might explain this advantage mathematically?
    - Role of non-linearities?
    - Role of convolutions?
    - Role of depth?
    - Interplay with geometrical, class-specific invariants?

- Convolutional Deep Learning models thus appear to capture high level image properties more efficiently than previous models.

- Which architectural choices might explain this advantage mathematically?

- Which optimization choices might explain this advantage?
  - Presence of local minima or saddle points?
  - Equivalence of local solutions?
  - Role of Stochastic optimization?

# Deep Learning Approximation Theory

- Deep Networks define a class of "universal approximators": Cybenko and Hornik characterization:

**Theorem** [C'89, H'91] Let $\rho()$ be a bounded, non-constant continuous function. Let $I_m$ denote the $m$-dimensional hypercube, and $C(I_m)$ denote the space of continuous functions on $I_m$. Given any $f \in C(I_m)$ and $\epsilon > 0$, there exists $N > 0$ and $v_i, w_i, b_i$, $i = 1 \ldots, N$ such that

$$F(x) = \sum_{i \leq N} v_i \rho(w_i^T x + b_i) \text{ satisfies}$$

$$\sup_{x \in I_m} |f(x) - F(x)| < \epsilon \ .$$

# Deep Learning Approximation Theory

- Deep Networks define a class of "universal approximators": Cybenko and Hornik characterization:

**Theorem** [C'89, H'91] Let $\rho()$ be a bounded, non-constant continuous function. Let $I_m$ denote the $m$-dimensional hypercube, and $C(I_m)$ denote the space of continuous functions on $I_m$. Given any $f \in C(I_m)$ and $\epsilon > 0$, there exists $N > 0$ and $v_i, w_i, b_i$, $i = 1 \ldots, N$ such that

$$F(x) = \sum_{i \leq N} v_i \rho(w_i^T x + b_i) \quad \text{satisfies}$$

$$\sup_{x \in I_m} |f(x) - F(x)| < \epsilon \ .$$

- It guarantees that even a single hidden-layer network can represent any classification problem in which the boundary is locally linear (smooth).
- It does not inform us about good/bad architectures.
- Or how they relate to the optimization.

# Deep Learning Estimation Theory

**Theorem** [Barron'92] The mean integrated square error between the estimated network $\hat{F}$ and the target function $f$ is bounded by

$$O\left(\frac{C_f^2}{N}\right) + O\left(\frac{Nm}{K}\log K\right) ,$$

where $K$ is the number of training points, $N$ is the number of neurons, $m$ is the input dimension, and $C_f$ measures the global smoothness of $f$.

# Deep Learning Estimation Theory

**Theorem** [Barron'92] The mean integrated square error between the estimated network $\hat{F}$ and the target function $f$ is bounded by

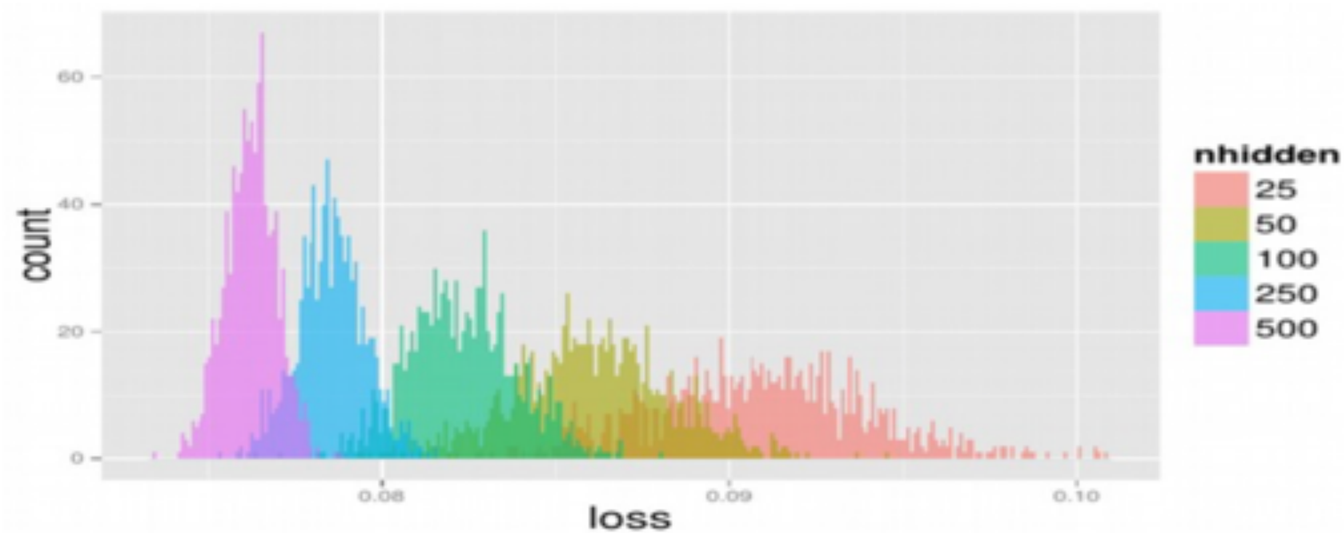
$$O\left(\frac{C_f^2}{N}\right) + O\left(\frac{Nm}{K}\log K\right) ,$$

where $K$ is the number of training points, $N$ is the number of neurons, $m$ is the input dimension, and $C_f$ measures the global smoothness of $f$.

- Combines approximation and estimation error.
- Does not explain why online/stochastic optimization works better than batch normalization.
- Does not relate generalization error with choice of architecture.

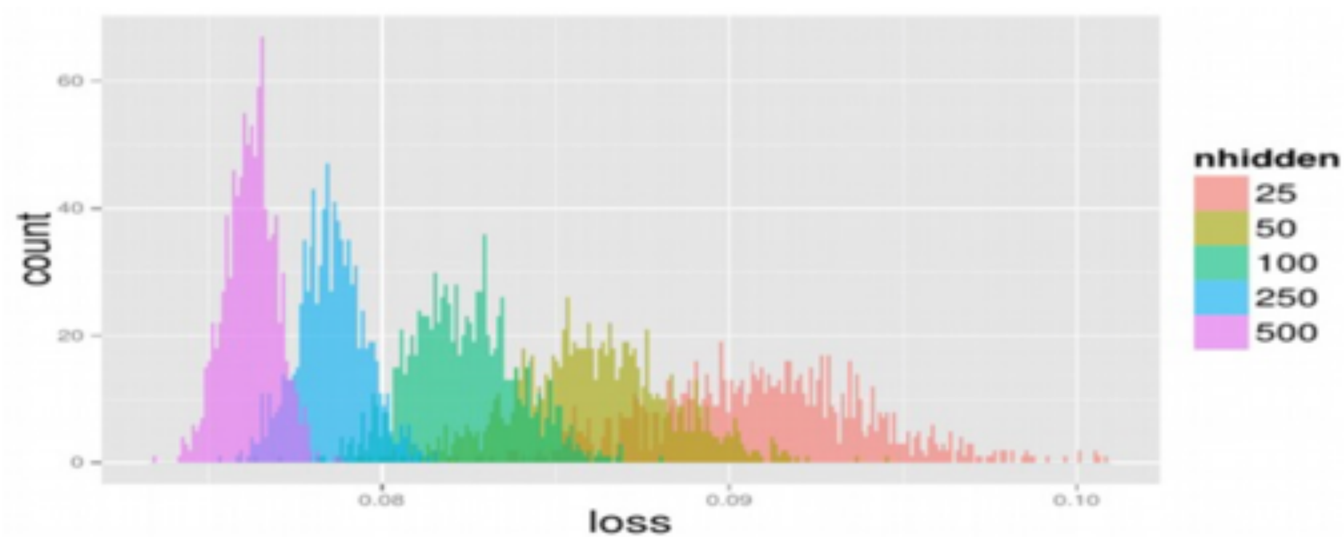
# Non-Convex Optimization

- [Choromaska et al, AISTATS'15] (also [Dauphin et al, ICML'15] ) use tools from Statistical Physics to explain the behavior of stochastic gradient methods when training deep neural networks.



# Non-Convex Optimization

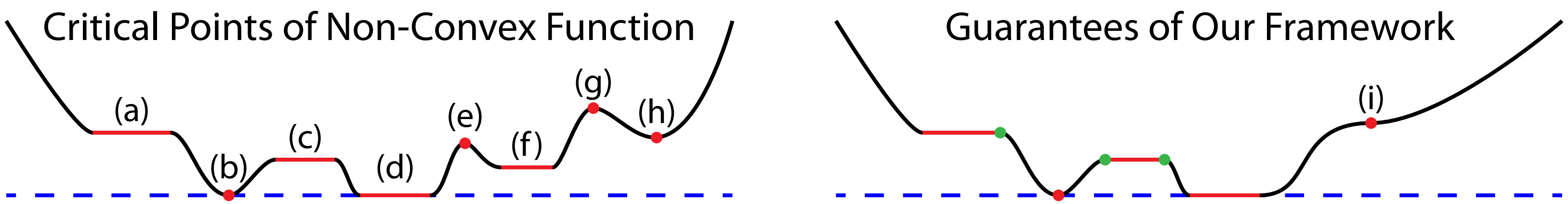
- [Choromaska et al, AISTATS'15] (also [Dauphin et al, ICML'15] ) use tools from Statistical Physics to explain the behavior of stochastic gradient methods when training deep neural networks.



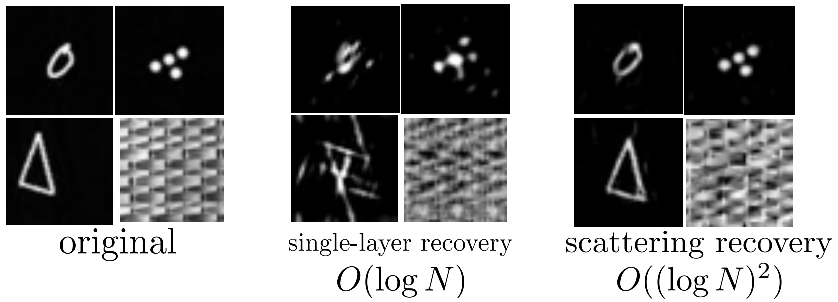

- Offers a macroscopic explanation of why SGD "works".
- Gives a characterization of the network depth.
- Strong model simplifications, no convolutional specification.

# Tutorial Outline

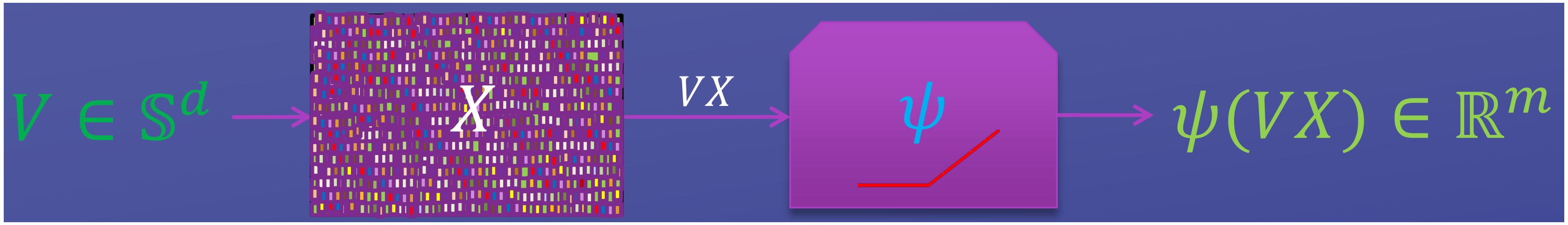
- **Part I**: Global Optimality in Deep Learning (René Vidal)




- **Part II**: Signal Recovery from Scattering Convolutional Networks (Joan Bruna)



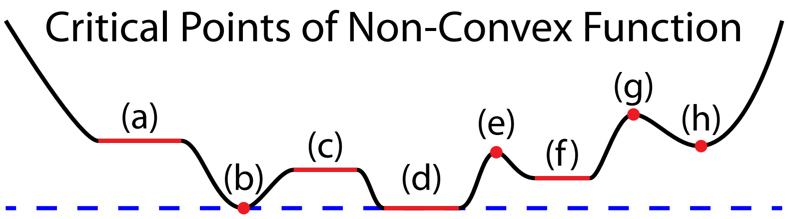

- **Part III**: On the Stability of Deep Networks (Raja Giryes)

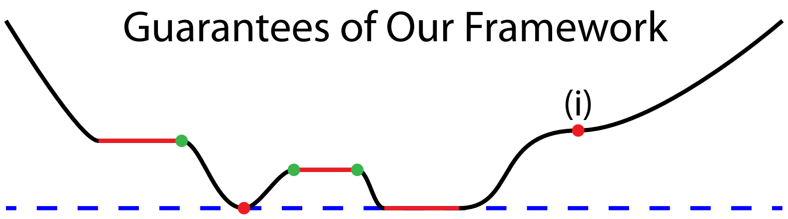


# Part I: Global Optimality in Deep Learning

- **Key Questions**
  - How to deal with the non-convexity of the learning problem?
  - Do learning methods get trapped in local minima?
  - Why many local solutions seem to give about equally good results?
  - Why using rectified linear rectified units instead of other nonlinearities?

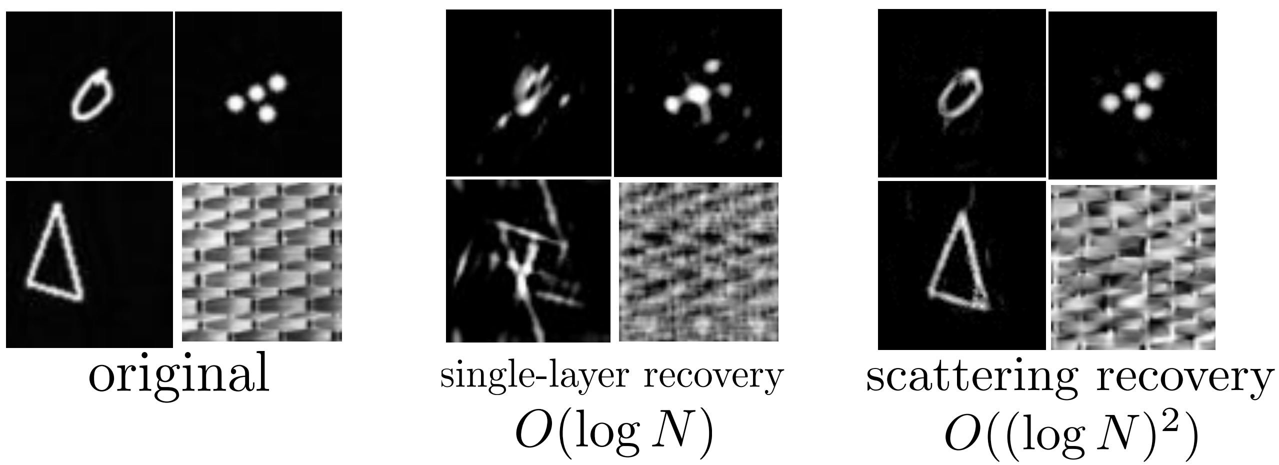
- **Key Results**
  - Deep learning is a positively homogeneous factorization problem
  - With proper regularization, local minima are global
  - If network large enough, global minima can be found by local descent



# Part II: Scattering Convolutional Networks

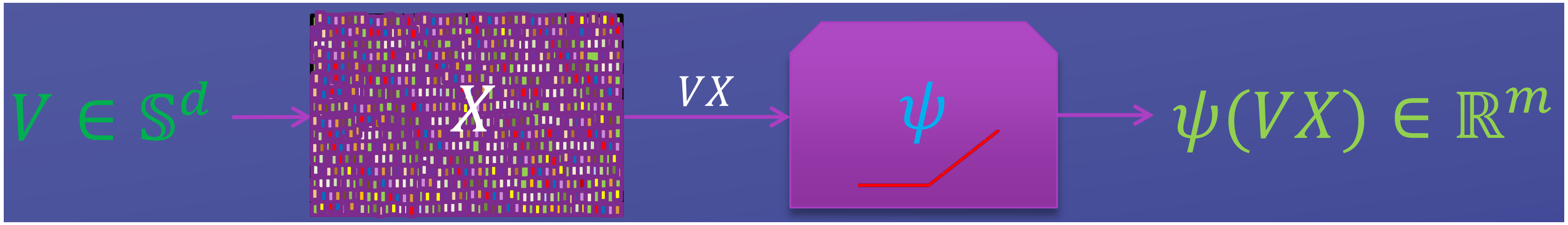
- **Key Questions**
  - What is the importance of "deep" and "convolutional" in CNN architectures?
  - What statistical properties of images are being captured/exploited by deep networks?




- **Key Results**
  - Scattering coefficients are stable encodings of geometry and texture
  - Layers in a CNN encode complex, class-specific geometry.

# Part III: On the Stability of Deep Networks

- **Key Questions**
  - **Stability:** Do small perturbations to the input image cause small perturbations to the output of the network?
  - Can I recover the input from the output?




- **Key Results**
  - Gaussian mean width is a good measure of data complexity.
  - DNN keep important information of the data.
  - Deep learning can be viewed as metric learning problem.

# Tutorial Schedule

- 14:00-14.30: Introduction
- 14:30-15.15: Global Optimality in Deep Learning (René Vidal)
- 15:15-16.00: Coffee Break
- 16:00-16:45: Scattering Convolutional Networks (Joan Bruna)
- 16:45-17:30: Stability of Deep Networks (Raja Giryes)
- 17.30-18:00: Questions and Discussion