Introduction to Course

EE5179: Deep learning for imaging

Instructor: Kaushik Mitra

Introduction to Machine Learning

Goal: Learning from the data with minimal intervention from the user

Supervised learning:

- Learns a mapping b/w *input* and *output* pairs (x_i, y_i) e.g. image classification

f(x) → Coffee mug

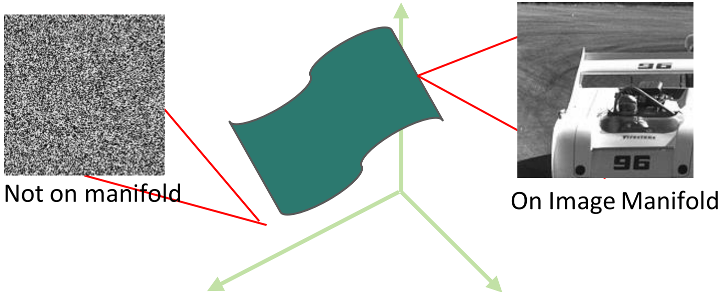
f(x) → tiger

f(x) → car

- *Applications*:
  - ❑ classification: object/scene/face
  - ❑ Regression: object detection, depth

Unsupervised learning:

- Given only data '*x*' learn the inherent underlying structure
- Consider a 64x64 binary image



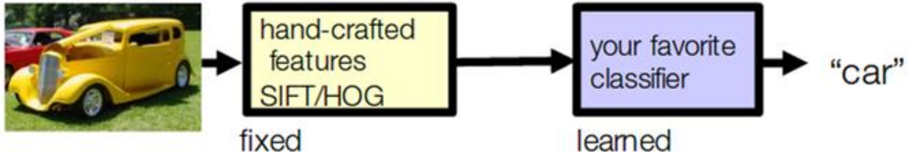
- *Applications*: clustering, dimensionality reduction, density estimation

Traditional Machine learning tasks

Supervised learning: Image classification

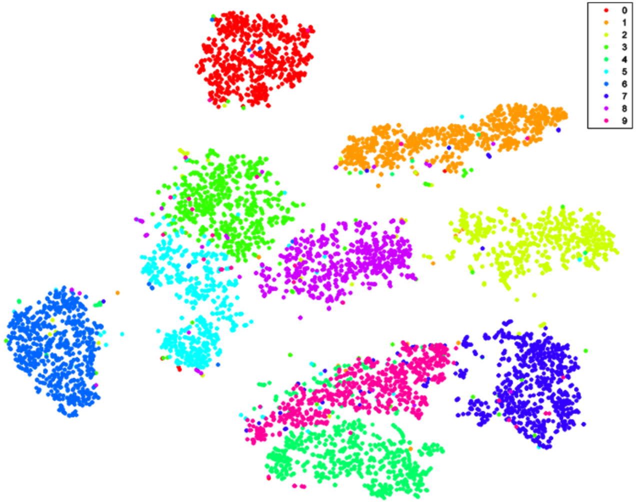
Intra-class variance
(coffee mug)

Inter-class variance
(ImageNet)



*pic courtesy, Yoshua Bengio and Yahn Lecun

Unsupervised learning: Manifold learning

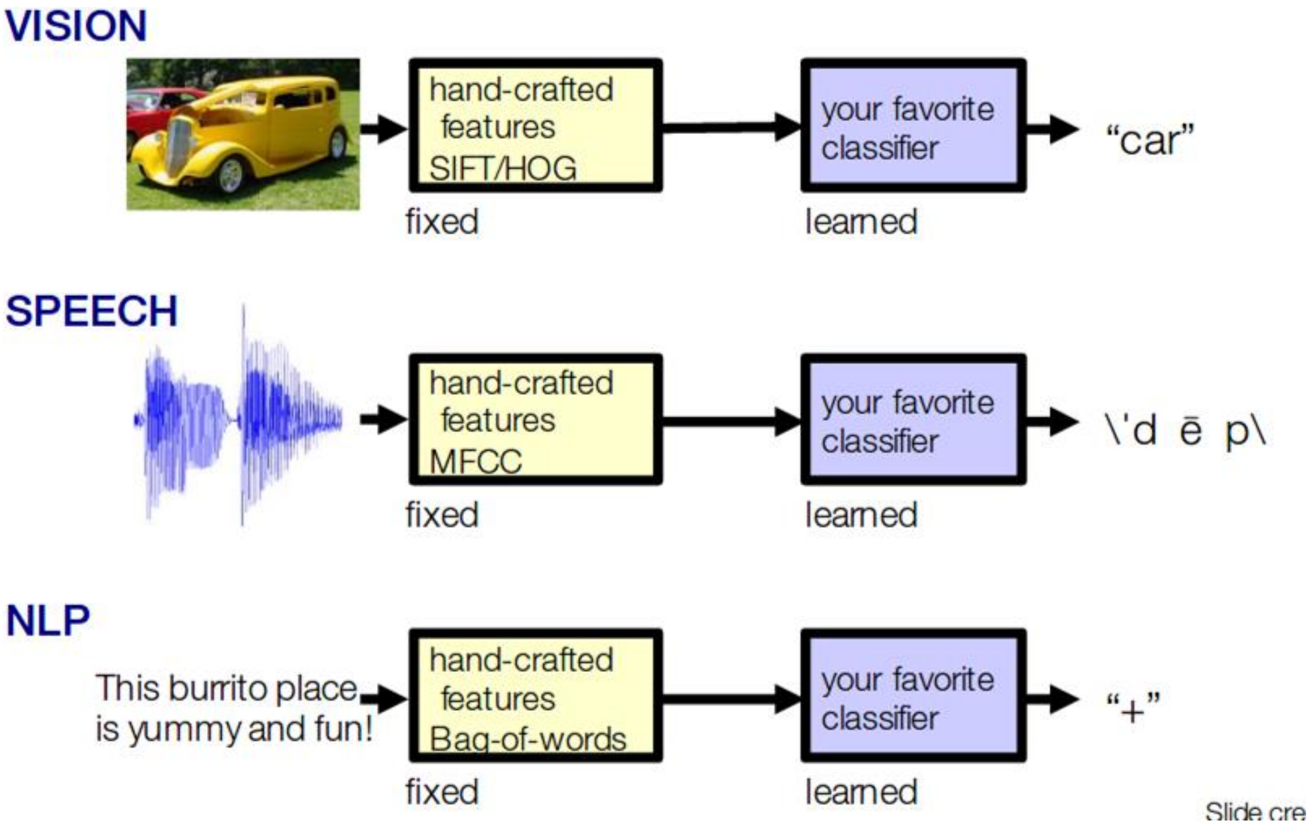


*pic courtesy, G. Hinton's course cs2535 lec11

t-SNE (unsupervised & nonlinear)
embedding of MNIST digits

Traditional approaches

- Manual feature extraction (SIFT/HOG)
- Classifier is learned independent of feature extraction

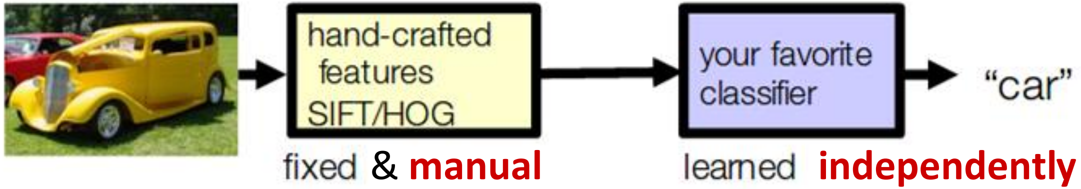


*slide courtesy, Yoshua Bengio and Yahn Lecun

Traditional approaches vs Deep learning

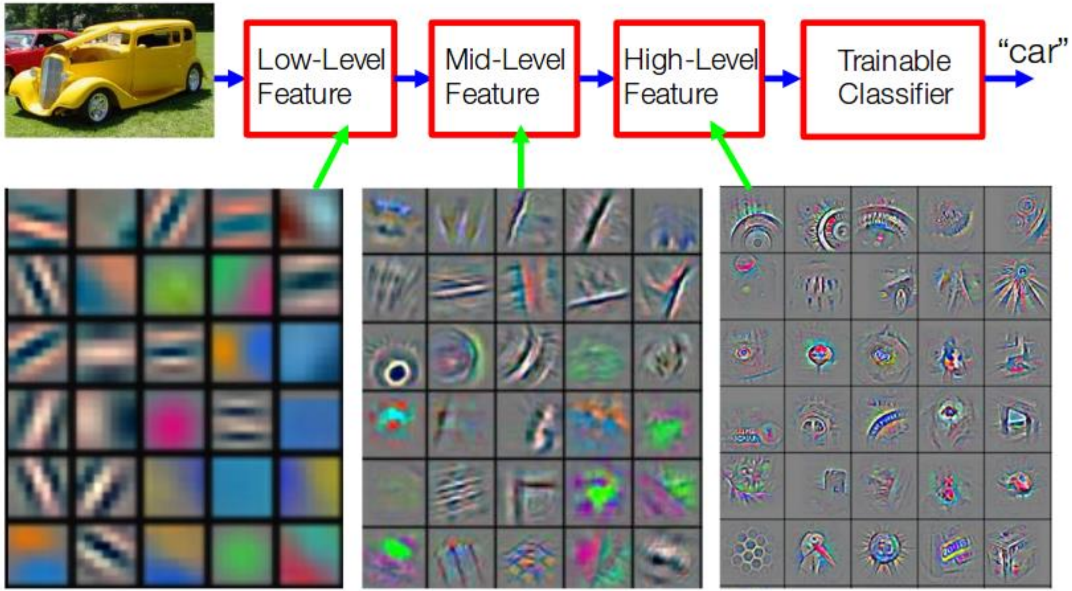
What's wrong with the traditional approaches?

- Compositional feature abstraction is missing



3 key ideas of deep learning

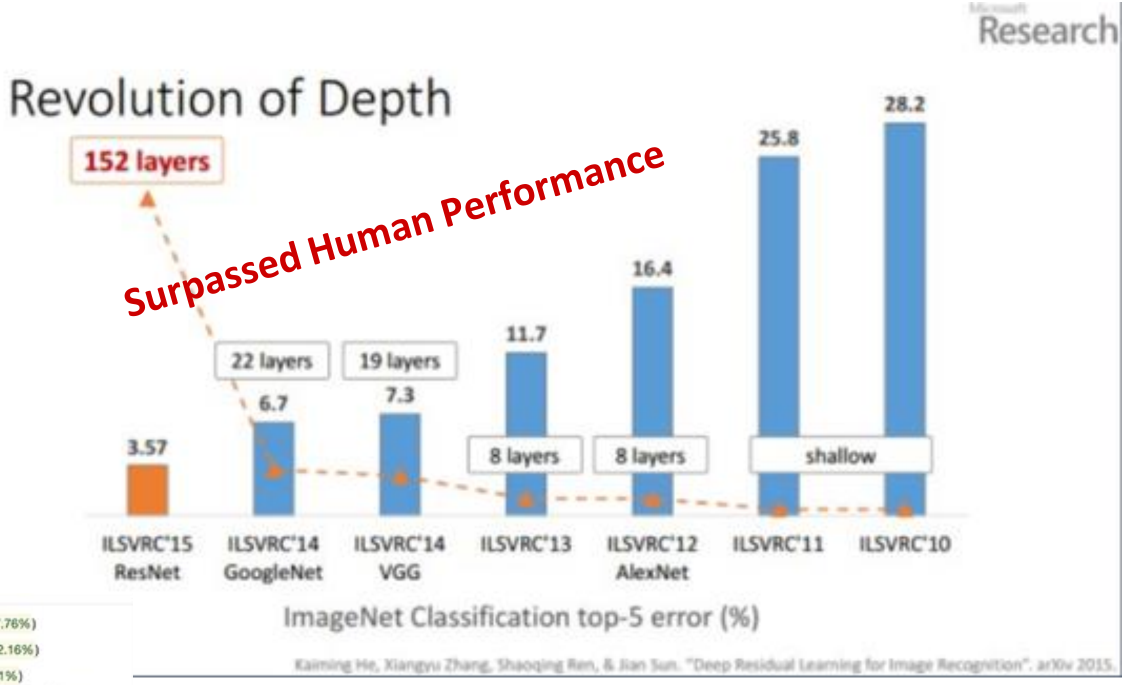
- (Hierarchical) Compositionality
  - Cascade of nonlinear functions
  - Multiple layers of abstractions
- End-to-End learning
  - Learning (task-driven) representations
  - Learning to extract features
- Distributed Representation
  - No single neuron encodes everything
  - Group of neurons work together



*slide courtesy, Yoshua Bengio and Yahn Lecun

Image Classification

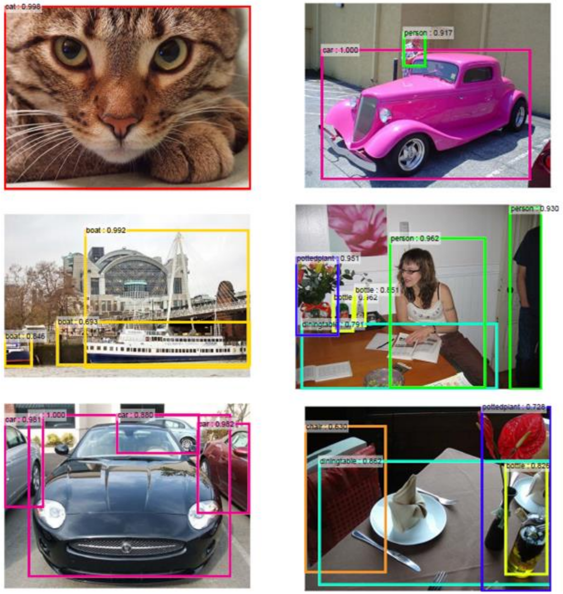
LeNet by Lecun et al. 1998 (MNIST)
AlexNet by Krish et al. NIPS’ 12
VGGNet by Simonyan et al. ICLR’ 15
GoogLeNet by Szegedy et al. CVPR’ 15
ResNet by He et al., CVPR’ 17 best paper



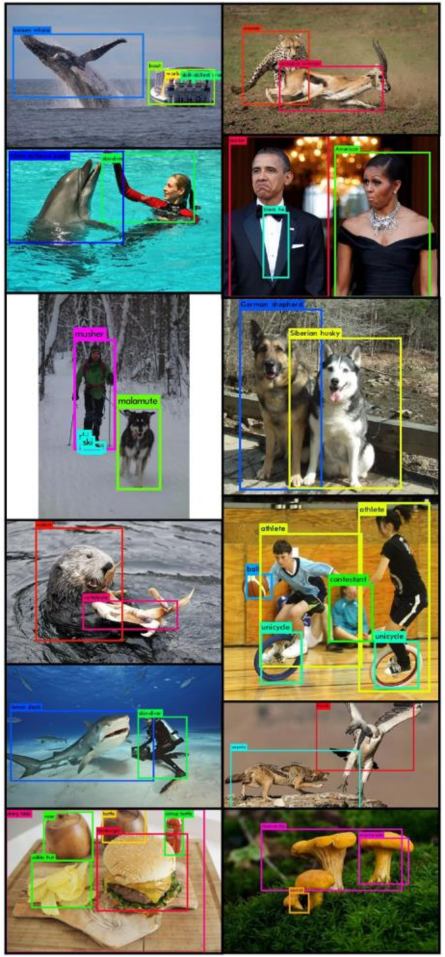
*pic courtesy: Kaiming He

Object Detection

- OverFeat by Sermanet et al., ICLR’ 14 (NYU)
- R-CNN by Girshick et al., CVPR’ 14 (UCB)
- SPP by He et al., ECCV’ 14 (MSR)
- Fast R-CNN by Girshick et al., arxiv (MSR)
- Faster R-CNN by Ren et al., NIPS’ 15 (MSR)
- YOLO by Redmon et al., arxiv 2015
- YOLO 9000 by Redmon et al., CVPR’ 17



Faster RCNN detections

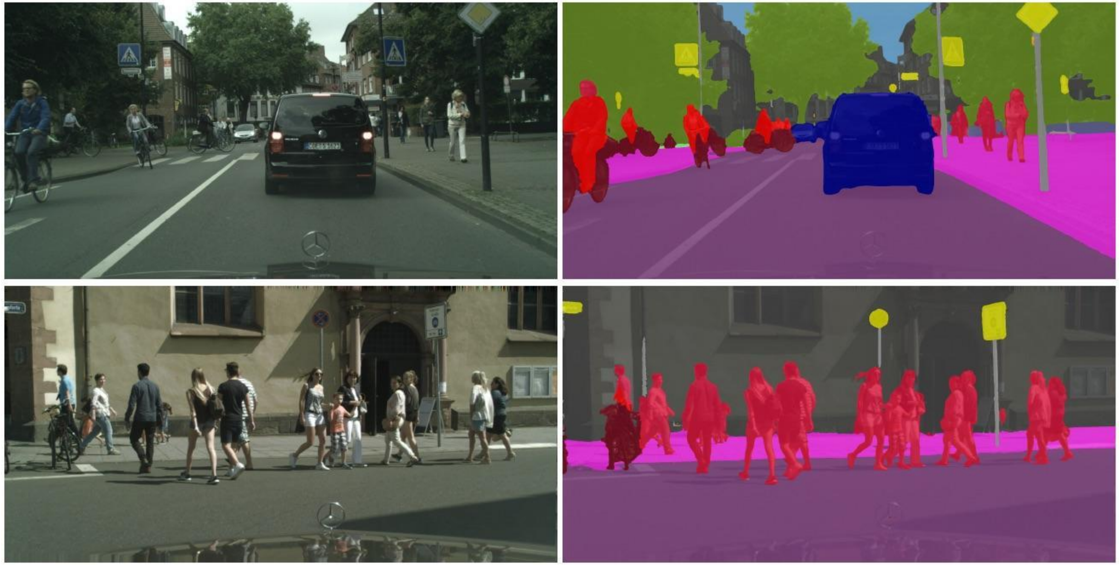


YOLO 9000 detections

pic courtesy: http://mrulafi.blogspot.in/2016_05_01_archive.html

Image Segmentation

FCN by Long et al. CVPR' 15
DeepLab by Chen et al. arxiv 2015
CRFS as RNNs by Zheng et al. ICCV' 15



*Pic courtesy: Kundu et al. CVPR 2016 on City scapes dataset

Style transfer

Neural style transfer by Gatys et al., CVPR’ 16

Deep Photo Style Transfer by Luan et al., CVPR’ 17

Actual | **Style** | **Actual with style**

Artistic

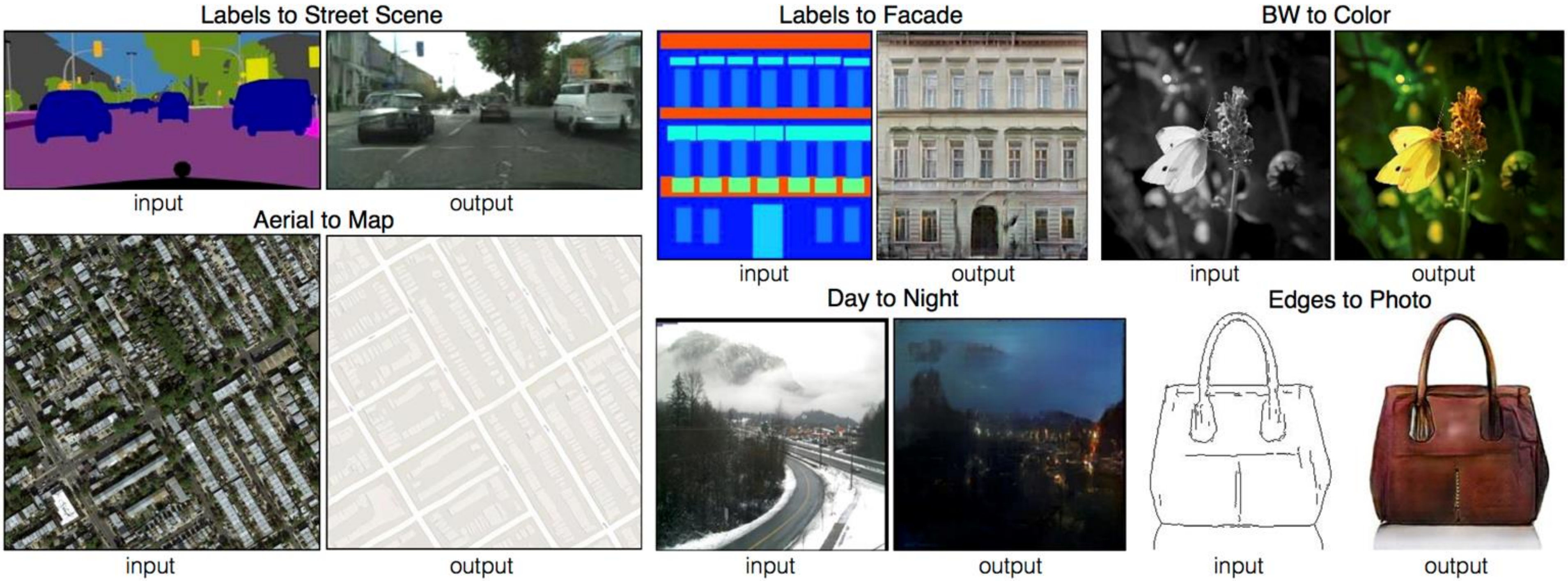
*Pic courtesy: http://deepart.io

Photo Realistic

*Pic courtesy: Deep Photo Style Transfer, Luan et al. CVPR 2017

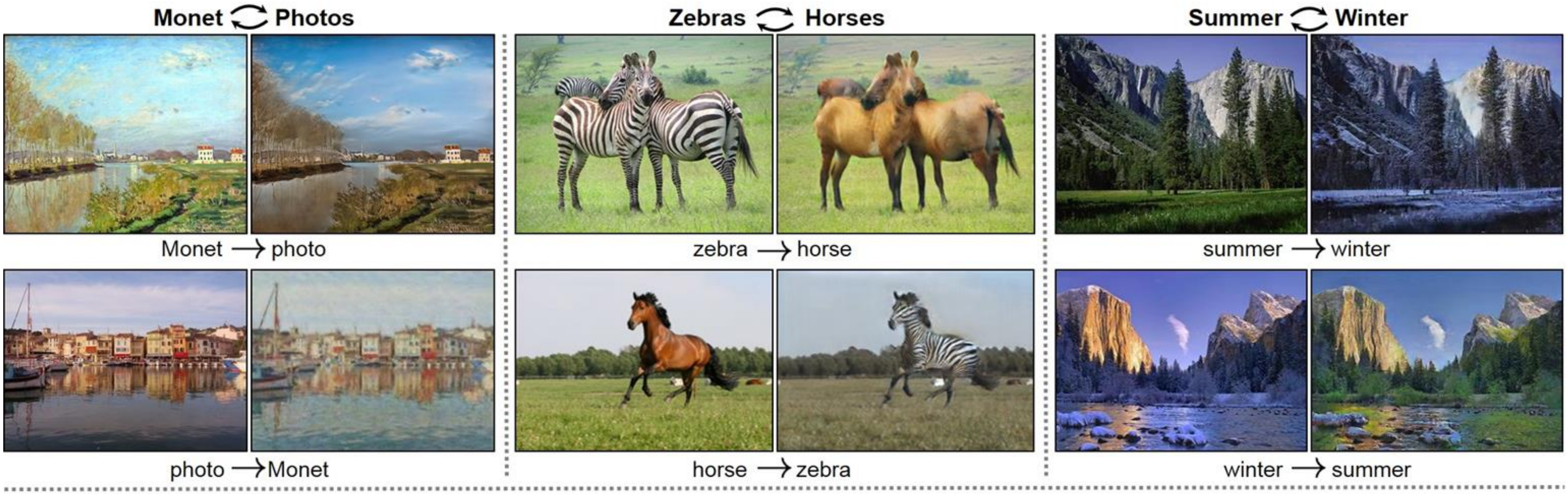
Artistic applications

Image-to-Image Translation with Conditional Adversarial Nets by Isola et al., CVPR’ 17



Artistic applications

Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks, arxiv 2017



Image, Video and Audio Generation

PixelRNN/CNN by Gregor et al.,
ICML' 16 Best paper (Samples from ImageNet)

DCGAN Chintala et al. ICLR' 16
Sample bedroom images

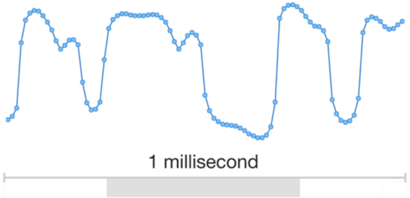
Generating **videos** with scene dynamics, by Vondrick et al., NIPS' 16

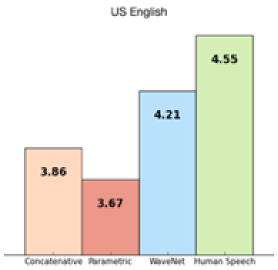
Ocean

Golf

Trains

WaveNet for **audio** synthesis by Oord et al. 2016, Deepmind

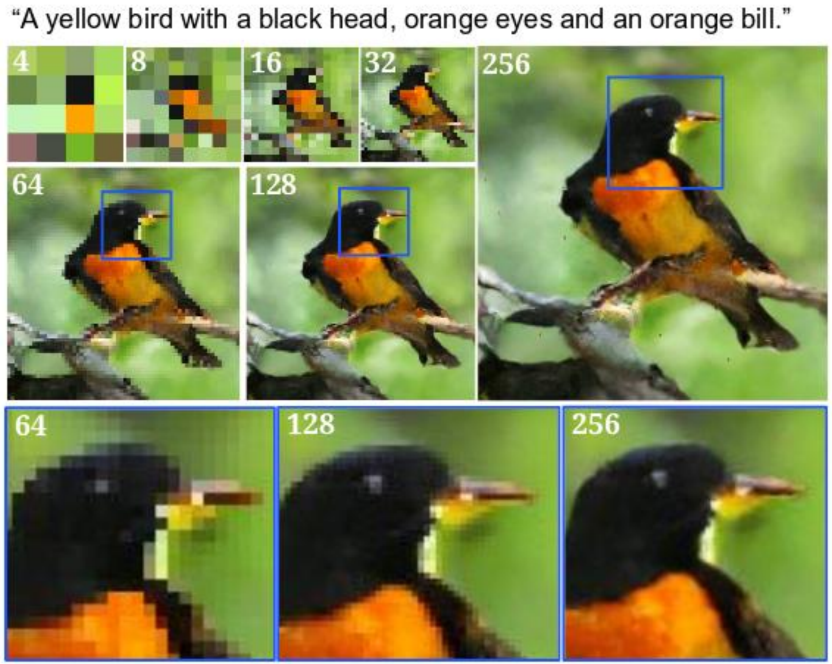




Check out wavenet's generated music
piano clips

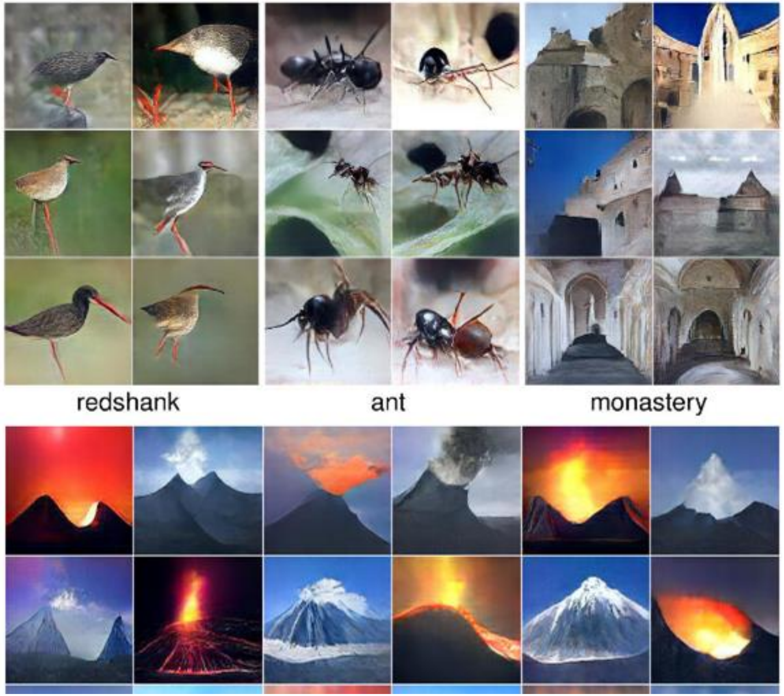
Conditional Generation

Bird image generated from CUB(bird) dataset conditioned on **text** and **bird parts** location



Parallel multiscale autoregressive density estimation by Reed et al. 2016, Deepmind.

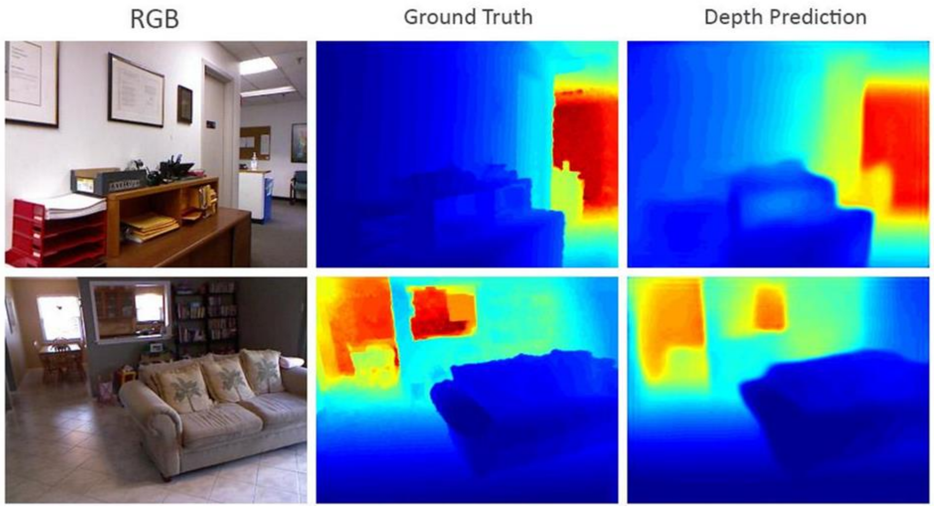
Class conditional image generation by Nguyen et al., CVPR’ 17 (samples from ImageNet)



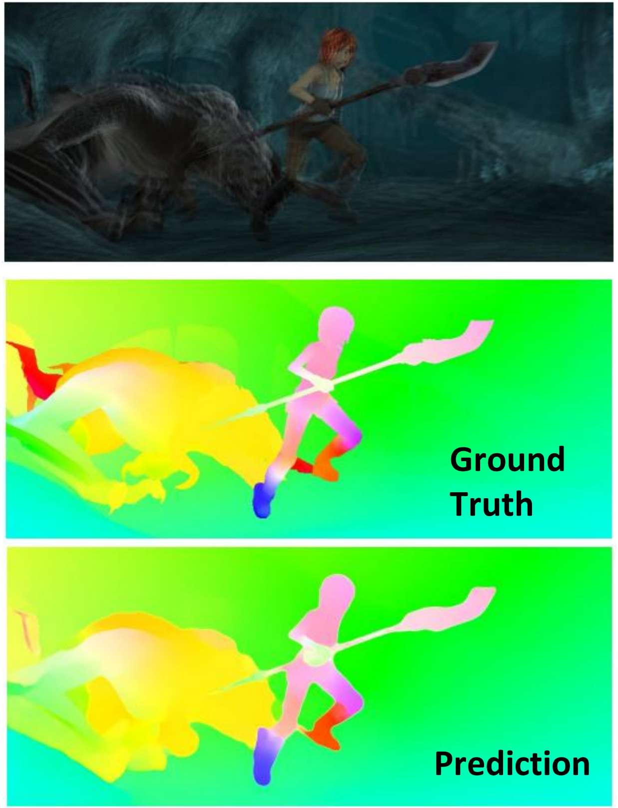
Other Vision problems

Depth from single image, Optical flow estimation

Depth prediction with Fully convolutional ResNets by Laina et al., 3DV' 16



FlowNet 2.0, Optical Flow Estimation with Deep Networks by Ilg et al. 2016, arxiv



Other Vision problems

Structure from Motion (SfM)

DeMon net by Ummenhofer et al., CVPR' 17

SfM-Net by Vijayanarasimhan et al., arxiv 2017

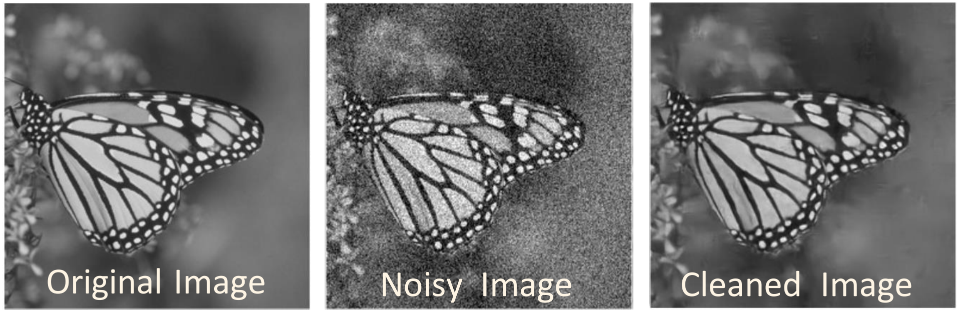
Depth and Motion Network for Learning Monocular Stereo by Ummenhofer et al., CVPR' 17

Image 1 **Image 2** **Estimated depth**

Estimated point cloud **Corresponding ground truth**

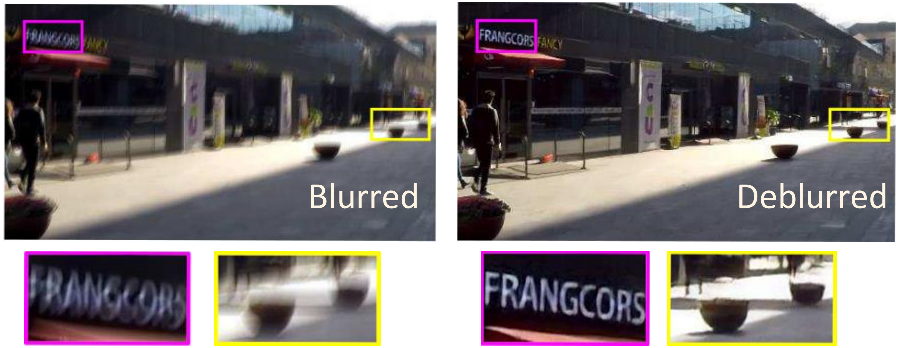
Deep Learning for Image Processing

Image Denoising



Deep CNN autoencoder for image restoration
by Mao et al. NIPS 16

Image Deblurring



Dynamic motion deblurring by Nah S et al., CVPR 17

Image Super-resolution

Photo-realistic
super-resolution
by Ledig et al. CVPR 17

Deep Learning for Imaging

Course Contents

1. Introduction to Neural Networks - **(5-6 classes)**
2. Convolutional Neural Networks (CNNs) - **(5-6 classes)**
3. Autoencoders (AE) - **(5-6 classes)**
4. Recurrent Neural Networks (RNNs) - **(5-6 classes)**
5. Deep Generative Models - **(5-6 classes)**
6. Generative Adversarial Nets (GANs) - **(5-6 classes)**

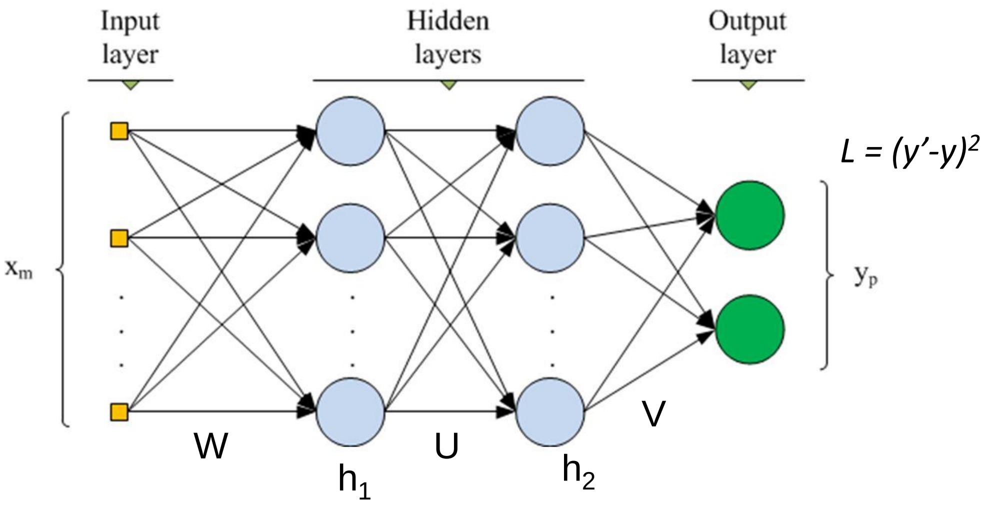
1. Basic Neural Networks

Multi-layer perceptrons (MLP) is a feed forward neural network with hidden layers

$$h_1 = f(Wx + b_1), \; f \text{ is an activation function}$$

$$h_2 = g(Uh_1 + b_2)$$

$$y' = Vh_2 + b_3$$



Hidden layers *increase* abstraction

- hence, better to have more hidden layers than a single layer with large number of neurons

Universal Approximation Theorem:

- simple neural networks can *represent* a wide variety of interesting functions when given appropriate parameters

*pic courtesy, Sancho course

1. Basic Neural Networks

What we will learn in the course about Neural Networks?

Introduction

- History
- Biological motivation
- McCulloch and Pits model
- Rosenblatt's perceptron

Perceptrons

- Geometry and linear separability
- XoR problem
- Multi-layer perceptron (MLP)
- Universal approximation theorem

Training MLPs

- Error back propagation
- Error surface geometry
- Stochastic Gradient Descent (SGD)
- Online vs Batch modes
- Loss functions

Optimization issues

- Over fitting / Under fitting
- Convergence
- Regularization

Optimization methods

- SGD with momentum
- Nesterov's accelerated gradient descent (NAG)
- ADAM, ADAGrad, ADADelta
- RMSProp

Applications

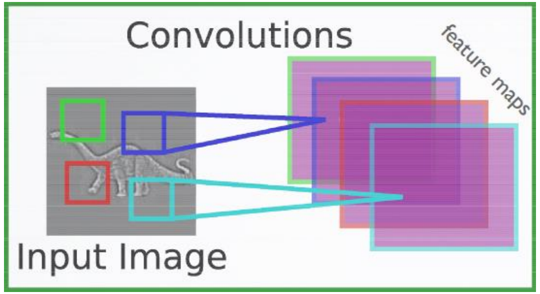
- MNIST digit classification
- Image denoising with MLPs

2. Convolutional Neural Networks (CNNs)

CNNs vs MLPs

- Naively using MLP to classify 224x224x3 (~ 3 x 40,000) typical ImageNet image -> parameter explosion
  - ❑ Doesn't exploit local spatial information (why?)
- Can we build special neural nets for images exploiting
  - ❑ 2D topology of pixels
  - ❑ Achieve invariance to translation, illumination ?

Category: tiger
ImageNet



Convolutional networks leverage these ideas,

- ❑ Local connectivity
- ❑ Parameter sharing
- ❑ Pooling/ Subsampling
- ❑ ReLu (rectifier) nonlinearity

*slide courtesy, Hugo Larochelle course on Neural networks

2. Convolutional Neural Networks (CNNs)

What we will learn in the course about CNNs??

Introduction

- Biological motivation
- Hand-coded to learnt filters
- CNNs over fully connected

Visualizing CNNs

- What features does CNN learn?
- Techniques for feature visualization
- Fooling CNNs with adversarial examples

Training/Optimization

- Backpropagation through different layers
  - ❑ Convolutional
  - ❑ Relu, Max pool, Average pool
- Regularization techniques
  - ❑ Batch Norm
  - ❑ Drop-out

State of the art architectures

- AlexNet and VGG Net
- GoogLeNet (inception module)
- ResNets (Residual module) CVPR’ 17 best paper

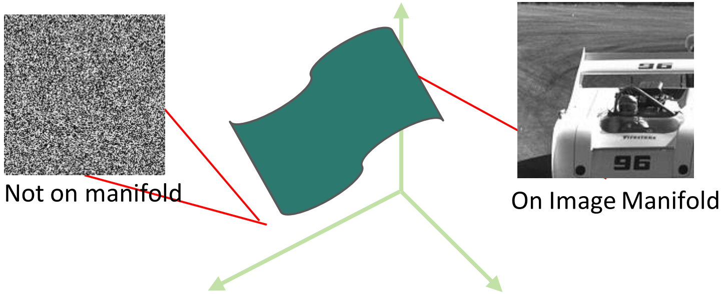
Applications

- Image Classification

3. Autoencoders (AE)

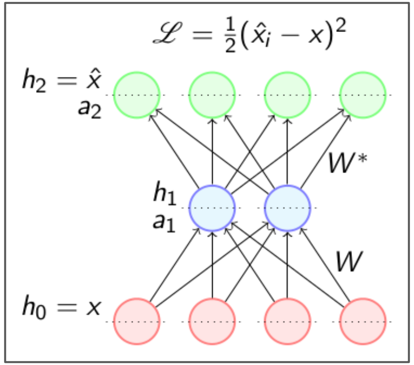
So, **What** are Autoencoders **???**

- A type of neural networks for unsupervised feature learning
- Used for dimensionality reduction
- With linear activations, AE does same job as PCA



Failure cases - without constraints, AE might learn trivial identity mapping (overfitting)

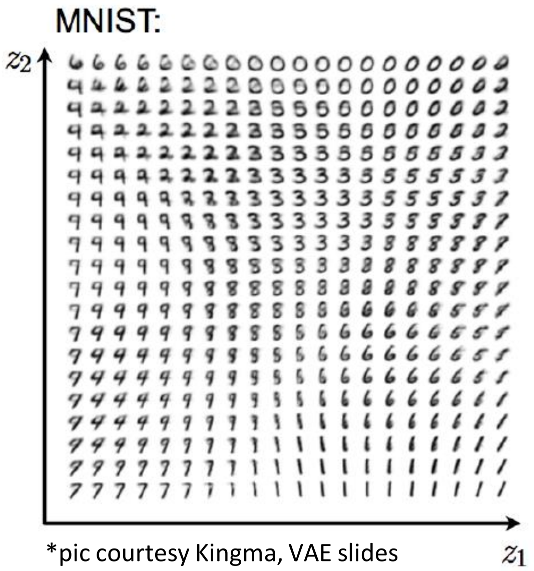
- Different sparsity constraints for regularization
- Prevents trivial identity learning



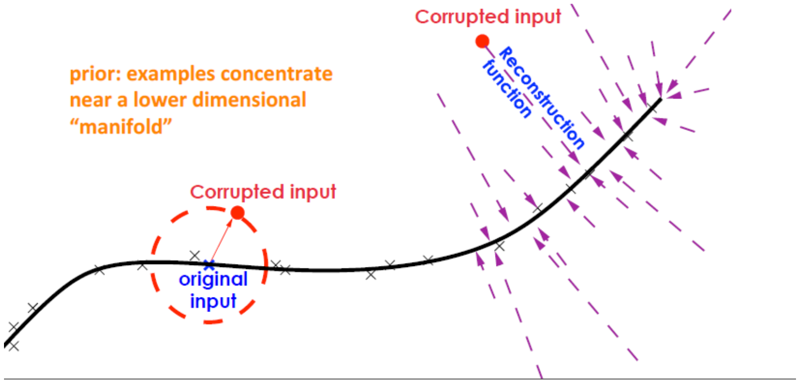
3. Autoencoders (AE)

What else? It can be used for **manifold** learning

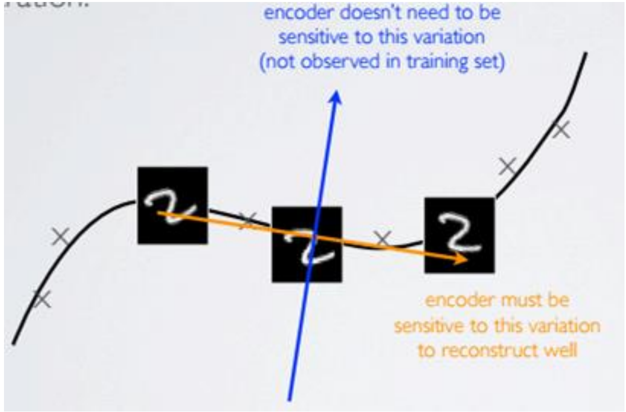
- Denoising Autoencoder, **DAE**
- Contractive Autoencoder, **CAE**



*pic courtesy Kingma, VAE slides



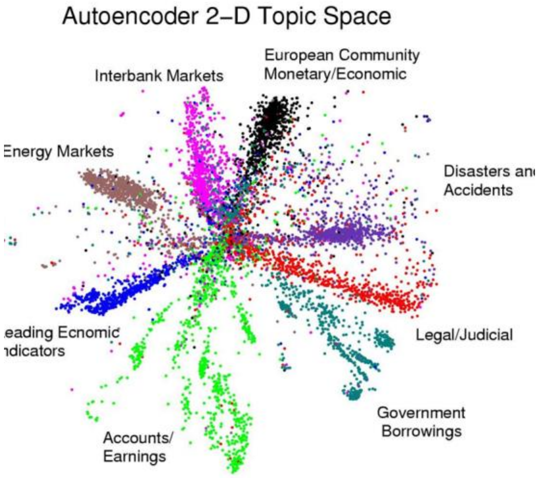
*pic courtesy Vincent pascal



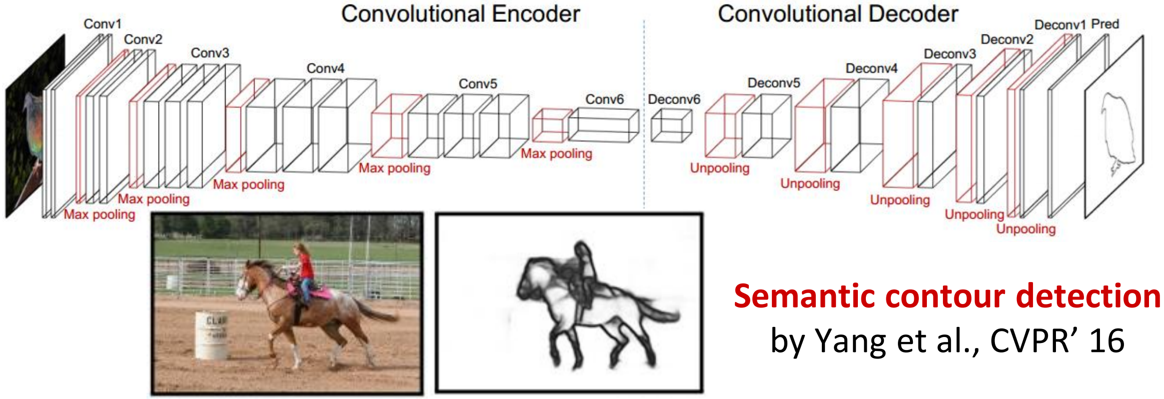
*pic courtesy Hugo larochelle's course

3. Autoencoders (AE)

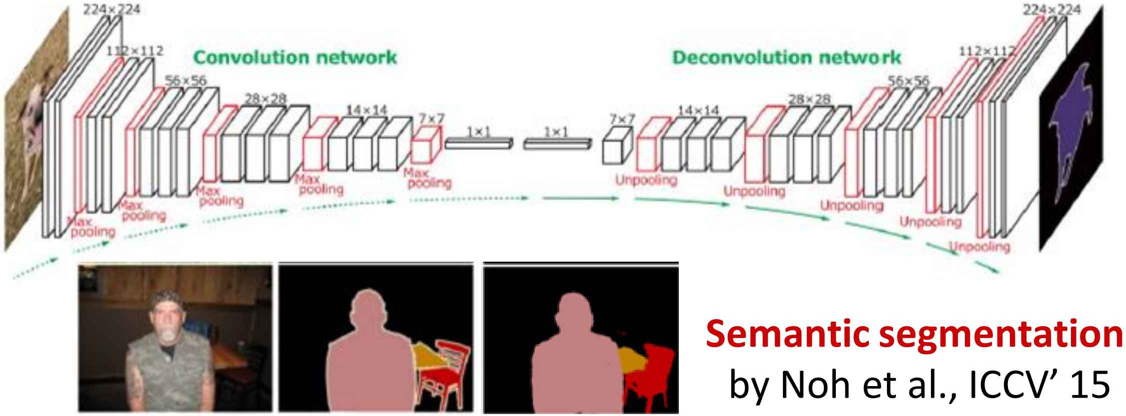
Applications



Document clustering, Hinton et al. 2006



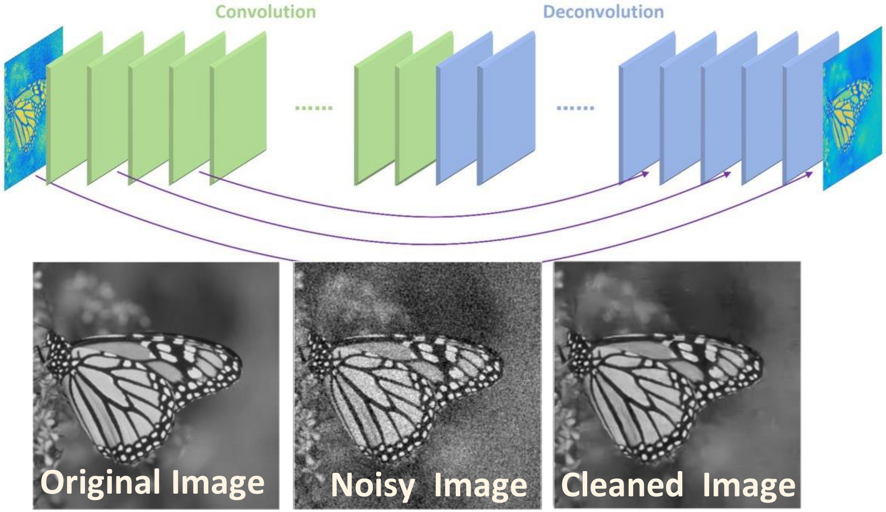
Semantic contour detection
by Yang et al., CVPR’ 16



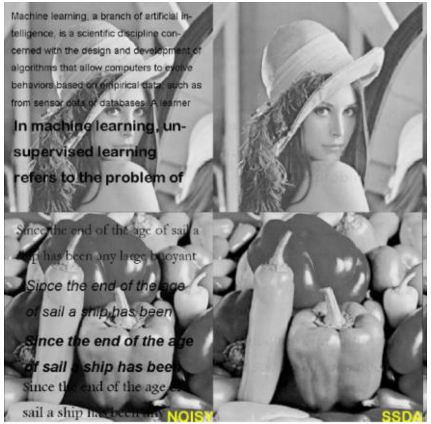
Semantic segmentation
by Noh et al., ICCV’ 15

3. Autoencoders (AE)

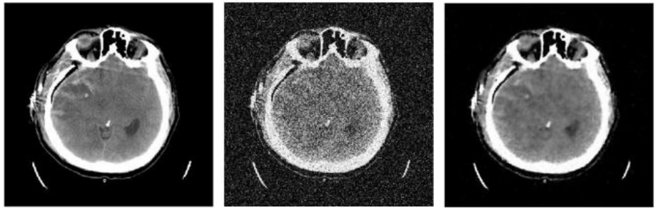
Applications in image restoration



Deep CNN autoencoder for image restoration
by Mao et al. NIPS 16



Text removal/inpainting with SSDA



original noisy AMC-SSDA

Image denoising by Forest et al., NIPS' 13

3. Autoencoders (AE)

What we will learn in the course about Autoencoders?

Introduction

- Dimensionality reduction
- Equivalence of AE and PCA

Applications

- Image restoration
  - ❑ Denoising
  - ❑ Inpainting
  - ❑ Super-resolution
- Semantic image segmentation
- Semantic contour detection

Regularization

- Sparse autoencoders
  - ❑ K-sparse activation
  - ❑ Sparse hidden representation
- Denoising Autoencoders
- Contractive Autoencoders

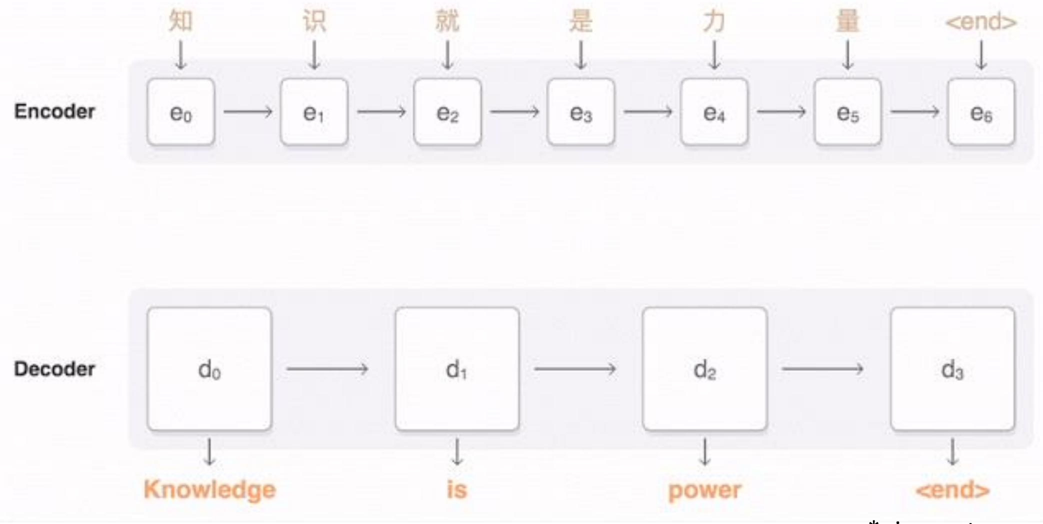
State of the art architectures

- Stacked sparse autoencoders
- Stacked denoising autoencoders
- Convolutional encoder-decoders

4. Recurrent Neural Networks

- RNNs have been developed to model sequences
- Many real life situations are sequences
  - ❑ Reading, writing, listening … Memory is crucial

Chinese - English Google translation



- Note that input output statements length can vary

*pic courtesy: [google research blog]

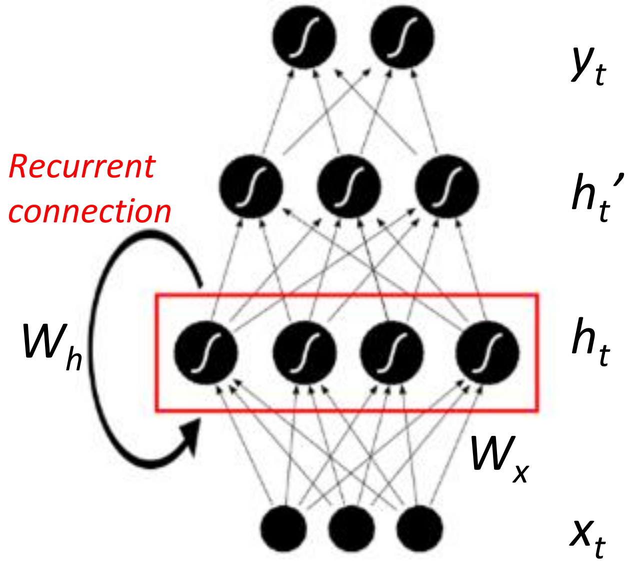
4. Recurrent Neural Networks

What are Recurrent Neural Nets **???**

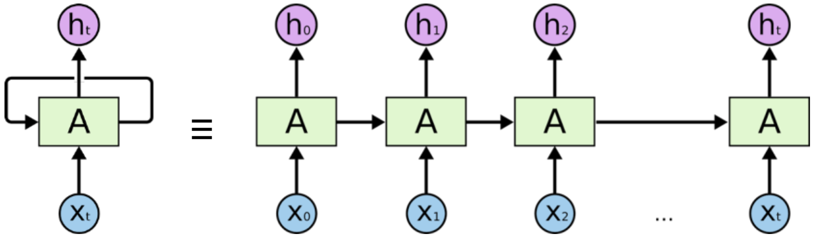
- Feedforward networks except that hidden layer (*h*) is connected to itself

$$h_t = f(\ W_h h_{t-1} + W_x x_t\ )$$

- Thus, recurrent connections build an internal representation of the past
- In effect, they add memory to the network



*pic courtesy Alex Graves

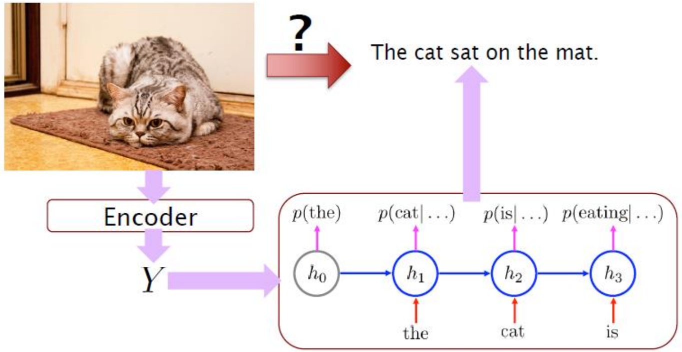


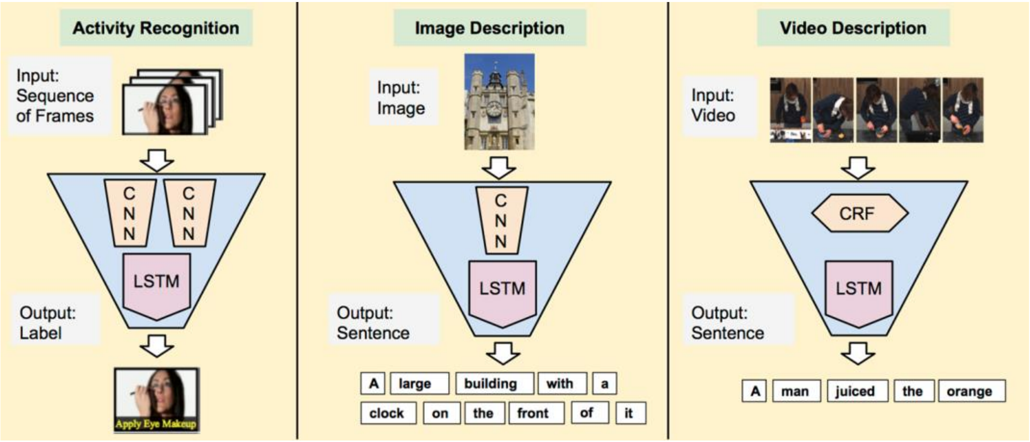
Unrolling RNN over time,
pic courtesy - [colah blog on lsmts]

4. Recurrent Neural Networks

Applications

Image captioning





*pic courtesy [jeffdonahue lrcn]

4. Recurrent Neural Networks

What's in the course?

Introduction

- Sequence modeling

Training RNNs

- Backpropagation Through Time (BPTT)
- Problems with BPTT
  - Gradient vanishing/ exploding
- Long Short Term Memory units (LSTMs) to the rescue

State of the art architectures

- Gated Recurrent Unit (GRU)
- CNN + LSTM for image captioning and action recognition

Applications

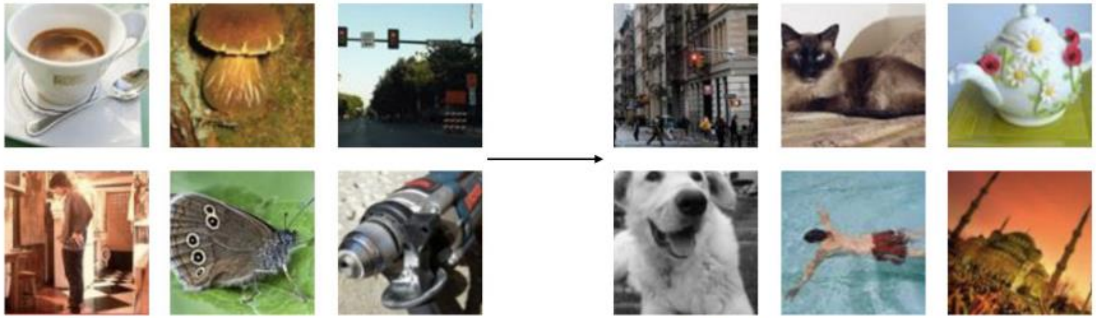
- Image captioning
- Action recognition
- Video summary

5. Deep Generative Models

Goal: To model the data density, *p(x)*

- Density estimation

- Sample generation



Training examples Model samples

*slide courtesy [Ian Goodfellow@]()

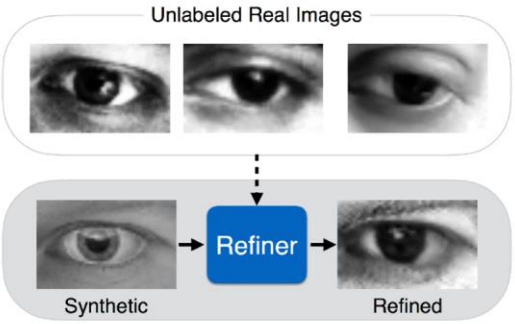
5. Deep Generative Models

Goal: To model the data density, *p(x)*

So, **Why** model *p(x)***?**

- To improve generalisation
- To create synthetic training data
- Practical tasks like speech synthesis
- To simulate situations
- To understand the data **better**

Generative models, appealing for image processing as data **priors**



(Shrivastava et al., 2016)

Apple's first research paper uses generative models to generate realistic data for eye tracking

*pic courtesy Ian Goodfellow@

5. Deep Generative Models

What's in the course?

Introduction

- Why generative models?
- Inference for image restoration

Deep Generative Models

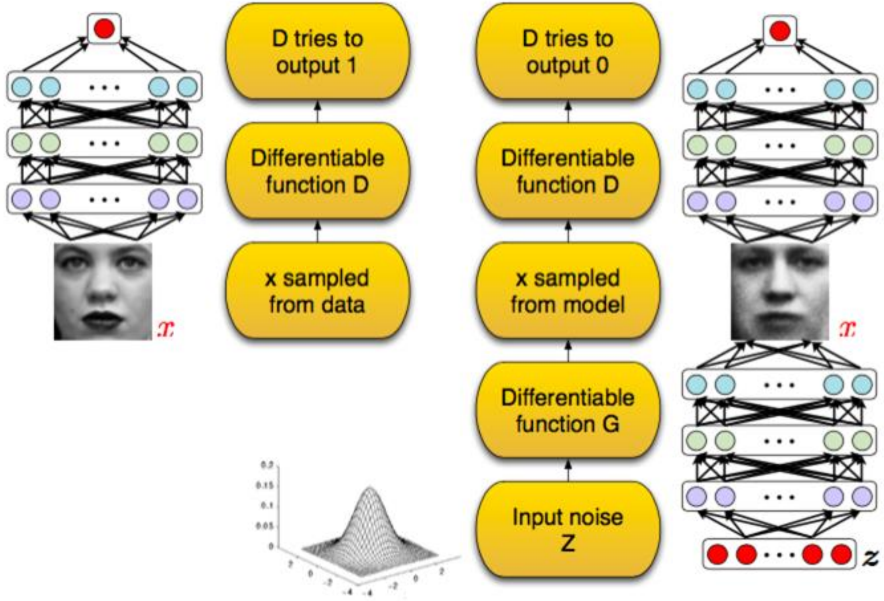
- Autoregressive models
  - ❑ Recurrent Image Density Estimator (RIDE)
  - ❑ Pixel Recurrent Neural Net (PixelRNN/CNN), PixelCNN++
- Variational Autoencoder (VAE)
  - ❑ Variational inference
  - ❑ Reparametrization trick

Applications

- Image restoration tasks
  - ❑ Denoising
  - ❑ Superresolution
  - ❑ Inpainting
- Compressive imaging
  - ❑ Single pixel camera
  - ❑ FlatCam

6. Generative Adversarial Networks (GANs)

GANs are generative models based on game theory



$$E_x[log(D(x))] + E_z[log(1 - D(G(z)))]$$

Min-max game: The generator tries to minimize this function while the discriminator tries to maximize it.

*slide courtesy Ian Goodfellow@

6. Generative Adversarial Networks (GANs)

Applications

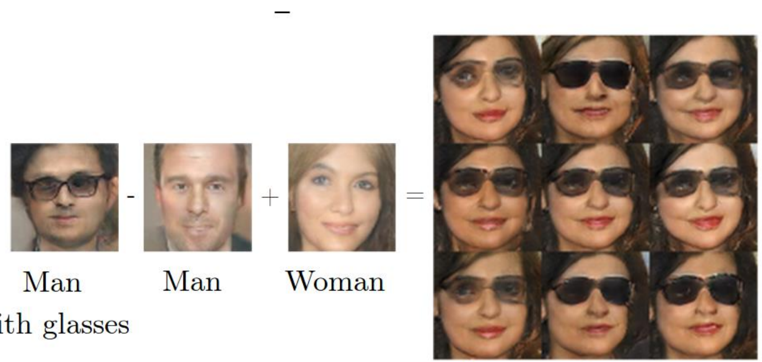
Sampling new data

Training Data Samples

(Salimans et al 2016)

(Goodfellow 2016)

Performing vector arithmetic



Man with glasses Man Woman

Woman with Glasses

(Radford et al, 2015)

*slide courtesy [Ian Goodfellow@](#)

6. Generative Adversarial Nets

What's in the course?

Introduction

- Motivation
- Two player min-max game

Training difficulties

- Balancing the players (discriminator and generator)
- Mode collapse

Architectures

- Deep Convolutional GAN (DCGAN)
- Conditional GANs

Applications

- Photo realistic super resolution
- Image to image translation
- Contextual autoencoder
- Image completing using GANs

EE-5179
Deep Learning for Imaging

- Instructor: Kaushik Mitra (kmitra@ee.iitm.ac.in)
- Classes: 3 Lectures (mostly Mon, Thurs, Fri)
- Interactive class: Tutorial (whenever needed)

Course info

- Course assumes
  - Some experience with computer vision or image processing
  - Programming experience
    - MATLAB or Python

Grading

(Subject to change)

Tutorial: 10%

Programming assignments : 20%

Mini-quizzes : 20%

Final exam : 30%

Kaggle comp/Paper: 20%

- About 4 programming assignments
- About 4 mini-quizzes

Quick introductions

- Name
- Department
- Undergraduate/graduate.
- Research interests
- Prior image processing/ computer vision experience