CVPR 2014 Tutorial

Deep Learning for Computer Vision

Graham Taylor (University of Guelph)
Marc’ Aurelio Ranzato (Facebook)
Honglak Lee (University of Michigan)

https://sites.google.com/site/deeplearningcvpr2014
Tutorial Overview

https://sites.google.com/site/deeplearningcvpr2014

• Basics
  – Introduction
  – Supervised Learning
  – Unsupervised Learning

• Libraries
  – Torch7
  – Theano/Pylearn2
  – CAFFE

• Advanced topics
  – Object detection
  – Regression methods for localization
  – Large scale classification and GPU parallelization
  – Learning transformations from videos
  – Multimodal and multi task learning
  – Structured prediction
Traditional Recognition Approach

Input data (pixels) \rightarrow feature representation (hand-crafted) \rightarrow Learning Algorithm (e.g., SVM)

Features are not learned

Image \rightarrow Low-level vision features (edges, SIFT, HOG, etc.) \rightarrow Object detection / classification
Computer vision features

SIFT

HoG

Spin image

Textons

and many others:

SURF, MSER, LBP, Color-SIFT, Color histogram, GLOH, ….
Motivation

• Features are key to recent progress in recognition

• Multitude of hand-designed features currently in use

• Where next? Better classifiers? building better features?

Felzenszwalb, Girshick, McAllester and Ramanan, PAMI 2007

Yan & Huang
(Winner of PASCAL 2010 classification competition)
What Limits Current Performance?

- Ablation studies on Deformable Parts Model
  - Felzenszwalb, Girshick, McAllester, Ramanan, PAMI’10

- Replace each part with humans (Amazon Turk):
  
  Parikh & Zitnick, CVPR’10

- Also removal of part deformations has small (<2%) effect.
  - Are “Deformable Parts” necessary in the Deformable Parts Model?
    Divvala, Hebert, Efros, ECCV 2012
Mid-Level Representations

• Mid-level cues

- Continuation
- Parallelism
- Junctions
- Corners

“Tokens” from Vision by D. Marr:

• Object parts:

- Difficult to hand-engineer → What about learning them?
Learning Feature Hierarchy

- Learn hierarchy
- All the way from pixels $\rightarrow$ classifier
- One layer extracts features from output of previous layer

- Train all layers jointly
Learning Feature Hierarchy

1. Learn **useful higher-level features** from images

2. Fill in representation gap in recognition

Lee et al., ICML 2009; CACM 2011
Learning Feature Hierarchy

• Better performance

• Other domains (unclear how to hand engineer):
  – Kinect
  – Video
  – Multi spectral

• Feature computation time
  – Dozens of features now regularly used [e.g., MKL]
  – Getting prohibitive for large datasets (10’s sec /image)
Approaches to learning features

• Supervised Learning
  – **End-to-end learning** of deep architectures (e.g., deep neural networks) with **back-propagation**
  – Works well when the amounts of labels is large
  – Structure of the model is important (e.g. convolutional structure)

• Unsupervised Learning
  – Learn **statistical structure or dependencies** of the data from unlabeled data
  – Layer-wise training
  – Useful when the amount of labels is not large
Taxonomy of feature learning methods

Supervised

Shallow
- Support Vector Machine
- Logistic Regression
- Perceptron
- Denoising Autoencoder
- Restricted Boltzmann machines
- Sparse coding

Deep
- Deep Neural Net
- Convolutional Neural Net
- Recurrent Neural Net
- Deep (stacked) Denoising Autoencoder
- Deep Belief Nets
- Deep Boltzmann machines
- Hierarchical Sparse Coding

Unsupervised

* supervised version exists
Supervised Learning
Example: Convolutional Neural Networks

- LeCun et al. 1989
- Neural network with specialized connectivity structure
Convolutional Neural Networks

• Feed-forward:
  – Convolve input
  – Non-linearity (rectified linear)
  – Pooling (local max)
• Supervised
• Train convolutional filters by back-propagating classification error

LeCun et al. 1998
Components of Each Layer

- **Pixels / Features**
  - Filter with Dictionary (convolutional or tiled)
    - + Non-linearity

- **Spatial/Feature (Sum or Max)**

- **Normalization**
  - between feature responses

- **Output Features**
Filtering

• Convolutional
  – Dependencies are local
  – Translation equivariance
  – Tied filter weights (few params)
  – Stride 1, 2, … (faster, less mem.)

Input

Feature Map

Slide: R. Fergus
Non-Linearity

- Non-linearity
  - Per-element (independent)
  - Tanh
  - Sigmoid: $1/(1+\exp(-x))$
  - Rectified linear
    - Simplifies backprop
    - Makes learning faster
    - Avoids saturation issues

→ Preferred option
• Spatial Pooling
  – Non-overlapping / overlapping regions
  – Sum or max
  – Boureau et al. ICML’10 for theoretical analysis
Normalization

• Contrast normalization (across feature maps)
  – Local mean = 0, local std. = 1, “Local” $\rightarrow$ 7x7 Gaussian
  – Equalizes the features maps

Feature Maps

Feature Maps After Contrast Normalization

Slide: R. Fergus
Compare: SIFT Descriptor

Image Pixels → Apply Gabor filters

Spatial pool (Sum)

Normalize to unit length

Feature Vector

Slide: R. Fergus
Applications

• Handwritten text/digits
  – MNIST (0.17% error [Ciresan et al. 2011])
  – Arabic & Chinese [Ciresan et al. 2012]

• Simpler recognition benchmarks
  – CIFAR-10 (9.3% error [Wan et al. 2013])
  – Traffic sign recognition
    • 0.56% error vs 1.16% for humans [Ciresan et al. 2011]
Application: ImageNet

- ~14 million labeled images, 20k classes
- Images gathered from Internet
- Human labels via Amazon Turk

[Deng et al. CVPR 2009]
Krizhevsky et al. [NIPS 2012]

- Same model as LeCun’98 but:
  - Bigger model (8 layers)
  - More data \((10^6 \text{ vs } 10^3 \text{ images})\)
  - GPU implementation (50x speedup over CPU)
  - Better regularization (DropOut)

- 7 hidden layers, 650,000 neurons, 60,000,000 parameters
- Trained on 2 GPUs for a week
ImageNet Classification 2012

- Krizhevsky et al. -- 16.4% error (top-5)
- Next best (non-convnet) – 26.2% error
ImageNet Classification 2013 Results

Feature Generalization

- Girshick et al. CVPR’14 (Caltech-101, SunS)
- Oquab et al. CVPR’14 (VOC 2012)
- Razavian et al. arXiv 1403.6382, 2014 (lots of datasets)

- Pre-train on ImageNet
- Retrain classifier on Caltech256


Bo, Ren, Fox, CVPR 2013
Sohn, Jung, Lee, Hero, ICCV 2011
Industry Deployment

• Used in Facebook, Google, Microsoft
• Image Recognition, Speech Recognition, ....
• Fast at test time

Taigman et al. DeepFace: Closing the Gap to Human-Level Performance in Face Verification, CVPR’14
Unsupervised Learning
Unsupervised Learning

• Model distribution of input data

• Can use unlabeled data (unlimited)

• Can be refined with standard supervised techniques (e.g. backprop)

• Useful when the amount of labels is small
Unsupervised Learning

• Main idea: model distribution of input data
  – Reconstruction error + regularizer (sparsity, denoising, etc.)
  – Log-likelihood of data

• Models
  – Basic: PCA, KMeans
  – Denoising autoencoders
  – Sparse autoencoders
  – Restricted Boltzmann machines
  – Sparse coding
  – Independent Component Analysis
  – ...
Example: Auto-Encoder

Feed-back / generative / top-down path

Decoder

Encoder

Feed-forward / bottom-up path

Input (Image/ Features)

Output Features

Bengio et al., NIPS’07; Vincent et al., ICML’08
Stacked Auto-Encoders

Bengio et al., NIPS’07; Vincent et al., ICML’08; Ranzato et al., NIPS’07
At Test Time

- Remove decoders
- Use feed-forward path
- Gives standard (Convolutional) Neural Network
- Can fine-tune with backprop

Class label

Encoder

Features

Encoder

Features

Encoder

Features

Input Image

Bengio et al., NIPS’07;
Vincent et al., ICML’08;
Ranzato et al., NIPS’07

Slide: R. Fergus
Learning basis vectors for images

Natural Images

Learned bases: “Edges”

Test example

\[ x \sim 0.8 * w_{36} + 0.3 * w_{42} + 0.5 * w_{65} \]

\[ [0, 0, \ldots, 0, 0.8, 0, \ldots, 0, 0.3, 0, \ldots, 0, 0.5, \ldots] \]

= coefficients (feature representation)

Compact & easily interpretable

[Olshausen & Field, Nature 1996, Ranzato et al., NIPS 2007; Lee et al., NIPS 2007; Lee et al., NIPS 2008; Jarret et al., CVPR 2009; etc.]
Learning Feature Hierarchy

Higher layer (Combinations of edges)

First layer (edges)

Input image (pixels)

[Olshausen & Field, Nature 1996, Ranzato et al., NIPS 2007; Lee et al., NIPS 2007; Lee et al., NIPS 2008; Jarret et al., CVPR 2009; etc.]
Learning object representations

• Learning objects and parts in images

• Large image patches contain interesting higher-level structures.
  – E.g., object parts and full objects
Unsupervised learning of feature hierarchy

“Shrink” (max over 2x2) output

“Filtering” output

“Eye detector”

Advantage of shrinking
1. Filter size is kept small
2. Invariance

Input image

H. Lee, R. Grosse, R. Ranganath, A. Ng, ICML 2009; Comm. ACM 2011
Unsupervised learning of feature hierarchy

H. Lee, R. Grosse, R. Ranganath, A. Ng, ICML 2009; Comm. ACM 2011
Unsupervised learning from natural images

First layer bases
localized, oriented edges

Second layer bases
contours, corners, arcs, surface boundaries

Related work: Zeiler et al., CVPR’10, ICCV’11; Kavuckuglou et al., NIPS’09
Learning object-part decomposition

Applications:

- Object recognition (Lee et al., ICML’09, Sohn et al., ICCV’11; Sohn et al., ICML’13)
- Verification (Huang et al., CVPR’12)
- Image alignment (Huang et al., NIPS’12)

Cf. Convnet [Krizhevsky et al., 2012]; Deconvnet [Zeiler et al., CVPR 2010]
Large-scale unsupervised learning

- Large-scale deep autoencoder (three layers)
- Each stage consists of
  - local filtering
  - L2 pooling
  - local contrast normalization
- Training jointly the three layers by:
  - reconstructing the input of each layer
  - sparsity on the code

Le et al. “Building high-level features using large-scale unsupervised learning, 2011

Slide: M. Ranzato
Large-scale unsupervised learning

- Large-scale deep autoencoder
- Discovers high-level features from large amounts of unlabeled data
- Achieved state-of-the-art performance on ImageNet classification 10k categories

Le et al. “Building high-level features using large-scale unsupervised learning, 2011"
Supervised vs. Unsupervised

• Supervised models
  – Work very well with large amounts of labels (e.g., imagenet)
  – Convolutional structure is important

• Unsupervised models
  – Work well given limited amounts of labels.
  – Promise of exploiting virtually unlimited amount of data without need of labeling
Summary

• Deep Learning of Feature Hierarchies
  – showing great promises for computer vision problems

• More details will be presented later:
  – Basics: Supervised and Unsupervised
  – Libraries: Torch7, Theano/Pylearn2, CAFFE
  – Advanced topics:
    • Object detection, localization, structured output prediction, learning from videos, multimodal/multitask learning, structured output prediction
Tutorial Overview

https://sites.google.com/site/deeplearningcvpr2014

- Basics
  - Introduction
  - Supervised Learning
  - Unsupervised Learning

- Libraries
  - Torch7
  - Theano/Pylearn2
  - CAFFE

- Advanced topics
  - Object detection
  - Regression methods for localization
  - Large scale classification and GPU parallelization
  - Learning transformations from videos
  - Multimodal and multi task learning
  - Structured prediction

- Honglak Lee
- Marc’Aurelio Ranzato
- Graham Taylor
- Marc’Aurelio Ranzato
- Ian Goodfellow
- Yangqing Jia
- Pierre Sermanet
- Alex Toshev
- Alex Krizhevsky
- Roland Memisevic
- Honglak Lee
- Yann LeCun