Classification, Localization and Detection using Deep Learning

Pierre Sermanet, David Eigen, Michael Mathieu, Xiang Zhang, Rob Fergus, Yann LeCun
New York University

OverFeat
- **ImageNet Challenge**
  - 2012: classification, localization, fine-grained classification
  - 2013: classification, localization, detection

- **Classification:**
  - 1000 classes
  - correct if in the top 5 answers (image may contain multiple classes)
Classification + Localization:
  ○ 1000 classes
  ○ predict correct class and return at most 5 bounding boxes that overlap by at least 50%.

Top 5:
white wolf
white wolf
timber wolf
timber wolf
Arctic fox

Groundtruth:
white wolf
white wolf (2)
white wolf (3)
white wolf (4)
white wolf (5)
● Localization:
  ○ a good measure?
  ○ classification < localization < detection
  ○ very good to evaluate localization method independently from other detection challenges (background training)
Detection:
- 200 classes
- Smaller objects than classification/localization
- Any number of objects (including zero)
- Penalty for false positives
Official results:

- **Classification:**
  - 14.2% error
  - 4th position behind Clarifai-ZF (11.1%), NUS (12.9%), Andrew Howard (13.5%)

- **Localization:**
  - 29.9% error
  - 1st position, followed by Alex Krizhevsky (34% in 2012), and Oxford VGG (46%)

- **Detection:**
  - 19.4% mean AP
  - 3rd position behind UvA (22.6%) and NEC (20.9%)

Only team entering all tasks
• **Classification:**
  - standard architecture
  - no normalization
  - voting:
    - multi-view (4 corners + 1 center views + flip = 10 views)
    - 7 models voting
  - GPU implementation
    - fast and low memory footprint important to train bigger models

• **Localization**
  - regression predicting coordinates of bounding boxes
    - top-left (x,y) and bottom-right (x,y)
    - center (x,y), height and width: center does not depend on scale
    - fancier (similar to yann’s face pose estimation)
  - replace classifier with regressor, inputs: 256x5x5 (right after last pooling)

• **Detection:**
  - training with background to avoid false positives, trade-off between positive/negative accuracy
Detection / Localization

- Detection / Localization

  - groundtruth bounding box
• **ConvNets and detection:**
  - particularly suited for detection
  - reusing neighbor computations
  - no need to recompute entire network at each location
ConvNets for Detection

- **Single output:**
  - 1x1 output
  - no feature space
  - **blue:** feature maps
  - **green:** operation kernel
  - typical training setup

---

![Diagram showing a 14x14 input going through a 5x5 convolution to a 10x10 output, followed by a 2x2 pooling, then a sequence of 5x5 convolutions and 1x1 convolutions leading to the output.](image-url)
- **Multiple outputs:**
  - 2x2 output
  - input stride 2x2
  - recompute only extra yellow areas
ConvNets for Detection

- **With feature space**
  - 3 input channels
  - 4 feature maps
  - 2 feature maps
  - 4 feature maps
  - 2 outputs (e.g. 2-class classifier)
● Traditional detection approach:
  ○ multi-scale
  ○ sliding window
  ○ non-maximum suppression (NMS)
- Our detection approach:
  - for each location, predict bounding box
  - accumulate instead of suppress
  - another form of voting
• Bounding boxes voting:
  ○ voting is good (classification: views voting + model voting)
  ○ boosts confidence high above false positives ([0,1] up to 10.43 here)
  ○ more robust to individual localization errors
  ○ relying less on an accurate background class
Augmenting views of a ConvNet:

- the more subsampling, the larger the output stride
- larger output stride means less views

- e.g.: subsampling x2, x3, x2, x3 => 36 pixels stride
- 1 pixel shift in output space corresponds to 36 pixels shift in input space
• Augmenting views of a ConvNet:
  ○ 9x more bounding boxes (with last pooling 3x3)
- **Reducing output stride:**
  - example: last pooling 3x3 with stride 3x3
  - change pooling stride to 1x1
  - following layer now must skip every 3 pixels and repeat 9 times

  ![Diagram](image)

  - technique introduced by Giusti et al.

- **Fine stride:**
  - stronger voting
  - e.g. 3x3 bounding boxes instead of 1x1 for first scale
• Fine stride voting:
  ○ confidence boosts from ~10 to ~75
  ○ more optimal input alignment with network yields stronger activations/confidence
Detection / Localization

OverFeat • Pierre Sermanet • New York University
Detection / Localization

OverFeat • Pierre Sermanet • New York University
Detection: Failures that make sense

Top predictions:
corkscrew (confidence 38.1)

Groundtruth:
snake
Detection: Failures that make sense

**Top predictions:**
- remote control (confidence 31.8)
- filing cabinet (confidence 2.2)

**Groundtruth:**
- table
- water bottle
- water bottle (2)
- water bottle (3)
- water bottle (4)
- refrigerator
Detection: Interesting Failures

Top predictions:
strawberry (confidence 201.5)

Groundtruth:
strawberry
strawberry (2)
strawberry (3)
strawberry (4)
strawberry (5)
strawberry (6)
Top predictions:
microwave (confidence 5.6)
refrigerator (confidence 2.5)

Groundtruth:
bowl
microwave
Top predictions:
artichoke (confidence 162.8)

Groundtruth:
sunglasses
artichoke
artichoke (2)
artichoke (3)
Top predictions:
trombone (confidence 26.8)
oboe (confidence 17.5)
oboe (confidence 11.5)

Groundtruth:
person
hat with a wide brim
hat with a wide brim (2)
hat with a wide brim (3)
oboe
obo (2)
saxophone
trombone
person (2)
person (3)

ILSVRC2012_val_00000614.JPEG
Top predictions:
person (confidence 6.0)

Groundtruth:
drum
lamp
lamp (2)
guitar
person
person (2)
person (3)
person (3)
microphone
microphone (2)
microphone (3)
Top predictions:
tennis ball (confidence 3.5)
banana (confidence 2.4)
banana (confidence 2.1)
hotdog (confidence 2.0)
banana (confidence 1.9)

Groundtruth:
strawberry
strawberry (2)
strawberry (3)
strawberry (4)
strawberry (5)
strawberry (6)
strawberry (7)
strawberry (8)
strawberry (9)
strawberry (10)

ILSVRC2012_val_00000320.JPEG
Some hard ones

- Moving to heat maps measure?

**Top predictions:**
watercraft (confidence 72.2)
watercraft (confidence 2.1)

**Groundtruth:**
watercraft
watercraft (2)
Some easy ones

**Top predictions:**
bird (confidence 86.0)
bird (confidence 70.9)

**Groundtruth:**
bird
bird (2)

ILSVRC2012_val_00001136.JPEG
Top predictions:
burrito (confidence 28.9)

Groundtruth:
person
burrito

Top predictions:
burrito (confidence 17.4)

Groundtruth:
burrito
burrito (2)
Coming up next week:

- release of our feature extractor (forward only)
  - based on TH tensor library (in C)
  - wrappers: torch, python, matlab
  - extract features at any layer up to 1000-classifier
  - fast in-house cuda code not released

- other libs:
  - cuda-conv (Alex Krizhevsky)
  - DeCAF (A Deep Convolutional Activation Feature for Generic Visual Recognition, berkeley)
• **Live demos:**
  - 1000-class classification
  - 1-shot learning

• **Speed:**
  - CPU: ~1 fps
  - GPU: ~10 fps (proprietary cuda code)
  - gpu code is fast in mini-batch mode but also for small batches