Unsupervised Visual Representation Learning by Context Prediction

Carl Doersch
Abhinav Gupta
Alexei A. Efros

ICCV 2015

Presenter: Ahmadreza Jeddi
July 2019
Presented in CS 898 course [instructor: professor Yuri Boykov]

University of Waterloo, department of computer science

Outline

- Learning representations for image fragments by using **context**
- Trained CNNs used as initialization for object detection by R-CNN on Pascal VOC 2007 dataset
- Visual data discovery (unsupervised object discovery) by using representations of image fragments

Introduction

- We like to have rich and high-performance representations of visual data

- Problem statement:

  - Datasets with millions of labeled examples have let CNN-based models learn excellent representations

  - But what about Internet-scale datasets (e.g. hundreds of billions of images) with no annotations?

    - Unsupervised learning …

      - But without labels, what should be represented?

      - How can one write an objective function to capture representation for an object if the object is not labeled?

Common unsupervised methods to tackle representation learning

- Method A) Image representations as latent variables of generative models

- Method B) Image representations as embeddings

Method A

- Image representations as latent variables of generative models

- Example: auto-encoders

- Inferring latent structure is intractable given an image, so these models use sampling to perform approximation

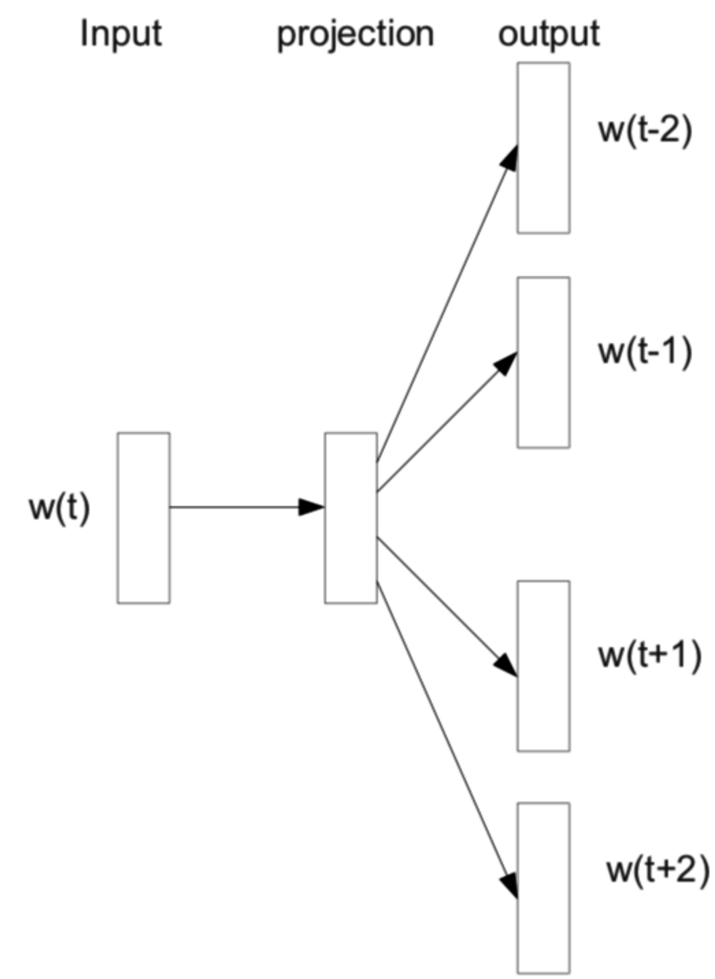
- Promising performance on smaller datasets (e.g. handwritten digits) but not effective for high resolution natural images

Method B

- Image representations as embeddings
- Semantically similar images should have close embedding
- Use a pretext task to create the embeddings
- Pretext task: converts the unsupervised problem into a self-supervised one
- Context prediction as a pretext task: successful in text domain
    - "Skip-gram" model: word embedding in text domain by using word context

Skip-gram model

- Predicts the context (n preceding and n succeeding words) of a word
- Converts the unsupervised problem of predicting representations into a self-supervised problem of predicting a word's context
- Training a neural network for this task generates the embedding of words
- But can we use this context idea in image domain?



The Skip-gram model architecture [2]

Context in image domain

- Challenge: predicting pixels is much harder than predicting words

- Two ideas:

  - Idea A: one patch in an image replaced by a random patch from elsewhere in the dataset

  - Goal: discriminate true patches from the randomly replaced patch

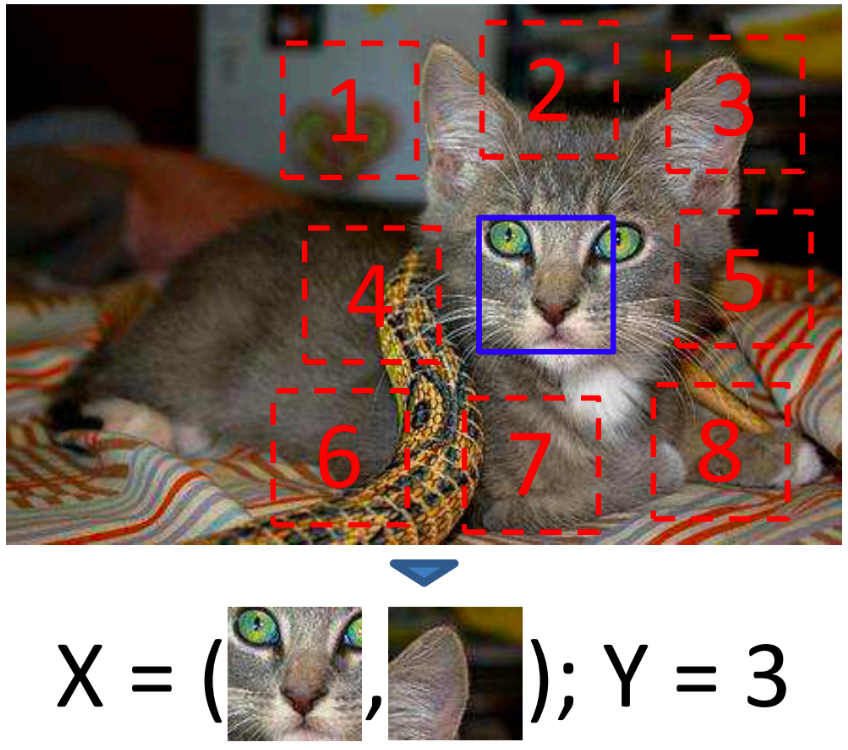
  - This task is trivial: discriminating low-level color statistics and lighting would be enough

Context in image domain, Cont.

- Idea B (this work): sample 9 patches (figure 2) from the ***same image***. Given the middle patch and a random one (from the ramaining 8 patches), what is the relative position of this random patch to the middle one?

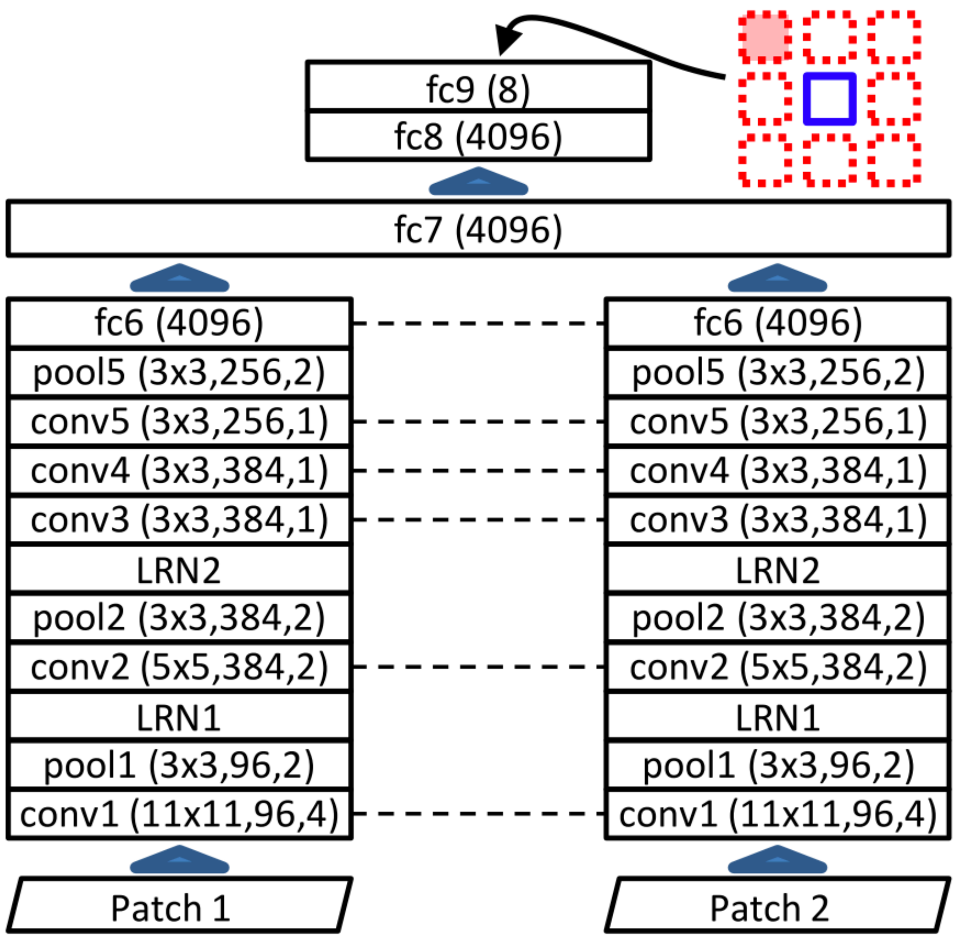
- All patches sharing the same lighting and color statistics

- Hypothesis: Doing well on this task requires understanding scenes and objects



Learning visual context prediction

- Each patch is processed separately until fc6
- Two representations are fused at fc7
- Weights are tied between the two AlexNets
- Output is one of the 8 possible configurations ⟶ Softmax
- Output of fc6 is the embedding of a patch



The late fusion architecture. A pair of AlexNet-style architectures.
Dotted lines indicate shared weights

Avoiding trivial solutions

- Care must be taken to ensure a pretext task does not take "trivial" shortcuts
- Possible trivial shortcuts in this work:
  - Low-level cues like boundary patterns or textures continuing between patches
    - Solution: gap between patches and random jittering
  - Chromatic aberration: raised from differences in the way the lens focuses light at different wavelengths
    - ConvNets can localize patches relative to lens itself
    - Solution: projecting color channels or dropping two of them and replacing them with gaussian noise

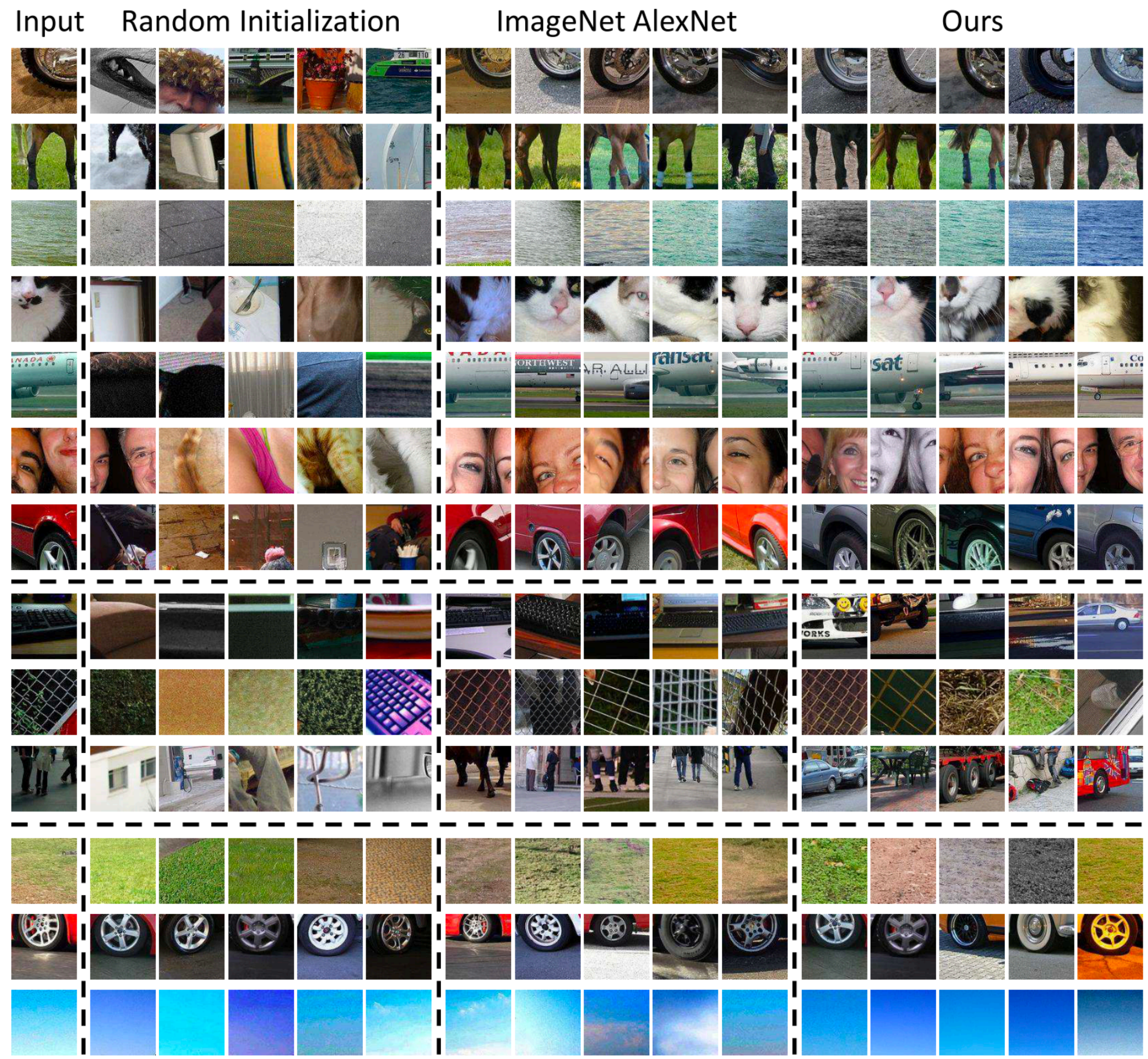
Experiments

- Nearest Neighbours
  - By using KNN, determine how good the learned embeddings are
- Object detection
  - R-CNN (Regions with CNN features) with different CNNs and initializations
  - Trained model used as an initialization: significant boost compared to learning from scratch
- Visual data mining
  - Find image fragments which depict the same semantic objects
  - Finding object clusters in unsupervised manner

Nearest Neighbours

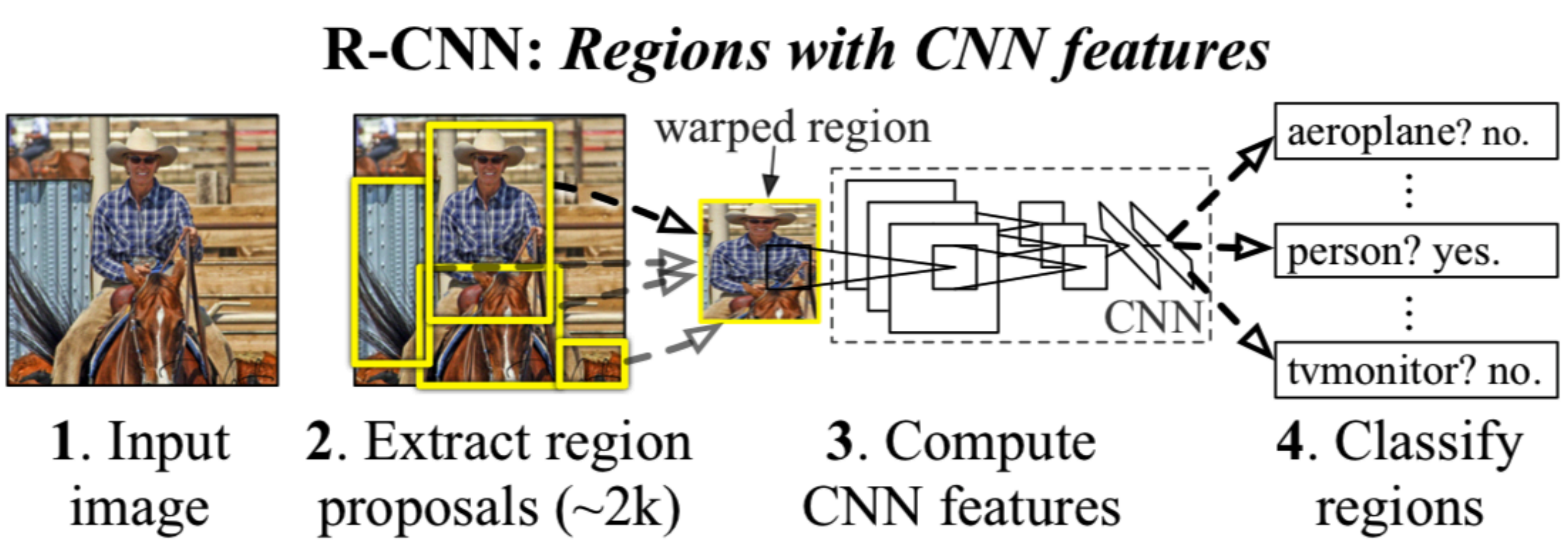
- Find nearest neighbours in embedding space to current patch's embedding vector

- Random queries: random patch selected as the input



Object detection

- R-CNN (Regions with CNN features)
- Different architectures and different initializations possible for CNN (part 3 in the pipeline)



Object detection, cont.

- Pascal VOC 2007 dataset
- MAP (mean average precision) used as comparison metric

VOC-2007 Test	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
DPM-v5[17]	33.2	60.3	10.2	16.1	**27.3**	54.3	58.2	23.0	20.0	24.1	26.7	12.7	58.1	48.2	43.2	12.0	21.1	36.1	46.0	43.5	33.7
[8] w/o context	52.6	52.6	19.2	25.4	18.7	47.3	56.9	42.1	16.6	41.4	41.9	27.7	47.9	51.5	29.9	20.0	41.1	36.4	48.6	53.2	38.5
Regionlets[55]	54.2	52.0	20.3	24.0	20.1	55.5	68.7	42.6	19.2	44.2	**49.1**	26.6	57.0	54.5	43.4	16.4	36.6	37.7	**59.4**	52.3	41.7
Scratch-R-CNN[2]	49.9	60.6	24.7	23.7	20.3	52.5	64.8	32.9	20.4	43.5	34.2	29.9	49.0	60.4	47.5	28.0	42.3	28.6	51.2	50.0	40.7
Scratch-Ours	52.6	60.5	23.8	24.3	18.1	50.6	65.9	29.2	19.5	43.5	35.2	27.6	46.5	59.4	46.5	25.6	42.4	23.5	50.0	50.6	39.8
Ours-projection	58.4	62.8	**33.5**	27.7	24.4	**58.5**	68.5	41.2	**26.3**	**49.5**	42.6	**37.3**	**55.7**	**62.5**	49.4	**29.0**	**47.5**	28.4	54.7	56.8	45.7
Ours-color-dropping	**60.5**	**66.5**	29.6	**28.5**	26.3	56.1	**70.4**	**44.8**	24.6	45.5	45.4	35.1	52.2	60.2	**50.0**	28.1	46.7	**42.6**	54.8	**58.6**	**46.3**
Ours-Yahoo100m	56.2	63.9	29.8	27.8	23.9	57.4	69.8	35.6	23.7	47.4	43.0	29.5	52.9	62.0	48.7	28.4	45.1	33.6	49.0	55.5	44.2
Ours-VGG	63.6	64.4	42.0	42.9	18.9	67.9	69.5	65.9	28.2	48.1	58.4	58.5	66.2	64.9	54.1	26.1	43.9	55.9	69.8	50.9	53.0
ImageNet-R-CNN[19]	64.2	69.7	50	41.9	32.0	62.6	71.0	60.7	32.7	58.5	46.5	56.1	60.6	66.8	54.2	31.5	52.8	48.9	57.9	64.7	54.2

Visual data mining

- Unsupervised object discovery
- Application example: content-based retrieval
- Method:
  - Transfer input image to 4 adjacent patches
  - Find 100 images with strongest matches for all four patches
  - Geometric validation: geometrical consistency of matched patches

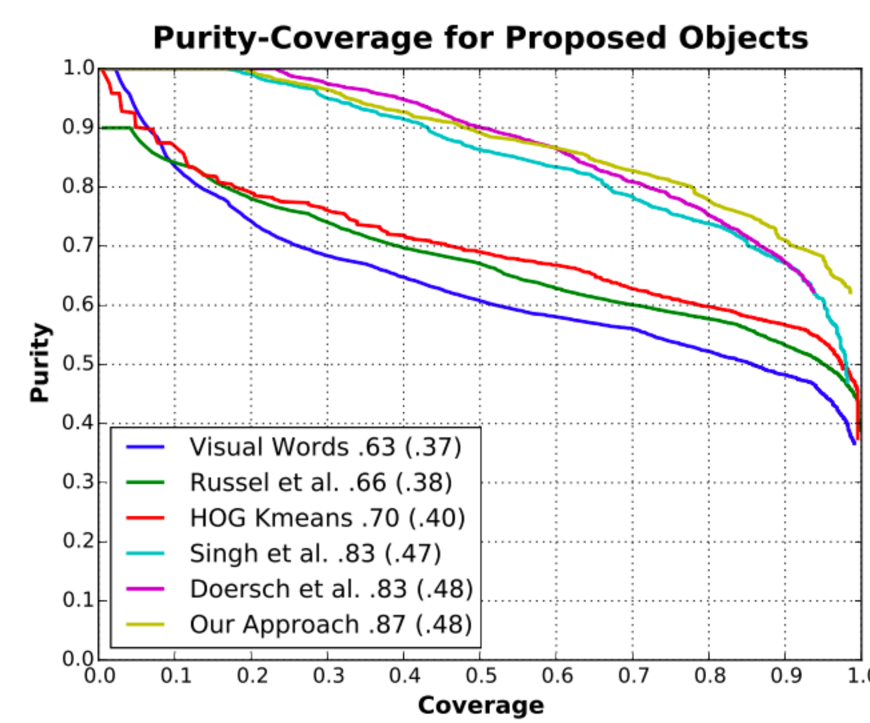
Qualitative results

- Image retrieval on VOC 2011 dataset

Discovered image clusters. Numbers show ranking, determined by the fraction of the top matches that are geometrically verified

Quantitative results

- Clustering images from a subset of Pascal VOC 2007
- Iterative clustering of 1000 sets each having 10 images
- Rank clusters and add them together
- Evaluation metric: AUC (Area Under Curve)
- Purity: the fraction of images in the cluster containing the same category
- Coverage: the fraction of images in the dataset that are contained in at least one of the sets up to a point



Purity vs coverage for objects discovered on a subset of Pascal VOC 2007. Legend numbers show AUC. Numbers in parentheses show AUC up to coverage of 0.5

References

[1] C. Doersch, A. Gupta, and A. A. Efros. Context as supervisory sig- nal: Discovering objects with predictable context. In ECCV. 2014.

[2] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Dis- tributed representations of words and phrases and their composition- ality. In NIPS, 2013.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hier- archies for accurate object detection and semantic segmentation. In CVPR, 2014.

[4] C. Doersch, A. Gupta, and A. A. Efros. Unsupervised visual representation learning by context prediction. ICCV, 2015.

Thank you

Any question?