Research Seminar
Related work and Review

Y Data Program

Asaf Dahan and Omer Dodi

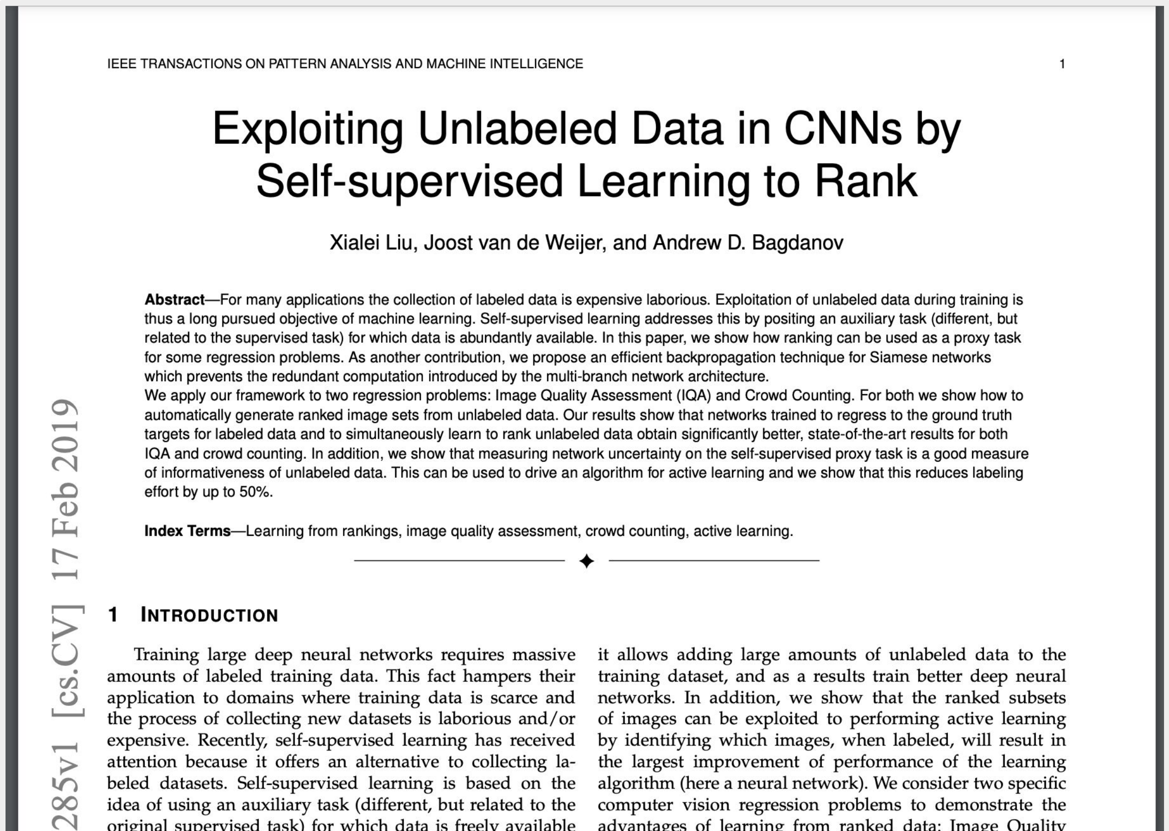
April 2022

Agenda

- Another self-supervised task
- Cited papers
- One cited paper
- Criticism

10 min

Another self-supervised task



Exploiting Unlabeled Data in CNNs by Self-supervised Learning to Rank

Xialei Liu, Joost van de Weijer, and Andrew D. Bagdanov

Abstract—For many applications the collection of labeled data is expensive laborious. Exploitation of unlabeled data during training is thus a long pursued objective of machine learning. Self-supervised learning addresses this by positing an auxiliary task (different, but related to the supervised task) for which data is abundantly available. In this paper, we show how ranking can be used as a proxy task for some regression problems. As another contribution, we propose an efficient backpropagation technique for Siamese networks which prevents the redundant computation introduced by the multi-branch network architecture.

We apply our framework to two regression problems: Image Quality Assessment (IQA) and Crowd Counting. For both we show how to automatically generate ranked image sets from unlabeled data. Our results show that networks trained to regress to the ground truth targets for labeled data and to simultaneously learn to rank unlabeled data obtain significantly better, state-of-the-art results for both IQA and crowd counting. In addition, we show that measuring network uncertainty on the self-supervised proxy task is a good measure of informativeness of unlabeled data. This can be used to drive an algorithm for active learning and we show that this reduces labeling effort by up to 50%.

Index Terms—Learning from rankings, image quality assessment, crowd counting, active learning.

1 Introduction

Training large deep neural networks requires massive amounts of labeled training data. This fact hampers their application to domains where training data is scarce and the process of collecting new datasets is laborious and/or expensive. Recently, self-supervised learning has received attention because it offers an alternative to collecting labeled datasets. Self-supervised learning is based on the idea of using an auxiliary task (different, but related to the original supervised task) for which data is freely available

it allows adding large amounts of unlabeled data to the training dataset, and as a results train better deep neural networks. In addition, we show that the ranked subsets of images can be exploited to performing active learning by identifying which images, when labeled, will result in the largest improvement of performance of the learning algorithm (here a neural network). We consider two specific computer vision regression problems to demonstrate the advantages of learning from ranked data: Image Quality

10 min

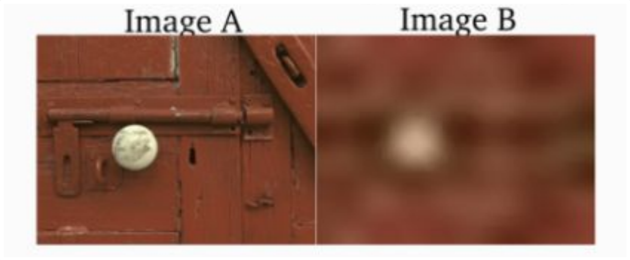
Self-supervised Learning to Rank

- Two regression problems which suffers from small datasets due to expensive handmade labeling.

Crowd Counting

Image Quality Assessment

Predict the perceptual quality of images



10 min

Self-supervised Learning to Rank

- Generate ranked datasets

Crowd Counting

Compare image crops which are contained within each other: a smaller image contained in another larger one will contain the same number or fewer persons than the larger image

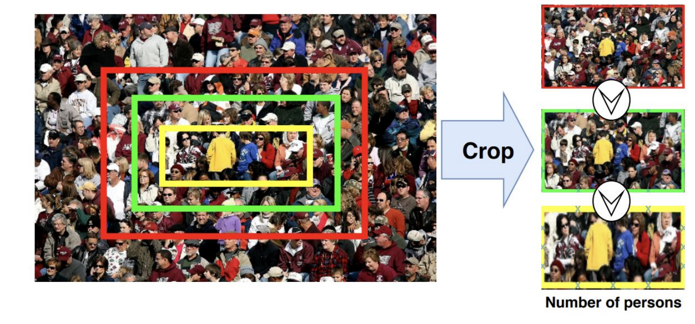
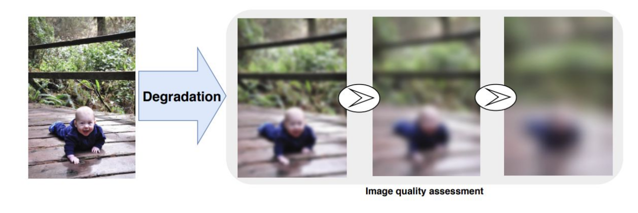


Image Quality Assessment

Add increasing levels of distortions to images



10 min

Self-supervised Learning to Rank

- Network architecture

Crowd Counting

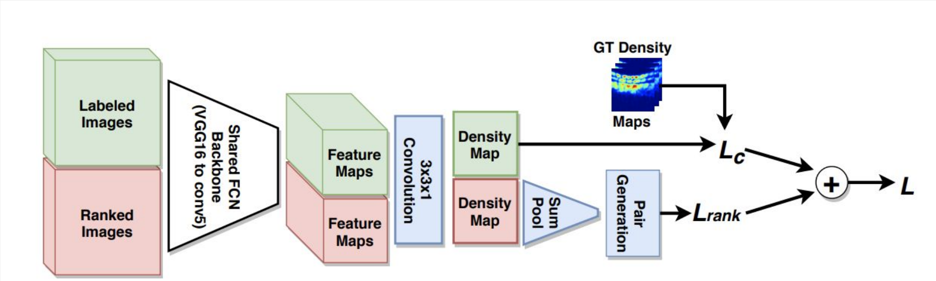


Fig. 4. Network architecture and ranked pair generation for crowd counting. **Top**: our counting network uses a VGG16 network truncated at the fifth convolutional layer (before maxpooling). To this network we add a $3 \times 3 \times 1$ convolutional layer with stride 1 which should estimate local crowd density. A sum pooling layer is added to the ranking channel of the network to arrive at a scalar value whose relative rank is known.

Image Quality Assessment

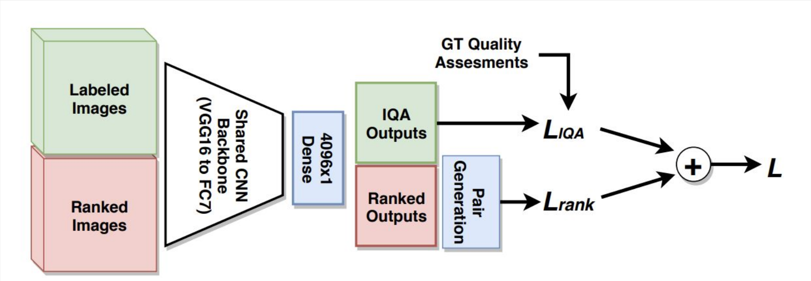
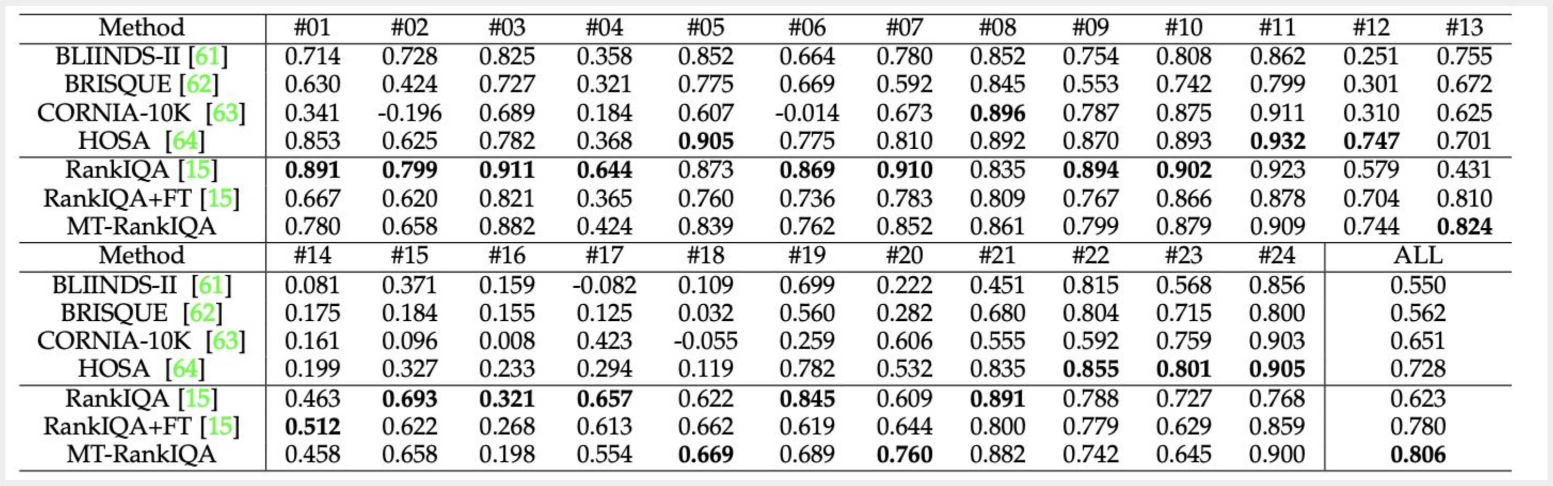


Fig. 2. Network architecture and ranked pair generation for IQA. **Top**: our network for no-reference IQA uses a VGG16 network pretrained on ImageNet. We decapitate the network and replace the original head with a new fully-connected layer generating a single output.

10 min

Self-supervised Learning to Rank

- Results improve significantly when adding unlabeled data of the ranking task
  - Example for comparison chosen score to different methods, IQA is Image Quality Assessment



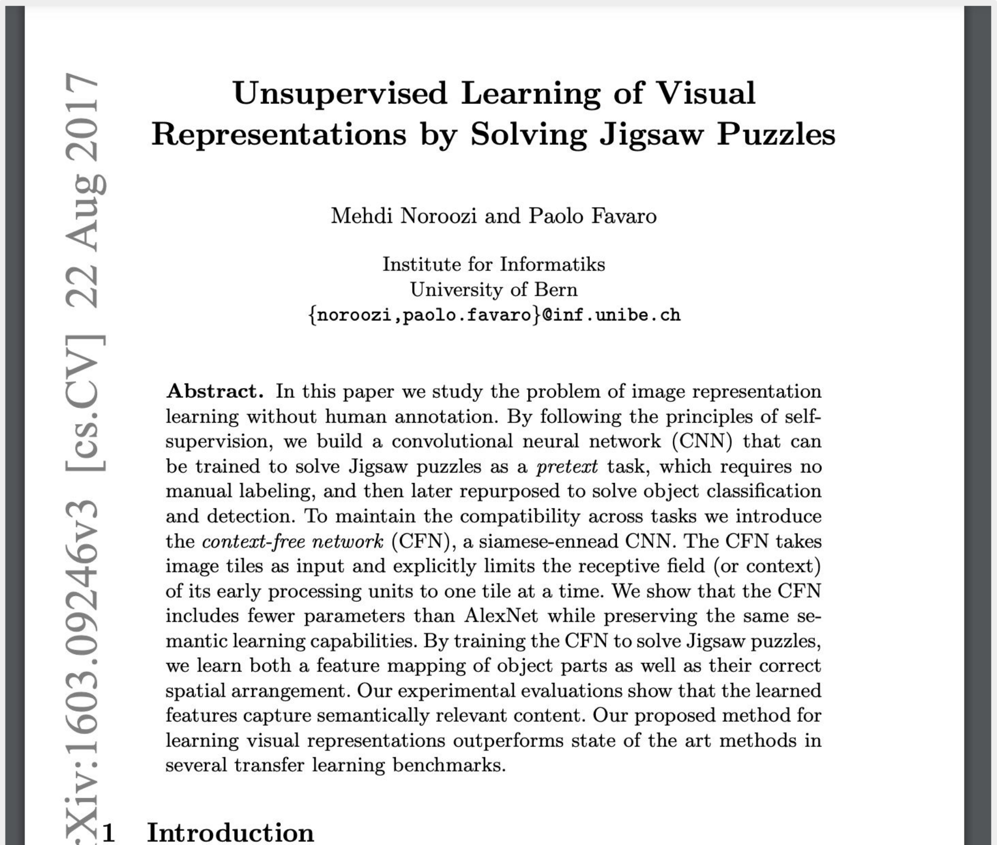
Method	#01	#02	#03	#04	#05	#06	#07	#08	#09	#10	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20	#21	#22	#23	#24	ALL
BLIINDS-II [61]	0.714	0.728	0.825	0.358	0.852	0.664	0.780	0.852	0.754	0.808	0.862	0.251	0.755	0.081	0.371	0.159	-0.082	0.109	0.699	0.222	0.451	0.815	0.568	0.856	0.550
BRISQUE [62]	0.630	0.424	0.727	0.321	0.775	0.669	0.592	0.845	0.553	0.742	0.799	0.301	0.672	0.175	0.184	0.155	0.125	0.032	0.560	0.282	0.680	0.804	0.715	0.800	0.562
CORNIA-10K [63]	0.341	-0.196	0.689	0.184	0.607	-0.014	0.673	**0.896**	0.787	0.875	0.911	0.310	0.625	0.161	0.096	0.008	0.423	-0.055	0.259	0.606	0.555	0.592	0.759	0.903	0.651
HOSA [64]	0.853	0.625	0.782	0.368	**0.905**	0.775	0.810	0.892	0.870	0.893	**0.932**	**0.747**	0.701	0.199	0.327	0.233	0.294	0.119	0.782	0.532	0.835	**0.855**	**0.801**	**0.905**	0.728
RankIQA [15]	**0.891**	**0.799**	**0.911**	**0.644**	0.873	**0.869**	**0.910**	0.835	**0.894**	**0.902**	0.923	0.579	0.431	0.463	**0.693**	**0.321**	**0.657**	0.622	**0.845**	0.609	**0.891**	0.788	0.727	0.768	0.623
RankIQA+FT [15]	0.667	0.620	0.821	0.365	0.760	0.736	0.783	0.809	0.767	0.866	0.878	0.704	0.810	**0.512**	0.622	0.268	0.613	0.662	0.619	0.644	0.800	0.779	0.629	0.859	0.780
MT-RankIQA	0.780	0.658	0.882	0.424	0.839	0.762	0.852	0.861	0.799	0.879	0.909	0.744	**0.824**	0.458	0.658	0.198	0.554	**0.669**	0.689	**0.760**	0.882	0.742	0.645	0.900	**0.806**

1 min

Cited papers

- What kind of papers cited this paper?
  - **Self Learning Methods**
    - Temporal - 12 papers
      - Divide video samples to adjacent frames as a learning signal
    - Spatial - 5 papers
      - Divide image samples for patches for learning
    - Colorization - 3 papers
      - Divide image samples to b&w and colored for learning
  - **Semantic Segmentation** - 5 papers
    - Assign label to every pixel in the image
    - As the final task for the self learned proxy task
  - And **others** like unsupervised learning (8 papers), Adversarial learning (2 papers), Semi-supervised (1 paper), Object segmentation (1 paper)

One cited paper, spatial self learning



Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles

Mehdi Noroozi and Paolo Favaro

Institute for Informatiks
University of Bern
{noroozi,paolo.favaro}@inf.unibe.ch

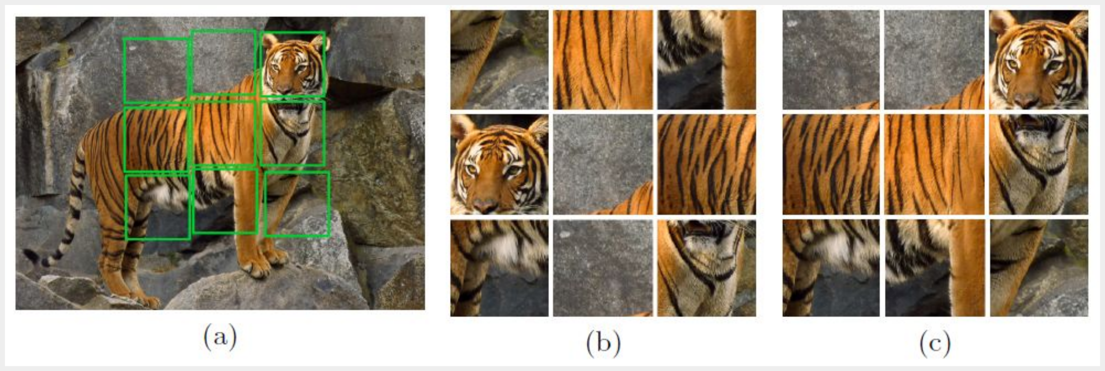
Abstract. In this paper we study the problem of image representation learning without human annotation. By following the principles of self-supervision, we build a convolutional neural network (CNN) that can be trained to solve Jigsaw puzzles as a *pretext* task, which requires no manual labeling, and then later repurposed to solve object classification and detection. To maintain the compatibility across tasks we introduce the *context-free network* (CFN), a siamese-ennead CNN. The CFN takes image tiles as input and explicitly limits the receptive field (or context) of its early processing units to one tile at a time. We show that the CFN includes fewer parameters than AlexNet while preserving the same semantic learning capabilities. By training the CFN to solve Jigsaw puzzles, we learn both a feature mapping of object parts as well as their correct spatial arrangement. Our experimental evaluations show that the learned features capture semantically relevant content. Our proposed method for learning visual representations outperforms state of the art methods in several transfer learning benchmarks.

1 Introduction

5 min

Jigsaw Puzzles

- Input is image tiles (self created)
- output is correct arrangement



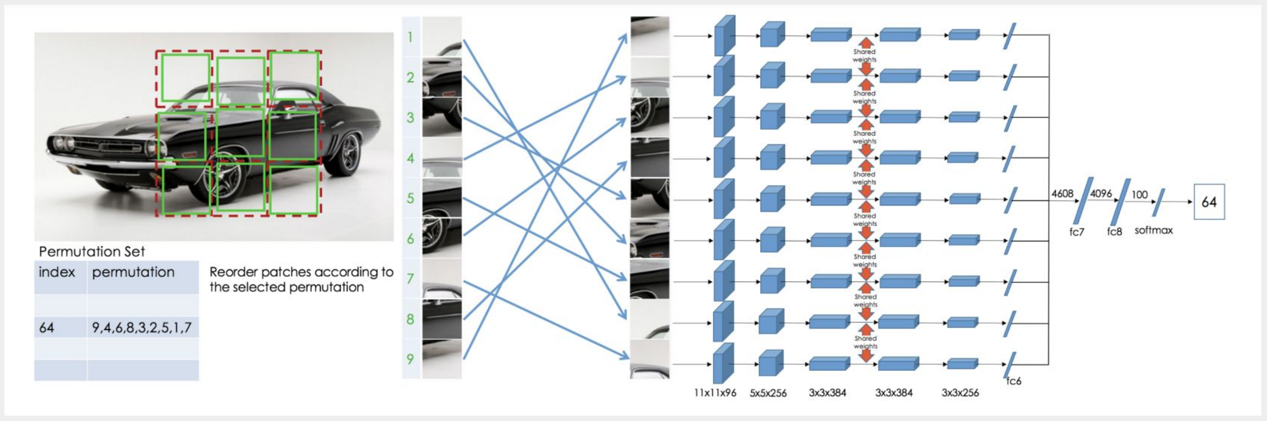
5 min

Jigsaw Puzzles

- Learn correct arrangement of object parts by Context Free Network (“CFN”)
    - AlexNet based
    - Crop
    - Permutate
    - Predict index of chosen permutation

Why “CFN”?

The data flow of each patch is explicitly separated until the fully connected layer and context is handled only in the last fully connected layers.



Permutation Set

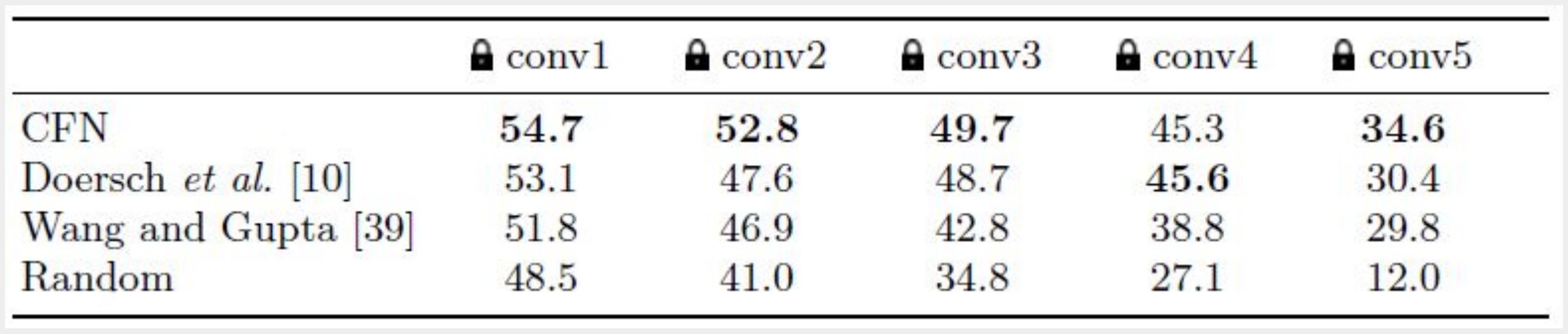
index	permutation
64	9,4,6,8,3,2,5,1,7

Reorder patches according to the selected permutation

5 min

Jigsaw Puzzles

- Results
  - Comparison of classification results on ImageNet 2012
  - 34.6% when only fully connected layers are trained
  - Major increase, 45.3%, when the conv5 layer is also trained
  - conv5 layer starts to be specialized



	🔒 conv1	🔒 conv2	🔒 conv3	🔒 conv4	🔒 conv5
CFN	**54.7**	**52.8**	**49.7**	45.3	**34.6**
Doersch *et al.* [10]	53.1	47.6	48.7	**45.6**	30.4
Wang and Gupta [39]	51.8	46.9	42.8	38.8	29.8
Random	48.5	41.0	34.8	27.1	12.0

5 min

Criticism

- Organized material for personal use - website and git repo

- Concepts within the article are not fully explained. It feels that the author expect from the reader to come with wide pre-knowledge.
  - For example, Concept of “hypercolumn” is not elaborated enough and you need to search it on your own.
  - “Hypercolumn at a pixel is the vector of activations of all CNN units above that pixel, as shown above.” (https://towardsdatascience.com/review-hypercolumn-instance-segmentation-367180495979)

- Conclusion part is very short - 3 sentences with high level overview.
  What about next steps? Important specific results?