Unsupervised pretraining

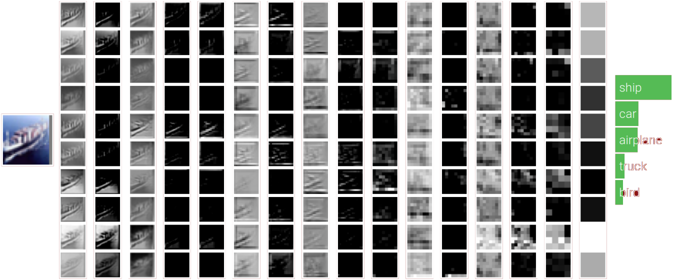
of deep neural networks

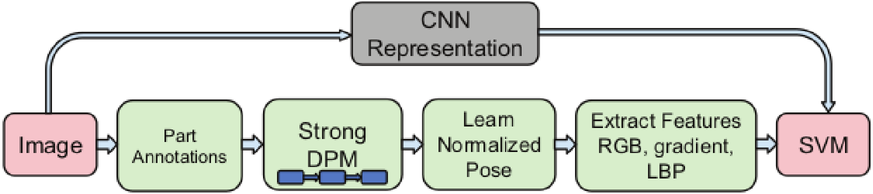
Olof Mogren, Research institutes of Sweden

Neural network pretraining

- Weight initialization
- Limited data
- Leverage knowledge from other data source
- Overcome vanishing gradients
- Start a revolution (author(s), year?)

Supervised pretraining





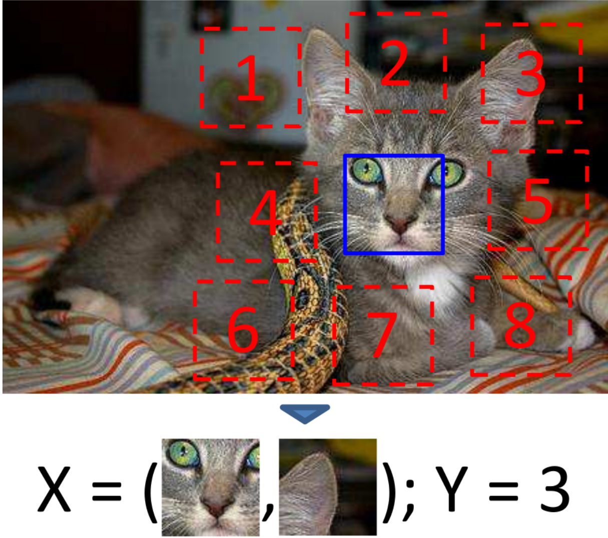
(Using Imagenet classification), Razavian, et.al., 2014

RI. SE

Self-supervised learning

A class of unsupervised learning techniques.

- Predict relative position of patches
- Reorder shuffled patches
- Image completion
- Video next frame prediction
- Word embeddings
- Language models, Transformers
- etc.



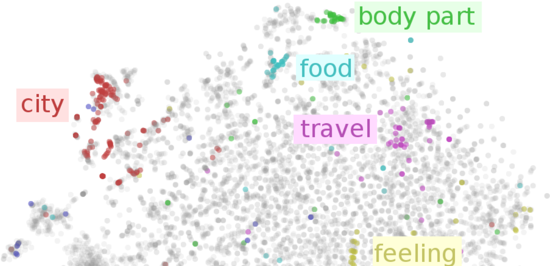
Unsupervised

-learning

- Hebbian learning
  ("fire together, wire together")
- Self-organization
- Model probability density of inputs
- Clustering
- Dimensionality reduction
- Self-supervised learning

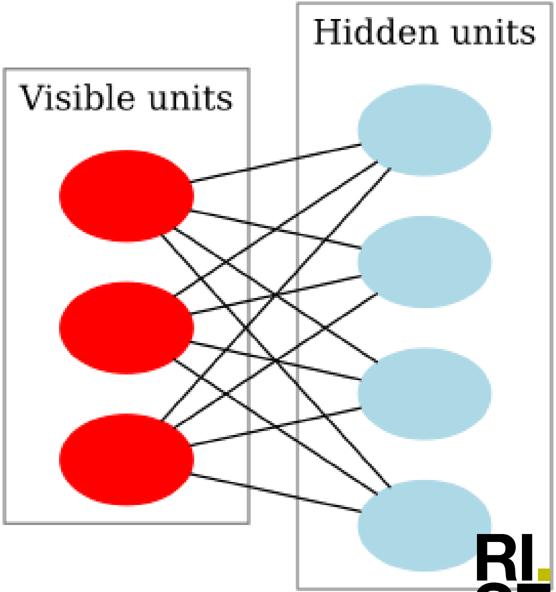
-pretraining

- Clustering
- Dimensionality reduction
- Restricted Boltzmann machines
- Autoencoders

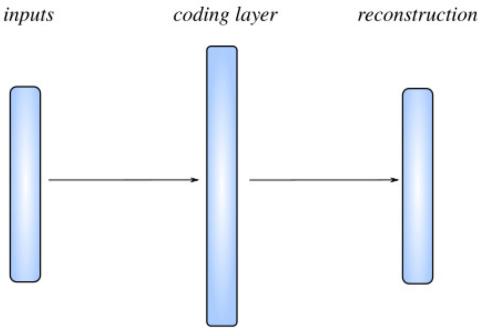


Restricted Boltzmann machines (RBMs)

- Generative model
- Contrastive divergence
- Maximum likelihood
- Deep belief networks (Hinton et.al. 2006)
- Deep Boltzmann machines



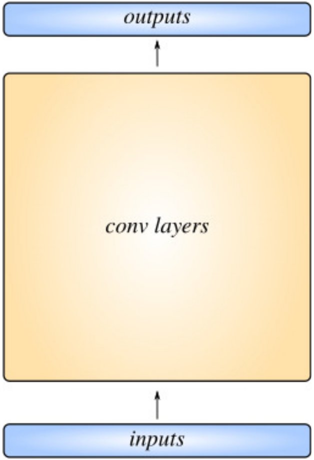
Autoencoders

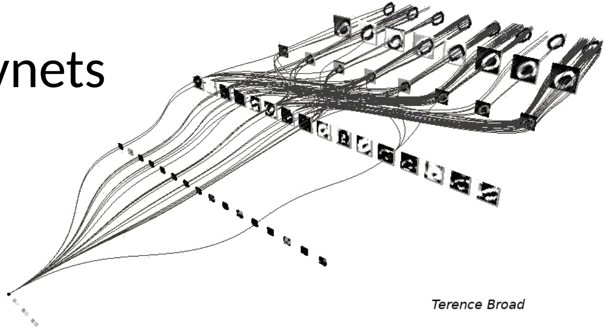


- Neural network trained to reproduce its input
- Unsupervised layerwise pretraining
- Bottleneck
- Denoising
- Stacked

Larochelle et al., 2009

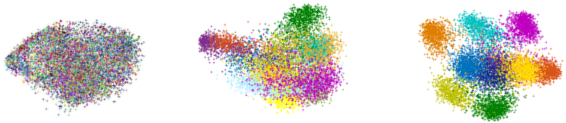
Convnets





- Popular for tasks such as image classification
- Randomly initialized convnet performs much better than chance

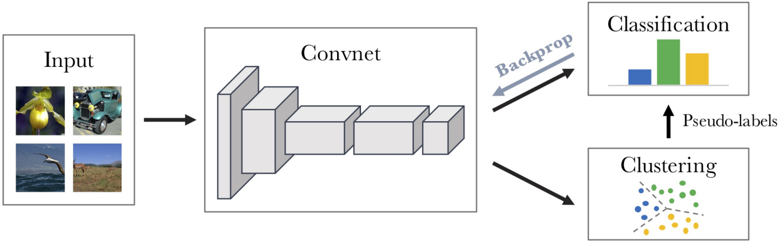
Convnet pretraining using clustering



(a) Initial stage (b) Middle stage (c) Final stage

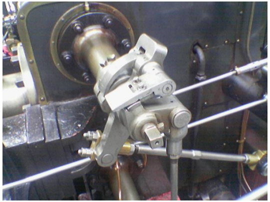
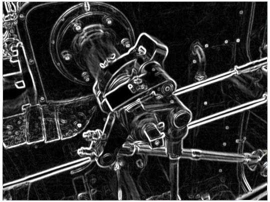
- Adam Coates, Andrew Ng. (2012): Layerwise k-means
- Dosovitskiy, et.al. (2014), “Surrogate” classification
- Yang, et.al. (2016), Recurrent agglomerative clustering
- Xie, et.al. (2016), Deep embedded clustering
- Liao, et.al. (2016), K-means

Deep clustering for unsupervised learning of visual features



- Begin with randomly initialized convnet
- Cluster data based on computed representations
  - (PCA reduction to 256 dims, whitened, ℓ_2-normalized)
- Train supervised with cluster assignments as labels
- Repeat

Nuts and bolts

- Avoiding trivial clusterings:
  When a cluster *i* becomes empty:
  - Pick another cluster *j*,
    use centroid of *j* with
    small perturbation as centroid of *i*
- Sobel filtering

More

- Dropout
- Constant step-size
- ℓ_2 regularization
- Momentum

- Linear classifier trained on frozen representations

Hyperparameters are selected based on a down-stream task.

Including K.

Conv1

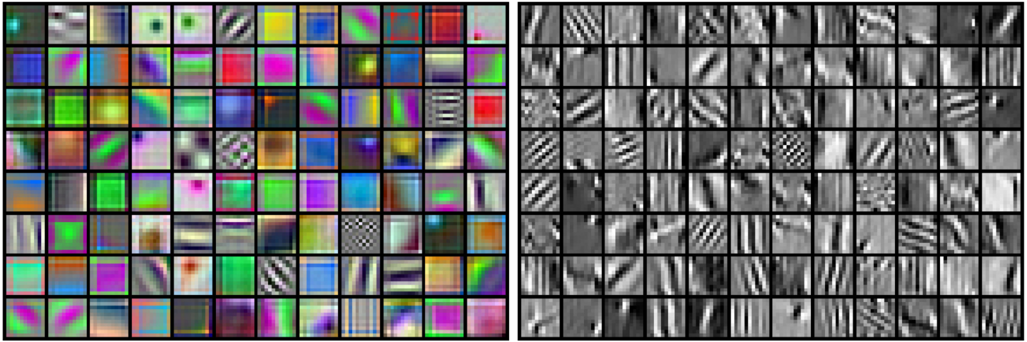


Fig. 3: Filters from the first layer of an AlexNet trained on unsupervised ImageNet on raw RGB input (left) or after a Sobel filtering (right).

Filter visualizations

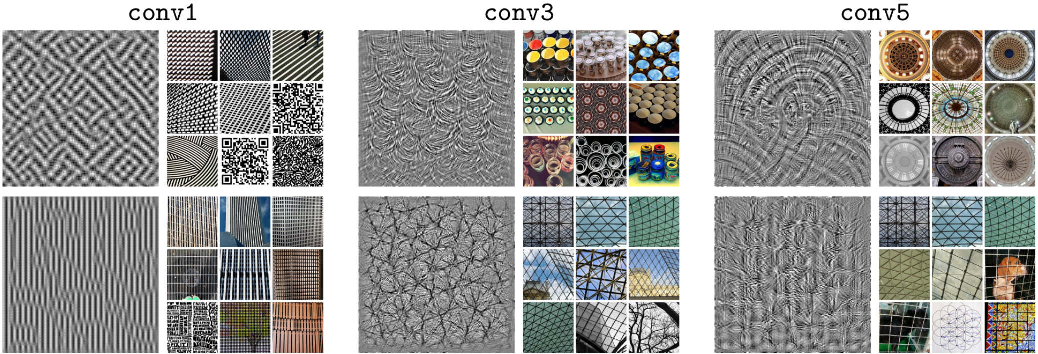


Fig. 4: Filter visualization and top 9 activated images from a subset of 1 million images from YFCC100M for target filters in the layers `conv1`, `conv3` and `conv5` of an AlexNet trained with DeepCluster on ImageNet. The filter visualization is obtained by learning an input image that maximizes the response to a target filter [64].

Conv1

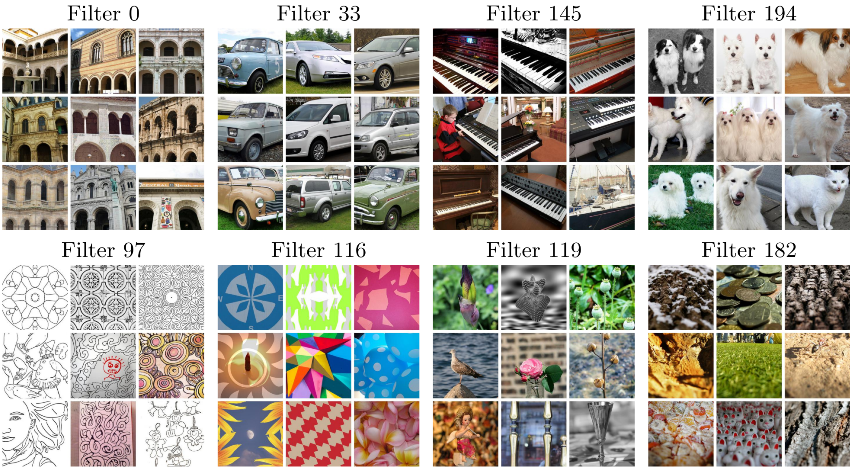


Fig. 5: Top 9 activated images from a random subset of 10 millions images from YFCC100M for target filters in the last convolutional layer. The top row corresponds to filters sensitive to activations by images containing objects. The

Method	ImageNet					Places				
	`conv1`	`conv2`	`conv3`	`conv4`	`conv5`	`conv1`	`conv2`	`conv3`	`conv4`	`conv5`
Places labels	–	–	–	–	–	22.1	35.1	40.2	43.3	44.6
ImageNet labels	19.3	36.3	44.2	48.3	50.5	22.7	34.8	38.4	39.4	38.7
Random	11.6	17.1	16.9	16.3	14.1	15.7	20.3	19.8	19.1	17.5
Pathak *et al.* [38]	14.1	20.7	21.0	19.8	15.5	18.2	23.2	23.4	21.9	18.4
Doersch *et al.* [25]	16.2	23.3	30.2	31.7	29.6	19.7	26.7	31.9	32.7	30.9
Zhang *et al.* [28]	12.5	24.5	30.4	31.5	30.3	16.0	25.7	29.6	30.3	29.7
Donahue *et al.* [20]	17.7	24.5	31.0	29.9	28.0	21.4	26.2	27.1	26.1	24.0
Noroozi and Favaro [26]	**18.2**	28.8	34.0	33.9	27.1	23.0	32.1	35.5	34.8	31.3
Noroozi *et al.* [45]	18.0	**30.6**	34.3	32.5	25.7	**23.3**	**33.9**	36.3	34.7	29.6
Zhang *et al.* [43]	17.7	29.3	35.4	35.2	32.8	21.3	30.7	34.0	34.1	32.5
DeepCluster	12.9	29.2	**38.2**	**39.8**	**36.1**	18.6	30.8	**37.0**	**37.5**	**33.1**

Table 1: Linear classification on ImageNet and Places using activations from the convolutional layers of an AlexNet as features. We report classification accuracy on the central crop. Numbers for other methods are from Zhang *et al.* [43].

Method	Classification FC6-8	Classification ALL	Detection FC6-8	Detection ALL	Segmentation FC6-8	Segmentation ALL
ImageNet labels	78.9	79.9	–	56.8	–	48.0
Random-rgb	33.2	57.0	22.2	44.5	15.2	30.1
Random-sobel	29.0	61.9	18.9	47.9	13.0	32.0
Pathak *et al.* [38]	34.6	56.5	–	44.5	–	29.7
Donahue *et al.* [20] *	52.3	60.1	–	46.9	–	35.2
Pathak *et al.* [27]	–	61.0	–	52.2	–	–
Owens *et al.* [44] *	52.3	61.3	–	–	–	–
Wang and Gupta [29] *	55.6	63.1	32.8†	47.2	26.0†	35.4†
Doersch *et al.* [25] *	55.1	65.3	–	51.1	–	–
Bojanowski and Joulin [19] *	56.7	65.3	33.7†	49.4	26.7†	37.1†
Zhang *et al.* [28] *	61.5	65.9	43.4†	46.9	35.8†	35.6
Zhang *et al.* [43] *	63.0	67.1	–	46.7	–	36.0
Noroozi and Favaro [26]	–	67.6	–	53.2	–	37.6
Noroozi *et al.* [45]	–	67.7	–	51.4	–	36.6
DeepCluster	**70.4**	**73.7**	**51.4**	**55.4**	**43.2**	**45.1**

Table 2: Comparison of the proposed approach to state-of-the-art unsupervised feature learning on classification, detection and segmentation on Pascal VOC. * indicates the use of the data-dependent initialization of Krähenbühl *et al.* [68]. Numbers for other methods produced by us are marked with a †.

Method	Training set	Classification		Detection		Segmentation	
		FC6-8	ALL	FC6-8	ALL	FC6-8	ALL
Best competitor	ImageNet	63.0	67.7	43.4†	53.2	35.8†	37.7
DeepCluster	ImageNet	72.0	73.7	51.4	55.4	43.2	45.1
DeepCluster	YFCC100M	67.3	69.3	45.6	53.0	39.2	42.2

Table 3: Impact of the training set on the performance of DeepCluster measured on the Pascal VOC transfer tasks as described in Sec. 4.4. We compare ImageNet with a subset of 1M images from YFCC100M [31]. Regardless of the training set, DeepCluster outperforms the best published numbers on most tasks. Numbers for other methods produced by us are marked with a †

Method	AlexNet	VGG-16
ImageNet labels	56.8	67.3
Random	47.8	39.7
Doersch *et al.* [25]	51.1	61.5
Wang and Gupta [29]	47.2	60.2
Wang *et al.* [46]	–	63.2
DeepCluster	**55.4**	**65.9**

Table 4: Pascal VOC 2007 object detection with AlexNet and VGG-16. Numbers are taken from Wang *et al.* [46].

Method	Oxford5K	Paris6K
ImageNet labels	72.4	81.5
Random	6.9	22.0
Doersch *et al.* [25]	35.4	53.1
Wang *et al.* [46]	42.3	58.0
DeepCluster	**61.0**	**72.0**

Table 5: mAP on instance-level image retrieval on Oxford and Paris dataset with a VGG-16. We apply R-MAC with a resolution of 1024 pixels and 3 grid levels [70].

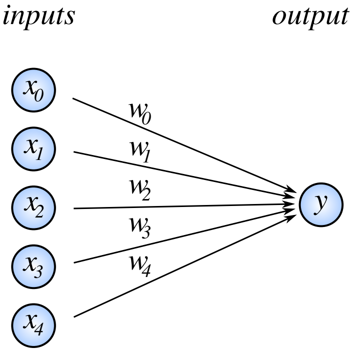
Conclusions

- Works also with random Flickr images
-

Appendix

Deep learning building block

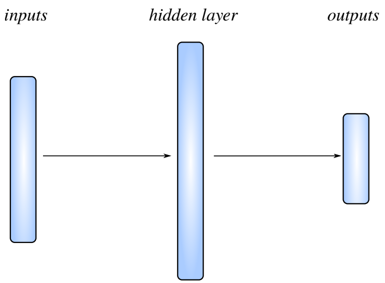
- Each layer contains a number of units
- *Loosely* inspired by biological neurons
- Deep networks can consist of millions of units
- $w_1, \ldots, w_n$ learned parameters



(Back)

Deep learning layer

- In practice: neurons arranged in layers
- Each layer:
  - linear transformation of input vector
  - non-linear squashing-function



(Back)