Boulder

Computer Vision; Image Classification; Transfer Learning

YouTube Playlist

Maziar Raissi

Assistant Professor

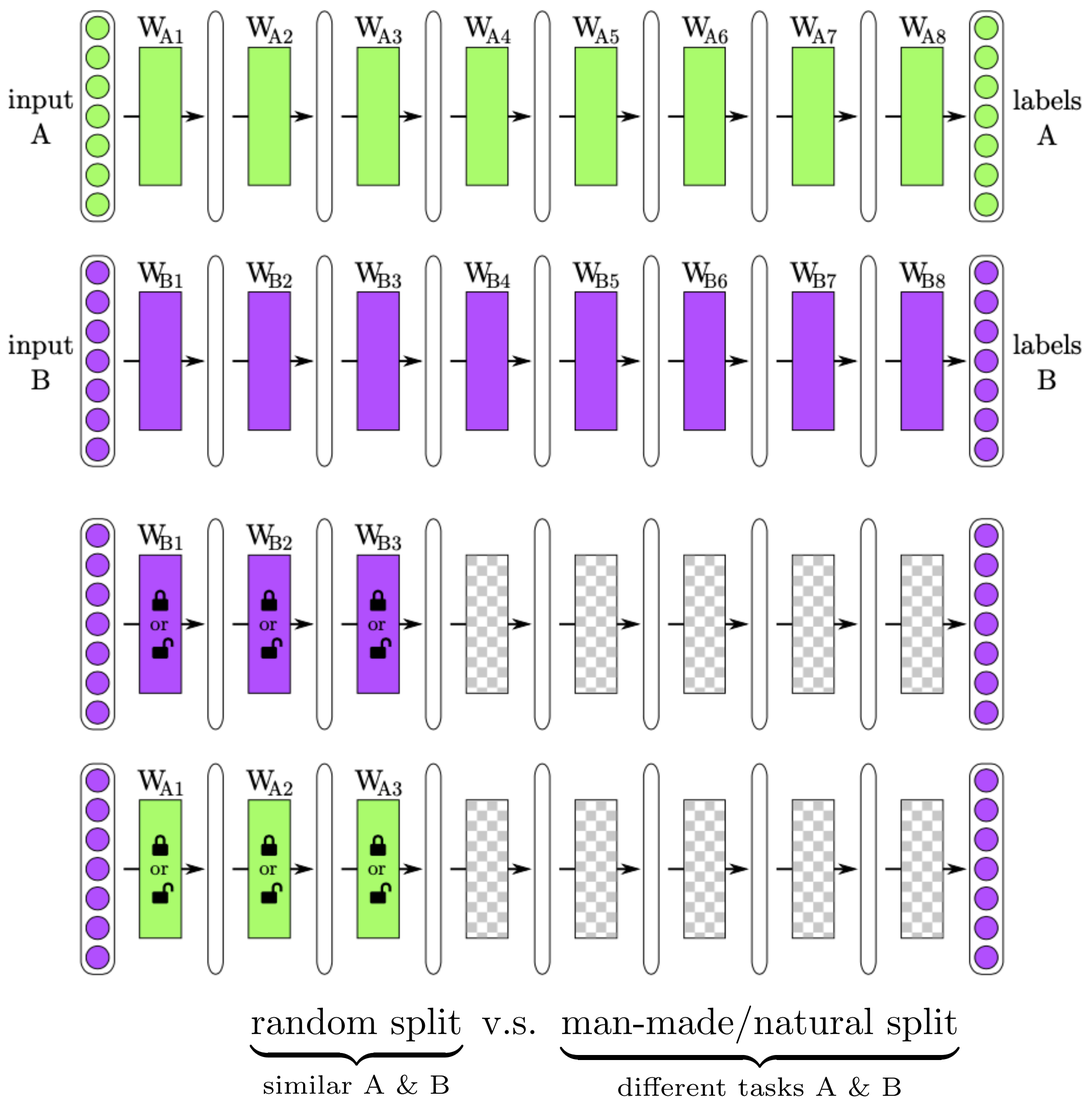
Department of Applied Mathematics

University of Colorado Boulder

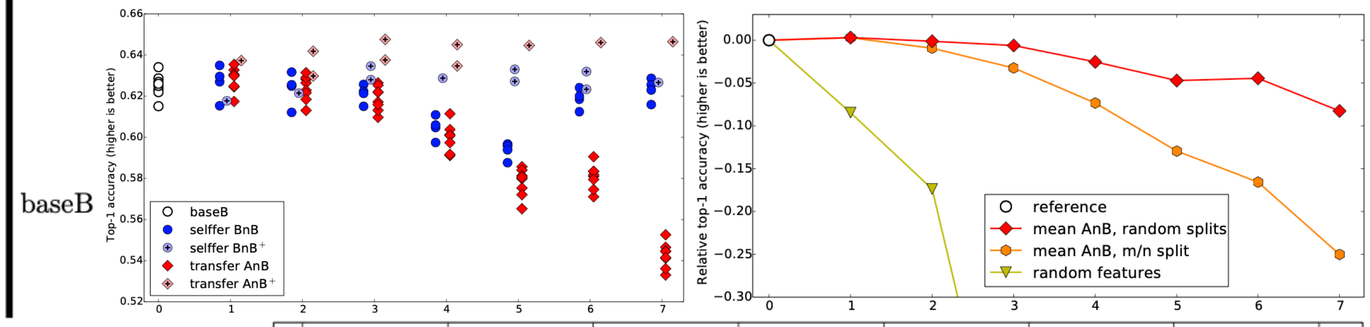
maziar.raissi@colorado.edu

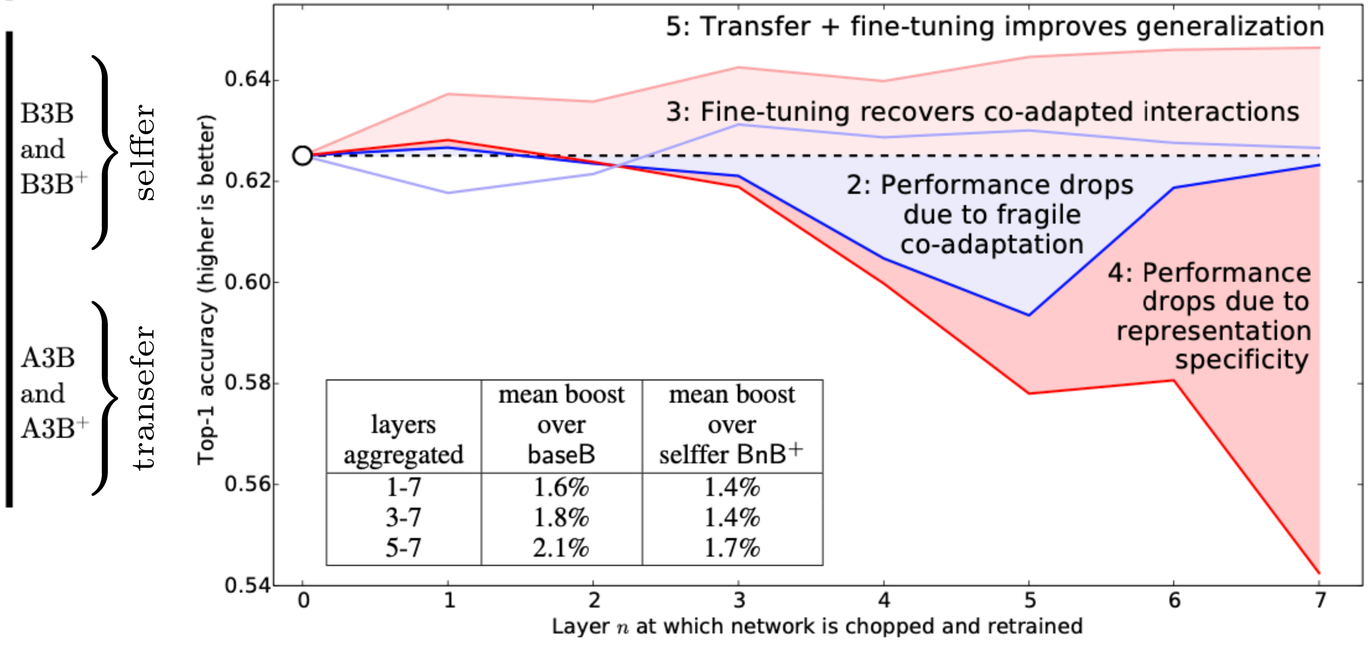
How transferable are features in deep neural networks?

YouTube Playlist



- Can we quantify the degree to which a particular layer is general or specific?
- Does the transition occur suddenly at a single layer, or is it spread out over several layers?
- Where does this transition take place: near the first, middle, or last layer of the network?





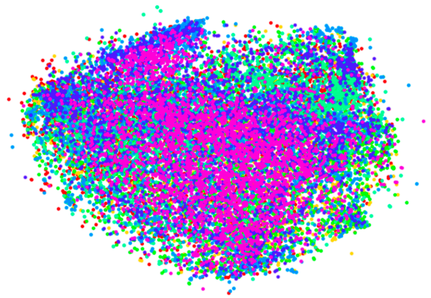
layers aggregated	mean boost over baseB	mean boost over selffer BnB+
1-7	1.6%	1.4%
3-7	1.8%	1.4%
5-7	2.1%	1.7%

Yosinski, Jason, et al. "How transferable are features in deep neural networks?." *Advances in neural information processing systems*. 2014.

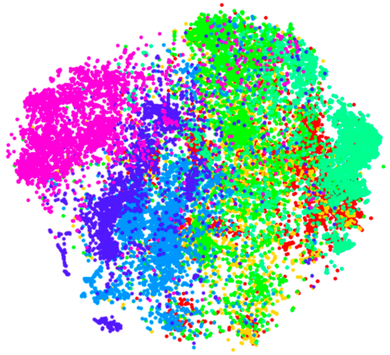
DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition

YouTube Playlist

t-SNE feature visualizations



DeCAF_1
first pooling layer

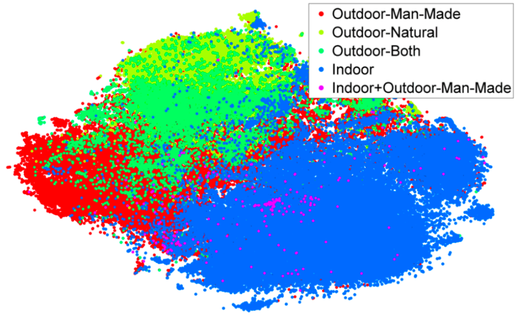


DeCAF_6
second to last hidden layer

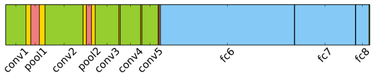
- structure, construction
- covering
- commodity, trade good, good
- conveyance, transport
- invertebrate
- bird
- hunting dog

First layers learn "low-level" features, whereas the latter layers learn semantic or "high-level" features.

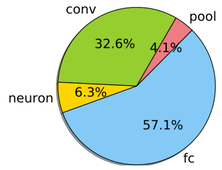
Features trained on ILSVRC-2012 generalized to SUN-397



Computation time



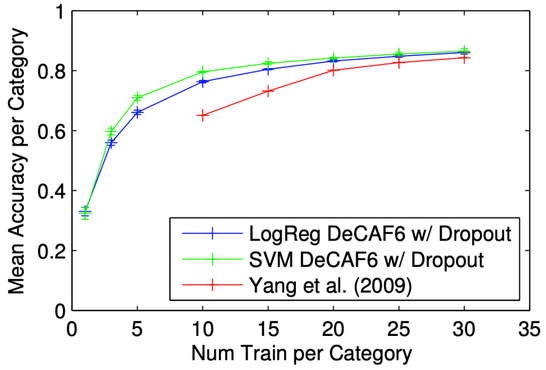
(a)



(b)

Object recognition (Caltech-101)

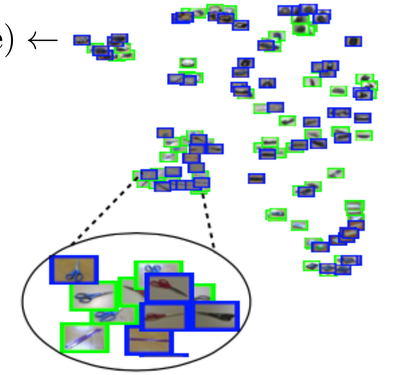
	DeCAF_5	DeCAF_6	DeCAF_7
LogReg	63.29 ± 6.6	84.30 ± 1.6	84.87 ± 0.6
LogReg with Dropout	-	86.08 ± 0.8	85.68 ± 0.6
SVM	77.12 ± 1.1	84.77 ± 1.2	83.24 ± 1.2
SVM with Dropout	-	$\mathbf{86.91 \pm 0.7}$	85.51 ± 0.9
Yang et al. (2009)		84.3	
Jarrett et al. (2009)		65.5	



Domain adaptation (Office dataset)

	Amazon → Webcam			Dslr → Webcam		
	SURF	DeCAF_6	DeCAF_7	SURF	DeCAF_6	DeCAF_7
Logistic Reg. (S)	9.63 ± 1.4	48.58 ± 1.3	53.56 ± 1.5	24.22 ± 1.8	88.77 ± 1.2	87.38 ± 2.2
SVM (S)	11.05 ± 2.3	52.22 ± 1.7	53.90 ± 2.2	38.80 ± 0.7	91.48 ± 1.5	89.15 ± 1.7
Logistic Reg. (T)	24.33 ± 2.1	72.56 ± 2.1	74.19 ± 2.8	24.33 ± 2.1	72.56 ± 2.1	74.19 ± 2.8
SVM (T)	51.05 ± 2.0	78.26 ± 2.6	78.72 ± 2.3	51.05 ± 2.0	78.26 ± 2.6	78.72 ± 2.3
Logistic Reg. (ST)	19.89 ± 1.7	75.30 ± 2.0	76.32 ± 2.0	36.55 ± 2.2	92.88 ± 0.6	91.91 ± 2.0
SVM (ST)	23.19 ± 3.5	80.66 ± 2.3	79.12 ± 2.1	46.32 ± 1.1	$\mathbf{94.79 \pm 1.2}$	92.96 ± 2.0
Daume III (2007)	40.26 ± 1.1	$\mathbf{82.14 \pm 1.9}$	81.65 ± 2.4	55.07 ± 3.0	91.25 ± 1.1	89.52 ± 2.2
Hoffman et al. (2013)	37.66 ± 2.2	80.06 ± 2.7	80.37 ± 2.0	53.65 ± 3.3	93.25 ± 1.5	91.45 ± 1.5
Gong et al. (2012)	39.80 ± 2.3	75.21 ± 1.2	77.55 ± 1.9	39.12 ± 1.3	88.40 ± 1.0	88.66 ± 1.9
Chopra et al. (2013)		58.85			78.21	

webcam (green) & dslr (blue) ←



Subcategory recognition (Caltech-UCSD bird dataset)

Method	Accuracy
DeCAF_6	58.75
DPD + DeCAF_6	**64.96**
DPD (Zhang et al., 2013)	50.98
POOF (Berg & Belhumeur, 2013)	56.78

Scene recognition (SUN-397)

	DeCAF_6	DeCAF_7
LogReg	$\mathbf{40.94 \pm 0.3}$	40.84 ± 0.3
SVM	39.36 ± 0.3	40.66 ± 0.3
Xiao et al. (2010)	38.0	

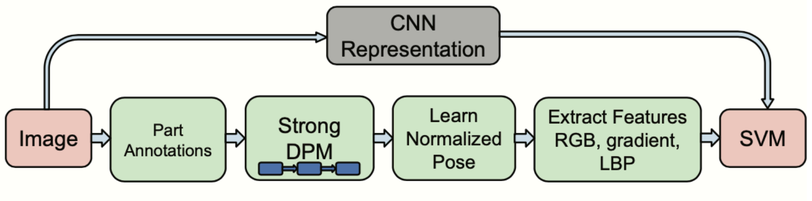
The dataset contains three domains: Amazon, which consists of product images taken from amazon.com; and Webcam and Dslr, which consist of images taken in an office environment using a webcam or digital SLR camera, respectively.

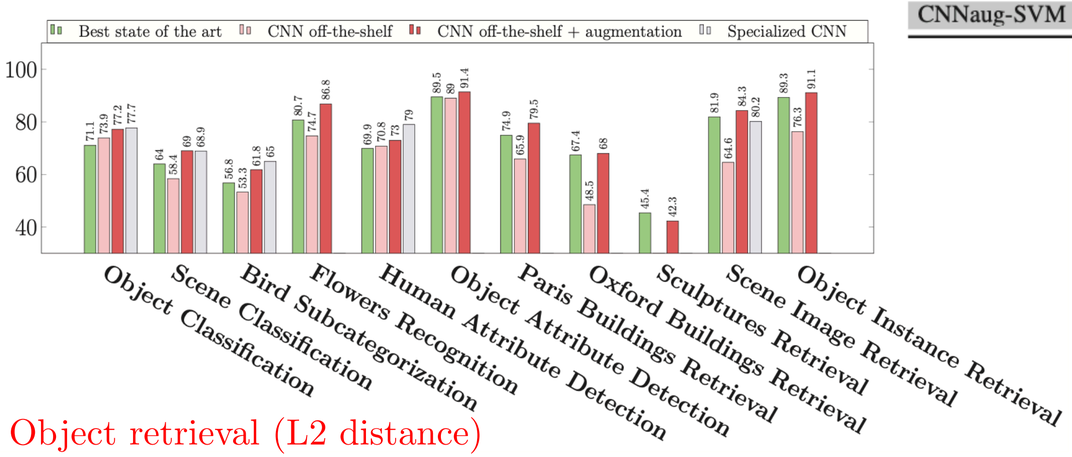
Donahue, Jeff, et al. "Decaf: A deep convolutional activation feature for generic visual recognition." *International conference on machine learning*. 2014.

CNN Features off-the-shelf: an Astounding Baseline for Recognition

Boulder

YouTube Playlist



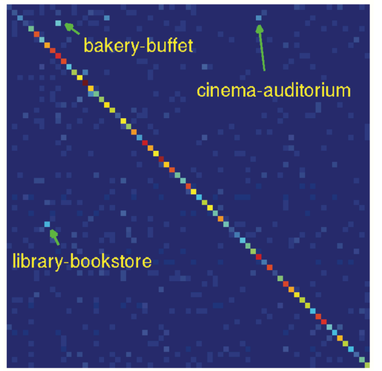


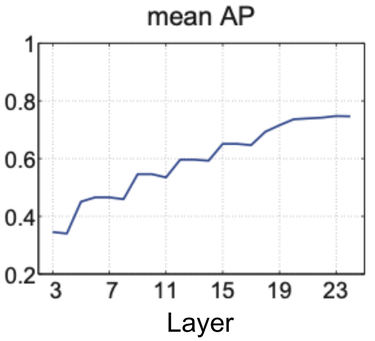
Pascal VOC 2007 Image Classification

	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
GHM[8]	76.7	74.7	53.8	72.1	40.4	71.7	83.6	66.5	52.5	57.5	62.8	51.1	81.4	71.5	86.5	36.4	55.3	60.6	80.6	57.8	64.7
AGS[11]	82.2	83.0	58.4	76.1	**56.4**	**77.5**	**88.8**	69.1	**62.2**	61.8	64.2	51.3	**85.4**	**80.2**	91.1	48.1	61.7	**67.7**	86.3	70.9	71.1
NUS[39]	82.5	79.6	64.8	73.4	54.2	75.0	77.5	79.2	46.2	62.7	41.4	74.6	85.0	76.8	91.1	53.9	61.0	67.5	83.6	70.6	70.5
CNN-SVM	88.5	81.0	83.5	82.0	42.0	72.5	85.3	81.6	59.9	58.5	66.5	77.8	81.8	78.8	90.2	54.8	71.1	62.6	87.2	71.8	73.9
CNNaug-SVM	**90.1**	**84.4**	**86.5**	**84.1**	48.4	73.4	86.7	**85.4**	61.3	**67.6**	**69.6**	**84.0**	**85.4**	80.0	**92.0**	**56.9**	**76.7**	67.3	**89.1**	**74.9**	**77.2**

MIT-67 indoor scenes dataset

Method	mean Accuracy
ROI + Gist[36]	26.1
DPM[30]	30.4
Object Bank[24]	37.6
RBow[31]	37.9
BoP[21]	46.1
miSVM[25]	46.4
D-Parts[40]	51.4
IFV[21]	60.8
MLrep[9]	64.0
CNN-SVM	58.4
CNNaug-SVM	**69.0**
CNN(AlexConvNet)+multiscale pooling [16]	68.9





mean accuracy: mean of the confusion matrix diagonal

CNNaug: augment the training set by adding cropped and rotated samples

Object retrieval (L2 distance)

	Dim	Oxford5k	Paris6k	Sculp6k	Holidays	UKBench
BoB[3]	N/A	N/A	N/A	**45.4**[3]	N/A	N/A
BoW	200k	36.4[20]	46.0[35]	8.1[3]	54.0[4]	70.3[20]
IFV[33]	2k	41.8[20]	-	-	62.6[20]	83.8[20]
VLAD[4]	32k	55.5 [4]	-	-	64.6[4]	-
CVLAD[52]	64k	47.8[52]	-	-	81.9[52]	89.3[52]
HE+burst[17]	64k	64.5[42]	-	-	78.0[42]	-
AHE+burst[17]	64k	66.6[42]	-	-	79.4[42]	-
Fine vocab[26]	64k	74.2[26]	74.9[26]	-	74.9[26]	-
ASMK*+MA[42]	64k	80.4[42]	77.0[42]	-	81.0[42]	-
ASMK+MA[42]	64k	**81.7**[42]	78.2[42]	-	82.2[42]	-
CNN	4k	32.2	49.5	24.1	64.2	76.0
CNN-ss	32-120k	55.6	69.7	31.1	76.9	86.9
CNNaug-ss	4-15k	**68.0**	**79.5**	42.3	**84.3**	**91.1**
CNN+BOW[16]	2k	-	-	-	**80.2**	-

CUB 200-2011 Bird dataset

Method	Part info	mean Accuracy
Sift+Color+SVM[45]	✗	17.3
Pose pooling kernel[49]	✓	28.2
RF[47]	✓	19.2
DPD[50]	✓	51.0
Poof[5]	✓	56.8
CNN-SVM	✗	53.3
CNNaug-SVM	✗	**61.8**
DPD+CNN(DeCaf)+LogReg[10]	✓	**65.0**

H3D Human Attributes dataset

Method	male	lg hair	glasses	hat	tshirt	lg slvs	shorts	jeans	lg pants	mAP
Freq[6]	59.3	30.0	22.0	16.6	23.5	49.0	17.9	33.8	74.7	36.3
SPM[6]	68.1	40.0	25.9	35.3	30.6	58.0	31.4	39.5	84.3	45.9
Poselets[6]	82.4	**72.5**	**55.6**	60.1	51.2	74.2	45.5	54.7	90.3	65.2
DPD[50]	83.7	70.0	38.1	**73.4**	49.8	78.1	64.1	**78.1**	93.5	69.9
CNN-SVM	83.0	67.6	39.7	66.8	52.6	82.2	78.2	71.7	95.2	70.8
CNNaug-SVM	**84.8**	71.0	42.5	66.9	**57.7**	**84.0**	**79.1**	75.7	**95.3**	**73.0**

Sharif Razavian, Ali, et al. "CNN features off-the-shelf: an astounding baseline for recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2014.

Return of the Devil in the Details: Delving Deep into Convolutional Nets

YouTube Video

$I \rightarrow$ image
$\phi \rightarrow$ encoding function
$\phi(I) \in \mathbb{R}^d \rightarrow$ vector image representation

Shallow Representation (IFV)

Improved Fisher Vector (IFV)

– Extract a dense collection of patches and corresponding local descriptors $x_i \in \mathbb{R}^D$ (e.g., SIFT) from the image at multiple scales

– Each descriptor x_i is then soft-quantized (i.e., clustered) using a Gaussian Mixture Model with K components.

$$u_k := \sum_{\{x_i : NN(x_i) = \mu_k\}} (x_i - \mu_k) \in \mathbb{R}^D$$

Nearest Neighbor

accumulated first order differences

$$v_k := \sum_{\{x_i : NN(x_i) = \mu_k\}} (x_i - \mu_k)^2 \in \mathbb{R}^D$$

accumulated second order differences

$$\phi_{\text{FV}}(I) = [u_1; v_1; \ldots; u_K; v_K] \in \mathbb{R}^{2KD}$$

$$\phi_{\text{IFV}}(I) \leftarrow \phi_{\text{FV}}(I) \triangleright \text{sign}(\cdot)\sqrt{|\cdot|} \triangleright \ell_2 \text{ normalize}$$

Deep representation (CNN) with pre-training

$\phi_{\text{CNN}}(I) \rightarrow$ vector activities of penultimate layer

Deep representation (CNN) with pre-training and fine-tuning

	ILSVRC-2012 (top-5 error)	VOC-2007 (mAP)	VOC-2012 (mAP)	Caltech-101 (accuracy)	Caltech-256 (accuracy)
(a) FK IN 512	-	68.0	–	–	–
(b) CNN F (Fast)	16.7	77.4	79.9	–	–
(c) CNN M (Medium)	13.7	79.9	82.5	87.15 ± 0.80	77.03 ± 0.46
(d) CNN M 2048	13.5	80.1	82.4	86.64 ± 0.53	76.88 ± 0.35
(e) CNN S (Slow)	**13.1**	79.7	82.9	87.76 ± 0.66	**77.61 ± 0.12**
(f) CNN S TUNE-CLS	**13.1**	-	83.0	**88.35 ± 0.56**	77.33 ± 0.56
(g) CNN S TUNE-RNK	**13.1**	**82.4**	**83.2**	–	–
(h) Zeiler & Fergus [19]	16.1	-	79.0	86.5 ± 0.5	74.2 ± 0.3
(i) Razavian *et al.* [9], [10]	14.7	77.2	–	–	–
(j) Oquab *et al.* [8]	18	77.7	78.7 (82.8*)	–	–
(k) Oquab *et al.* [16]	-	-	**86.3***	–	–
(l) Wei *et al.* [17]	-	81.5 (**85.2***)	81.7 (**90.3***)	–	–
(m) He *et al.* [29]	13.6	80.1	-	**91.4 ± 0.7**	–

Pascal VOC

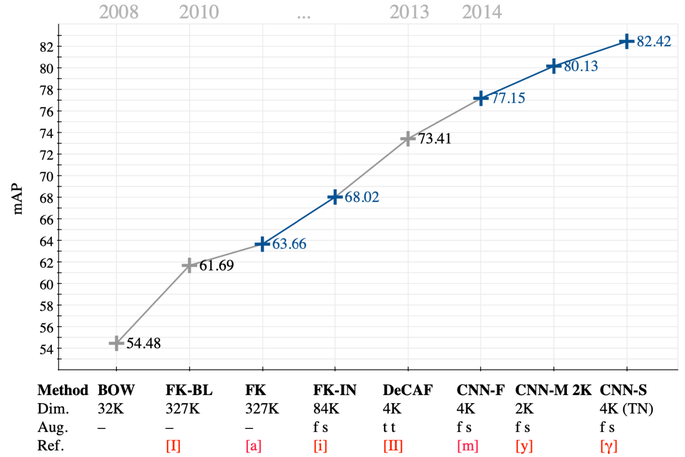
multi-label dataset

$I_{\text{pos}}, I_{\text{neg}} \rightarrow$ positive, negative images for each class c

one-vs-rest classification hing loss

$w_c^T \phi(I_{\text{pos}}) > 1 - \xi, w_c^T \phi(I_{\text{neg}}) < -1 + \xi \rightarrow$ slack

ranking hing loss $\quad w_c^T \phi(I_{\text{pos}}) > w_c^T \phi(I_{\text{neg}}) + 1 - \xi$



Method	SPool	Image Aug.	Dim	mAP
(I) FK BL	spm	–	327K	**61.69**
(II) DECAF	–	(C) t t	327K	**73.41**
(a) FK	spm	–	327K	**63.66**
(b) FK IN (Intra-normalisation)	spm	–	327K	**64.18**
(c) FK	(x,y)	–	42K	**63.51**
(d) FK IN	(x,y)	–	42K	**64.36**
(e) FK IN	(x,y)	(F) f -	42K	**64.35**
(f) FK IN	(x,y)	(C) f s	42K	**67.17**
(g) FK IN	(x,y)	(C) s s	42K	**66.68**
(h) FK IN 512	(x,y)	–	84K	**65.36**
(i) FK IN 512	(x,y)	(C) f s	84K	**68.02**
(j) FK IN COL 512	–	–	82K	**52.18**
(k) FK IN 512 COL+	(x,y)	–	166K	**66.37**
(l) FK IN 512 COL+	(x,y)	(C) f s	166K	**67.93**
(m) CNN F	–	(C) f s	4K	**77.38**
(n) CNN S	–	(C) f s	4K	**79.74**

Performance of shallow representations can be significantly improved by adopting data augmentation, typically used in deep learning. In spite of this improvement, deep architectures still outperform the shallow methods by a large margin.

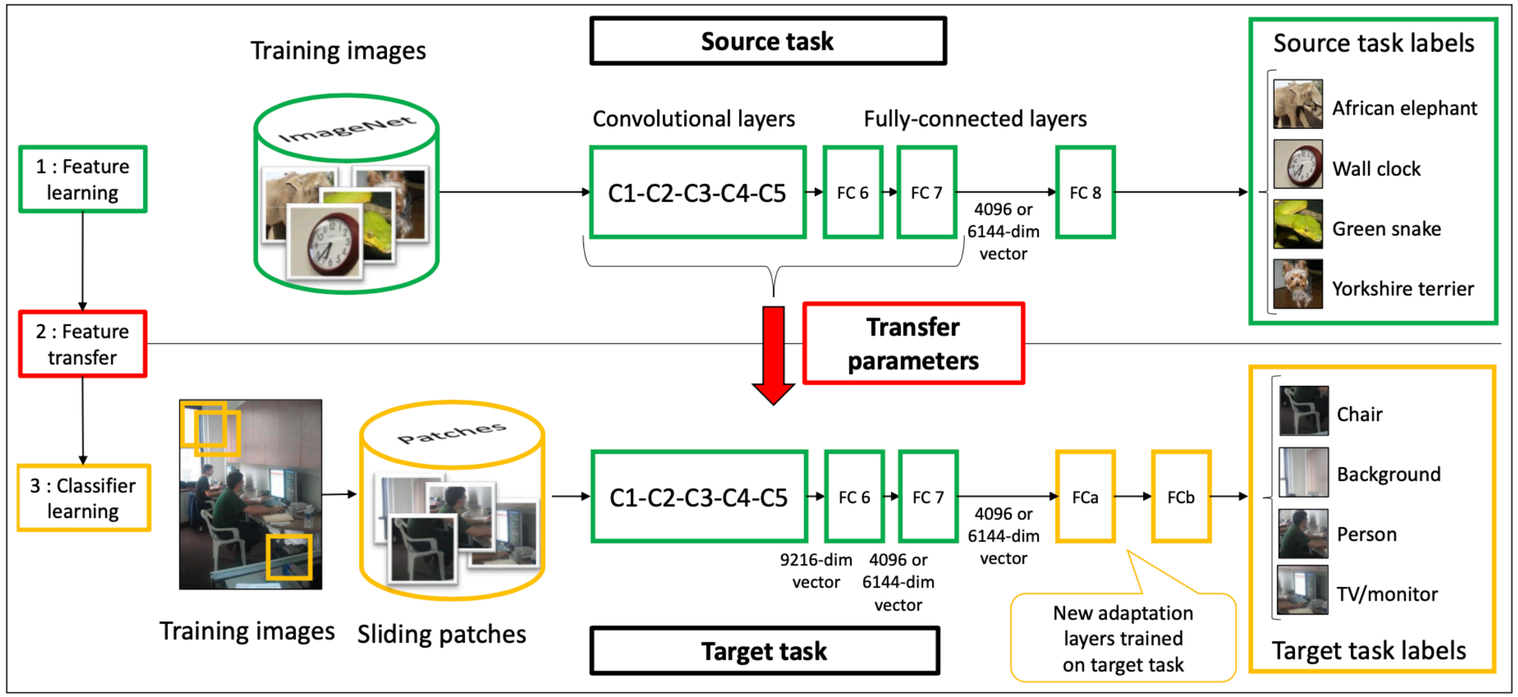
(x) CNN M 2048	–	(C) f s	2K	**80.10**
(y) CNN M 1024	–	(C) f s	1K	**79.91**
(z) CNN M 128	–	(C) f s	128	**78.60**
(α) FK+CNN F	(x,y)	(C) f s	88K	**77.95**
(β) FK+CNN M 2048	(x,y)	(C) f s	86K	**80.14**
(γ) CNN S TUNE-RNK	–	(C) f s	4K	**82.42**

Chatfield, Ken, et al. "Return of the devil in the details: Delving deep into convolutional nets." *arXiv preprint arXiv:1405.3531* (2014).

Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks

YouTube Video

Transfer: ImageNet → Pascal VOC (Limited Training Data)



ImageNet

Pascal VOC

"dataset capture bias" and "negative data bias"

Inference

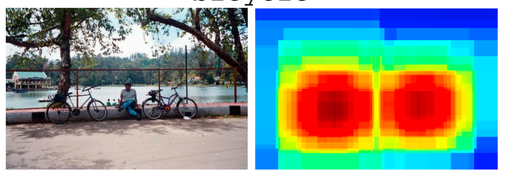
$$\text{score}(C_n) = \frac{1}{M}\sum_{i=1}^{M} y(C_n|P_i)^k$$

Output of the network for class C_n on image patch P_i

$M \rightarrow$ number of patches in the image

$k \geq 1$ $(k = 5) \rightarrow$ higher values of k focus on highest scoring patches and attenuate the contributions of low- and mid-scoring patches

bicycle



bicycle

Network Architecture: AlexNet
Input: 224×224 RGB Image
Output: Distribution over the ImageNet object classes
Label Bias: ImageNet → husky dog, australian terrier, etc.
Pascal VOC → dog
Pascal VOC → reading, running, etc.
500 square patches per each image
at least 50% overlap between neighboring patches
$\lambda \in \{1, 1.3, 1.6, 2, 2.4, 2.8, 3.2, 3.6, 4\} \rightarrow$ 8 different scales

$s = \min(w, h)/\lambda$ pixels
↳ width of square patches
rescale each patch to 224×224
$P \rightarrow$ bounding box of a patch
$B_o \rightarrow$ ground truth bounding box for class o
1. $|P \cap B_o| \geq 0.2|P| \rightarrow B_o$ overlaps sufficiently with the patch
2. $|P \cap B_o| \geq 0.6|B_o| \rightarrow$ the patch contains large portion of the object
3. the patch overlaps with no more than one object
$\Longrightarrow$ the patch is labeled as a positive label example for class o

Oquab, Maxime, et al. "Learning and transferring mid-level image representations using convolutional neural networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.

Boulder

Questions?

YouTube Playlist