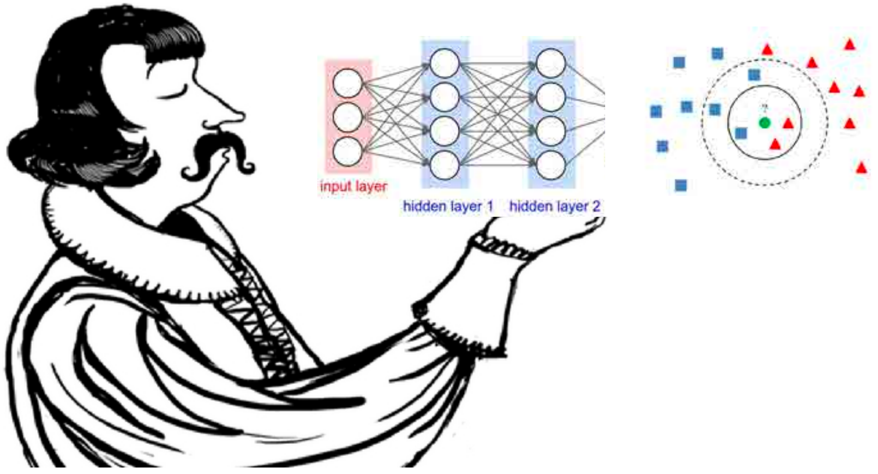
The Need for Geometric Regularization: Theory and Examples in Image Classification and Face/Person Challenges

Guillermo Sapiro
Duke University

DNN or kNN:
That is the Generalize vs Memorize Question

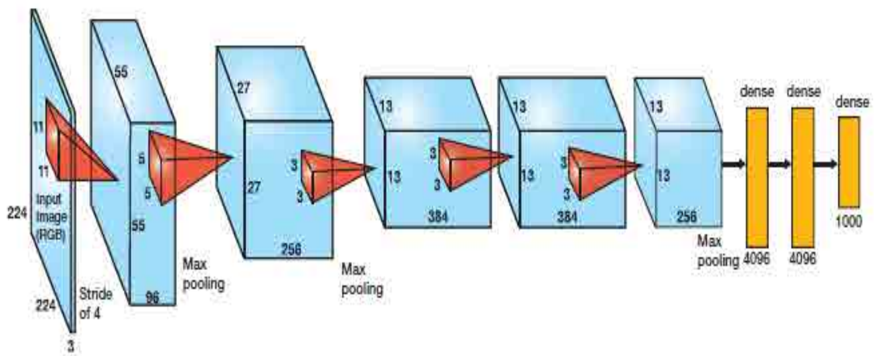
Guillermo Sapiro
Duke University

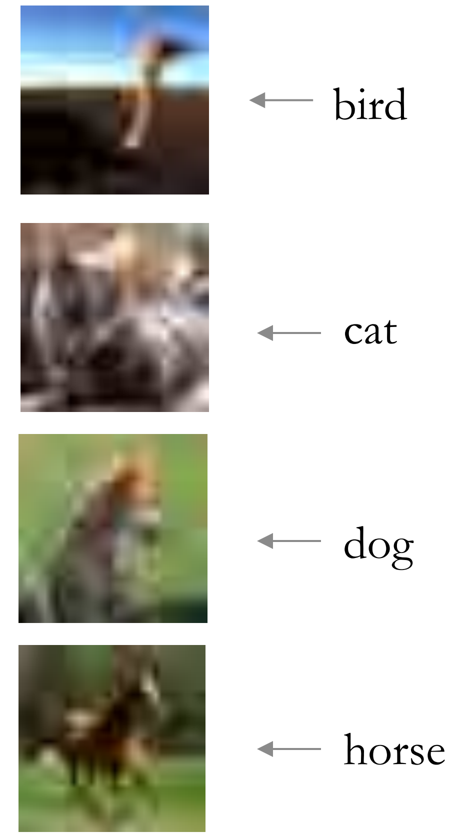
With Gilad Cohen and Raja Giryes



Object Recognition using Deep Learning

Alexnet
(1,387,786 parameters)





Train accuracy: 99.90%

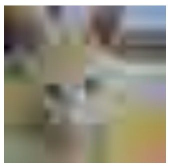
Test accuracy: 81.22%

Krizhevsky, Alex and Sutskever, Ilya and Hinton, Geoffrey E, ImageNet Classification with Deep Convolutional Neural Networks, NIPS 12

Object Recognition using Deep Learning

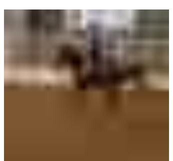
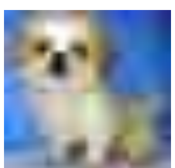

bird

horse


airplane

dog

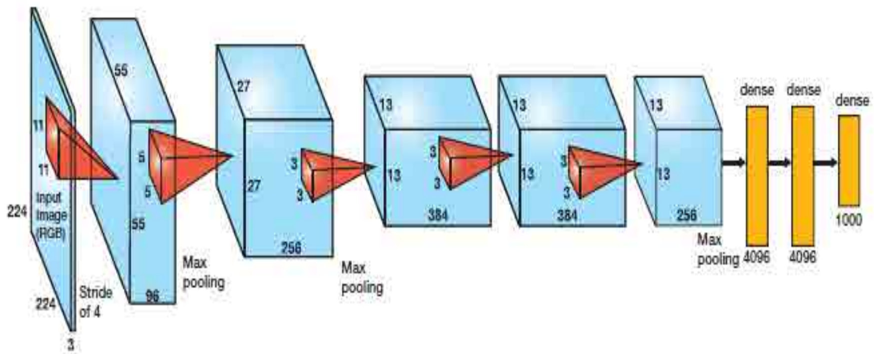

bird

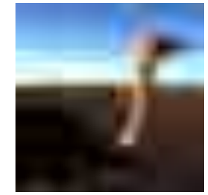
cat

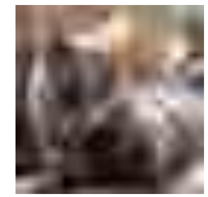

dog

airplane

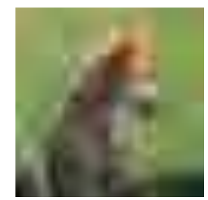
Alexnet

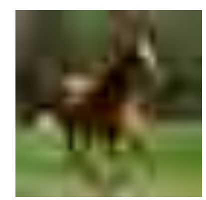
(1,387,786 parameters)




← ?


← ?

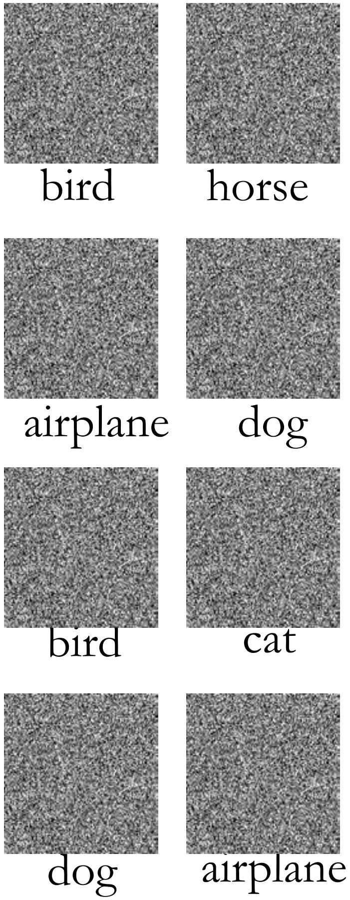

← ?


← ?

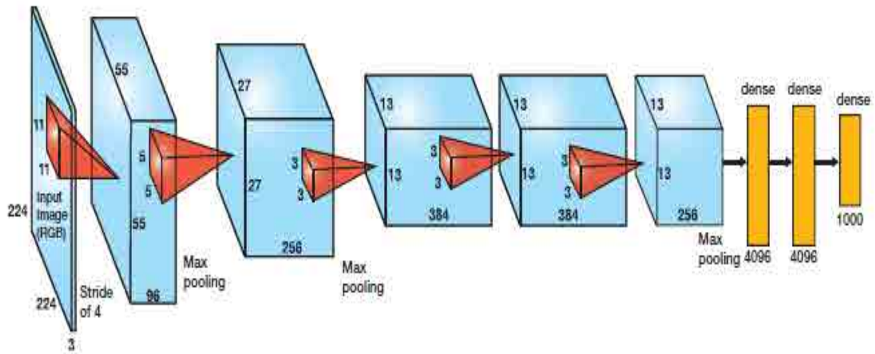
Train accuracy: 99.82%

Test accuracy: 9.86%

Object Recognition using Deep Learning



Alexnet
(1,387,786 parameters)



Train accuracy: 

Test accuracy: 

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, Oriol Vinyals. Understanding deep learning requires rethinking generalization. ICLR 2017

Goal and Challenges

- Explain the excellent generalization of DNNs
- Are DNNs memorizing or generalizing?
- Are DNNs “just” good kNNs?

- **Contributions**
  - DNNs are kNNs in a learned feature space
  - DNNs both memorize and generalize
    - *Memorize* the training set and *generalize* via kNN
  - DNNs might be Bayesian optimal

Experimental Study

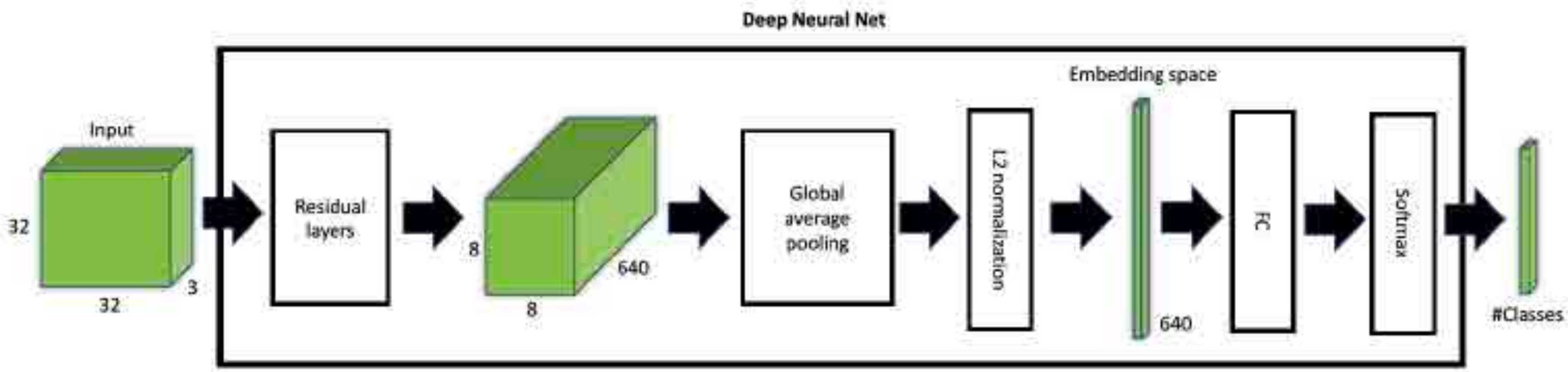
- CIFAR 10/100 and MNIST
- Wide-Resnet 28-10, LeNet, multilayer perception (MLP)

$$MC \triangleq p\big(f_{kNN}(s) = l \,|\, f_{DNN}(s) = l\big),$$

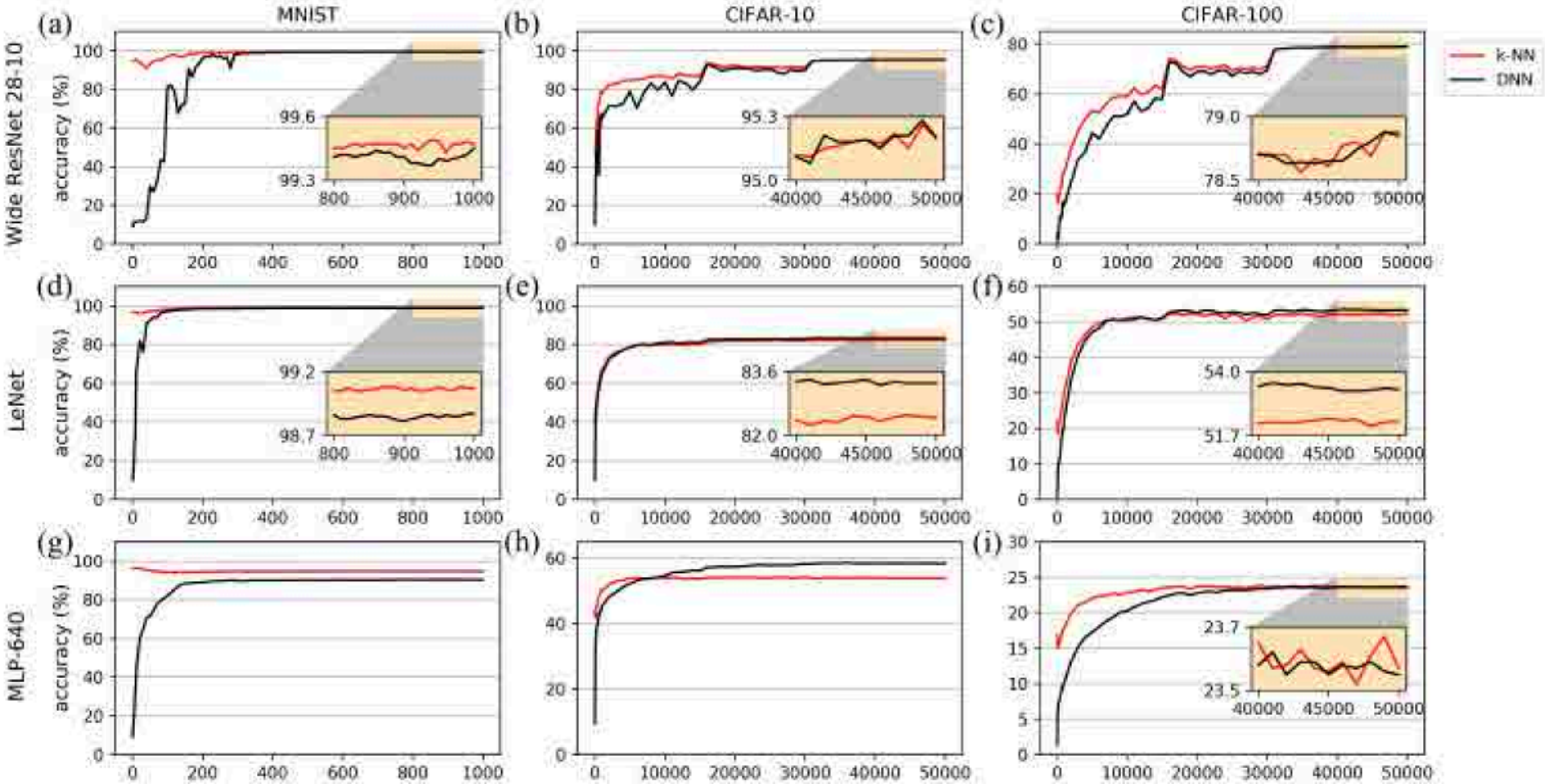
$$ME \triangleq p\big(f_{kNN}(s) = f_{DNN}(s) \,|\, f_{DNN}(s) \neq l\big),$$

$$\begin{aligned} P_{SAME} &\triangleq p\big(f_{kNN}(s) = f_{DNN}(s)\big) \\ &= p\big(f_{kNN}(s) = l \,|\, f_{DNN}(s) = l\big) p\big(f_{DNN}(s) = l\big) \\ &\quad + p\big(f_{kNN}(s) = f_{DNN}(s) \,|\, f_{DNN}(s) \neq l\big) p\big(f_{DNN}(s) \neq l\big) \\ &= MC \times acc + ME \times (1 - acc). \end{aligned}$$

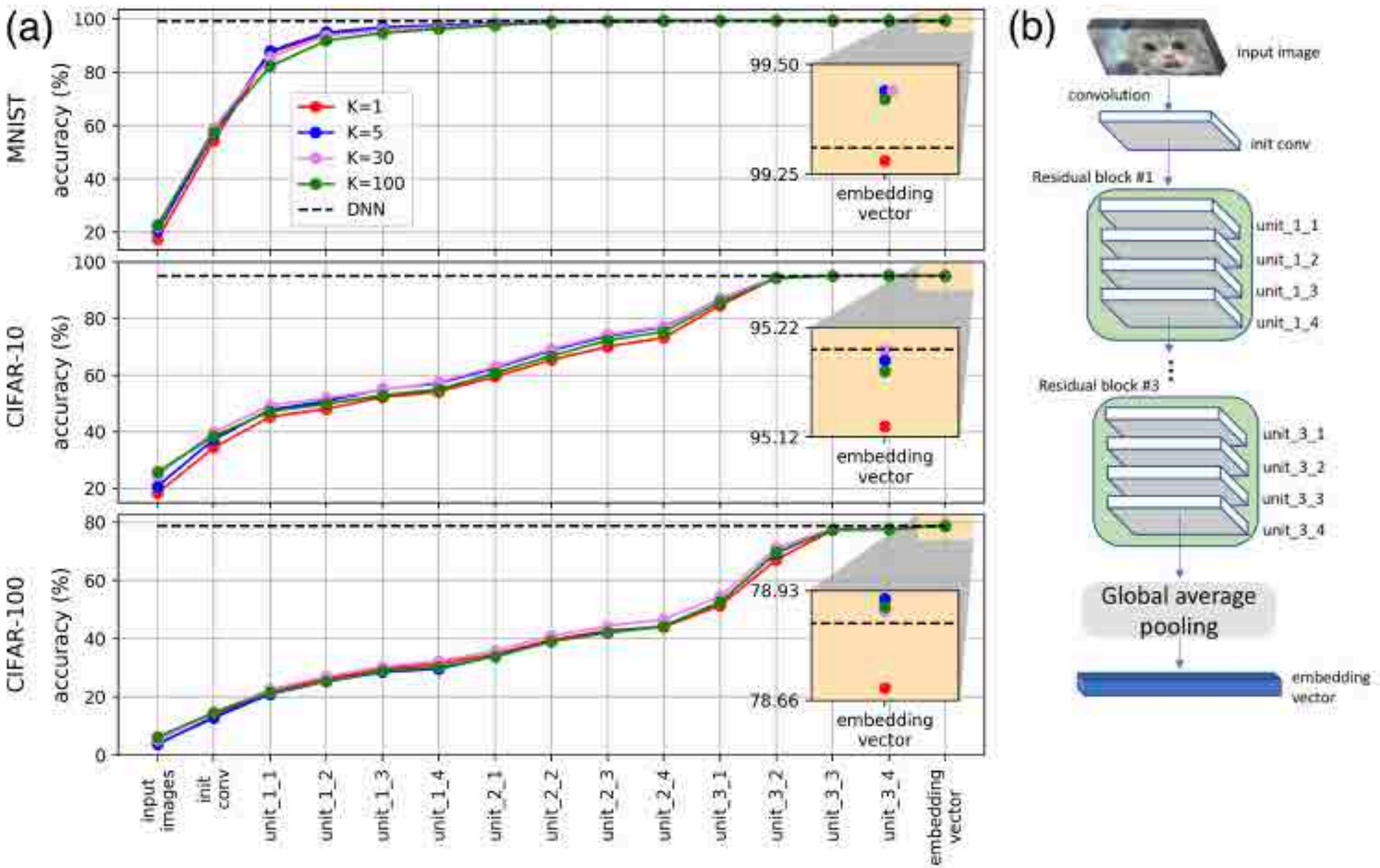
Experimental Study (cont.)



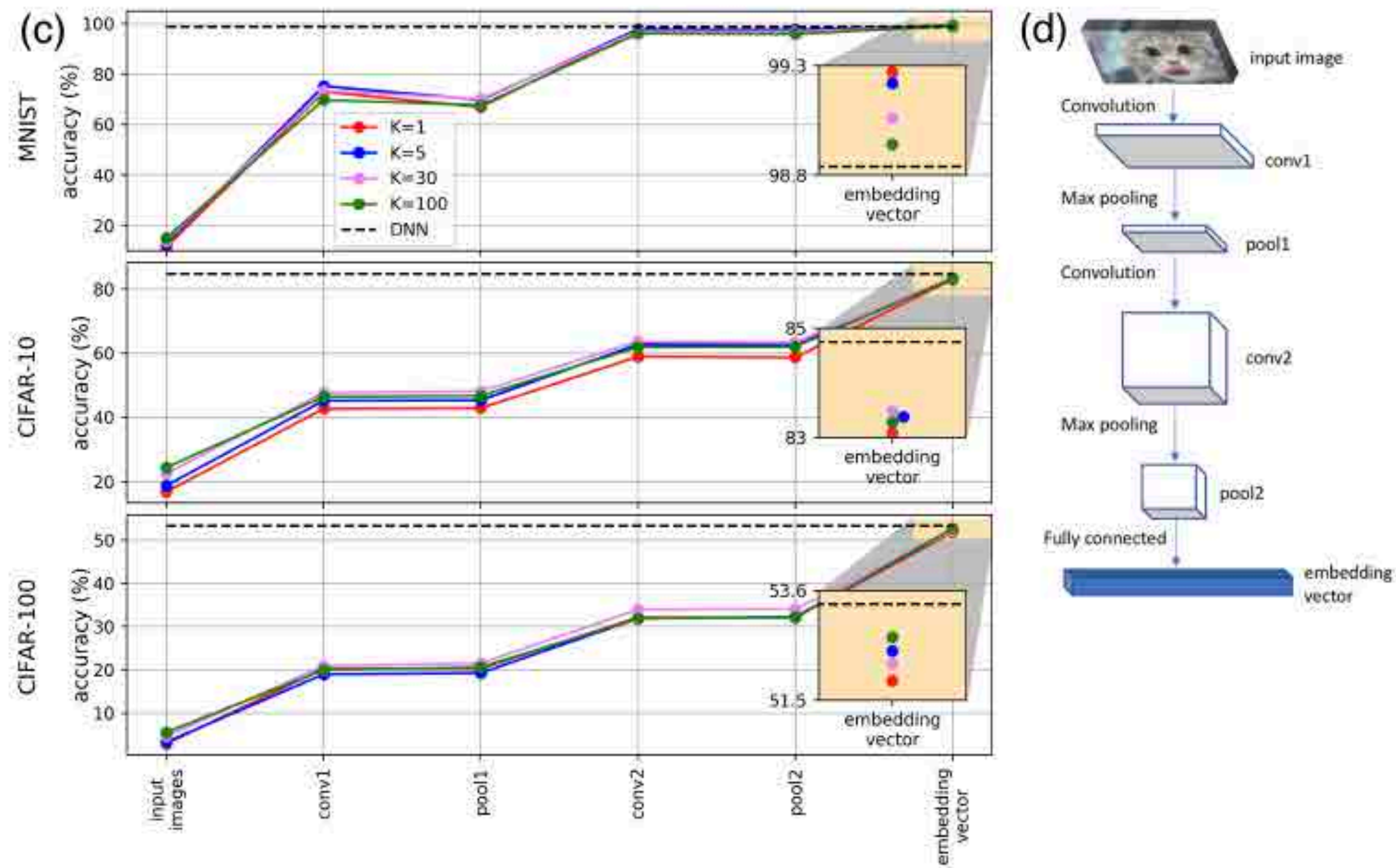
Accuracy as a Function of Training Step



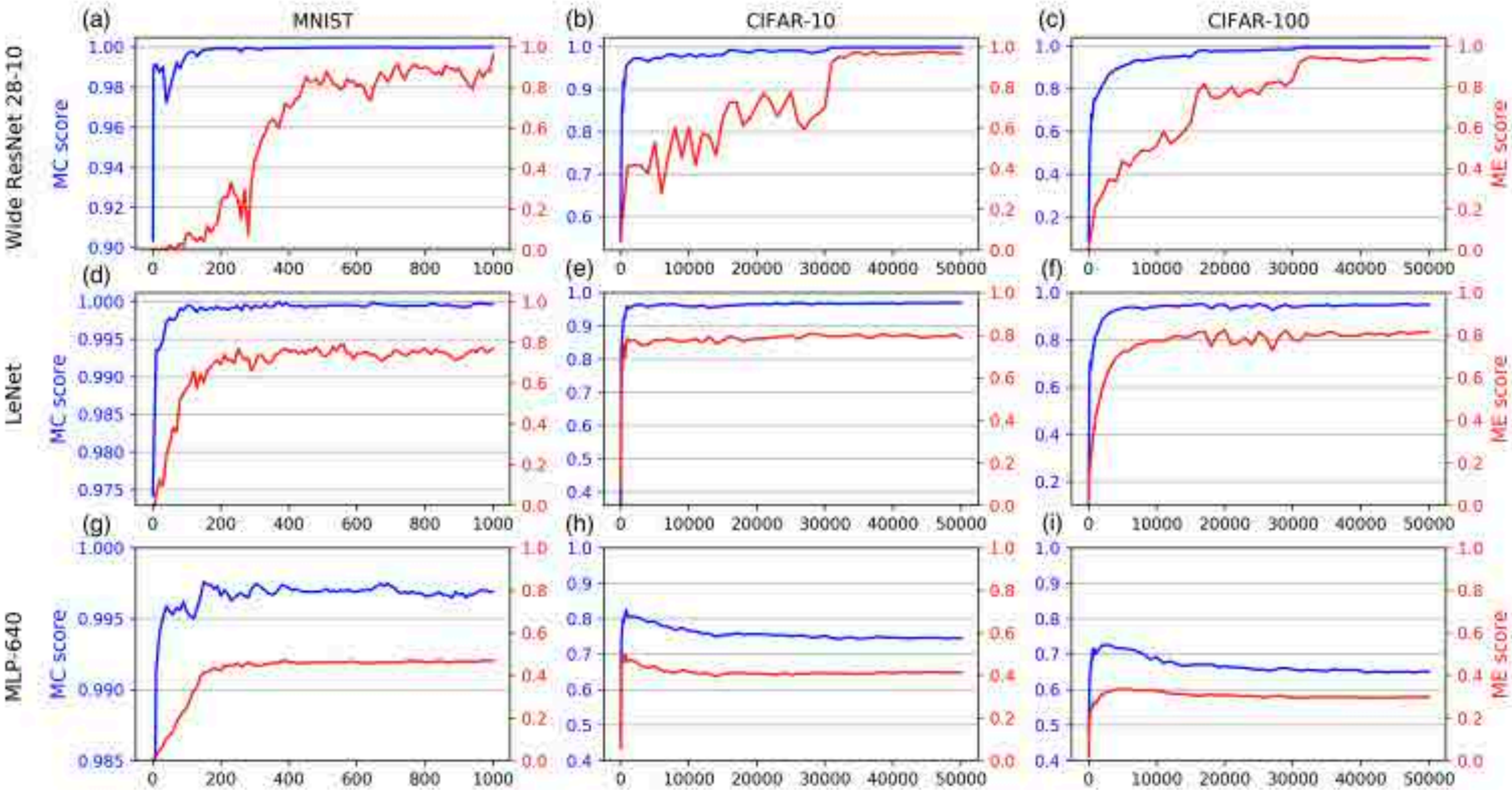
Accuracy as a Function of Layer



Accuracy as a Function of Layer

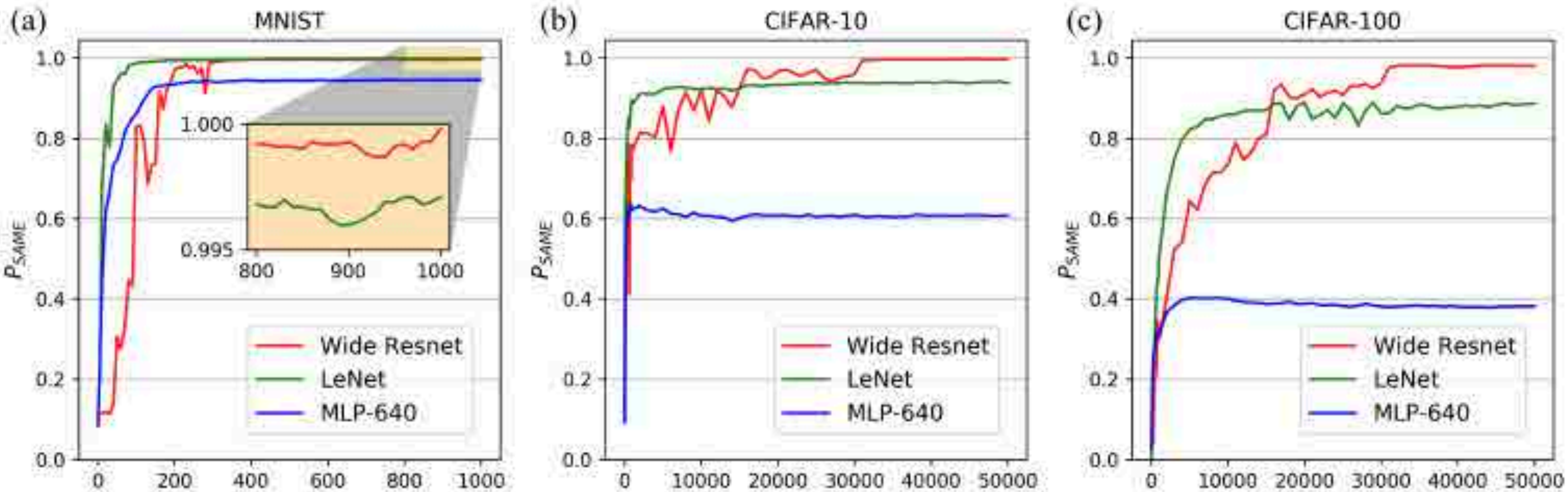


MC and ME



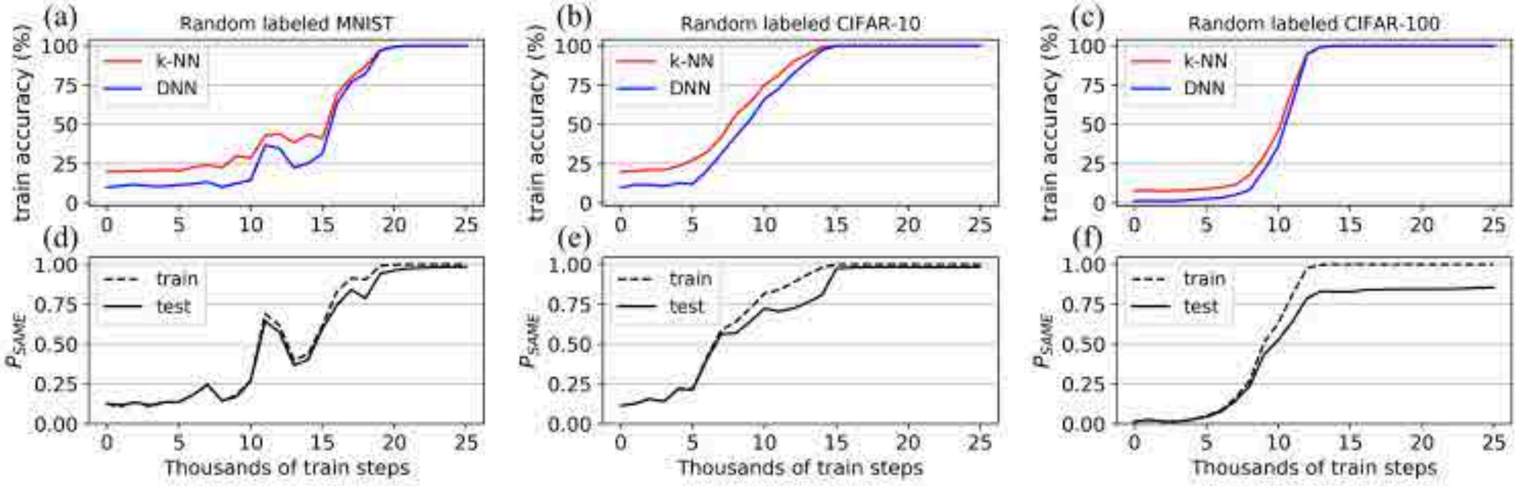
(a)
MNIST
(b)
CIFAR-10
(c)
CIFAR-100
(d)
(e)
(f)
(g)
(h)
(i)
Wide ResNet 28-10
LeNet
MLP-640
MC score
ME score

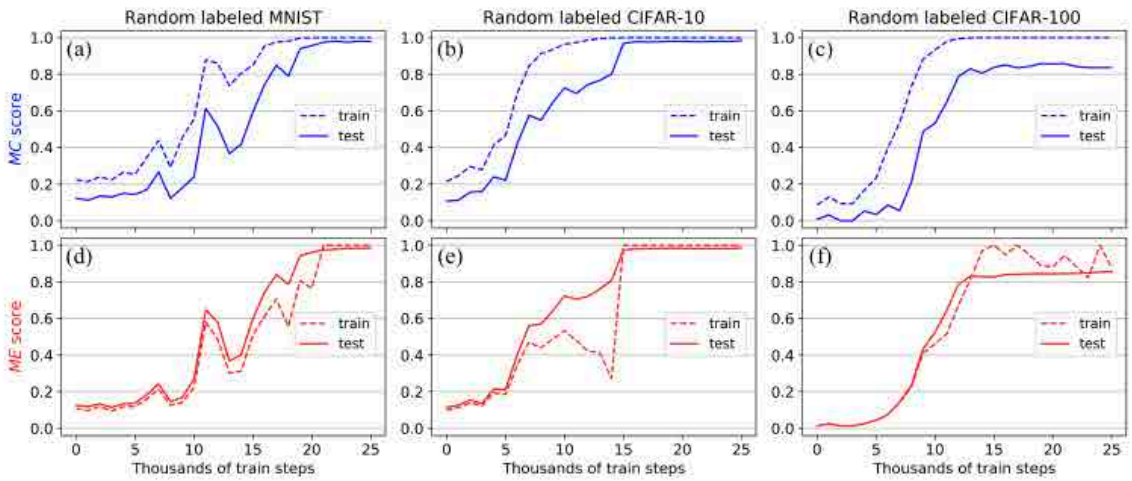
P_{SAME}



(a)
MNIST
(b)
CIFAR-10
(c)
CIFAR-100
P_{SAME}
1.000
0.995
800
900
1000
Wide Resnet
LeNet
MLP-640
0.0
0.2
0.4
0.6
0.8
1.0
0
200
400
600
800
1000
10000
20000
30000
40000
50000

Training Performance on Random Labels





Discussion

- DNN and kNN are "scary" similar
- DNNs both memorize and generalize
- kNN approach Bayes optimum

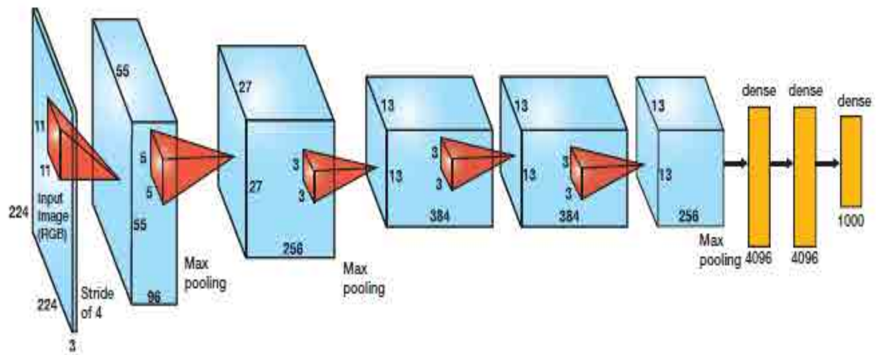
$$E^* \leq E \leq E^*\left(2 - \frac{ME^*}{M-1}\right)$$

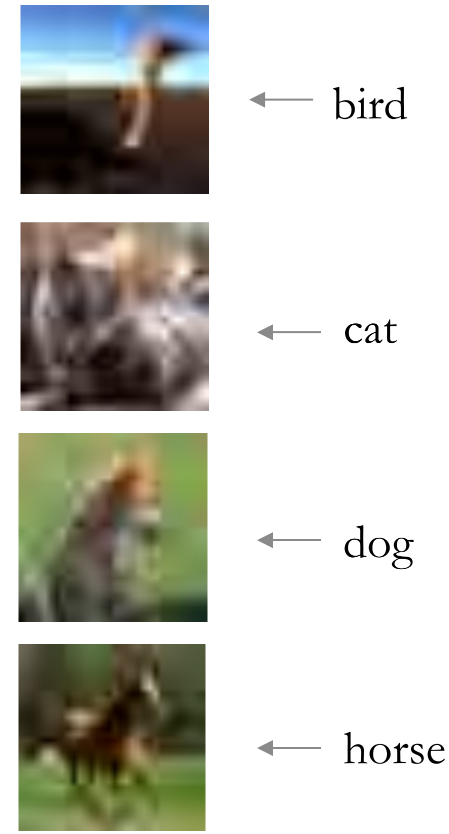
  - Is DNN Bayes optimum?

Regularized Deep Learning with Geometry and Structures

Object Recognition using Deep Learning

Alexnet
(1,387,786 parameters)





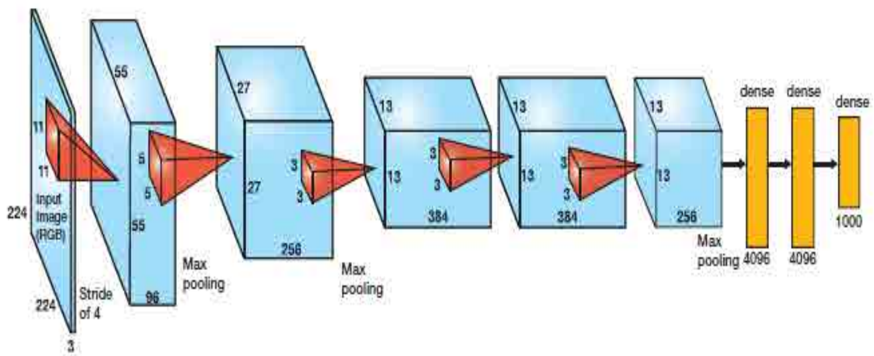
Train accuracy: 99.90%

Test accuracy: 81.22%

Krizhevsky, Alex and Sutskever, Ilya and Hinton, Geoffrey E, ImageNet Classification with Deep Convolutional Neural Networks, NIPS 12

Object Recognition using Deep Learning

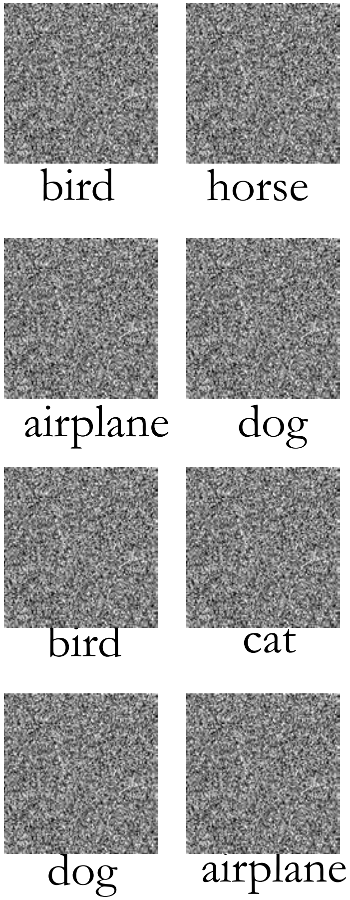
Alexnet
(1,387,786 parameters)



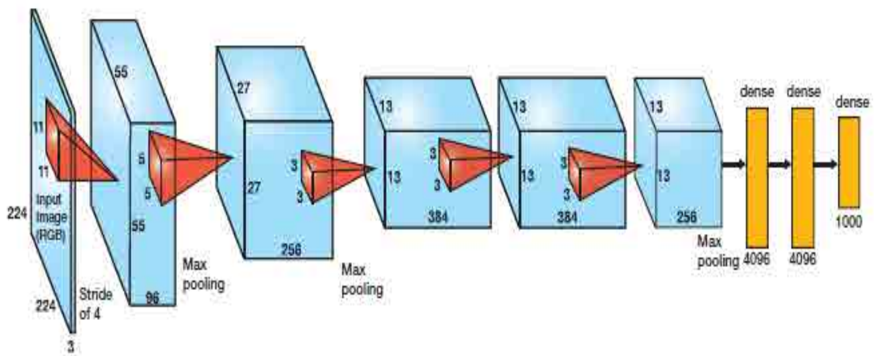
Train accuracy: 99.82%

Test accuracy: 9.86%

Object Recognition using Deep Learning



Alexnet
(1,387,786 parameters)



Train accuracy: 

Test accuracy: 

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, Oriol Vinyals. Understanding deep learning requires rethinking generalization. International Conference on Learning Representations (ICLR), Best Paper Award, 2017

Regularizing Deep Learning

Regularizing with data geometry:

- Low-rank subspace.
- Low-dimensional manifold.

Regularizing with structures imposed over and across convolutional filters.

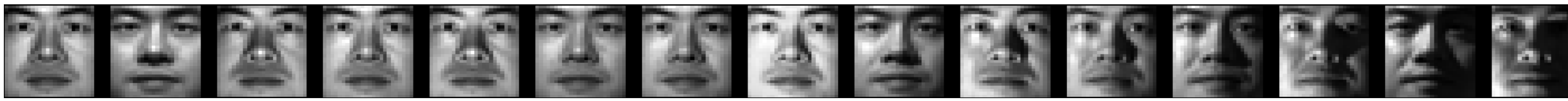
Regularizing with Data Geometry

Low-rank Subspace

9D linear subspace

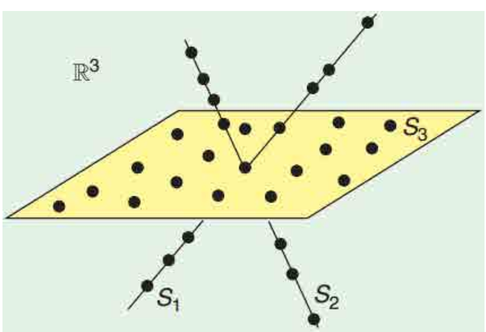
R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 2, pp. 218-233, Feb 2003.

$\mathbf{Y}_1 = [$  $]$

$\mathbf{Y}_2 = [$  $]$

$\mathbf{Y}_3 = [$  $]$

$$\mathbf{Y} = [\,\mathbf{Y}_1 \quad \mathbf{Y}_2 \quad \mathbf{Y}_3\,]$$



Orthogonal Low-rank Transform

$$\arg\min_{\mathbf{T}} \sum_{c=1}^{C} ||\mathbf{T}\mathbf{Y}_c||_* - ||\mathbf{T}\mathbf{Y}||_*$$

$|\mathbf{X}|_*$ denotes the nuclear norm of the matrix $\mathbf{X}$:

- The sum of the singular values of $\mathbf{A}$.
- A good approximation to the matrix rank.

Qiang Qiu, Guillermo Sapiro, "Learning Transformations for Clustering and Classification", Journal of Machine Learning Research (JMLR), 16(Feb):187−225, 2015

Orthogonal Low-rank Transform

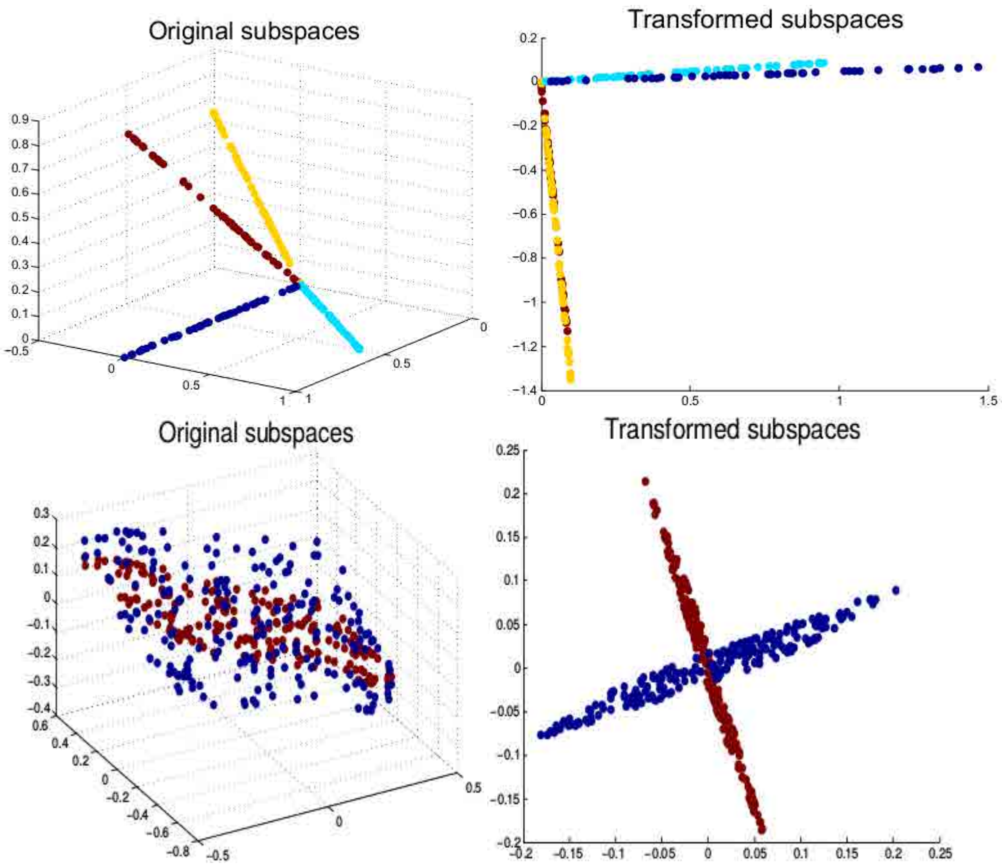
$$\arg\min_{\mathbf{T}} \sum_{c=1}^{C} ||\mathbf{T}\mathbf{Y}_c||_* - ||\mathbf{T}\mathbf{Y}||_*$$

Theorem

$||[\mathbf{A}, \mathbf{B}]||_* \leq ||\mathbf{A}||_* + ||\mathbf{B}||_*$, equality is satisfied iff $\mathbf{A}$ and $\mathbf{B}$ are orthogonal.

Simultaneously

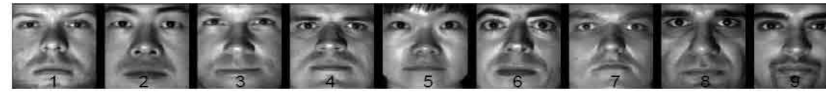
- reduces intra-class variance
- maximize inter-class margin



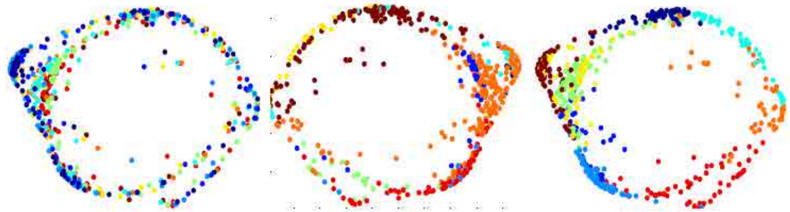
Qiang Qiu, Guillermo Sapiro, "Learning Transformations for Clustering and Classification", Journal of Machine Learning Research (JMLR), 16(Feb):187−225, 2015

Face Clustering

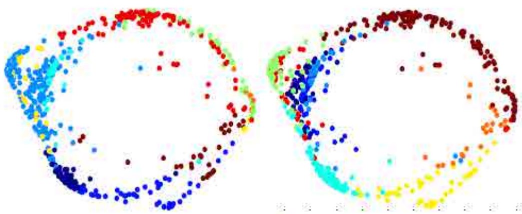
(a) Example illumination conditions.



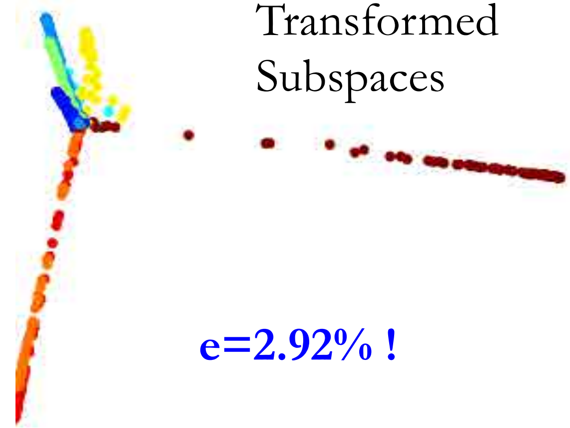
(b) Example subjects.



(a) Ground truth. (b) SSC, $e = 71.25\%$, $t = 714.99$ sec. (c) LBF, $e = 76.37\%$, $t = 460.76$ sec.



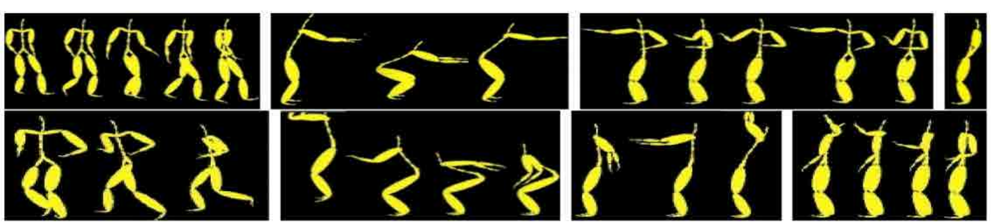
(d) LSA, $e = 71.96\%$, $t = 22.57$ sec. (e) R-SSC, $e = \mathbf{67.37}\%$, $t = \mathbf{1.83}$ sec.



Transformed Subspaces

e=2.92% !

Motion Segmentation



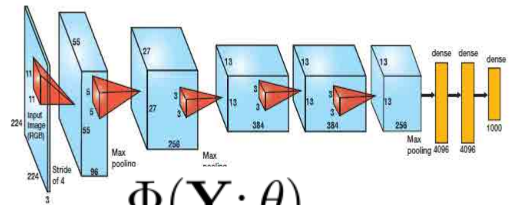
Method	Misclassification (%)
SSC [6]	21.8693
LSA [27]	17.8766
LBF [28]	33.8475
R-SSC	19.0653
R-SSC+RSC	**3.902**

Qiang Qiu, Guillermo Sapiro, "Learning Transformations for Clustering and Classification", Journal of Machine Learning Research (JMLR), 16(Feb):187−225, 2015

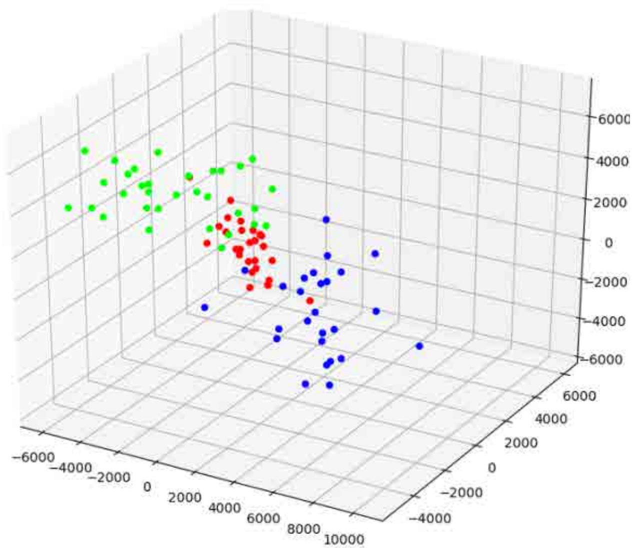
Orthogonal Low-rank Loss

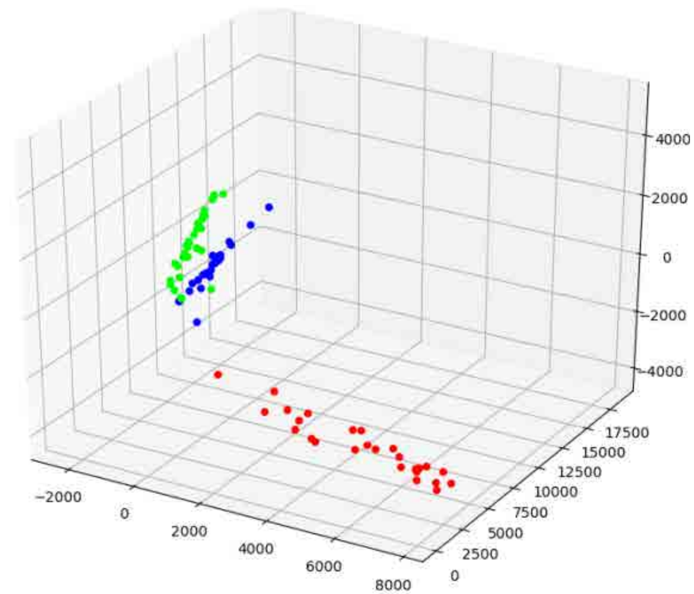
$$\sum_{c=1}^{C} ||\mathbf{TY}_c||_* - ||\mathbf{TY}||_* \quad \Rightarrow \quad \sum_{c=1}^{C} ||\Phi(\mathbf{Y}_c;\theta)||_* - ||\Phi(\mathbf{Y};\theta)||_*$$

T =>



$\Phi(\mathbf{Y};\theta)$

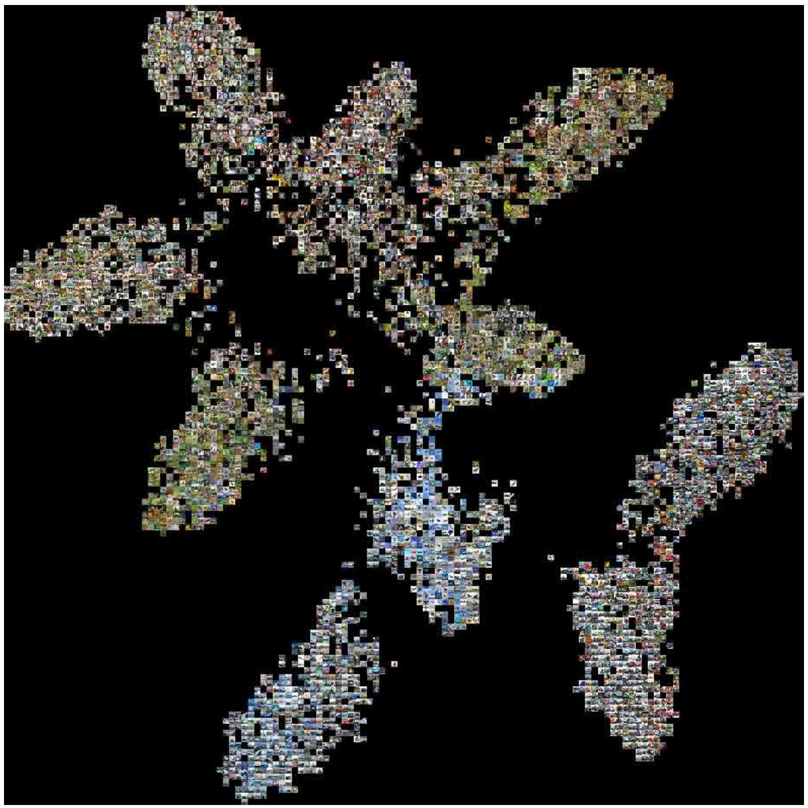




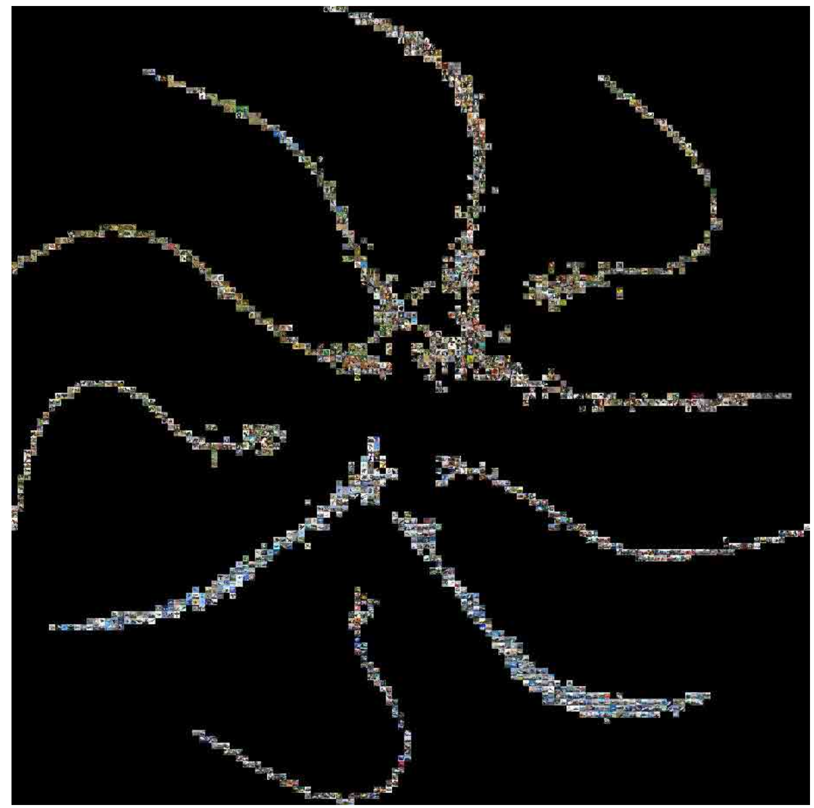
José Lezama, Qiang Qiu, Pablo Musé, Guillermo Sapiro, "OLÉ: Orthogonal Low-rank Embedding, A Plug and Play Geometric Loss for Deep Learning", Computer Vision and Patt. Recn. (CVPR), 2018

https://github.com/jlezama/OrthogonalLowrankEmbedding

Orthogonal Low-rank Loss



Softmax

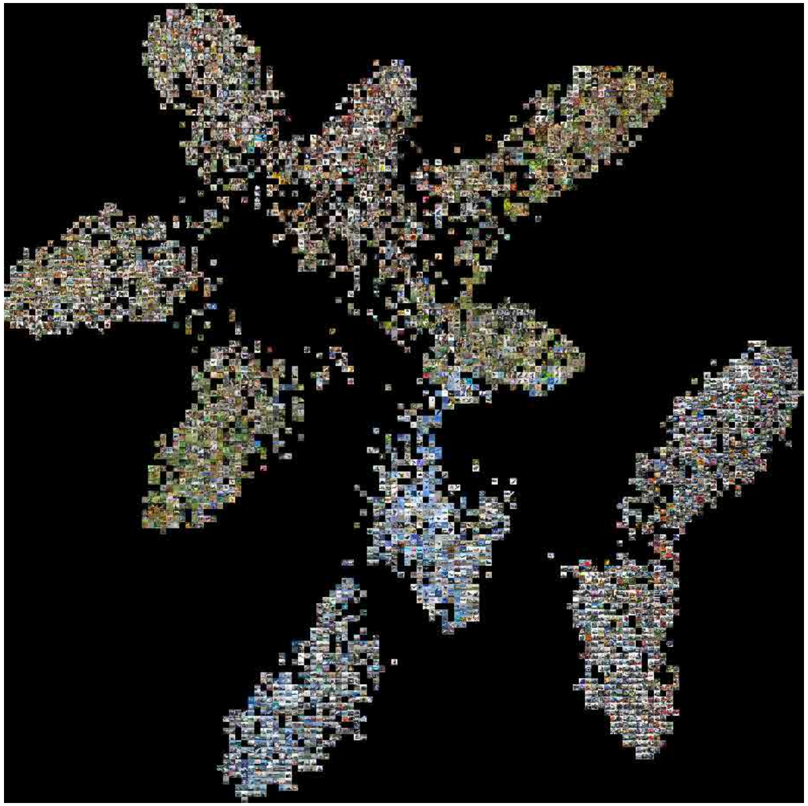


Low-rank

José Lezama, Qiang Qiu, Pablo Musé, Guillermo Sapiro, "OLÉ: Orthogonal Low-rank Embedding, A Plug and Play Geometric Loss for Deep Learning", Computer Vision and Patt. Recn. (CVPR), 2018

https://github.com/jlezama/OrthogonalLowrankEmbedding

Orthogonal Low-rank Loss



Softmax

Low-rank

José Lezama, Qiang Qiu, Pablo Musé, Guillermo Sapiro, "OLÉ: Orthogonal Low-rank Embedding, A Plug and Play Geometric Loss for Deep Learning", Computer Vision and Patt. Recn. (CVPR), 2018

https://github.com/jlezama/OrthogonalLowrankEmbedding

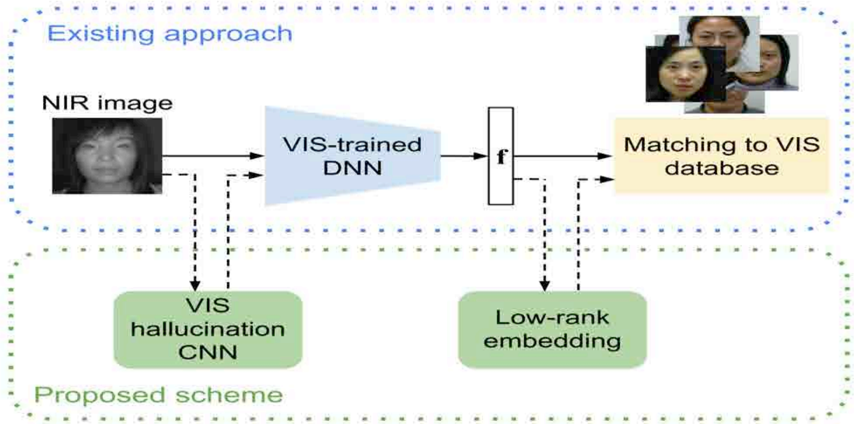
Orthogonal Low-rank Loss

Dataset	Architecture	λ	% Error ($L_o + \lambda \cdot L_s$)	% Error (L_s only)
SVHN	DenseNet-40-12 [11]	1/2	$\mathbf{3.62} \pm 0.04$	3.93 ± 0.08
MNIST	DenseNet-40-12	1/2	$\mathbf{0.78} \pm 0.04$	0.88 ± 0.03
CIFAR10+	DenseNet-40-12	1/8	$\mathbf{5.30} \pm 0.26$	5.54 ± 0.13
CIFAR10+	ResNet-110 [8]	1/4	$\mathbf{5.39} \pm 0.25$	6.05 ± 0.8
CIFAR10+	VGG-19 [29]	1/4	$\mathbf{7.13} \pm 0.2$	7.37 ± 0.11
CIFAR10+	VGG-11	1/2	$\mathbf{7.73} \pm 0.14$	8.06 ± 0.22
CIFAR10	VGG-16 [18]	1/2	$\mathbf{7.22} \pm 0.14$	8.23 ± 0.13
CIFAR100+	PreResNet-110 [9]	1/20	$\mathbf{22.8} \pm 0.34$	23.01 ± 0.19
CIFAR100+	VGG-19	1/10	$\mathbf{27.54} \pm 0.11$	28.04 ± 0.42
CIFAR100	VGG-19	1/10	$\mathbf{37.25} \pm 0.33$	38.15 ± 0.28
FaceScrub-500	VGG-FACE [22]	1/10	$\mathbf{1.55} \pm 0.02$	2.49 ± 0.01
STL-10	CNN-5	1/16	$\mathbf{25.42} \pm 0.20$	28.68 ± 0.67
STL-10+	CNN-5	1/4	$\mathbf{16.68} \pm 0.24$	18.22 ± 0.27

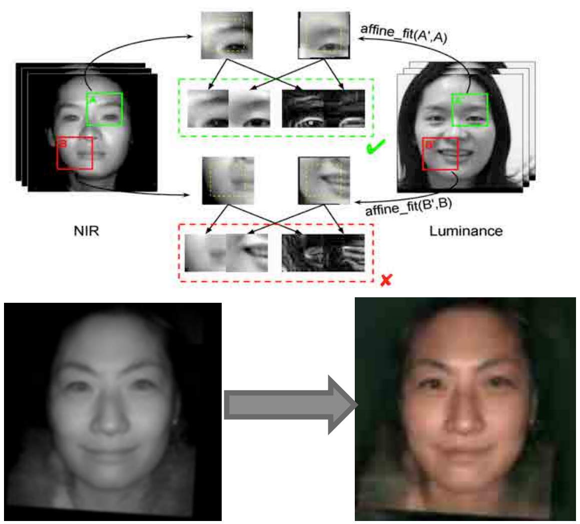
José Lezama, Qiang Qiu, Pablo Musé, Guillermo Sapiro, "OLÉ: Orthogonal Low-rank Embedding, A Plug and Play Geometric Loss for Deep Learning", Computer Vision and Patt. Recn. (CVPR), 2018

https://github.com/jlezama/OrthogonalLowrankEmbedding

Cross-spectral Face Recognition

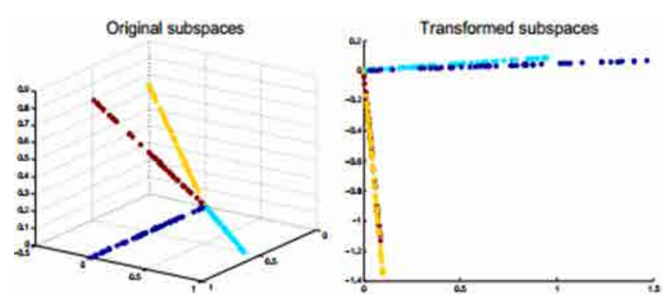


NIR-VIS Hallucination



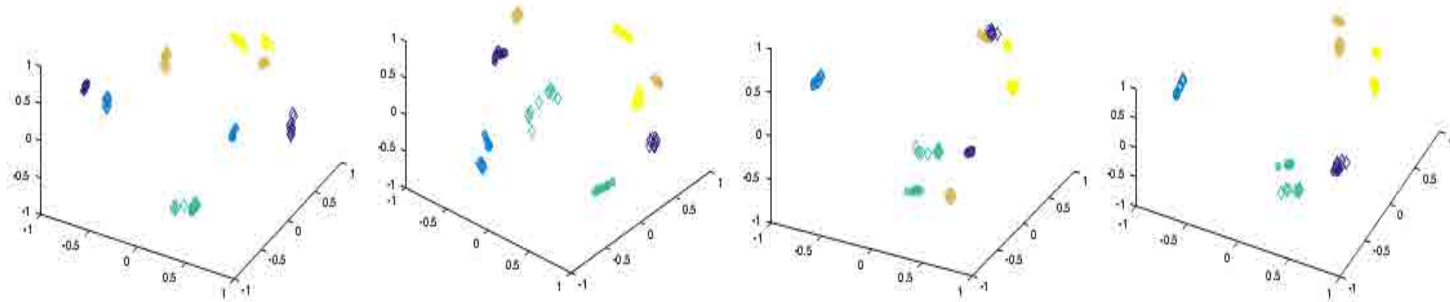
Low-rank Embedding

$$\sum_{c=1}^{C} ||\mathbf{TY}_c||_* - ||\mathbf{TY}||_*$$

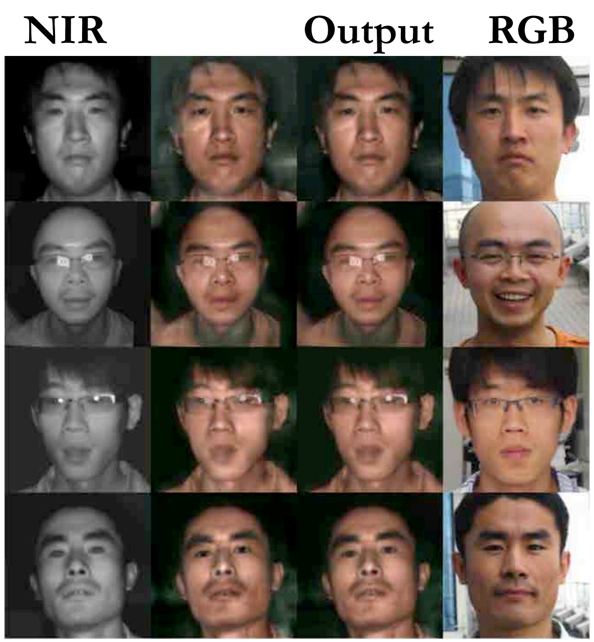


Jose Lezama, Qiang Qiu, Guillermo Sapiro, "Not Afraid of the Dark: NIR-VIS Face Recognition via Cross-spectral Hallucination and Low-rank Embedding", Computer Vision and Patt. Recn. (CVPR), 2017

Cross-spectral Face Recognition



(a) No embedding (b) Pairwise embedding (c) Triplet embedding (d) Low-rank embedding

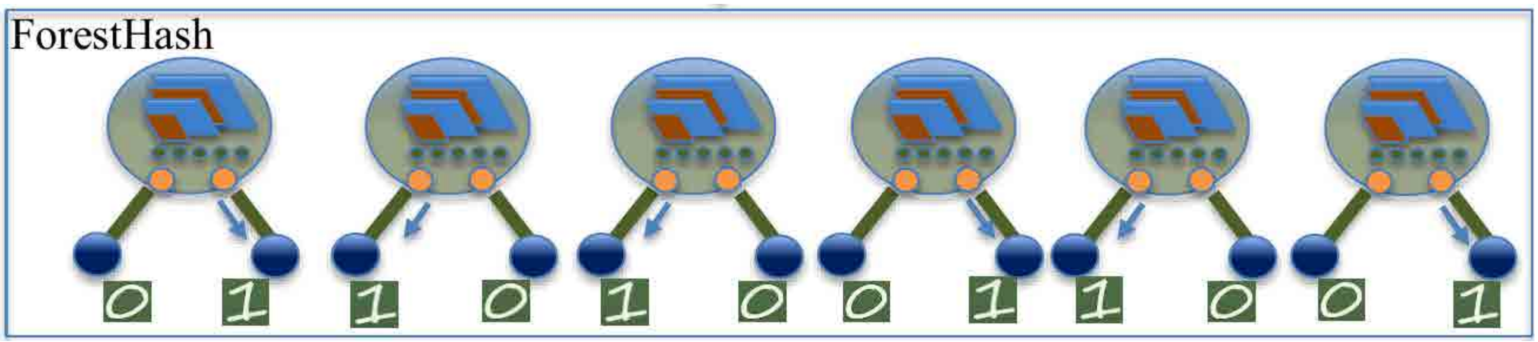


	Accuracy (%)
VGG-S	75.04
VGG-S + Hallucination	80.65
VGG-S + Low-rank	89.88
VGG-S + Hallucination + Low-rank	95.72
VGG-face	72.54
VGG-face + Hallucination	83.10
VGG-face + Low-rank	82.26
VGG-face + Hallucination + Low-rank	91.01
COTS	83.84
COTS + Hallucination	93.02
COTS + Low-rank	91.83
COTS + Hallucination + Low-rank	**96.41**

Jose Lezama, Qiang Qiu, Guillermo Sapiro, "Not Afraid of the Dark: NIR-VIS Face Recognition via Cross-spectral Hallucination and Low-rank Embedding", Computer Vision and Patt. Recn. (CVPR), 2017

Image Hashing

Each face is represented by a 48-bit hash code.



We set ‘1’ for the visited nodes, and ‘0’ for the rest, obtaining a (2^d-2)-bit hash code.

- Random class grouping (uniquness)
- Orthogonal Low-rank loss (consistency)
- Near-optimal Code aggregation

CNN2

1	Conv+ReLU+MaxPool	$5 \times 5 \times 3 \times 64$
2	Conv+ReLU+MaxPool	$5 \times 5 \times 64 \times 32$
3	FC	output: 256

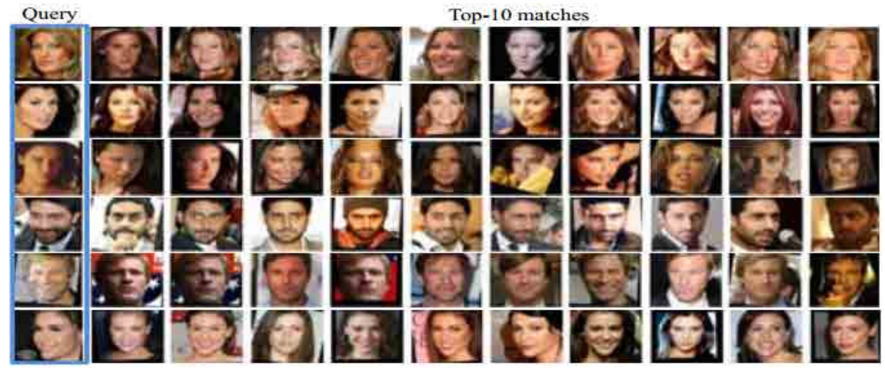
Train and deploy in parallel!!

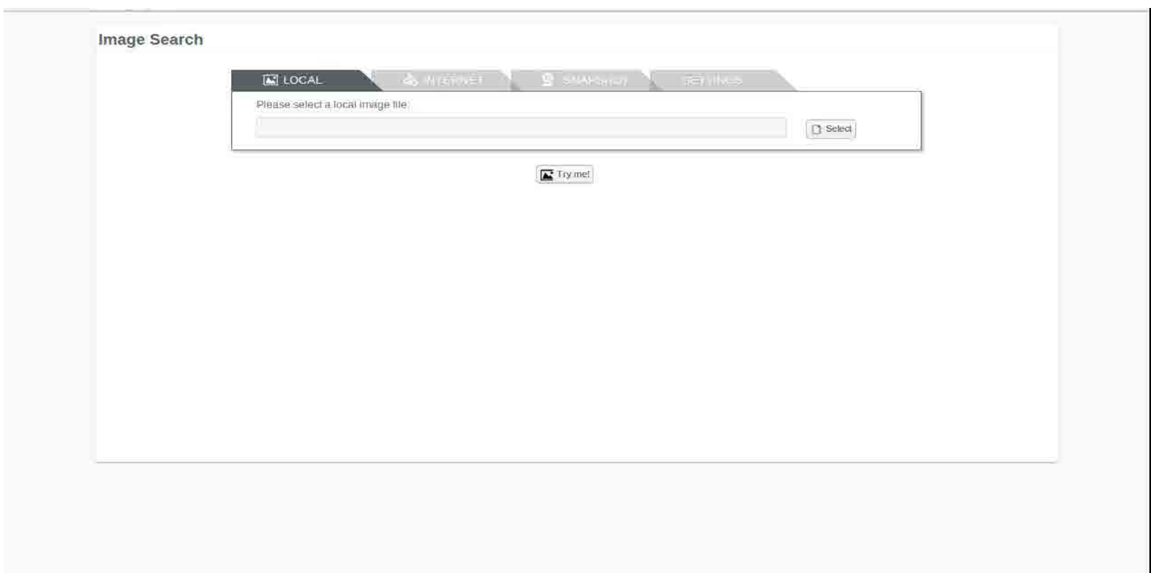
Qiang Qiu, Jose Lezama, Alex Bronstein, Guillermo Sapiro, "ForestHash: Semantic Hashing With Shallow Random Forests and Tiny Convolutional Networks", arXiv:1711.08364, 2017

Image Hashing

Method	radius = 0			radius ≤ 2		
	Precision	Recall	F1	Precision	Recall	F1
KLSH (36-bit) [15]	16.97	3.73	6.11	31.93	8.38	13.28
AGH1 (36-bit) [19]	18.38	56.12	27.69	7.75	82.30	14.16
AGH2 (36-bit) [19]	13.56	57.48	21.94	5.53	89.52	10.41
LDAHash (36-bit) [64]	23.42	0.65	1.26	45.11	10.25	16.71
Proposed (36-bit)	82.17	**82.29**	82.23	47.58	**89.38**	62.10
Proposed (48-bit)	**90.74**	80.42	**85.27**	**81.74**	87.41	**84.48**

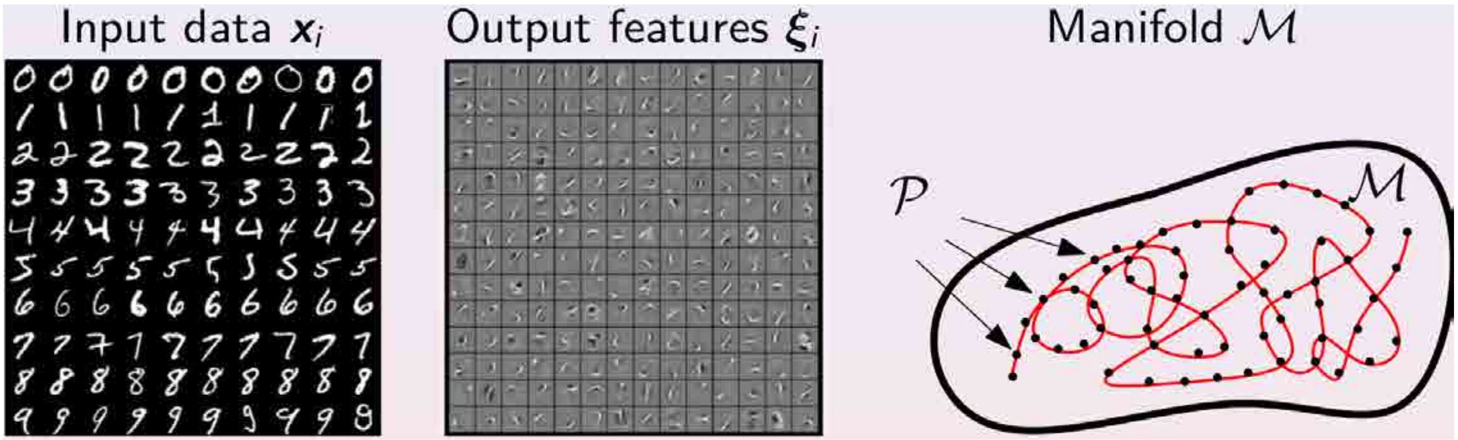
~30 microseconds to index a face.
~20 milliseconds to scan one million faces.





Live Demo over 5M faces.

Low-dimensional Manifold



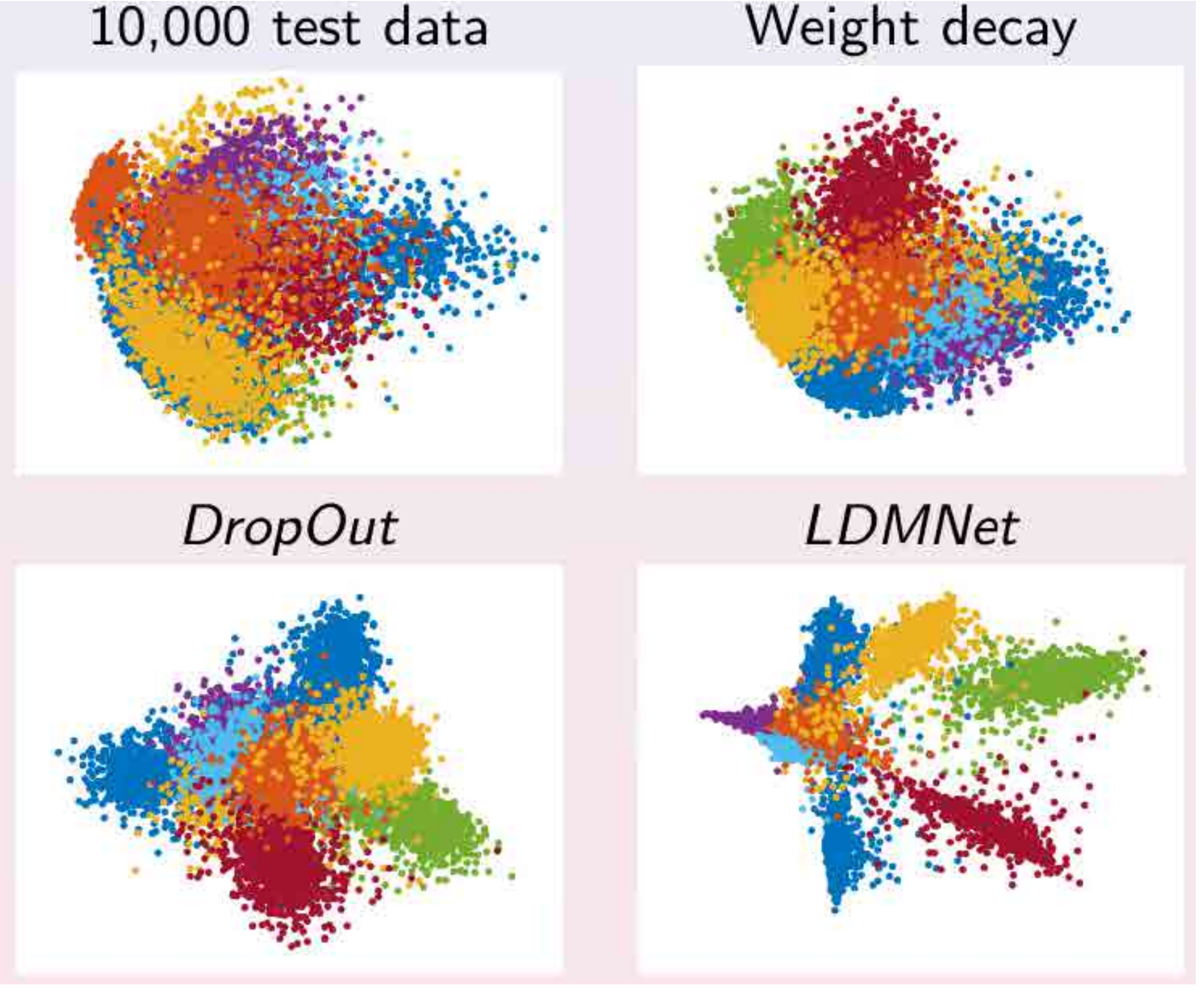
< 56 dimensions

$$\min_{\theta,\mathcal{M}} \quad J(\theta) + \frac{\lambda}{|\mathcal{M}|}\int_{\mathcal{M}} \dim(\mathcal{M}(\boldsymbol{p}))d\boldsymbol{p}$$

$$\text{s.t.} \quad \{(\boldsymbol{x}_i, f_\theta(\boldsymbol{x}_i))\}_{i=1}^{N} \subset \mathcal{M},$$

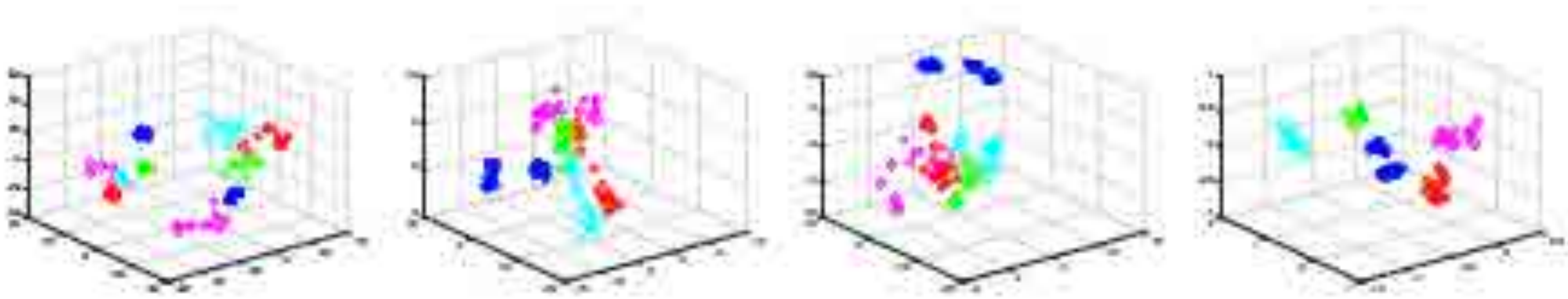
Wei Zhu, Qiang Qiu, Jiaji Huang, Robert Calderbank, Guillermo Sapiro, Ingrid Daubechies, "LDMNet: Low Dimensional Manifold Regularized Neural Networks", Computer Vision and Patt. Recn. (CVPR), 2018

Low-dimensional Manifold



Wei Zhu, Qiang Qiu, Jiaji Huang, Robert Calderbank, Guillermo Sapiro, Ingrid Daubechies, "LDMNet: Low Dimensional Manifold Regularized Neural Networks", Computer Vision and Patt. Recn. (CVPR), 2018

Low-dimensional Manifold

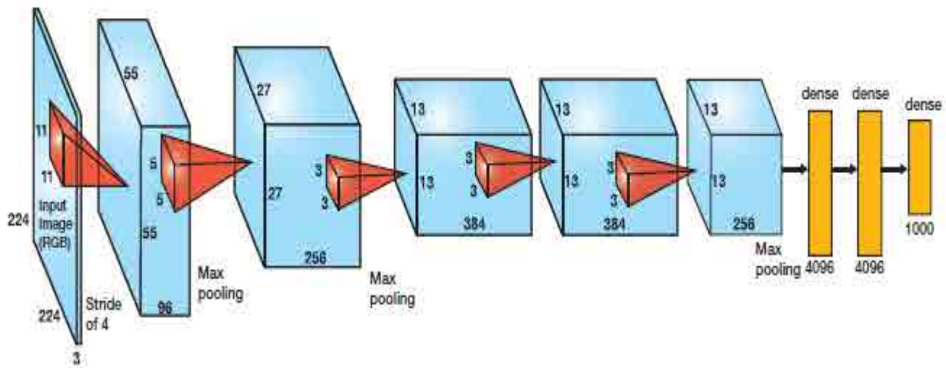


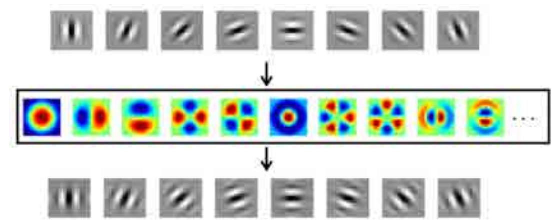
(a) VGG-face (b) Weight decay (c) *DropOut* (d) *LDMNet*

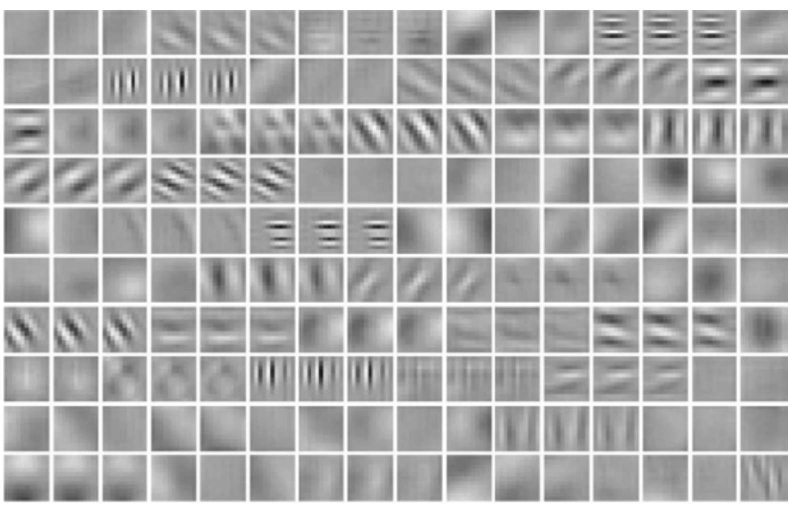
Wei Zhu, Qiang Qiu, Jiaji Huang, Robert Calderbank, Guillermo Sapiro, Ingrid Daubechies, "LDMNet: Low Dimensional Manifold Regularized Neural Networks", Computer Vision and Patt. Recn. (CVPR), 2018

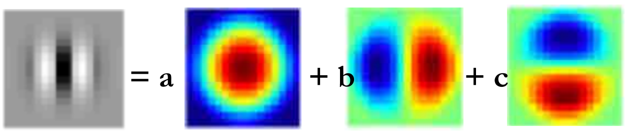
Regularizing with Filter Structures

Decompose Filters over Bases





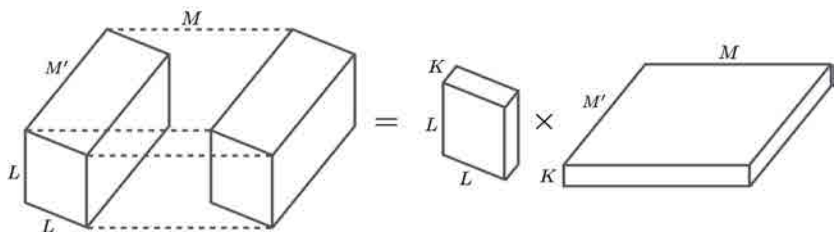




- Reduce parameters and computations (K/L^2).
- Impose filter regularity by bases truncation.

Qiang Qiu, Xiuyuan Cheng, Robert Calderbank, Guillermo Sapiro, "DCFNet: Deep Neural Network with Decomposed Convolutional Filters", International Conf. on Machine Learning, ICML, 2018

Decomposed Convolutional Filters



Layer	CNN	DCFNet
1	conv $3 \times 3 \times 3 \times 64$	3 3×3 basis conv $1 \times 1 \times 9 \times 64$
2	ReLu	
3	conv $3 \times 3 \times 64 \times 64$	3 3×3 basis conv $1 \times 1 \times 192 \times 64$
4-5	ReLu, maxPool 2×2	
6	conv $3 \times 3 \times 64 \times 128$	3 3×3 basis conv $1 \times 1 \times 192 \times 128$
7	ReLu	
8	conv $3 \times 3 \times 128 \times 128$	3 3×3 basis conv $1 \times 1 \times 384 \times 128$
9-10	ReLu, maxPool 2×2	
(1-31 CNN layers are identical to *vgg-face* model in [25].)		
32	conv $5 \times 5 \times 512 \times 512$	8 5×5 basis conv $1 \times 1 \times 4096 \times 512$
33-34	ReLu, dropout	
35	conv $3 \times 3 \times 512 \times 512$	3 3×3 basis conv $1 \times 1 \times 1536 \times 512$
36-39	ReLu, dropout, FC, softmax	

Qiang Qiu, Xiuyuan Cheng, Robert Calderbank, Guillermo Sapiro, "DCFNet: Deep Neural Network with Decomposed Convolutional Filters", International Conf. on Machine Learning, ICML, 2018

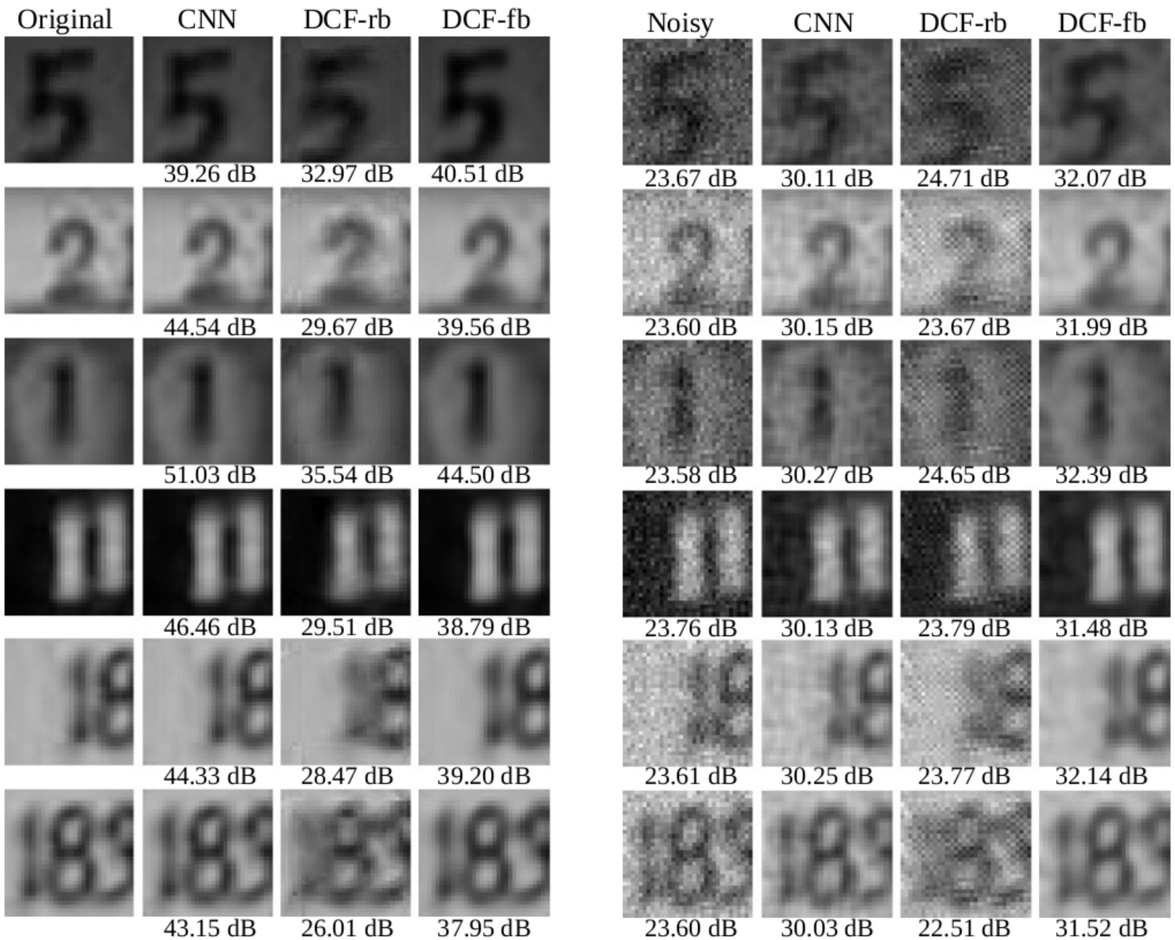
Decomposed Convolutional Filters

MNIST conv-2, 5x5						
	fb	rb	pca-s	pca-f	# param.	# MFlops
CNN	99.40				2.61×10^4	3.37
K=14	99.47	99.35	99.38	99.41	1.46×10^4	2.40
K=8	99.48	99.26	99.28	99.45	8.40×10^3	1.37
K=5	99.39	99.28	99.28	99.43	5.28×10^3	0.86
K=3	99.40	98.69	99.19	99.35	3.20×10^3	0.51
SVHN conv-3, 5x5						
	fb	rb	pca-s	pca-f	# param.	# MFlops
CNN	94.22				1.03×10^6	201.64
K=14	94.63	93.75	94.52	94.42	5.74×10^5	121.91
K=8	94.39	92.05	93.85	94.30	3.30×10^5	69.67
K=5	93.93	91.28	92.34	94.03	2.06×10^5	43.55
K=3	92.84	88.47	91.88	93.10	1.24×10^5	26.13
Cifar10 conv-3, 5x5						
	fb	rb	pca-s	pca-f	# param.	# MFlops
CNN	85.66					
K=14	85.88	84.76	85.27	85.34		
K=8	85.30	81.27	84.70	85.09	(same as above)	
K=5	84.35	77.96	83.12	83.94		
K=3	83.12	74.05	80.94	82.91		
Cifar10 vgg-16, 3x3						
	fb	rb	pca-s	pca-f	# param.	# MFlops
CNN	87.02				1.47×10^7	547.20
K=5	87.79	84.16	87.98	87.60	8.18×10^6	311.68
K=3	88.21	78.46	87.45	87.54	4.91×10^6	187.02

	Accuracy	# param.	# GFlops
VGG-face	97.27 %	-	-
CNN	97.65 %	21.26×10^6	30.05
DCFNet	97.32 %	7.01×10^6	10.09

Qiang Qiu, Xiuyuan Cheng, Robert Calderbank, Guillermo Sapiro, "DCFNet: Deep Neural Network with Decomposed Convolutional Filters", International Conf. on Machine Learning, ICML, 2018

Decomposed Convolutional Filters

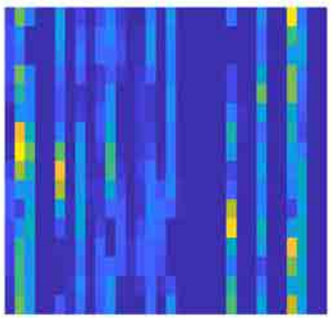


(a) Original (b) Gaussian noise

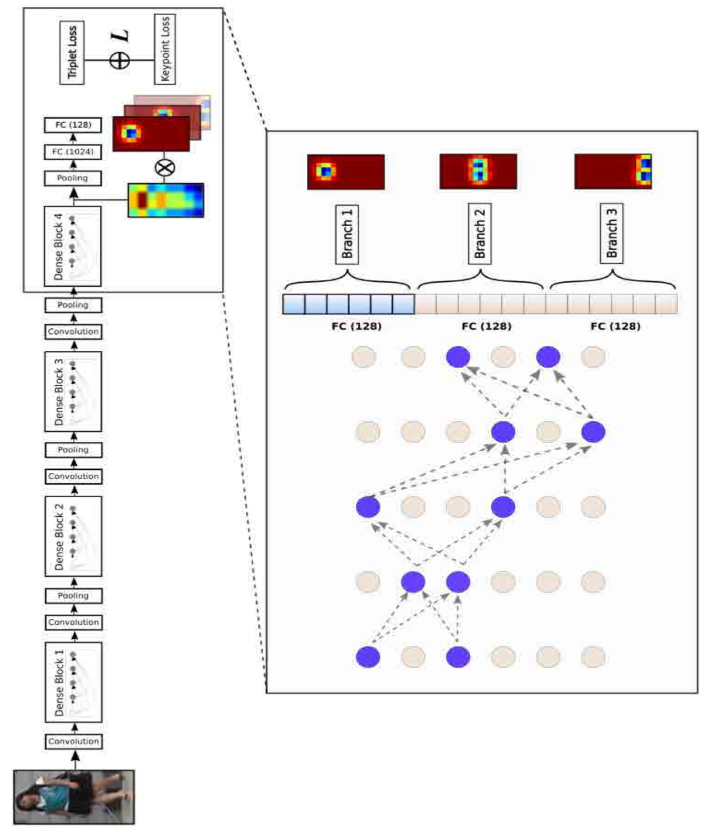
Qiang Qiu, Xiuyuan Cheng, Robert Calderbank, Guillermo Sapiro, "DCFNet: Deep Neural Network with Decomposed Convolutional Filters", International Conf. on Machine Learning, ICML, 2018

Structures across Filters

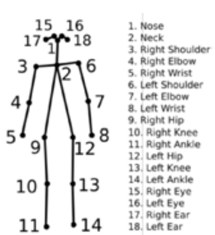
Orientation equivariant learning

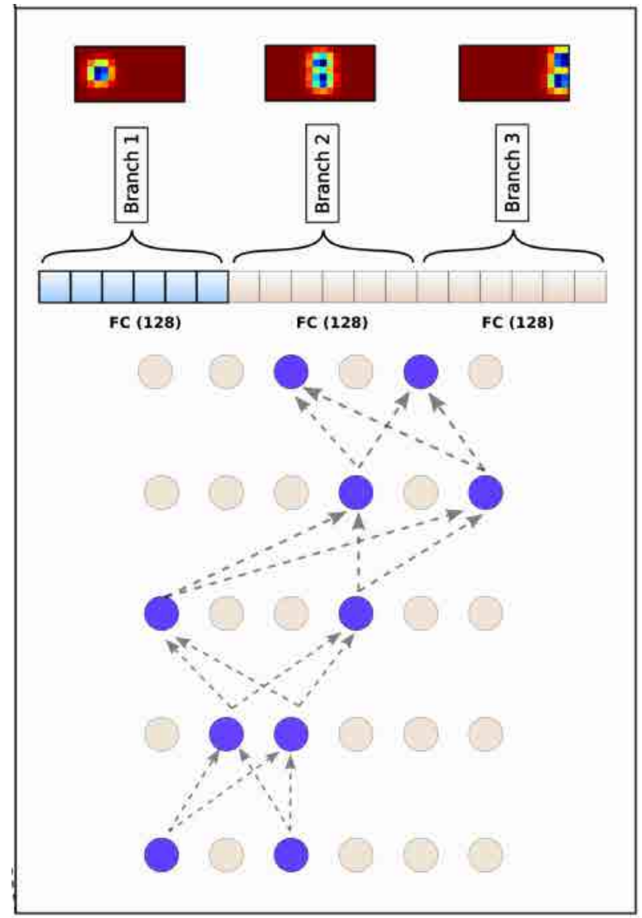


Virtual branching



Person Re-Identification





Albert Gong, Qiang Qiu, Guillermo Sapiro, "Virtual CNN Branching: Efficient Feature Ensemble for Person Re-Identification", arXiv:1803.05872, 2018 (High school research)

Person Re-Identification

Method	Single Query mAP	Single Query Rank-1	Single Query Rank-5	Multi-query mAP	Multi-query Rank-1	Multi-query Rank-5
Gated Siamese CNN [34]	39.95	65.88	-	48.45	76.04	-
PIE [3]	53.87	78.65	90.26	-	-	-
JLML [22]	65.5	85.1	-	74.5	89.7	-
PAN [3]	63.35	82.81	-	71.72	88.18	-
Res50 + Attribute [21]	64.67	84.29	93.20	-	-	-
GoogLeNet + DTL [20]	65.5	83.7	-	73.08	89.6	-
PDC [4]	63.41	84.14	92.73	-	-	-
SpindleNet [5]	-	76.9	91.5	-	-	-
TriNet† [17]	69.14	84.92	94.21	76.42	90.53	96.29
MobileNet + DML† [35]	68.86	87.73	-	77.14	91.66	-
Baseline	68.3	83.3	93.1	75.8	90.1	95.9
Human Landmark	70.8	**87.9**	96.0	**79.4**	**93.5**	**98.9**
Pose Orientation	**71.1**	87.7	**96.5**	79.3	93.3	98.5

Market-1501

Method	Labeled mAP	Labeled Rank-1	Labeled Rank-5	Detected mAP	Detected Rank-1	Detected Rank-5
Gated Siamese CNN [34]	-	-	-	51.25	61.8	80.9
PIE [3]	-	-	-	67.21	61.50	89.30
JLML [22]	-	83.2	98.0	-	80.6	96.9
PAN [3]	35.03	36.86	56.86	34.00	36.29	55.50
GoogLeNet + DTL [20]	-	85.4	-	-	84.1	-
PDC [4]	-	88.70	-	-	78.29	-
SpindleNet [5]	-	-	-	-	88.5	97.8
TriNet† [17]	-	89.63	99.01	-	87.58	98.17
Baseline	93.6	88.2	99.4	92.0	86.4	98.1
Human Landmark	95.5	**91.0**	**99.8**	**94.7**	88.9	**99.4**
Pose Orientation	**95.8**	90.9	99.6	94.5	**89.3**	**99.4**

CUHK03

Albert Gong, Qiang Qiu, Guillermo Sapiro, "Virtual CNN Branching: Efficient Feature Ensemble for Person Re-Identification", arXiv:1803.05872, 2018 (High school research)

Thank you!