Sparse Modeling of Human Actions from Motion Imagery

ALEXEY CASTRODAD & GUILLERMO SAPIRO
PRESENTER: YUXIANG WANG

About the authors

- Guillermo Sapiro
  - U Minnesota -> Duke
  - Pioneer of using sparse representation in Computer Vision/Graphics
- Alexey Castrodad
  - PhD of Sapiro
  - Nothing much online…

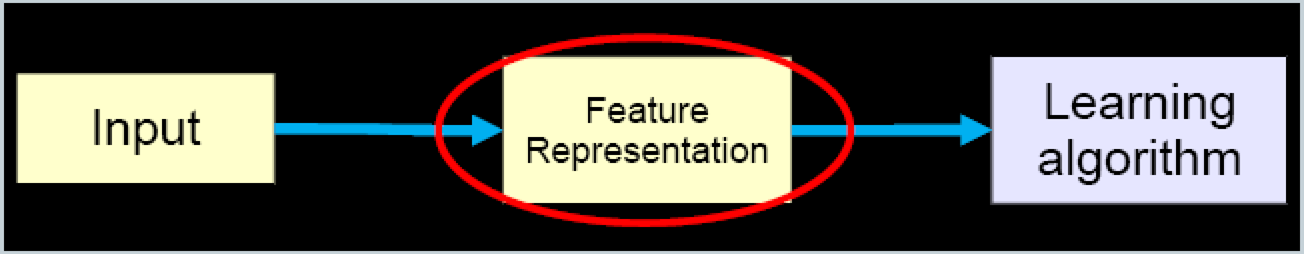
Structure of presentation

- On Deep Learning
- Technical details of this paper
  - Features.
  - Dictionary learning.
  - Classification
- Experiments
- Questions and discussions

A slide on deep learning

- People:
  - Andrew Ng @ Stanford
  - Yann LeCun @ NYU
  - Geoffery Hinton @ Toronto U
- Deep learning:
  - Multi-layer neural networks with sparse coding
  - “Deep” is only a marketing term, usually 2-3 layers
  - Very good in practice, but a bit nasty in theory

Unsupervised feature learning

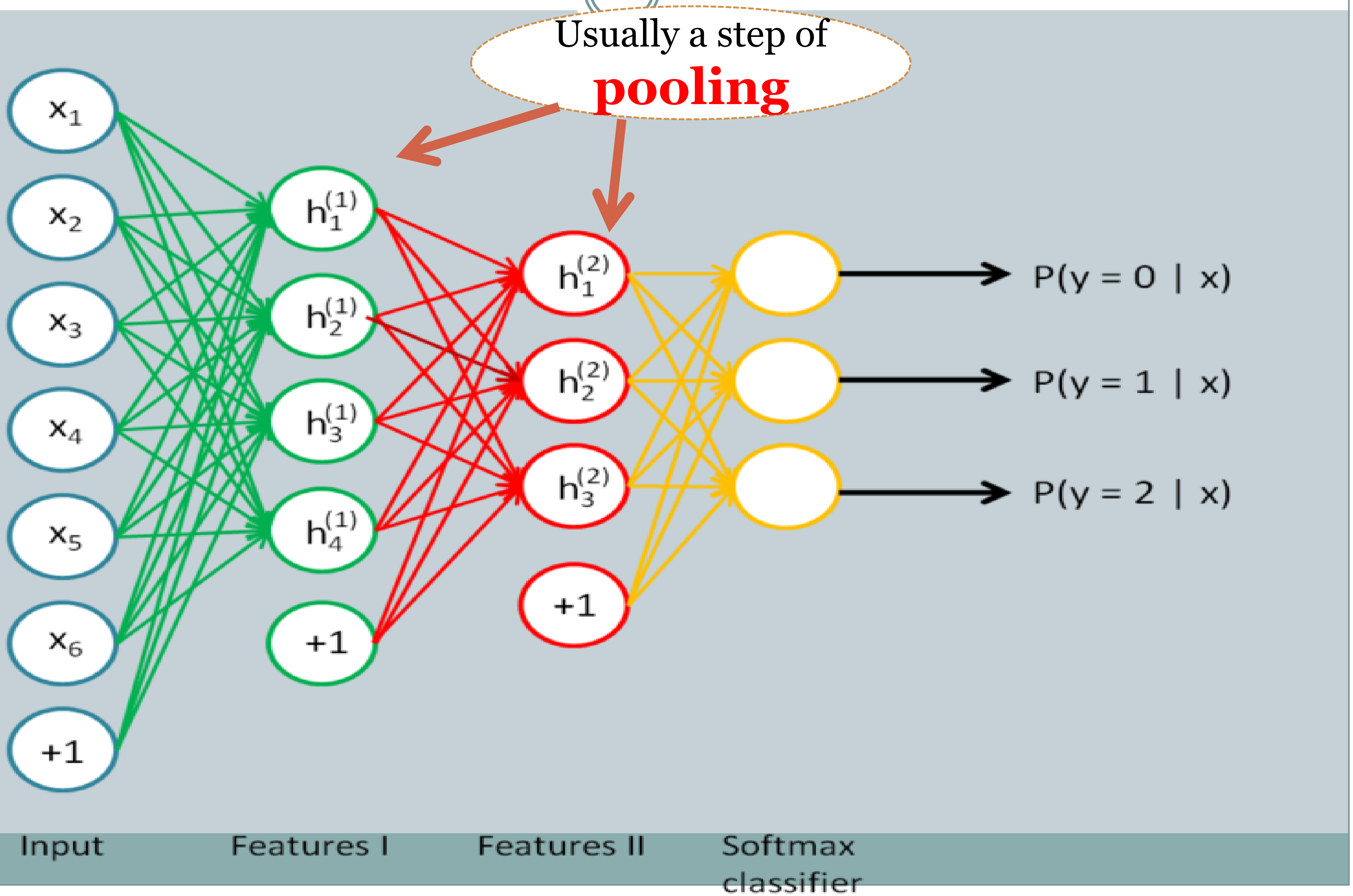


- Usually hand-crafted: SIFT, HOG, etc…
- Now learn from data directly and
  - No engineering/research effort
  - Equally good if not better

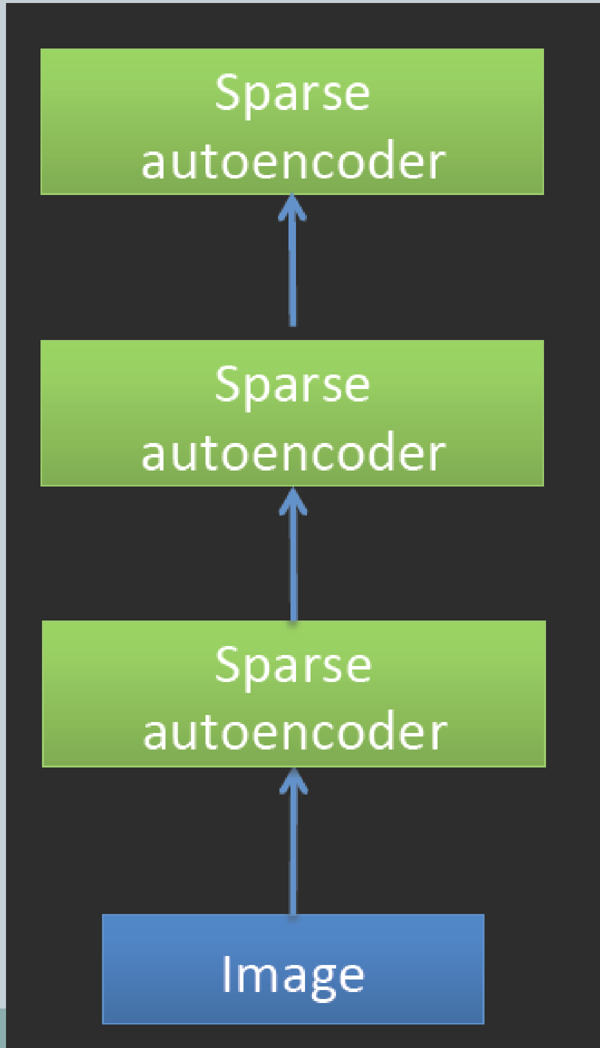
Unsupervised feature learning

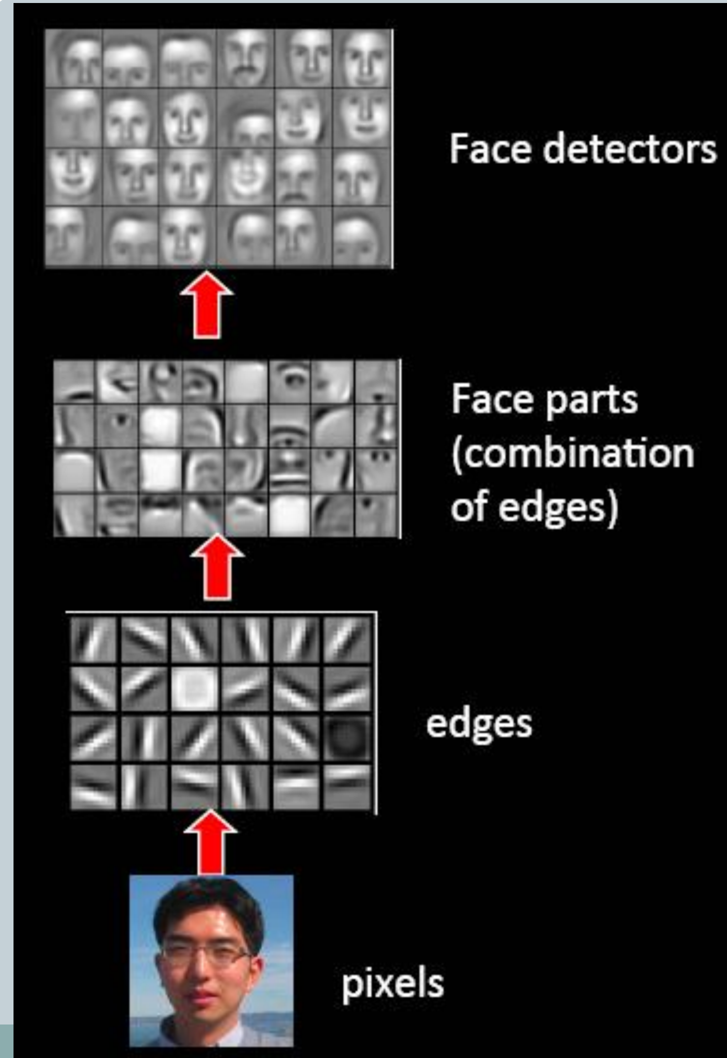
- Outperformed state-of-the-art in:
  - Activity recognition: Hollywood 2 Benchmark
  - Audio recognition/Phoneme classification
  - Parsing sentence
  - Multi-class segmentation: (topic discussed last week)
  - The list goes on...

Deep learning for classification



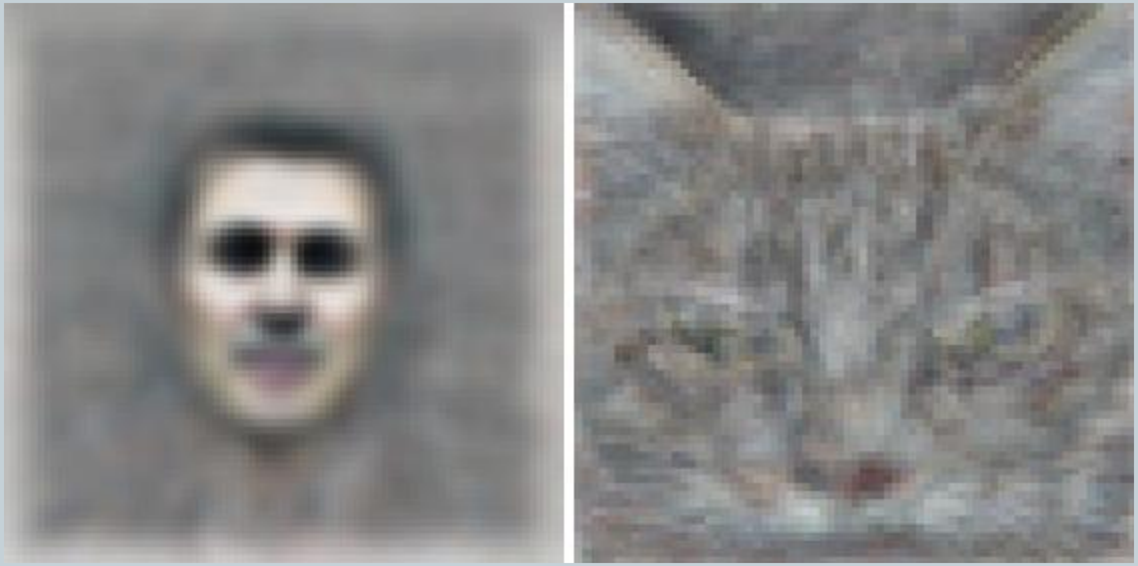
A brainless algorithm…





Unsupervised feature learning

- Large-scale unsupervised feature learning
  - Human learns features (sometimes very high level features: grandmother cell)
  - 16000 CPUs of Google run weeks to simulate human brain and watch YouTube. It gives:



Criticism on deep learning

- Advocates say deep learning is SVM in the 80s.
- Critics say it's yet another a flashback/relapse of the neural network rush.
  - Little insights into how/why it works.
  - Computational intensive
  - A lot of parameters to tune

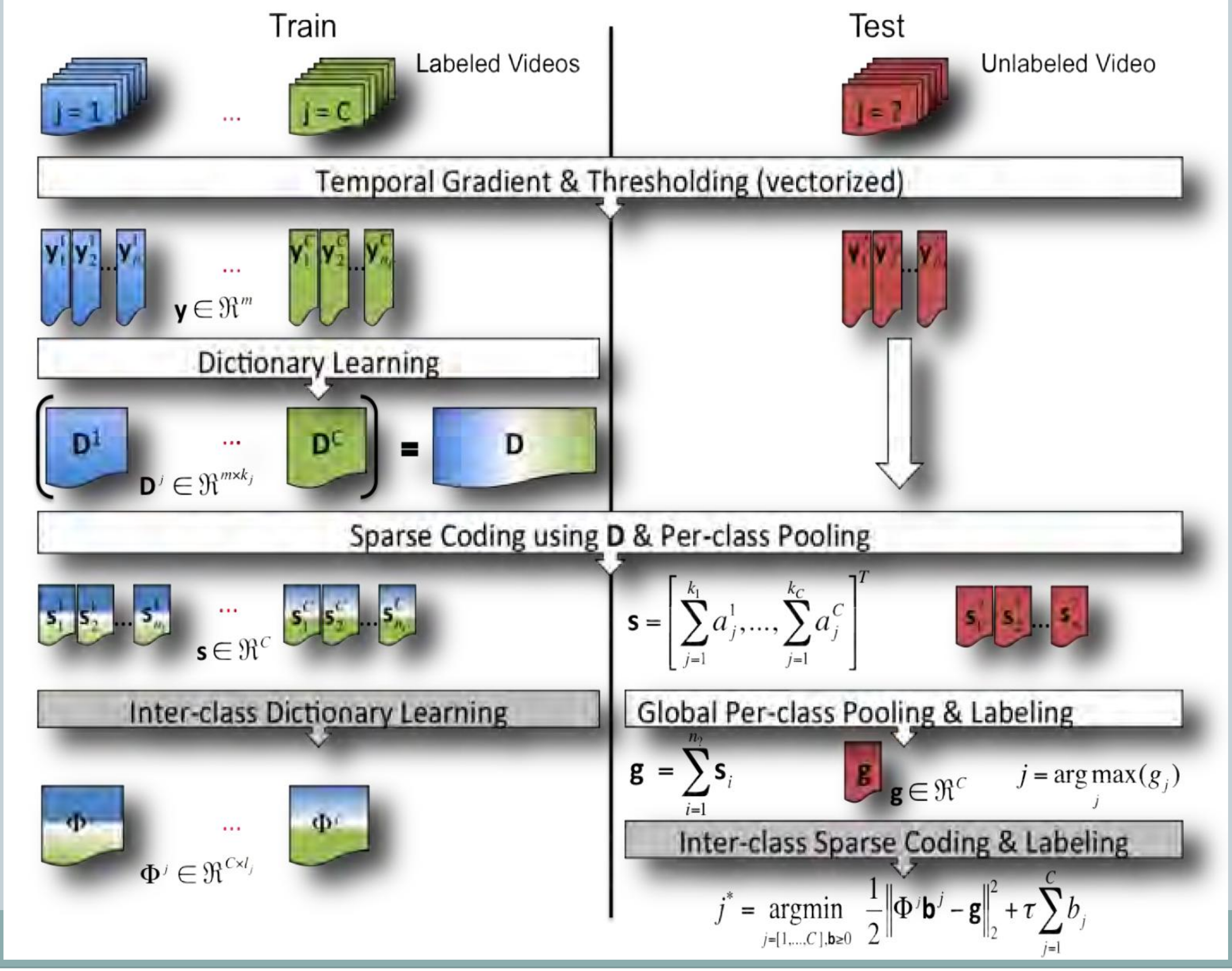
Wanna know more?

- Watch YouTube video:
  - **Bay Area Vision Meeting: Unsupervised Feature Learning and Deep Learning**
- A great step-by-step tutorial:
  - http://deeplearning.stanford.edu/wiki

Back to this paper

- Use deep learning framework for action recognition (with some variations).
- Not the first, but the most successful.
- Supply physical meaning to the second layer.
- Benefits from Blessing of dimensionality?

Flow chart of the algorithm



Minimal feature used?

- Data vector y:
  - 15*15*7 volume patch in temporal gradient
- Thresholding:
  - Only those patch with large variations used
- Simple but captures the essence.
  - Invariant to location
- More sophisticated feature descriptors are automatically learned!

Dictionary learning/Feature learning

- First layer

$$\mathbf{D}^{j*} = \arg \min_{(\mathbf{D}^j, \mathbf{A}^j) \succeq 0} \frac{1}{2} \|\mathbf{D}^j \mathbf{A}^j - \mathbf{Y}^j\|_F^2 + \lambda \sum_{i=1}^{n_j} \mathcal{S}(\mathbf{a}^j),$$

- Per Class Sum-Pooling

$$\mathbf{s} = [\mathcal{S}(\mathbf{a}^1), ..., \mathcal{S}(\mathbf{a}^C)]^T \in \Re_+^C$$

- Second layer

$$\mathbf{\Phi}^{j*} = \arg \min_{(\mathbf{\Phi}^j, \mathbf{B}^j) \succeq 0} \frac{1}{2} \|\mathbf{\Phi}^j \mathbf{B}^j - \mathbf{S}^j\|_F^2 + \tau \sum_{i=1}^{n_j} \mathcal{S}(\mathbf{b}^j),$$

Procedure for classification

- Video i has n_i patches: $Y=[y_1, y_2,\ldots, y_{ni}]$
- Layer 1 sparse coding to get A

$$\mathbf{A}^* = \underset{\mathbf{A}\succeq 0}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{D}\mathbf{A} - \mathbf{Y}\|_F^2 + \lambda \sum_{i=1}^{n} \mathcal{S}(\mathbf{a}_i),$$

- Class-Sum Pooling from A to $S = [s_1,\ldots s_{ni}]$
- Patch-Sum Pooling from S to $g = s_1 +\ldots+ s_{ni}$
- Class-wise layer 2 sparse coding

$$\mathcal{R}(\mathbf{\Phi}^j, \mathbf{g}) = \min_{\mathbf{b}^j\succeq 0} \frac{1}{2}\|\mathbf{\Phi}^j\mathbf{b}^j - \mathbf{g}\|_2^2 + \tau\mathcal{S}(\mathbf{b}^j)$$

Procedure for classification

- Either by (pooled) sparse code of first layer

$$f_1(\mathbf{g}) = \{j | g_j > g_i, j \neq i, (i, j) \in [1, ..., C]\}.$$

- Or use residual of second layer

$$f_2(\mathbf{g}) = \{j | \mathcal{R}(\mathbf{\Phi}^j, \mathbf{g}) < \mathcal{R}(\mathbf{\Phi}^i, \mathbf{g}), j \neq i, (i, j) \in [1, ..., C]\}.$$

Analogy to Bag-of-Words model

- D contains:
  - Latent local 'words'
  - learned from training image patches
- For a new video:
  - Each local patch is represented by 'words'
  - Then sum pooled over each class, and over all patches, obtaining 'g'
  - If reverse the order, then exactly Bag-of-Words.

Looking back to their two approach

- Given Bag-of-Words representation: v = R^k.
  - Classification Method A: is in fact a simple voting scheme.
  - Classification Method B is to manipulate the voting results by representing them with a set of pre-defined rules (each class has a set of rules), then check how fitting each set of rules is.

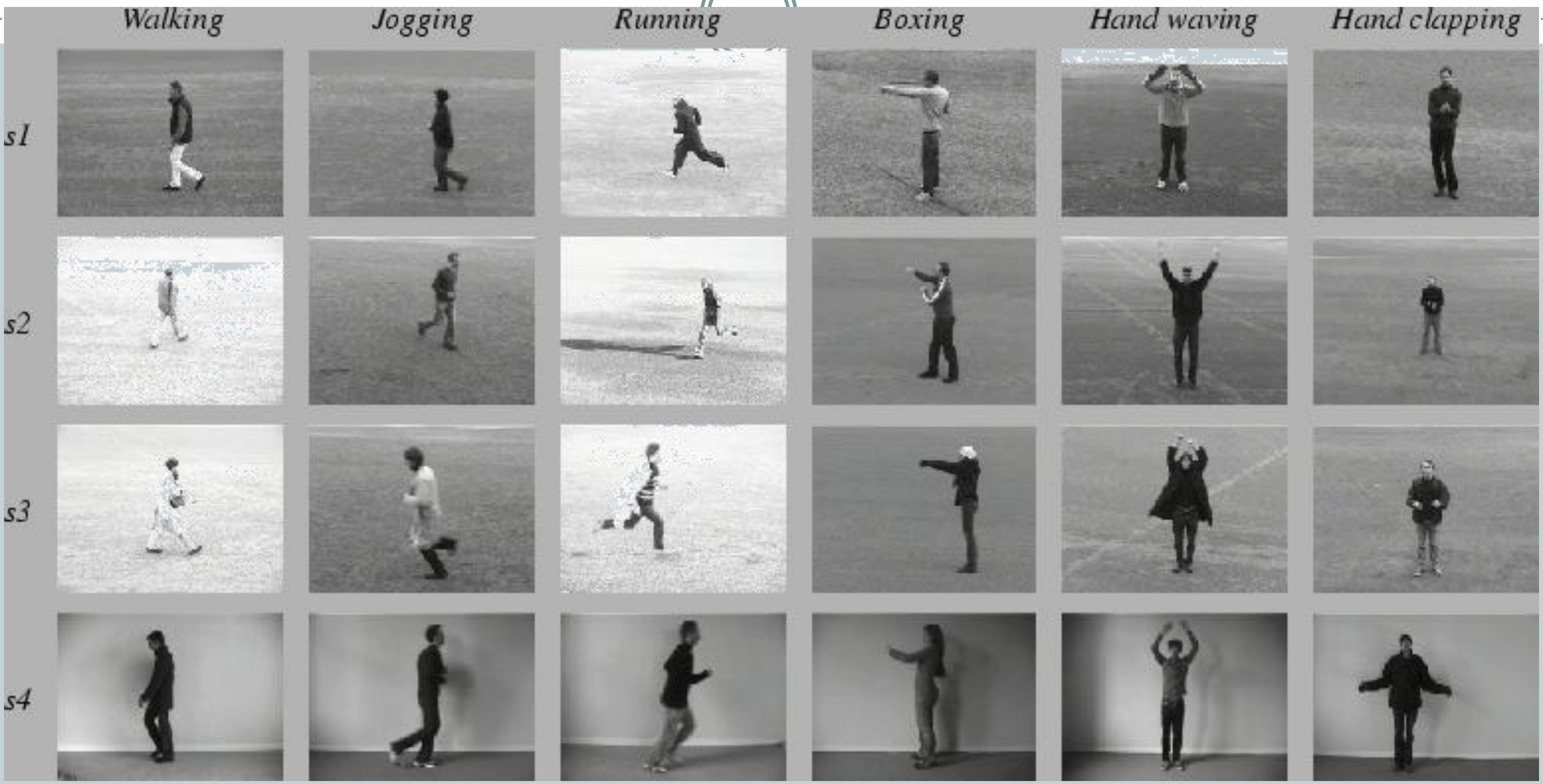
Blessing of dimensionality

- Since the advent of compressive sensing
  - Donoho, Candes, Ma Yi and etc…
- Basically:
  - Redundancy in data
  - Random data are almost orthogonal (incoherent)
  - Sparse/low-rank representation of data
  - Great properties for denoising, handling corrupted/missing data.
- This paper uses sparse coding but never explicitly handle data noise/corruption.
- Only implicitly benefits from such blessing.

Experiments

- Top 3 previous results vs.
  1. SM-1: Classification by pooled first layer output
  2. SM-2: Classification by second layer output
  3. SM-SVM: One-against-others SVM classification using per-class class-sum-pooled vectors S.

KTH dataset

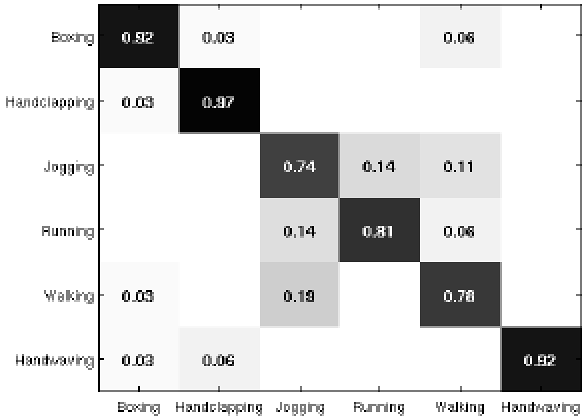


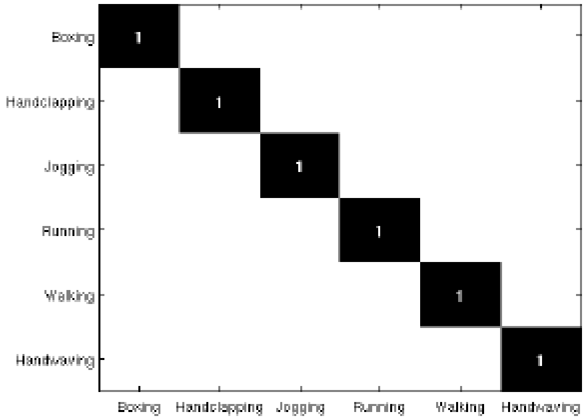
- Indoor, outdoor
- change of clothing, change of viewpoint

KTH Dataset

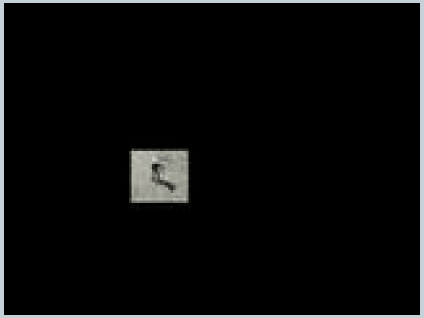
Table 2: Results for the KTH dataset.

Method	Overall Accuracy (%)
Wang *et al.* [34]	94.2
Kovashka *et al.* [17]	94.5
Guo *et al.* [13]	97.4
SM-SVM	87.6
SM-1	88.8
SM-2	100





UT-Tower Dataset



- Low resolution (20 pixels) (a blessing or a curse?)
- Bounding box is given
- Relatively easy among all UT action dataset.

UT-Tower dataset

Table 3: Results for the UT-Tower dataset.

Method	Overall Accuracy (%)
Guo *et al.* [13, 26]	97.2
Vezzani *et al.* [33]	93.9
Gall *et al.* [12]	93.9
SM-SVM	93.3
SM-1	97.2
SM-2	**100**

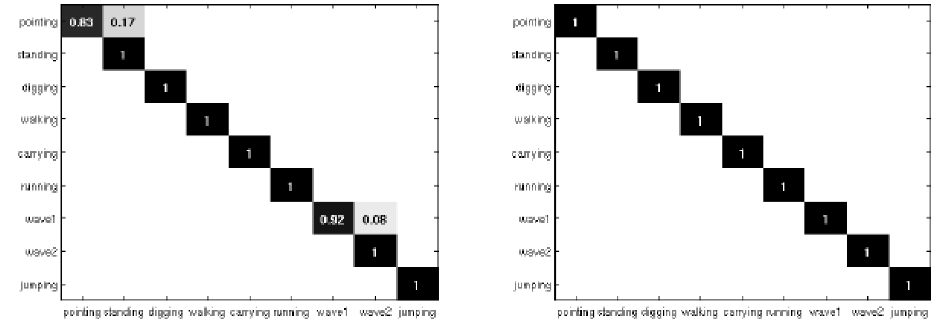


Figure 6: Confusion matrices from classification results on the UT-Tower dat
using SM-1 and SM-2.

Table 5. System accuracies (%) of the aerial-view challenge.

	Point	Stand	Dig	Walk	Carry	Run	Wave1	Wave2	Jump	Total
Team BIWI	100	91.7	100	100	100	100	83.3	83.3	100	95.4
BU	91.7	83.3	100	100	100	100	100	100	100	97.2
ECSU_ISI	100	83.3	91.7	100	100	100	100	91.7	91.7	95.4
Imagelab	83.3	83.3	100	100	100	100	100	100	100	96.3
Baseline	100	83.3	100	100	100	100	83.3	100	100	96.3

UT-Interaction Dataset

Table 4. Activity classification accuracies of the systems tested on the UT-Interaction dataset #2.

	Shake	Hug	Kick	Point	Punch	Push	Total
Laptev + kNN	0.3	0.38	0.76	0.98	0.34	0.22	0.497
Laptev + Bayes.	0.36	0.67	0.62	0.9	0.32	0.4	0.545
Laptev + SVM	0.49	0.64	0.68	0.9	0.47	0.4	0.597
Latpev + SVM (best)	0.5	0.7	0.8	0.9	0.5	0.5	0.65
Cuboid + kNN	0.65	0.75	0.57	0.9	0.58	0.25	0.617
Cuboid + Bayes.	0.26	0.68	0.72	0.94	0.28	0.33	0.535
Cuboid + SVM	0.61	0.75	0.55	0.9	0.59	0.36	0.627
Cuboid + SVM (best)	0.8	0.8	0.6	0.9	0.7	0.4	0.7
Team BIWI	0.5	0.9	1	1	0.8	0.4	0.77

UCF-Sports dataset

- Real data from ESPN/BBC Sports

- Total 200 videos, each class has 15-30 videos.
- Camera motion, varying background
- Quite realistic/challenging

UCF-Sports dataset

Table 4: Results for the UCF-Sports dataset.

Method	Overall Accuracy (%)
Le *et al.* [20]	86.5
Wang *et al.* [34]	88.2
Kovashka *et al.* [17]	87.5
SM-SVM	87.6
SM-1	66.3
SM-2	**96.0**

	dive	g_swing	kick	lift	h_ride	run	skate	b_swing	h_swing	walk
dive	0.86			0.07	0.07					
g_swing		0.39	0.06		0.06	0.06	0.06			0.39
kick	0.05		0.45	0.05	0.1	0.2	0.05	0.05		0.05
lift				1						
h_ride			0.08		0.67	0.25				
run	0.08		0.23		0.15	0.46		0.08		
skate			0.17	0.17		0.08	0.08	0.08		0.42
b_swing			0.05			0.05	0.1	0.8		
h_swing	0.08								0.92	
walk	0.05		0.05		0.05	0.05				0.82

	dive	g_swing	kick	lift	h_ride	run	skate	b_swing	h_swing	walk
dive	1									
g_swing		0.89			0.06					0.06
kick			0.95	0.05						
lift				1						
h_ride			0.08		0.92					
run					0.08	0.92				
skate							1			
b_swing								1		
h_swing									0.92	0.08
walk										1

Close look at 1st Layer results

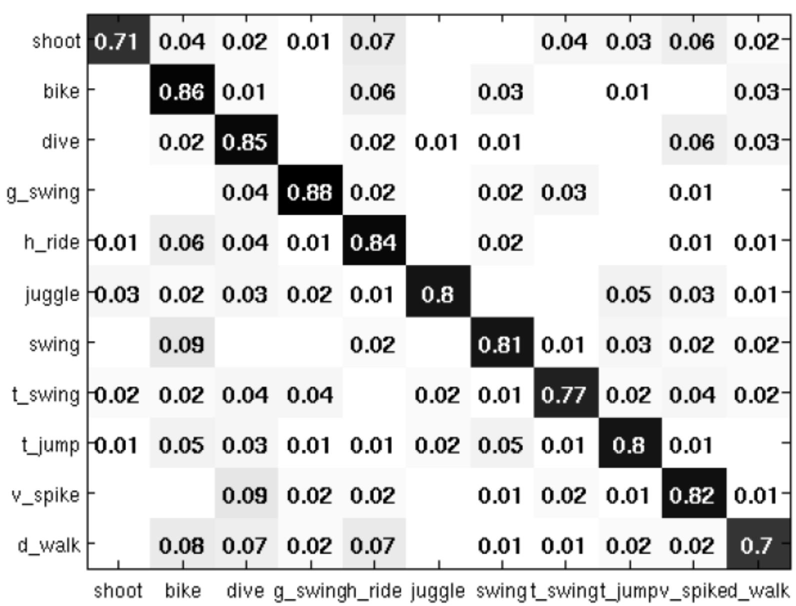
	dive	g_swing	kick	lift	h_ride	run	skate	b_swing	h_swing	walk
dive	0.86			0.07	0.07					
g_swing		0.39	0.06		0.06	0.06	0.06			0.39
kick	0.05		0.45	0.05	0.1	0.2	0.05	0.05		0.05
lift				1						
h_ride			0.08		0.67	0.25				
run	0.08		0.23		0.15	0.46		0.08		
skate			0.17	0.17		0.08	0.08	0.08		0.42
b_swing			0.05			0.05	0.1	0.8		
h_swing	0.08								0.92	
walk	0.05		0.05		0.05	0.05				0.82

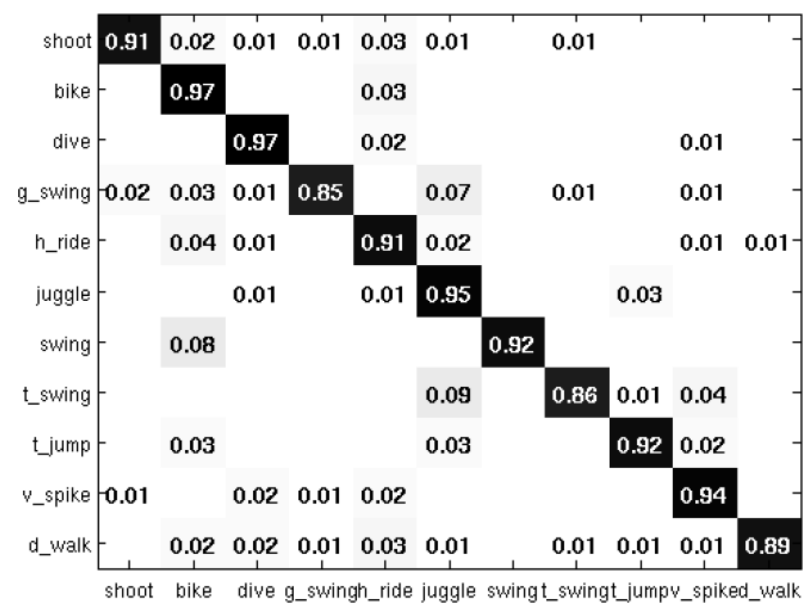
UCF-YouTube dataset

- User-uploaded home video
- Camera motion, background clutter
- Of course different viewing directions

UCF-YouTube dataset

Method	Overall Accuracy (%)
Le *et al.* [20]	75.8
Wang *et al.* [34]	84.2
Ikizler-Cinbis *et al.* [14]	75.2
SM-SVM	88.18
SM-1	80.29
SM-2	**91.9**





Comments from class

- Shahzor:
  - not-scalable with the number of action categories.
  - Hollywood-2: multi-cam shots and rapid scale variations
  - Need multi-scale feature extraction, as well as more sophisticated features
- Ramesh:
  - No rigorous theoretical analysis.
  - Effect of choosing different k, n, and patch size.
  - Non-Negative Sparse Matrix Factorization is slower than L1, why use it?
  - What about using PCA?

Questions & Answers