# Online Semi-Supervised Discriminative Dictionary Learning for Sparse Representation

Guangxiao Zhang, Zhuolin Jiang, Larry S. Davis

Computer Vision Laboratory

University of Maryland, College Park

{gxzhang, zhuolin, lsd}@umiacs.umd.edu

# Motivations

- Traditional dictionary learning focuses on minimizing the reconstruction error only, i.e. $\arg\min_{D,z} ||x - Dz||_2^2$
  - Sparse code $z$ has no discriminative power.
- Supervised dictionary learning:
  - Learning discriminative dictionaries has shown to achieve better performance in image classification tasks.
    - Approach 1: Learn one dictionary for each class, and combine the dictionaries together to obtain a discriminative dictionary.
    - Approach 2: Jointly learn the dictionary and the classifier. (LC-KSVD)
  - Drawbacks of supervised dictionary learning
    - Labeled training data is expensive and difficult to obtain.
    - Not suitable for large-scale dataset.
- Semi-supervised dictionary learning:
  - Learn from a few labeled training data;
  - Also learn from large amount of cheap unlabeled training data;
  - Can be cast to an online learning framework
    - $\Rightarrow$ suitable for large-scale learning
- **Our proposal: online semi-supervised dictionary learning**

# Objective Function

- Objective function should encourage the dictionary to be:
  - Representative : Learning for reconstruction
  - Discriminative : Learning for classification

Jointly learn the dictionary and the classifier

- Proposed optimization:

$$\boxed{< D, G, W, Z > = \arg \min_{D,G,W,Z} \underbrace{\alpha \|X^u - DZ^u\|_2^2}_{\substack{\text{reconstruction} \\ \text{error} \\ \textbf{(unlabeled)}}} + \underbrace{\beta \|X^l - DZ^l\|_2^2}_{\substack{\text{reconstruction} \\ \text{error} \\ \textbf{(labeled)}}} + \underbrace{\gamma \|Q - GZ^l\|_2^2}_{\substack{\text{label consistency} \\ \text{term}}} + \underbrace{\|H - WZ^l\|_2^2}_{\substack{\text{classification} \\ \text{error}}}}$$

s.t. $\|z_i\|_0 \leq \varepsilon, \forall i$

$X$: input signals; $Z$: sparse codes of $X$ with respect to $D$

$Q = [q_1, \dots, q_N]$, label consistency matrix; $G$ is a linear transformation matrix

where $q_i = [q_i^1, \dots, q_i^K]^t$. For example, $[0,1,0, \dots, 1,1]^t$

$q_i^k = 1$ if the input signal $y_i$ and the dictionary item $d_k$ share the same label

A column of $H$, $h_i$, is a label vector for $x_i$, where non-zero position indicates the category label of $x_i$.

- A linear predictive classifier: $f(z; W) = Wz$ is used in the classification.

# Optimization

- Initialization:
  - Learn multiple class-specific dictionaries using K-SVD, and combine the items together to form the initial dictionary $D_0$
- Alternate between sparse coding and dictionary learning:
  - Online sparse coding:
    - At time $t$, given $D_{t-1}, G_{t-1}, W_{t-1}$, find the sparse code $z_t$ for the signal $x_t$
    - For unlabeled $x_t$, $\mathbf{z}_t = \arg\min_{\mathbf{z}\in\mathbb{R}^K} ||\mathbf{x}_t - D\mathbf{z}||_2^2, s.t. ||\mathbf{z}||_0 \leq \varepsilon.$

      The orthogonal matching pursuit (OMP) algorithm is adopted.
    - For labeled $x_t$, the sparse coding problem can be written in augmented matrix form:

$$\mathbf{z}_t = \arg\min_{\mathbf{z}\in\mathbb{R}^K} \left\| \begin{pmatrix} \sqrt{\beta}\mathbf{x}_t \\ \sqrt{\gamma}\mathbf{q}_t \\ \mathbf{h}_t \end{pmatrix} - \begin{pmatrix} \sqrt{\beta}D \\ \sqrt{\gamma}G \\ W \end{pmatrix} \mathbf{z} \right\|_2^2 = \arg\min_{\mathbf{z}\in\mathbb{R}^K} ||\tilde{\mathbf{x}}_t - \tilde{D}\mathbf{z}||_2^2,$$

      which can also be solved by OMP.

$$\tilde{\mathbf{x}}_t = [\sqrt{\beta}\mathbf{x}_t^T, \sqrt{\gamma}\mathbf{q}_t^T, \mathbf{h}_t^T]^t$$

$$\tilde{D} = [\sqrt{\beta}D^T, \sqrt{\gamma}G^T, W^T]^T$$

  - Online dictionary update:
    - Given the sparse code for $x_t$, update the dictionary:

$$D_t = \arg\min_{D\in\mathcal{C}} \frac{1}{t}\sum_{i=1}^{t} \frac{1}{2}||\mathbf{x}_i - D\mathbf{z}_i||_2^2 + \lambda||\mathbf{z}_i||_0$$

# Learning from unlabeled data

- *How to choose which sample in the input stream to label?*

<u>Our goal</u>:

(1) keep the manual labeling effort minimum;

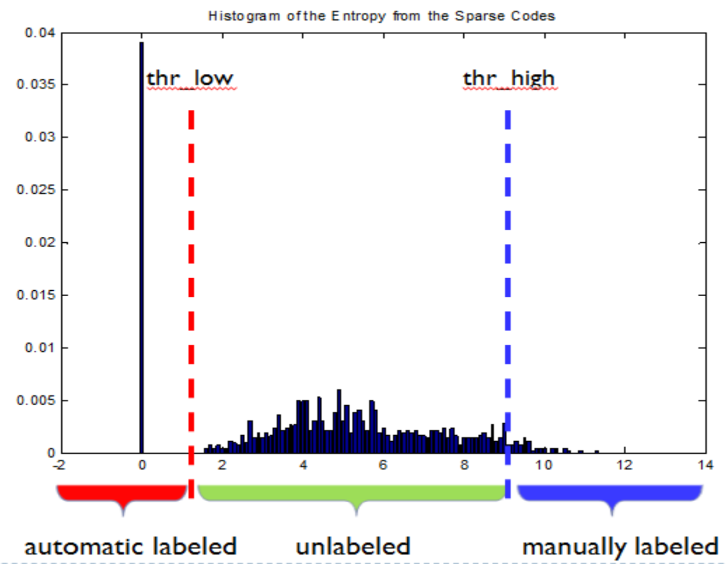
(2) keep discriminative capacity of the sparse codes.

<u>Key observation</u>:

A sparse code is a vector of coefficients of the corresponding dictionary items (with labels).

For example, a sparse code

$$z_i = [\underbrace{z_i^1, z_i^2, \dots, z_i^6}_{d_1 \text{ to } d_6\text{: class 1}}, \underbrace{z_i^7, \dots, z_i^{12}}_{d_7 \text{ to } d_{12}\text{: class 2}}, \dots, z_i^K]^T$$




- $z_i^j$ can be used to compute the probability of $x_i$ being in the same class as dictionary item $d_j$.

The sparse code informs us how well the current dictionary discriminates the input signal. Quantitatively, the confidence level of the discriminability is defined as:

$$ent(\mathrm{x}) = -\sum_{l=1}^{m} p_l(\mathrm{x}) \log p_l(\mathrm{x})$$ , where $p_l(\mathrm{x})$ is the probability of x being in class $l$.

<u>Set two thresholds on entropy</u>: "easy" points: automatic labeling; "hard" points: manual labeling.

# An Outline of Our Algorithm

**Input:** input signals $X = \{x_1, \ldots, x_N\}$ and their labels, if any;
regularization constants $\alpha, \beta, \gamma$;
lower and upper bound of entropy $\phi_{low}, \phi_{high}$

**Initialization:** Compute $D_0, G_0, W_0$ via LC-KSVD

for $t = 1, 2, \ldots, N$ do

Draw $x_t$ from the input sequence;

Compute sparse code $z_t$;

if $x_t$ is unlabeled,

Compute the entropy of the sparse code $z_t$;

if entropy $< \phi_{low}$,

Automatically label it as the most probable class;

Dictionary update by labeled point $x_t$ :

$$D_{t-1} \rightarrow D_t; G_{t-1} \rightarrow G_t; W_{t-1} \rightarrow W_t$$

else if entropy $> \phi_{high}$,

Manually label it by the user;

Augmented dictionary update by labeled point $x_t$ :

$$D_{t-1} \rightarrow D_t; G_{t-1} \rightarrow G_t; W_{t-1} \rightarrow W_t$$

else

Augmented dictionary update by unlabeled data:

$$D_{t-1} \rightarrow D_t;$$

else if $x_t$ is labeled,

Augmented dictionary update by labeled point $x_t$ :

$$D_{t-1} \rightarrow D_t; G_{t-1} \rightarrow G_t; W_{t-1} \rightarrow W_t$$

end for

**Return:** $D, G$, and $W$

Combine multiple class-specific dictionaries

Sparse coding

*Unlabeled point:*

1. Decide if it can be automatically labeled
2. Decide if it needs manual labeling
3. Dictionary update

*Labeled point:*
Dictionary update

# An Outline of Our Algorithm

- **Dictionary Update:**

**Algorithm 1:** Dictionary Update

**Input:** current dictionary $D_{t-1}$;
$A_t = \sum_{i=1}^{t} \mathbf{z}_i \mathbf{z}_i^T = [\mathbf{a}_1 ... \mathbf{a}_t]$,
$B_t = \sum_{i=1}^{t} \mathbf{x}_i \mathbf{z}_i^T = [\mathbf{b}_1 ... \mathbf{b}_t]$;
**Output:** updated dictionary $D_t$.
**repeat**
  **for** $j = 1, 2, ...., K$ **do**
    Update the $j$-th column
      $\mathbf{u}_j \leftarrow \frac{1}{A_{j,j}}(\mathbf{b}_j - D\mathbf{a}_j) + \mathbf{d}_j$.
      $\mathbf{d}_j \leftarrow \frac{1}{\max ||\mathbf{u}_j||_2, 1} \mathbf{u}_j$.
  **end for**
**until convergence**
**Return**

This algorithm can also be used for augmented dictionary

Matrices A and B records information from the past

Iteratively update each dictionary items

- **Classification**
  - For a test image $x_i$, first compute its sparse representation:

$$z_i = \arg\min_z ||x_i - Dz||_2^2, \text{ s.t. } ||z_i||_0 \leq \varepsilon$$

Then the label of $x_i$ is the index $i$ corresponding to the largest element of a class label vector .

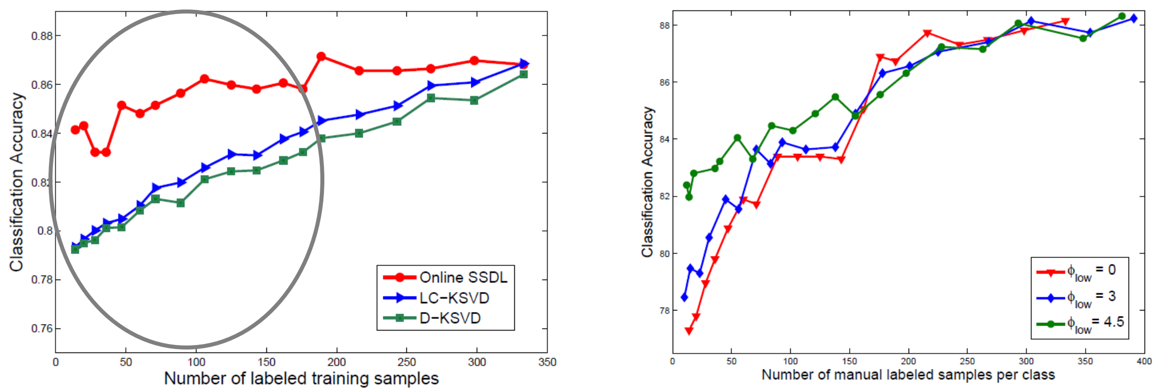
# Experiment (1)

- Extended YaleB database:
  - Random face-based features

    - feature dims = 504 ; number of dictionary items: 6*38=228
  - Classification accuracy comparison:

| Method | K-SVD [11] | D-KSVD [5] | SRC [3] | LLC [14] | LC-KSVD [9] |
|---|---|---|---|---|---|
| **Acc.** | 93.1 | 94.1 | 80.5 | 82.2 | 94.5 |
| **Method** | LSDL [17] | ODLSC [13] | IDL [14] | **Online SSDL** | |
| **Acc.** | 90.5 | 91.4 | 89.6 | **94.7** | |

  - Semi-supervised learning curves:



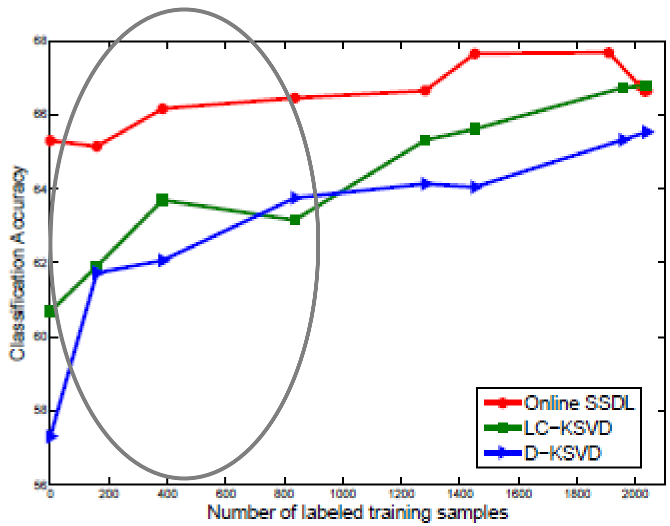

(a) (b)

**Fig. 2.** Recognition performance on the Extended YaleB. (a) Recognition performance with varying number of labeled samples, where $K = 6 \times 38$ and $N = 24 \times 38$; (b) An illustration of the effect of the lower bound. The curves are obtained with the same set of parameters: $\alpha$, $\beta$, $\gamma$ and the same set of higher entropy thresholds.

# Experiment (2)

- Caltech 101 dataset:
  - Spatial pyramid features

- feature dim. = 3000;
- number of dictionary items:10*102=1020




(a)

| Training Images | 5 | 10 | 15 | 20 | 25 | 30 |
|---|---|---|---|---|---|---|
| Malik [28] | 46.6 | 55.8 | 59.1 | 62.0 | - | 66.20 |
| Lazebnik [29] | - | - | 56.4 | - | - | 64.6 |
| Griffin [27] | 44.2 | 54.5 | 59.0 | 63.3 | 65.8 | 67.60 |
| Irani [30] | - | - | 65.0 | - | - | 70.40 |
| Grauman [31] | - | - | 61.0 | - | - | 69.10 |
| Venkatesh [6] | - | - | 42.0 | - | - | - |
| Gemert [32] | - | - | - | - | - | 64.16 |
| Yang [2] | - | - | 67.0 | - | - | 73.20 |
| Wang [14] | 51.15 | 59.77 | 65.43 | 67.74 | 70.16 | 73.44 |
| SRC [3] | 48.8 | 60.1 | 64.9 | 67.7 | 69.2 | 70.7 |
| K-SVD [11] | 49.8 | 59.8 | 65.2 | 68.7 | 71.0 | 73.2 |
| D-KSVD [5] | 49.6 | 59.5 | 65.1 | 68.6 | 71.1 | 73.0 |
| IDL [14] | 51.2 | 61.5 | 65.7 | 68.4 | 71.6 | - |
| LSDL [17] | 52.8 | 61.5 | 65.7 | 68.4 | 71.5 | - |
| ODLSC [13] | 52.8 | 61.5 | 65.6 | 68.5 | 71.3 | 72.4 |
| LC-KSVD [9] | 54.0 | 63.1 | 67.7 | 70.5 | 72.3 | 73.6 |
| **Online SSDL** | **55.0** | **62.6** | **67.2** | **69.6** | **72.4** | **74.3** |

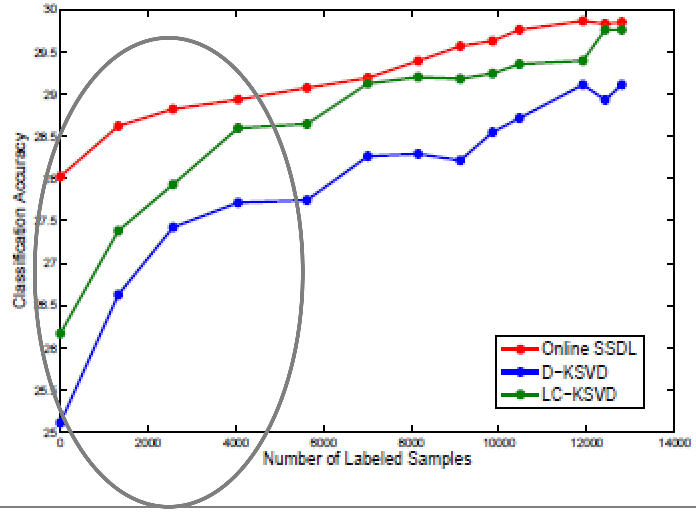
**Table 2.** Recognition results using spatial pyramid features on the Caltech101. The accuracies of the other results are copied from the references.

# Experiment (3) and Conclusion

- Caltech 256 dataset
  - Spatial pyramid features

  - feature dim. = 305 (PCA applied)

  - number of dictionary items: 3*256=768



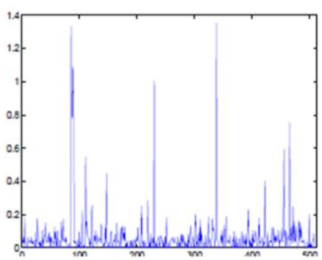

- Accuracy comparison and the learning curves (right) :

| Training Images | 15 | 30 | 45 | 60 |
|---|---|---|---|---|
| Griffin [27] | 28.30 | 34.10 | - | - |
| Gemert [32] | - | 27.17 | - | - |
| Yang [2] | 27.73 | 34.02 | 37.46 | 40.14 |
| IDL [14] | 19.9 | 21.7 | 23.9 | 26.3 |
| LSDL [17] | 23.3 | 25.6 | 28.4 | 30.5 |
| ODLSC [13] | 19.3 | 21.3 | 23.6 | 26.1 |
| LC-KSVD [9] | 24.6 | 28.6 | 30.3 | 34.9 |
| **Online SSDL** | **27.9** | **31.9** | **34.4** | **36.7** |

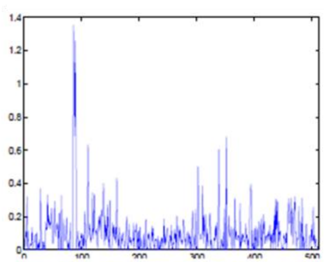
- Notice that our semi-supervised method has an obvious advantage when the manual labels are few.
- As the number of manual labels increases, the advantage over others decreases, until our performance finally converges to fully-supervised methods.

# Examples of sparse codes

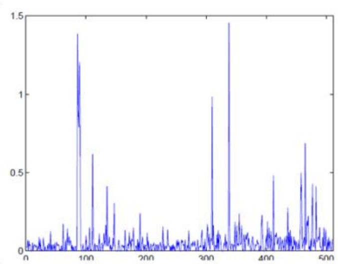
- Caltech 101, Class 18 fans (with 61 testing frames):



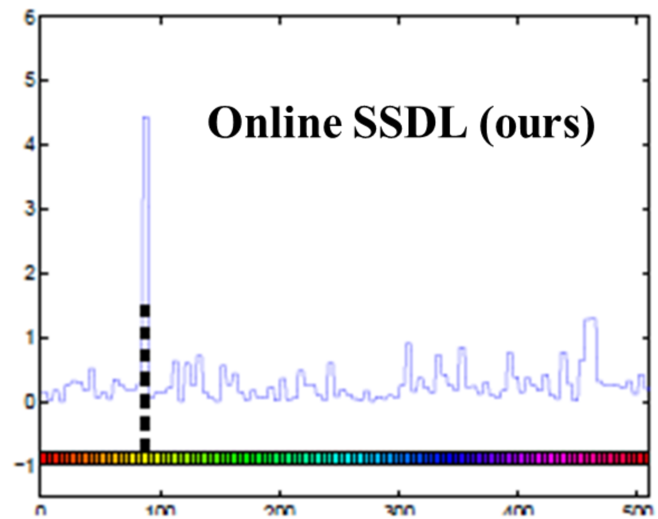
ODLSC[4]



IDL[5]



LSDL[6]




- Y-axis indicates a sum of absolute sparse codes.
- Sparse codes are expected to peak at the 18*5 = 90$^{th}$ , where 5 being the number of dictionary items per category and 18 being the category index.

# References and Acknowledgement

- **Key references**


1. J. Wright, A. Yang, A. Ganesh, S. Sastry and Y. Ma. Robust face recognition via sparse representation, TPAMI 2009.
2. M. Aharon, M. Elad and A. Bruchstein. K-SVD: An algorithm for designing over-complete dictionaries for sparse representation. Sig. Proc., 2006.
3. Q. Zhang and B. Li. Discriminative k-svd for dictionary learning in face recognition, CVPR 2010.
4. J. Marial, F. Bach, J. Ponce and G. Sapiro: Online dictionary learning for sparse coding, ICML 2009
5. J. Want, J. Yang, K. Yu, F. Lv, T. Huang and Y. Gong: Locality-constrained linear coding for image classification, CVPR 2010
6. B. Xie, M. Song, D.T. : Large-scale dictionary learning for local coordinate coding BMVC, 2010
7. Z. Jiang, Z. Lin, and L. Davis: Learning a discriminative dictionary for sparse coding via label consistent k-svd, CVPR 2011


- **Acknowledgement:**


This work was supported by the Army Research Office MURI Grant W911NF-09-1-0383