Online/Incremental Learning
Merging and Splitting Eigenspace Models

Tae-Kyun (T-K) Kim
KAIST, Imperial College London
https://sites.google.com/view/tkkim/

Further reading:

T-K. Kim, B. Stenger, J. Kittler and R. Cipolla, Incremental Linear Discriminant Analysis Using Sufficient Spanning Sets and Its Applications, International Journal of Computer Vision, 91(2):216-232, 2011.

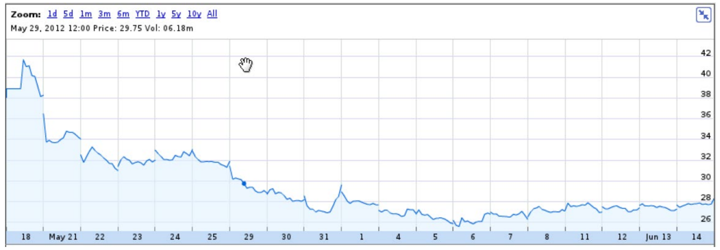
P Hall, D Marshall, R Martin, Merging and splitting eigenspace models, IEEE Trans. on PAMI, 22 (9), 1042-1049, 2000.

Why Learning Online?

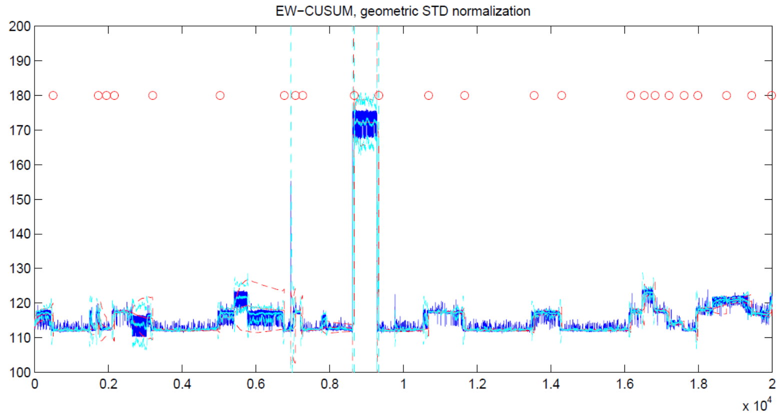
- Non-stationarity of real data
  - ⇒ Update of models is needed when new data is available
  - ⇒ Evaluation of relevance of past data

- Learning on a tight budget
  - ⇒ Some data may be irrelevant
  - ⇒ Cost-sensitive learning

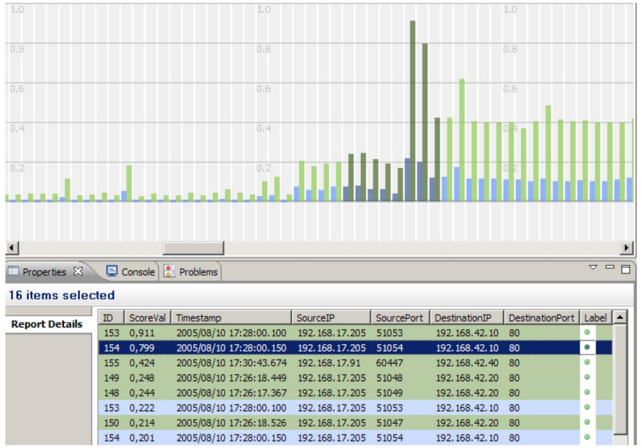
Applications of Online Learning



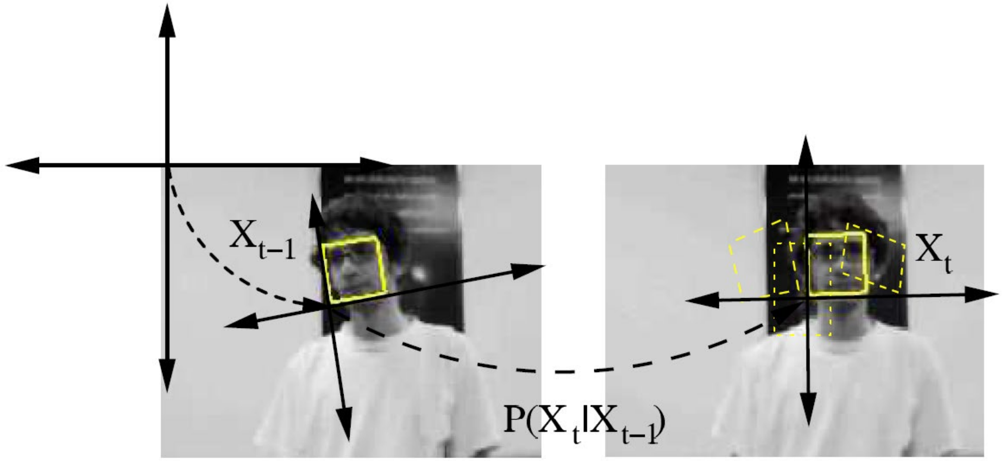
Time series analysis



Anomaly and change point detection



Security event monitoring



Object tracking

Computational Considerations

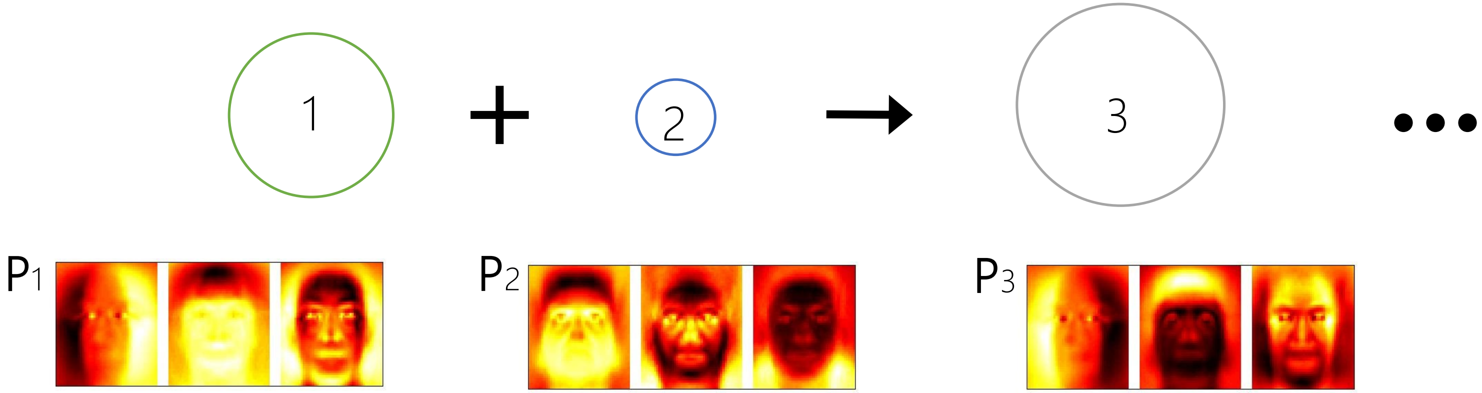
- How much do we pay for training?
  - PCA: O($\boldsymbol{n}^3$) (eigenvalue decomposition)
  - Ridge regression: O($\boldsymbol{n}^3$) (matrix inversion)
  - SVM: O($\boldsymbol{n}^2 \log \boldsymbol{n}$) (theoretical bound on feasible direction decomposition)

- How much are we willing to pay?
  - An order of magnitude less: to match batch learning
  - Constant or linear time: if we are really greedy!

Dynamically updating eigenspace

- Eigenspace models have a wide variety of applications, such as classification for recognition systems.
- In practice, we need to build the engenspace models for numerous images: those images may not be given all initially, but incrementally.

- Our goal is to dynamically update the eigenspace models, when new data entries are given or existing data points are removed.
- The mean also needs to be updated.



Incremental PCA

Batch vs Incremental

- In batch computation: all observations are used simultaneously to compute the eigenspace model.
- In incremental computation: an existing eigenspace model is updated using new observations.

Requirements: methods need to

- handle a change in the mean.
- add multiple new observations than exactly one observation at a time.

Pros and Cons

- Benefits: an incremental method
  - does not need all observations at once - thus, reducing storage requirements and making large problems feasible.
  - Even if all observations are available, is usually faster to compute a new eigenspace model by incrementally than by batch computation.
- Disadvantage: is their accuracy compared to batch methods. When only a few incremental updates are made, the inaccuracy is small.

Merging and splitting eigenspace models

- We learn a deterministic method that given two eigenspace models - each representing a set of D-dimensional observations - will:

  1) Merge the models to yield a representation of the union of the sets,
  2) Split one model from another to represent the difference between the sets.

Eigenspace models and notations

- For a set of N data vectors, $\mathbf{x} \in R^D$, the covariance matrix is

$$\mathbf{S} = \frac{1}{N}\sum_{all\ \mathbf{x}} (\mathbf{x}-\boldsymbol{\mu})(\mathbf{x}-\boldsymbol{\mu})^T$$

where μ is the data mean.

- PCA decomposes the covariance matrix s.t.

$$\mathbf{S} \cong \mathbf{P}\boldsymbol{\Lambda}\mathbf{P}^{\mathrm{T}}$$

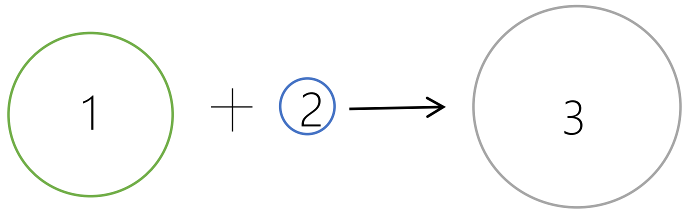
where P is the matrix containing the first d eigenvectors in columns, and Λ is the diagonal matrix with the first d eigenvalues.

Incremental PCA

- Problem setting:

Input : given two sets of data represented by eigenspace models $\{\boldsymbol{\mu}_i, N_i, \mathbf{P}_i, \boldsymbol{\Lambda}_i\}_{i=1,2}$

Output : compute the eigenspace model of the combined data $\{\boldsymbol{\mu}_3, N_3, \mathbf{P}_3, \boldsymbol{\Lambda}_3\}$



- The combined mean is obtained as

$$\boldsymbol{\mu}_3 = (N_1\boldsymbol{\mu}_1 + N_2\boldsymbol{\mu}_2)/N_3$$

- The combined covariance matrix is

$$\mathbf{S}_3 = \frac{N_1}{N_3}\mathbf{S}_1 + \frac{N_2}{N_3}\mathbf{S}_2 + \frac{N_1 N_2}{N_3^2}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\boldsymbol{T}}$$

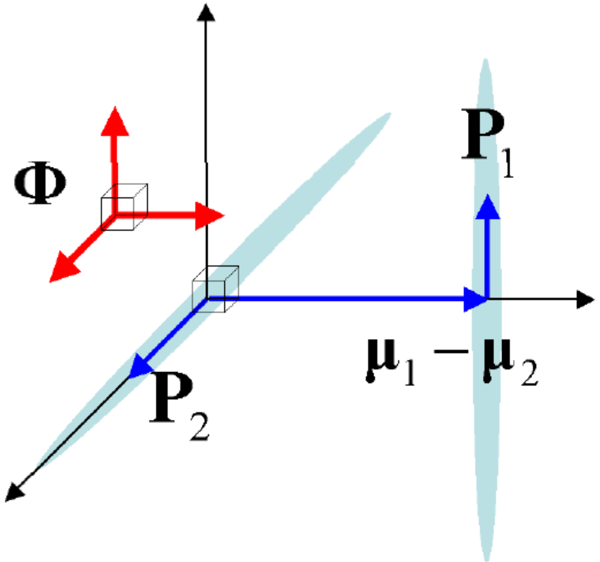
where $\{\mathbf{S}_\mathbf{i}\}$, i=1,2 are the covariance matrices of the first two sets and $N_3 = N_1 + N_2$.

Incremental PCA

- The eigenvector matrix $\mathbf{P}_3$ can be represented as

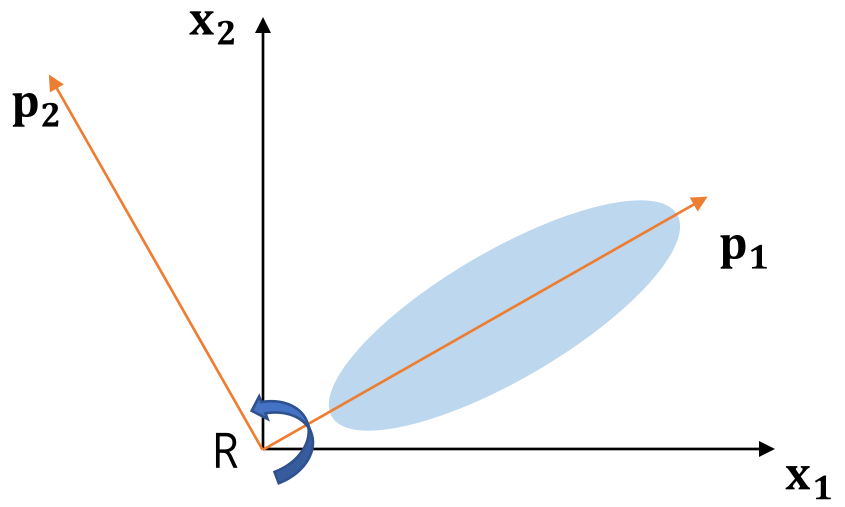
$$\mathbf{P}_3 = \mathbf{\Phi R} = h([\mathbf{P}_1, \mathbf{P}_2, \boldsymbol{\mu}_1 - \boldsymbol{\mu}_2])\mathbf{R}$$

where,
$\mathbf{\Phi}$ is the orthonormal matrix spanning the combined covariance matrix i.e. *the sufficient spanning set*,
R is a rotation matrix, and
h is an *orthonormalization function followed by removal of zero vectors.



*: e.g. Gram-Schmidt orthogonalization, https://en.wikipedia.org/wiki/Gram–Schmidt_process

PCA by the sufficient spanning set



$P = [\mathbf{p_1}, \mathbf{p_2}] \in R^{Dx2} = \Phi R$

$\Phi = [\mathbf{x_1}, \mathbf{x_2}] \in R^{Dx2}$: the sufficient spanning set

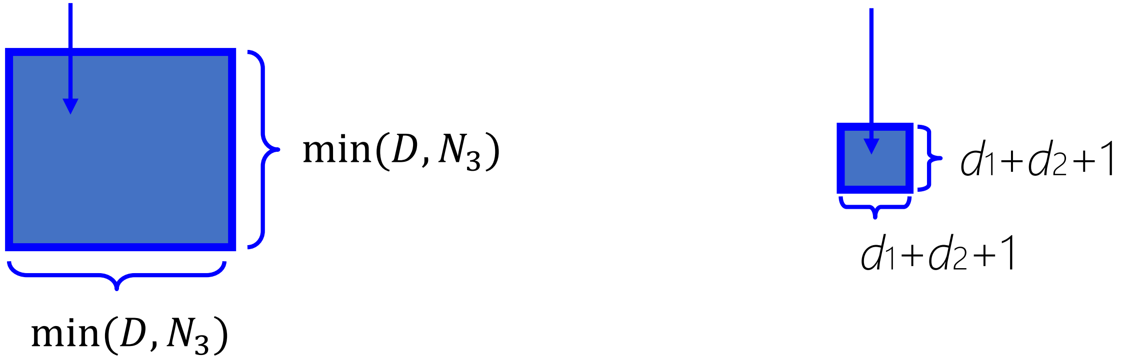
$R \in R^{2x2}$: a rotation matrix

We can reduce the dimension by removing eigenvectors with non-significant eigenvalues: $P = [\mathbf{p_1}] \in R^{Dx1}$

Incremental PCA

- Using this representation $\mathbf{P}_3 = \mathbf{\Phi R}$, the eigenproblem is converted into a smaller eigenproblem as

$$\mathbf{S}_3 \cong \mathbf{P}_3 \mathbf{\Lambda}_3 \mathbf{P}_3^{\mathrm{T}} \Longrightarrow \mathbf{\Phi}^T \mathbf{S}_3 \mathbf{\Phi} \cong \mathbf{R} \mathbf{\Lambda}_3 \mathbf{R}^T$$



- By computing the eigendecomposition on the r.h.s. Λ_3 and R are obtained as the respective eigenvalue and eigenvector matrices.

Incremental PCA

- The eigenvector matrix to seek is given as

$$\mathbf{P}_3 = \mathbf{\Phi}\mathbf{R}$$

- Note the eigenanalysis on the r.h.s. only takes computations

$$O((d_1 + d_2 + 1)^3)$$

Where *d*1, *d*2 are the number of the eigenvectors stored in P1 and P2.

- The eigenanalysis in a batch mode on the l.h.s. requires

$$O(\min(D, N_3)^3)$$

Splitting eigenspace models

- Problem setting:

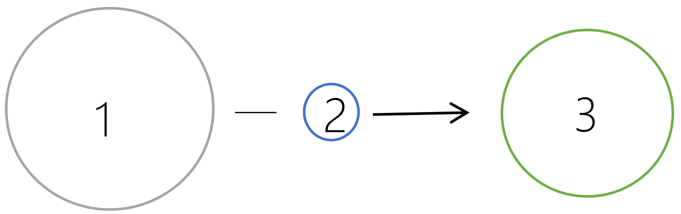
Input : given the first eigenspace model,

we remove the second from it,

$$\{\boldsymbol{\mu}_i, N_i, \mathbf{P}_i, \boldsymbol{\Lambda}_i\}_{i=1,2}$$

Output : to give the third model

$$\{\boldsymbol{\mu}_3, N_3, \mathbf{P}_3, \boldsymbol{\Lambda}_3\}$$



- Splitting means removing a subset of observations; the method is the inverse of merging in this sense.

- $N_3 = N_1 - N_2$.
- The new mean is: $\mu_3 = (N_1\mu_1 - N_2\mu_2)/N_3$

Splitting eigenspace models

- The new covariance matrix is

$$\mathbf{S}_3 = \frac{N_1}{N_3}\mathbf{S}_1 - \frac{N_2}{N_3}\mathbf{S}_2 - \frac{N_2}{N_1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_3)(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_3)^{\boldsymbol{T}}$$

- The eigenvector matrix P_3 can be represented as $P_3 = \Phi R = P_1 R$

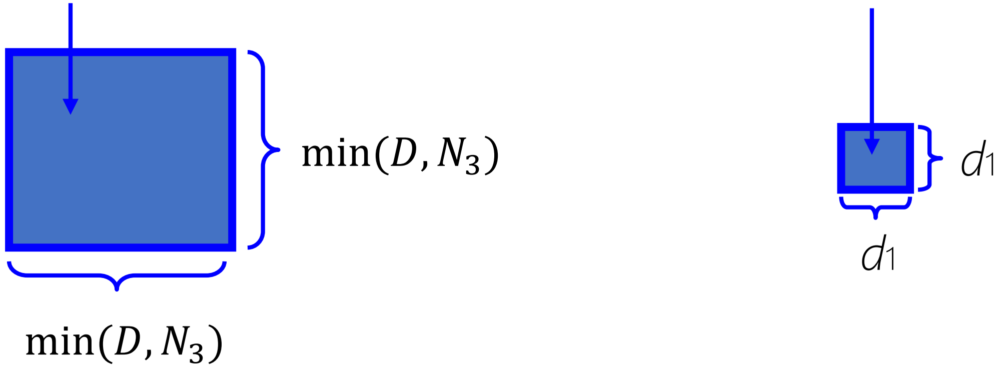
where
Φ is the orthonormal matrix spanning the new covariance matrix
i.e. *the sufficient spanning set*, and
R is a rotation matrix.

- It is impossible to regenerate information which was discarded when the overall model was created. Thus, if we split one eigenspace model from a larger one, the eigenvectors of the remnant must still form some subspace of the larger.

Splitting Eigenspace Models

- Using this representation, the eigenproblem is converted into a smaller eigenproblem as

$$\mathbf{S}_3 \cong \mathbf{P}_3\boldsymbol{\Lambda}_3\mathbf{P}_3^{\mathrm{T}} \quad \longrightarrow \quad \boldsymbol{\Phi}^T\mathbf{S}_3\boldsymbol{\Phi} \cong \mathbf{R}\boldsymbol{\Lambda}_3\mathbf{R}^T$$



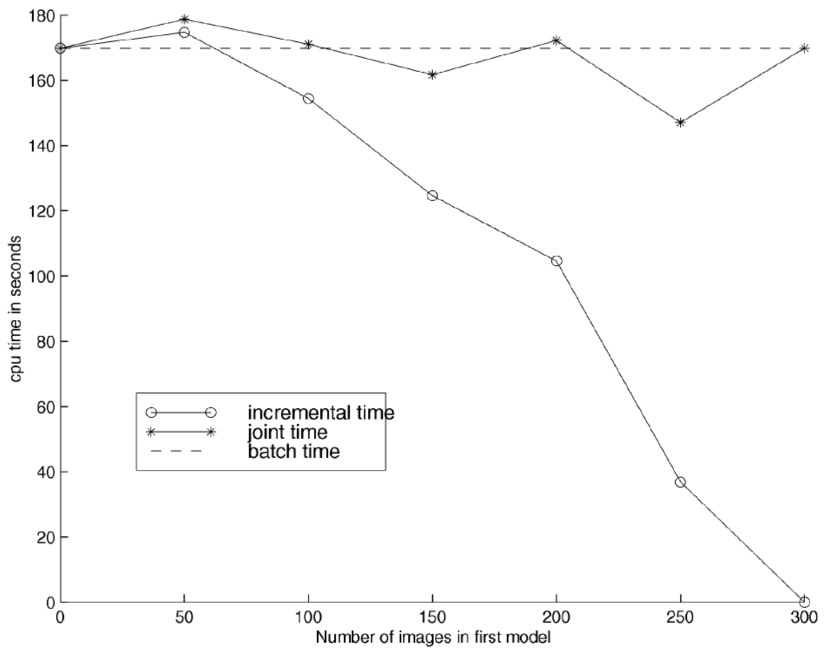
- By computing the eigendecomposition on the r.h.s. Λ_3 and R are obtained as the respective eigenvalue and eigenvector matrices.

- The eigenvector matrix to seek is given as $P_3 = \Phi R = P_1 R$

Results

- \- We examine the efficiency and accuracy of the incremental method, compared to the batch method.
- \- We used a database of 300 face images (each of 112x92=10,304 pixels).
- \- The gray levels in the images were scaled into the range [0, 1] by division only, but no other preprocessing was done.
- \- 300 images were partitioned into two data sets, each containing a multiple of 50 images.
- \- The number of eigenvectors retained in any model, including a merged model, was set to be 100 as a maximum, for ease of comparing results.

- \- The *incremental* time is the time needed to compute and then merge one eigenspace model to an existing one.
- \- The *joint* time is the time to compute both eigenmodels and then merge them.



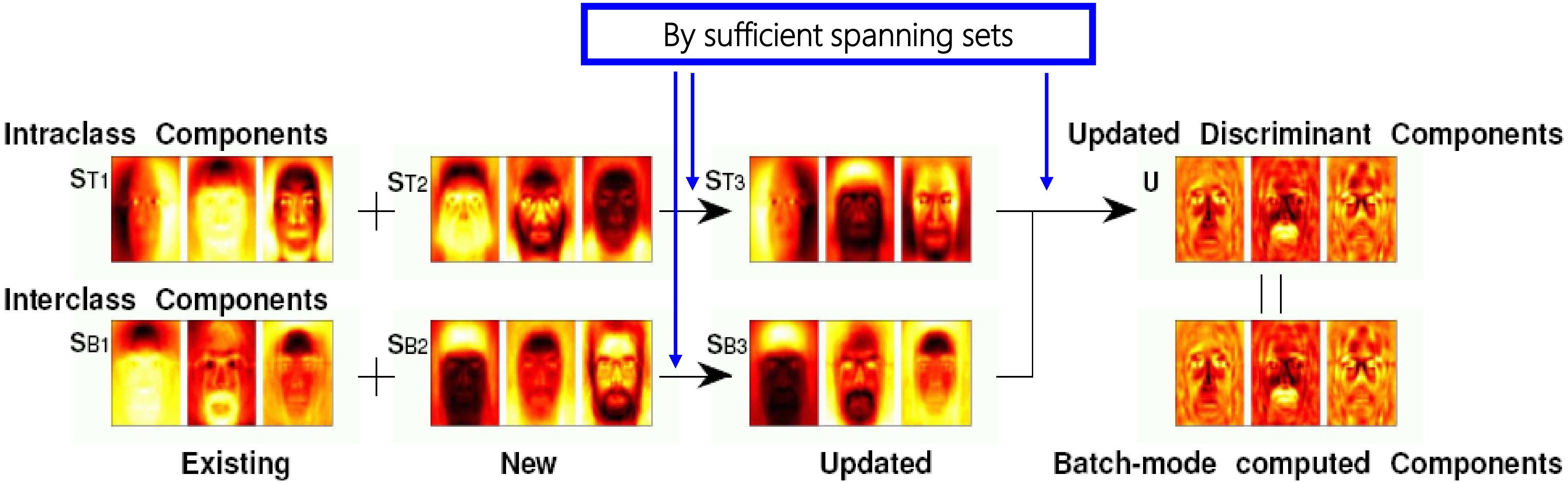
Results

- First compared the means of the models produced by each method using Euclidean distance.
- Next compared the directions of the eigenvectors produced by each method, the error measured by the mean angular deviation of corresponding eigenvectors.
- The sizes of eigenvalues from both methods were compared, more precisely the mean relative absolute error was measured: $|a - b|/a$ for eigenvalues a and b.

Experiments

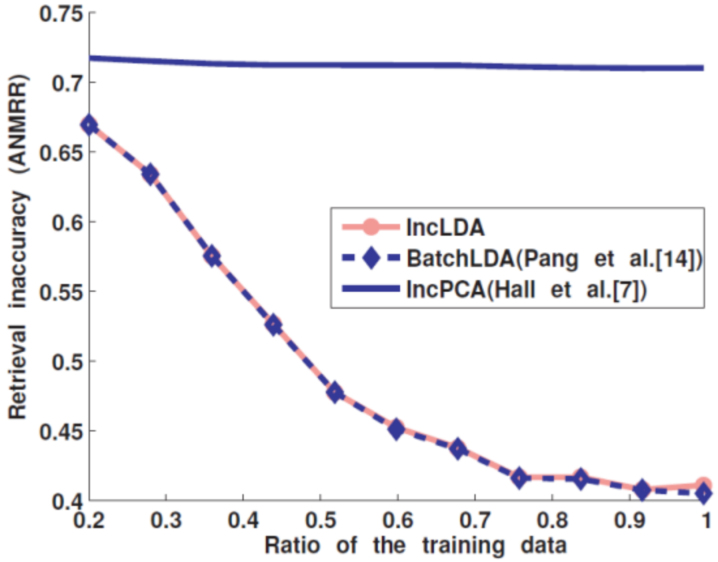
- Similarly, we can compute LDA (Linear Discriminant Anlaysis) incrementally.
- We apply the *sufficient spanning set* approximation in each update step, i.e. for the between-class scatter matrix, the total scatter matrix and the projected data matrix:

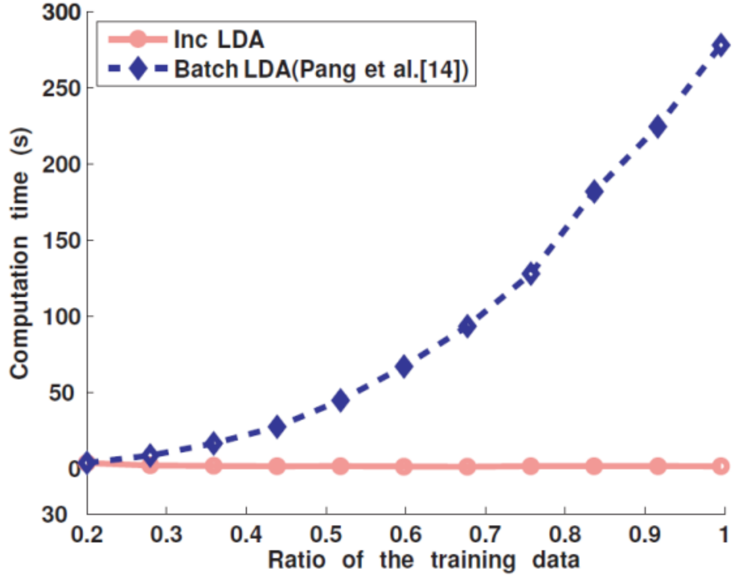
$$\max_{\arg \mathbf{U}} \frac{\mathbf{U}^T \mathbf{S}_B \mathbf{U}}{\mathbf{U}^T \mathbf{S}_W \mathbf{U}} = \boxed{\max_{\arg \mathbf{U}} \frac{\mathbf{U}^T \mathbf{S}_B \mathbf{U}}{\mathbf{U}^T \mathbf{S}_T \mathbf{U}}} \qquad \mathbf{S}_T = \mathbf{S}_B + \mathbf{S}_W$$



Experiments

- MPEG7 face image datasets of 6370 images





Experiments

- Caltech101 datasets (using BoW representations), up to 800 images per category