Unitary Transforms

- Unitary Transforms
  - Energy conservation and distribution
  - Karhunen-Loeve Transform
- Basis images and eigenimages
  - Eigenimages for recognition
  - Computing eigenimages from a training set

Linear Image Processing Problems

For the linear image processing system

$$\vec{g} = \mathbf{H}\vec{f}$$

how does one choose **H** . . .

- . . . so $\vec{g}$ separates the salient features from the image signal?
- . . . so $\vec{g}$ looks better?
- . . . in order for $\vec{g}$ to be sparse?

Unitary Transforms

- Sort samples f(x,y) of an MxN image (or a rectangular block in the image) into a column vector of length MN
- Compute transform coefficients

$$\vec{c} = \mathbf{A}\vec{f}$$

  where **A** is a matrix of size MNxMN
- The transform **A** is unitary, iff

$$\mathbf{A}^{-1} = \mathbf{A}^{*T} \equiv \mathbf{A}^{H}$$

Hermitian conjugate

- If **A** is real-valued, i.e., **A**=**A***, transform is **orthonormal**

Energy Conservation with Unitary Transforms

- For any unitary transform $\vec{c} = \mathbf{A}\vec{f}$, we obtain

$$||\vec{c}||_2^2 = \vec{c}^H \vec{c} = \vec{f}^H \mathbf{A}^H \mathbf{A} \vec{f} = ||\vec{f}||_2^2$$

- Interpretation: Every unitary transform is simply a rotation of the coordinate system (and, possibly, sign flips).
- Vector lengths (“energies”) are conserved.

Energy Distribution for Unitary Transforms

- Energy is conserved, but often will be unevenly distributed among coefficients.
- Autocorrelation matrix

$$\mathbf{R}_{cc} = E\left\{\vec{c}\vec{c}^{H}\right\} = E\left\{\mathbf{A}\vec{f}\vec{f}^{H}\mathbf{A}^{H}\right\} = \mathbf{A}\mathbf{R}_{ff}\mathbf{A}^{H}$$

- Mean squared values ("average energies") of the coefficients c_i are on the diagonal of $\mathbf{R}_{cc}$

$$E\left\{c_i^2\right\} = [\mathbf{R}_{cc}]_{i,i}$$

Eigenmatrix of the Autocorrelation Matrix

Definition: Eigenmatrix $\boldsymbol{\Phi}$ of the autocorrelation matrix $\mathbf{R}_{ff}$

- $\boldsymbol{\Phi}$ is unitary
- The columns of $\boldsymbol{\Phi}$ form a set of eigenvectors of $\mathbf{R}_{ff}$, i.e.,

$$\mathbf{R}_{ff}\boldsymbol{\Phi} = \boldsymbol{\Phi}\boldsymbol{\Lambda}$$

$\boldsymbol{\Lambda}$ is a diagonal matrix of eigenvalues λ_i

$$\boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & & & 0 \\ & \lambda_2 & & \\ & & \ddots & \\ 0 & & & \lambda_{MN} \end{bmatrix}$$

- $\mathbf{R}_{ff}$ is symmetric nonnegative definite, hence $\lambda_i \geq 0$ for all i
- $\mathbf{R}_{ff}$ is normal matrix, i.e., $\mathbf{R}_{ff}^H \mathbf{R}_{ff} = \mathbf{R}_{ff} \mathbf{R}_{ff}^H$, hence unitary eigenmatrix exists

Karhunen-Loeve Transform

- Unitary transform with matrix

$$\mathbf{A} = \mathbf{\Phi}^H$$

  where the columns of $\mathbf{\Phi}$ are ordered according to decreasing eigenvalues.
- Transform coefficients are pairwise uncorrelated

$$\mathbf{R}_{cc} = \mathbf{A}\mathbf{R}_{ff}\mathbf{A}^H = \mathbf{\Phi}^H\mathbf{R}_{ff}\mathbf{\Phi} = \mathbf{\Phi}^H\mathbf{\Phi}\mathbf{\Lambda} = \mathbf{\Lambda}$$

- Energy concentration property:
  - No other unitary transform packs as much energy into the first J coefficients, where J is arbitrary
  - Mean squared approximation error by choosing only first J coefficients is minimized.

Optimum Energy Concentration by KLT

- To show optimum energy concentration property, consider the truncated coefficient vector

$$\vec{b} = I_J \vec{c}$$

- Where I_J contains ones on the first J diagonal positions, else zeros.
- Energy in first J coefficients for arbitrary transform A

$$E = tr(\mathbf{R}_{bb}) = tr(I_J \mathbf{R}_{cc} I_J) = tr(I_J \mathbf{A} \mathbf{R}_{ff} \mathbf{A}^H I_J) = \sum_{k=1}^{J} a_k^T \mathbf{R}_{ff} a_k^*$$

where a_k^T is the k-th row of **A**.

- Lagrangian cost function to enforce unit-length basis vectors

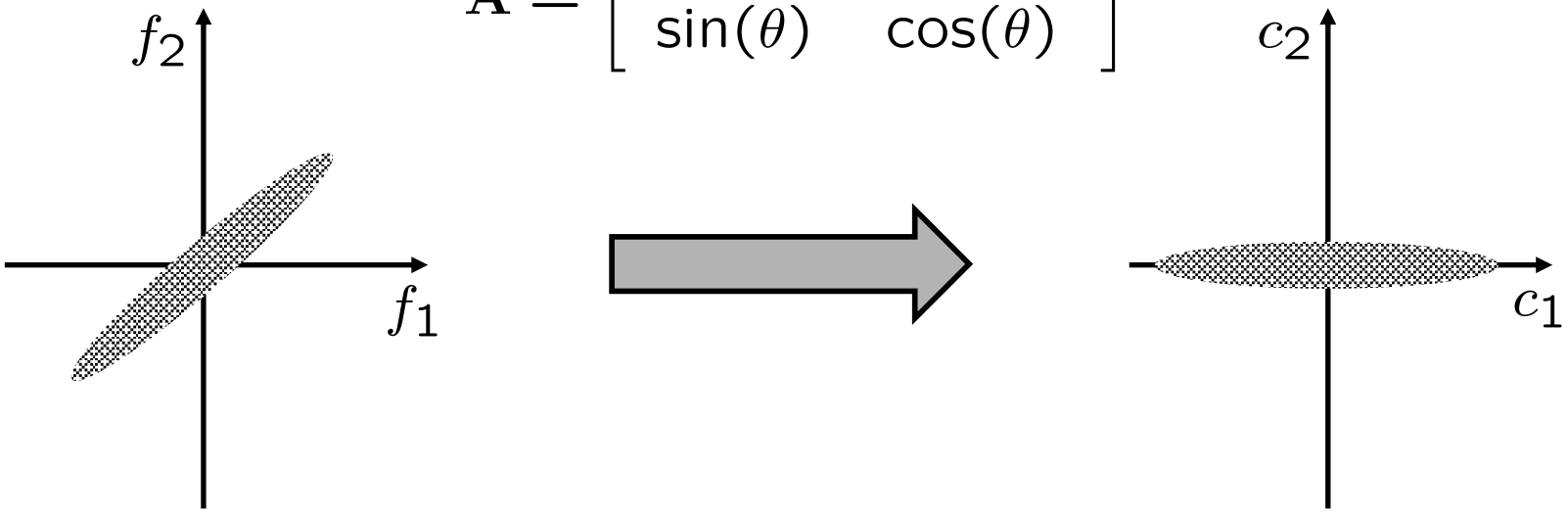
$$L = E + \sum_{k=1}^{J} \lambda_k (1 - a_k^T a_k^*) = \sum_{k=1}^{J} a_k^T \mathbf{R}_{ff} a_k^* + \sum_{k=1}^{J} \lambda_k (1 - a_k^T a_k^*)$$

- Differentiating L with respect to a_j yields necessary condition

$$\mathbf{R}_{ff} a_j^* = \lambda_j a_j^* \quad \forall \quad j \leq J$$

Illustration of Energy Concentration

$$\mathbf{A} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$



Before KLT:
Strongly correlated samples, equal energies

After KLT:
Uncorrelated samples, most of the energy in first coefficient

Basis Images and Eigenimages

- For any unitary transform, the inverse transform

$$\vec{f} = \mathbf{A}^H \vec{c}$$

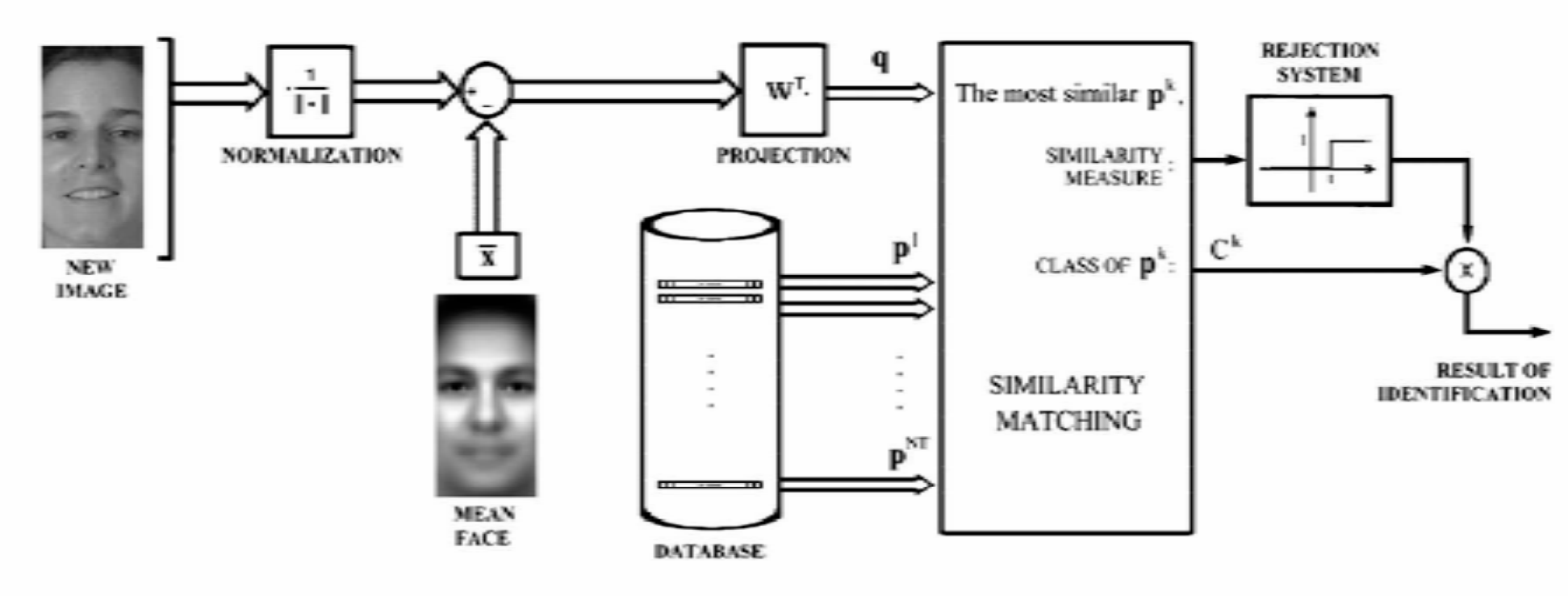
  can be interpreted in terms of the superposition of "basis images" (columns of $\mathbf{A}^H$) of size MN.
- If the transform is a KLT, the basis images, which are the eigenvectors of the autocorrelation matrix $\mathbf{R}_{ff}$, are called "eigenimages."
- If energy concentration works well, only a limited number of eigenimages is needed to approximate a set of images with small error. These eigenimages form an optimal linear subspace of dimensionality J.

Eigenimages for Recognition

- To recognize complex patterns (e.g., faces), large portions of an image (say of size MN) might have to be considered
- High dimensionality of "image space" means high computational burden for many recognition techniques

  Example: Nearest-neighbor search requires pairwise comparison with every image in a data base
- Transform $\vec{c} = \mathbf{W}\vec{f}$ can reduce dimensionality from MN to J by representing the image by J coefficients
- **Idea:** Tailor a KLT to the specific set of images of the recognition task to preserve the salient features

Eigenimages for Recognition



[Ruiz-del-Solar, Navarrete, 2005]

Computing Eigenimages from a Training Set

- How to measure the MNxMN covariance matrix?
  - Use training set $\vec{\Gamma}_1, \vec{\Gamma}_2, \ldots, \vec{\Gamma}_L$ (each column vector represents one image)
  - Let $\vec{\mu}$ be the mean image of all samples
  - Define training set matrix $\mathbf{S} = \left[\vec{\Gamma}_1 - \vec{\mu}, \vec{\Gamma}_2 - \vec{\mu}, \ldots, \vec{\Gamma}_L - \vec{\mu}\right]$ and calculate $$\mathbf{R} = \mathbf{S}\mathbf{S}^H = \sum_{l=1}^{L} \left(\vec{\Gamma}_l - \vec{\mu}\right)\left(\vec{\Gamma}_l - \vec{\mu}\right)^H$$

  Problem1: Training set size should be L >> MN.

  If L < MN, covariance matrix **R** is rank-deficient.

  Problem2: Finding eigenvectors of an MNxMN matrix.
- Can we find a small set of the most important eigenimages from a small training set L << MN?

Sirovich and Kirby Method

- Instead of eigenvectors of $\mathbf{SS}^H$, consider the eigenvectors of $\mathbf{S}^H\mathbf{S}$, i.e., $\mathrm{S}^H\mathrm{S}\vec{v}_i = \lambda_i\vec{v}_i$

  Multiply both sides by $\mathbf{S}$

$$\mathrm{S}\mathrm{S}^H\mathrm{S}\vec{v}_i = \lambda_i\mathrm{S}\vec{v}_i$$

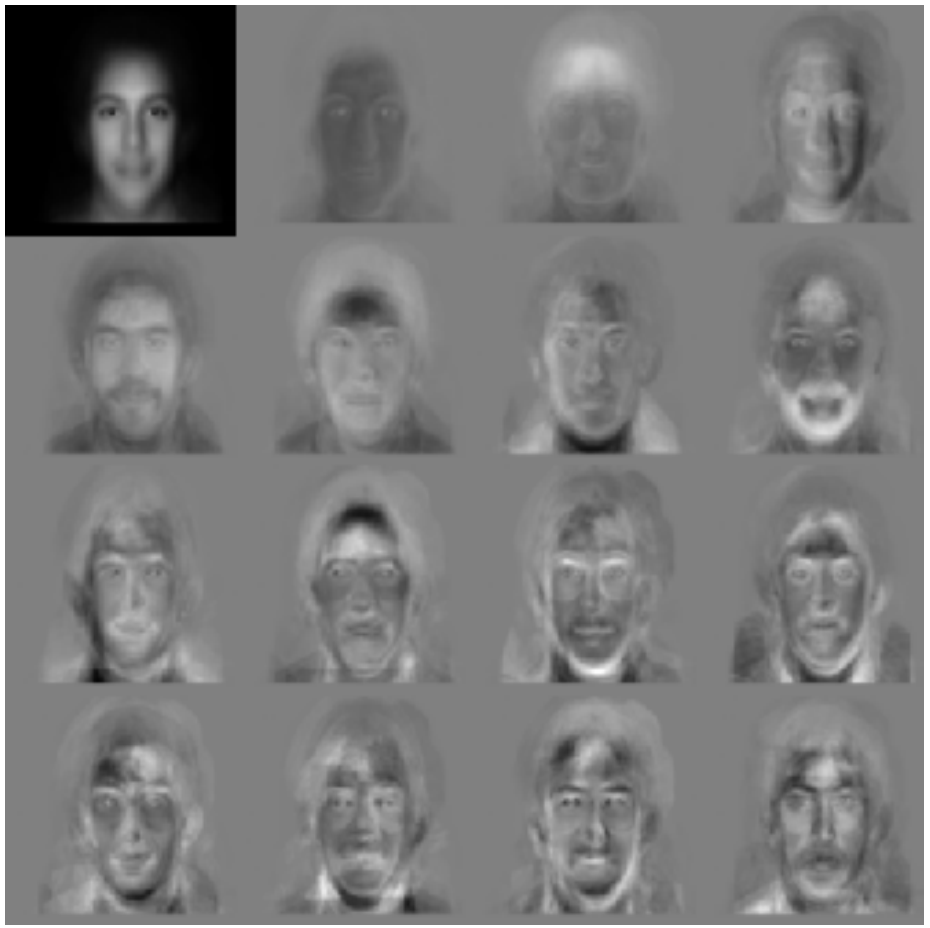
  By inspection, we find that $\mathrm{S}\vec{v}_i$ are eigenvectors of $\mathbf{SS}^H$

- For L << MN this gives rise to great computational savings
  - Computing the LxL matrix $\mathbf{S}^H\mathbf{S}$
  - Computing L eigenvectors $\vec{v}_l$ of $\mathbf{S}^H\mathbf{S}$
  - Computing eigenimages corresponding to the $L_0 \leq L$ largest eigenvalues according as $\mathrm{S}\vec{v}_i$

[Sirovich, Kirby, 1987]

Example: Eigenfaces

- The first 16 eigenfaces obtained from a training set of 128 frontal views of human faces.
- Can be used for face recognition by nearest neighbor search in 16-d "face space."
- Can be used to generate faces by adjusting 16 coefficients.



Source: MIT Media Lab