SVD and Applications

Olga Sorkine-Hornung

INTERACTIVE GEOMETRY LAB

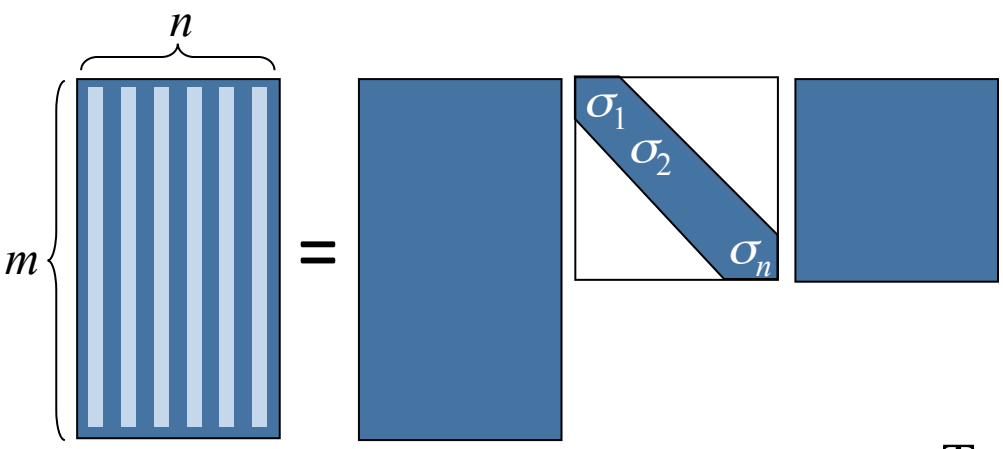
December 23, 2016

Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Dimensionality reduction

data vectors $\mathbf{a}_j$

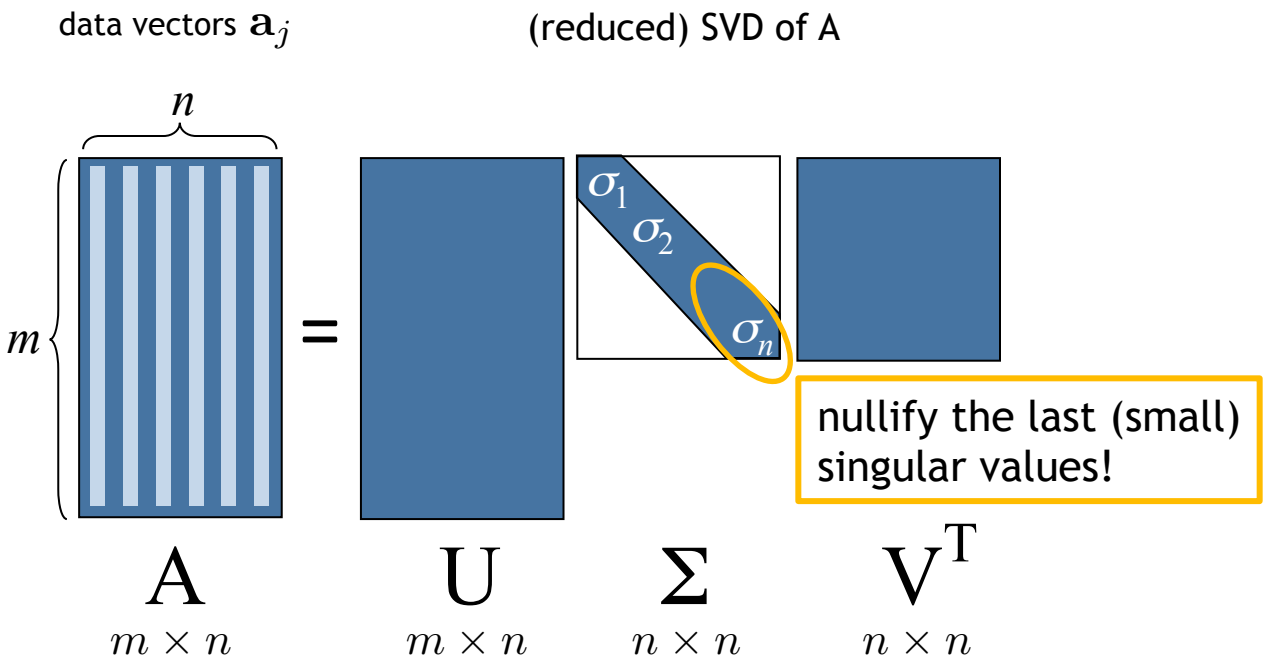
(reduced) SVD of A



$$\mathbf{A} = \mathbf{U}\,\boldsymbol{\Sigma}\,\mathbf{V}^{\mathrm{T}}$$

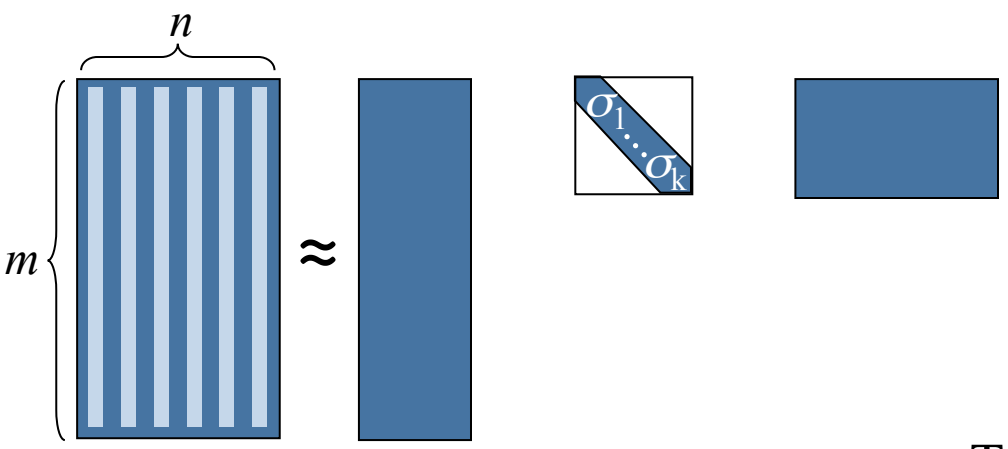
$\mathbf{A}$	$\mathbf{U}$	$\boldsymbol{\Sigma}$	$\mathbf{V}^{\mathrm{T}}$
$m \times n$	$m \times n$	$n \times n$	$n \times n$

Dimensionality reduction



Dimensionality reduction

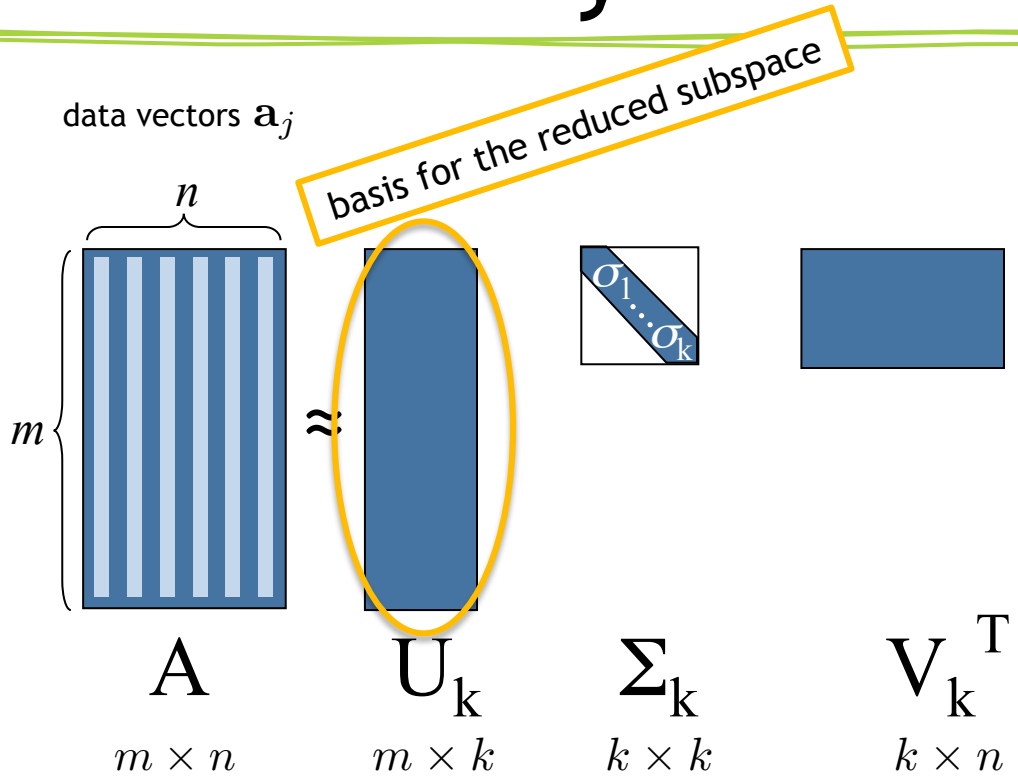
data vectors $\mathbf{a}_j$



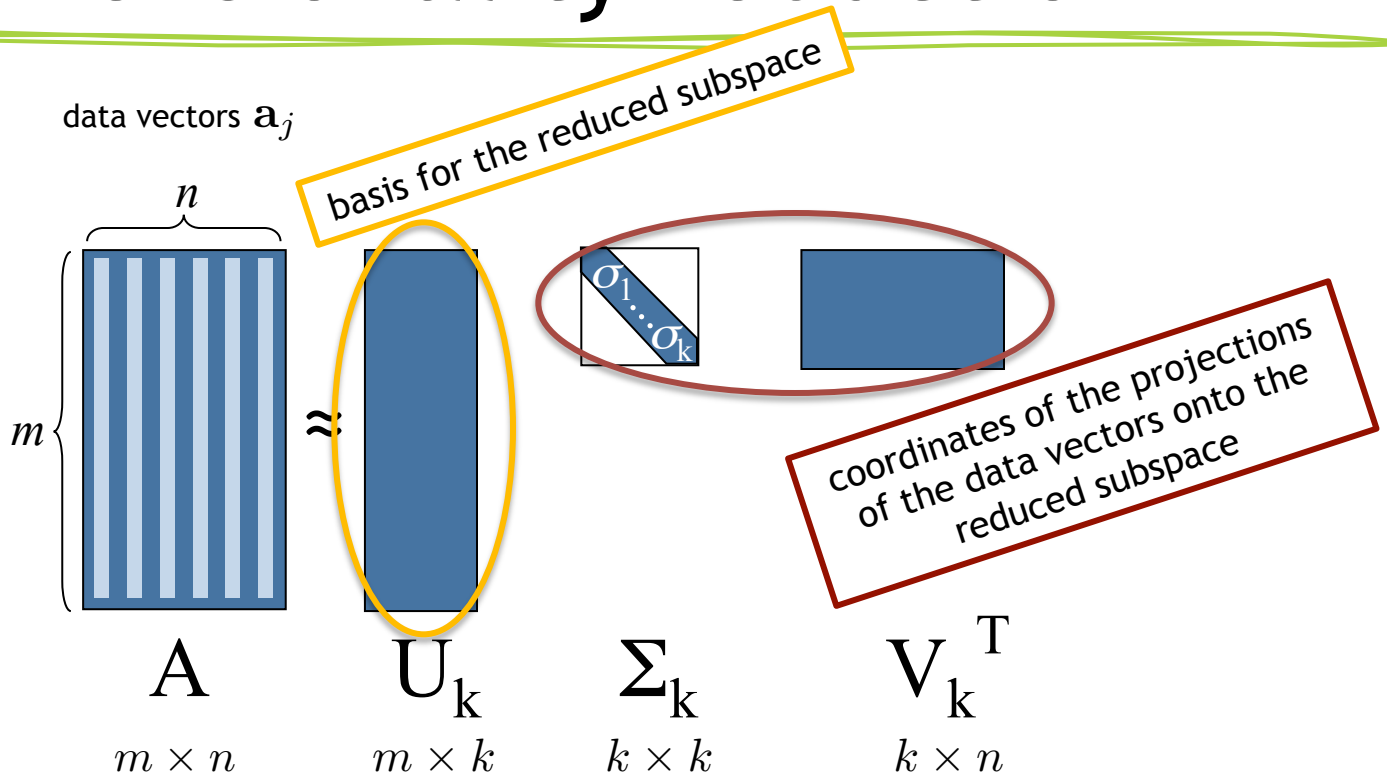
$$\mathbf{A} \approx \mathbf{U}_k \, \Sigma_k \, \mathbf{V}_k^T$$

$\mathbf{A}$	$\mathbf{U}_k$	Σ_k	$\mathbf{V}_k^T$
$m \times n$	$m \times k$	$k \times k$	$k \times n$

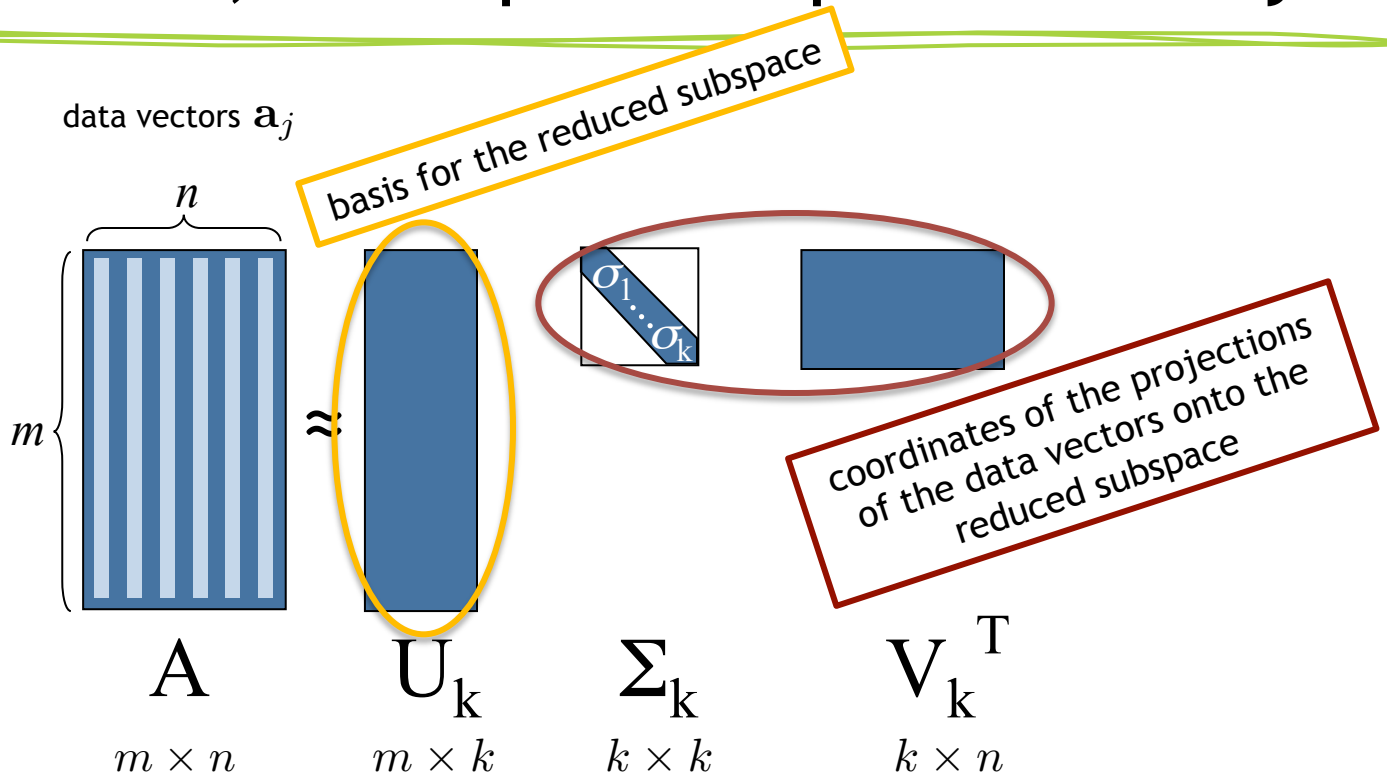
Dimensionality reduction



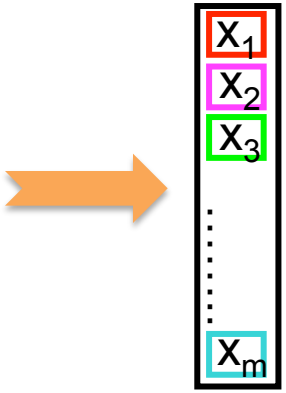
Dimensionality reduction



Also called PCA, Principal Component Analysis



The space of human faces



Concatenate grayscale pixel values into data point $\mathbf{a} \in \mathbb{R}^m$

Database of faces

- Many 100x100 pixel face images
- Different people
- Different poses
- Different illumination
- …
- Each image is labeled with the person's name/identification

- Hypothesis: Faces live in a **low-dimensional** subspace of $\mathbb{R}^{10,000}$

PCA on face database: Eigenfaces

average face

$$\mathbf{a}_{\text{avg}} = \frac{1}{n}\sum_{j=1}^{n}\mathbf{a}_j$$

PCA on face database: Eigenfaces

data vectors $\tilde{\mathbf{a}}_j := \mathbf{a}_j - \mathbf{a}_{\text{avg}}$

average face

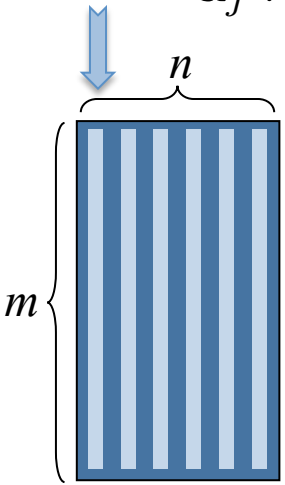
$$\mathbf{a}_{\text{avg}} = \frac{1}{n} \sum_{j=1}^{n} \mathbf{a}_j$$

PCA on face database: Eigenfaces

average face

$$\mathbf{a}_{\text{avg}} = \frac{1}{n}\sum_{j=1}^{n} \mathbf{a}_j$$

data vectors $\tilde{\mathbf{a}}_j := \mathbf{a}_j - \mathbf{a}_{\text{avg}}$



$$\mathbf{A}$$

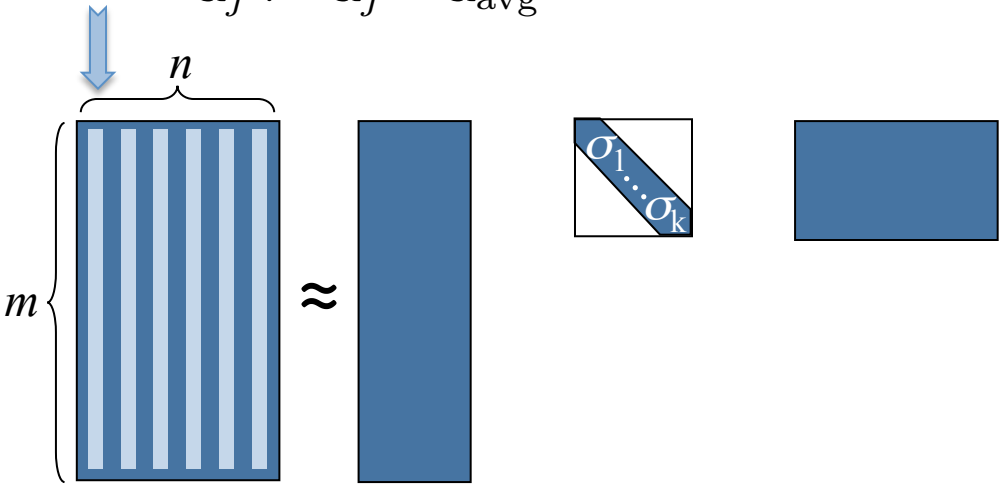
$$m \times n$$

PCA on face database: Eigenfaces

average face

$$\mathbf{a}_{\text{avg}} = \frac{1}{n}\sum_{j=1}^{n} \mathbf{a}_j$$

data vectors $\tilde{\mathbf{a}}_j := \mathbf{a}_j - \mathbf{a}_{\text{avg}}$



$$\mathrm{A} \quad \mathrm{U_k} \quad \Sigma_\mathrm{k} \quad \mathrm{V_k}^\mathrm{T}$$

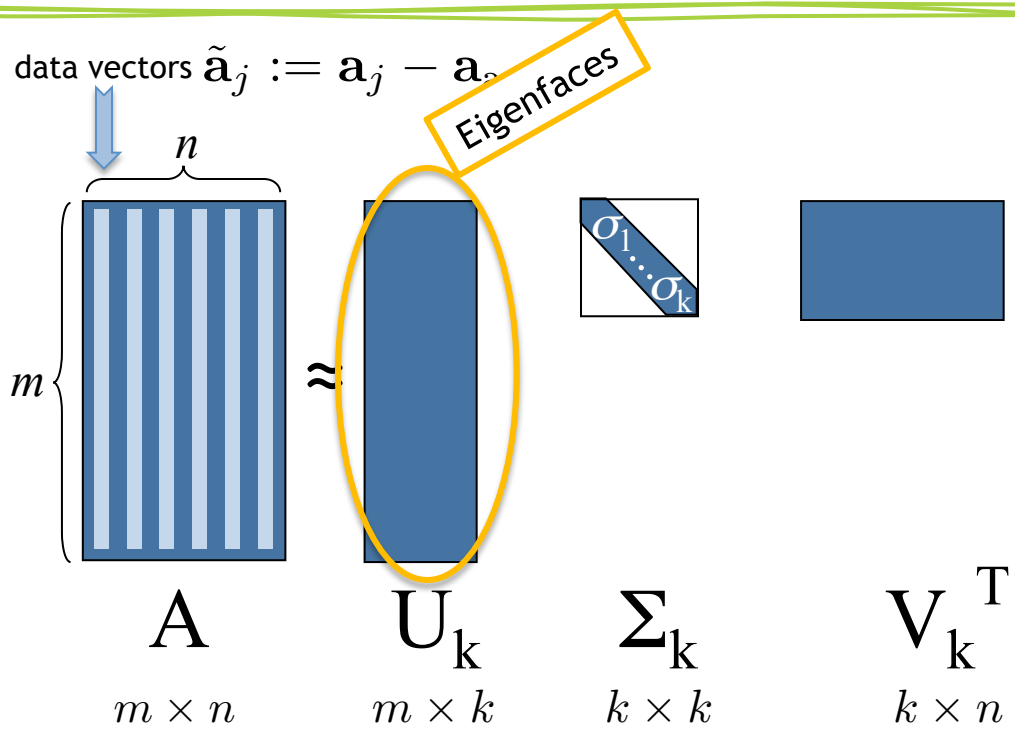
$$m \times n \quad m \times k \quad k \times k \quad k \times n$$

PCA on face database: Eigenfaces

average face

$$\mathbf{a}_{\text{avg}} = \frac{1}{n}\sum_{j=1}^{n} \mathbf{a}_j$$

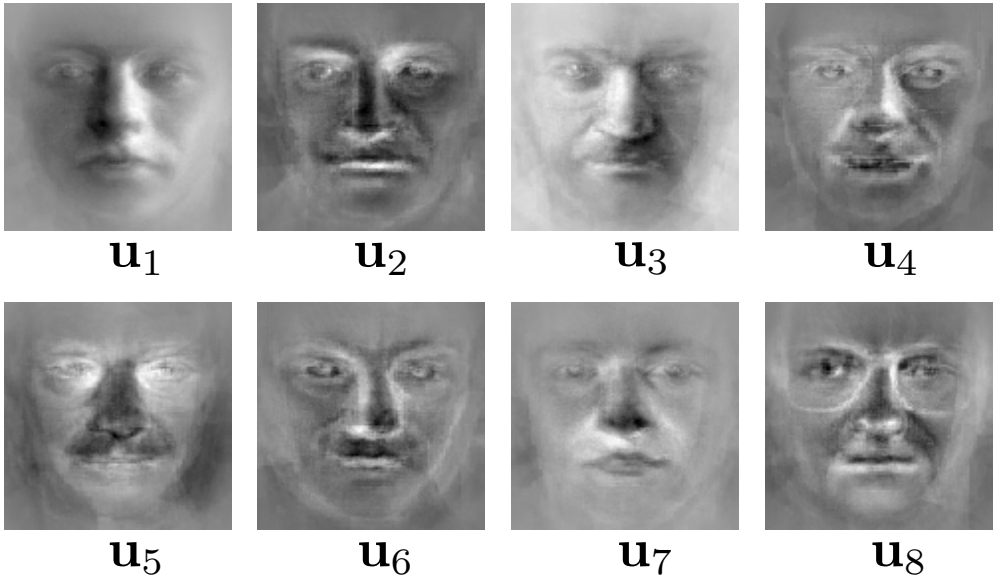
data vectors $\tilde{\mathbf{a}}_j := \mathbf{a}_j - \mathbf{a}_{\text{a}}$



PCA on face database: Eigenfaces

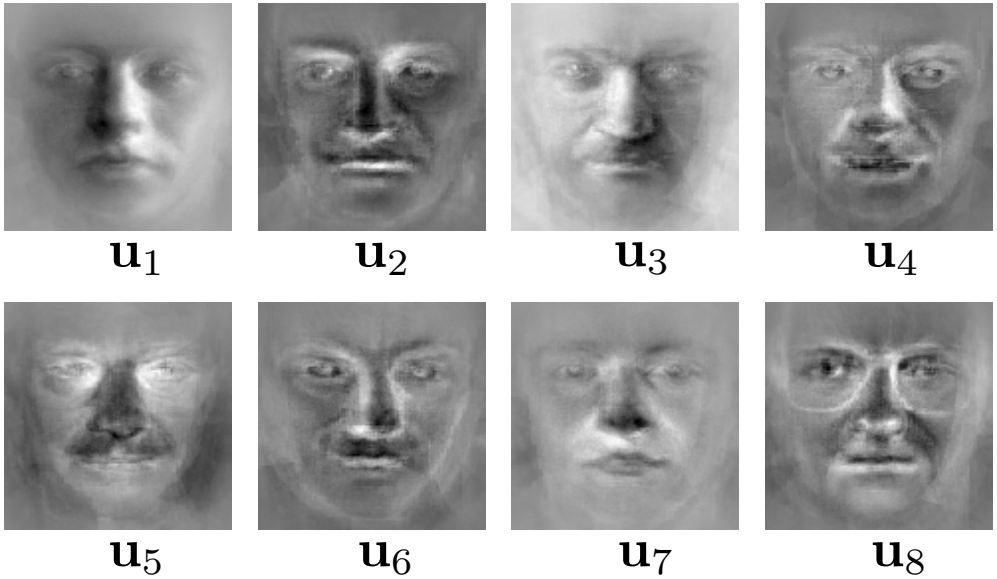
average face

$$\mathbf{a}_{\text{avg}} = \frac{1}{n}\sum_{j=1}^{n}\mathbf{a}_j$$



k=8 first principal components (columns of **U**)

PCA on face database: Eigenfaces



k=8 first principal components (columns of **U**)

Coordinates of each projection of each $\mathbf{a}_j$ onto the subspace of eigenfaces:

$$\Sigma_k V_k^T$$

$$k \times n$$

Each face is now represented by *k* numbers!

Face recognition

input image

downsample to
100x100 = m pixels,
subtract $\mathbf{a}_{avg}$

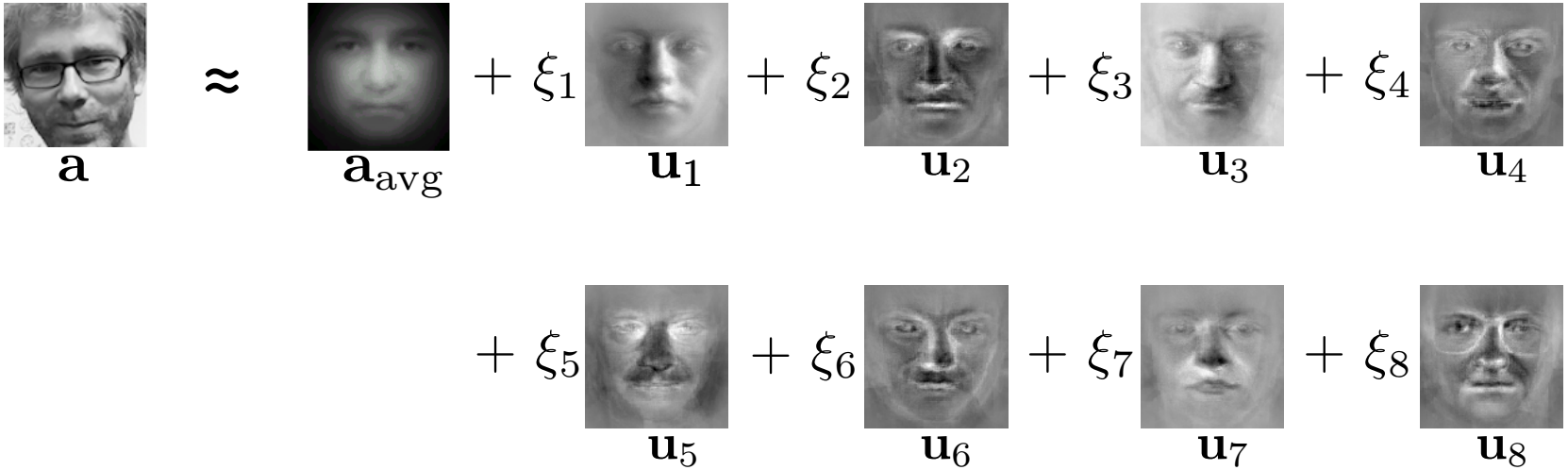
$$\tilde{\mathbf{a}}$$

$$m \times 1$$

project onto subspace,
i.e. calculate
coordinates w.r.t. $\mathbf{u}_j$

$$\xi_1 = \tilde{\mathbf{a}}^T \mathbf{u}_1$$
$$\xi_2 = \tilde{\mathbf{a}}^T \mathbf{u}_2$$
$$\vdots$$
$$\xi_8 = \tilde{\mathbf{a}}^T \mathbf{u}_8$$

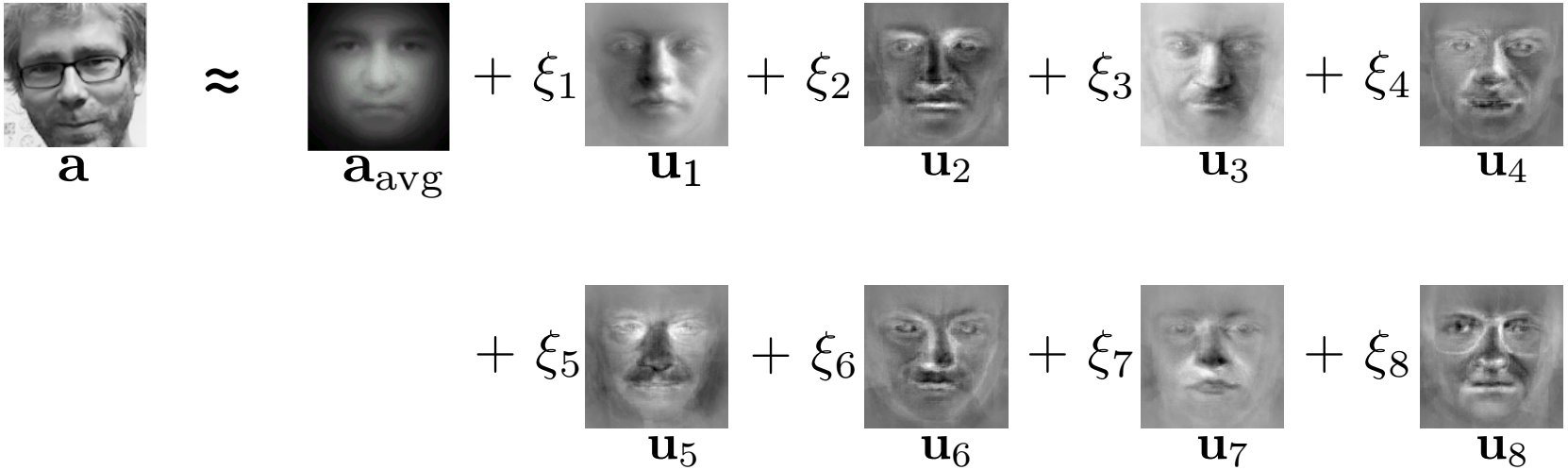
Face recognition



Compare $\xi = (\xi_1, \ldots, \xi_k)^T$ with the coordinate vectors ξ_j of $\tilde{\mathbf{a}}_j$ in the database

Find the "nearest neighbor" j^* such that $\|\xi - \xi_{j^*}\|$ is the smallest

Face recognition



Compare $\xi = (\xi_1, \ldots, \xi_k)^T$ with the coordinate vectors ξ_j of $\tilde{\mathbf{a}}_j$ in the database

Find the "nearest neighbor" j^* such that $\|\xi - \xi_{j^*}\|$ is the smallest

most likely this face in the database is labeled as "Marc Pollefeys" ☺

Face recognition

- Searching for the nearest neighbor in a k-dimensional space is very fast if k is small! And gives quite accurate results
- Searching for the most similar face in a $m=10{,}000$-dimensional space is practically infeasible
- Most $10{,}000$-dimensional vectors do not represent a human face!
- This kind of dimensionality reduction is the basis of many various machine learning methods.