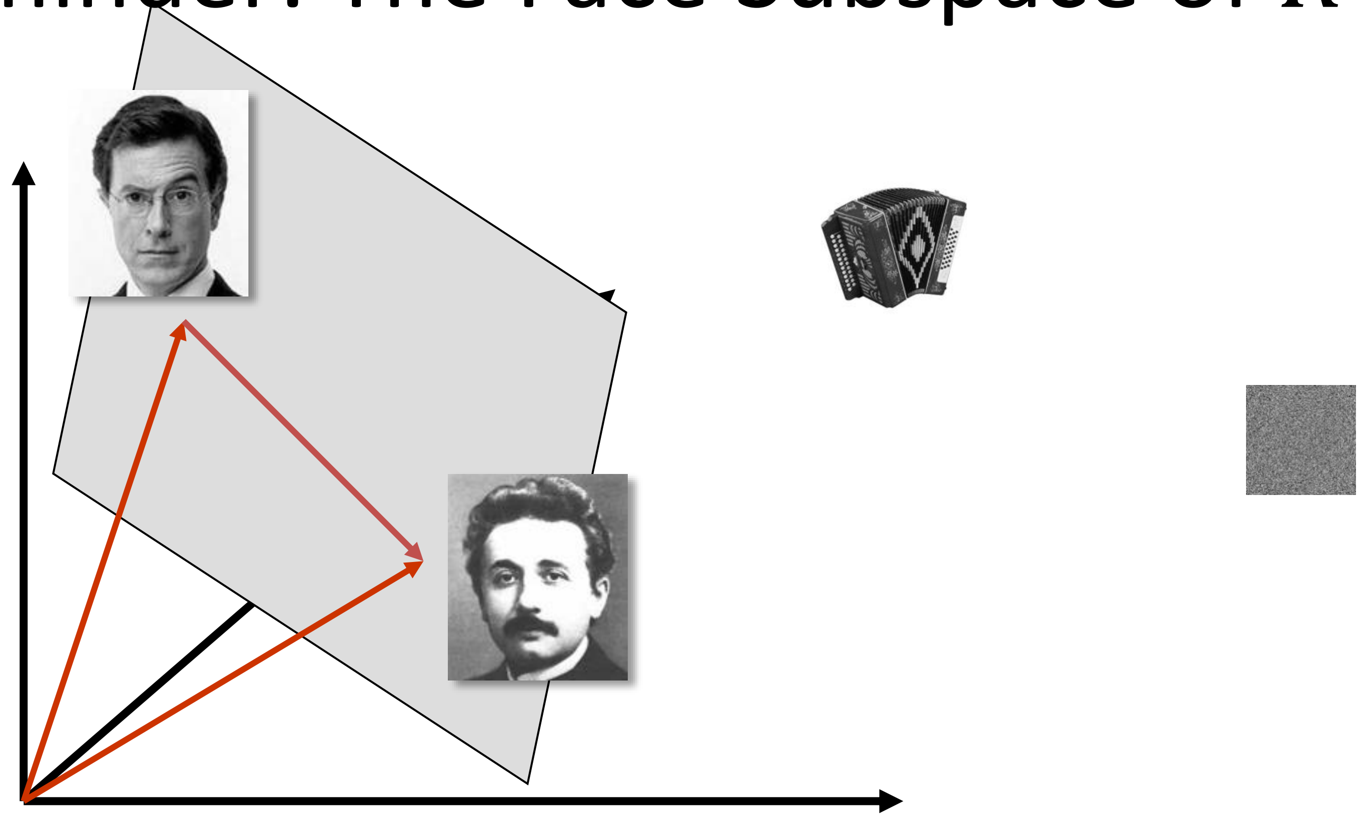
PCA: Part II

Pablo Picasso, “The Kitchen”

Many slides from
Noah Snavely, Derek Hoeim, Robert Collins

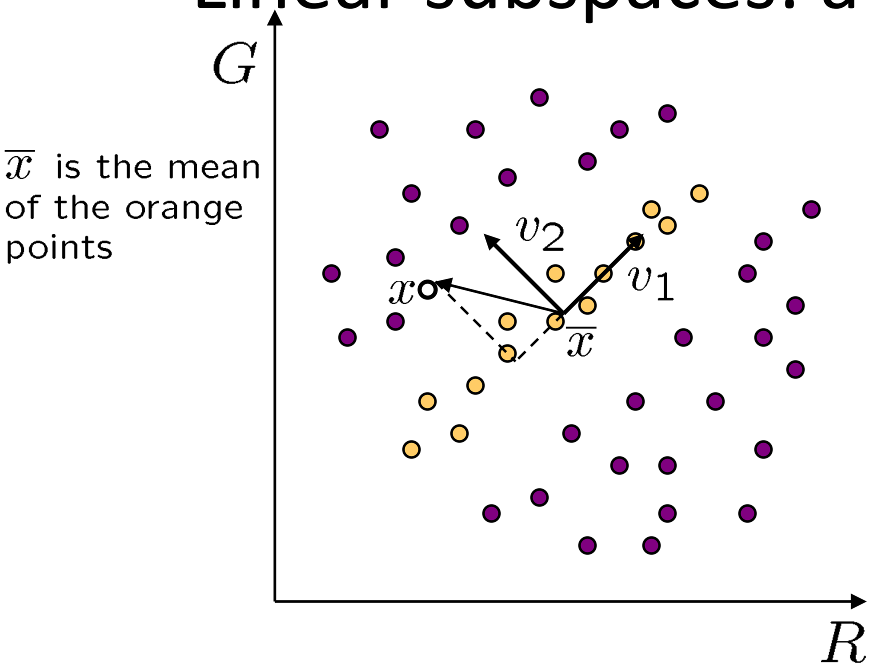
CSC320: Introduction to Visual Computing
Michael Guerzhoy

Reminder: The Face Subspace of R^n



- All the images of size *n* pixels are vectors in R^n
  - Basis: {(1,0,0,0,…)’, (0,1,0,0,0,…)’, (0,0,1,0,0,..), …}
- Hope/assumption/model: images of (centered) faces of size *n* pixels are all (approximately) lying in a k-dimensional *subspace* of R^n
  - We can find the best subspace (i.e., k vectors of size n) using PCA

Linear subspaces: a 1-D subspace of a Plane



convert $\mathbf{x}$ into $\mathbf{v_1}$, $\mathbf{v_2}$ coordinates

$$\mathbf{x} \rightarrow ((\mathbf{x} - \overline{x}) \cdot \mathbf{v_1}, (\mathbf{x} - \overline{x}) \cdot \mathbf{v_2})$$

What does the $\mathbf{v_2}$ coordinate measure?

- distance to line
- use it for classification—near 0 for orange pts

What does the $\mathbf{v_1}$ coordinate measure?

- position along line
- use it to specify which orange point it is

Aside: Why are the Faces Centered?

- Centering: if the faces are the columns of matrix X, the centered faces are $X - \bar{X},$ where $\bar{X}$ is the average column of X
  - The average column of $(X - \bar{X})$ is 0
- All linear spaces must contain 0

Representation and reconstruction

- Face **x** in "face space" coordinates:

$$\mathbf{x} \rightarrow [\mathbf{u}_1^{\mathrm{T}}(\mathbf{x} - \mu), \ldots, \mathbf{u}_k^{\mathrm{T}}(\mathbf{x} - \mu)]$$

$$= \quad w_1, \ldots, w_k$$

- Reconstruction:

 =  + 

$\hat{x} = \mu + w_1u_1+w_2u_2+w_3u_3+w_4u_4+ \ldots$

slide by Derek Hoiem

Face Detection using PCA

- For each (centered) window x and for a set of principal components V, compute the Euclidean distance $|VV^T x - x|$
- That is the distance between the reconstruction of *x* and x. The reconstruction of x is similar to x if x lies in the face subspace

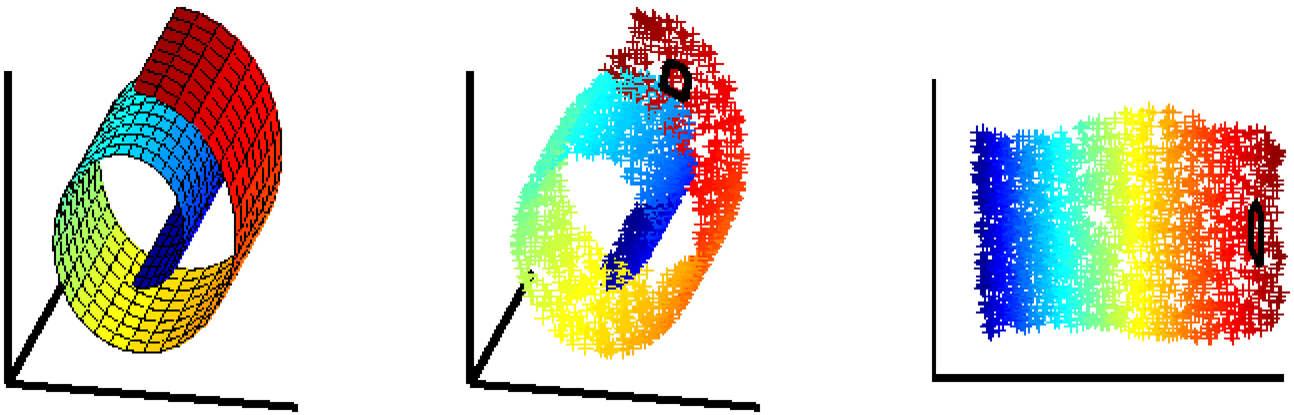
  Note: the reconstruction is *always* in the face subspace

- Win: instead of comparing *x* to a large dataset of faces, we are only comparing *x* to the columns of V
  - $V^T x$ is just a vector of the dot products $v_i \cdot x$ for every I
  - That still works, since *V* contains (we hope) all the information about the appearance of faces that there is

Issues: dimensionality

What if your space isn't *flat*?

- PCA may not help



Nonlinear methods
LLE, MDS, etc.

Moving forward

Faces are pretty well-behaved

- Mostly the same basic shape
- Lie close to a low-dimensional subspace

Not all objects are as nice

Different appearance, similar parts

Idea: Denoising images of the letter “a” with PCA

- Denoising: taking an image corrupted by some noise process, and recovering the original
  - Recall: convolving with a Gaussian filter worked pretty well for Gauassian noise, median filtering worked pretty well for salt-and-pepper noise
- Idea: take the noisy image x, and reconstruct it using PCA
  - In other words, project x onto the subspace of “a”s