Data-driven Methods: Faces


CS194: Image Manipulation & Computational Photography
Alexei Efros, UC Berkeley, Fall 2017
The Power of Averaging
8-hour exposure

© Atta Kim
Image Composites

Sir Francis Galton
1822-1911

Multiple Individuals

Composite

Average Images in Art

“60 passagers de 2e classe du metro, entre 9h et 11h” (1985)
Krzysztof Pruszkowski

“Spherical type gasholders” (2004)
Idris Khan
<table>
<thead>
<tr>
<th>Image</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td>Little Leaguer</td>
</tr>
<tr>
<td><img src="image2.png" alt="Image" /></td>
<td>Kids with Santa</td>
</tr>
<tr>
<td><img src="image3.png" alt="Image" /></td>
<td>The Graduate</td>
</tr>
<tr>
<td><img src="image4.png" alt="Image" /></td>
<td>Newlyweds</td>
</tr>
</tbody>
</table>

“100 Special Moments” by Jason Salavon

Why blurry?
Object-Centric Averages by Torralba (2001)

Manual Annotation and Alignment

Average Image

Slide by Jun-Yan Zhu
Computing Means

Two Requirements:
• Alignment of objects
• Objects must span a subspace

Useful concepts:
• Subpopulation means
• Deviations from the mean
Images as Vectors

\[ \text{Images as Vectors} \]
Vector Mean: Importance of Alignment

\[ n \times m = \frac{1}{2} + \frac{1}{2} = \text{mean image} \]
How to align faces?

http://www2.imm.dtu.dk/~aam/datasets/datasets.html
Shape Vector

Provides alignment!

43
Appearance Vectors vs. Shape Vectors

**Appearance Vector**
- 200*150 pixels (RGB)
- Vector of 200*150*3 Dimensions
- Requires Annotation
- Provides alignment!

**Shape Vector**
- 43 coordinates (x,y)
- Vector of 43*2 Dimensions

Slide by Kevin Karsch
Average Face

1. Warp to mean shape
2. Average pixels

Objects must span a subspace
Example

Does not span a subspace
Subpopulation means

Examples:
- Male vs. female
- Happy vs. said
- Average Kids
- Happy Males
- Etc.
- http://www.faceresearch.org

Average male
Average female
Average kid
Average happy male
Average male
Average Women of the world
Average Men of the world
Deviations from the mean

\[ \Delta X = X - \bar{X} \]
Deviations from the mean

\[ \Delta X = X - \bar{X} \]

\[ X = \bar{X} + 1.7 \]
Extrapolating faces

- We can imagine various meaningful directions.
Manipulating faces

• How can we make a face look more female/male, young/old, happy/sad, etc.?

• http://www.facerresearch.org/demos/transform
Manipulating Facial Appearance through Shape and Color

Duncan A. Rowland and David I. Perrett

St Andrews University

IEEE CG&A, September 1995
Face Modeling

Compute *average* faces
(color and shape)

Compute *deviations*
between male and female (vector and color differences)
Changing gender

Deform shape and/or color of an input face in the direction of “more female”

original

shape

color

both
Enhancing gender

more same original androgynous more opposite
Changing age

Face becomes “rounder” and “more textured” and “grayer”

original                  shape

original

color

both
Back to the Subspace
Linear Subspace: convex combinations

Any new image $X$ can be obtained as weighted sum of stored “basis” images.

$$X = \sum_{i=1}^{m} a_i X_i$$

Our old friend, change of basis! What are the new coordinates of $X$?
The Morphable Face Model

The actual structure of a face is captured in the shape vector $S = (x_1, y_1, x_2, ..., y_n)^T$, containing the $(x, y)$ coordinates of the $n$ vertices of a face, and the appearance (texture) vector $T = (R_1, G_1, B_1, R_2, ..., G_n, B_n)^T$, containing the color values of the mean-warped face image.
The Morphable face model

Again, assuming that we have \( m \) such vector pairs in full correspondence, we can form new shapes \( S_{\text{model}} \) and new appearances \( T_{\text{model}} \) as:

\[
S_{\text{model}} = \sum_{i=1}^{m} a_i S_i \quad \quad T_{\text{model}} = \sum_{i=1}^{m} b_i T_i
\]

\[
s = \alpha_1 \cdot \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_T \end{pmatrix} + \alpha_2 \cdot \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_T \end{pmatrix} + \alpha_3 \cdot \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_T \end{pmatrix} + \alpha_4 \cdot \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_T \end{pmatrix} + \ldots = S \cdot a
\]

\[
t = \beta_1 \cdot \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_T \end{pmatrix} + \beta_2 \cdot \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_T \end{pmatrix} + \beta_3 \cdot \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_T \end{pmatrix} + \beta_4 \cdot \begin{pmatrix} t_1 \\ t_2 \\ \vdots \\ t_T \end{pmatrix} + \ldots = T \cdot \beta
\]

If number of basis faces \( m \) is large enough to span the face subspace then:

**Any new face can be represented as a pair of vectors**

\((\alpha_1, \alpha_2, \ldots, \alpha_m)^T\) and \((\beta_1, \beta_2, \ldots, \beta_m)^T\)!
Issues:

1. How many basis images is enough?
2. Which ones should they be?
3. What if some variations are more important than others?
   • E.g. corners of mouth carry much more information than haircut

Need a way to obtain basis images automatically, in order of importance!

But what’s important?
Principal Component Analysis

Given a point set \( \{\mathbf{p}_j\}_{j=1}^P \), in an \( M \)-dim space, PCA finds a basis such that

- coefficients of the point set in that basis are uncorrelated
- first \( r < M \) basis vectors provide an approximate basis that minimizes the mean-squared-error (MSE) in the approximation (over all bases with dimension \( r \))
PCA via Singular Value Decomposition

\[ [u,s,v] = \text{svd}(A); \]

EigenFaces

First popular use of PCA on images was for modeling and recognition of faces [Kirby and Sirovich, 1990, Turk and Pentland, 1991]

- Collect a face ensemble
- Normalize for contrast, scale, & orientation.
- Remove backgrounds
- Apply PCA & choose the first $N$ eigen-images that account for most of the variance of the data.
First 3 Shape Basis

Mean appearance

Principal Component Analysis

Choosing subspace dimension $r$:

- look at decay of the eigenvalues as a function of $r$
- Larger $r$ means lower expected error in the subspace data approximation
Using 3D Geometry: Blinz & Vetter, 1999

http://www.youtube.com/watch?v=jrutZaYoQJo