Robust Face Recognition Measuring 2D Deformations

Anne Jorstad
Applied Mathematics & Statistics, and Scientific Computation
Candidacy Talk
May 8, 2009

Outline

- Face Recognition
  - Background
  - Current Deformable Models
    - Feature-based methods
    - Optical Flow
    - Methods using dense correspondences
- A more robust measure of deformation
  - Long range dense correspondences
  - Statistical models of these correspondences and resulting deformations

Problem Statement

- Given a single unknown 2D image of a face, determine the most similar face from a database of known faces.

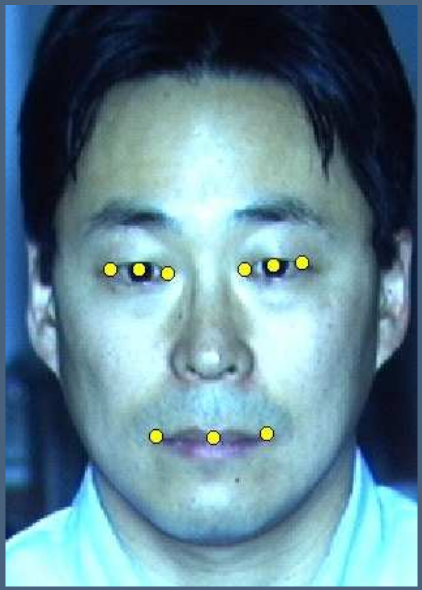
Images from: A.M. Martinez, R. Benavente. The AR Face Database. CVC Technical Report #24, June 1998.

Background: Image Alignment

- State of the art can reliably detect a small number of corresponding face feature points

- Align images
  - Rotation, translation, scaling
  - Minimize sum of squared distances between individual face point locations and average face point locations

- Image alignment is assumed preprocessing for all methods to follow



OMRON face points

Background: Image Differences

Sum of pixel differences

320,423

less different

731,101

Background: Image Differences

Sum of pixel differences

676,316

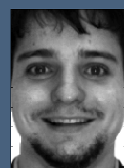
less different?

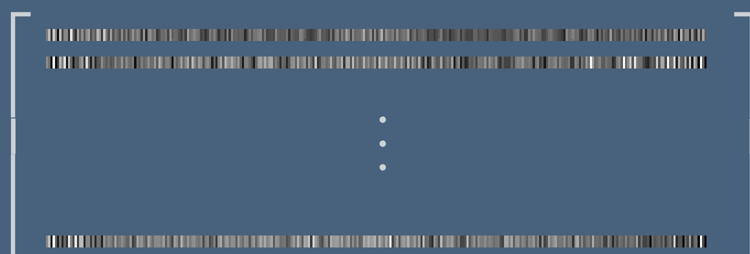
635,435

Face Representation Algorithms

- First attempts
  - Methods that handle images directly
- Majority of talk
  - Methods that deform input images
  - Measure constructed images and deformations

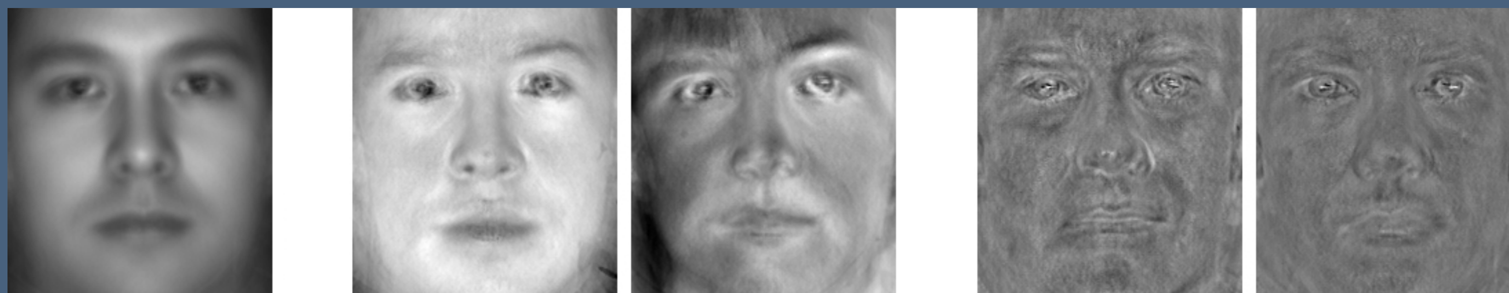
Face Representation Algorithms: First Attempts

- Principal Component Analysis (PCA):
  - Find low dimensional linear subspace that captures the most important variations in the dataset
    - I:  $\longrightarrow$ 

      Normalize: $\mathrm{mean}(I) = 0$, $\mathrm{var}(I) = 1$
    -  $= USV^T$
  - First k principal component vectors of V are the “eigenfaces” of the dataset. Linear combinations provide approximations to true images.

Face Representation Algorithms: First Attempts

- Some eigenfaces:



average face | first two eigenfaces | last two eigenfaces

- A face projected into its eigenbasis:

Images from: http://www.scholarpedia.org/article/Eigenfaces

Face Representation Algorithms: First Attempts

- Linear Discriminant Analysis (LDA):
  - Instead of finding the best subspace representation, find the best classification:
    - Maximize difference between classes

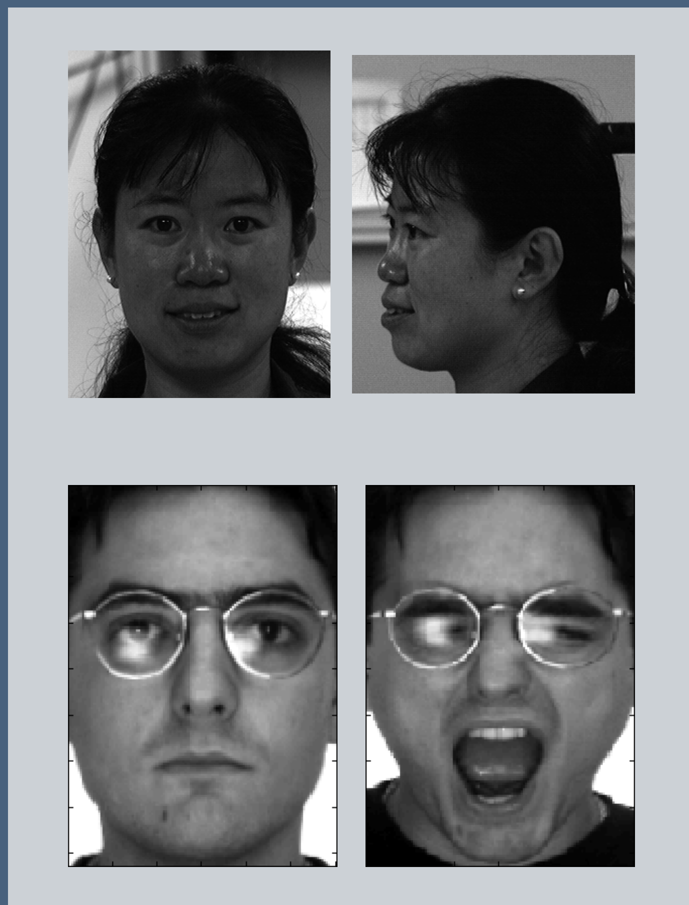
      S_B: between-class covariance matrix
    - Minimize difference within each class

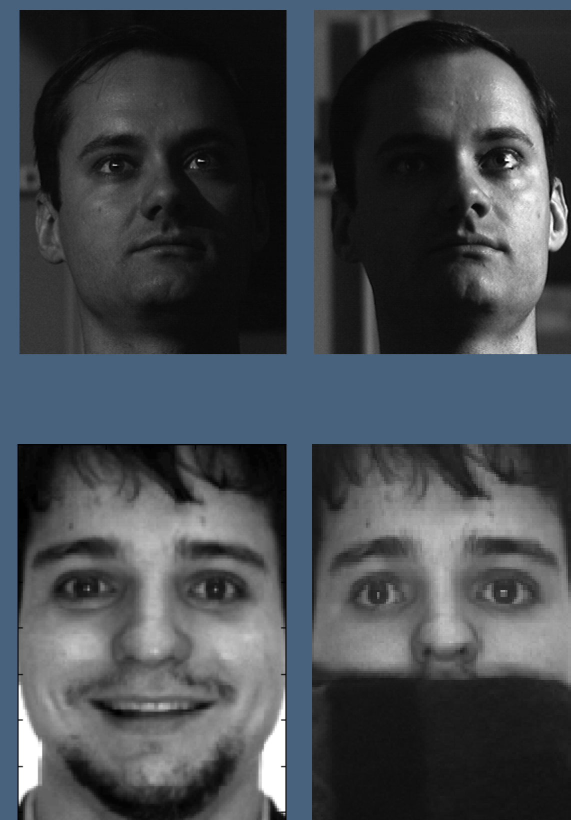
      S_W: within-class covariance matrix
    - $I_{LDA} = \omega^T I$, pick projection ω to maximize $\frac{\omega^T S_B \omega}{\omega^T S_W \omega}$

P. Belhumeur, J. Hespanha, D. Kriegman. "Eigenfaces cs. Fisherfaces: Recognition Using Class Specific Linear Projection." IEEE Transactions on Pattern Analysis and Machine Intelligence, 1997.

Problems with Pixels

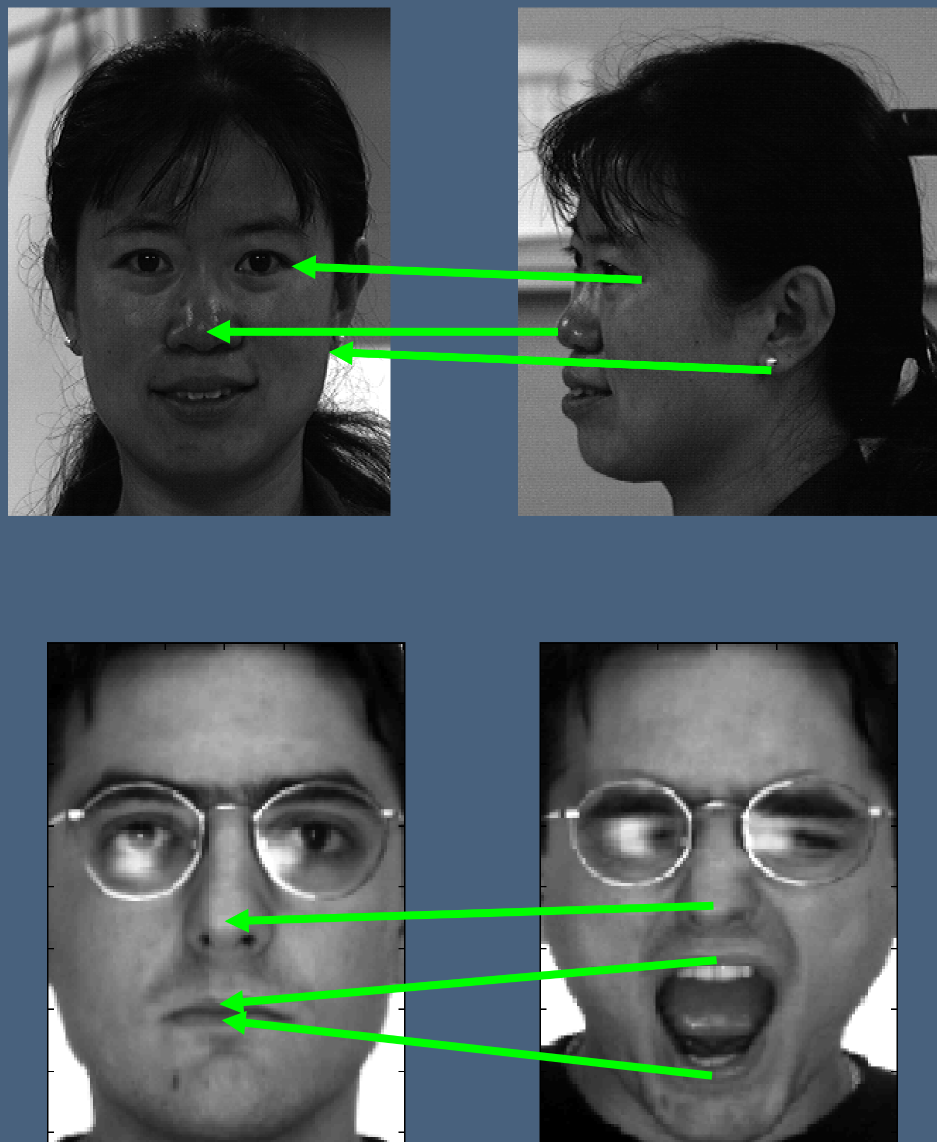
- Pixel-based methods fail when variations in *pose*, *expression*, *lighting* and *occlusions* are introduced.





- Want to warp input face to standard expression and pose before calculating the image difference.

Finding Correspondences

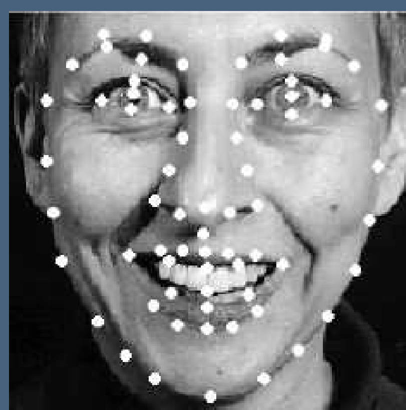


How to determine correspondences?

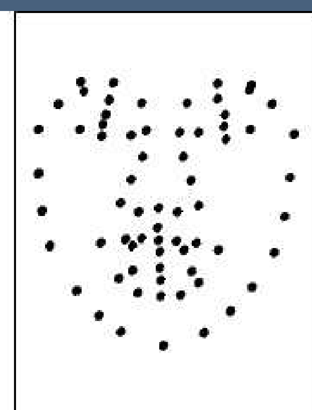
What to do with them once they are found?

Active Appearance Models

- Separate Shape (location) information from Texture (intensity) information:
  - Indentify corresponding feature points in each image
  - Warp points to average locations, interpolate all other points
  - Map texture values respectively for “shape-free patch”



original labeled image



average point locations

shape-free image

T. Cootes, G. Edwards, C. Taylor. “Active Appearance Models.” IEEE Transactions on Pattern Analysis and Machine Intelligence, 2001.

Active Appearance Models

- An individual image has shape vector x and texture vector t, where:

$$x = \bar{x} + Q_s c$$

$$t = \bar{t} + Q_t c$$

Q_s: modes of shape variation
(PCA over point locations)

Q_t: modes of texture variation
(PCA over warped images)

c: image-specific parameter values

Active Appearance Models

- Iterate model to generate good match to input image
  - Residual error at iteration m: $r(c_m) = t_0 - t_m$

    t_0 = input image texture

    t_m = current warped model texture

  - Update parameters: $c_{m+1} = c_m + \delta c_m$

    where δc_m is chosen to minimize $\|r(c_m + \delta c_m)\|^2$

    using the first order Taylor expansion:

$$r(c_m + \delta c_m) = r(c_m) + \frac{\partial r}{\partial c}\delta c$$

estimated from training data

Automatic Correspondences

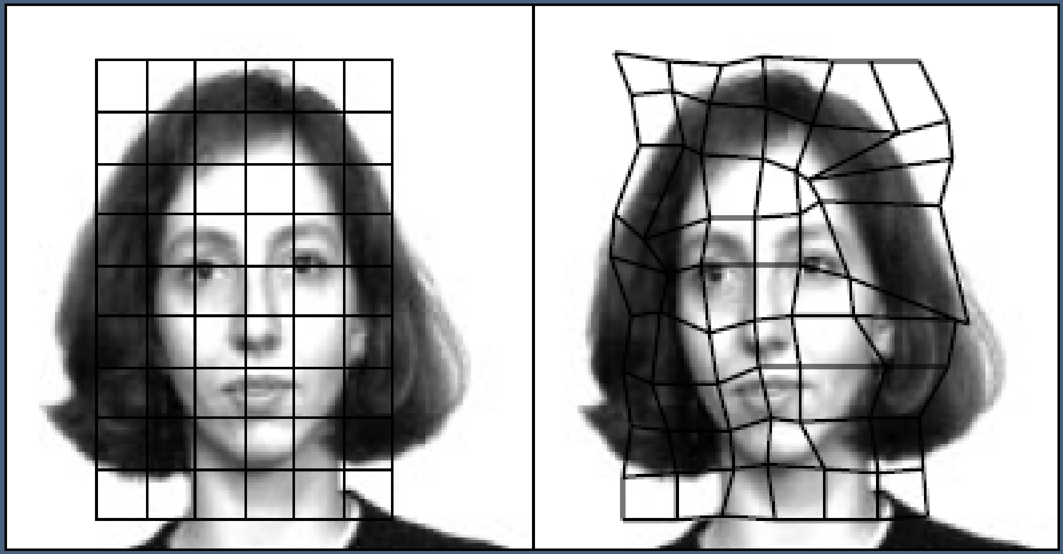
- Unreasonable to expect large number of feature point correspondences

- State of the art can reliably detect a small number of face feature points
  - Useful for image alignment
  - Insufficient for warping

- Would like to automatically obtain correspondences

Dynamic Link Matching

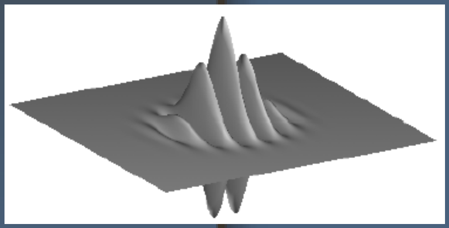
- To find correspondences for comparison
- Fit a uniform grid of nodes over a face, adjusting each node locally to best fit a model.



M. Lades, J. Vorbruggen, J. Buhmann, J. Lange, C. v.d. Malsburg, R. Wurtz, W. Konen. "Distortion Invariant Object Recognition in the Dynamic Link Architecture." IEEE Transactions on Computers, 1993.

Dynamic Link Matching

- Each node = “jet”, a vector:
  - Gabor wavelet convolution with the image



    - Gabor wavelets are a “good approximation to the sensitivity profiles of neurons found in the visual cortex” of the brain
  - 5 scales

- Fit new image jet J^I with model jet J^M:

$$\max \; C_v(J^I, J^M) = \frac{\langle J^I, J^M \rangle}{\|J^I\| \|J^M\|}$$

Dynamic Link Matching

- Also want to minimize the image distortion
  - Distance between nodes:

$$\Delta_{ij} = \vec{x}_j - \vec{x}_i$$

  - Overall distortion:

$$\min \; C_e(\Delta_{ij}^I, \Delta_{ij}^M) = \left(\Delta_{ij}^I - \Delta_{ij}^M\right)^2$$

Dynamic Link Matching

- Total cost to be minimized:

$$C(x_i^I) = \lambda \sum_{(i,j)\in E} C_e(\Delta_{ij}^I, \Delta_{ij}^M) - \sum_{i\in V} C_v(J^I(x_i^I), J_i^M)$$

distortion penalty constant

minimize distortions

maximize node match similarity

- Optimize via simulated annealing
  - Randomly shift the nodes

Pictorial Structures

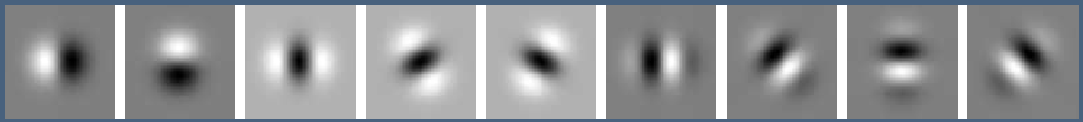
- Learn cost function for deformations specific to faces, depends on:
  - Local image similarity
  - Amount of deformation required to arrive at this similarity

- Consider connections between few higher level "parts"
  - Unlike other algorithms, this method is only for face *detection*

P. Felzenszwalb, D. Huttenlocher. "Pictorial Structures for Object Recognition." International Journal of Computer Vision, 2005.

Pictorial Structures

- "Part"
  - 27-D vector
  - Gaussian derivative filters



  - Varies order, orientation and scale

- Learn what parts look like from labeled training examples

Pictorial Structures

- Best match of new image to model:

$$L^* = \operatorname*{argmin}_{L} \left(\sum_{i \in V} m_i(\ell_i) + \sum_{(i,j) \in E} d_{ij}(\ell_i, \ell_j) \right)$$

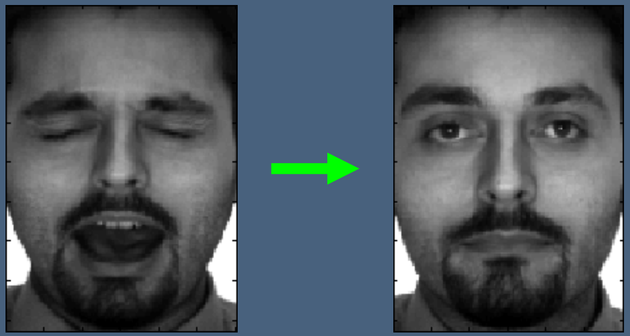
mismatch to model when part v_i is placed at location ℓ_i

deformation of the model between parts v_i and v_j (Mahalanobis correlation distance)

- Method detects faces
  - Not discriminative enough for identification

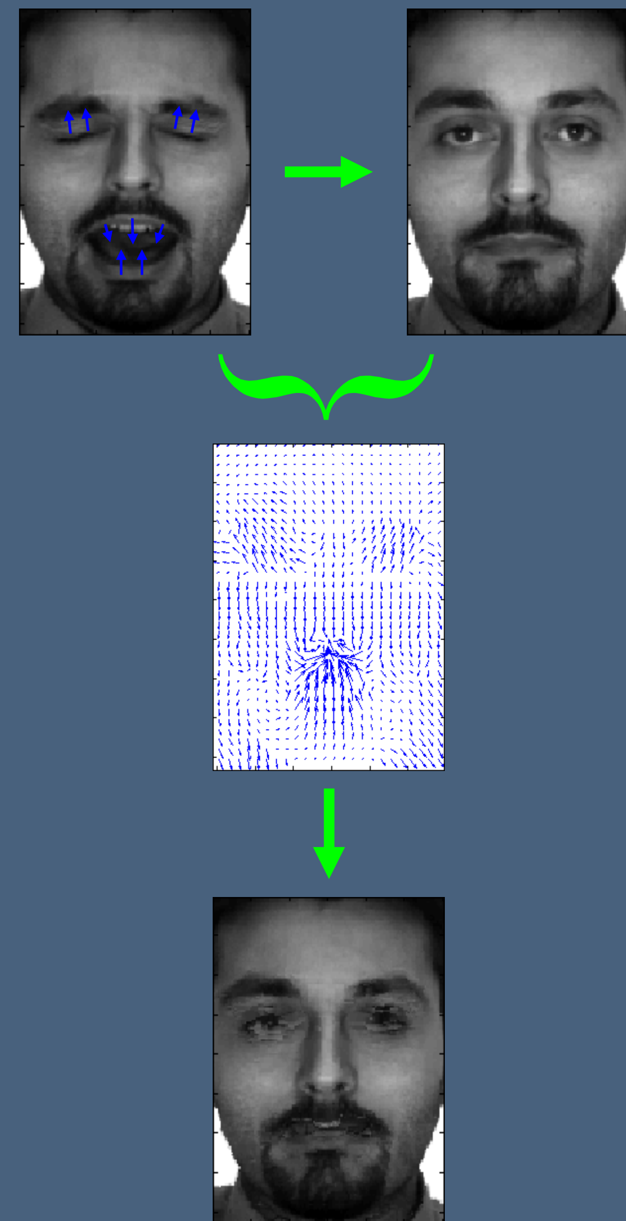
Dense Correspondences

- Match every point in new face to some point in known face



Dense Correspondences

- Match every point in new face to some point in known face.
- Optical flow
  - Determine the displacement of every pixel in the first image to the most similar pixel in the second
  - Return $[u, v]$ vector for each point
    - Vector field over the image
  - Assume images are similar
  - Assume intensity is preserved between corresponding patches



Optical Flow

- Intensity constraint equation:

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t)$$

- Taylor series:

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t$$

$$0 = \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \frac{\partial I}{\partial t}.$$

Optical flow values to be returned

$\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}$ calculated using finite differences of pixels

- Optical Flow equation: $\nabla I \cdot \vec{v} + I_t = 0$

Let $E_b = \nabla I \cdot \vec{v} + I_t$

- Need second constraint to explicitly solve for u, v

Optical Flow

- Horn and Schunk
  - Enforce smoothness by minimizing gradient of flow:

$$E_c^2 = \left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2$$

  - To solve:

$$\min \int\int \left(E_b^2 + \lambda E_c^2\right) dxdy$$

B. Horn, B. Schunck. "Determining Optical Flow." Artificial Intelligence, 1981.

Optical Flow

- Problems at motion boundaries

First two frames in video sequence

Least squares estimate of horizontal flow
(Horn and Schunk)

Robust gradient estimate of horizontal flow
(Black and Anandan)

Optical Flow

- Black and Anandan
  - Robust Statistics
    - Exclude outliers to handle object boundaries
    - Incorporate robust ρ-function (error) and its derivative ψ (proportional to the influence function)

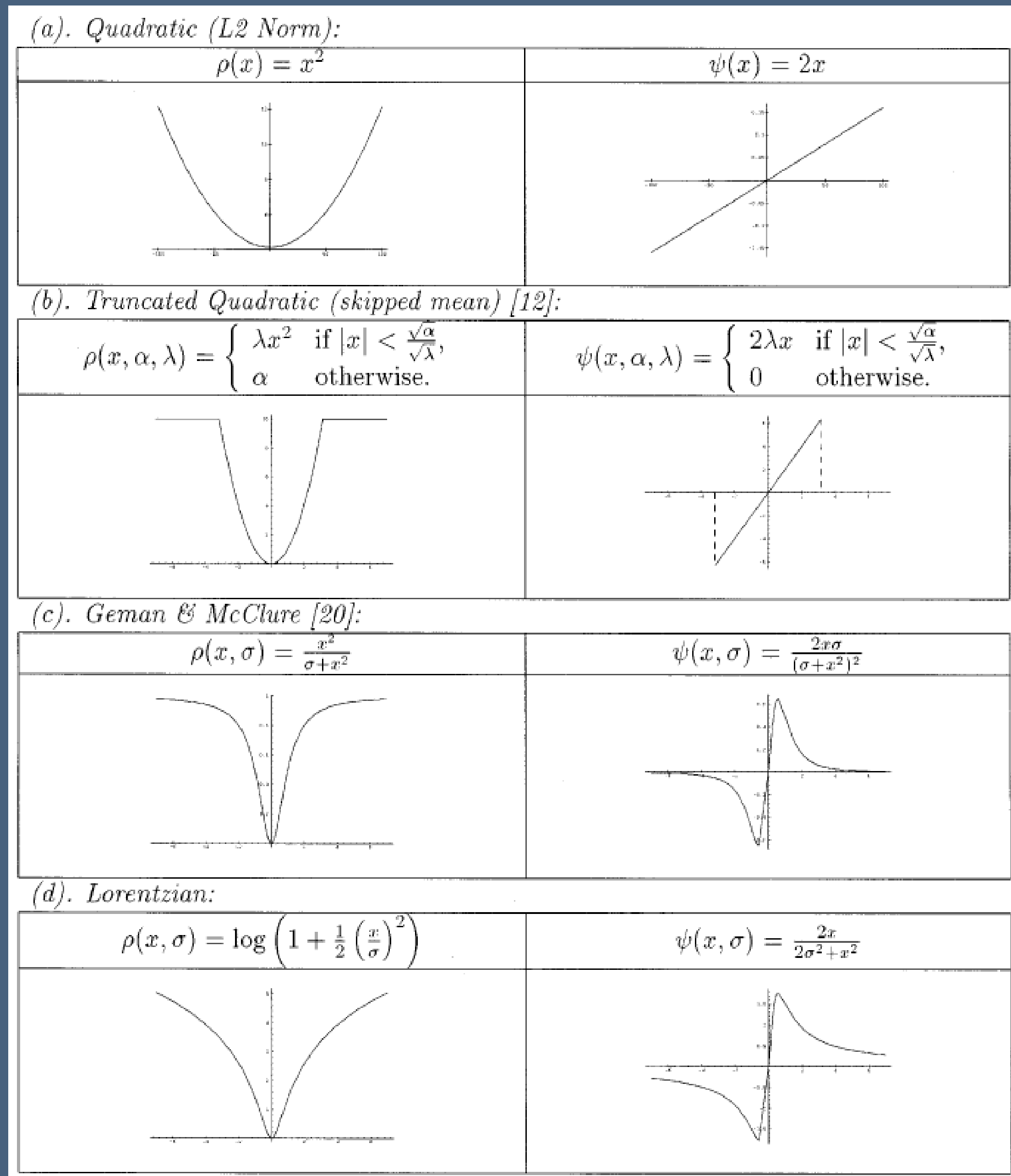
$$\min \int\int \left(\rho_b(E_b^2) + \lambda\rho_c(E_c^2)\right) dxdy$$

function to limit influence of outliers

M. Black, P. Anandan. "The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields." *Computer Vision and Image Understanding*, 1996.

Optical Flow

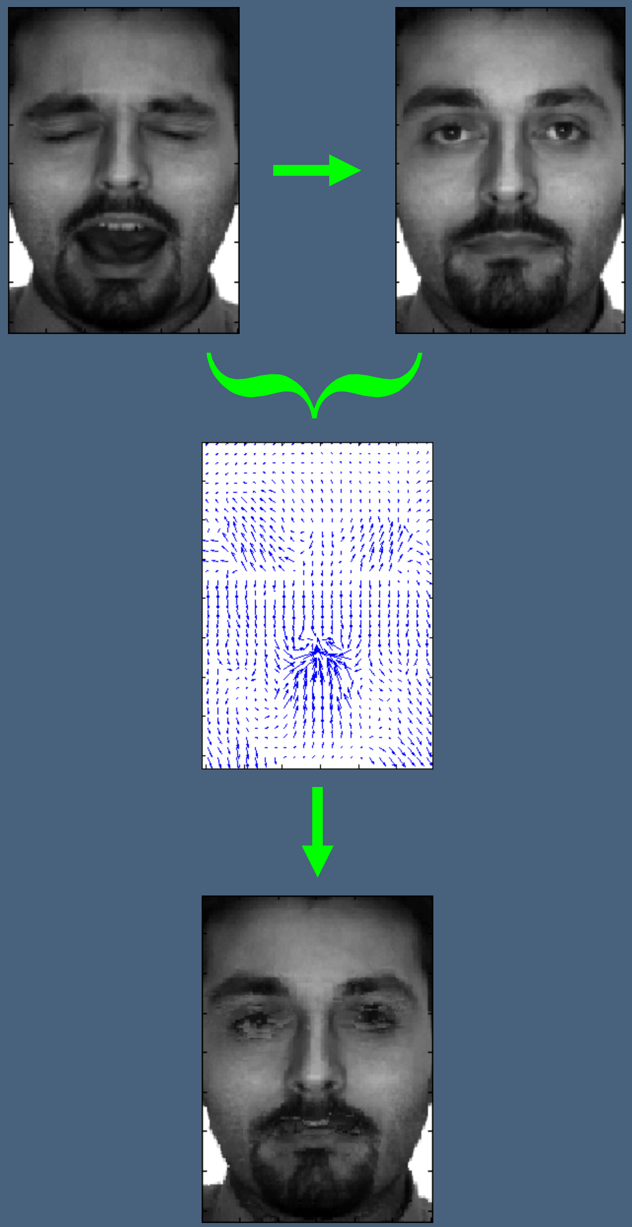
ρ-function (error)



ψ-function = derivative of ρ (proportional to the influence function)

Methods Using Dense Correspondences

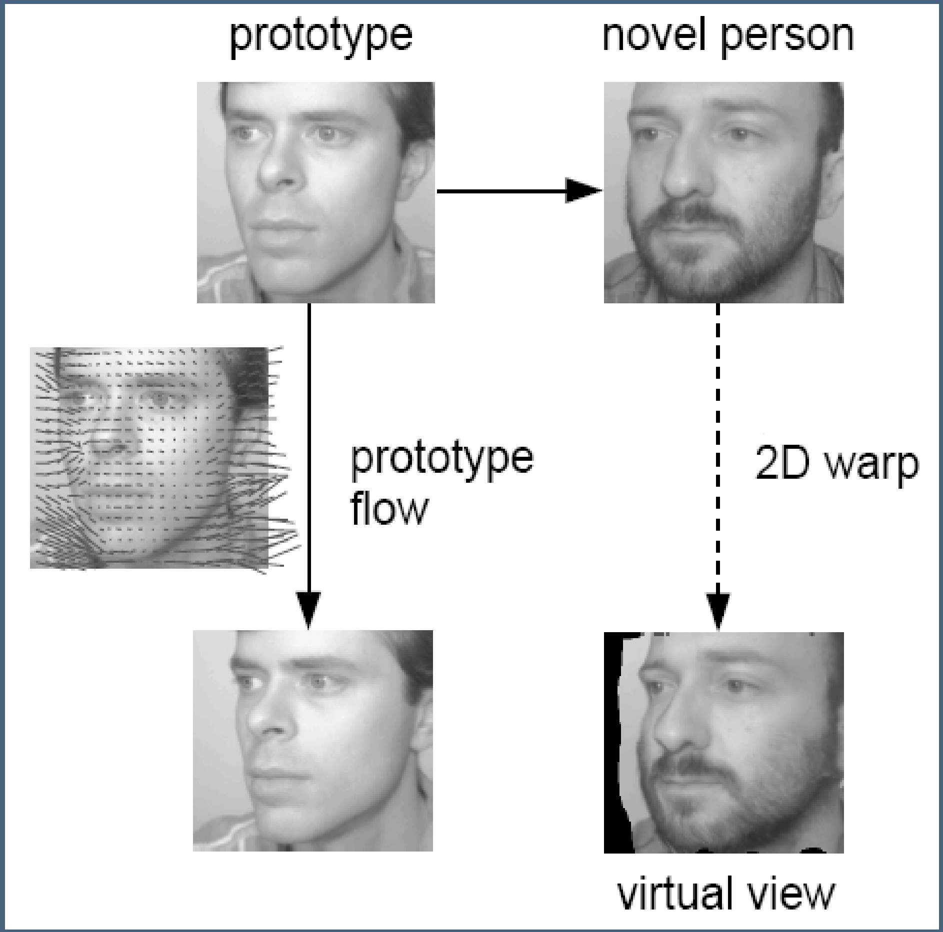
- Use optical flow to obtain corresponding pixel for every point in an image.



Warping A Single Image

- Use prior knowledge of face pose change to warp a single known image to a new artificial image.

Known images ⟶

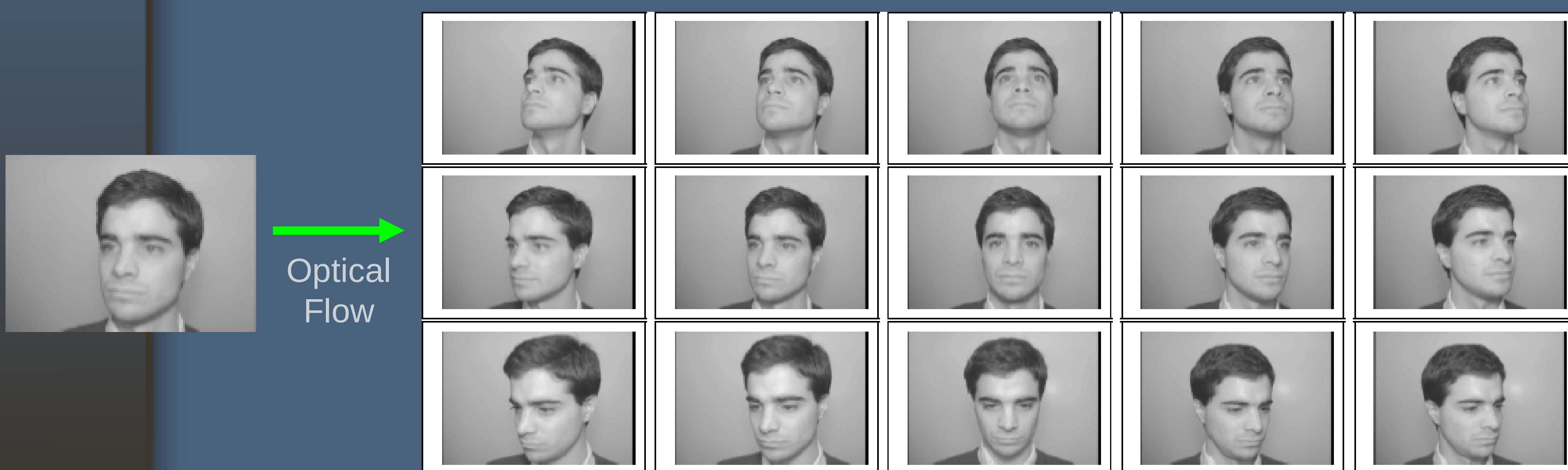


Apply same transformation to generate virtual image

D. Beymer, T. Poggio. *"Face Recognition From One Example View." Proceedings of the Fifth International Conference on Computer Vision (ICCV), 1995.*

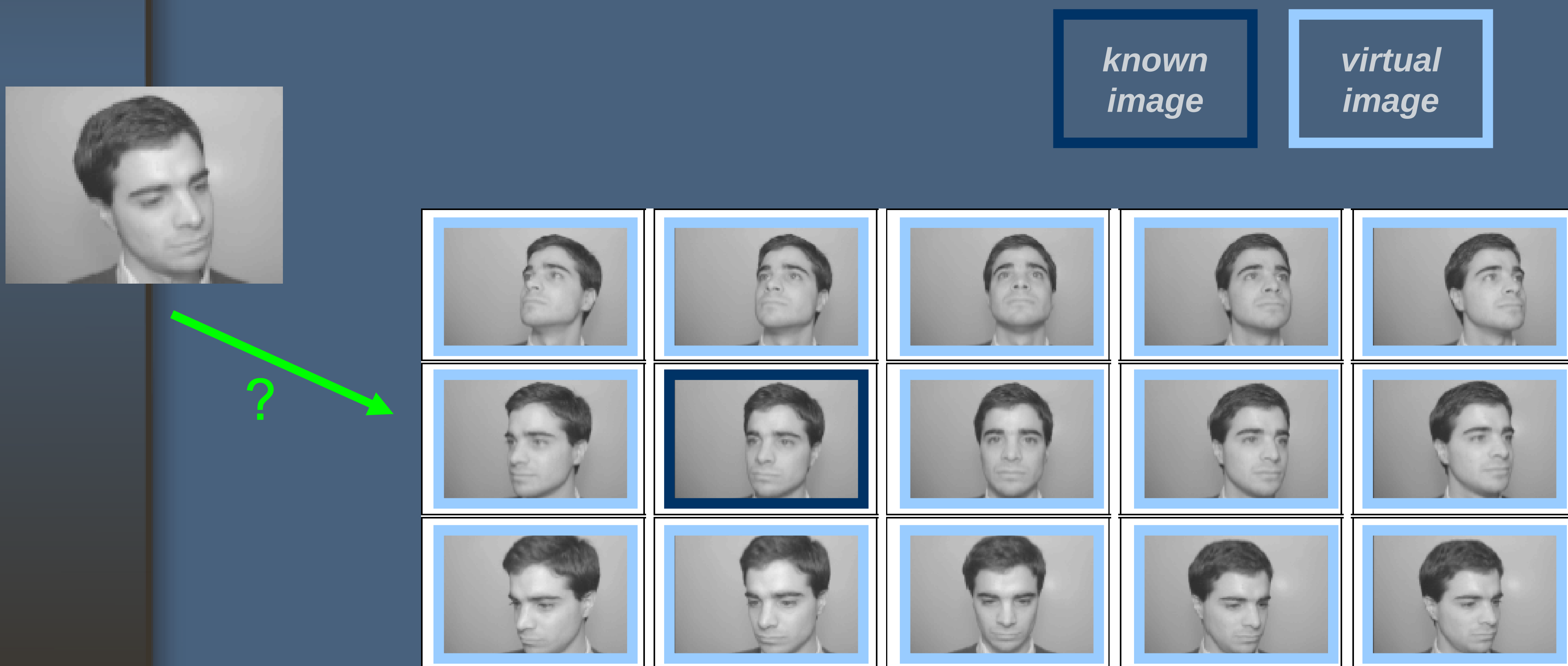
Warping A Single Image

- Algorithm to build database:
  - Have single image of most people
  - Find correspondence between new face and known face
    - Provide key features by hand, interpolate for other points
    - Similar to Active Appearance Models
  - Apply known transformations to generate many virtual views
    - Optical Flow at each point



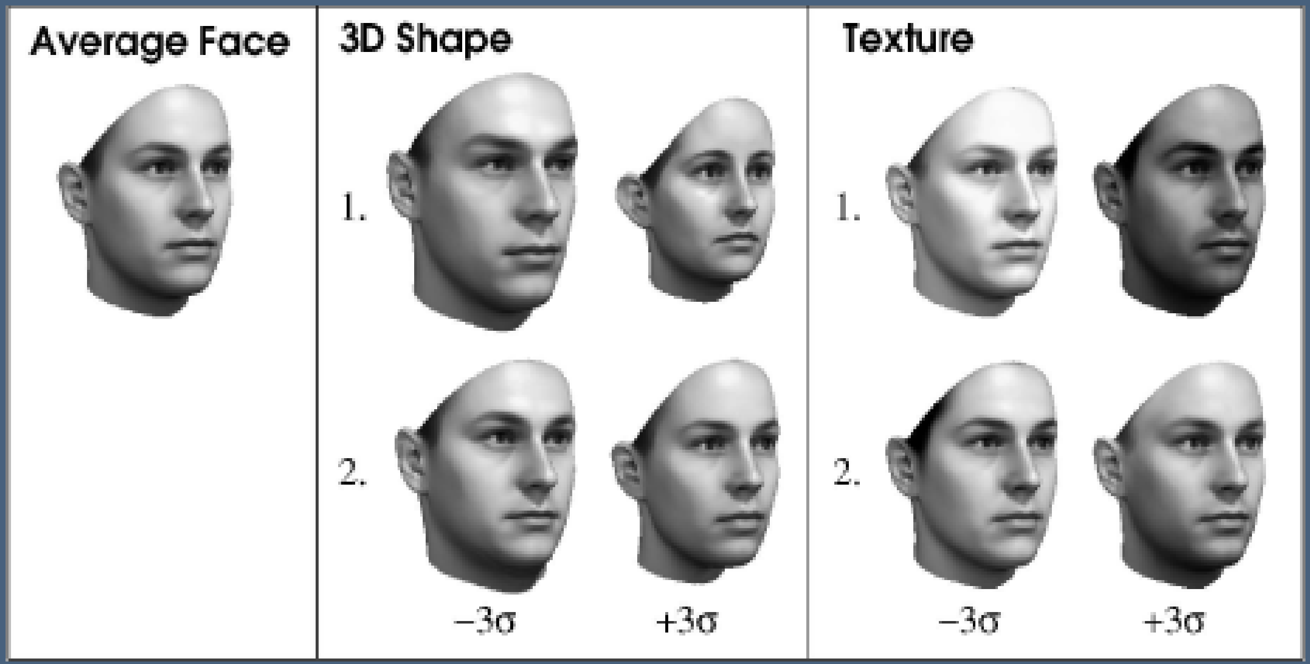
Warping A Single Image

- Testing:
  - Compare new image to most similar pose of every individual in database
  - Nearest neighbor wins



3D Morphable Model

- A “state of the Art” method solving the correspondence problem under pose and lighting variation
- Statistical 3D model instead of several 2D images



adjusting 1st principal component

adjusting 2nd principal component

PCA on 3D vector describing how a specific point differs from model average of that point

PCA on intensity value at each point

V. Blanz, T. Vetter. *“Face Recognition Based on Fitting a 3D Morphable Model.” IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 2003.*

3D Morphable Model

- m significant eigenvectors define variation of shape S and texture T
- Influence of each dimension on a particular face defined by coefficient vectors α and β

$$s = \bar{s} + \sum_{i=1}^{m-1} \alpha_i S_i$$

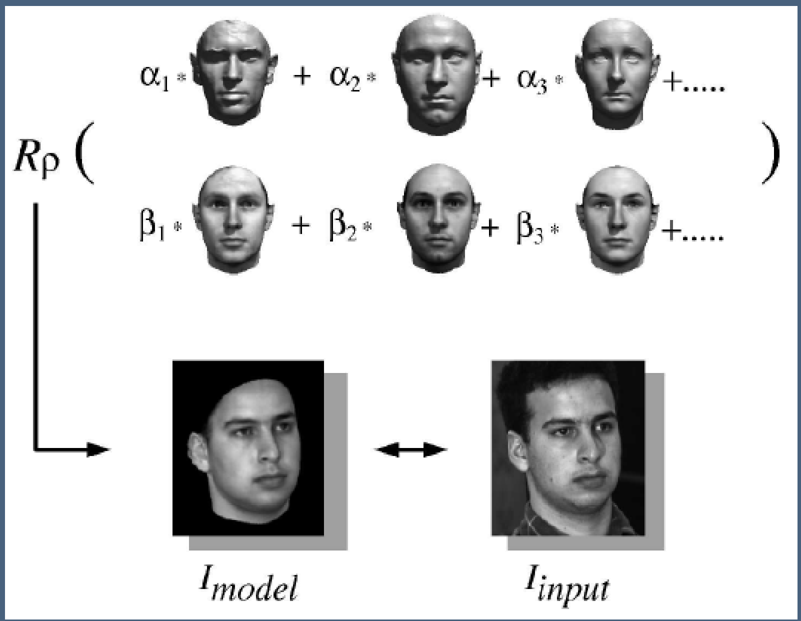
$$t = \bar{t} + \sum_{i=1}^{m-1} \beta_i T_i$$

- Construct synthetic image to closely match unknown face image
  - Minimize sum of squared distances between real and synthetic pixel intensities

3D Morphable Model

- Construct model to match image:
  - a posteriori estimate via Bayes:

$$\max P\left(\alpha, \beta, \rho | I_{in}, F\right) \sim \max P(I_{in}, F | \alpha, \beta, \rho) \cdot P(\alpha, \beta, \rho)$$

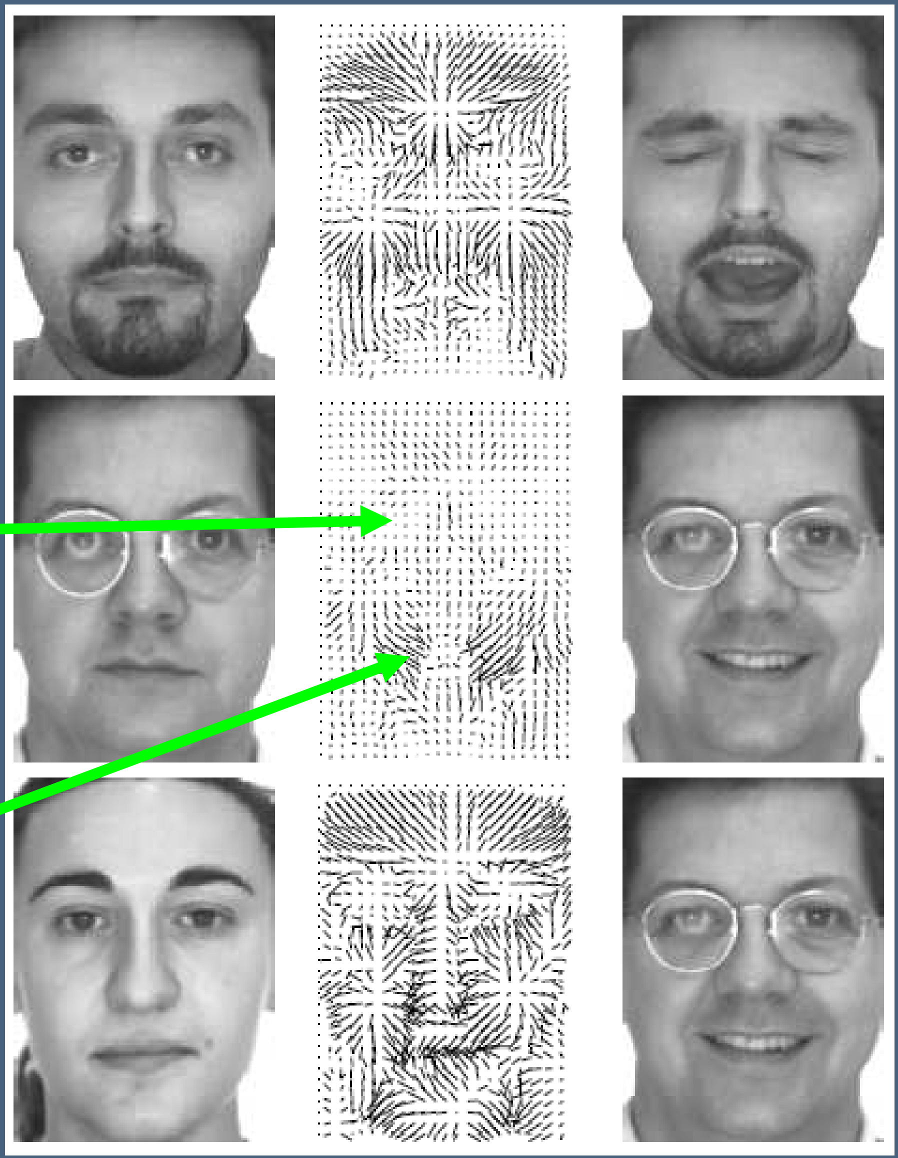


α = shape control parameters
β = texture control parameters
ρ = pose control parameters
I_{in} = new image
F = small set of feature points found during preprocessing

- Match constructed model to known person:
  - Compare model coefficients

Examine the Optical Flow

- Martinez: Weight importance of pixels by how much they deform
  - Small change: important for recognition
  - Large change: ignore



A. Martinez. "Recognizing Expression Variant Faces from a Single Sample Image per Class." IEEE Computer Vision and Pattern Recognition (CVPR), 2003.

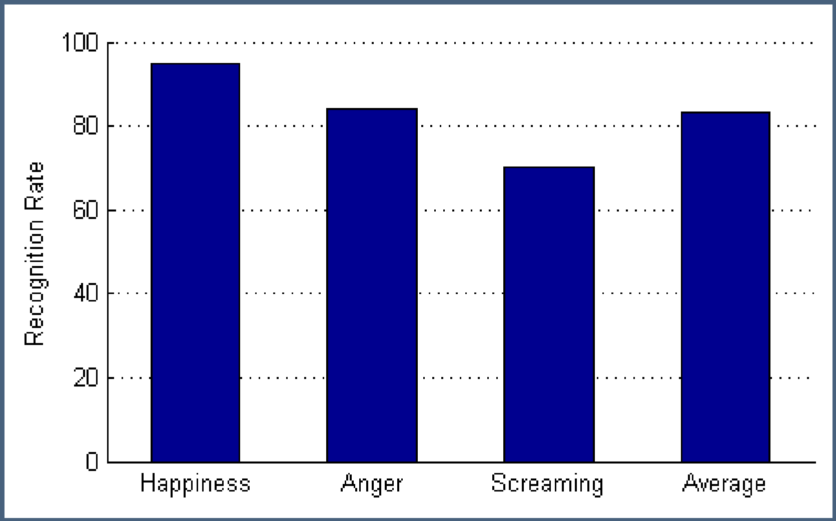
Examine the Optical Flow

- Weighting scheme
  - Compare new image T to known images I_n

$$\mathrm{F}_n = \mathrm{OpticalFlow}(I_n, T)$$

$$\mathrm{W}_{n,i} = \max_i \|F_{n,i}\| - \|F_{n,i}\|$$ (weight for each pixel i)

$$\mathrm{C}_n = \|W_n(I_n - T)\|$$ (cost to match T to I_n)

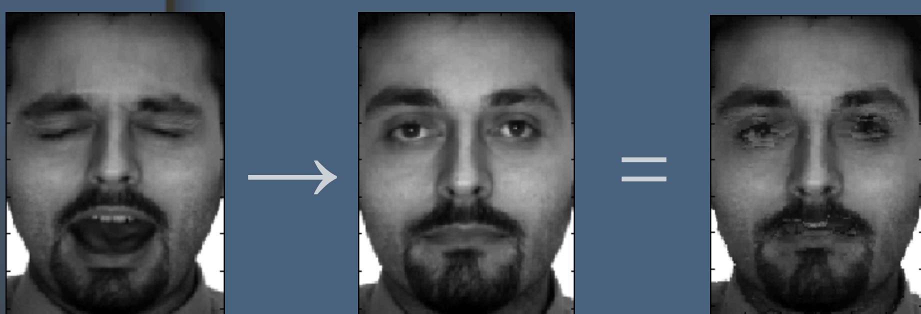


Limitations of Current Approaches

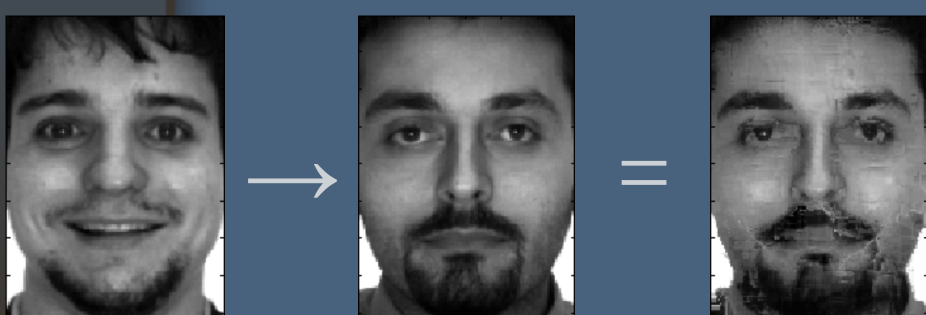
- Methods using dense correspondences only measure resulting image similarity
- Optical flow meant to solve the small motion correspondence problem
  - No reason to expect it to work for large pose or expression changes
- Need statistical models of deformation change due to expression/pose of same person vs change in identity

Measuring the Deformation

- A successful face recognition system should consider:
  - Similarity between images
  - Amount and type of deformation required to achieve this similarity



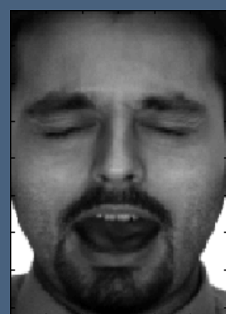
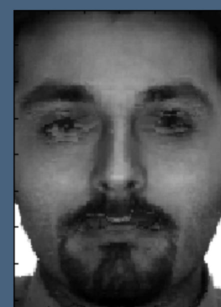
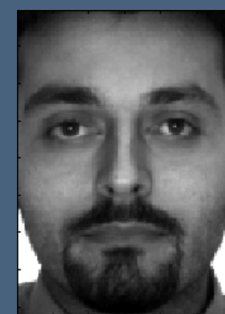
smaller $\sum \|\text{Optical Flow}\|_2$



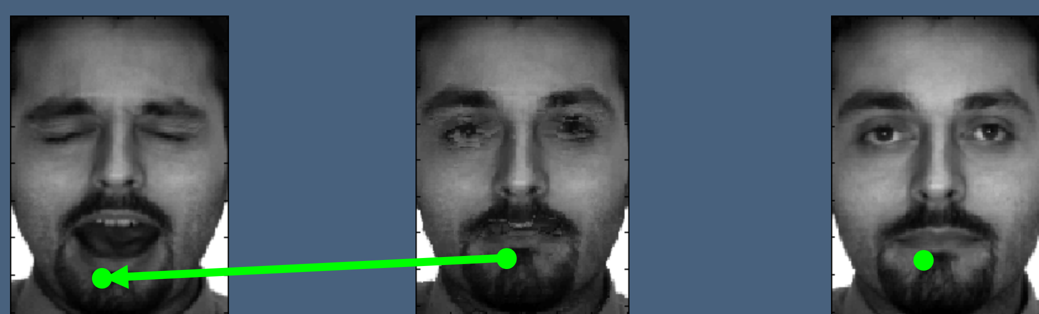
smaller $\sum \|I_1 - I_2\|_2$

Measuring the Deformation

- Similarity between images I and J
  - Let v be a transformation defined on every pixel of I such that $v(I) \approx J$

$v($  $) =$  $\approx$ 

  - For each pixel x in J, the corresponding pixel in I is $I(v^{-1}(x))$



Measuring the Deformation

- Similarity between images I and J, for all points x:

$$d\left(I(x), J(x)\right) = \|J(x) - I(v^{-1}(x))\|_2 + \lambda \|v(x)\|_g$$

deformed image intensity difference

measure of deformation

- To define:
  - Deformation v
  - Deformation norm g
  - Relative weighting λ

Measuring the Deformation

- Deformation v:
  - Traditional optical flow
  - Longer range dense correspondence
- Deformation norm g:
  - Optical flow: any metric defined on a vector field, $\sum_i \|v_i\|_2$, …
  - New field?
- Relative weighting λ:
  - Implicit using Machine Learning techniques
  - Learn from training set
  - Incorporate into g

Measuring the Deformation

- Previous methods
  - Dynamic Link Matching

$$C(x_i^I) = \lambda \sum_{(i,j)\in E} C_e(\Delta_{ij}^I, \Delta_{ij}^M) - \sum_{i\in V} C_v(J^I(x_i^I), J_i^M)$$

minimize distortions

maximize node match similarity

  - Pictorial Structures

$$L^* = \operatorname*{argmin}_L \left(\sum_{i\in V} m_i(\ell_i) + \sum_{(i,j)\in E} d_{ij}(\ell_i, \ell_j) \right)$$

part-to-model mismatch

model deformation

Optical Flow Limitations

- Optical flow meant to solve small motion correspondence problem
  - Correspondence between faces involves different set of requirements
- Alternative method meant to handle larger changes:
  - Deformations through Lie group action

Deformations Through Lie Group Action

- Image: continuous Riemannian manifold

- Lie group: diffeomorphisms of the manifold
  - The possible image deformations

- Lie algebra: vector space of infinitesimal steps in the direction of these deformations
  - Continuous vector fields deforming the image

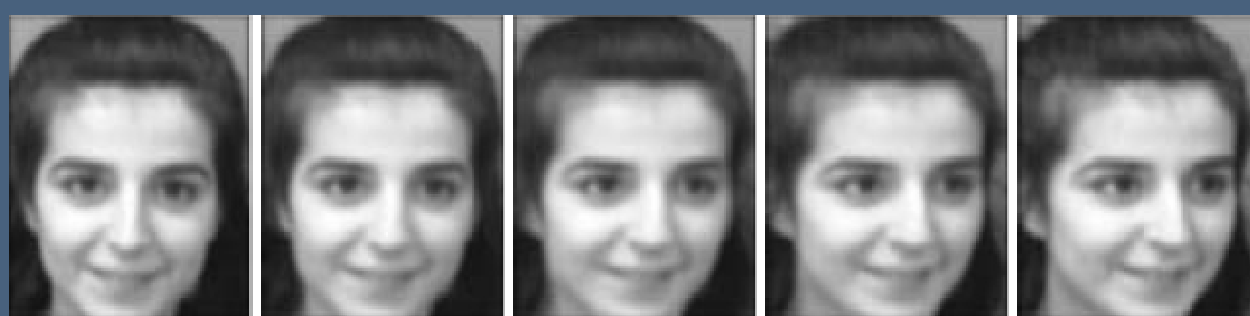
- Geodesic: the deformation requiring the least energy (v)

A. Trouve, L. Younes. "Metamorphoses Through Lie Group Action." Foundations of Computational Mathematics, 2004.

Deformations Through Lie Group Action

- Energy $E = \min_{v} \left(\int_0^1 \left\| \frac{\partial I}{\partial t} \right\|_2^2 dt + \int_0^1 \| v_t \|_g^2 \, dt \right)$

- A geodesic obtained by minimizing the energy between two given images:



input 1
t = 0

input 2
t = 1

path generated by algorithm

Future Research

- Define robust long range dense correspondences between images.
- Build statistical models of these correspondences and resulting deformations.
- Solve image classification problems using this information.