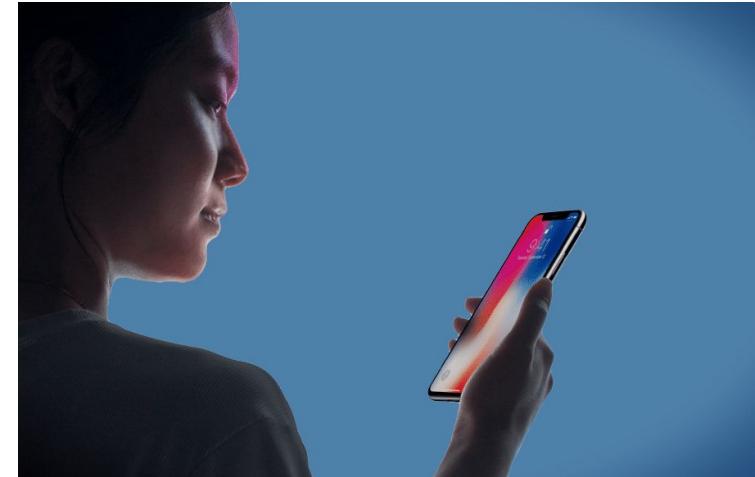


Face Representations: Classical vs Modern

Binod Bhattacharai
Imperial College London
NAAMII Nepal

Introduction



Source: *google image*

Applications (Face verification for surveillance)



Credit: AFP/Getty Images

China's police [have been testing sunglasses](#) with built-in facial recognition since at least last month to catch suspects and those traveling under false identities. Now China is

Source: The verge



Source: google image

How facial recognition is taking over airports



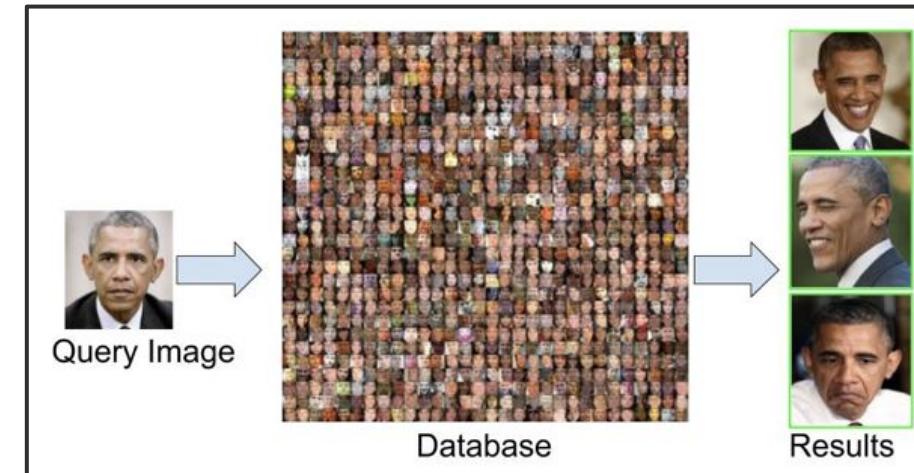
travel

(CNN) — Instead of scanning her boarding pass, the airport gate scanned her face.

In April 2019, traveler MacKenzie Fegan was left surprised and confused when she boarded a JetBlue flight from the United States to Mexico, without handing over her passport, or travel documents.

Face Analysis Tasks: Face retrieval

- Given a query image I_q , compute the ***similarity score*** between the query and the images in the database
- Rank them based on the ***sim_score***
- Retrieve **Top-K** ranked images



Face Analysis Tasks: Face Verification

Output whether the a pair of images are of a person



I_1 I_2

Same (+1)



I_2 I_3

Different (-1)

$$f:(I_i, I_j) \rightarrow \{+1, -1\}$$

Challenges in Face Matching Algorithms

Illuminations: Same face looks differently at different illuminations



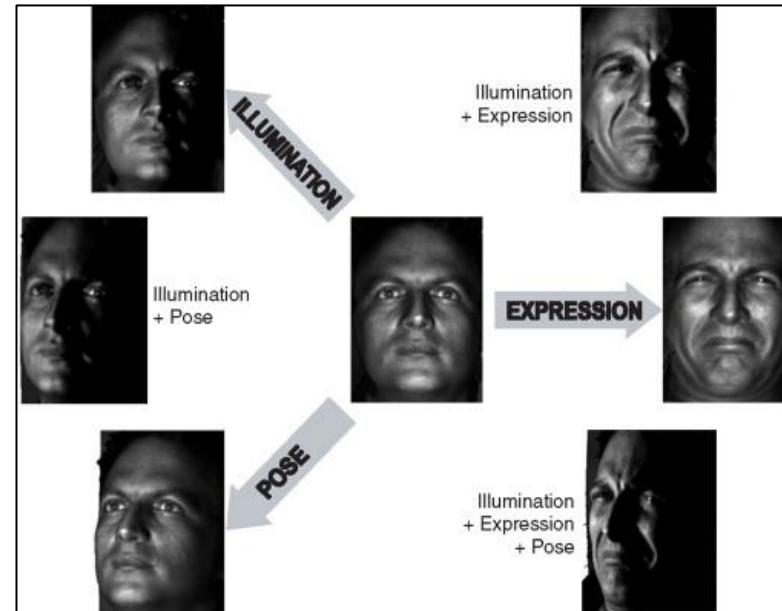
Challenges in Face Matching Algorithms

Poses: Same face looks differently at different poses

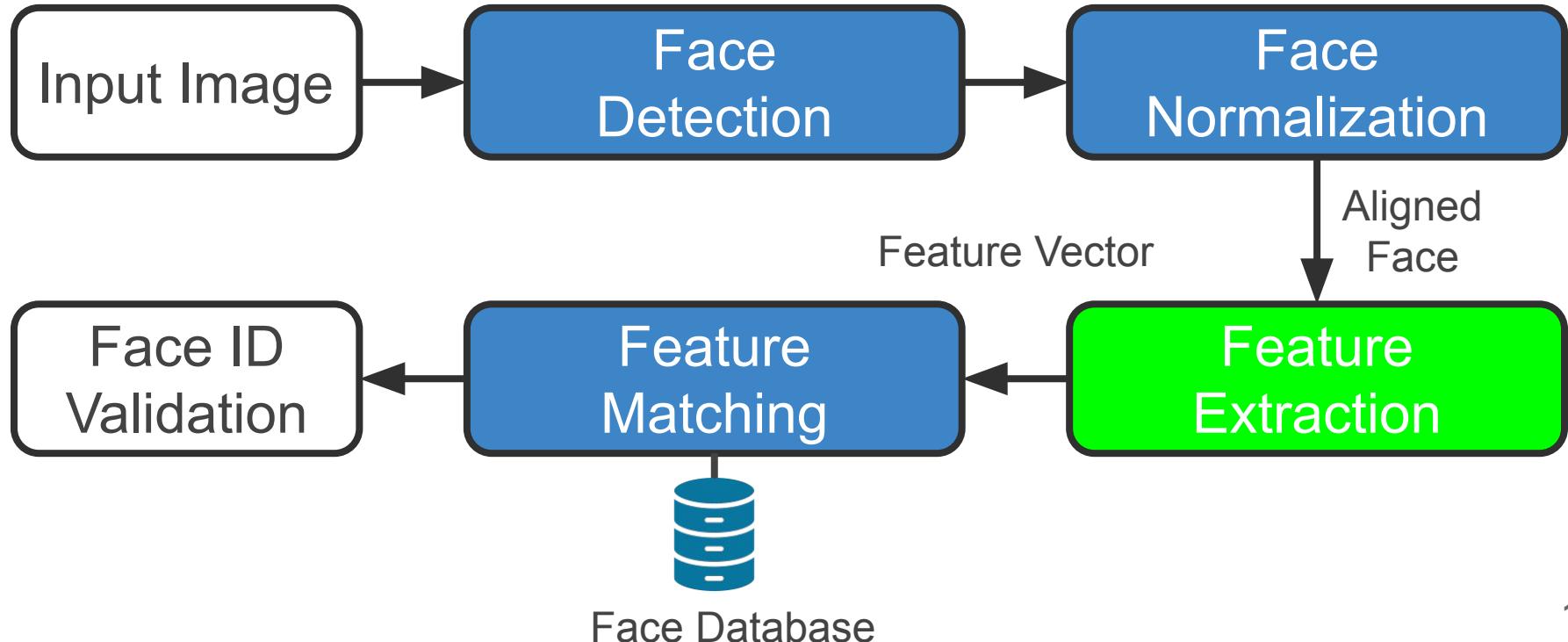


Challenges in Face Matching Algorithms

Expressions: Same face looks differently at different poses

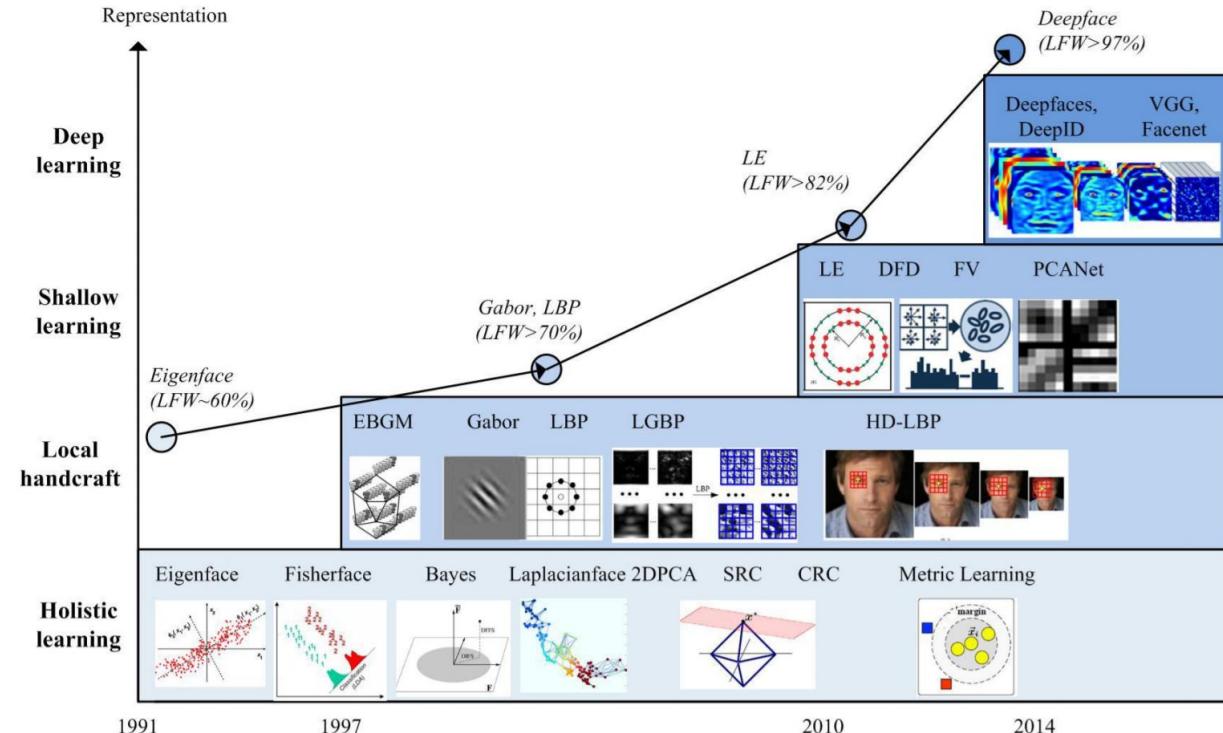


Face Analysis Pipeline



Face Analysis Pipeline

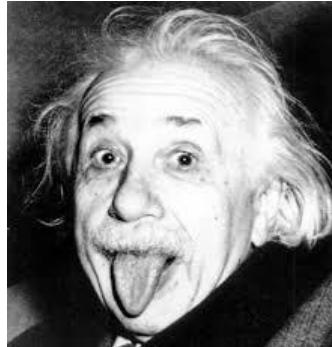
How the face matching problem is addressed?



What we discuss today?

1. Naive Approach of Image Representation
2. Eigen Faces
3. Fisher Faces
4. Local Binary Patterns
5. Distance Metric Learning
6. CNN

Naive Face Matching



128	75	72	105	149	169	127	100
122	84	83	84	146	138	142	139
118	98	89	94	136	96	143	188
122	106	79	115	148	102	127	167
127	115	106	94	155	124	103	155
125	115	130	140	170	174	115	136
127	110	122	163	175	140	119	87
146	114	127	140	131	142	153	93

Image

Pixel
intensities

Naive Face Matching

1



Image Pixels to Vector (x)

2



y_1, \dots, y_n

Test if x matches some y_i in database

$$SSD: (y_i - x)^2$$

Zero Mean Normalized Cross
Correlation (ZNCC):

The higher the ZNCC gets, the more are
the two images correlated.

Problem of Naive Approach

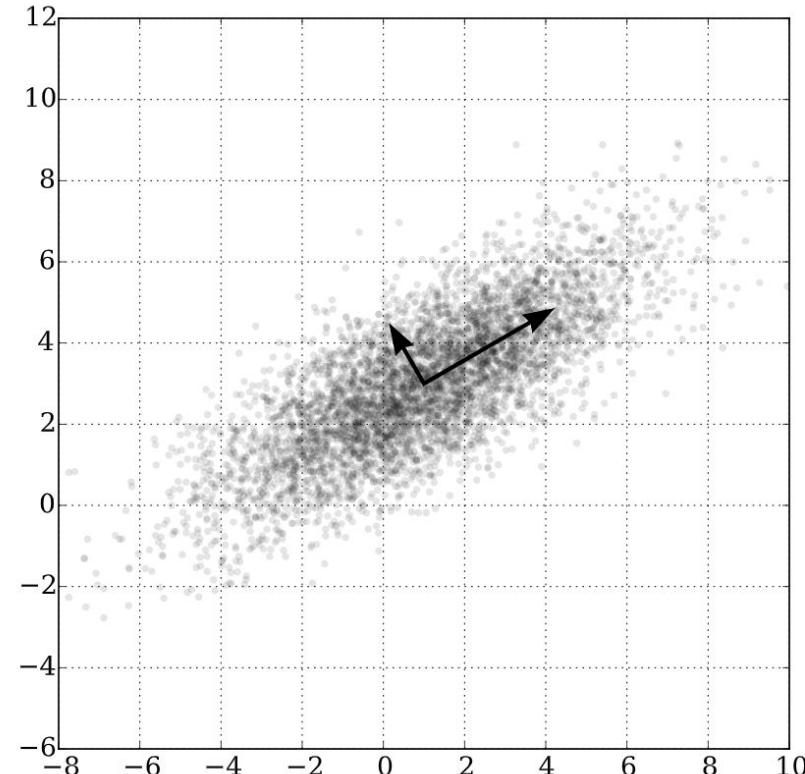
- High dimensionality
- Not robust to illuminations, poses, expressions

Eigenfaces

1. PCA : Principal component analysis

- ❑ 2D scatterplot showing the distribution of data
- ❑ Each point represents a data
- ❑ Projecting data on First principal axis alone does not incur much error
- ❑ How to identify such Principal axis of data ?
- ❑ Use PCA

<http://setosa.io/ev/principal-component-analysis/>



PCA

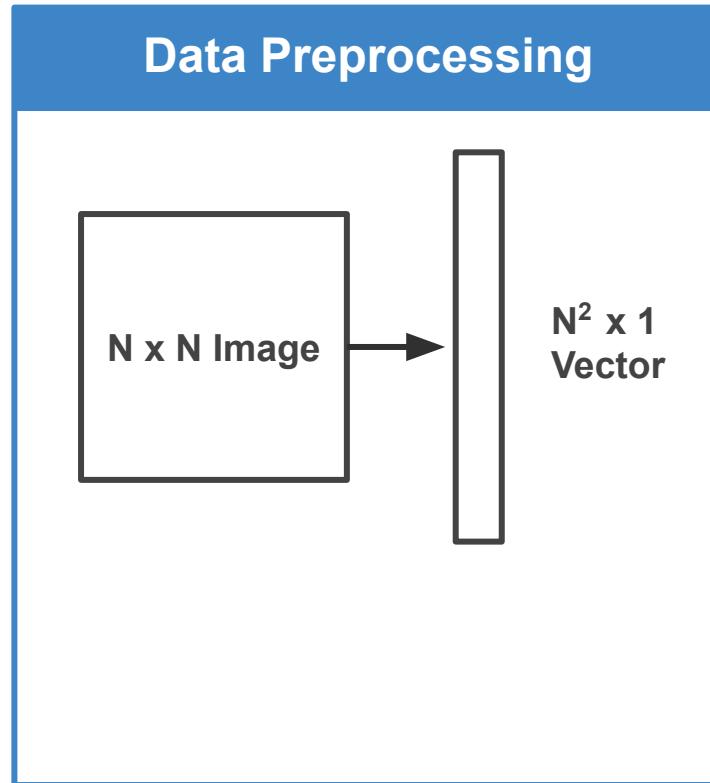
$$Cov(X, Y) = \frac{\sum_{i=1}^N (X - \hat{X})(Y - \hat{Y})^T}{N}$$
$$\hat{X} = \frac{1}{N} \sum_{i=1}^N X_i$$

$\det(A - \lambda I) = 0$, A is covariance matrix

Eigen Faces

Step

1



- Each sample image is rearranged into column vector of length $N \times N = N^2 \times 1$

Eg: For a image of 112*92
Column Vector length is:
 $112 \times 92 = 10304$
- All images are put into a **matrix A** of size $N^2 \times M$
For a dataset of size $M=400$ images A is 10304×400
- Mean face is subtracted from each column.
- We apply PCA to compute eigen faces

Eigen Faces

Step
2

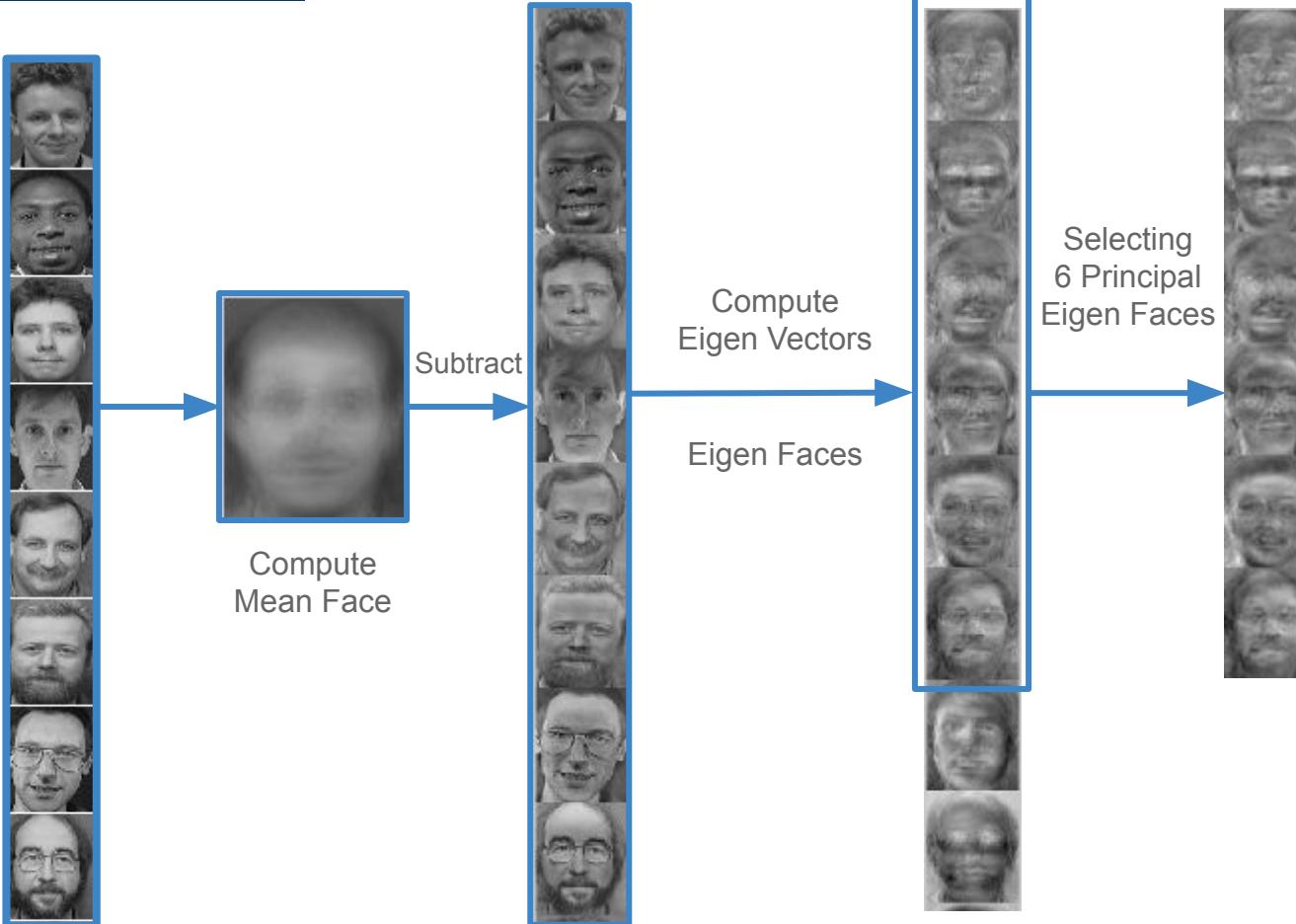
PCA

Find the eigenvectors of
covariance matrix C of
(face - mean face)

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = AA^T \quad (N^2 \times N^2 \text{ matrix})$$

where $A = [\Phi_1 \ \Phi_2 \ \dots \ \Phi_M] \quad (N^2 \times M \text{ matrix})$

Eigen Faces



Eigen Faces

Projection

- Select top k eigenvectors with k largest eigenvalues (**k eigenfaces**)

Step

3

- Do projection along these eigenfaces to find

Each face (minus the mean) in the training set can be represented as a linear combination of the best k eigenvectors:

$$\Phi_i(\text{norm face}) = \sum_{j=1}^K B_j u_j , \text{ where } B_j = \Phi_i u_j^T$$
$$B \in R^{n \times k}$$

Typical eigenfaces when k=4:



\bar{u}_1

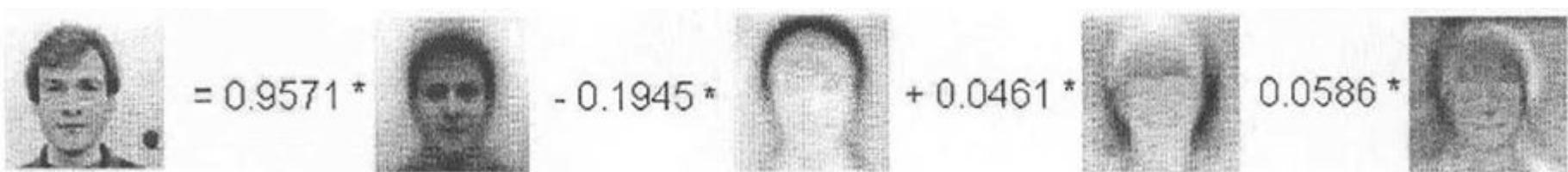
\bar{u}_2

\bar{u}_3

\bar{u}_4

Eigen Faces

Since $\{\bar{u}_1, \bar{u}_2, \bar{u}_3, \bar{u}_4\}$ is an orthonormal basis, any face (after mean subtraction) can be represented by this basis:



Each normalized training face Φ_i is represented in this basis by a vector:

PCA

Merits

- Removes redundancies and noise to make feature more discriminative
- Transform the representations into compact form (dimensionality reduction)
- Does not require label information

PCA

- ❑ **Demerits**
 - ❑ Does not embed category information
 - ❑ Large variations on illuminations or poses (not necessarily the axis with maximum variance contains discriminative features)



- ❑ Not robust to illuminations, expressions, poses etc.

Fisher Faces

- ❑ Developed in 1997 by P.Belhumeur et al.
- ❑ Unlike PCA, uses **class label information**
- ❑ Works well even if different illumination
- ❑ Works well even if different facial expressions

How Fisherfaces Works?

$$y = w^T x$$

$$y \in R^m, x \in R^n$$



- N Sample Images $\{x_1, x_2, \dots, x_N\}$
- N dimensional image space
- Each image belongs to one of c classes $\{X_1, X_2, \dots, X_c\}$

Linear transformation mapping of n-dim image space to m-dim Feature space

Where $w \in R^{n \times m}$ is a matrix with orthonormal columns (i.e. $m < n$)

Note

We want to learn projection w that converts all points from x to a New space y.

How Fisherfaces Works?

Scatter Matrices for c classes

- Scatter of class i:

$$S_i = \sum_{x_k \in Y_i} (x_k - \mu_i)(x_k - \mu_i)^T$$

- Within class scatter:

$$S_W = \sum_{i=1}^c S_i$$

- Between class scatter:

$$S_B = \sum_{i=1}^c N_i (\mu_i - \mu)(\mu_i - \mu)^T$$

How Fisherfaces Works?

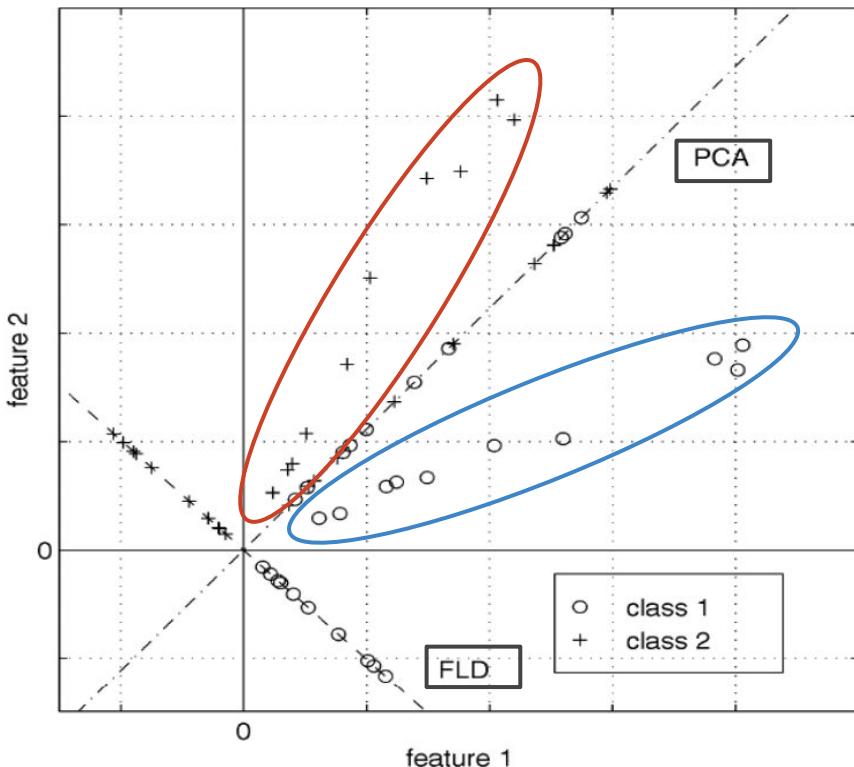
Now we need an **objective function** to maximize between class scatter and minimize within class scatter

$$J(w) = \max \frac{\text{scatter between class}}{\text{scatter within class}}$$

Between Class Scatter: $W^T S_B W$ Within Class Scatter: $W^T S_w W$

$$W_{opt} = \operatorname{argmax}_w \left| \frac{W^T S_B W}{W^T S_w W} \right|$$

Eigenface and Fisherface



- PCA:**
Component axes that Maximize the variance
- LDA:**
Maximize the component axes for class separation

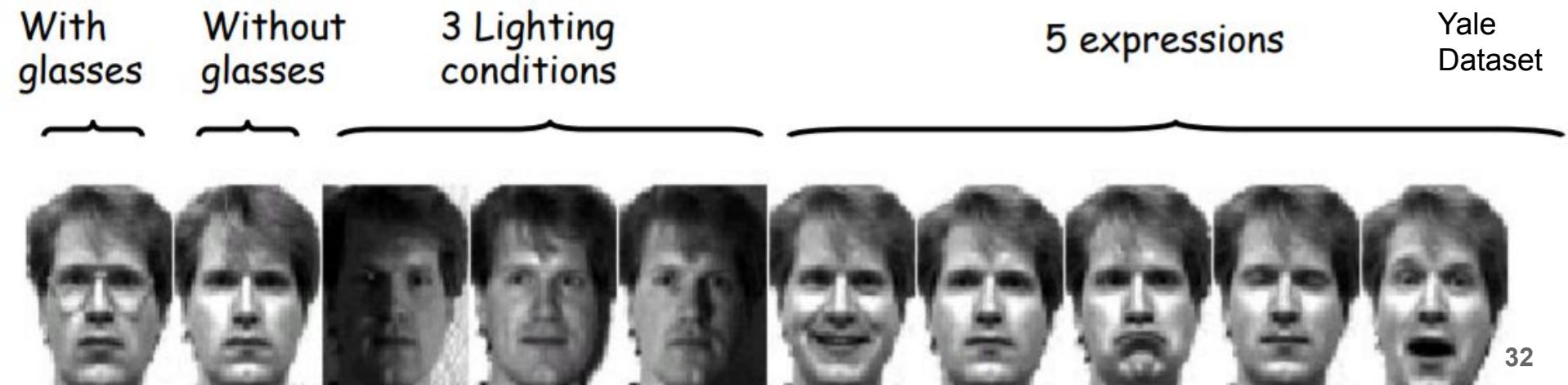
Eigenface and Fisherface

- ❑ Eigenfaces attempt to **maximize the scatter** of the training images in face space.
- ❑ LDA seeks directions that are efficient for discrimination between the data
- ❑ Fisherfaces attempt to **maximize the between class scatter**, while **minimizing the within class scatter**.
- ❑ Moving Images of the same face closer together, while moving images of difference faces further apart.

Eigenface Vs Fisherface

Input: 160 images of 16 people

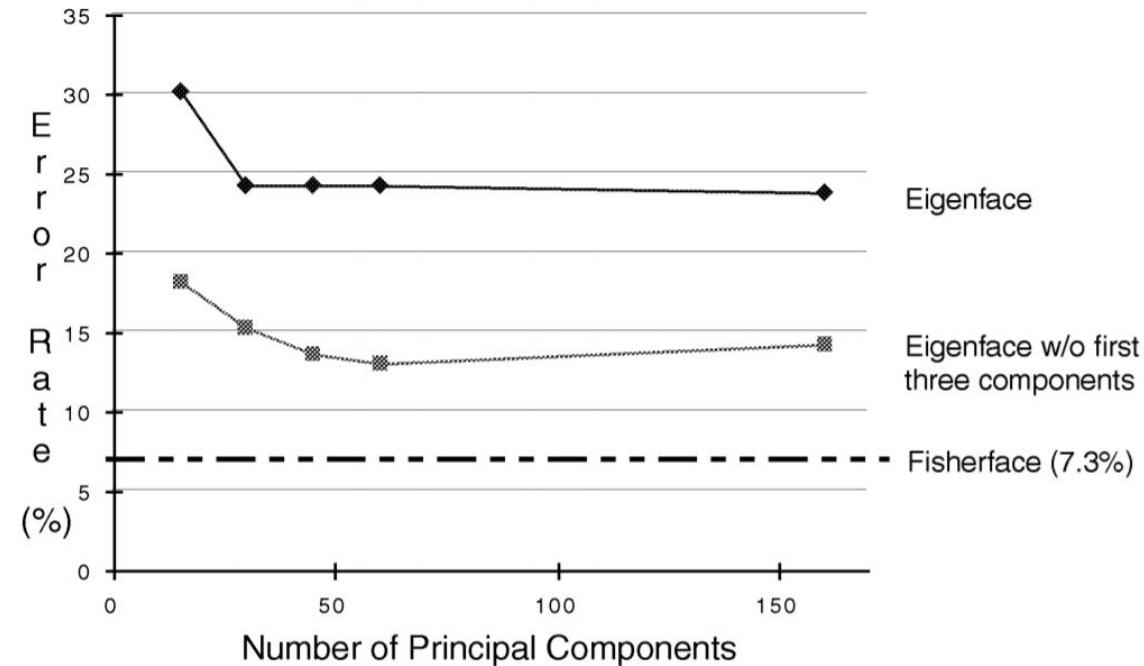
- Train: 159 images
- Test: 1 image
- Variation in Facial Expression, Eyewear, and Lighting



Fisher Face

Yale dataset

Test results demonstrated LDA/FDA is better than eigenface using linear PCA (1997).



Local Binary Patterns

For finding good descriptor for face

Holistic methods: 1. PCA 2. LDA

Local Region methods:

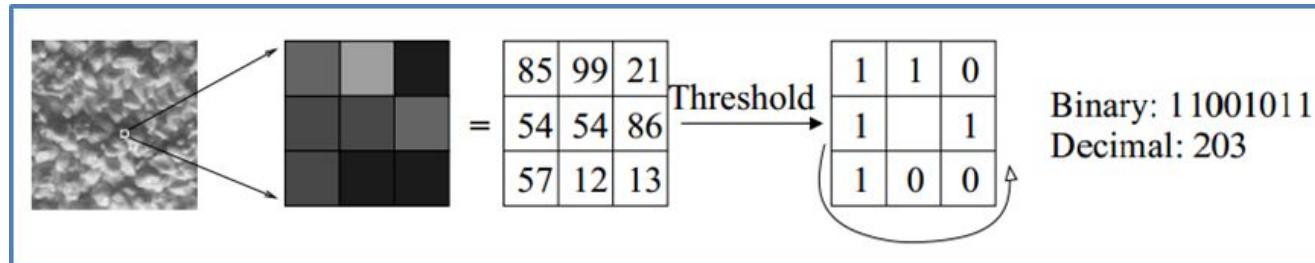
Facial image representation based on LBP texture features from local facial regions

Note

Local descriptors gained attention due to their robustness to challenges such as pose and illumination changes

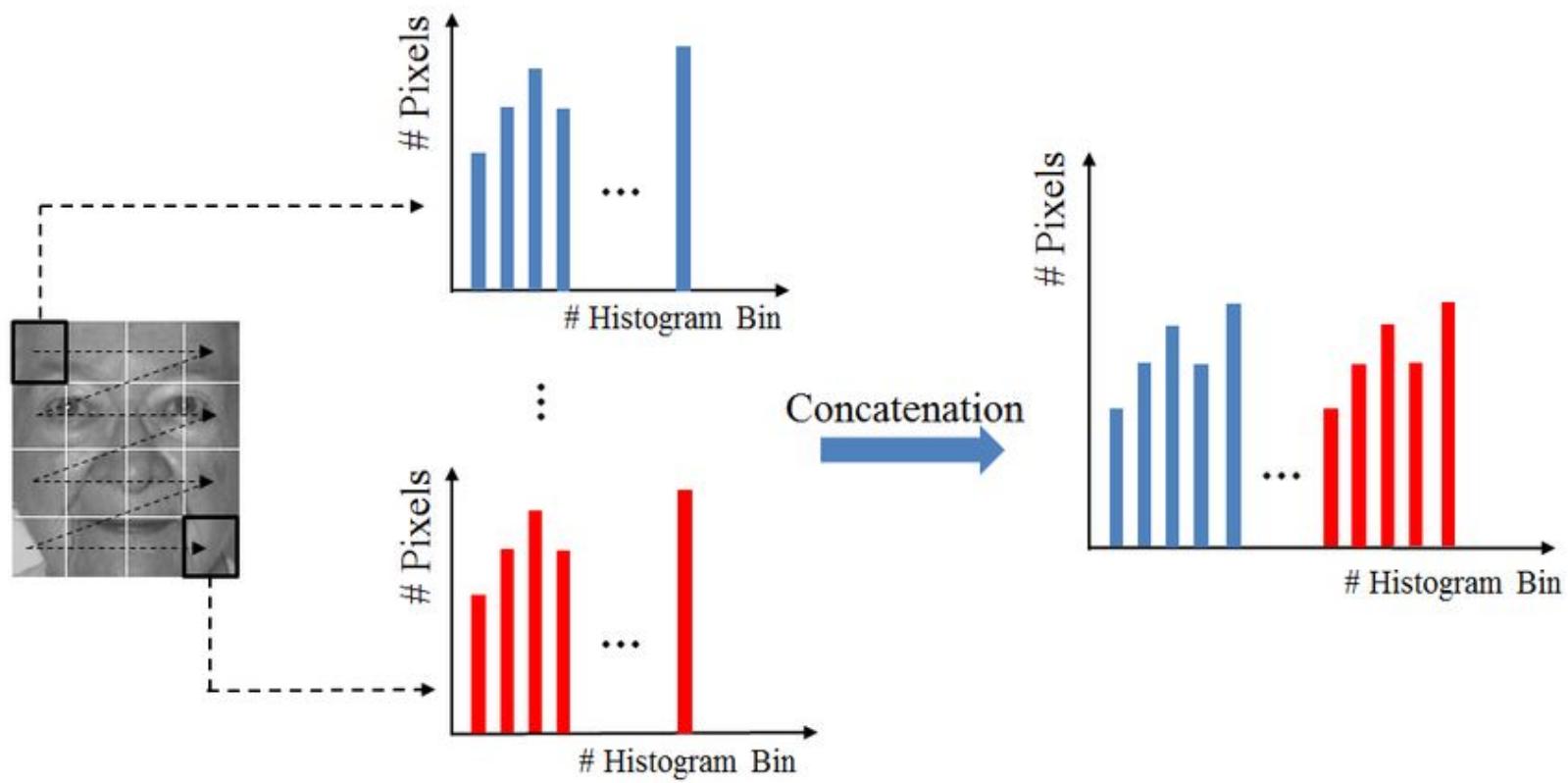
Local Binary Patterns

- One the best performing texture descriptors
- A label is assigned to every pixel
- Use center pixel value to threshold the 3x3 neighborhood
- Result in binary number
- Histogram of the labels is used as a texture descriptor

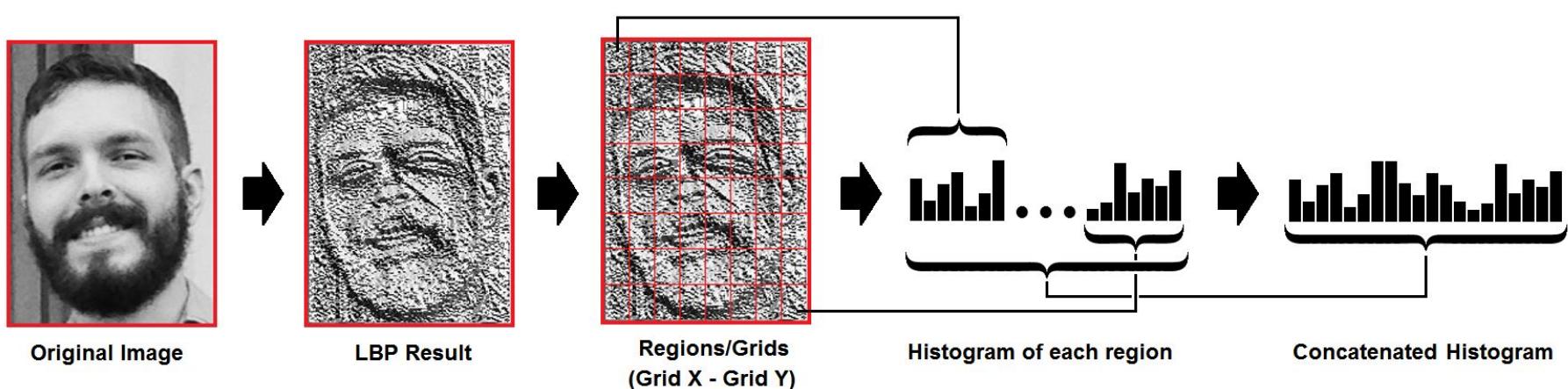


Basic LBP Operator

Linear Binary Patterns



Extracting the Histograms



Description of the face on three levels of locality

1. The LBP labels for the histogram contain information about the patterns on a **pixel-level**
2. The labels are summed over a small region to produce information on a **regional level**
3. The regional histograms are concatenated to build a **global description of the face**.

LBP advantages over Linear representations

- Non-linear representations
- Illumination invariant
- Age Invariant

Conclusion of LBP

- Facial images can be seen as micro-patterns(spots, edges, lines, etc.).
- This texture-based facial descriptor is based on dividing image into small regions.
- Texture description of each region can be computed using LBP
- Combining these descriptors into a spatially enhanced histogram or feature vector can be used for face recognition.

OH, WAIT.....Limitations

- They produce rather long histograms, which slow down the recognition speed especially on large-scale face database.
- Generic in nature, not optimised for the end task

Metric Learning

- Euclidean or L2 distance is probably the most well known metric

$$d_{L2}(x,y) = (x - y)^T (x - y)$$

- No parameter to learn

- Most common form of learned metrics are Mahalanobis

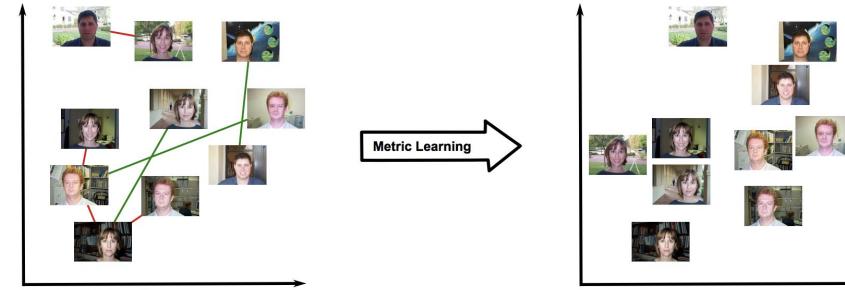
$$d_M(x,y) = (x - y)^T M(x - y)$$

- $M \in \mathbb{R}^{(D \times D)}$ is a semi-definite matrix as it measures distance
 - Generalization of Euclidean metric (setting $M=I$)
 - M can be decomposed into $L^T L$, $L \in \mathbb{R}^{(D \times d)}$ $d << D$
 - Corresponds to Euclidean metric after linear projection of data

$$d_M(x,y) = (x - y)^T M(x - y) = (x - y)^T L^T L(x - y) = d_{L2}(Lx, Ly)$$

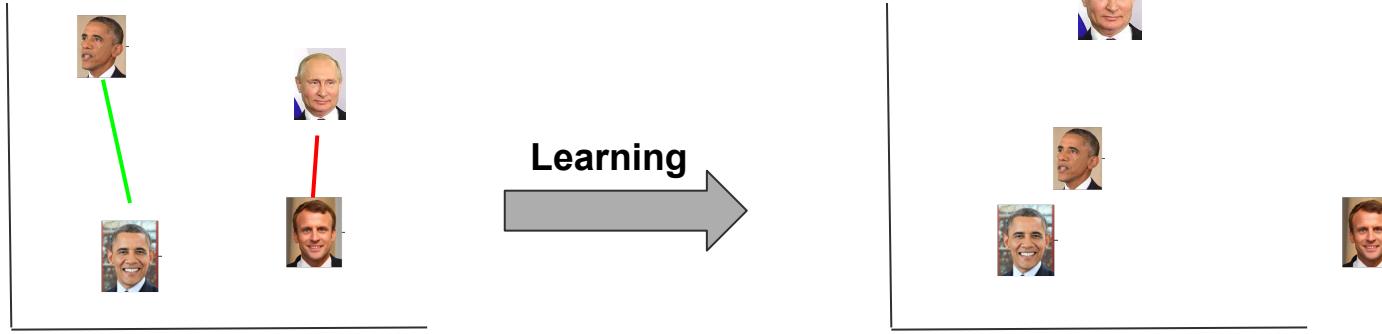
- Reduces the dimension by large-margin

Supervised Metric Learning (ML) for face verification



- ❑ Learn a projection matrix where the imposed constraints are **better satisfied**
- ❑ Commonly used constraints are: ***pairwise similarity and dissimilarity constraints*** and ***triplet constraints***

Pairwise constraints

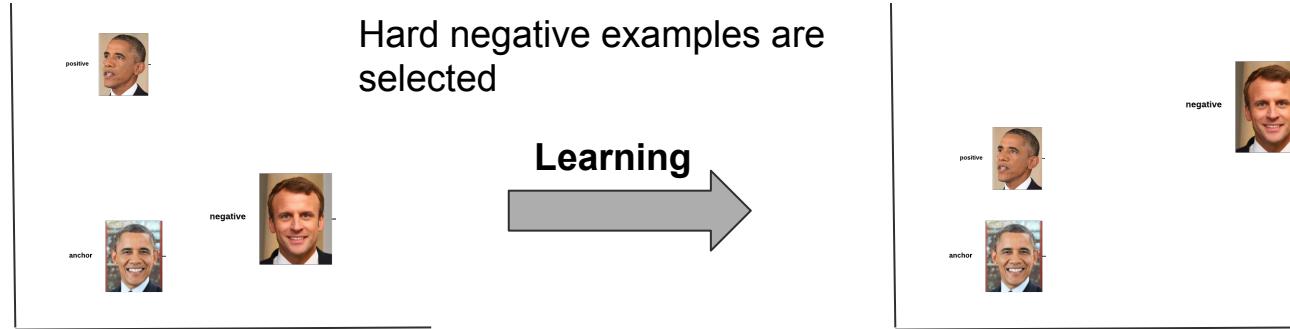


- Must-link / cannot-link constraints (sometimes called positive / negative pairs):

$$\mathcal{S} = \{(x_i, x_j) : x_i \text{ and } x_j \text{ should be similar}\},$$

$$\mathcal{D} = \{(x_i, x_j) : x_i \text{ and } x_j \text{ should be dissimilar}\}.$$

Triplet constraints



- Relative constraints (sometimes called training triplets):

$$\mathcal{R} = \{(x_i, x_j, x_k) : x_i \text{ should be more similar to } x_j \text{ than to } x_k\}.$$

Learning projection matrix

- We minimize the max-margin objective function to learn the projection matrix satisfying pairwise dis(similar) constraints ($y_{ij} = +/-1$)

$$\underset{L}{\operatorname{argmin}} \sum_{t=1}^{t=n} \max \left(m - y_{ij}^t (b - d_L^2(\hat{x}_i^t, \hat{x}_j^t)), 0 \right)$$

- Where $d_L^2(x_i, x_j) = \|Lx_i - Lx_j\|^2$
- Pushes the examples s.t. Distance of negative pairs is larger by ' m ' than bias ' b '

Learning the parameters of the projection matrix

- We use stochastic gradient descent

$$\frac{d_L^2(x_i, x_j)}{dL} = L(x_i - x_j)(x_i - x_j)^T$$

- Update rule
 - if** $y_{ij}(b - d_L^2(x_i, x_j)) < m$ **then**
 - $L \leftarrow L - \eta y_{ij} L(x_i - x_j)(x_i - x_j)^T$
 - else**
 - no update
 - end if**

Performance comparison

- ❑ Database
 - ❑ LFW: Labeled Faces in the Wild, contains 13K of 5K identities
 - ❑ Standard benchmark for face analysis task
- ❑ Performed face retrieval task
- ❑ Metric used is $1\text{-call}@K$ (2, 5, 10)

Method	K=2	K=5	K=10	K=20
PCA	30.0	37.4	43.3	51.3
ML	38.1	51.1	60.5	69.3

Source: Bhattacharai et al CVPR 2016

Limitations of parameterized distance

- In summary:



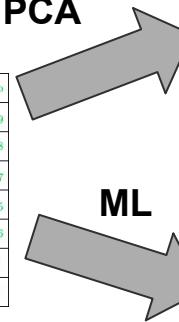
Image



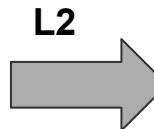
Handcrafted
feature

128	75	72	105	149	169	127	100
122	128	75	72	105	149	169	127
118	122	84	83	84	140	138	142
122	118	98	89	94	136	96	143
127	122	106	79	115	148	102	127
125	127	115	106	94	155	124	103
127	125	115	130	140	170	174	115
146	127	110	122	163	175	140	119
	146	114	127	140	131	142	153
							93

PCA

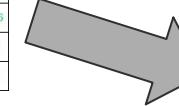


PCA
subspace

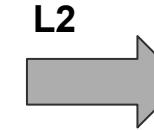


Face compare

ML



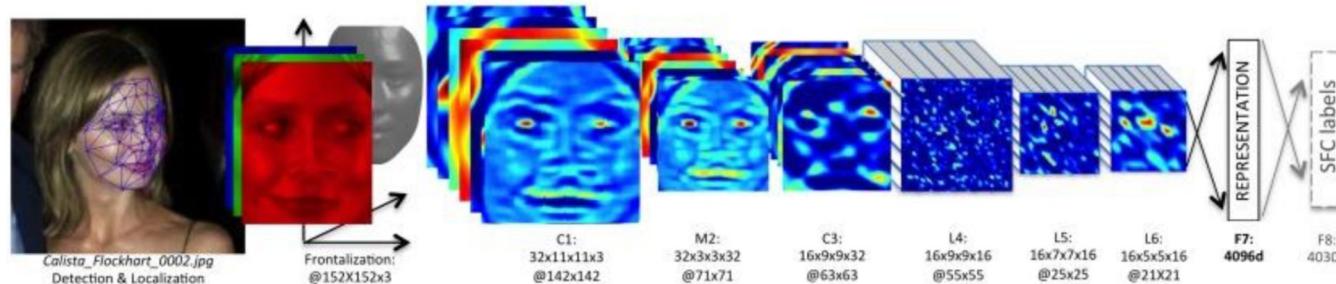
ML
subspace



Face compare

- Multi-stage approach
- Features are not optimal for end task (no feedback mechanism to propagate error to input)

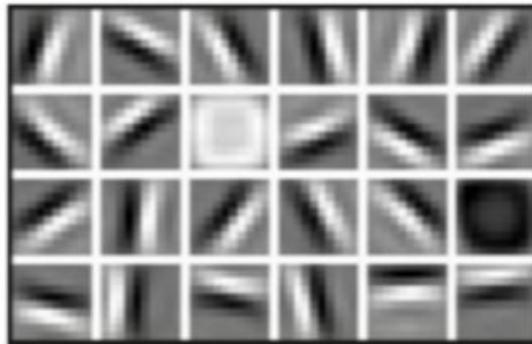
Deepface



- Train set size: 4M images from 4K identities
- Minimize cross-entropy loss to learn the parameters

Taigman, Yaniv, et al. "Deepface: Closing the gap to human-level performance in face verification." CVPR 2014

Low Level Features



Lines & Edges

Mid Level Features



Eyes & Nose & Ears

High Level Features



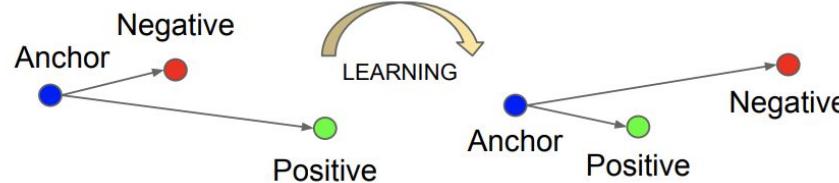
Facial Structure

Input---**Shallow Layers**-----**Middle Layers**-----**Deeper Layers** ----> Output

Experiments

- ❑ Dataset: LFW
- ❑ Accuracy: 0.9963 Vs 0.9735 (Deepface) Vs 0.975 (Human)

FaceNet



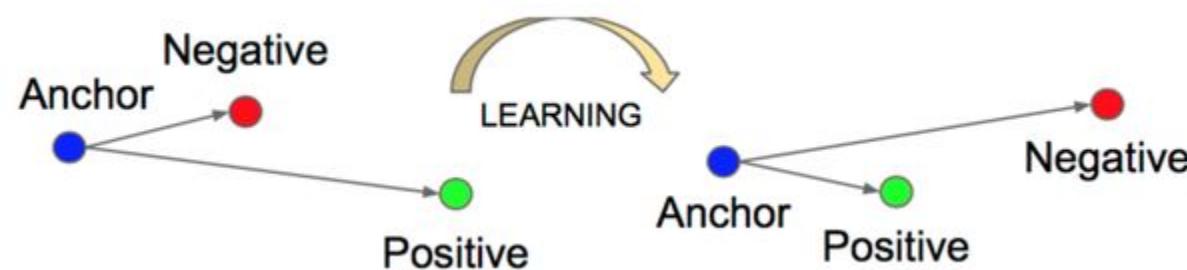
- ❑ Objective of this architecture is to minimize L2 distance between same identity's faces representations
- ❑ Directly transforms image representations at a low dimensional feature space (128D vs 4096D (Deepface)) rather than bottleneck intermediate representations

Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." CVPR 2015

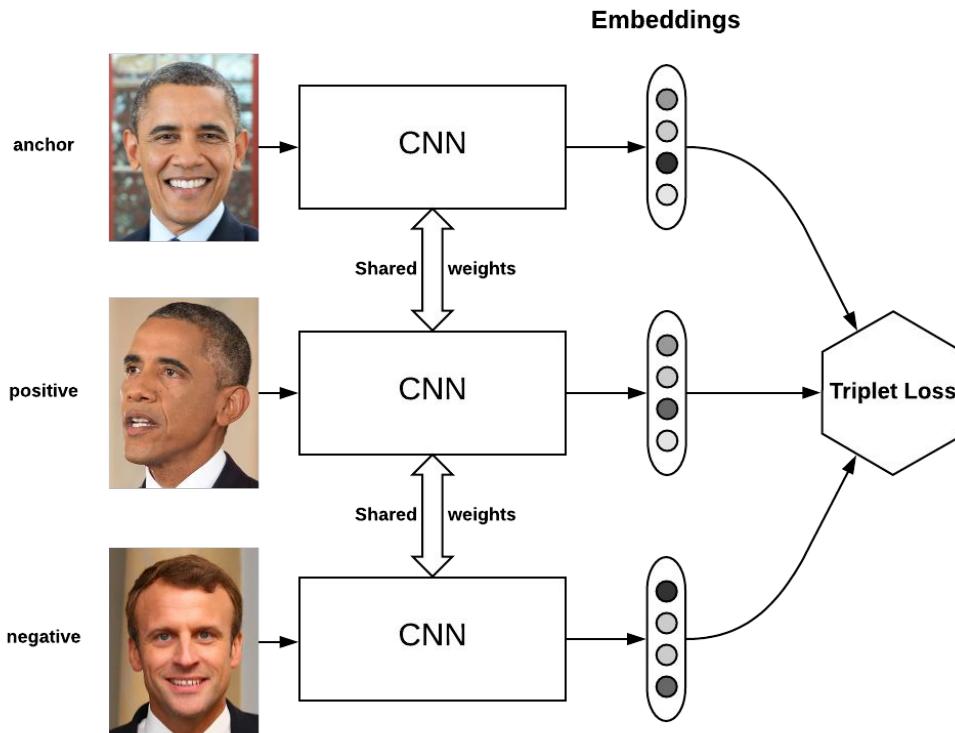
Triplet Loss

FaceNet: In the FaceNet paper, a convolutional neural network architecture is proposed. For a loss function, FaceNet uses “triplet loss”. Triplet loss relies on minimizing the distance from positive examples, while maximizing the distance from negative examples.

$$\sum_i^N \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+$$



Triplet Loss



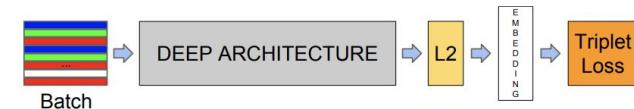
A, P, N

$$d(A, P) + \alpha \leq d(A, N)$$

Choose triplets that are hard to train on

$$d(A, P) \text{ close to } d(A, N)$$

FaceNet



- ❑ Uses triplet loss
- ❑ Minimize the max-margin objective

$$\sum_i^N \left[\|f(x_i^a) - f(x_i^p)\|_2^2 - \|f(x_i^a) - f(x_i^n)\|_2^2 + \alpha \right]_+$$

s.t.

$$\begin{aligned} & \|f(x_i^a) - f(x_i^p)\|_2^2 + \alpha < \|f(x_i^a) - f(x_i^n)\|_2^2 , \\ & \forall (f(x_i^a), f(x_i^p), f(x_i^n)) \in \mathcal{T} . \end{aligned}$$

- ❑ This ensures all positive examples are nearer than negative examples
- ❑ Very useful for clustering of faces

Schroff, Florian, Dmitry Kalenichenko, and James Philbin. "Facenet: A unified embedding for face recognition and clustering." CVPR 2015

Mis-classified examples

False reject



False accept



Bottlenecks in face analysis using deep learning

- ❑ Computing resource
- ❑ Data hungry

Additional Slides

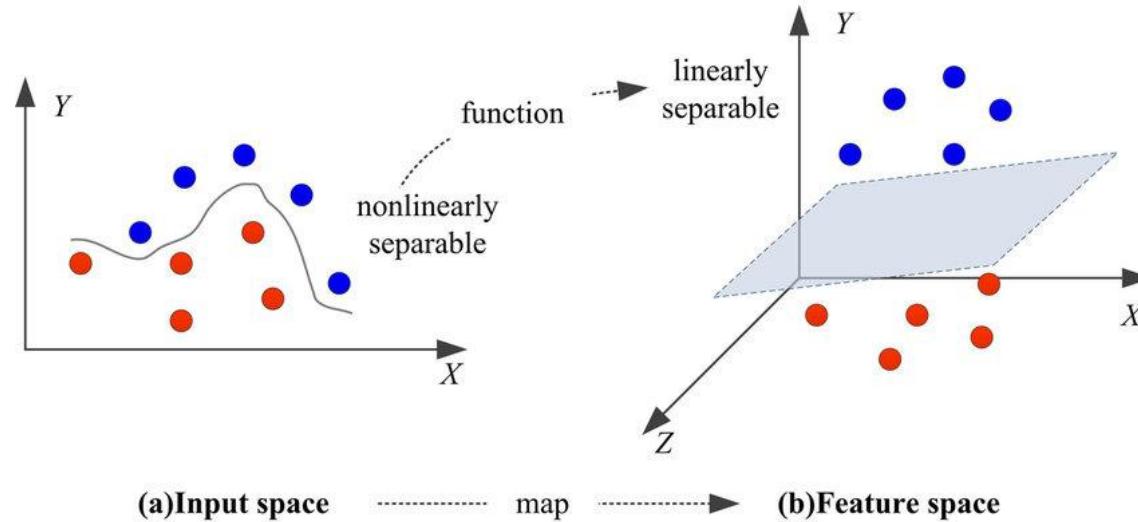
Face Analysis Tasks: Face Recognition

1:N Problem

Database of N persons

Input Image - Output whether the input image is any of the N persons (or Not)

Feature Representations



What makes face recognition a difficult problem?



Illumination



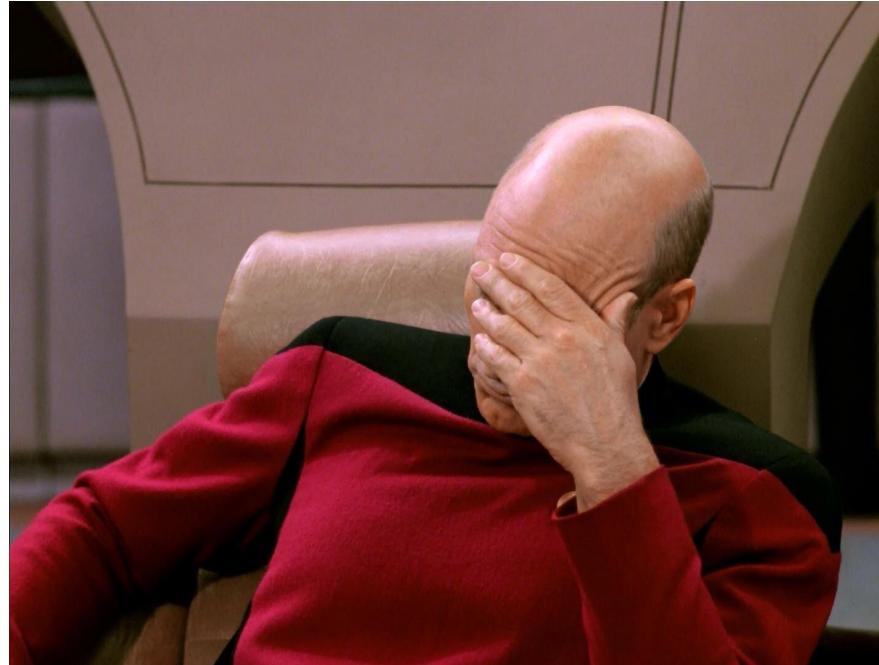
What makes face recognition a difficult problem?

Viewpoint



What makes face recognition a difficult problem?

Occlusion



Simplest way for solving face recognition problem

1



Image Pixels to Vector (x)

2



y_1, \dots, y_n

Test if x matches some y_i in database

$$\text{SSD: } (y_i - x)^2$$

Zero Mean Normalized Cross Correlation (ZNCC):

The higher the ZNCC gets, the more are the two images correlated.

Face Representation Methods: Conventional to Modern

Eigen Faces

Fisher Faces

Local Binary Patterns

SIFT

Fisher Vectors

Face Detection Using Eigenfaces

- Given an unknown image Γ

Step 1: compute $\Phi = \Gamma - \Psi$

- Minimum distance classifier
- SVMs
- Neural networks
- Quadratic classifier
- Gaussian mixture models

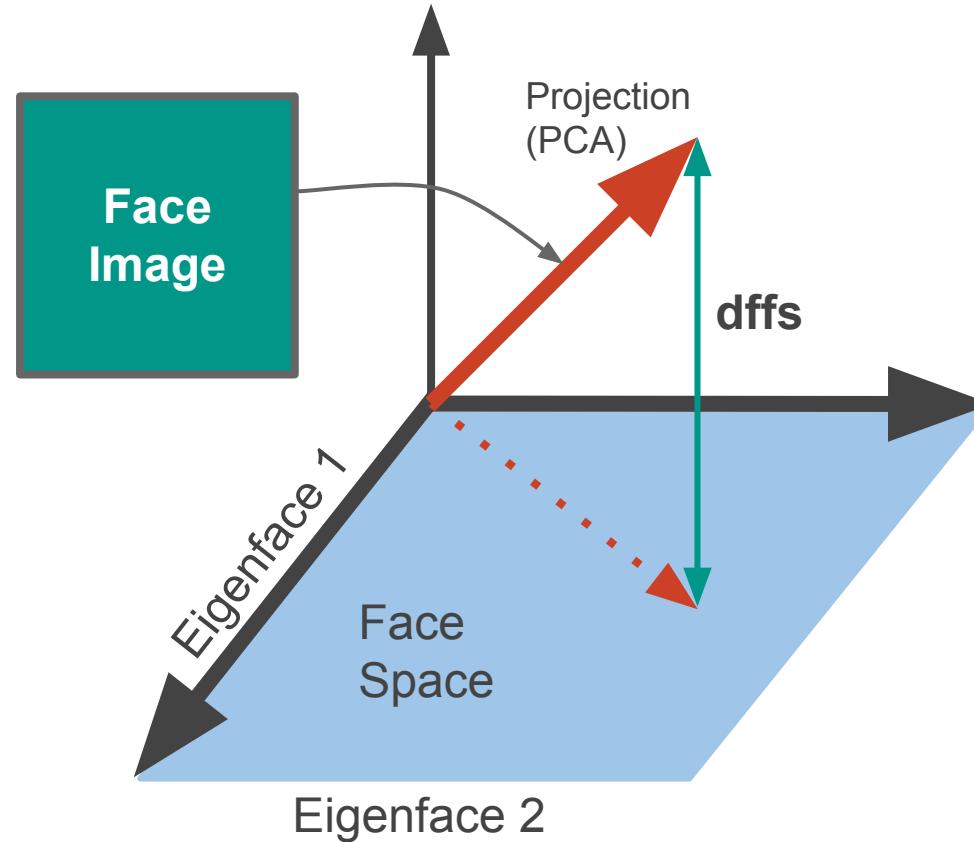
Step 2: compute $\hat{\Phi} = \sum_{i=1}^K w_i u_i$ ($w_i = u_i^T \Phi$)

Step 3: compute $e_d = \|\Phi - \hat{\Phi}\|$

Step 4: if $e_d < T_d$, then Γ is a face.

- The distance e_d is called distance from face space (dffs)

Face Detection Using Eigenfaces



Eigen Faces

Problems

- Background
- Lighting Conditions (Controlled lighting, Colour Norm..)
- Orientation (multiple enrolment at various orientations)
- Scale

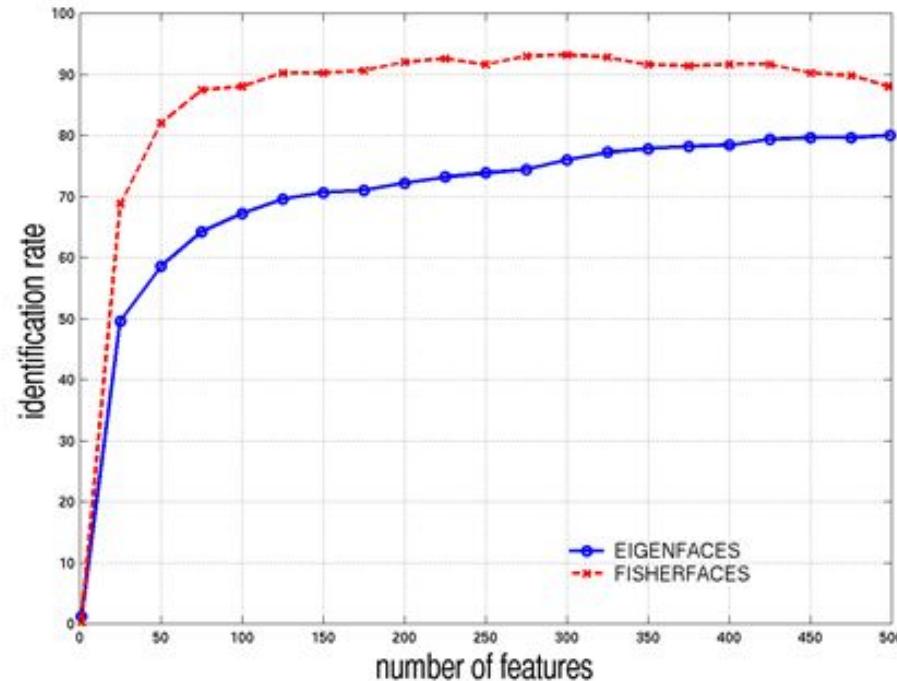
Fisher Face

FERET dataset

Best ID rate:

Eigenfaces 80.0%,

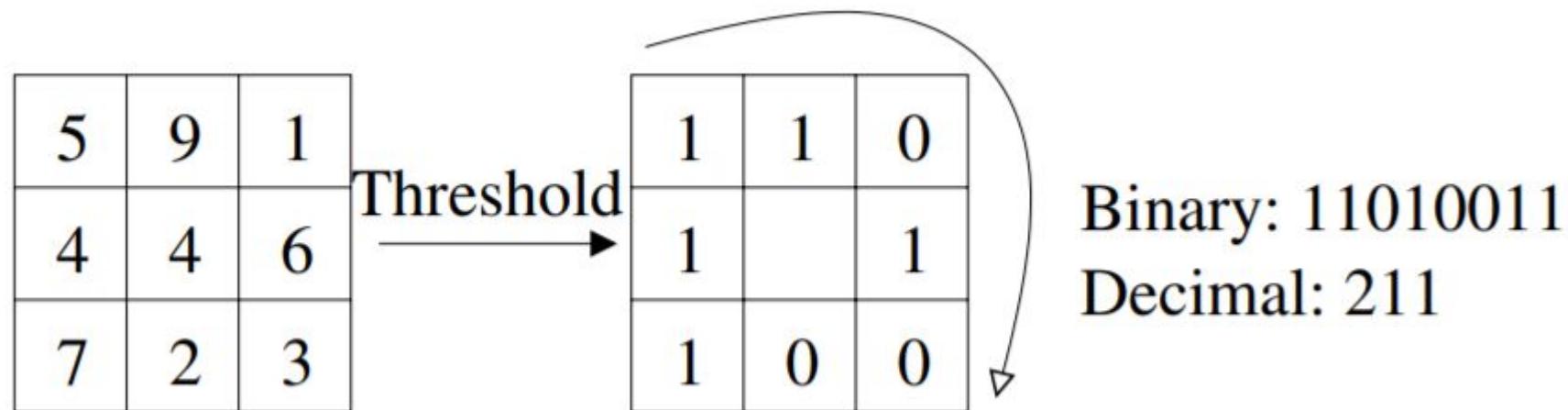
Fisherfaces 93.2%



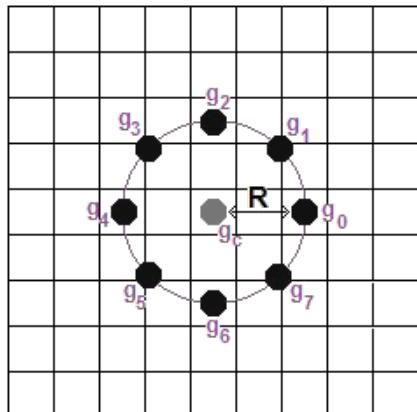
Why Face Recognition is a Nonlinear Problem?

- **Pose (Out-of-Plane Rotation):** frontal, 45 degree, profile, upside down
- **Presence or absence of structural components:** beards, mustaches, and glasses
- **Facial expression:** face appearance is directly affected by a person's facial expression
- **Occlusion:** faces may be partially occluded by other objects
- **Orientation (In-Plane Rotation):** face appearance directly vary for different rotations about the camera's optical axis
- **Imaging conditions:** lighting (spectra, source distribution and intensity) and camera characteristics (sensor response, gain control, lenses), resolution

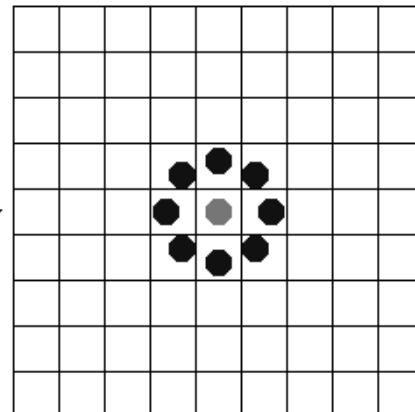
Applying the LBP Operation



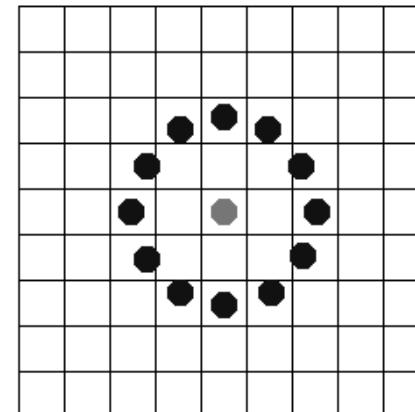
Circular LBP



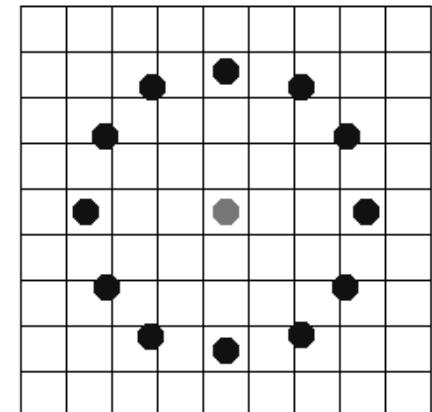
P=8, R=2
(a)



P=8, R=1
(b)



P=12, R=2
(c)



P=12, R=3
(d)

Face Recognition with LBP

Method	fb	fc	dup I	dup II	lower	mean	upper
LBP, weighted	0.97	0.79	0.66	0.64	0.76	0.81	0.85
LBP, nonweighted	0.93	0.51	0.61	0.50	0.71	0.76	0.81
PCA, MahCosine	0.85	0.65	0.44	0.22	0.66	0.72	0.78
Bayesian, MAP	0.82	0.37	0.52	0.32	0.67	0.72	0.78
EBGM_Optimal	0.90	0.42	0.46	0.24	0.61	0.66	0.71

SIFT

How does it works?

What are its advantages over linear representations?

What are its limitations

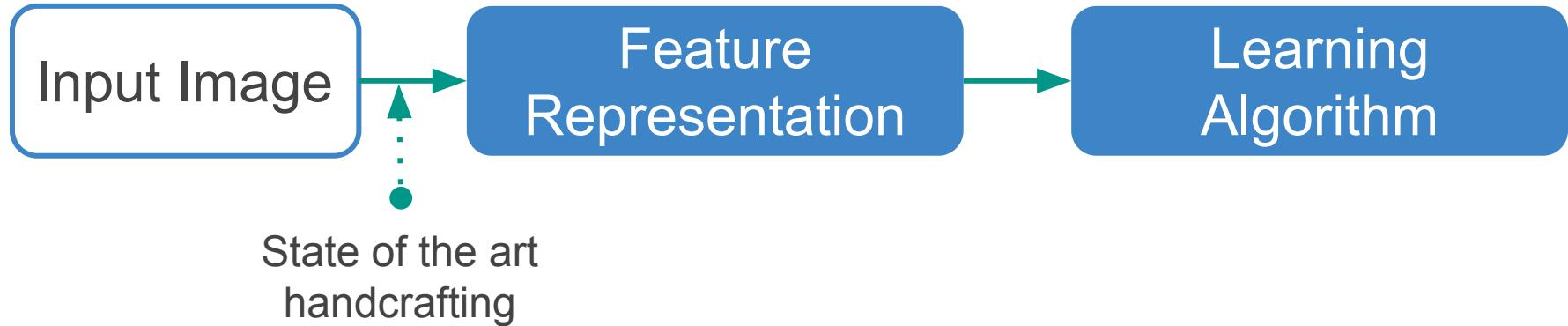
Fisher Vector

How does it works?

What are its advantages over linear representations?

What are it's limitations

Feature Representations



- **Needs expert knowledge**
- **Time expensive hand tuning**
- **One of the limiting factors of Computer Vision Systems**

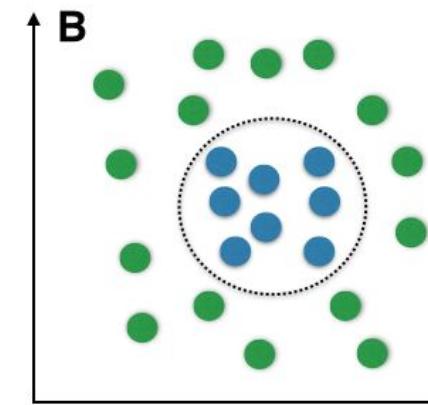
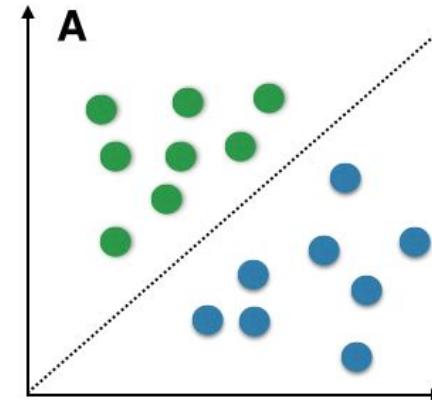
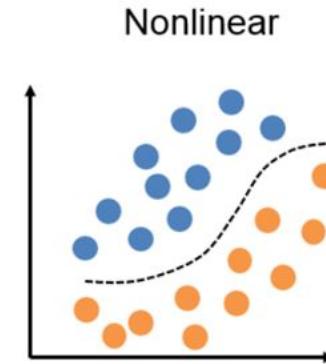
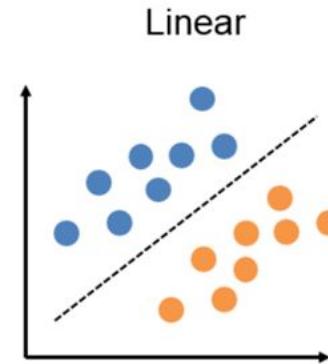
Face Representation Methods: Conventional to Modern

PCA

Metric Learning

CNN

Linearity and Non Linearity



One shot learning

Learning from one example to recognize the person again.

CNN with softmax output - certain caveats here for one shot learning

Learning a similarity function:

$d(\text{img1}, \text{img2})$ = degree of differences between images

The Viola and Jones method

The most famous method

Training may need weeks

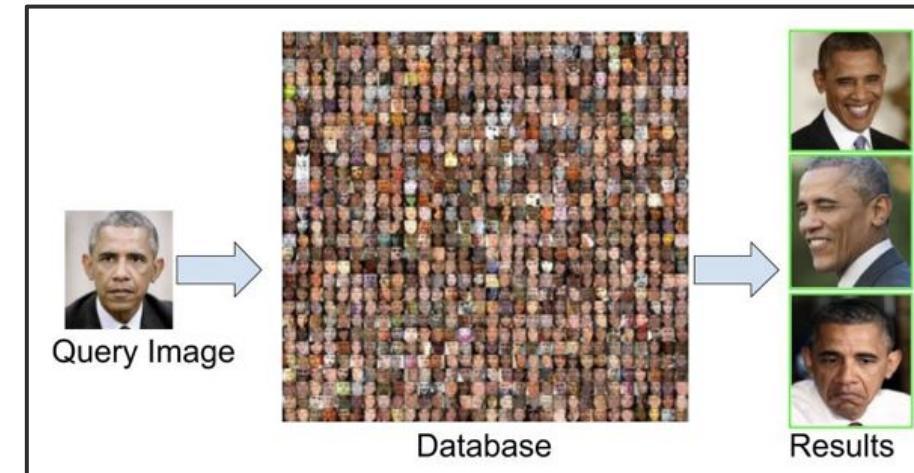
Recognition is very fast, e.g. real-time for digital cameras.

Techniques

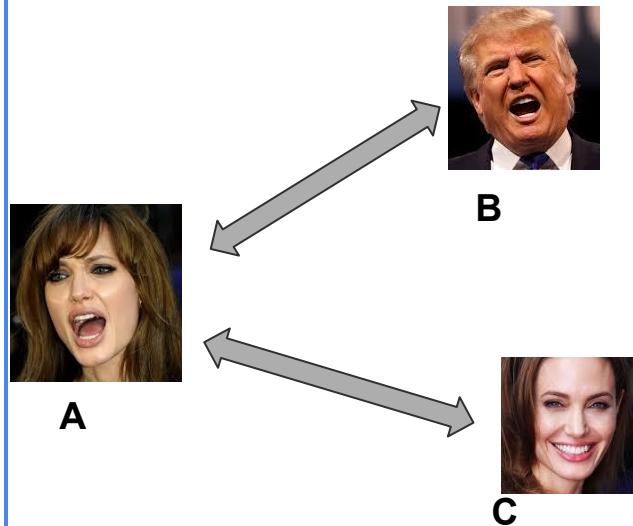
- Integral image for feature extraction
- Ada-Boost for face detection
- Attentional cascade for fast rejection of non-face sub-windows

Face retrieval

- Given a query image I_q , compute the ***similarity score*** between the query and the images in the database
- Rank them based on the ***sim_score***
- Retrieve **Top-K** ranked images

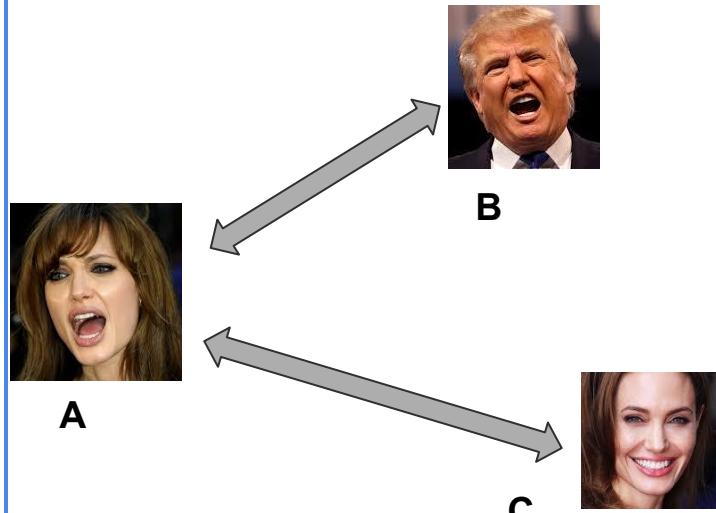


Similarity between the Faces



- ❑ Which pair is more similar?
 - ❑ (A, B)
 - ❑ (A, C)

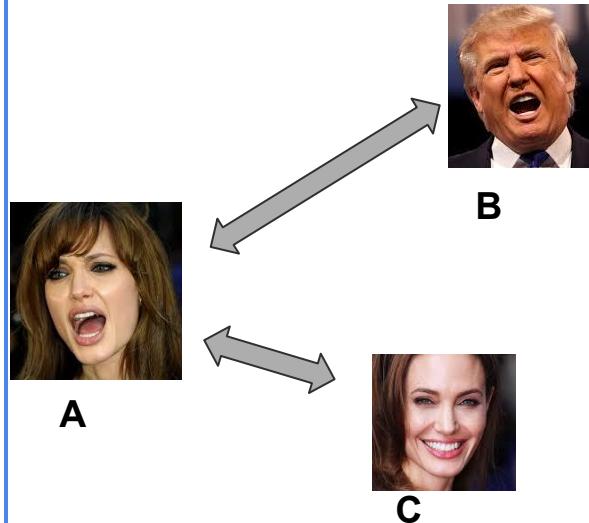
Similarity between the Faces



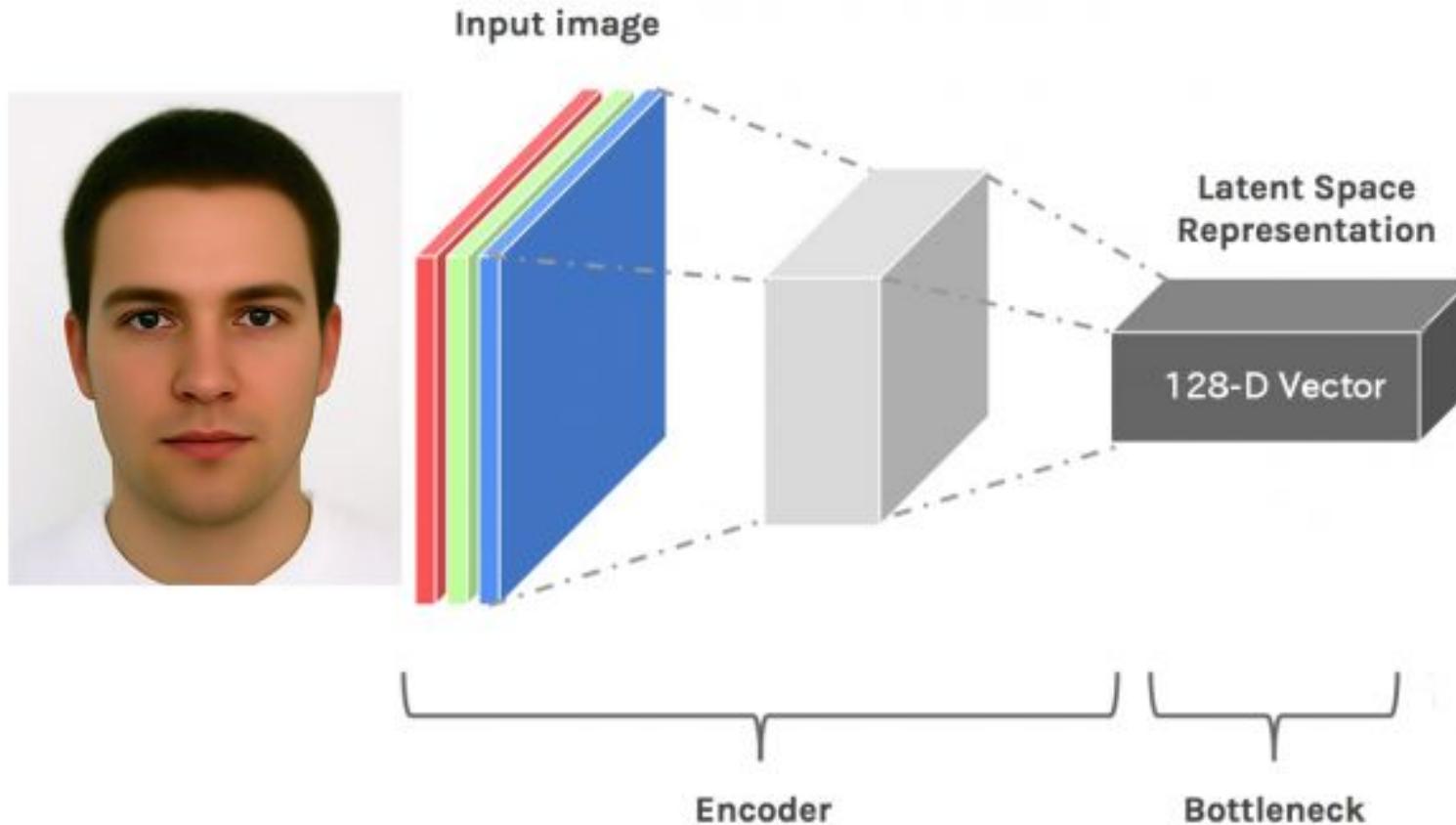
- When we care about expression

$$d_{(mood)}(A, B) < d_{mood}(A, C)$$

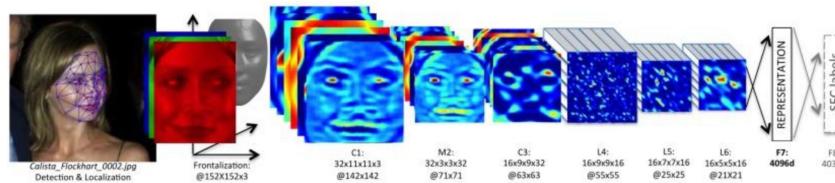
Similarity between the Faces



- ❑ When we care about identity
$$d_{(id)}(A, B) > d_{(id)}(A, C)$$
- ❑ Similarity between the images depends on the **the task we care about**
- ❑ Need a methodology to learn such metric



Deepface



- F8 calculates probability with softmax $p_k = \exp(o_k) / \sum_h \exp(o_h)$
- Cross-entropy loss function: $L = -\sum_k \log(p_k)$
- Computed using SGD and performs backpropagation

Experiments

- DeepFace was evaluated in LFW
- Human cropped: 97.5% vs Deepface 97.35%

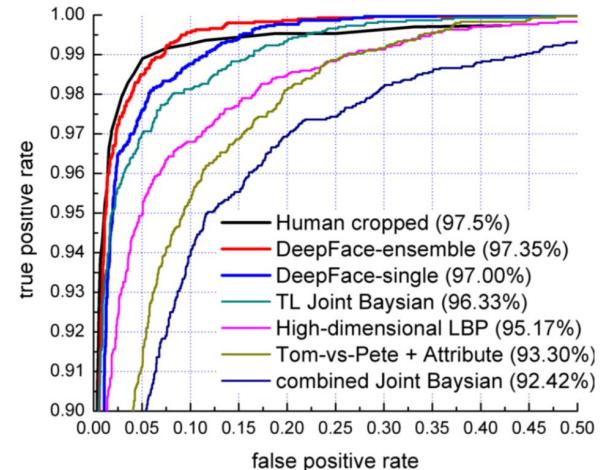
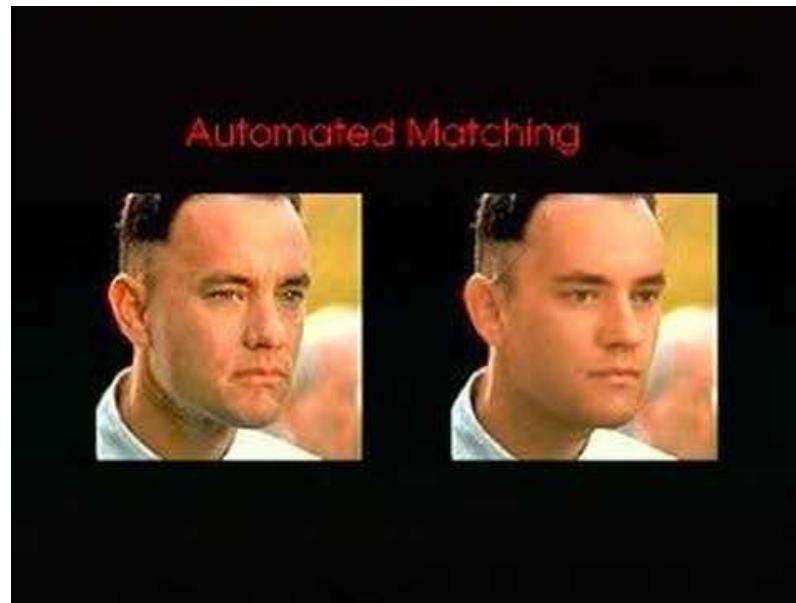


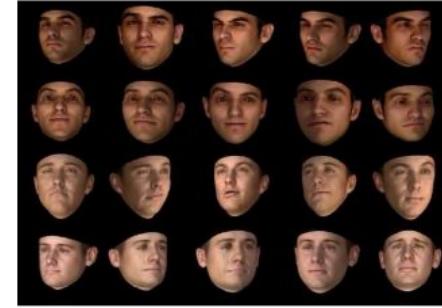
Fig. Roc Curve on LFW

3D Morphable Model



Synthetic Face Images by 3DMM

- + Easy to manipulate attributes such as identity, pose, expression, and lighting
- + Can generate millions of images with controlled attributes
- - Domain gap with real face images



Synthetic images generated by 3DMM

Problem Definition

- Generating photorealistic face images 3DMM rendered faces of new identities with arbitrary poses, expressions, and illuminations

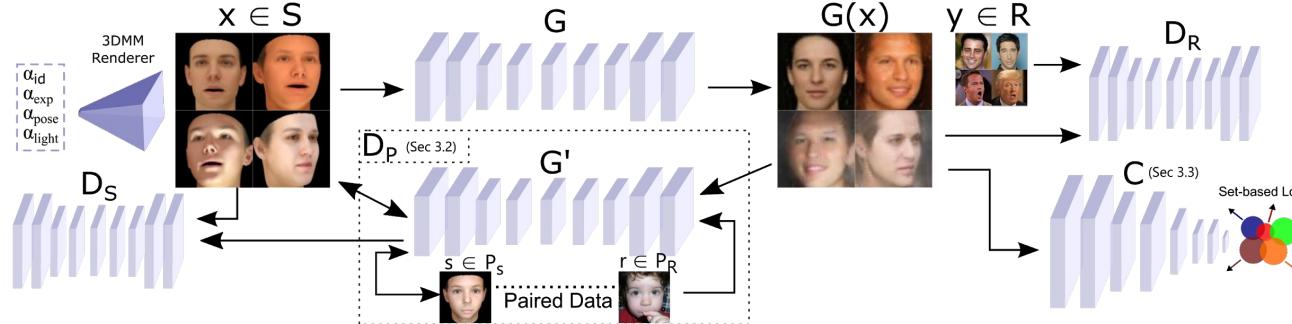


Problem Definition

- Generating photorealistic face images 3DMM rendered faces of new identities with arbitrary poses, expressions, and illuminations
- We formulate this problem as domain adaptation problem/ style transfer problem (3DMM -> Real)
 - *Pixel2Pixel (Isola et al 2017)*
 - *CycleGAN(Zhu et al 2017)*
- How can we benefit small amount of paired data in unsupervised style transfer GAN?
- How to prove identity consistency of generated images?

Photorealistic identity synthesis

(Gecer, Bhattacharai, Kittler, and Kim *ECCV' 2018*)



- ❖ Randomly generated 3DMM images with random pose, expression and lighting attributes for the new IDs.
- ❖ Unsupervised training with forward cycle consistency.
- ❖ Adversarial Pair Matching network G' by the help of a limited number of paired data.
- ❖ ID preservation by a set-based supervision through a pre-trained classification network C .

Experiments (Quantitative)

VGG(%100)	1.8M	-	96×96	94.8
VGG(%100) + GANFaces-500K	1.8M	500K	96×96	94.9
VGG(%100) + GANFaces-5M	1.8M	5M	96×96	95.2

Tab. Verification accuracy on LFW benchmark

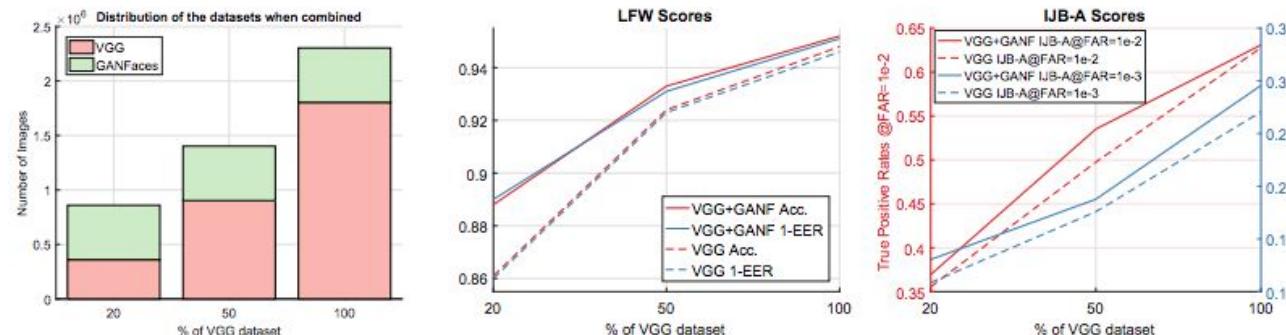


Fig. Verification on LFW and IJB-A database with different size of original and synthetic data

Experiments (Qualitative)

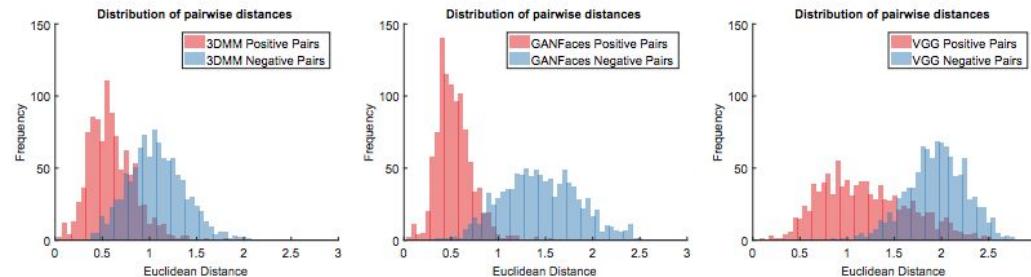


Fig. Face pairs euclidean distance distribution

Experiments (Qualitative)



Fig. Face generated by the proposed method conditioned on 3DMM identity, expression and pose parameters

Experiments (Qualitative)



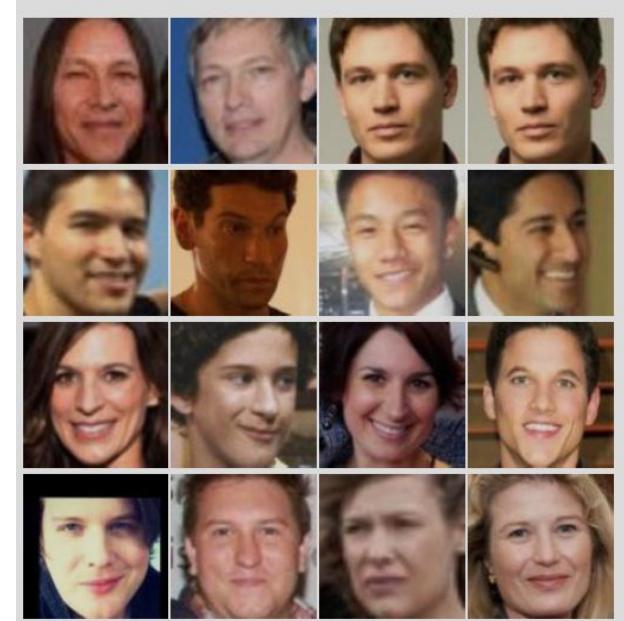
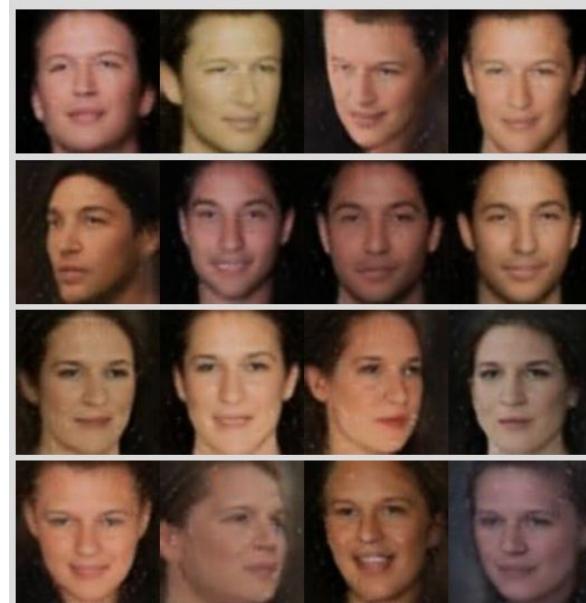
- ❖ Interpolation in identity space
- ❖ Smooth transition from one identity space to another identity space shows that manifold of image generator is smooth.

Experiments (illumination preservation)



Experiments(Qualitative)

- ❖ The nearest images from the training set in terms of identity features for the images
- ❖ Variation in the nearest images shows diversity of GANFaces in the embedding space while they bear similar higher order attributes such as gender, shape of face etc.



Comparison with existing methods

A) 3DMM



B) CycleGAN



C) Our approach



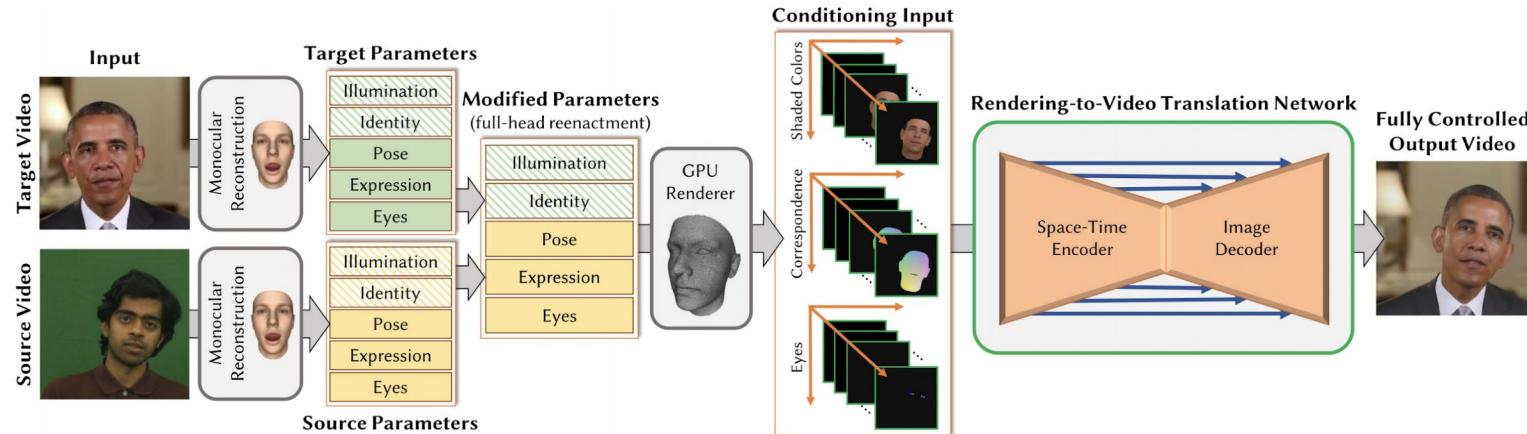
Deep video portrait for video translation

Interactive Editing



2x Speed

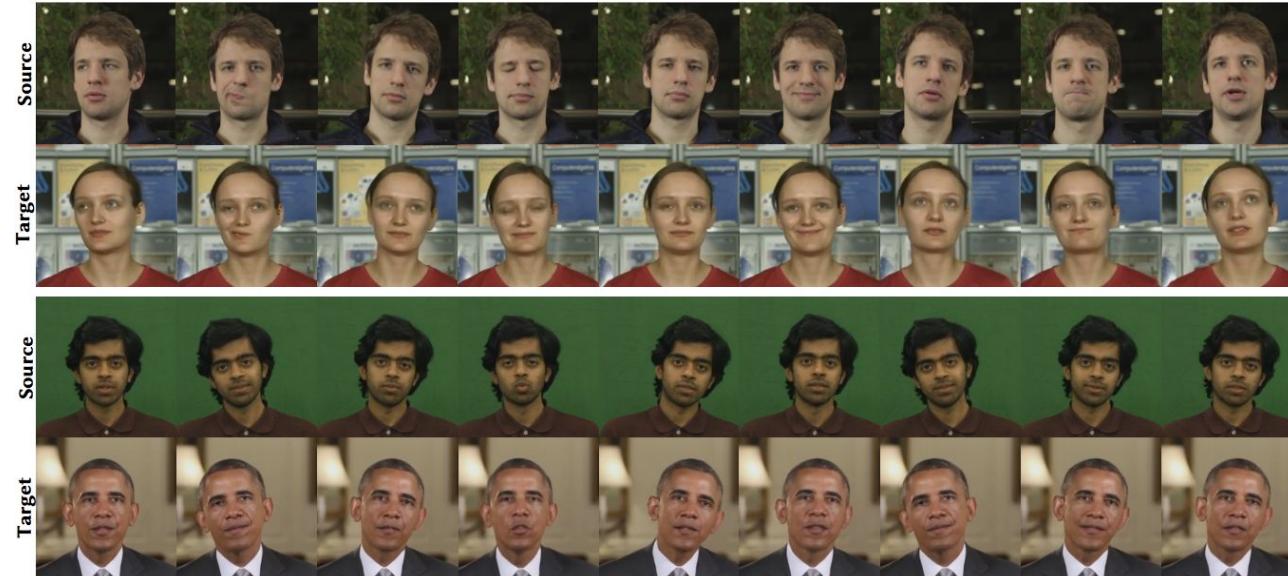
Deep video portrait for video translation



Kim, Hyeongwoo, et al. "Deep Video Portraits." SIGGRAPH 2018

APA

Qualitative results



Native Dubbing



PCA

- ❑ **Merits**
 - ❑ Removes redundancies and noise to make feature more discriminative
 - ❑ Transform the representations into compact form (dimensionality reduction)
 - ❑ Does not require label information
- ❑ **Demerits**
 - ❑ Does not embed category information
 - ❑ Imagine large variations is due to illuminations due external sources (not necessarily the axis with maximum variance contains discriminative features)
 - ❑ Sub-optimal for specific task(s)

Properties of pseudo-distance

1. $d_M(x, x') \geq 0$ (nonnegativity),
2. $d_M(x, x) = 0$ (identity),
3. $d_M(x, x') = d(x', x)$ (symmetry),
4. $d_M(x, x'') \leq d(x, x') + d(x', x'')$ (triangle inequality).

Additional Applications

Security



Surveillance



Robotics



All these applications are face matching tasks

- Euclidean or L2 distance is probably the most well known

$$d_{L_2}(x, y) = (x - y)^T (x - y)$$

- Most common form of learned metrics are Mahalanobis

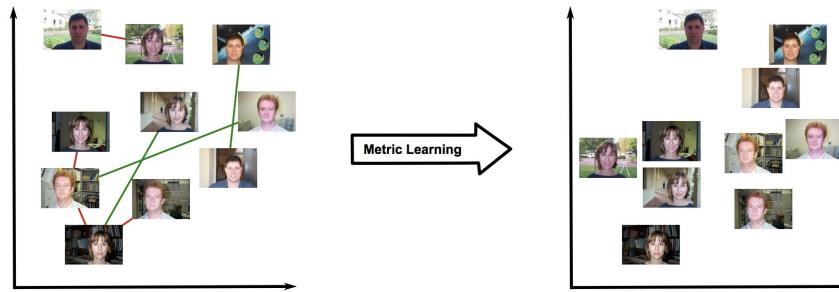
$$d_M(x, y) = (x - y)^T M(x - y)$$

- M is a positive definite matrix
- Generalization of Euclidean metric (setting M=I)
- Corresponds to Euclidean metric after linear projection of data

$$d_M(x, y) = (x - y)^T M(x - y) = (x - y)^T L^T L(x - y) = d_{L_2}(Lx, Ly)$$

- Not all methods fit this formulation of fixed vectorial data representation, eg based on matching image regions [Nowak & Jurie 2007]

Metric Learning for face recognition



$$d_M^2(x_i, x_j) = (x_i - x_j)^\top M(x_i, x_j)$$

$$M = L^\top L; L \in \mathbb{R}^{d \times D}; d \ll D$$

$$d_L^2(x_i, x_j) = \|Lx_i - Lx_j\|^2$$

- Pairwise constraints (same person vs different persons) are imposed
- Projection matrix is learned to better satisfy the imposed constraints

Recap (Image representation)

- ❑ Pixel Intensity
 - ❑ Not robust to scaling, illuminations, rotations etc
- ❑ SIFT
 - ❑ invariant to scaling, rotation and translations and partially illuminations

References:

1. Eigen Faces:
 - a. http://www.vision.jhu.edu/teaching/vision08/Handouts/case_study_pca1.pdf
 - b. <https://www.cin.ufpe.br/~rps/Artigos/Face%20Recognition%20Using%20Eigenfaces.pdf>
 - c.
2. Fisher Face:
 - a. <https://apps.dtic.mil/dtic/tr/fulltext/u2/1015508.pdf>

3. Local Binary Patterns:

- a. http://www.ee.oulu.fi/mvg/files/pdf/pdf_494.pdf

4. SIFT

- a. Aly, Mohamed. "Face recognition using SIFT features." *CNS/Bi/EE report* 186 (2006).

5. Fisher vector:

- a. https://www.robots.ox.ac.uk/~vgg/publications/2013/Simonyan13/extras/simonyan13_ext.pdf

Dimensionality Reduction:

- a. <https://hal.archives-ouvertes.fr/inria-00439290/>
- b. https://jurie.users.greyc.fr/papers/12_cvpr_ldca.pdf
- c. [https://zpascal.net/cvpr2016/Bhattarai CP-mtML Coupled Projection CVP R 2016 paper.pdf](https://zpascal.net/cvpr2016/Bhattarai_CP-mtML_Coupled_Projection_CVP_R_2016_paper.pdf)
- d.

CNN

1. <https://research.fb.com/publications/deepface-closing-the-gap-to-human-level-performance-in-face-verification/>

Autoencoders:

<https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7962256>

References

1. Ojala, Timo, Matti Pietikäinen, and Topi Mäenpää. "Gray scale and rotation invariant texture classification with local binary patterns." *European Conference on Computer Vision*. Springer, Berlin, Heidelberg, 2000.
2. Ahonen, Timo, Abdenour Hadid, and Matti Pietikainen. "Face description with local binary patterns: Application to face recognition." *IEEE Transactions on Pattern Analysis & Machine Intelligence* 12 (2006): 2037-2041.