IMAGE COMPRESSION AND FACE RECOGNITION: TWO IMAGE PROCESSING APPLICATIONS OF PRINCIPAL COMPONENT ANALYSIS

Aleš Hladnik

Chair of Information and Graphic Arts Technology

Department of Textiles

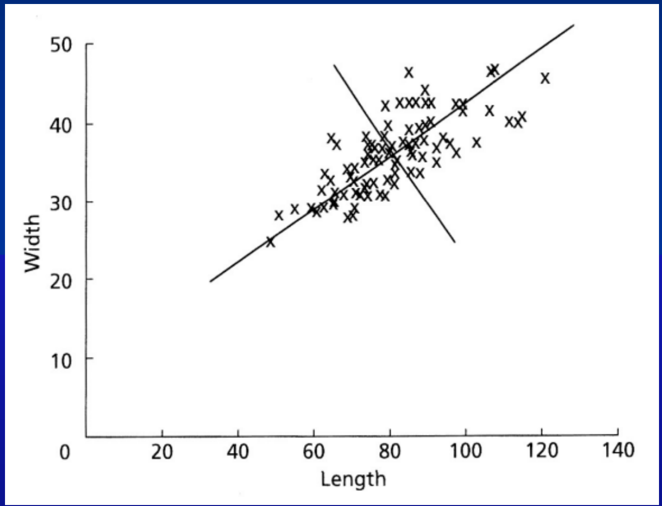
Faculty of Natural Sciences and Engineering

University of Ljubljana, Slovenia

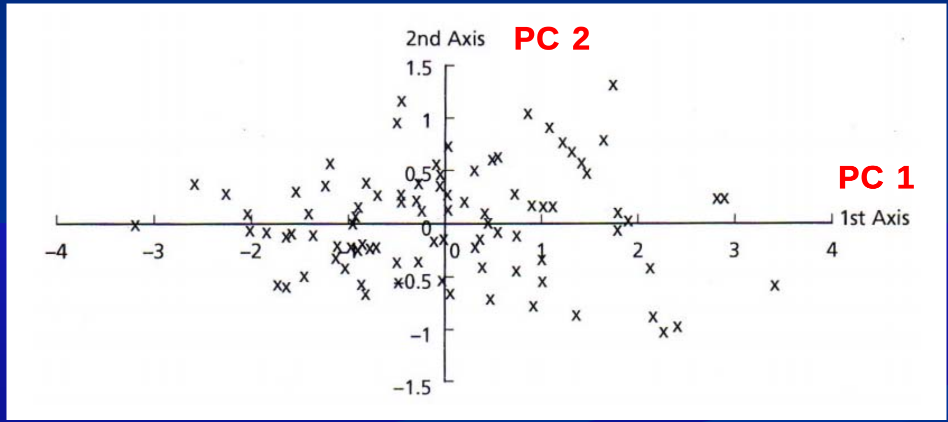
international circle of educational institutes for graphic arts: technology and management

44th Conference of the International Circle of Educational Institutes for Graphic Arts Technology and Management,
19 – 22 June 2012, Budapest, Hungary
Óbuda University
Rejtő Sándor Faculty of Light Industry and Environmental Engineering

What is Principal component analysis (PCA) ?

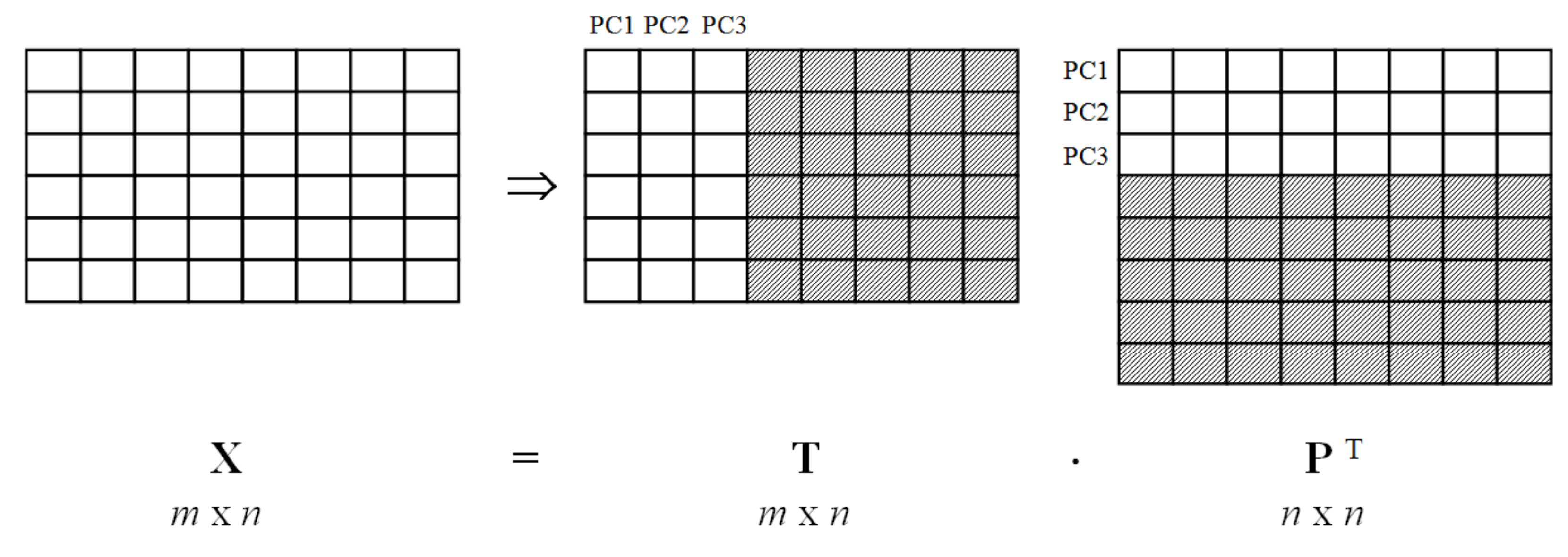


PCA



- PC1 is calculated such that it accounts for (= explains) the largest possible variance in the dataset
- PC2 accounts for the largest remaining variance, etc.
- PCs are linear combinations of original variables
- PCs are uncorrelated (perpendicular, orthogonal) to each other

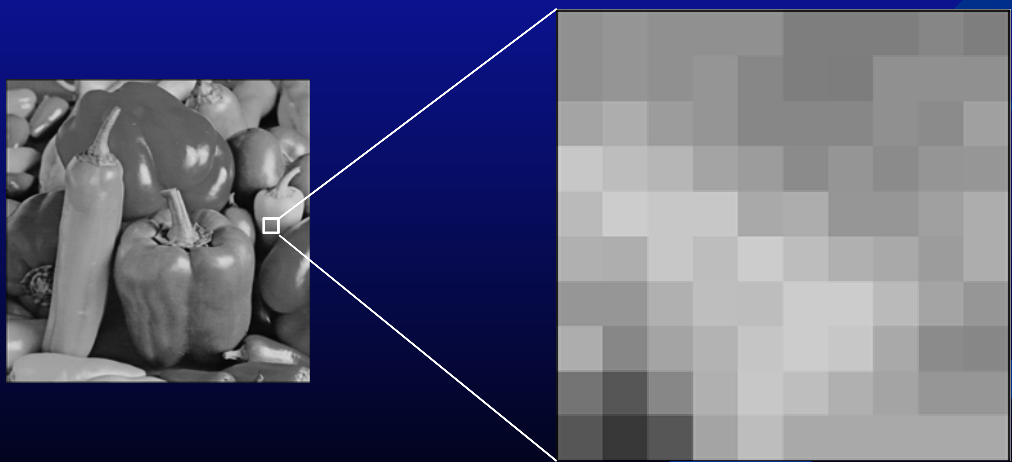
PCA as a dimensionality reduction tool



- Number of PCs equals the number of original variables, but higher PCs mainly explain data noise
- T: scores → relationships among samples
- P: loadings → (dis)similarities among original variables

Digital image

- 2D function of spatial coordinates *x* and *y*: ***I = f(x, y)***
- *I* = intensity (gray level in GS images)
- *I*, *x* and *y* are discrete quantities
- RGB color images consist of three 2D image planes: R, G and B



144	149	144	144	144	126	126	126	134	126
144	149	144	149	135	126	125	144	144	144
164	173	156	149	135	135	135	144	139	160
199	189	182	164	156	139	149	139	149	150
186	204	199	199	169	173	150	150	160	173
176	174	199	189	204	189	176	169	156	173
150	150	176	190	189	204	204	186	164	150
173	135	164	179	197	204	199	169	139	135
116	86	135	176	199	190	176	164	150	150
86	56	86	164	189	169	169	169	169	169

8-bit GS image: $0 \leq I \leq 255$

Image compression

- Ideal image compression method should remove all redundant and/or irrelevant information while preserving the **important** information
- In practice, both non-redundant and relevant information are discarded to achieve a higher compression ratio (CR) and reduced file size
- CR = size of original image / size of compressed image
- Lossless (reversible) compression
  - Reconstructed image – after compression and decompression – is identical to the original
  - Examples: LZW, LZ77 implemented in GIF, PNG, TIFF file formats
  - Typical CR ~ 2:1 for natural scenes, somewhat higher for document images

Lossy compression

- Lossy (irreversible) compression
  - Reconstructed image contains degradations with respect to the original
  - Examples: JPEG, JPEG 2000
  - Trade-off between image quality and storage size
  - Much higher CR than with lossless compression

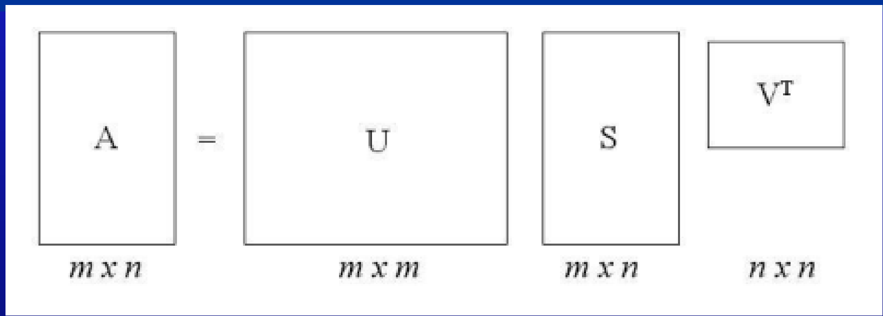
CR = 2.6 : 1
File size = 83261 bytes

CR = 23 : 1
File size = 9553 bytes

CR = 144 : 1
File size = 1523 bytes

Lossy compression using PCA

- Singular value decomposition (SVD)
  - A linear algebra tool related to PCA



$$S = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_r & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \sigma_{r+1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & \sigma_n \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix}$$

A = U·S·V^T

A … mean-centered digital image matrix
U … orthogonal matrix containing left singular vectors
V … orthogonal matrix containing right singular vectors

S … diagonal matrix with singular values (SV) on its diagonal

To compress image A, we retain only the first few SV's ("Principal components")

$$U = [\mathrm{u}_1, \mathrm{u}_2, \ldots \mathrm{u}_r, \mathrm{u}_{r+1}, \ldots, \mathrm{u}_m]$$

$$V = [\mathrm{v}_1, \mathrm{v}_2, \ldots \mathrm{v}_r, \mathrm{v}_{r+1}, \ldots, \mathrm{v}_n]$$

Image processing workflow

Scene → Canon EOS 400D → 24-bit RGB image (3648 x 2736 pix) → $I = 0.30R + 0.59G + 0.11B$ → 8-bit GS image

Cropping

Downsampling

1024 x 1024

512 x 512

256 x 256

SVD / PCA → Lossy compressed image → Image quality assessment → SSIM PSNR CR

Hedgehog

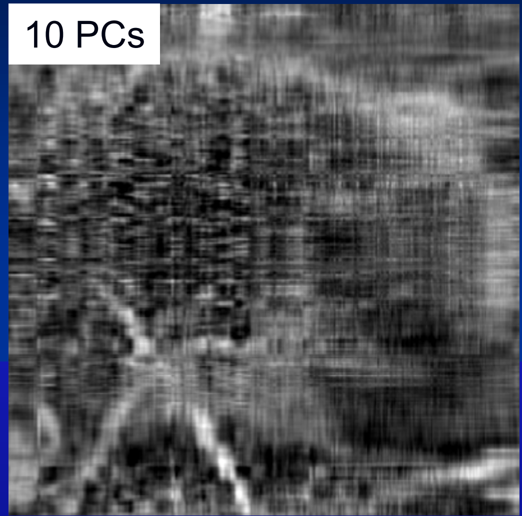
Man

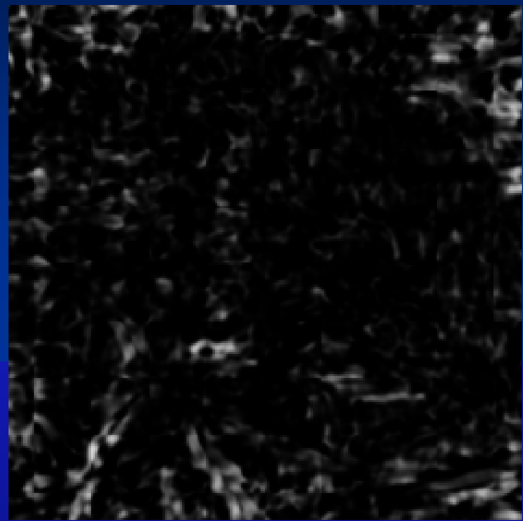
Buildings

Results

SSIM map

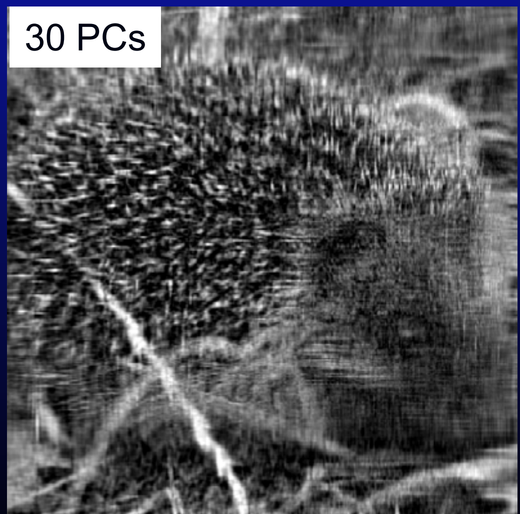
10 PCs

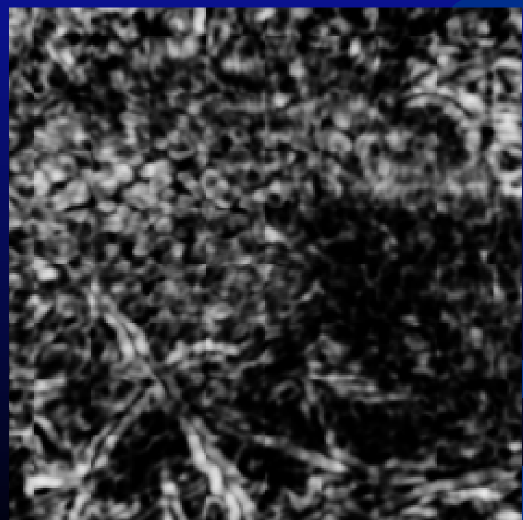




CR = 48.8:1 SSIM = 0.34 PSNR = 14.6

30 PCs





CR = 16.8:1 SSIM = 0.62 PSNR = 16.7

Hedgehog

1024 X 1024

CR = 1024/(2*PCs + 1)

Results (cont'd)

SSIM map

CR = 10.1:1 SSIM = 0.76 PSNR = 18.4

CR = 5.1:1 SSIM = 0.91 PSNR = 21.7

Hedgehog

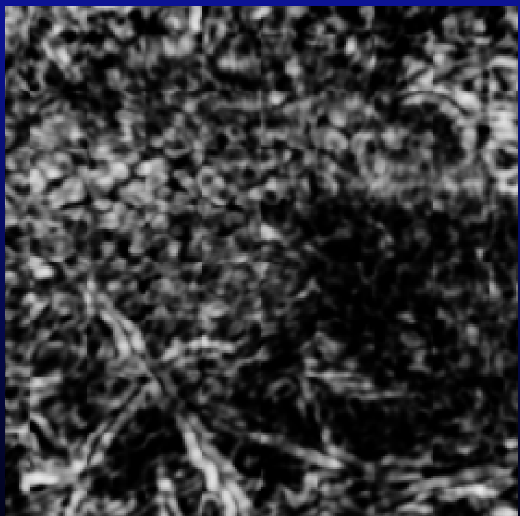
1024 X 1024

CR = 1024/(2*PCs + 1)

Results (cont‘d)

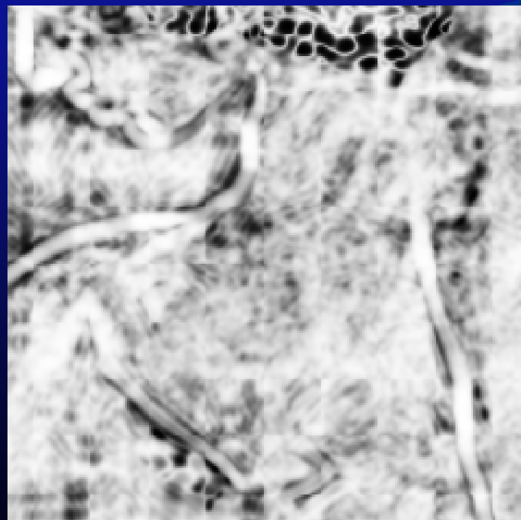
30 PCs

Hedgehog



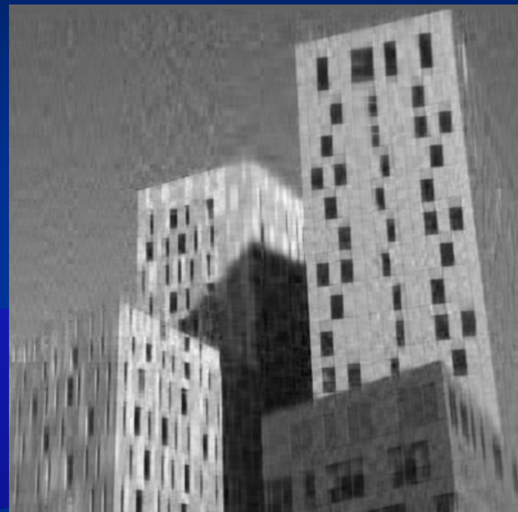
SSIM = 0.62

Man



SSIM = 0.92

Buildings



SSIM = 0.85

Face recognition

- Ability of a computer to scan, store, and recognize people by their facial characteristics
- Subfield of computer vision and of biometrics
  - Related to voice-, iris-, handwriting- and optical character recognition (OCR)

Areas	Applications
Information Security	Access security (OS, data bases) Data privacy (e.g. medical records) User authentication (trading, on line banking)
Access management	Secure access authentication (restricted facilities) Permission based systems Access log or audit trails
Biometrics	Person identification (national IDs, Passports, voter registrations, driver licenses) Automated identity verification (border controls)
Law Enforcement	Video surveillance Suspect identification Suspect tracking (investigation) Simulated aging Forensic Reconstruction of faces from remains
Personal security	Home video surveillance systems Expression interpretation (driver monitoring system)
Entertainment - Leisure	Home video game systems Photo camera applications

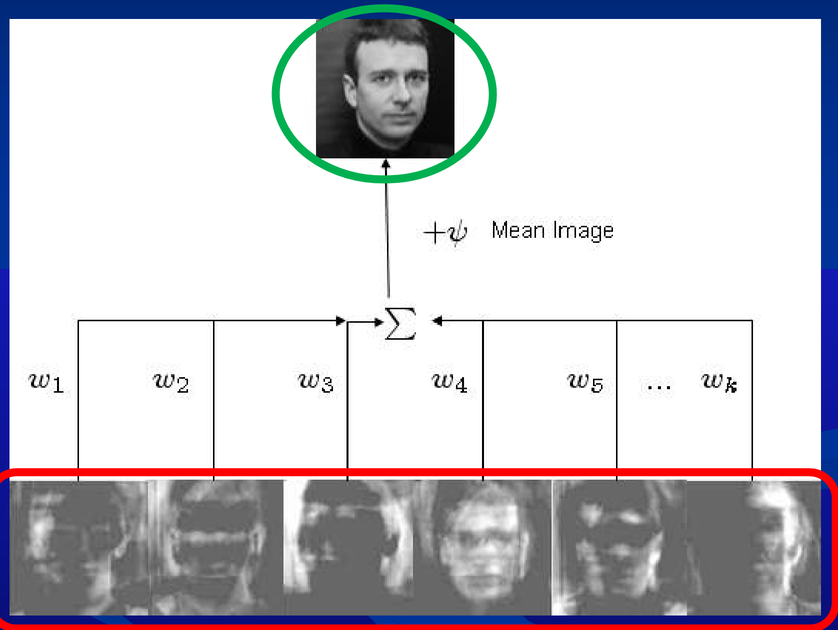
Face recognition algorithms

Image or video → Face detection → **Face recognition** | **Identification Classification**

- We know very little about how *human beings* recognize faces
- How do computers perform such a complex task?
- Various strategies
  - Model-based approach – use of different face models
  - Feature extraction
  - Texture-based approach
  - Template (pattern) matching
  - Artificial neural networks
  - **Data compression (PCA)**

PCA for face recognition: eigenfaces

- Eigenfaces (eigenimages) – a set of eigenvectors derived from the covariance matrix of the high-dimensional vector space of human faces
- Eigenfaces are obtained using PCA
- By appropriate weighted summation (= linear combination) of the eigenfaces and adding the mean face, each training database image can be reconstructed



$$\Phi_i = \sum_{j=1}^{K} w_j u_j$$

Φ_i … mean-subtracted i th face
w_j … j th weight
u_j … j th eigenvector

PCA for face recognition: eigenfaces (cont'd)

- After the eigenfaces have been computed, the second step depends on the application:
  - Identification – labels of individuals must be obtained
  - Recognition of a person where it must be decided if the individual has already been seen
  - Classification – a face must be assigned to a certain class
- If an unknown (= test) face is to be recognized, classification has to be made using some distance measure
  - Common distance metrics: Euclidean or Mahalanobis distance
- Drawback of the method: poorer recognition of faces when viewed with different levels of light or angles
  - Faces need to be seen from a frontal view under similar lighting

Simple example

- MATLAB algorithm based on the code written by A. H. Omidvarnia http://www.mathworks.com/matlabcentral/fileexchange/17032-pca-based-face-recognition-system
- Five step-procedure:
  1. Select training and test database paths.
  2. Select path of the test image.
  3. Run 'CreateDatabase' function to create 2D matrix of all training images.
  4. Run 'EigenfaceCore' function to produce basis of facespace.
  5. Run 'Recognition' function to get the name of equivalent image in training database.

References

- S. M. Holland: Principal Components Analysis (PCA). http://strata.uga.edu/software/pdf/pcaTutorial.pdf
- B. Brower: Image Compression Basics. http://www.lokminglui.com/CompressionTutorial.pdf
- L. Cao: Singular Value Decomposition Applied To Digital Image Processing. http://www.lokminglui.com/CaoSVDintro.pdf
- M. Richardson: Principal Component Analysis. http://people.maths.ox.ac.uk/richardsonm/SignalProcPCA.pdf
- I. Marques: Face Recognition Algorithms http://www.ehu.es/ccwintco/uploads/e/eb/PFC-IonMarques.pdf
- S. Trivedi: Face Recognition using Eigenfaces and Distance Classifiers - A Tutorial http://onionesquereality.wordpress.com/2009/02/11/face-recognition-using-eigenfaces-and-distance-classifiers-a-tutorial/