Face recognition

Axel Beauvisage, Carole Belloni, Adrian Kostkowski, Gareth Hughes, Domonkos Huszar

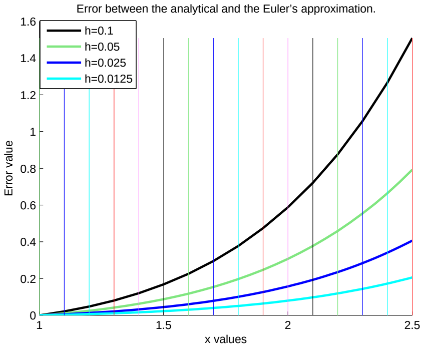

Fig. 1. Simulation Results

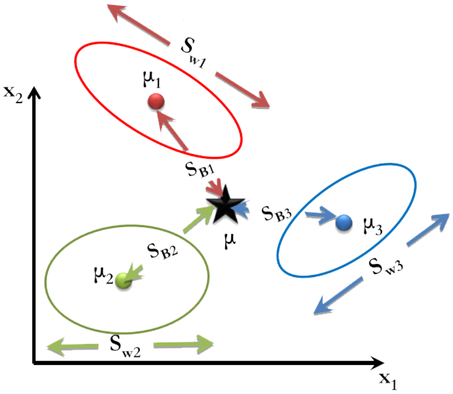

Fig. 2. Az LDA algoritmus ltal megvizsglt szrsok. The source of the image: [**?**]

***Abstract*—**

I. Introduction

THIS demo file is intended to serve as a "starter file" for IEEE journal papers produced under LaTeX using IEEEtran.cls version 1.7 and later. I wish you the best of success.

mds
January 11, 2007

A. Subsection Heading Here

Subsection text here.

1) Subsubsection Heading Here: Subsubsection text here.

II. Methods

A. Fisher method

Ha m dimenzis vektorokat szeretnk lekpezni egy C-dimenzis trbe akkor azt a kvetkez mdon tehetem meg.

$$y = w^T x, \text{ where} \tag{1}$$

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} w = \begin{pmatrix} w_{1,1} \dots w_{1,C} \\ \vdots \ddots \vdots \\ w_{m,1} \dots w_{m,C} \end{pmatrix} y = \begin{pmatrix} y_1 \\ \vdots \\ y_C \end{pmatrix} \tag{2}$$

B. Eigenfaces method

The Eigenfaces method is based on a principal components analysis from a training set of images. To get those, each normalized image is translated into a vector. The principal components will be then determined : They have the highest eigenvalue in the diagonalized correlation matrix (made out of the image and the mean values of the training set). After selecting the principals components (first eigenvectors called eigenfaces in this case), each image can be then be approximated with those weighed eigenfaces :

$$x \approx \hat{x} = \bar{x} + E_x * W, \text{ where} \tag{3}$$

x = vector representing the image,
$\hat{x}$ = approximated vector,
$\bar{x}$ = mean values of the vector during training,
E_x = Matrix of eigenfaces (principal components),
W = Matrix of weights of eigenfaces

W is particular to each image and is calculated to minimize the distance between x and $\bar{x}$:

$$W = E_x^T(x - \bar{x}) \tag{4}$$

The face recognized will be then the one in the training having its weight matrix W the closest to the weight matrix of the current face analyzed.

C. Local Binary Pattern Histogram

The local binary pattern histogram method uses the local feature of the face. First, the image is sliced into squares. To avoid problems of scale, The neighbours of a pixel analysed will be the ones on a circle with a varying radius and this pixel as center. The intensity of the central pixel is then compared to the ones of its neighbours. (higher values become 1, lower ones become 0). We then compute the new value of the central pixel, being the sum of powers of two combined with the value of neighbor taken clockwise. The point is to spot particular patterns of neighbor such as edges, lines, corner, flat areas.

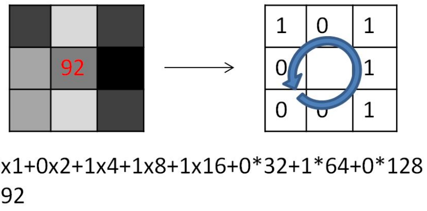

Fig. 3. Analysing the local structure

The histograms are then added (and not merged) for each part of the image. The resulting vector is then compared to the vectors obtained during the training. A fixed number of the closest vectors to the resulting vector is chosen and the person assigned to the greater number of these vector will be associated with the initial image.

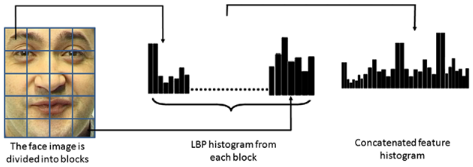
The face image is divided into blocks
LBP histogram from each block
Concatenated feature histogram

Fig. 4. Adding histograms (source : http://what-when-how.com/face-recognition/)

III. Implementation

The implementation goes here.

IV. Results

The result goes here.

V. Conclusion

The conclusion goes here.

Appendix A
Important code parts

Appendix two text goes here.