# Face Analysis and Recognition in Mobile Devices

Mauricio Villegas Santamaría

Máster en Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital
Departamento de Sistemas Informáticos y Computación
Universidad Politécnica de Valencia

Valencia, 5th of December 2008

# Outline


1. Introduction
2. Face Recognition in Mobile Devices
3. Results
4. Demonstration Applications
5. Conclusions

# Introduction

## Motivation

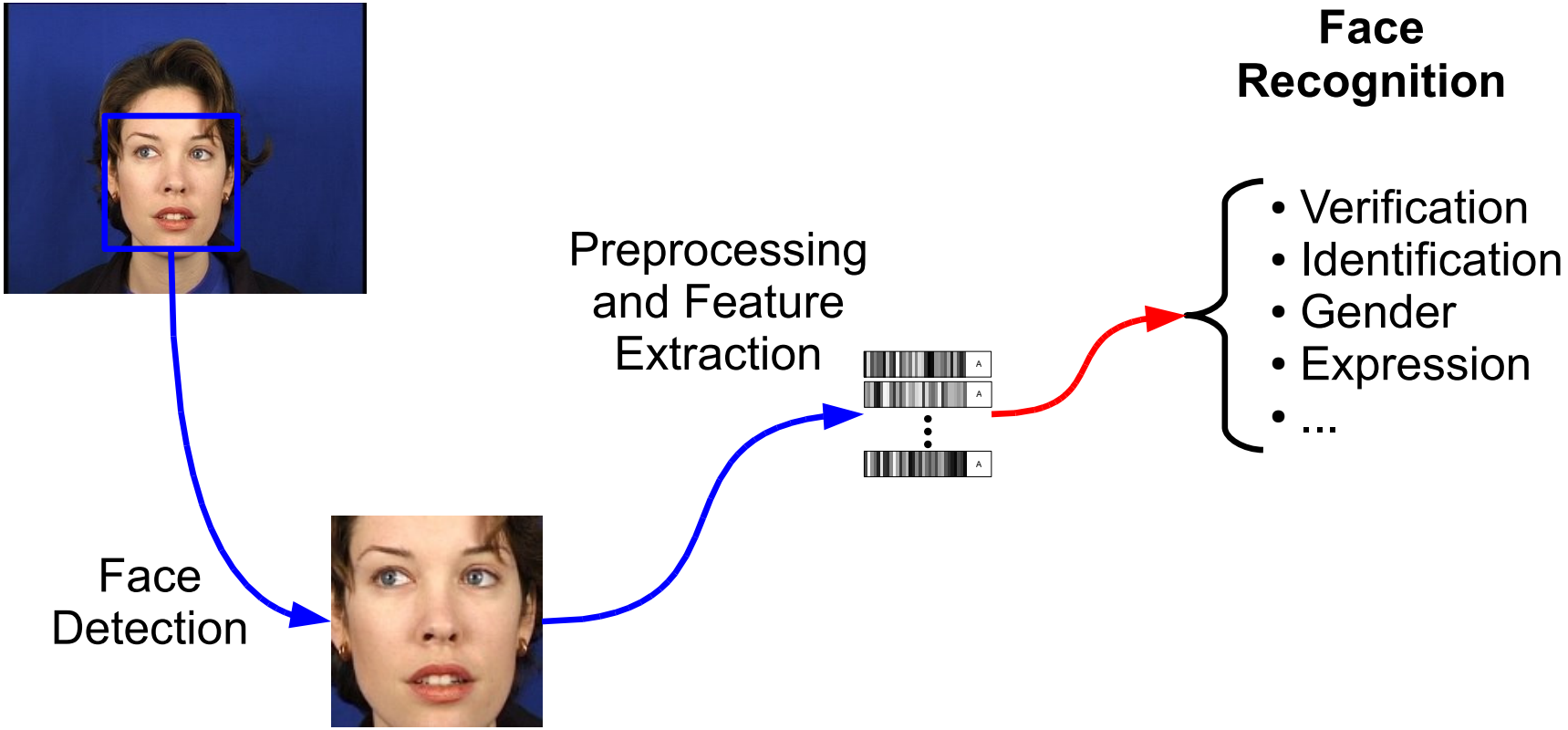
- Biometrics is the study of methods for recognising humans based on some intrinsic physical or behavioural trait.
- These methods have proved to be very useful in the current society.
- Nowadays mobile phones and PDAs with an integrated camera are common.
- This gives the opportunity to develop face recognition applications for these devices.

# Introduction

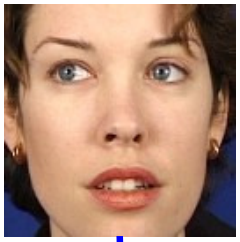
## Objectives of the work

- Develop a face verification application for commercial mobile devices.
- Address the difficulties encountered in mobile devices:
  - Low quality cameras.
  - Slow processors.
  - No native floating point arithmetic.
  - Small memory capacity.
- Perform an empirical analysis of the recognition algorithms under these conditions.

# What is Face Recognition?



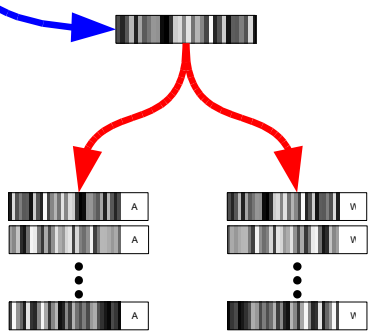
# Eigenfaces and Fisherfaces for Face Verification



Illumination Normalisation (LHE)

Subspace Projection

- PCA = eigenfaces
- PCA+LDA = fisherfaces



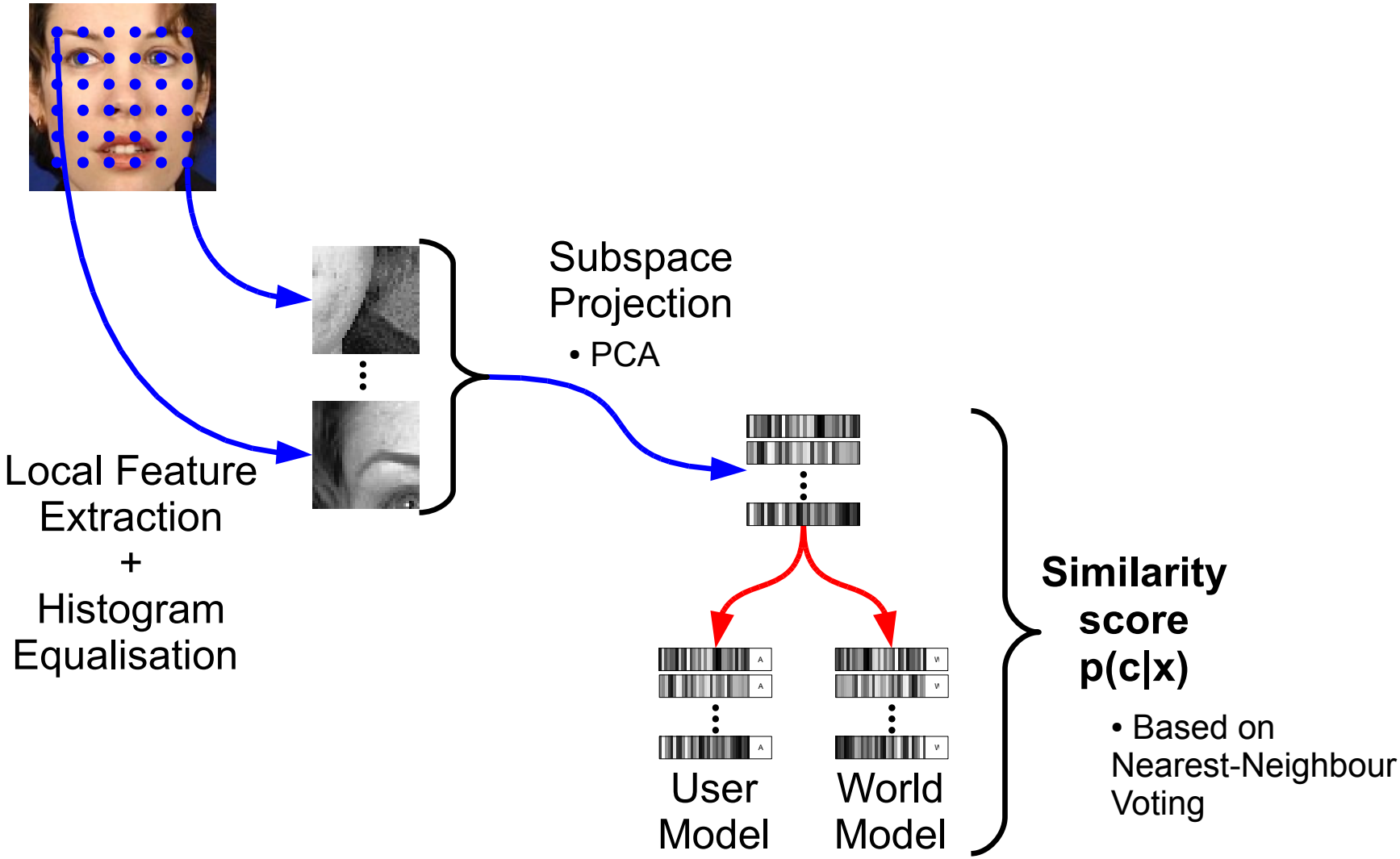
User Model

World Model

**Similarity score p(c|x)**

- Based on Nearest-Neighbour Distances

# Local Features for Face Verification



# Learning Discriminative Projections and Prototypes

- LDPP is a classifier learning algorithm we proposed [Villegas and Paredes, CVPR 2008].
- Learns a **projection base** and a set of **prototypes** for nearest neighbour classification.
- Specially designed for image recognition, overcoming the curse of dimensionality.
- Directly minimises the training set classification error.

$$J(B,P) = \frac{1}{N} \sum_{\forall \mathbf{x} \in X} S_\beta \left( \frac{d(\mathbf{y}, \mathbf{q}^{=})}{d(\mathbf{y}, \mathbf{q}^{\neq})} \right)$$

- In this work LDPP was used for face detection and gender recognition.

# Algorithm Optimisations

The algorithms have two things in common: 1) a linear projection, and 2) a nearest neighbour search using the euclidean distance.

1. Do the projection using integer arithmetic and represent the vectors as 8-bit integers.
   - Integer arithmetic is faster
   - The trained models are smaller which requires less memory
   - More data fits in the cache making the process faster

$$\mathbf{y}' = \frac{\mathbf{B}^T(\mathbf{x} - \mu)}{2^m} \approx \mathit{fitrange}\left[\frac{\mathit{round}(2^n \mathbf{B}_d^T)(\mathbf{x} - \mu)}{2^{n+m}}\right]$$

2. Do the nearest neighbour search using a *kd*-tree structure.
   - The approximate nearest neighbour search algorithms can be extended to discrete data

# Results

## Effect of Optimisations on Recognition Performance

- The approximations using 8-bit integers does not affect significantly the recognition results.

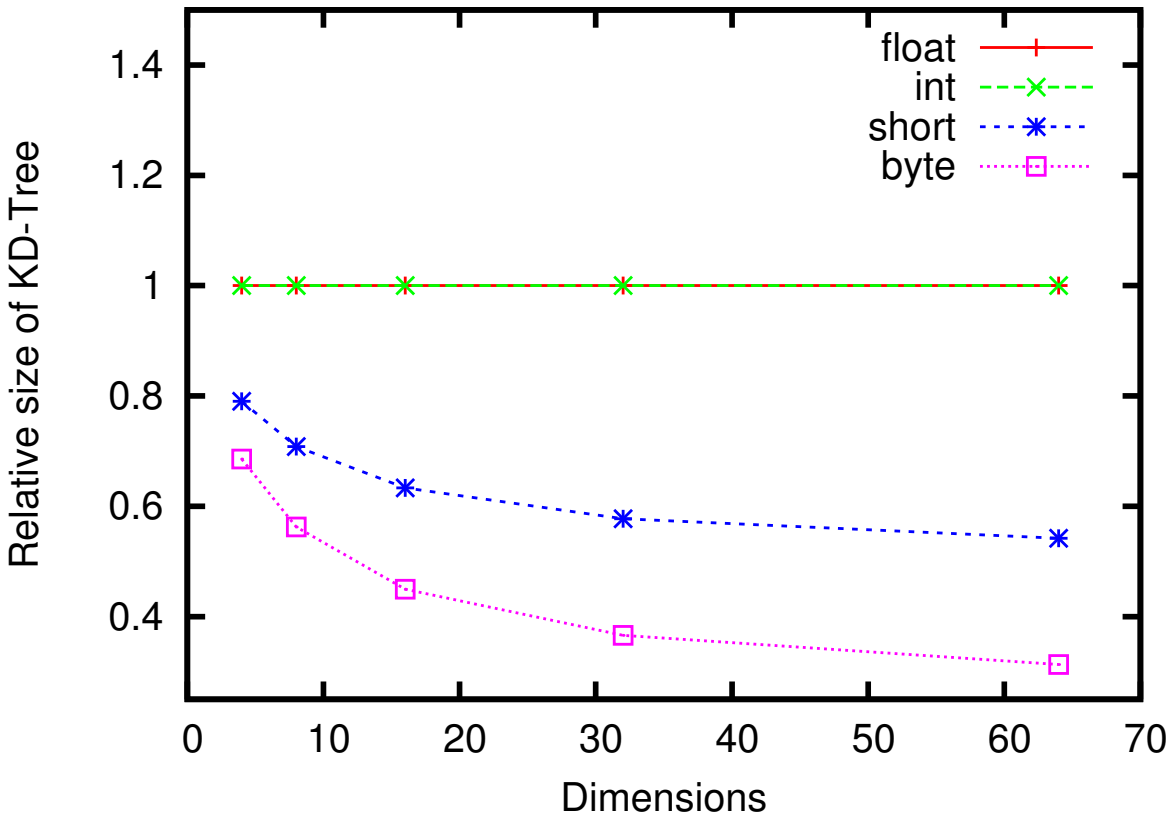
Face verification results on the XM2VTS database.

| **Algorithm** | **TER@EER** [%] | |
|---|---|---|
| | float | byte |
| Eigenfaces | 19.0 | 19.1 |
| Fisherfaces | 11.0 | 11.0 |
| Local Features | 4.7 | 4.8 |

# Results

## Memory Usage Reduction

- The size of the models for storage is reduced to a quarter (with no data compression).
- The memory required during execution can be approximated by the size of the *kd*-tree structure.



# Results

## Verification Times on a Nokia N70

| Algorithm | | extr. | proj. | score | Total |
|---|---|---|---|---|---|
| **Eigenfaces** | float | 379 | 92.4 | 22.8 | $494 \pm 9$ |
| | byte | 368 | 17.8 | 2.8 | $389 \pm 5$ |
| **Fisherfaces** | float | 375 | 55.1 | 14.2 | $444 \pm 8$ |
| | byte | 363 | 11.1 | 1.9 | $375 \pm 6$ |
| **Local Features** | float | 232 | 1266 | 2175 | $\mathbf{3764 \pm 13}$ |
| | byte | 119 | 271 | 369 | $\mathbf{759 \pm 14}$ |

- Operating system: Symbian OS v8.1a
- CPU: ARM-926 220MHz
- Memory: 22 MB

# Results

## Face Verification on a Nokia N70 - Summary

| Algorithm | TER@ FAR=0.1% [%] | Model Size [kB] | Memory [MB] | Enrol. Time [s] | Verif. Time [s] |
|---|---|---|---|---|---|
| **Eigenfaces** | 57.0 | 0.141 | 0.118 | 0.387 | 0.389 |
| **Fisherfaces** | 37.2 | 0.085 | 0.075 | 0.374 | 0.375 |
| **Local Features** | 29.8 | 28.23 | 1.093 | 0.566 | 0.759 |

# Results

## Face Detection

Three face detectors were implemented:

1. LDPP 1 (16 dimensions, 8 prototypes per class)
   - Detects only one face.
   - Takes 4.5 seconds for an image of $160 \times 120$ on a Nokia N70.
2. LDPP 2 (same model as LDPP 1)
   - Detects multiple faces.
   - Takes 13.4 seconds for an image of $160 \times 120$ on a Nokia N70.
3. Viola and Jones (uses the model that comes with the OpenCV)
   - Detects multiple faces.
   - Takes 5 seconds for an image of $160 \times 120$ on a Nokia N70.

# Demonstration Applications

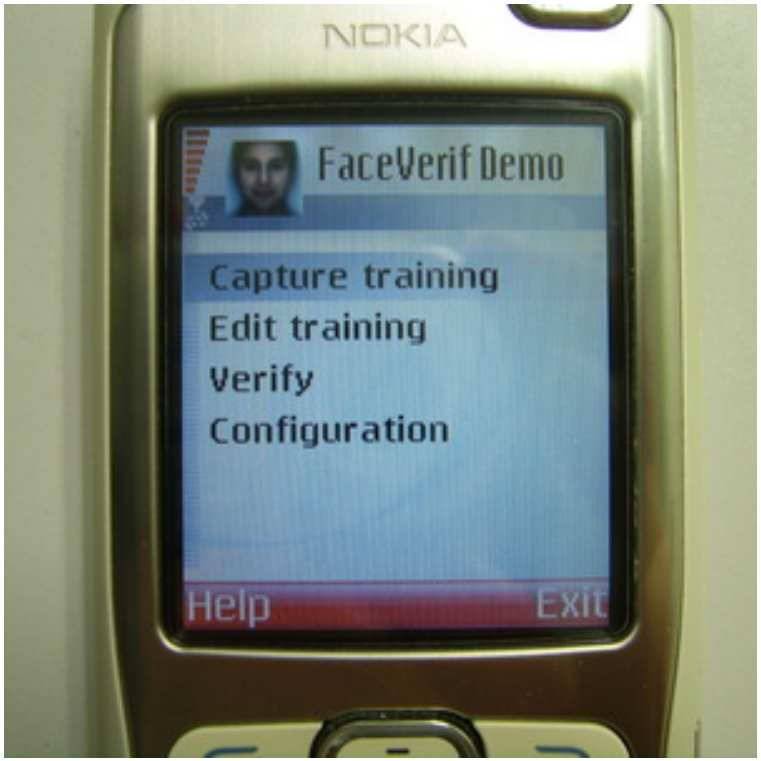
## Face Detection

- J2ME MIDlet
- Captures an image and detects faces
- Can choose among LDPP 1, LDPP 2 and OpenCV

Available in: `http://www.iti.upv.es/~mvillegas/research/`

# Demonstration Applications

## Face Verification




- J2ME MIDlet
- Captures an image and verifies the identity of the user
- Can choose among eigenfaces, fisherfaces and local features

Available in: `http://www.iti.upv.es/~mvillegas/research/`

# Demonstration Applications

## Face Gender Recognition

- J2ME MIDlet
- Captures an image decides if is male or female
- Error rate 8.5% for perfect detection
- Error rate 10%–20% with detection errors

Available in: `http://www.iti.upv.es/~mvillegas/research/`

# Conclusions

- In this work some face recognition algorithms were analysed and optimisations targeted at mobile devices were proposed.
- With the proposed optimisations:
  - The recognition performance is not affected.
  - The algorithms are almost 5 times faster.
  - The memory requirements is reduced by a factor of 2.
- Three demonstration applications were developed: face detection, face verification and gender recognition.
- The face detection implementations need further work to achieve competitive performance.

# Conclusions

- With a little more of work, a user authentication application can be developed, e.g. to unblock a mobile phone.
- A java implementation makes the software accessible to a great variety of devices.
- The proposed optimisations also work for non-mobile applications, e.g. face identification in big databases, and they can also be used on other pattern recognition problems.

# Publications


M. Villegas and R. Paredes.
Simultaneous learning of a discriminative projection and prototypes for nearest-neighbor classification.
In ***CVPR 2008***, Anchorage AK (USA), June 2008. IEEE Computer Society.

M. Villegas, R. Paredes, A. Juan, and E. Vidal.
Face verification on color images using local features.
In *Workshop on Biometrics, associated with CVPR 2008*, Anchorage AK (USA), June 2008. IEEE Computer Society.

M. Villegas and R. Paredes.
Face recognition in color using complex and hypercomplex representations.
In *IbPRIA*, Girona (Spain), June 2007.

M. Villegas and R. Paredes.
Illumination Invariance for Local Feature Face Recognition.
In *1st Spanish Workshop on Biometrics*, Girona (Spain), June 2007.

M. Villegas and R. Paredes.
Comparison of illumination normalization methods for face recognition.
In *Third COST 275 Workshop*, University of Hertfordshire (UK), October 2005.

Thank you for your attention !

Questions ?