Classification

Daniel Novák

20.12, 2018, Prague

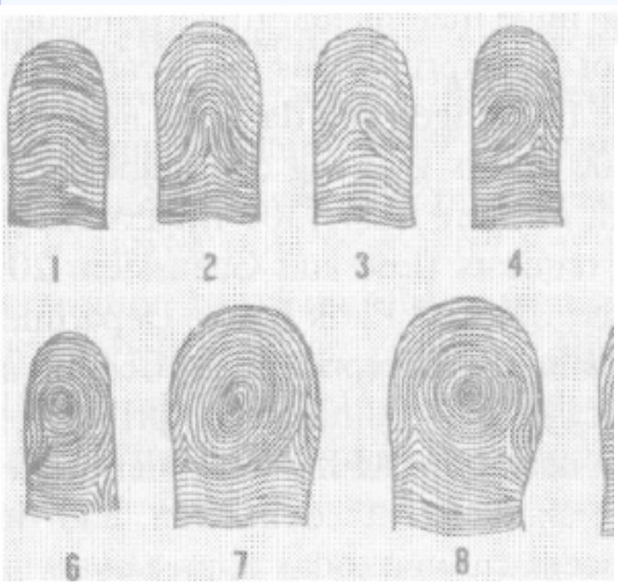
Acknowledgments: Xavier Palathingal, Andrzej Drygajlo,
Handbook of Fingerprint Recognition

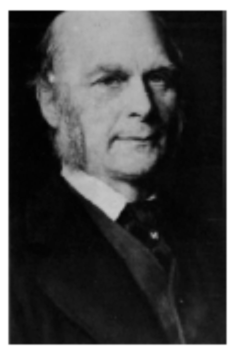
laboratory Gerstner

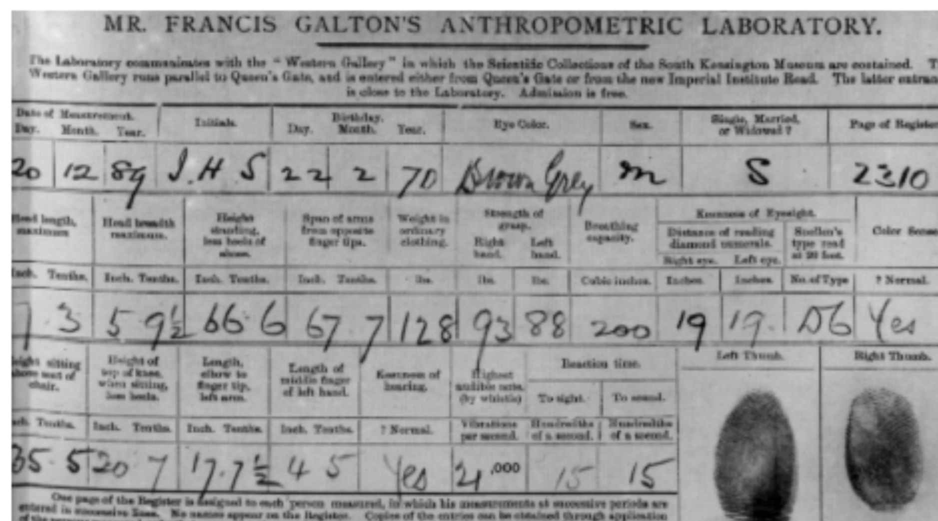
Department of Cybernetics, Czech Technical University

History

- In 1823, Purkinji proposed the first fingerprint classification, which classified into nine categories: (transverse curve, central longitudinal stria, oblique stripe, oblique loop, almond whorl, spiral whorl, ellipse, circle, and double whorl)
- Sir Francis Galton introduced the minutae features for fingerprint matching in late 19th century







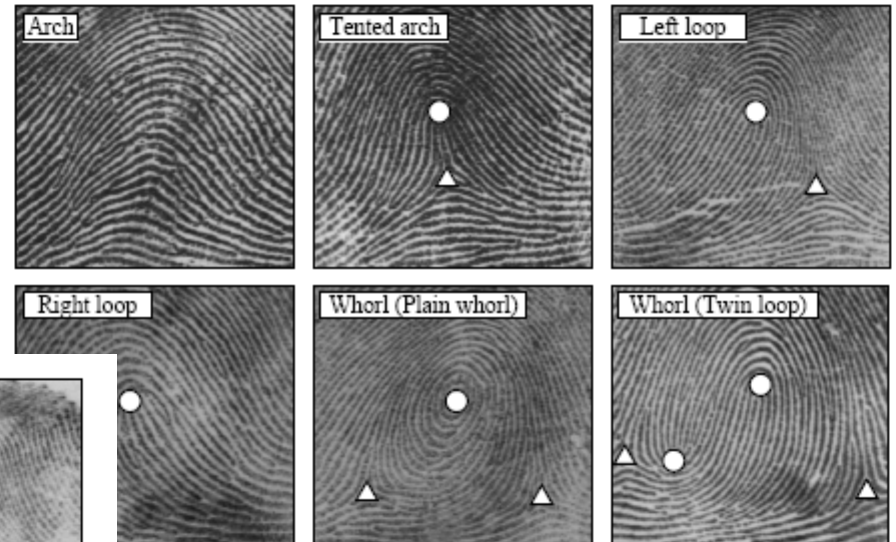
Classes

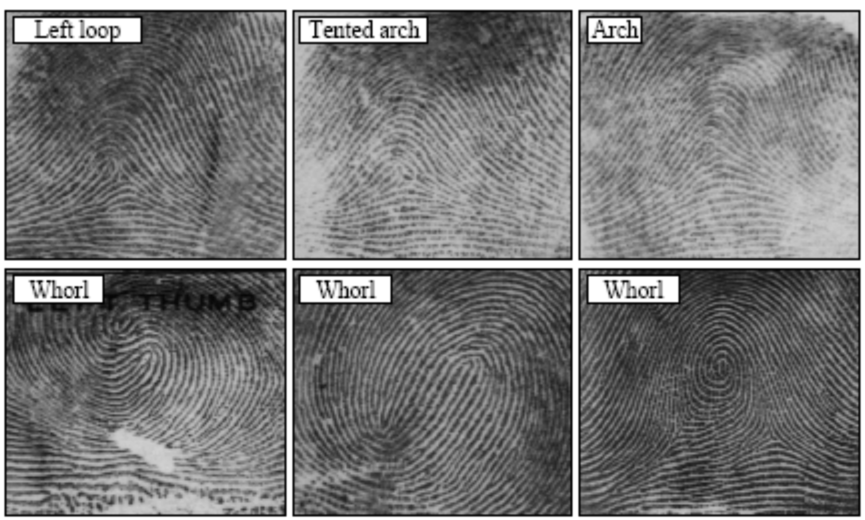
- Arch: 3.7%, tented arch: 2.9%, left loop: 33.8%, right loop: 31.7%, whorl: 27.9%
- Pattern recognition

PROBLEM

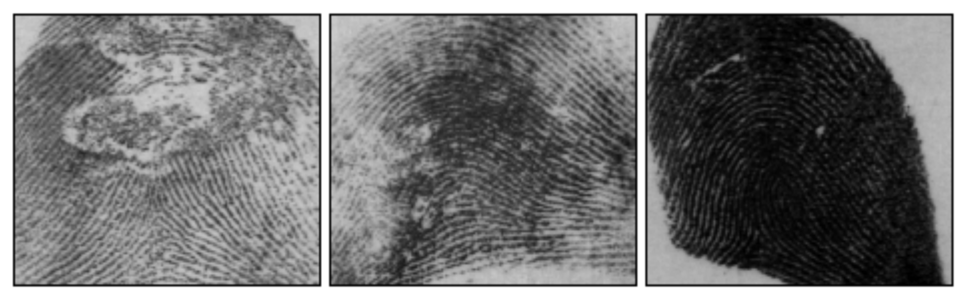
Small-inter class variability

Large intra-class variability



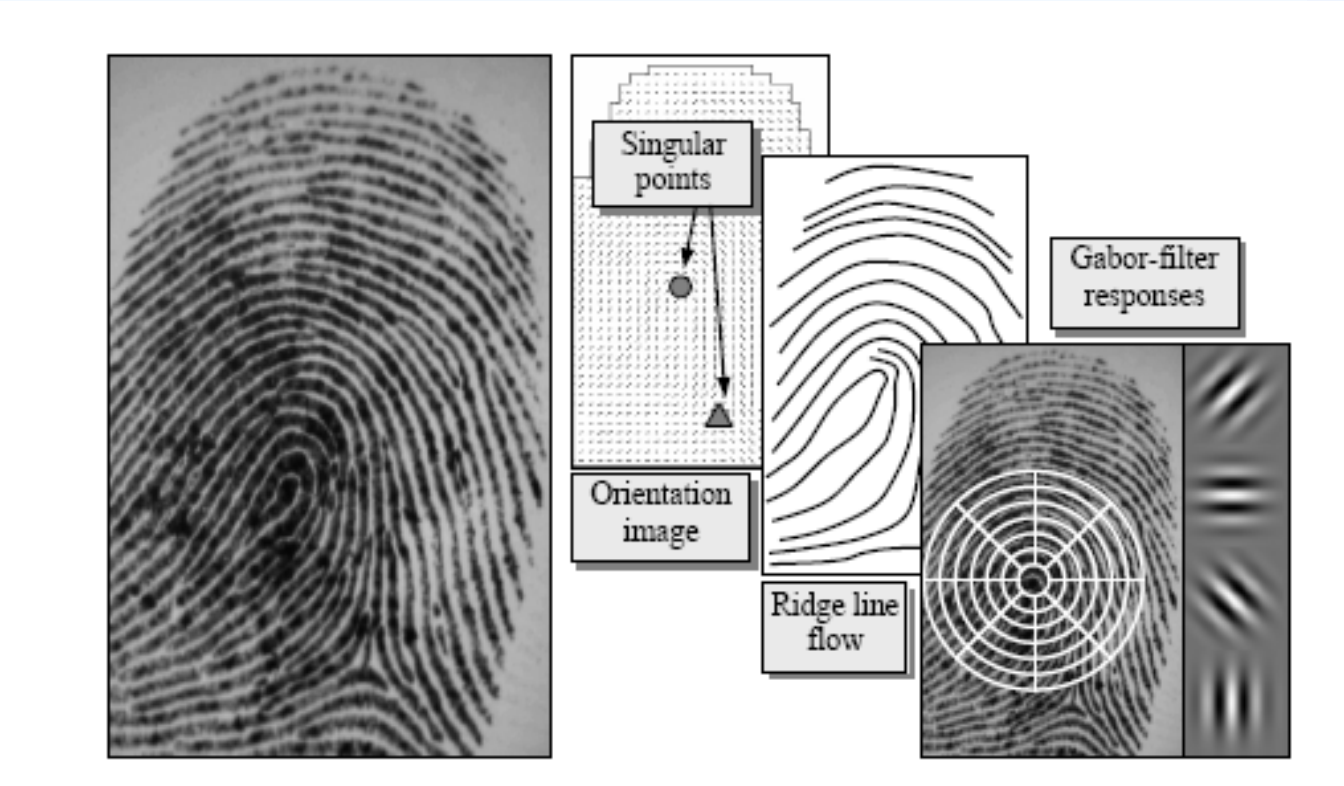


presence of noise



Features

- Based on global features
- Ridge line flow, orientation image, singular point, Gabor filters
- A-priori information: sex, race, age.



Techniques

- Features:
  - O = orientation image
  - S = singularities
  - R = ridge flow,
  - G = Gabor
- classification technique
  - Rb = rule-based
  - Sy = syntactic
  - Str = structural,
  - Sta = statistical
  - Nn = neural network
  - Mc = multiple classifiers

Fingerprint classification approach	*Features*				*Classifier*					
	O	S	R	G	Rb	Sy	Str	Sta	Nn	Mc
Moayer and Fu (1975)	√					√				
Moayer and Fu (1976)	√					√				
Rao and Balck (1980)	√					√				
Kawagoe and Tojo (1984)		√	√		√					
Hughes and Green (1991)	√								√	
Bowen (1992)	√	√							√	
Kamijo, Mieno, and Kojima (1992)	√								√	
Kamijo (1993)	√								√	
Moscinska and Tyma (1993)	√				√				√	
Wilson, Candela, and Watson (1994)	√								√	
Candela et al. (1995)	√		√		√				√	√
Omidvar, Blue, and Wilson (1995)	√								√	
Halici and Ongun (1996)	√								√	
Karu and Jain (1996)		√			√					
Maio and Maltoni (1996)	√						√			
Ballan, Sakarya, and Evans (1997)		√			√					
Chong et al. (1997)			√		√					
Senior (1997)			√				√			
Wei, Yuan, and Jie (1998)	√				√				√	√
Cappelli et al. (1999)	√						√			
Cappelli, Maio, and Maltoni (1999)	√							√		
Hong and Jain (1999)		√	√		√					√
Jain, Prabhakar, and Hong (1999)				√				√	√	√
Lumini, Maio, and Maltoni (1999)	√						√			
Cappelli, Maio, and Maltoni (2000a)	√							√		√
Cho et al. (2000)		√			√					
Bartesaghi, Fernández, and Gómez (2001)		√			√					
Bernard et al. (2001)	√								√	
Marcialis, Roli, and Frasconi (2001)	√			√			√	√	√	√
Pattichis et al. (2001)	√				√				√	√
Senior (2001)	√		√		√		√		√	√
Yao, Frasconi, and Pontil (2001)				√				√		√
Cappelli, Maio, and Maltoni (2002a)	√							√		√
Jain and Minut (2002)			√		√					
Cappelli et al. (2003)	√							√		√
Yao et al. (2003)	√			√			√	√	√	√

Overview

Fingerprint classification approach	*Features*				*Classifier*					
	O	S	R	G	Rb	Sy	Str	Sta	Nn	Mc
Cappelli and Maio (2004)	√							√		√
Klimanee and Nguyen (2004)	√	√			√					
Senior and Bolle (2004)	√		√		√		√		√	√
Shah and Sastry (2004)								√	√	√
Wang and Xie (2004)	√	√	√		√					
Zhang and Yan (2004)	√	√	√		√					
Park and Park (2005)	√							√		
Neuhaus and Bunke (2005)	√						√			
Tan, Bhanu, and Lin (2005)	√							√		
Min, Hong, and Cho (2006)				√				√		√
Kristensen, Borthen, and Fyllingsnes (2007)				√					√	
Wang and Dai (2007)	√	√			√					
Hong et al. (2008)	√	√		√				√		√
Li, Yau, and Wang (2008)	√	√						√		

Syntactic & Structal approach

– Syntactic: terminal symbols & production rules, nice source

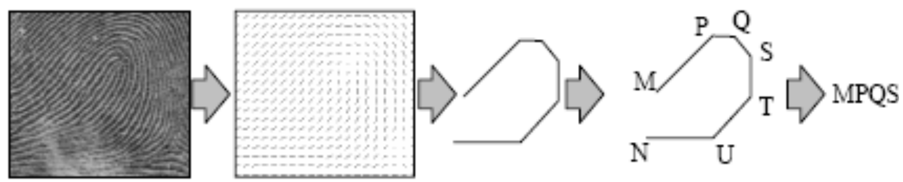
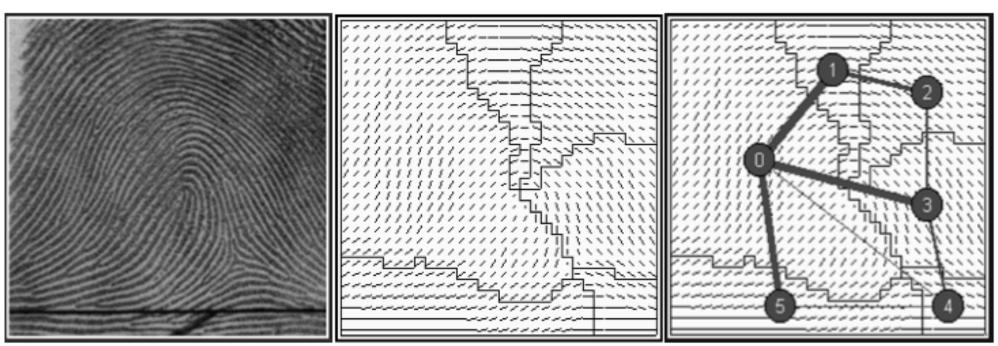
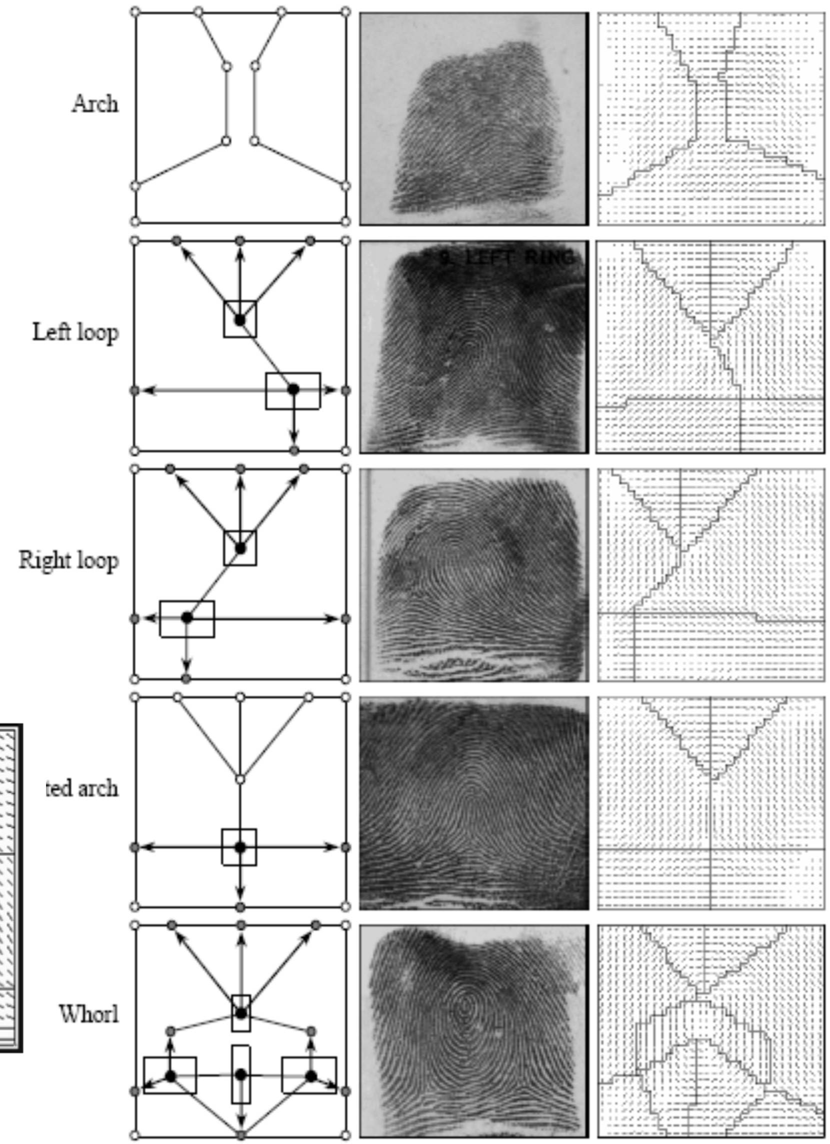


Figure 5.7. A schema of the string-construction approach in Rao and Balck (198(

– Structural approach
– COST fce-> minimazing variation of orientation





Statistical approach

- Statistical
  - Orientation image
  - k-nearest neighbor
  - 30x30 array, 1800 elements, training impossible, high dimension
  - PCA (the Karhunen-Loeve) transform

$$\mathbf{d} = \left[r \cdot cos\left(2\theta\right), r \cdot sin\left(2\theta\right) \right].$$

  - Another examples: Bayes decision rule, Support Vector Machine
- Neural networks
  - orientation image
  - multilayer percepton

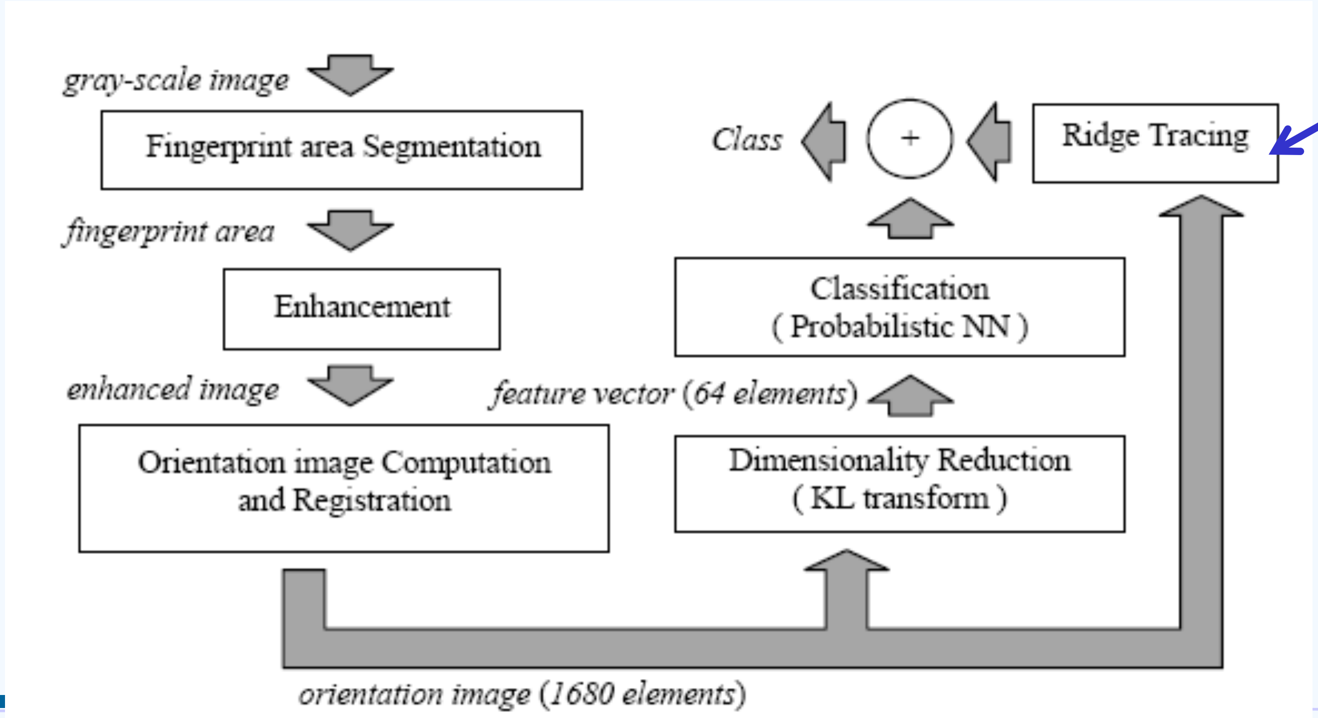
Multiple approach I

	Distinct features	Distinct classi-fiers	Distinct training sets	Combination strategy
Candela et al. (1995)	Yes	Yes	No	Rule-based
Jain, Prabhakar, and Hong (1999)	No	Yes	No	Sequential (two stages)
Cappelli, Maio, and Maltoni (2000a)	No	Yes	Yes	Majority vote rule
Senior (2001)	Yes	Yes	No	Neural network
Marcialis, Roli, and Frasconi (2001)	Yes	Yes	No	k-nearest neighbor
Yao et al. (2003)	Yes	Yes	No	k-nearest neighbor
Cappelli et al. (2003)	No	Yes	No	Sequential (two stages)
Shah and Sastry (2004)	Yes	Yes	No	Sequential (two stages)
Hong et al. (2008)	Yes	Yes	No	Bayes rule

Multiply approach II

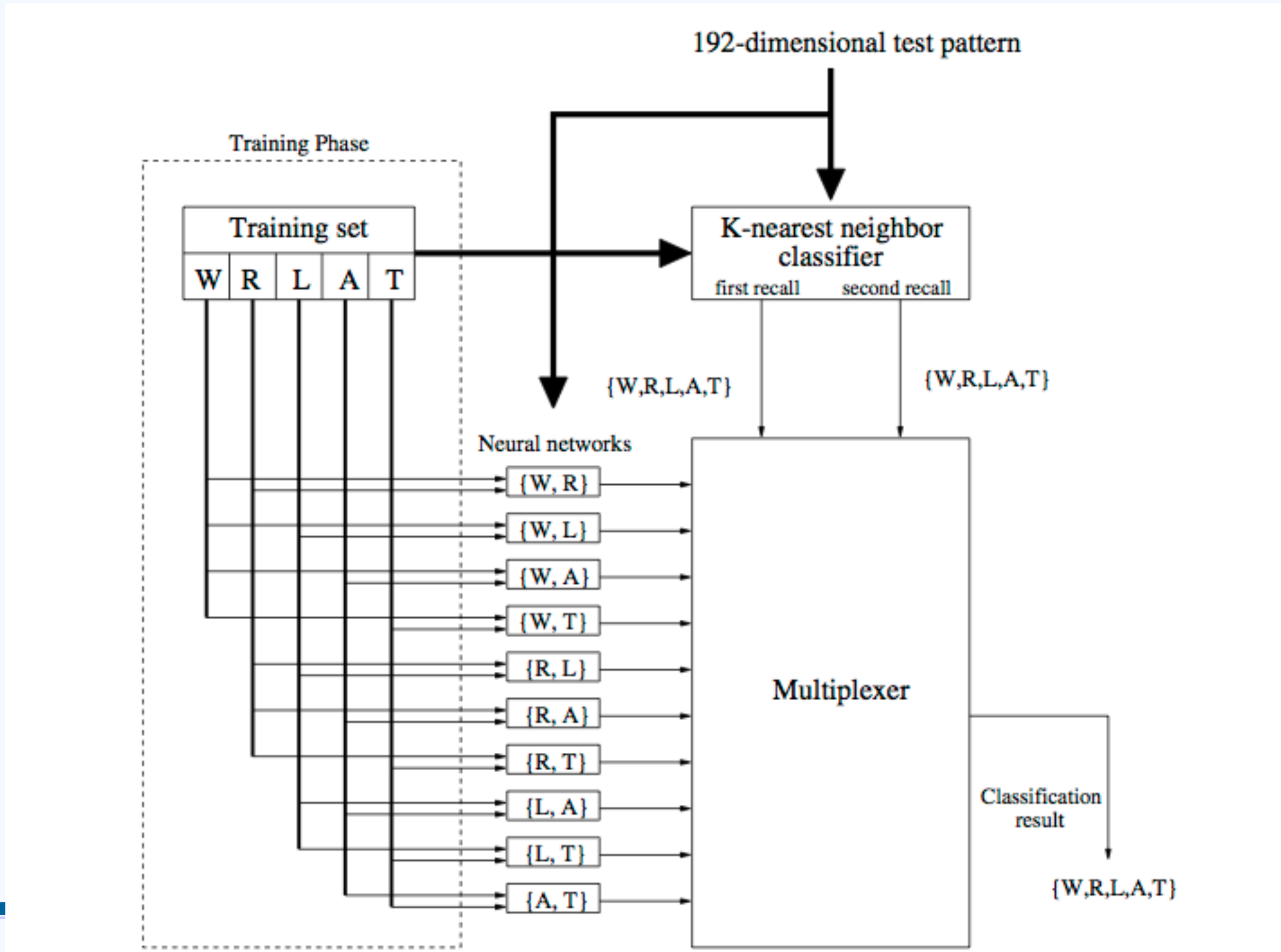
- Multiple classifier-based apparoaches
- PCASYS: Pattern-level Classification Automation system
- Open Source: http://ffpis.sourceforge.net/
- Developed by NIST: http://www.nist.gov/index.html
  - National Institute of Standards and Technology

Whorl module



Multiply approach III

– Finger code features – k-two most likely classes, neural networks distinguishing two classes



Prize of GOLD

- Good large database> very expensive (FP,ECG,EEG, etc.)
  - DB4,DB14: STANDARDS for classification systems
  - 8bit- grey level images of rolled FP scanned from cards,
  - Manual annotation by a human expert (A,L,R,T,W)
  - 2000 FPs: DB4
  - 27000 FPs: DB14
  - All classes are distributed equally. However, Arch: 3.7%, tented arch: 2.9%, left loop: 33.8%, right loop: 31.7%, whorl: 27.9%
  - Therefore some authors weight results according class distribution

Classification Evaluation

- Accuracy
  - Rejection can improve accuracy

$$\text{error rate} = \frac{\text{number of misclassified fingerprints} \times 100}{\text{total number of fingerprints}} \%$$

$$\text{accuracy} = 100\% - \text{error rate}.$$

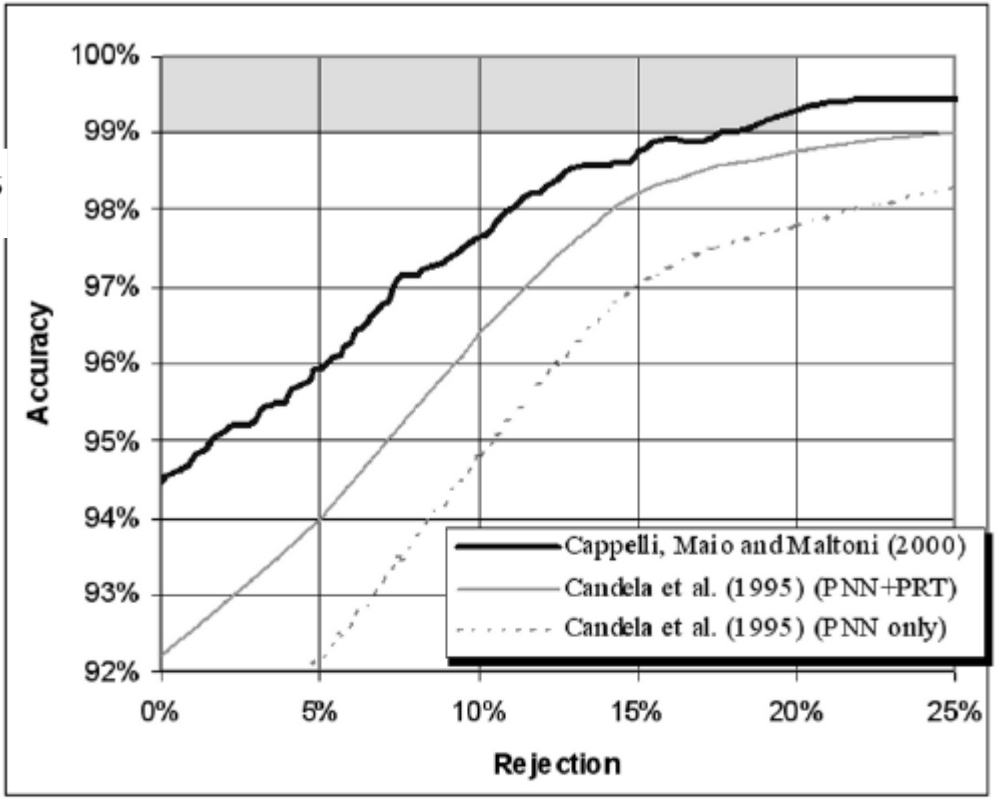
- Penetration rate: time constraint

$$\text{penetration rate} = \frac{\text{number of accessed fingerprints} \times 100}{\text{total number of fingerprints in the database}} \%$$

- Confusion matrix (DB4)

True class	Hypothesized class				
	A	L	R	W	T
A	**420**	6	3	1	11
L	3	**376**	3	9	11
R	5	1	**392**	6	16
W	2	5	14	**377**	1
T	33	18	9	0	**278**

- Rejection can improve accuracy (DB14)
  - Unknown class
  - FBI target:shaded area
  - **NIST database -DB 4, DB14**



Results on DB4, DB14

– Results on DB4

True class	Hypothesized class				
	A	L	R	W	T
A	**420**	6	3	1	11
L	3	**376**	3	9	11
R	5	1	**392**	6	16
W	2	5	14	**377**	1
T	33	18	9	0	**278**

True class	Hypothesized class			
	A+T	L	R	W
A+T	**782**	10	17	6
L	6	**373**	2	4
R	7	1	**381**	9
W	0	4	7	**391**

Method	Test set	5 classes		4 classes	
		%	*Weighted* (%)	%	*Weighted* (%)
Candela et al. (1995)	Second half	–	–	11.4	6.1
Karu and Jain (1996)	Whole DB	14.6	11.9	8.6	9.4
Senior (1997)	Random 542	–	–	–	8.4
Cappelli, Maio, and Maltoni (1999)	Second half	7.9	6.5	5.5	–
Hong and Jain (1999)	Whole DB	12.5	10.6	7.7	–
Jain, Prabhakar, and Hong (1999)	Second half (*)	10.0	7.0	5.2	–
Marcialis, Roli, and Frasconi (2001)	Second half (*)	12.1	9.6	–	–
Senior (2001)	Second half	–	–	–	5.1
Yao, Frasconi, and Pontil (2001)	Second half (*)	10.7	9.0	6.9	–
Jain and Minut (2002)	Whole DB	–	–	8.8	9.3
Cappelli et al. (2003)	Second half	4.8	3.7	3.7	3.4
Yao et al. (2003)	Second half (*)	10.0	8.1	-	-
Cappelli and Maio (2004)	Second half	7.0	5.9	4.7	5.4
Wang and Xie (2004)	Whole DB	-	-	18.0	-
Zhang and Yan (2004)	Whole DB	15.7	11.0	7.3	8.3
Neuhaus and Bunke (2005)	Second half	19.8	-	-	-
Park and Park (2005)	Second half	9.3	7.9	6.0	-
Tan, Bhanu, and Lin (2005)	Second half	8.4	8.0	6.7	7.5
Min, Hong, and Cho (2006)	Second half (*)	9.6	7.2	-	-
Wang and Dai (2007)	Whole DB	11.5	9.4	-	-
Hong et al. (2008)	Second half (*)	9.2	6.2	5.1	-
Li, Yau, and Wang (2008)	Second half	6.5	7.0	5.0	-

– Results on DB14, last 2700 examples for testing

Method	Error rate (%)
Candela et al. (1995)	7.8
Wei, Yuan, and Jie (1998)	6.0
Cappelli, Maio, and Maltoni (2000a)	5.6

Synthetic fingerprint generation

Daniel Novák

20.12, 2018, Prague

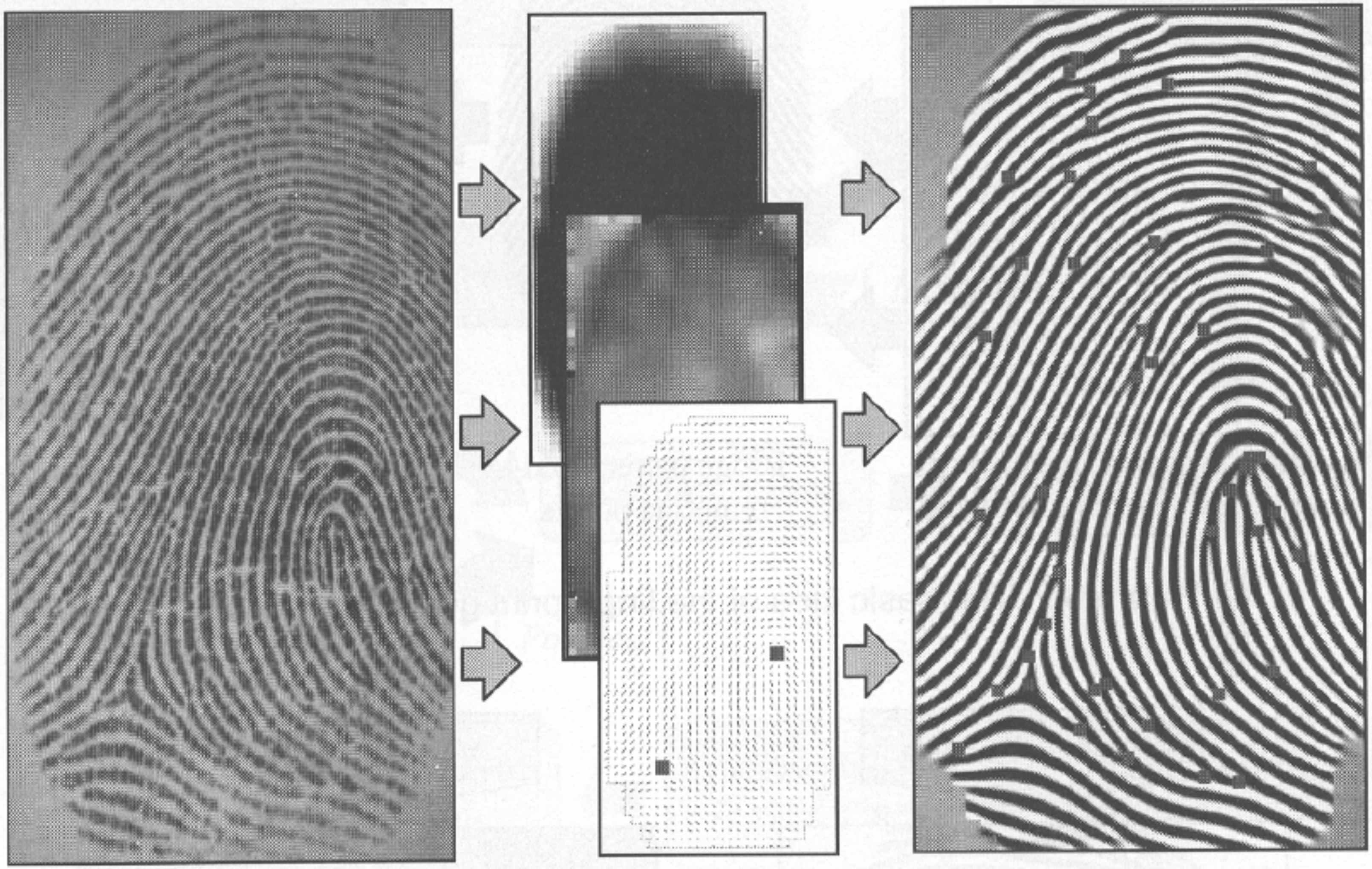
Acknowledgments: Xavier Palathingal, Andrzej Drygajlo, Handbook of Fingerprint Recognition

Department of Cybernetics, Czech Technical University

Synthetic fingerprint generation

- Motivation
  - Accuracy of each algorithm is usually evaluated on relatively small proprietary databases
  - Evaluation on small databases makes the accuracy estimates highly data dependent
  - When the databases are proprietary, the accuracy of various fingerprint matching algorithms cannot be compared directly
- Synthetic fingerprint generation can be used to automatically create large databases of fingerprints, thus allowing fingerprint recognition algorithms to be effectively trained, tested, optimized, and compared

Feature extraction process

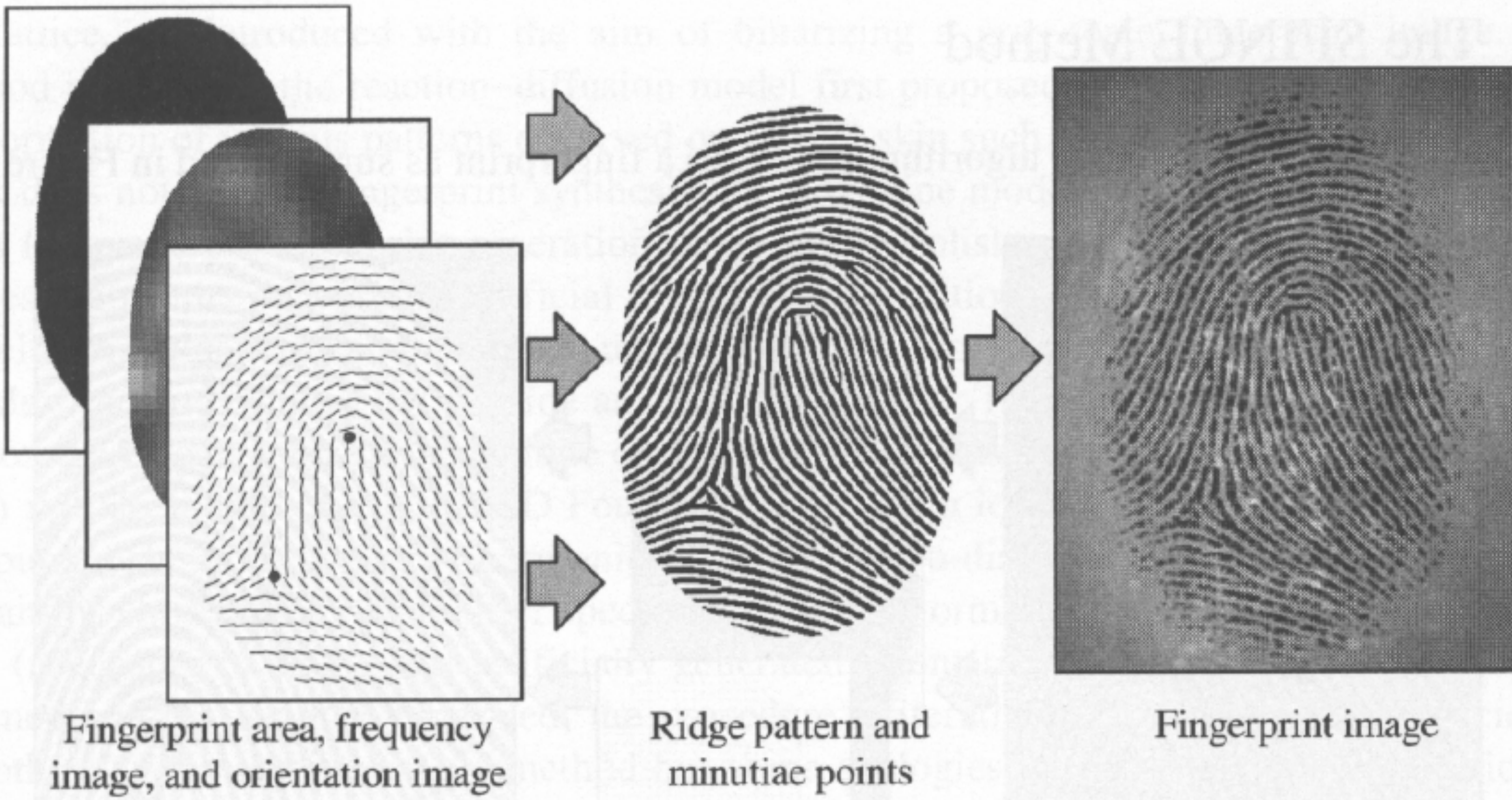


Fingerprint image

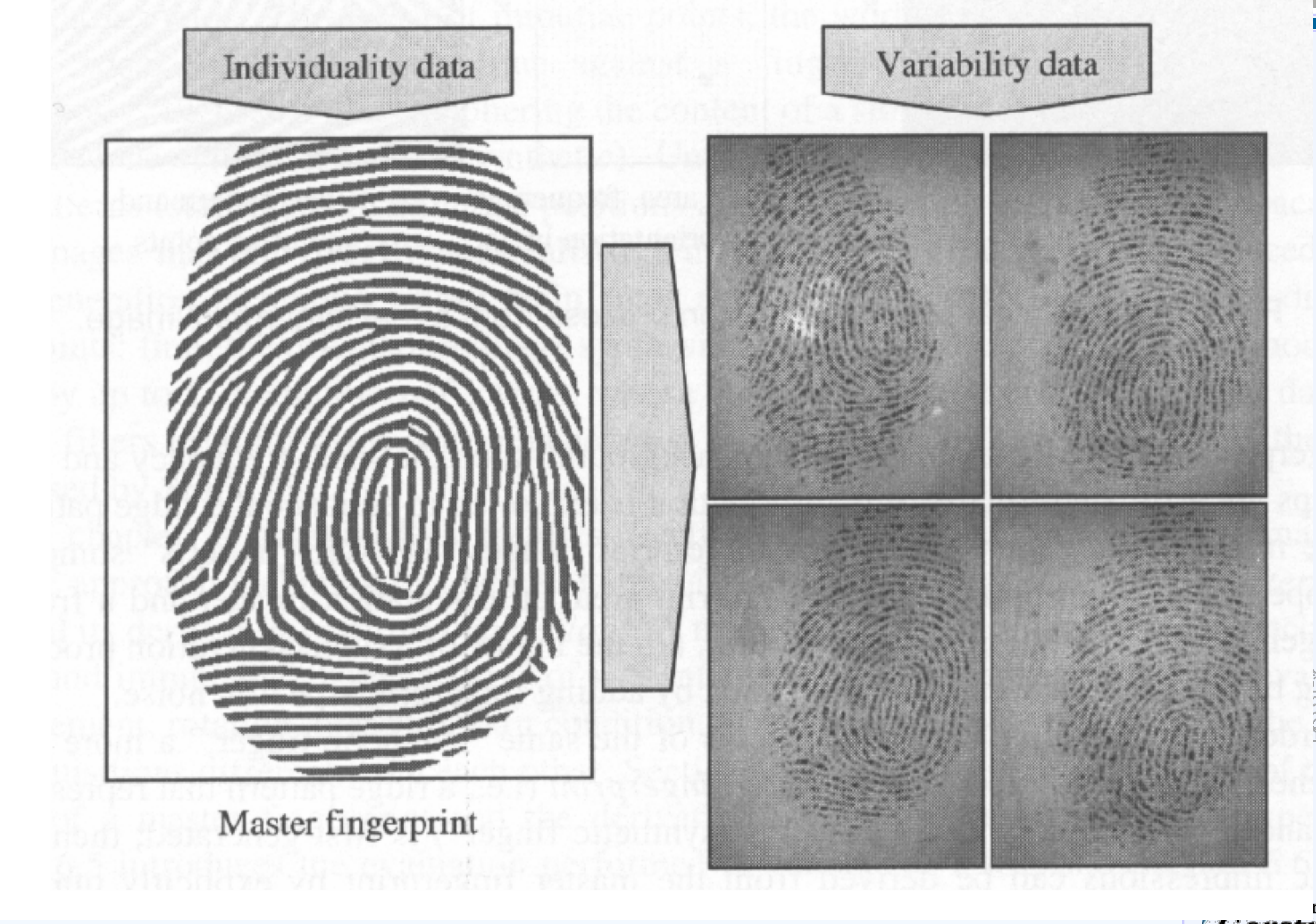
Fingerprint area, frequency image, and orientation image

Ridge pattern and minutiae points

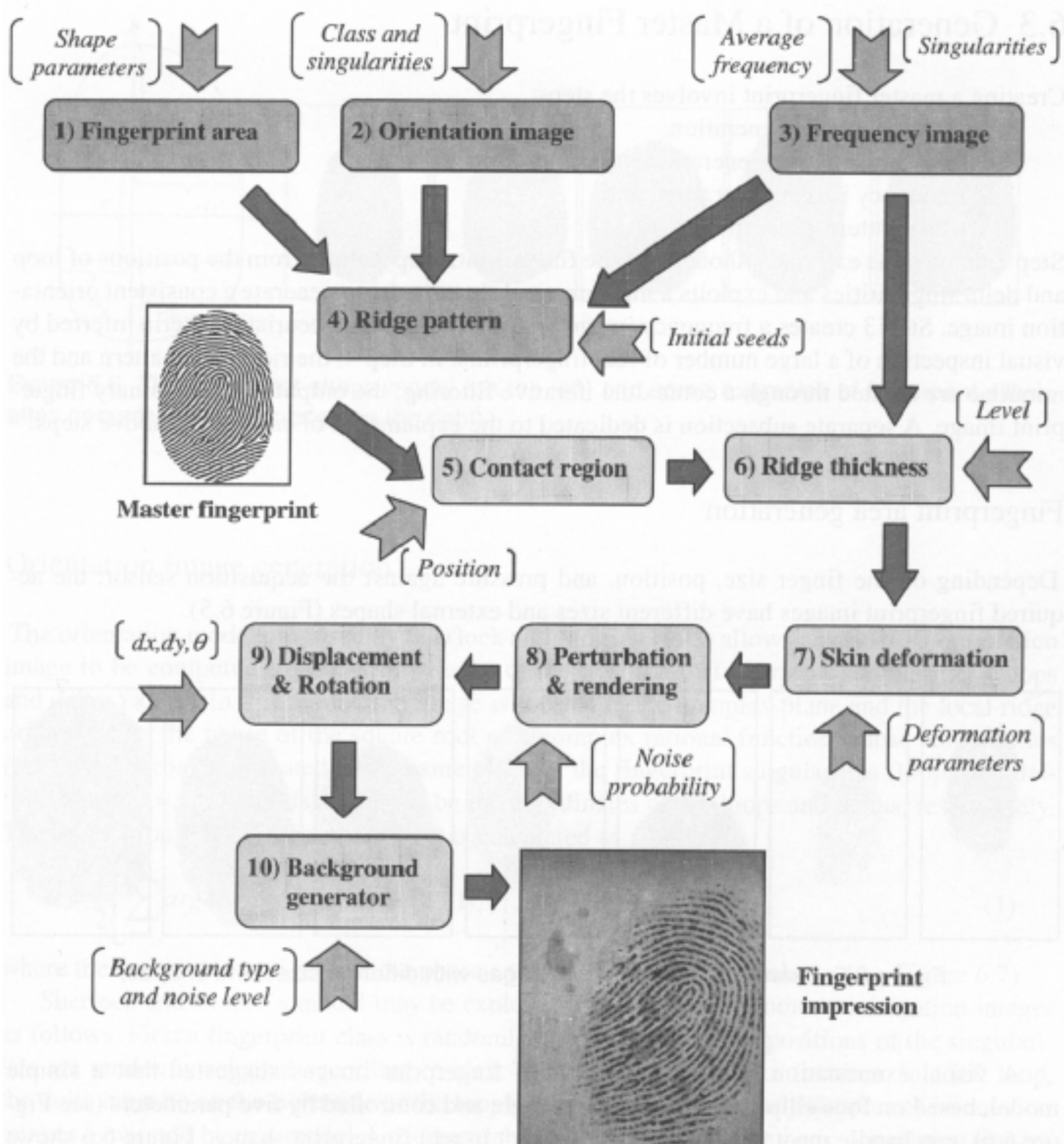
Basic idea


Fingerprint area, frequency image, and orientation image
Ridge pattern and minutiae points
Fingerprint image

From master to final impression

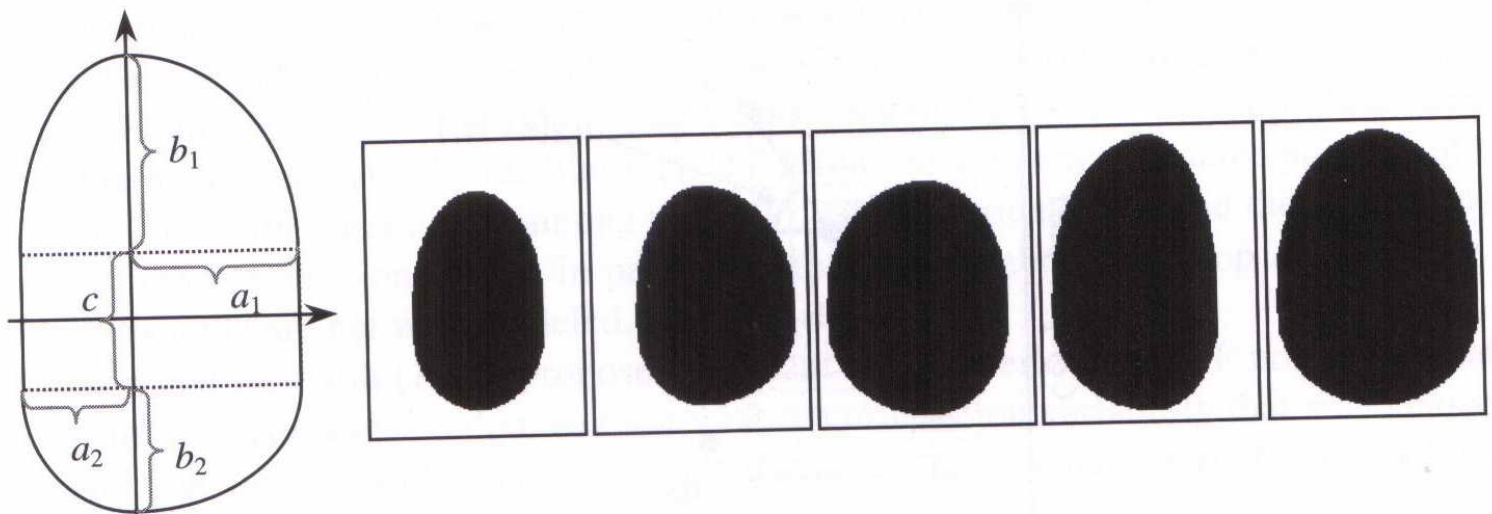


SFINGE

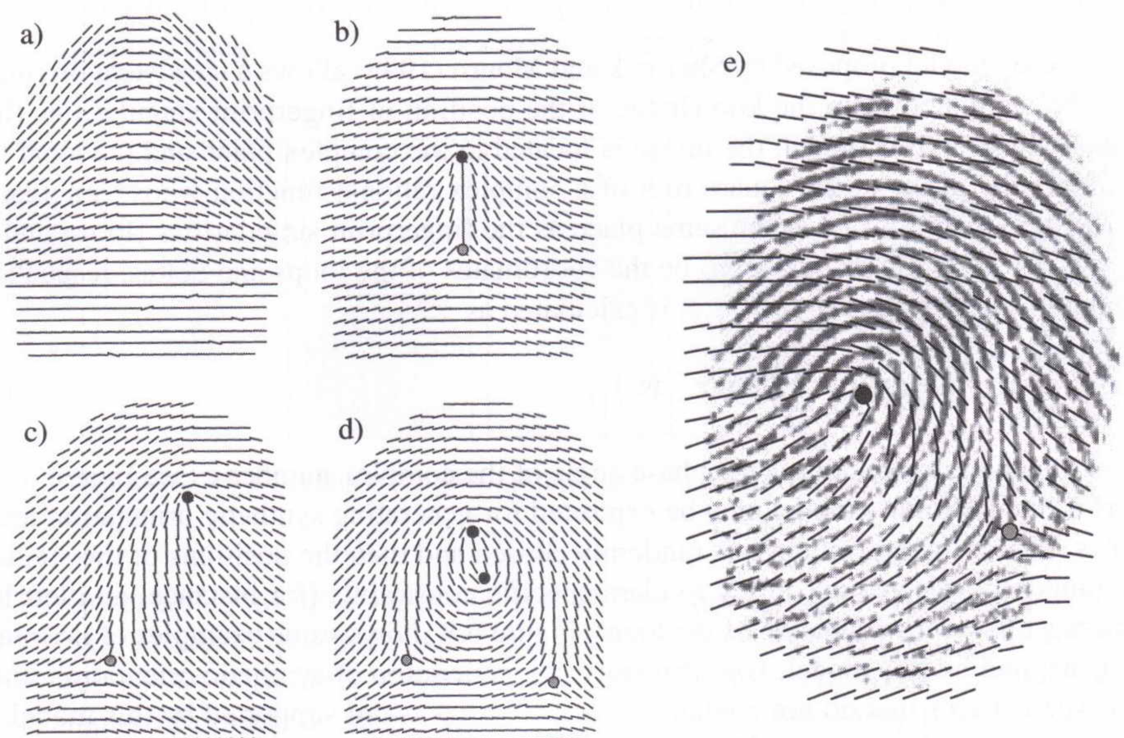


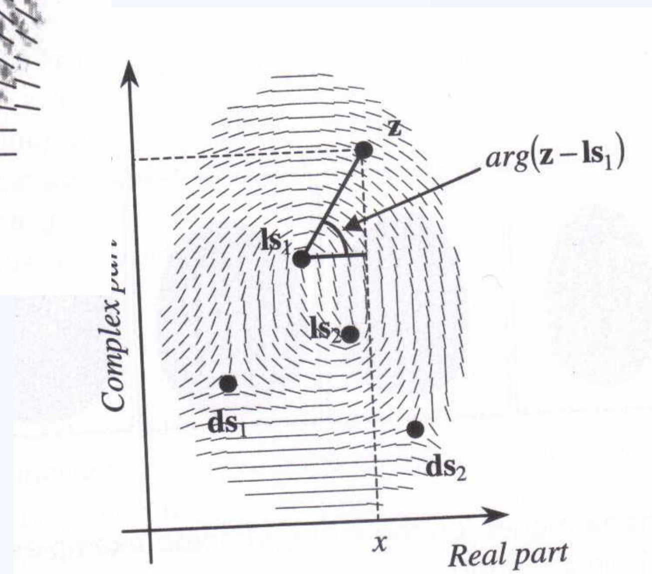
Shape parameters
Class and singularities
Average frequency
Singularities
1) Fingerprint area
2) Orientation image
3) Frequency image
4) Ridge pattern
Initial seeds
Level
Master fingerprint
5) Contact region
6) Ridge thickness
Position
dx,dy,θ
9) Displacement & Rotation
8) Perturbation & rendering
7) Skin deformation
Deformation parameters
Noise probability
10) Background generator
Fingerprint impression
Background type and noise level

FP area generation

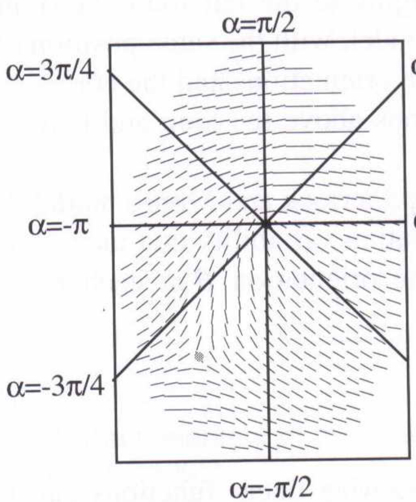


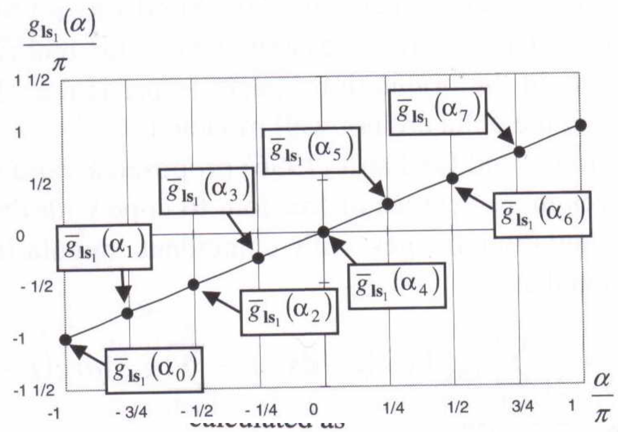
Orientation





Orientation



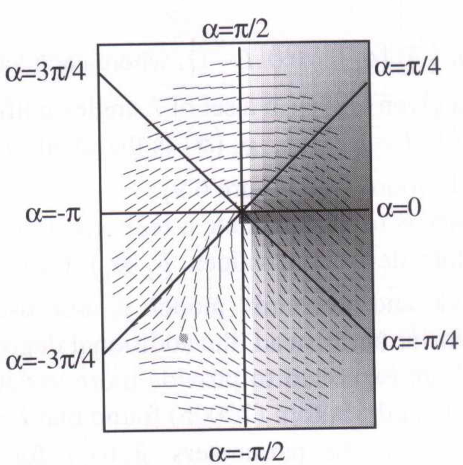


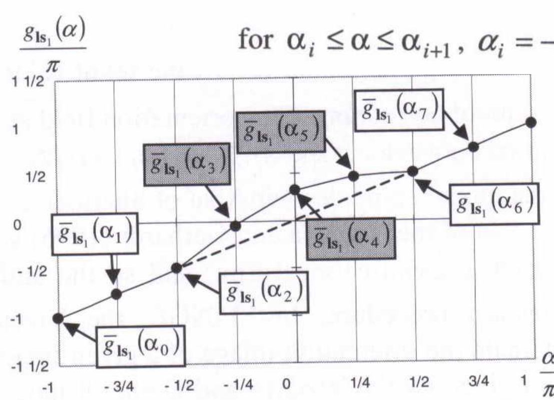
$$\theta = \frac{1}{2}\left[\sum_{i=1}^{n_d} g_{\mathbf{ds}_i}\left(arg\left(\mathbf{z}-\mathbf{ds}_i\right)\right) - \sum_{i=1}^{n_c} g_{\mathbf{ls}_i}\left(arg\left(\mathbf{z}-\mathbf{ls}_i\right)\right)\right], \quad (2)$$

where $g_k(\alpha)$, for $k \in \{\mathbf{ls}_1, ..., \mathbf{ls}_{n_c}, \mathbf{ds}_1, ..., \mathbf{ds}_{n_d}\}$, are piecewise linear functions capable of locally correcting the orientation field with respect to the value given by the Sherlock and Monroe model:

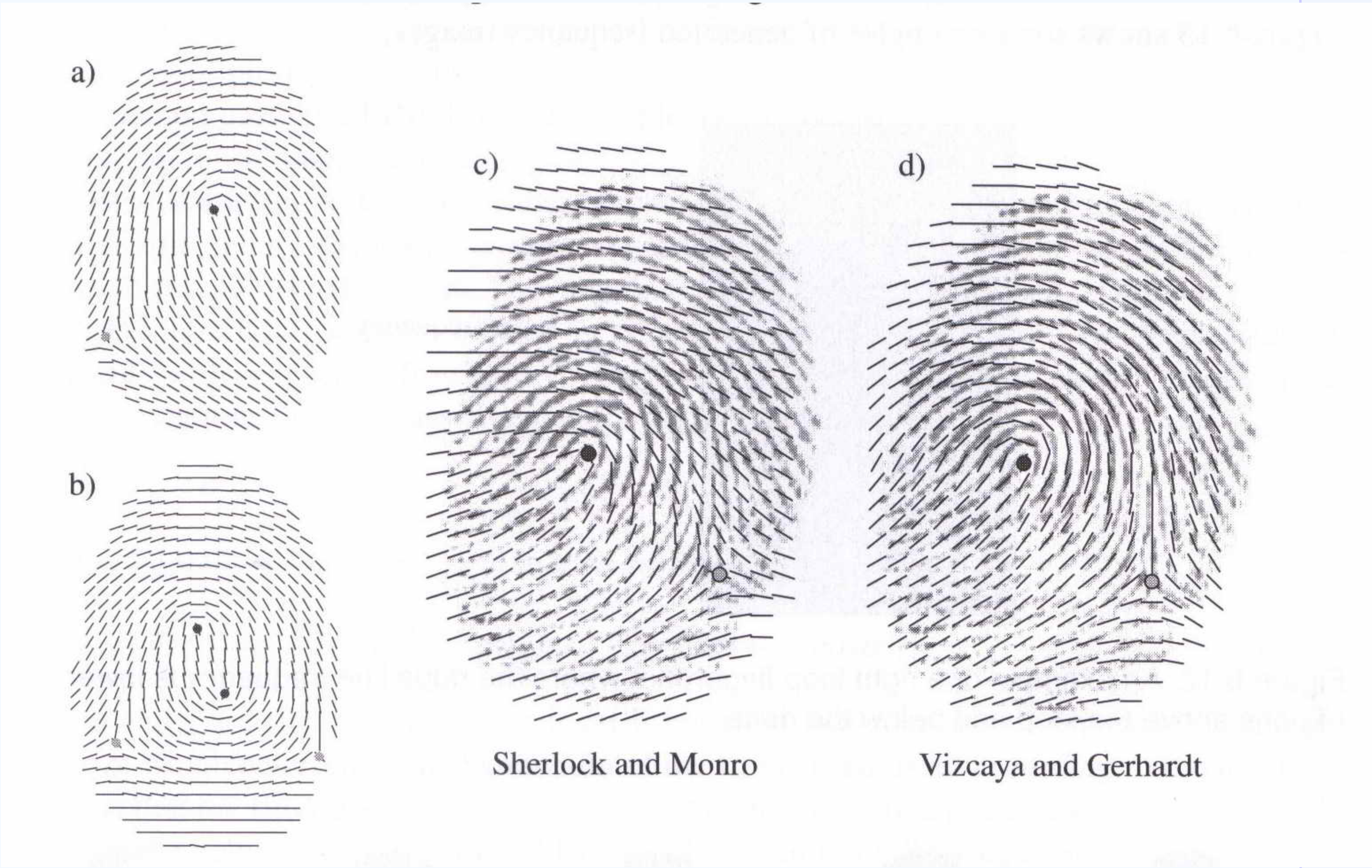
$$g_k(\alpha) = \bar{g}_k(\alpha_i) + \frac{\alpha-\alpha_i}{2\pi/L}\left(\bar{g}_k(\alpha_{i+1}) - \bar{g}_k(\alpha_i)\right), \quad (3)$$

for $\alpha_i \le \alpha \le \alpha_{i+1}$, $\alpha_i = -\pi + \frac{2\pi i}{L}$.

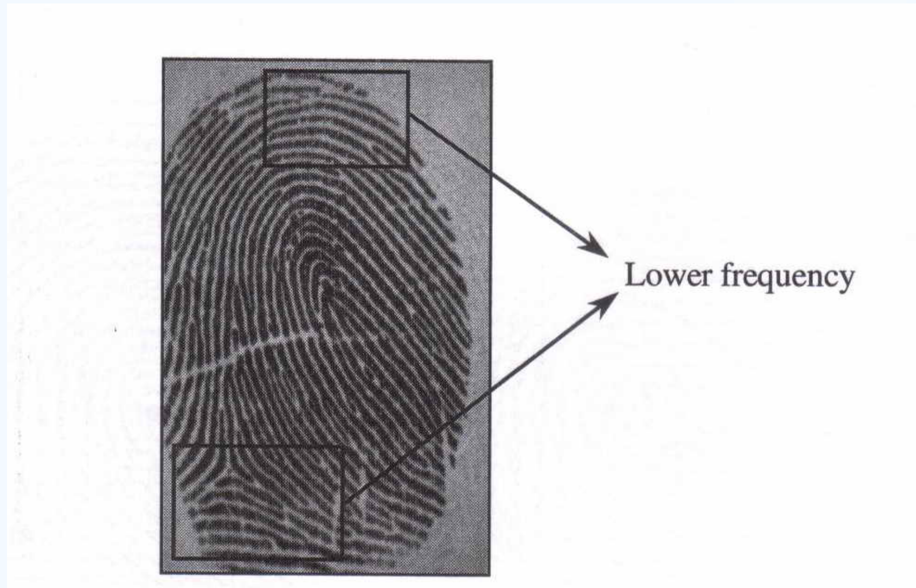


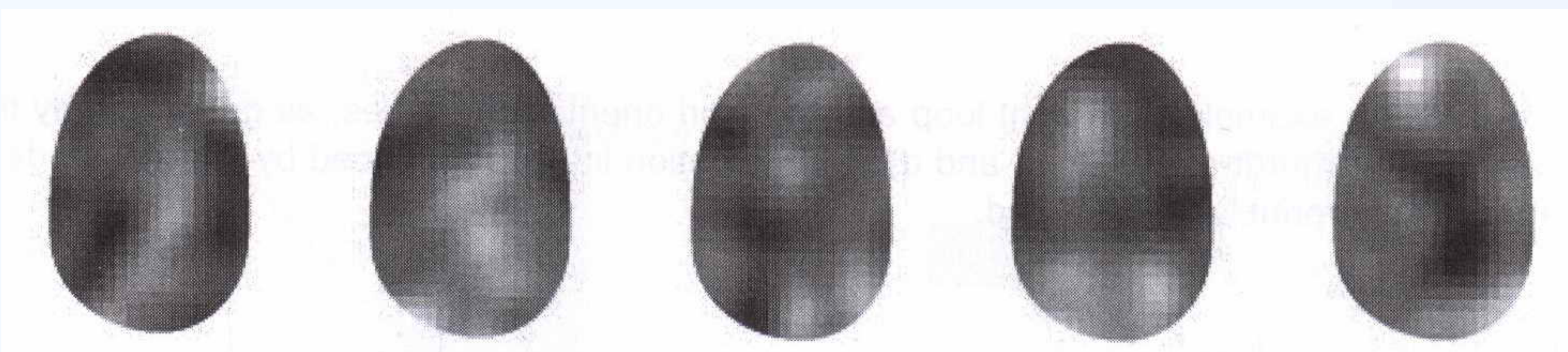


Orientation



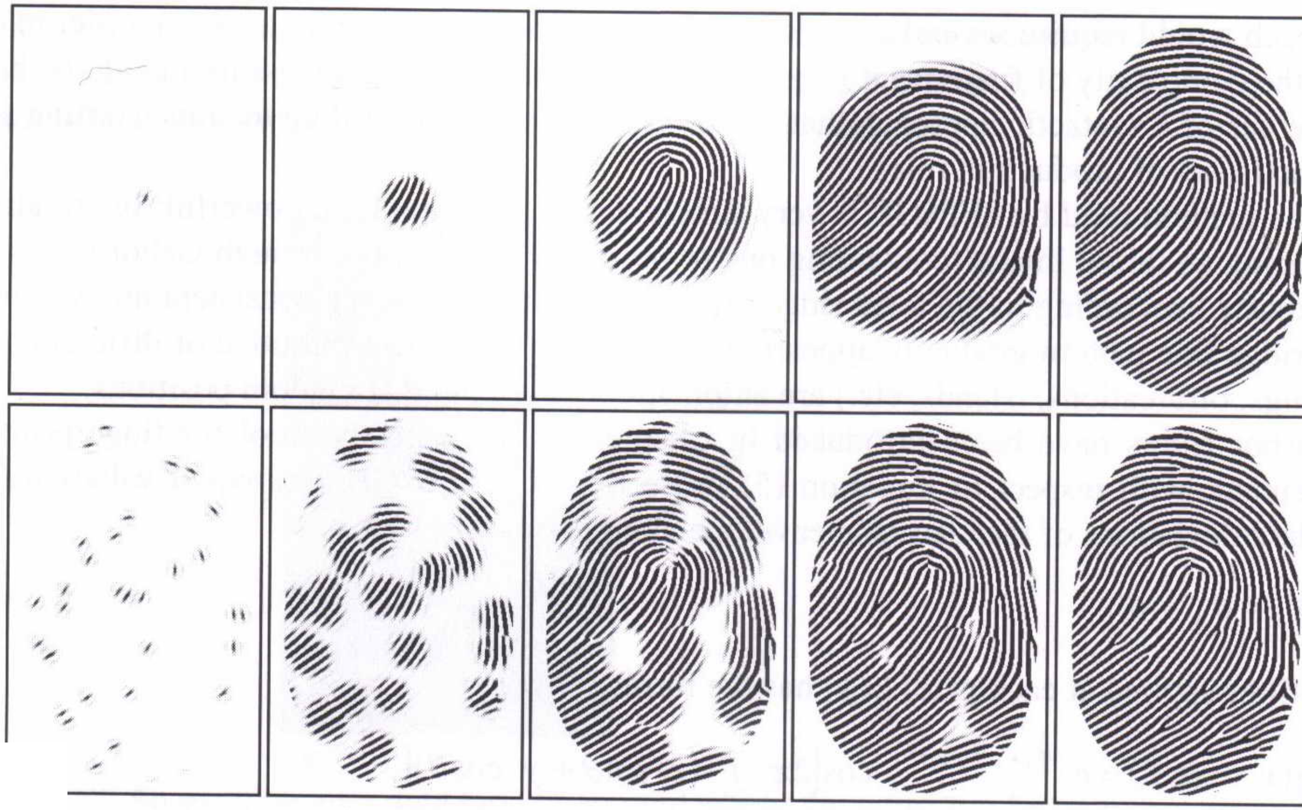
Frequency





Ridge line

- Gabor filter
- Seeds

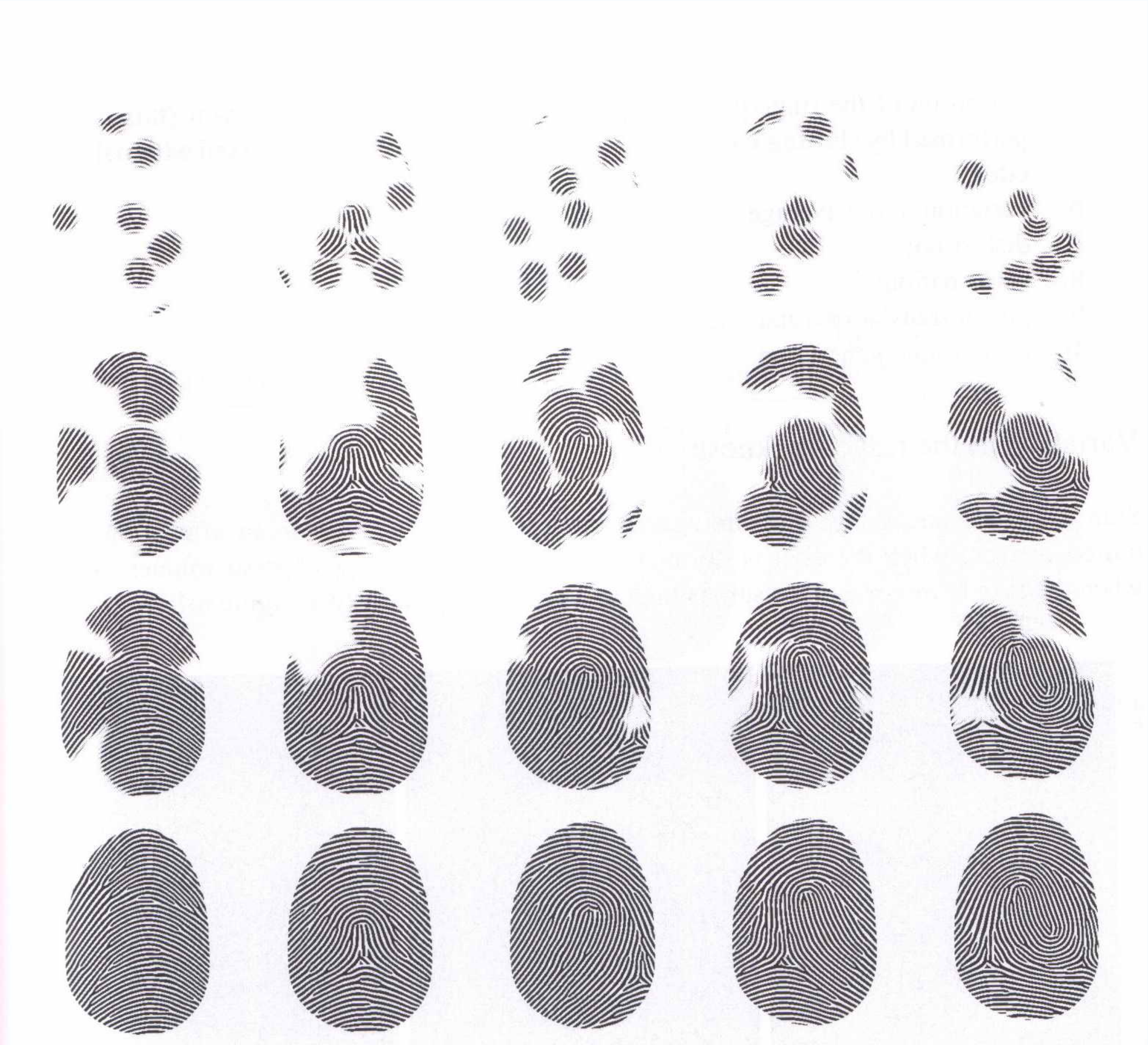


$$\sigma_x = \sigma_y = \sigma$$

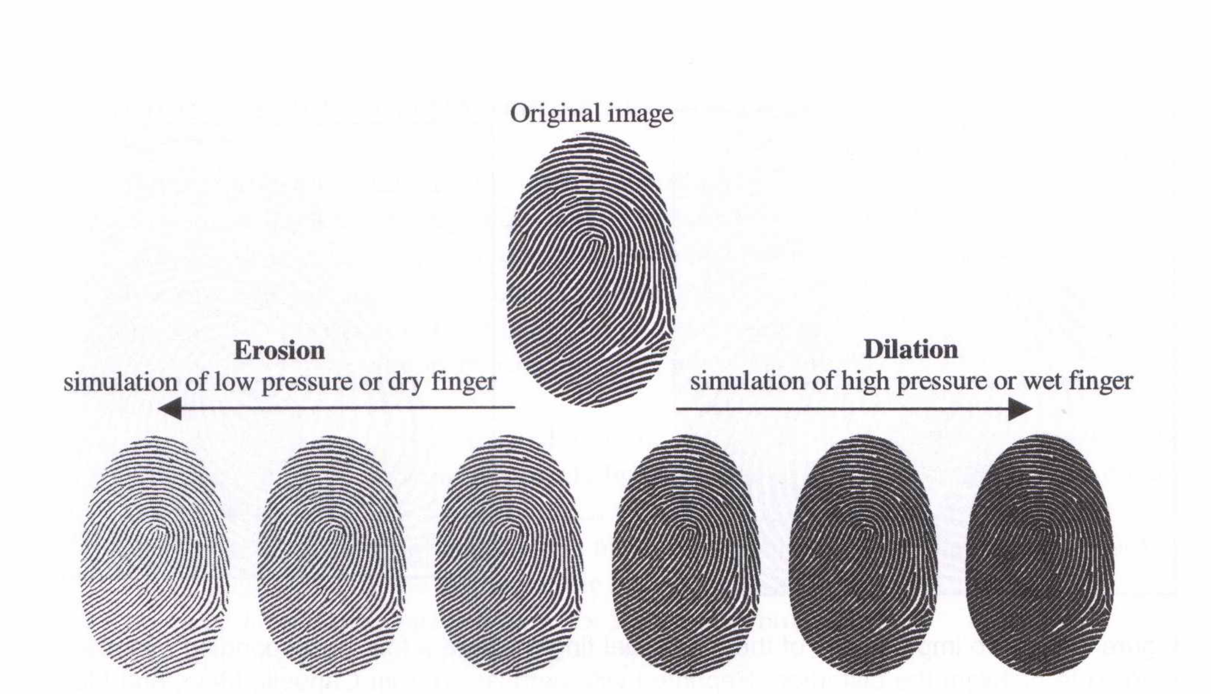
$$e^{-\left(\left(\frac{3}{2f}\right)^2 / 2\sigma^2\right)} = 10^{-3}.$$

$$g(x, y : \theta, f) = e^{-\left((x^2+y^2)/2\sigma^2\right)} \cdot cos\left[2\pi \cdot f \cdot (x \cdot sin\theta + y \cdot cos\theta)\right],$$

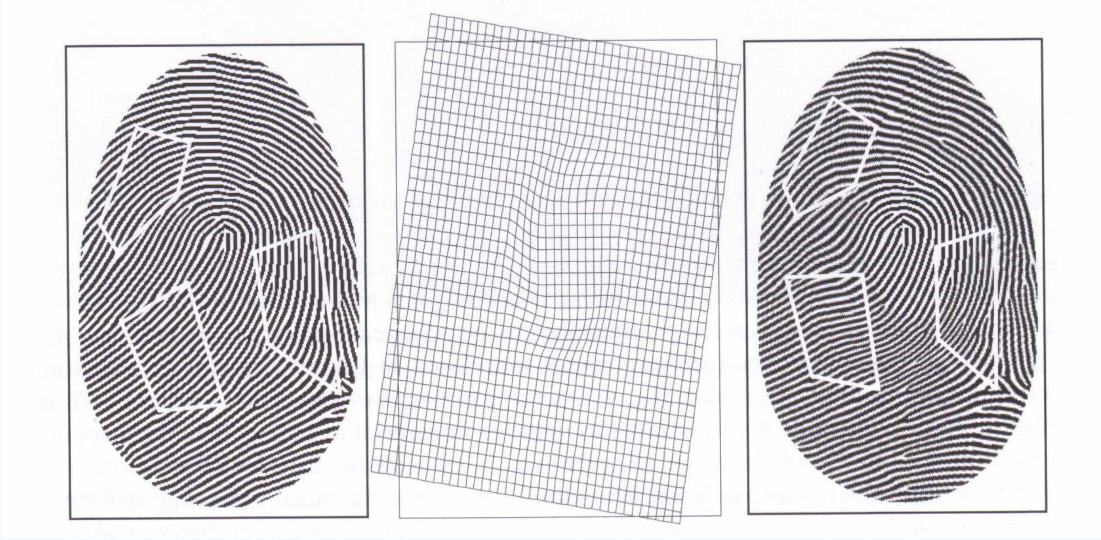
Ridge line



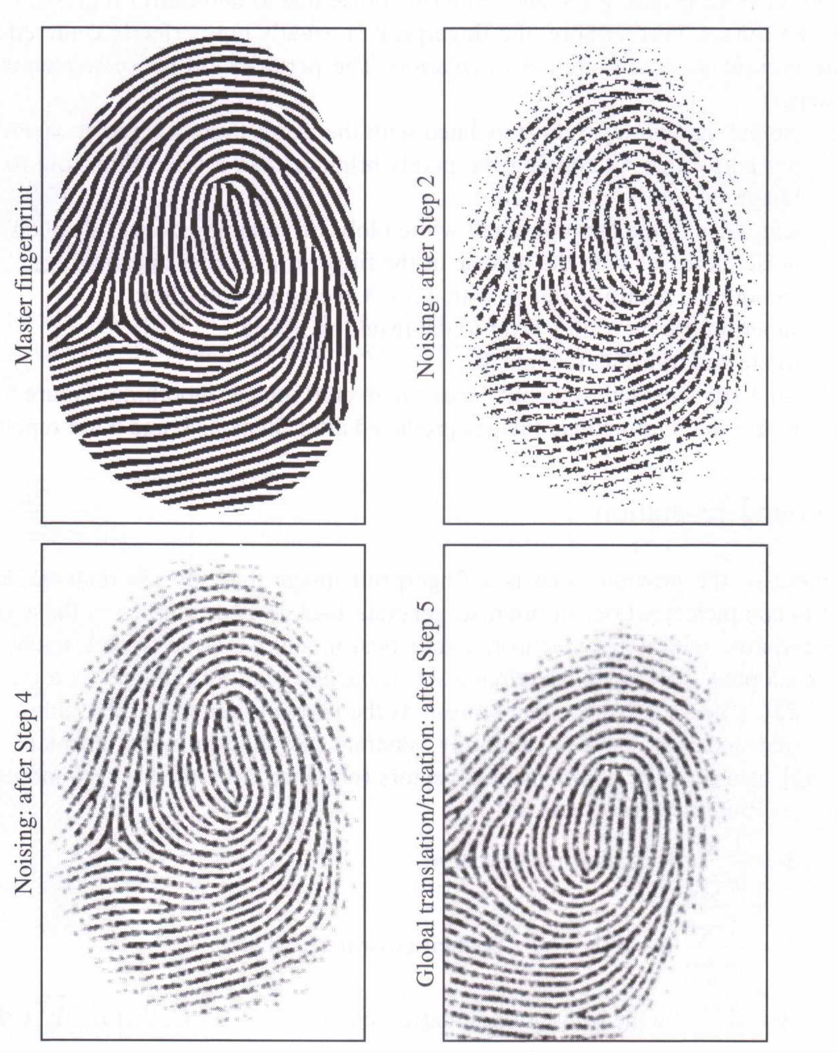
Ridge thickness



Distortion



Perturbation & Translation



Background

- $\overline{\mathbf{b}} = \frac{1}{m}\sum_{\mathbf{b}\in \mathrm{B}} \mathbf{b}$ be their mean vector;
- $\mathbf{C} = \frac{1}{m}\sum_{\mathbf{b}\in \mathrm{B}} (\mathbf{b}-\overline{\mathbf{b}})(\mathbf{b}-\overline{\mathbf{b}})^T$ be their covariance matrix;
- $\mathbf{\Phi} \in \Re^{n\times n}$ be the orthonormal matrix that diagonalizes $\mathbf{C}$; that is, $\mathbf{\Phi}^T\mathbf{C}\mathbf{\Phi} = \mathbf{\Lambda}$,

  $\mathbf{\Lambda} = Diag(\lambda_1, \lambda_2, ..., \lambda_n)$, $\mathbf{\Phi} = [\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, ..., \boldsymbol{\varphi}_n]$,

where λ_i and $\boldsymbol{\varphi}_i$, $i = 1..n$ are the eigenvalues and the eigenvectors of $\mathbf{C}$, respectively.

1. a k-dimensional vector $\mathbf{y} = [y_1, y_2, ..., y_k]$ is randomly generated according to k normal distributions: $y_j = N\left(0, \lambda_{i_j}^{1/2}\right)$, $j = 1..k$;
2. the corresponding n-dimensional vector $\mathbf{b}$ is obtained as: $\mathbf{b} = \mathbf{\Phi}_k \mathbf{y} + \overline{\mathbf{b}}$.

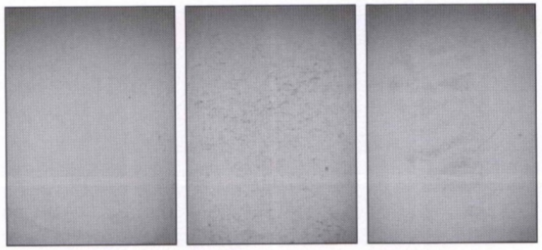
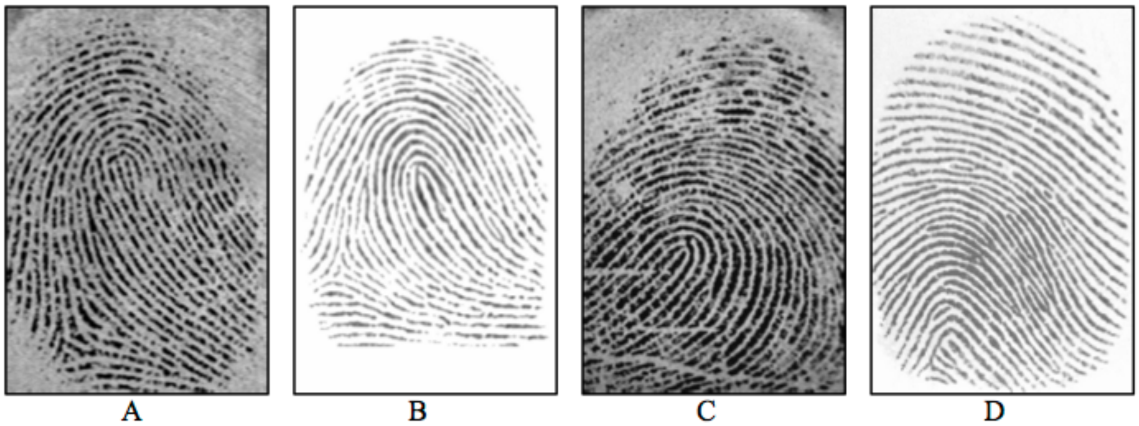


Figure 6.22. Examples of background-only images (acquired from an optical scanner) used for training the background generator.

Figure 6.23. Three synthetic images with backgrounds generated according to the model (the parameters used for training are m = 65 and k = 8).

Is it working?

- 1st experimtn, 90 experts recognizing synthetic FP -> 23%



- 2nd experiment
- DB1,2,3-real, DB4 synthetic, FMR FNMR