Image Based Biometry

Peter Peer, Žiga Emeršič, Blaž Meden

Ljubljana, January 2017

v095

biometry
/bʌɪˈɒmɪtri/

noun
noun: **biometry**; noun: **biometrics**

the application of statistical analysis to biological data.

Translate biometry to Slovenian

1. biometrija

Use over time for: biometry

Biometry?!?

What is this course all about?

- Biometrics:
  “the measurement of the properties of
  living beings”
- Greek: βίος = “life”, μέτρον = “measurement”
- Or maybe it’s:
  “the use of physical or behavioral properties of
  human beings for automatic identity recognition”
- Image-based!

Course Info/Overview

The course relies mostly on computer vision, as most biometrics technologies are based on it. Students interested in cutting edge technology, much of which is still in a research stage, are the intended target for the course. The main content (will evolve due to developments in the field):

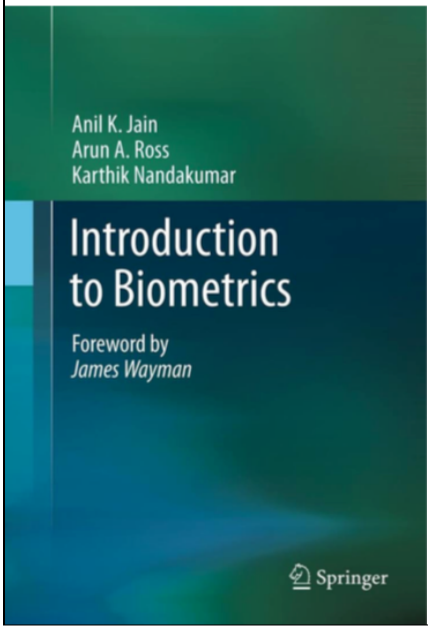
1. Biometry basics
2. Biometrical modalities
3. Structure of a typical biometric system
4. Recognition/verification/identification
5. Metrics
6. Conditions for correct comparisons of the systems (databases, frameworks)
7. Performance and usefulness of the systems
8. Computer vision as the foundation of the biometric systems
9. Fingerprint
10. Iris
11. Face
12. Gait
13. Ear
14. Multi-biometric systems / multi-modality / fusions
15. Key problems of modalities/systems (research challenges)

The lectures introduce the approaches and explain their operation. At tutorial the knowledge is applied to practical problems in Matlab and open source tools.

Why are YOU interested in it?

- Related background?
- Motivation?
- Expectations?

Literature ?!?



+ some other books &
a bunch of scientific articles

Including:

1. EMERŠIČ, Žiga, ŠTRUC, Vitomir, PEER, Peter. **Ear Recognition: More than a Survey**, *Neurocomputing*, 2016
2. PEER, Peter, EMERŠIČ, Žiga, BULE, Jernej, ŽGANEC GROS, Jerneja, ŠTRUC, Vitomir. **Strategies for exploiting independent cloud implementations of biometric experts in multibiometric scenarios**. *Mathematical problems in engineering*, ISSN 1024-123X. [Print ed.], 13 Mar. 2014, vol. 2014, str. 1-15
3. KOVAČ, Jure, PEER, Peter. **Human skeleton model based dynamic features for walking speed invariant gait recognition.** *Mathematical problems in engineering*, ISSN 1024-123X. [Print ed.], Jan. 2014, vol. 2014, str. 1-15
4. KLOPČIČ, Uroš, PEER, Peter. **Fingerprint-based verification system : a research prototype.** V: LETA, Fabiana R. (ur.), CONCI, Aura (ur.). IWSSIP 2010 : proceedings. [S. l.]: EdUFF Editora da Universidade Federai Fluminese, 2010, str. 150-153
5. VRČEK, Gorazd, PEER, Peter. **Iris-based human verification system : a research prototype.** V: 16th International Conference on Systems, Signals and Image Processing, 2009 [and] IWSSIP 2009, 18-20 June 2009, Chalkida, Greece. Piscataway: IEEE, cop. 2009, str. 1-4
6. BULE, Jernej, PEER, Peter. **Technical, legal, economic and social aspects of biometrics for cloud computing.** *Journal of information and organizational sciences*, ISSN 1846-3312. [Print ed.], 2014, vol. 38, no. 2, str. 83-95

Tutorials

- Hands on practice!
- 3 assignments/homeworks + 1 project
- (Really) short reports with IMRAD structure (with proper professional and scientific terminology)
- 2/3 of contact hours in the form of labs,
  1/3 in the form of a seminar
- What is a seminar? Panels/discussions based on articles, presentations etc.

Predavanja Lectures	Seminar Seminar	Vaje Tutorial	Klinične vaje Laboratory work	Druge oblike študija Field work	Samost. delo Individ. work	ECTS
45	10	20	/	/	105	6

Grading

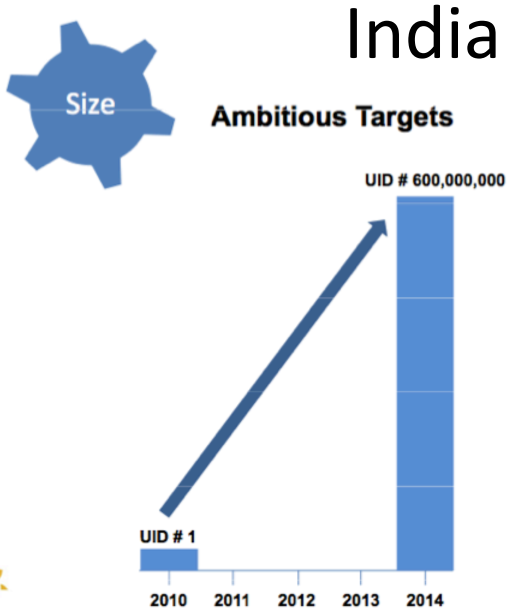
Weight (in %)	**Assessment:**
	Type (examination, oral, coursework, project):
67%	**Continuing** (assignments/project, presentations)
33%	**Final**: (written or oral exam)
	Grading: 6-10 pass, 1-5 fail.

Motivation … through Pictures

India from 2009 … collected information

- KYR Fields – Name, Address, Gender, DOB
- Photo & Address Verification
- Photo
- 10-fingerprints on Slap scanner
- Iris Scan

India … Goals



- July 2015:
  879M enrollments
- By December:
  1 Billion ?!?
- Problems:
  legislation, privacy, security, ethics

Understanding the Term Biometrics

- "the use of physical or behavioral properties of human beings for automatic identity recognition"

- Something characteristic **only** to me (eg. my face, my handwriting, my voice)
- **Not** something I know (eg. a password or PIN)
- **Not** something I have (eg. a key or authentication token)

Understanding the Term Biometrics

- "the use of physical or behavioral properties of human beings for automatic identity recognition"

- We need a **living subject**

Understanding the Term Biometrics

- "the use of physical or behavioral properties of human beings for automatic identity recognition"
- Why do we want computers to do this?
- **High throughput // Repeatability // Predictability**

Biometric Recognition Types

- Positive Recognition
- A sample represents a subject **known** to the system (i.e., already registered)
- Negative Recognition
- A sample represents a subject **unknown** to the system (i.e., not yet registered)

Biometric Modality

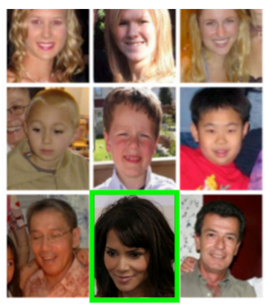
- A **single** physical or behavioral property we use for biometric recognition

- Fingerprint
- Iris
- Face

Probe & Gallery

- Probe: A sample presented to a biometric system
- Gallery: A collection of enrolled templates

"Who is this?"



"Halle Barry"

Authentication Types

- Pair matching: Do these two images match?
- Verification: Does this sample of Halle Barry match the one in our system? (1:1)
- Identification: Does this person exist in our system? (1:n)
- Negative authentication: I'm not the subject *X* / I'm not a member of group **X**
- Deduplication: Do we have to enroll you?

Enrollment Pipeline

- New user comes in
- Acquisition ⟹
  Feature extraction ⟹
  Template generation

Recognition Pipeline

- We need to make a decision

- Acquisition $\Longrightarrow$
  Feature extraction $\Longrightarrow$
  Matching $\Longrightarrow$
  Labeling

Acquisition

A biometric sample is the raw or preprocessed data collected from a subject by a sensor

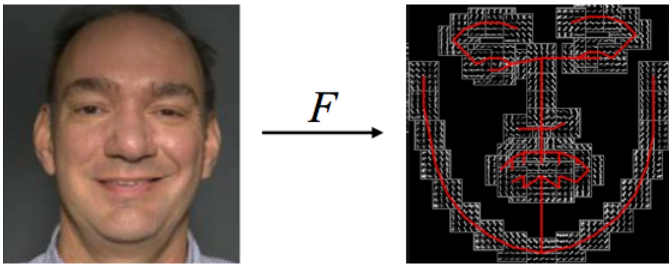
Let $I \in \mathbb{R}^{v}$ be an image
v = number of pixels

Features

$$x = F(I, \varphi_F),\ x \in \mathbb{R}^D$$

feature extractor assumptions

y = label (if we know it)



Templates

- **Class** = $y \in \mathbb{N}$
- **Labeled** training data = $X_y = \{(x_1, y_1), \ldots\}$
- Number of feature vectors = $m = |\{(x_1, y_1), \ldots\}| \geq 1$
- Model specific assumptions = φ_M
- **Model** = $M_y = f(X_y, \varphi_M)$

(If the template is a learned model then $m > 1$)

Matching

Compare x_0 to M_y

$$s_y = R(x_0, M_y(X_y, \varphi_M), \varphi_R), s_y \in \mathbb{R}$$

Similarity Score Recog. Function Matching Assumptions

Labeling

- x_0 was compared to all models $\{M_1, \ldots\}, n \geq 1$
- $$C = L(S, \varphi_L), C \subsetneq \mathbb{N}$$

  Labeling Function Labeling Assumptions

- Ranked set of labels = $C = \{y_1^*, \ldots\}$
- Non-match label = y_0^*

Enrollment – the user's perspective

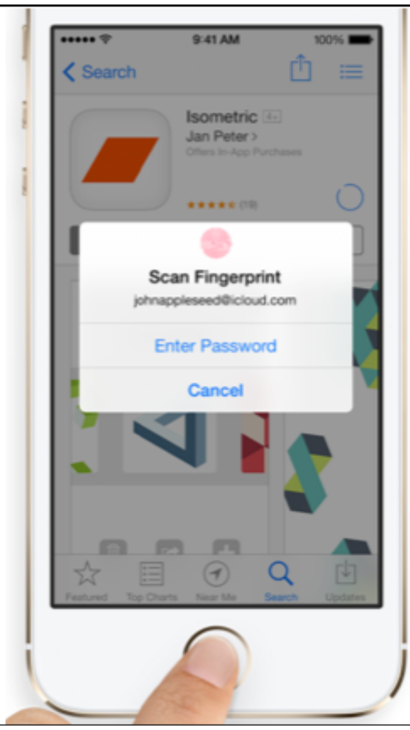
Multiple measurements

- Quality control
- Select best samples
- Merge samples

Enrollment – the security engineer's perspective

- Is meta-data associated with the record?
- Are multiple templates stored for a user?
- Operator supervision?
- How is the template stored?
  - Encryption
  - Template protection

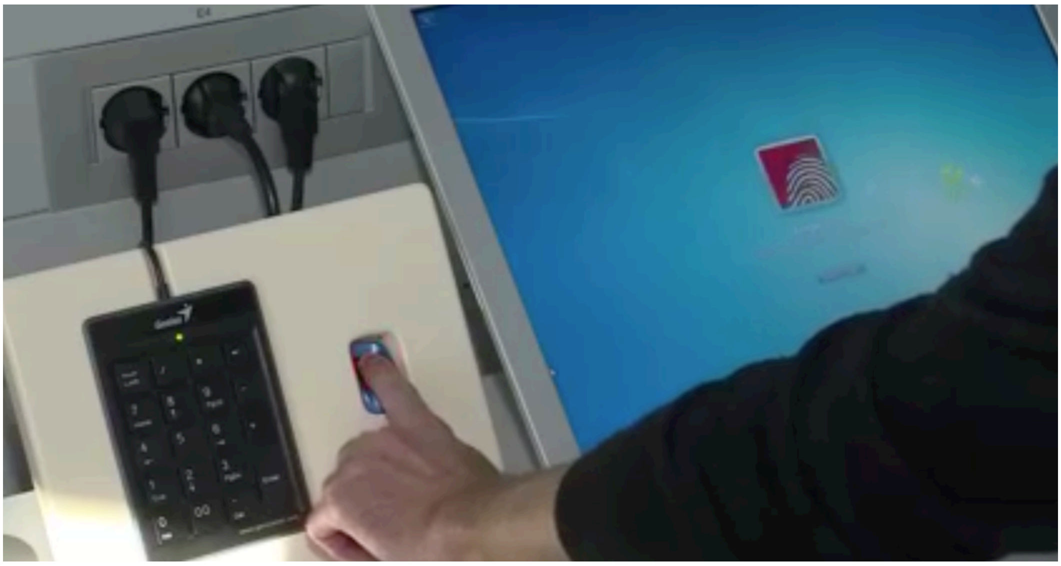
Authentication – the user's perspective



Authentication – the security engineer's perspective

- How is the stored template retrieved?
  - On device
  - Local server
  - Cloud
- Operational matching threshold?
- How many attempts?
- Speed vs. security tradeoff?
- Auditing?

Multi-Factor Authentication



Error Statistics

- **False Match (Type I Error):** an <u>impostor</u> sample matched a reference template

Error Statistics

- **False Non-Match (Type II Error):** a genuine sample did not match a reference template

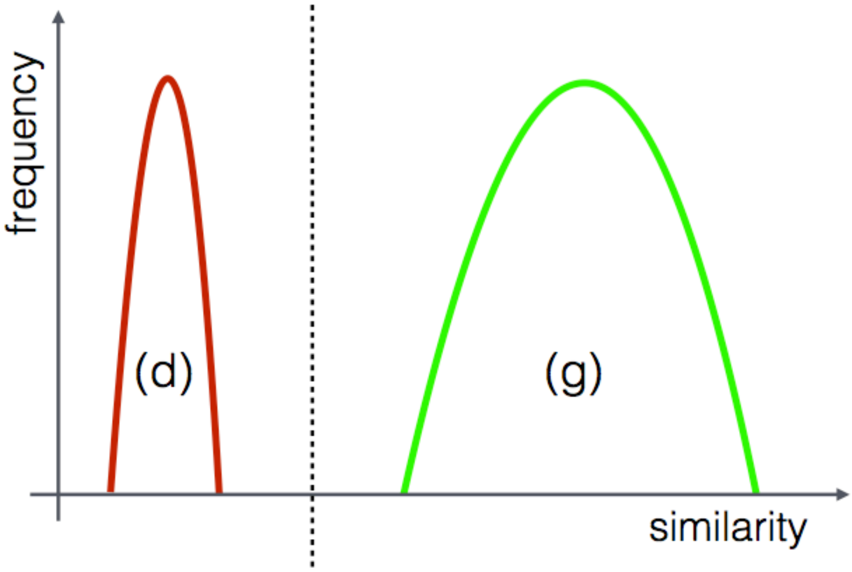
Decisions

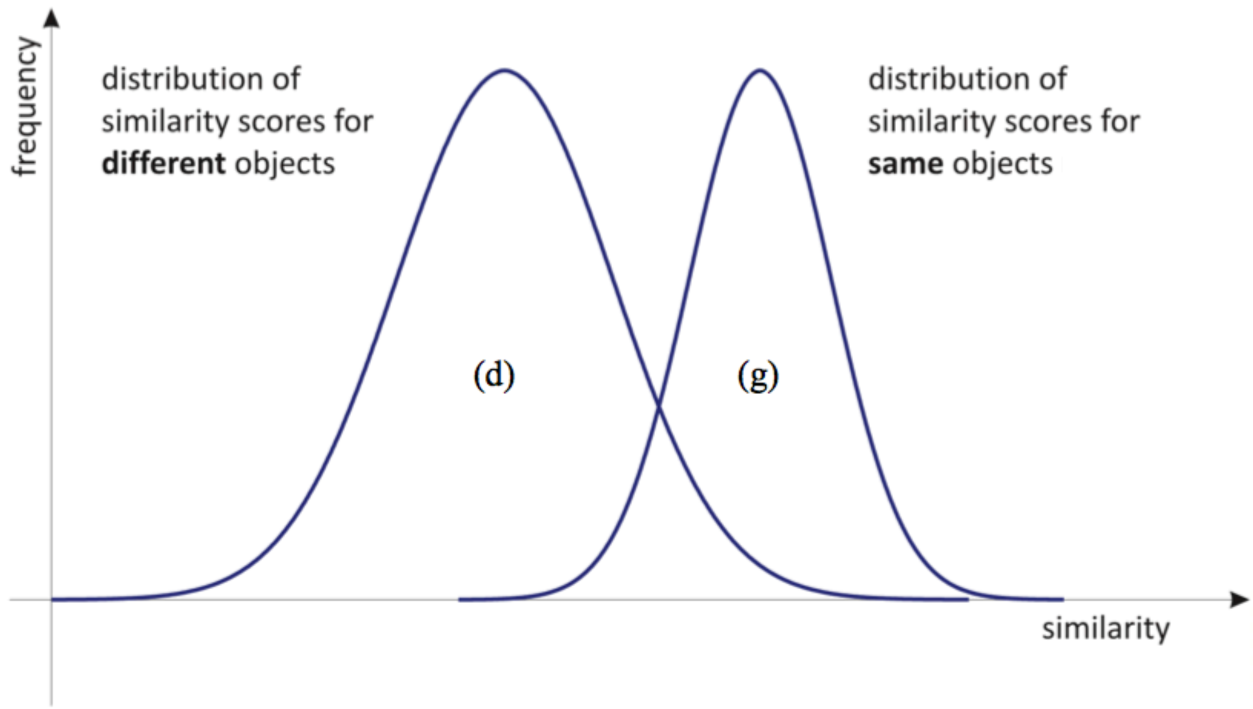
- **Similarity** score: higher is better
- **Distance** score: lower is better (0 for self match)
- Decision threshold: τ

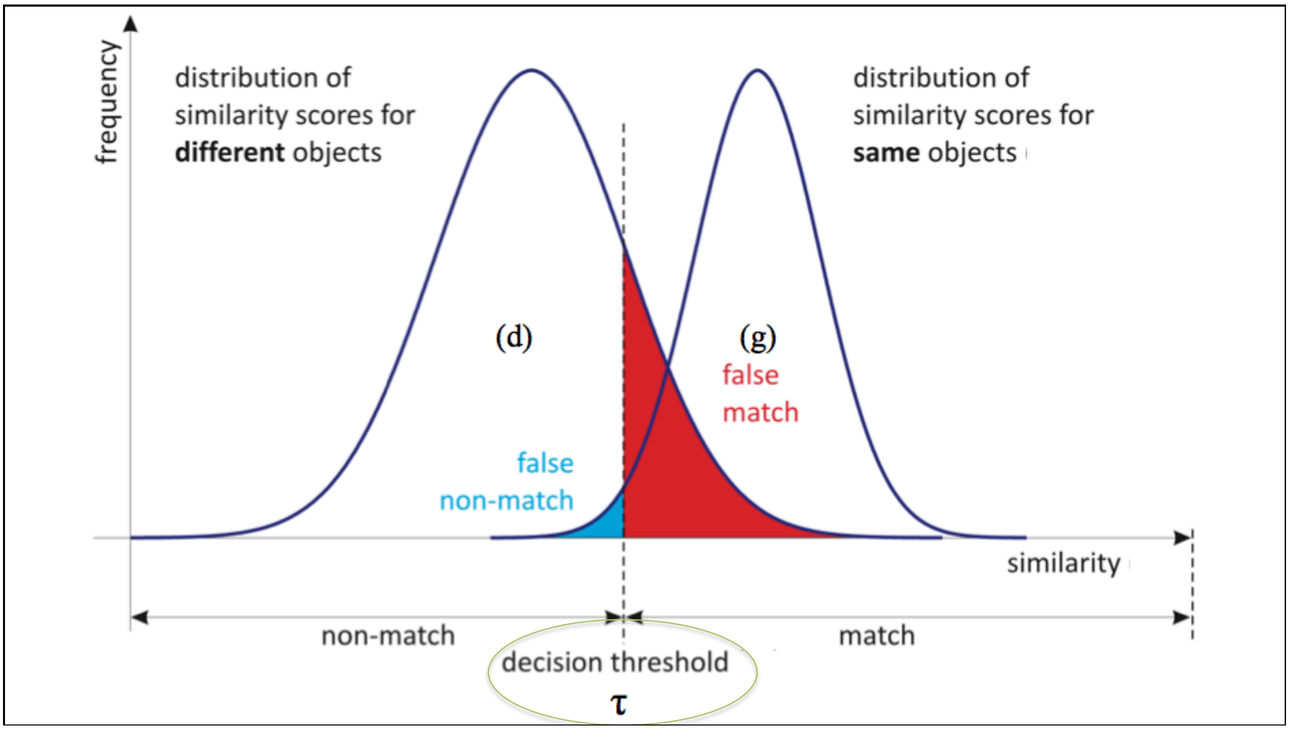
- Similarity Score Match: $s_y > \tau$
- Distance Score Match: $s_y < \tau$

Score Distributions

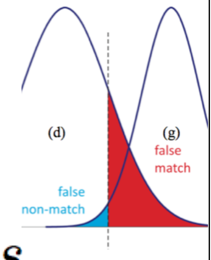
Ideal world: different (d) and same (g) score distributions well-separated







False Non-Match Rate



- If we know g:

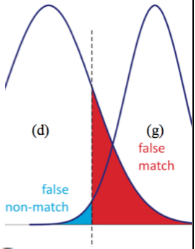
$$\mathrm{g}_{\mathrm{FNM}}(\tau) = \int_{-\infty}^{\tau} \mathrm{g}(s)ds$$

- But typically we don't, thus:

$$\hat{\mathrm{g}}_{\mathrm{FNM}}(\tau) = \mathrm{FNMR}(\tau) = \frac{\text{Number of false non-matches for } \tau}{\text{Number of all genuine comparisons}}$$

- Calculate with a **benchmark data set**

False Match Rate



- If we know d: $$\mathrm{d}_{\mathrm{FM}}(\tau) = \int_{\tau}^{\infty} \mathrm{d}(s)ds$$
- But typically we don't, thus:

$$\hat{\mathrm{d}}_{\mathrm{FM}}(\tau) = \mathrm{FMR}(\tau) = \frac{\text{Number of false matches for } \tau}{\text{Number of all impostor comparisons}}$$

- Calculate with a **benchmark data set**

Measurement Tradeoffs

Problem 1: Biometric **verification** – Why am I rejected?

Large throughput volume is a problem.

- Example: frequent flyer verification from face image
- Assume a system where each person is 1:1 verified at an airport kiosk with 5,000 people per hour (14h/day) requesting access (Newark (NJ) airport hourly passenger volume)
- The system has a FMR of 0.1% and a **FNMR of 2%**
  - 100 people **per hour will fail** to be verified
  - 1,400 people **per day will fail** to be verified

Measurement Tradeoffs

Problem 2:
Biometric **(mis)identification** – Why am I delayed as a "suspect"?

Large watch lists exacerbate the problem.

- Example: Faces checked against terrorist watch list
- Assume a system that **checks each person's face against a watch list** of 1,000 suspects (assume Newark airport: 5,000 people per hour & 14h/day)
- The system has a FMR of 0.1%
  - Over 70K false matches will occur per day from 1K watch list
- Note: 2011 US TSC TSDB list was > 450K

Measurement Tradeoffs

Problem 3: Biometric **identification** – "Who can I be today?"

Biometric databases are a security problem.

- Example: Faces checked against government database
- Criminals gain access to a large face database and start looking for someone their gang can use to steal an identity
- With a FMR of 0.1%, a single face will match (0.001 * 6,000,000) = **6,000** people in the DC area. With a "gang" of 10 or **100** what can they do?
- Note: Colorado DMV records have fingerprint, photo and all driving information

Doppelganger Attack

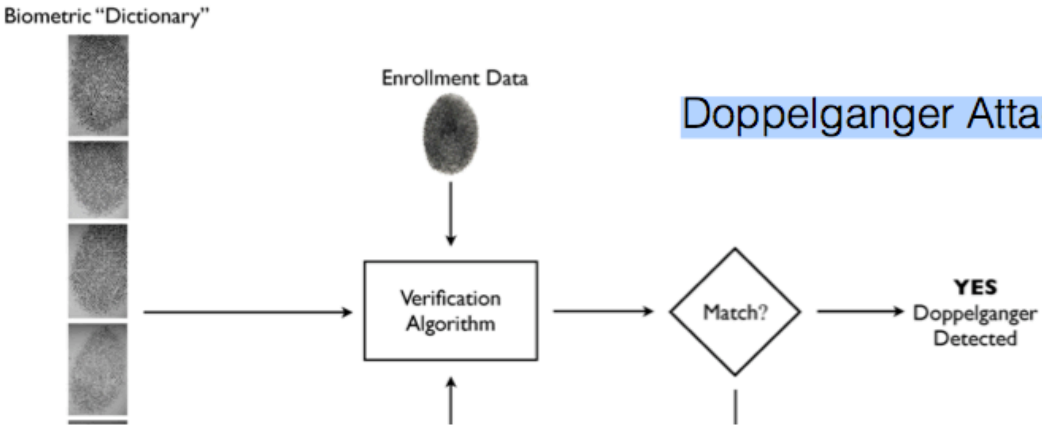


Fig. 3.4 The Doppelganger Threat. What happens when a large database of biometric data is stolen, hacked, or sold? For example, consider a database containing unique records for 4 million individuals. If the matching system is operating at a False Accept Rate (FAR) of 1 in 1,000, a malicious attacker in possession of the database may, on average, be given a choice of approximately 4,000 identities to use to compromise the matching system. Even at a FAR of 1 in 1,000,000, the attacker still gets four choices, on average, to compromise the matching system

Measurement Tradeoffs

False matches during duplicate checks require additional processing

Number of False Hits ***Per Search***
(*i.e.*, each person being checked)

FAR	50 Million	500 Million
1%	500,000	5M
0.1%	50,000	500,000
0.01%	5,000	50,000
0.001%	500	5,000
0.0001%	50	500

Total *false matches* that must be resolved in determining duplicates

FAR	50 Million	500 Million
1%	25Trillion	25000Trillon
0.1%	2.5Trillion	2500Tillion
0.01%	250Billion	250Trillion
0.001%	25Billon	2.5Trillion
0.0001%	2.5Billion	250Billion

How to address these false hits?

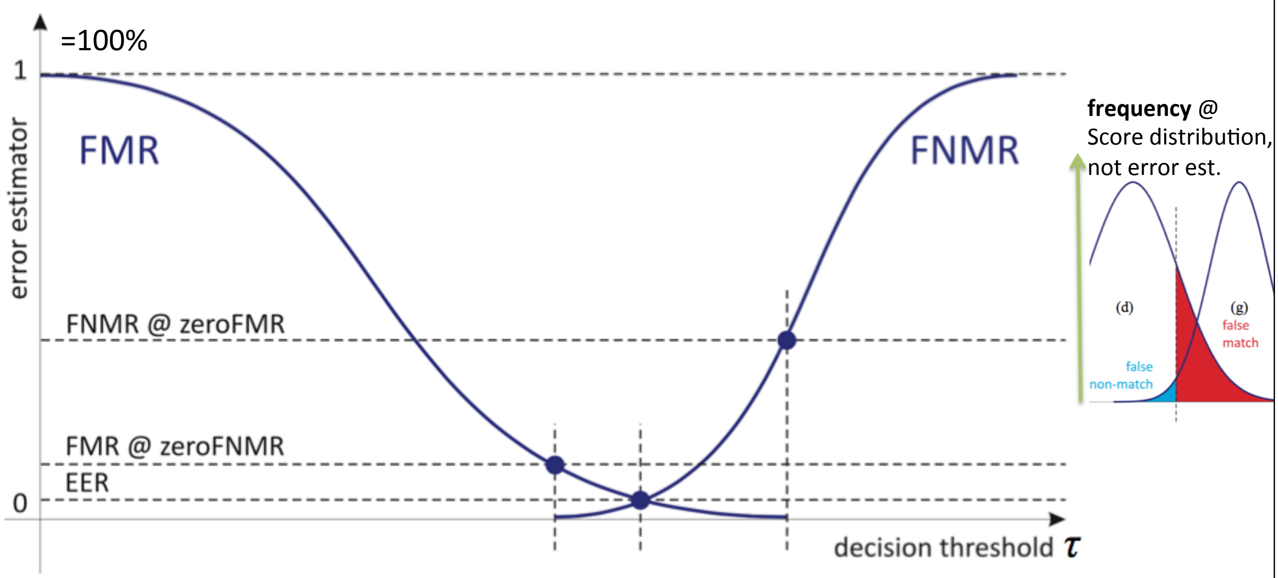
- Manually? (Note: 2.5 Billion minutes is about 4,750 years!)
- *Automatically use **another biometric** and hope to reduce FMR (FAR)*

How to Measure **Correct** Decisions?

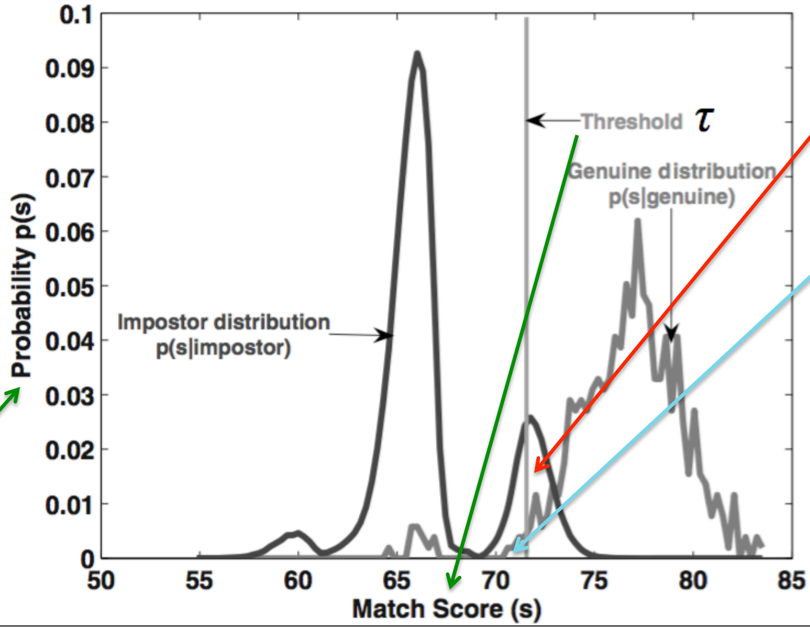
Maximize — GMR = 1 - FNMR — Minimize

Maximize — GNMR = 1 - FMR — Minimize

Error Estimators as a Function of τ



BTW: Score Distribution – Real World Case

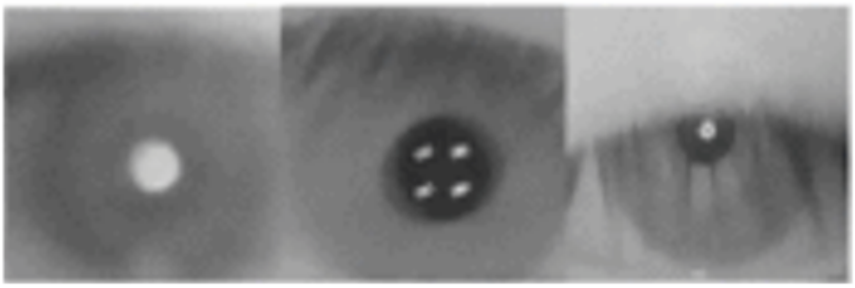


Calculated from the frequency

The FAR and the FRR cannot be reduced *simultaneously* by adjusting the threshold!

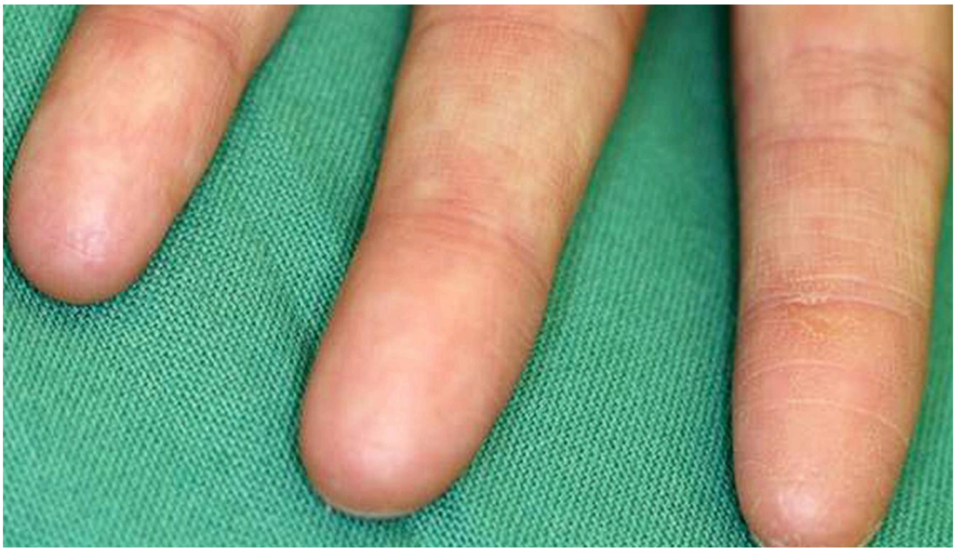
Failure to Acquire

Falsely rejected biometric samples (due to problems at **acquisition**)



Failure to Enroll

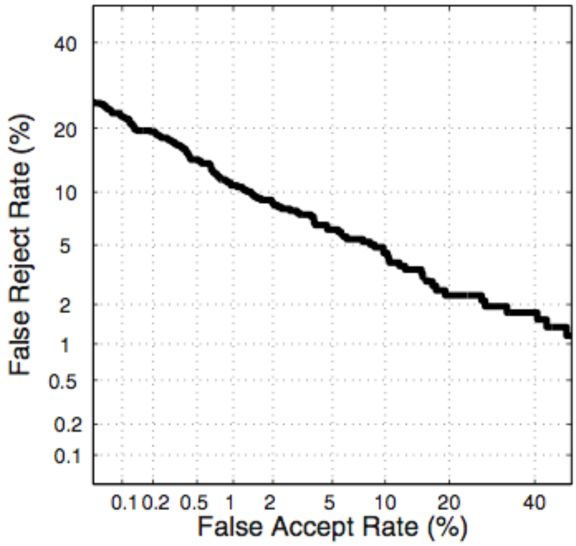
Falsely rejected biometric samples (due to problems at **enrollment**)



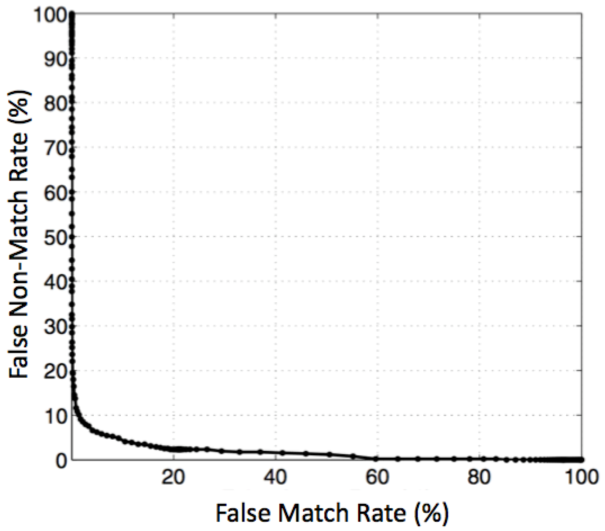
FNMR and FMR dependence

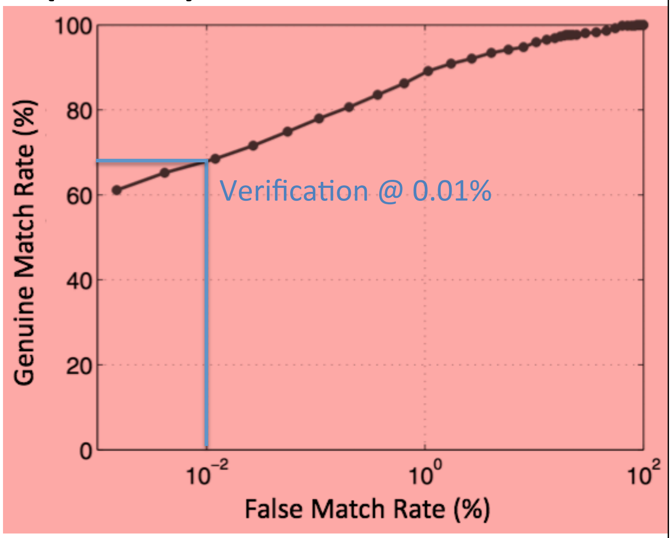
- The same biometric system can be operated at different thresholds depending on the changing security level or different requirements of different applications
- **The FAR and FRR at different values of threshold are measured and summarized in the form of curves**
  - Detection Error Tradeoff (DET) curve
  - Receiver Operating Characteristic (ROC) curve

Detection **E**rror **T**radeoff Curve (DET)



Receiver **O**perating **C**haracteristic Curve (ROC)

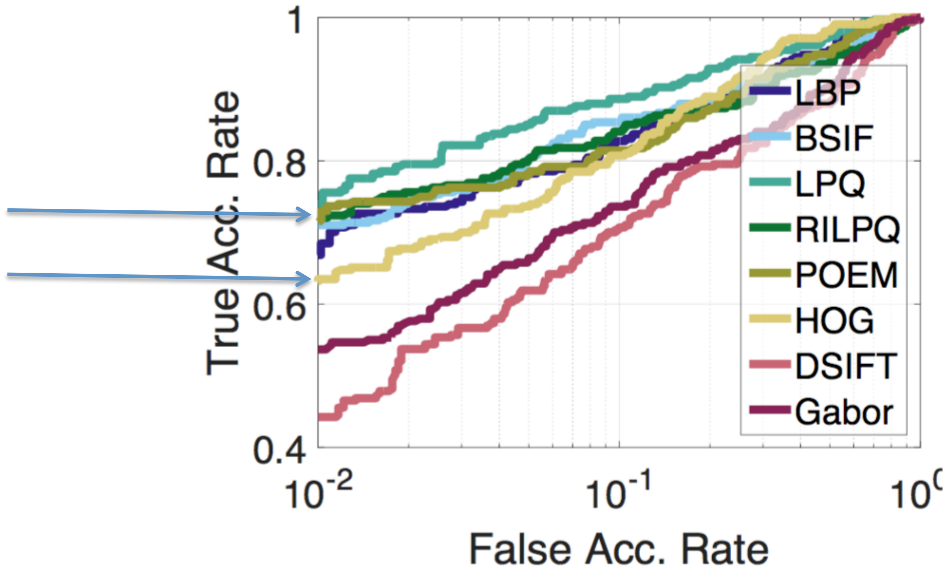




ROC reveals …

- The best way to compare the performance of two biometric systems is to examine their ROC curves
- If the FRR of one biometric system (A) is consistently lower than the FRR of the other system (B) for corresponding values of FAR, one can conclude that the matching performance of biometric system A is better than that of B

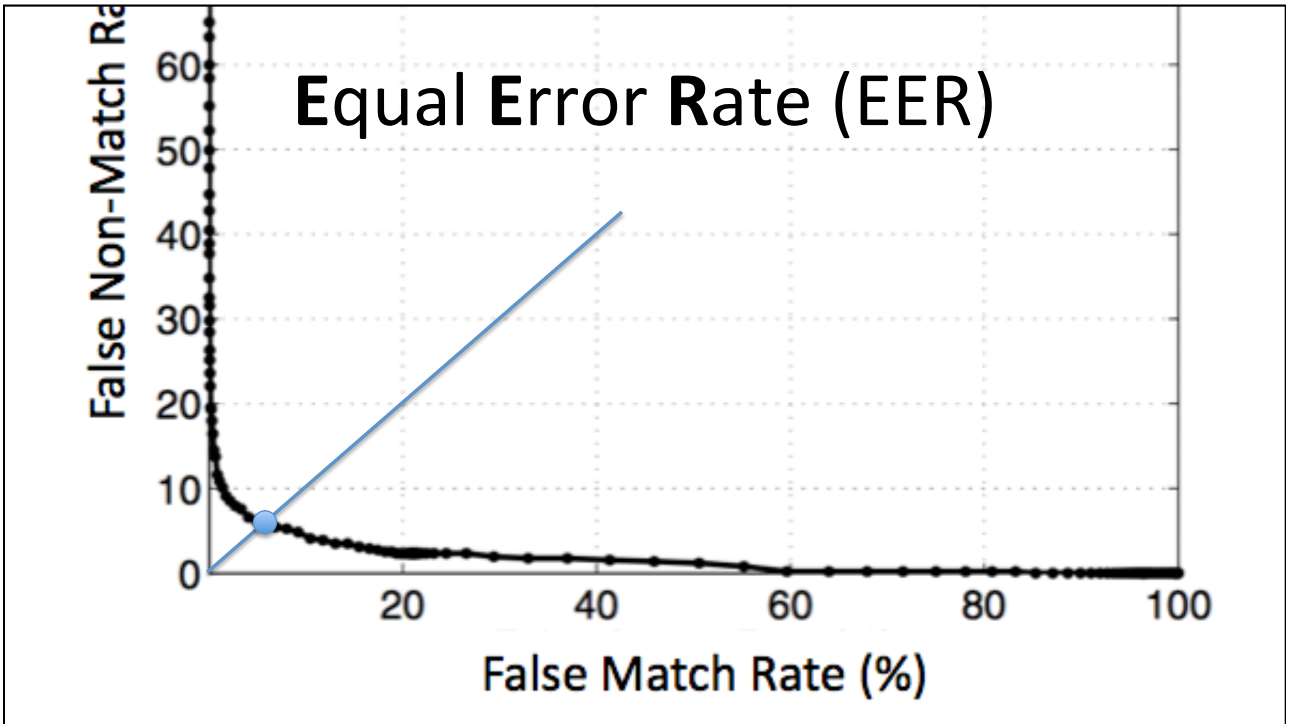
Area **U**nder the ROC **C**urve (AUC)



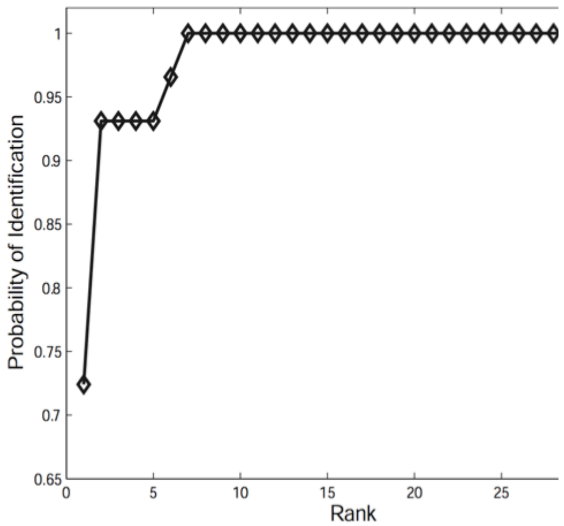
0.9 - 1.0 = excellent
0.8 - 0.9 = good
0.7 - 0.8 = fair
0.6 - 0.7 = poor
0.5 - 0.6 = fail

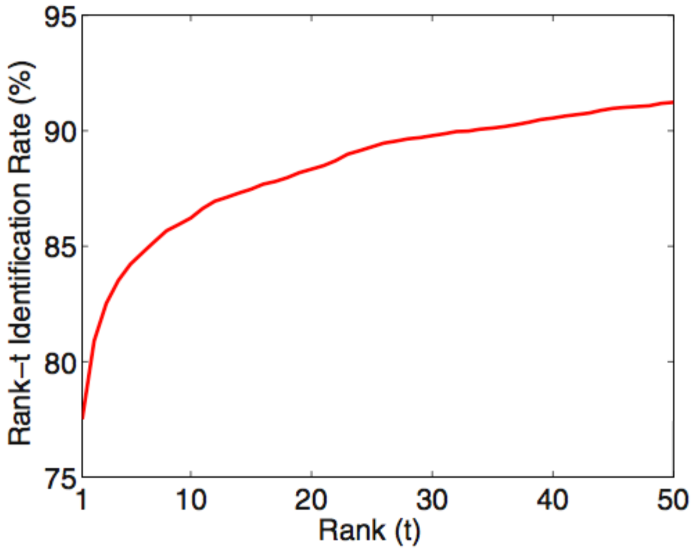
use when you can't pick a threshold

Equal Error Rate (EER)



Cumulative Match Curve (CMC)





Accuracy

Accuracy =

$$\frac{|\text{genuine matches}| + |\text{genuine non-matches}|}{|\text{genuine matches}| + |\text{genuine non-matches}| + |\text{false matches}| + |\text{false non-matches}|}$$

$$\text{Error} = E = 1 - \text{Accuracy}$$

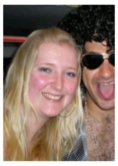
Is there something wrong with it?

- Example: sliding window face detector
- 11 faces, 1K negatives
- Majority class (no face) classifier: 98.9% accurate!
- Face classifier: 1.1%! :-/

Recall

$$\text{Recall} = \frac{\{\text{Relevant Images}\} \cap \{\text{Retrieved Images}\}}{\{\text{Relevant Images}\}}$$

Probe



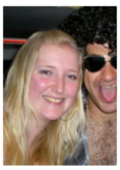
Gallery

Recall = 50%

Precision

$$\text{Precision} = \frac{\{\text{Relevant Images}\} \cap \{\text{Retrieved Images}\}}{\{\text{Retrieved Images}\}}$$

Probe



Precision = 33%

F-measure

Calculate harmonic mean of precision and recall:

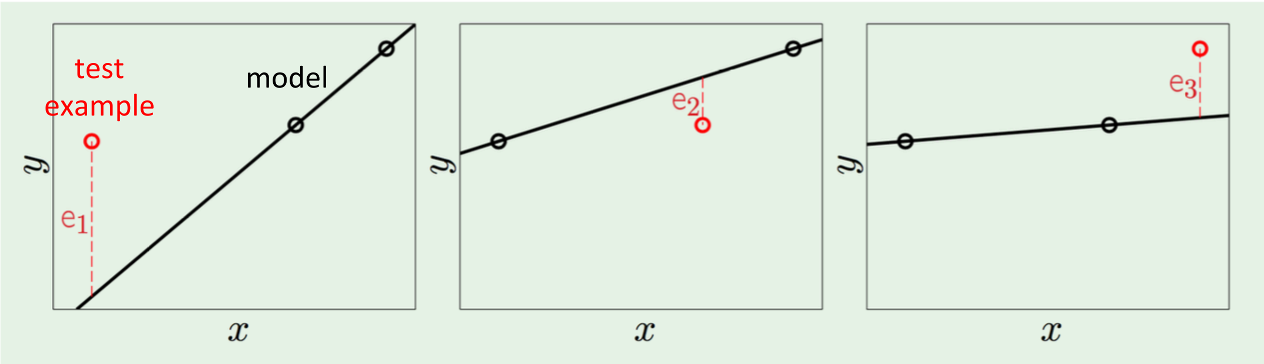
$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Evaluating … using Cross Validation

labeled training data

- N training samples: $X_y = \{(x_1, y_1), \ldots, (x_N, y_N)\}$
- Build a model $M_y = f(X_y, \varphi_M)$
  that minimizes empirical error on the training set
- Test the model:
  - Leave one out
  - K-fold

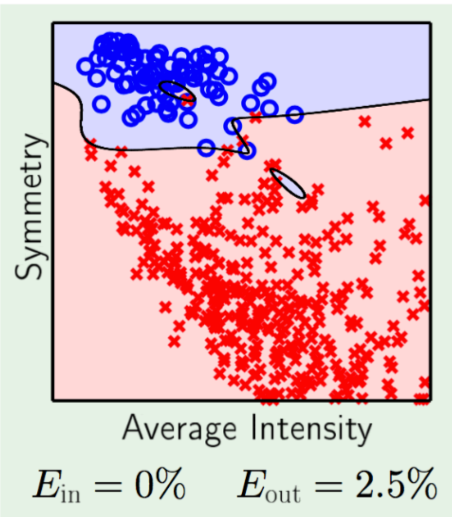
Why?



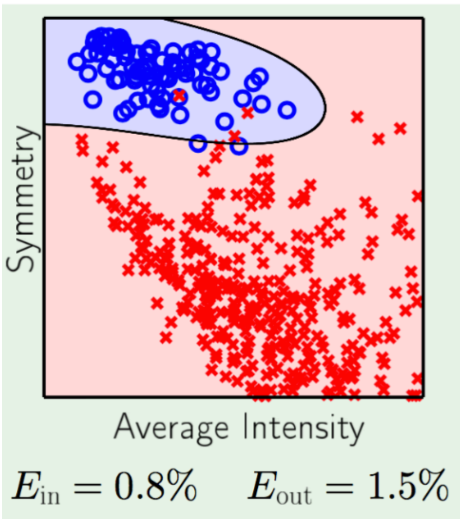
$$E_{cv} = \frac{1}{3}\,(e_1 + e_2 + e_3)$$

Why?

Without Validation



With Validation



Leave One Out

$$f_n = (x_1, y_1), \ldots, (x_{n-1}, y_{n-1}), \cancel{(x_n, y_n)}, (x_{n+1}, y_{n+1}), \ldots (x_N, y_N)$$

$$e_n = E_{\text{val}}(f) = e(f(x_n), y))$$

$$E_{cv} = \frac{1}{N} \sum_{n=1}^{N} e_n$$

K-Fold Cross Validation

- In general:
  N/*K* training sessions on *N*-*K* samples each

- In practice:
  10-fold => *K*=*N*/10

Misunderstandings in Biometry ;-)

- **<u>Uniqueness</u>** -> the physical trait itself may not be unique
  - "Only once during the existence of our solar system will two human beings be born with similar finger markings" - *Harper*'s headline, 1910.
  - "Two like fingerprints would be found only once every 1048 years" - *Scientific American*, 1911.
  - Uniqueness of biometric modalities has not been scientifically clearly established
- **<u>Permanence</u>** -> also not an established scientific fact

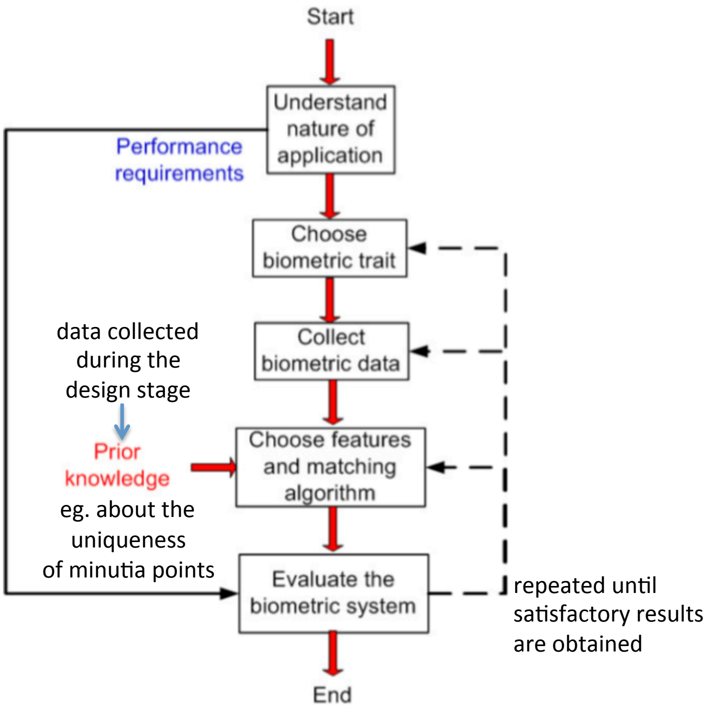
Facts

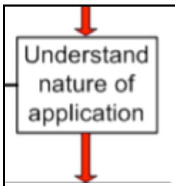
- Fact 1: biometric systems rely only on the **digital** measurements
- Fact 2: sensing introduces **variations** in the samples of the same biometric trait of a user obtained over a period of time
- Fact 3: feature sets obtained from different samples of the same biometric trait of a user are **seldom** identical; we look for **close** and not perfect match

Intra-User Variations

The same person?

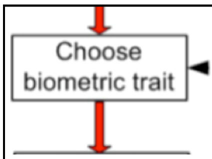
The Design Cycle of a Biometric System





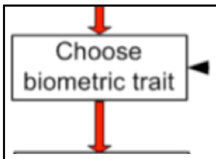
Nature of the Application

1. Verification vs. identification
2. Cooperative vs. non-cooperative users
3. Overt vs. covert deployment
4. Habituated users vs. non-habituated
5. Attended vs. unattended operation
6. Controlled vs. uncontrolled operation
7. Open vs. closed system



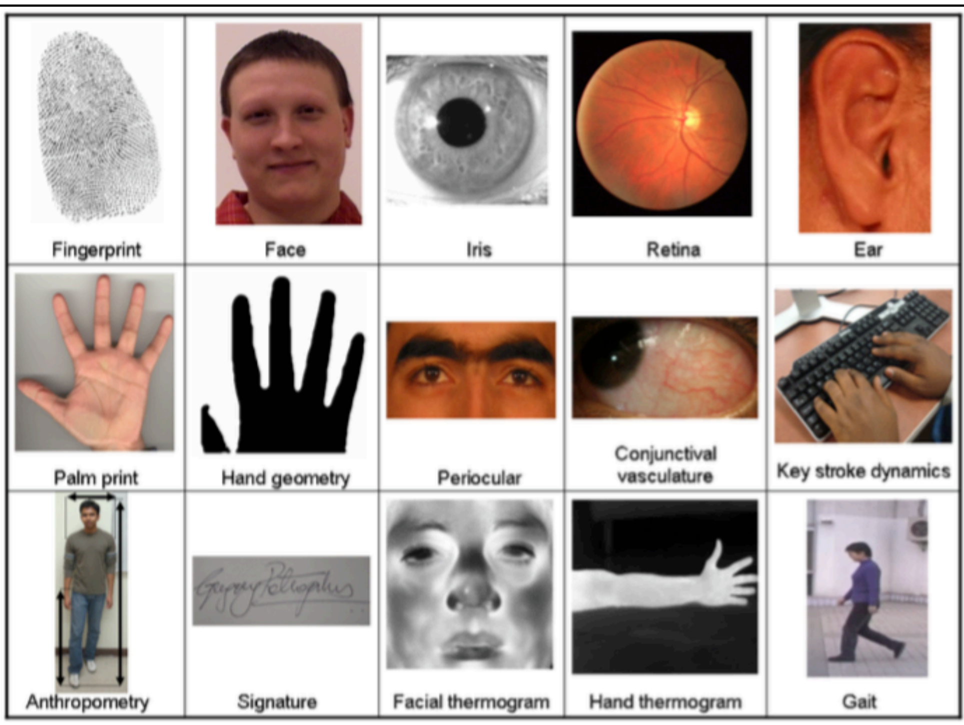
Choice of Biometric Trait

1. Universality – possess the trait
2. Uniqueness – sufficiently different traits
3. Permanence – invariant over a period of time
4. Measurability – possible to acquire and digitize
5. Performance – metric + resources & throughput
6. Acceptability – willingness to present the trait
7. Circumvention – imitate traits / evade recognition



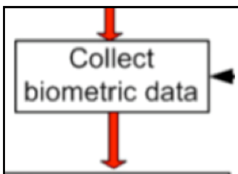
Choice of Biometric Trait

- No single biometric is expected to effectively meet all the requirements
- No biometric is **ideal** but a number of them are **admissible**
- Which ones?



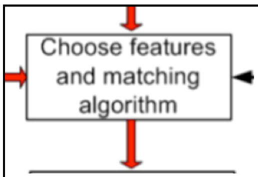
Summary of the Error Rates

Biometric Trait	Test (competition)	Test Conditions	False Reject Rate	False Accept Rate
Fingerprint	FVC 2006	Heterogeneous population including manual workers and elderly people	4.2%	0.1%
	FpVTE 2003	U.S. government operational data	0.6%	0.1%
Face	FRVT 2006	Controlled illumination, high resolution	0.8-1.6%	0.1%
Voice	NIST 2008	Text independent, multi-lingual	12%	0.1%
Iris	ICE 2006	Controlled illumination large quality range	1.1-1.4%	0.1%



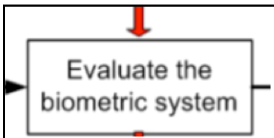
Data Collection

- Sensor size, cost, ruggedness, ability to capture good quality biometric samples
- **Include samples that are representative of the population and must preferably exhibit realistic intra-class variations (different sessions, times, conditions)**
- Legal and privacy issues must be considered (information commissioner in SLO)
- Biometric data collection is time-consuming, relatively expensive, and cumbersome process
- On the other hand, careful planning and implementation of data collection process is likely to lead to successful and operational biometric systems



Choice of Features and Matching Algorithm

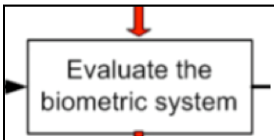
- Use the prior knowledge about the selected trait (if possible; compare fingerprints and faces)
- Check state-of-the-art results in scientific literature
- Problem of *interoperability* – algorithms that operate seamlessly **across different sensors** is paramount and will significantly impact the usability of the system over a period of time



Evaluation

- Fields: statistics, computer science, engineering, business, psychology, system design, UX
- Questions:

1. What are the error rates of the biometric system in a given application?
2. What is the reliability, availability, and maintainability of the system?
3. What are the vulnerabilities of the biometric system?
4. What is the user acceptability of the system?
5. What is the cost, throughput, benefits?



AD Q1: Matching Performance

- **Technology evaluation** compares competing algorithms from a single technology on a standardized database (repeatability)
- In **scenario evaluation**, the testing of the prototype biometric systems is carried out in an environment that closely resembles the real-world application
- **Operational evaluation** is used to ascertain the performance of a complete biometric system in a specific real-world application environment on a specific target population

Applications

FORENSICS	GOVERNMENT	COMMERCIAL
Corpse identification	National ID card	ATM
Criminal investigation	Driver's license; voter registration	Access control; computer login
Parenthood determination	Welfare disbursement	Mobile phone
Missing children	Border crossing	E-commerce; Internet; banking; smart card

Schipol Airport (Amsterdam)

Employs **iris** scan smart cards to speed up the immigration procedure

Passengers who are **voluntarily** enrolled in this scheme insert their **smart card** at the gate and peek into a camera

Ben Gurion International Airport (Tel Aviv)

Outbound passengers are required to provide **fingerprints** and **facial** images, which are stored in a smart card that is issued to each passenger

These smart cards are then used to **track** the passengers as they pass through different locations at the airport

US-VISIT Program

Visitor and Immigration Status Indicator Technology

Utilizes all ten fingerprints to validate the travel documents of visitors to the US

Ten fingerprints are matched against a dynamic watch list containing up to a few million records in less than 10 sec

Example of negative recognition, where the purpose is to find if the visitor has multiple aliases

If not on a watch-list, the visitor is admitted into the US and the person's fingerprints are enrolled into the database for future matching

Unique Identity (UID) Card Project in India

Facilitate efficient delivery of various welfare schemes

Ten fingers, two irises, and face

An example of de-duplication (negative recognition), where a fusion of 10 fingers and 2 irises is determining if the same person is trying to acquire two different 12-digit identification numbers

Disney World in Orlando

Prevent ticket fraud –
prevents more than one person using the same ticket

System works in all weather conditions because it uses a rugged fingerprint sensor that is able to capture good quality fingerprint images even in adverse imaging conditions

ATMs in Japan

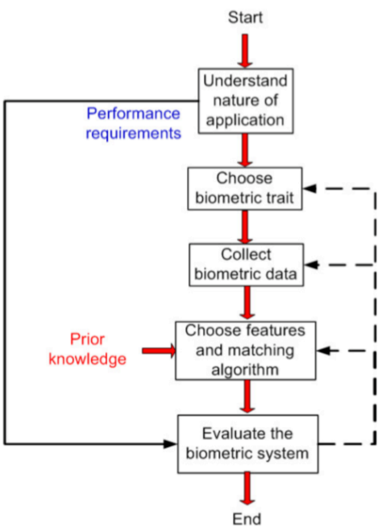
Palm-vein authentication systems

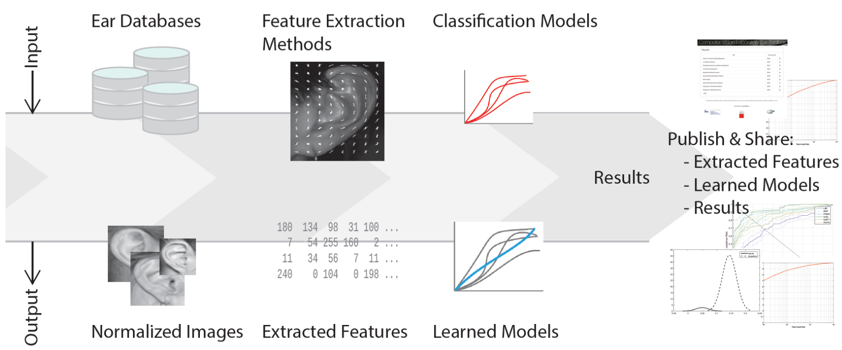
Using a near infrared lighting source

Information Security

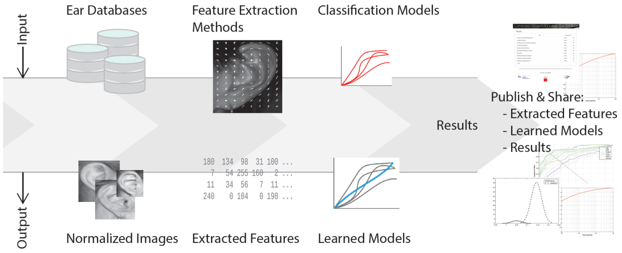
- "Protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction"
- Four major aspects to be considered in it:
  1. **integrity** – guard against improper **modification** or **destruction** of the data and ensure **non-repudiation** and authenticity of information
  2. **data confidentiality** – prevent **illegitimate access** or **disclosure** of sensitive information
  3. **availability** – guarantee timely and reliable access to and **use** of information
  4. **authentication** – only legitimate and authorized users should be able to access the data and carry out specific tasks

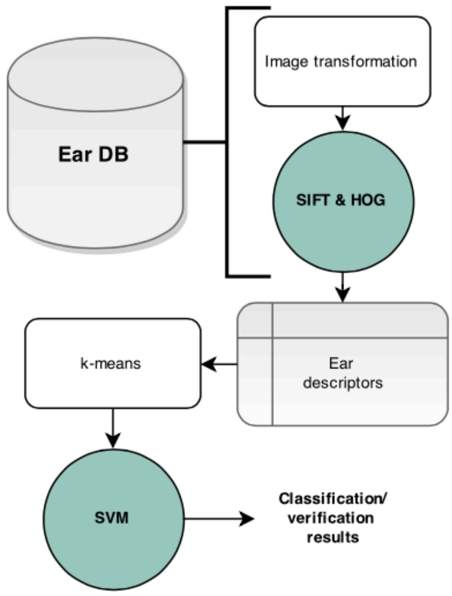
From General to Specific Scheme ... Based on Modality



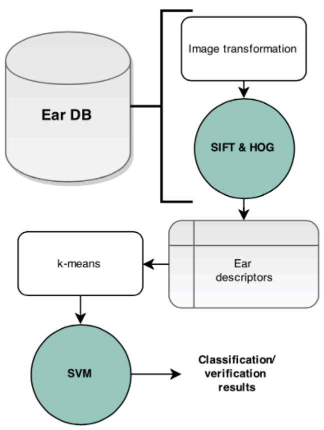


From General to Specific Scheme
... Based on Modality

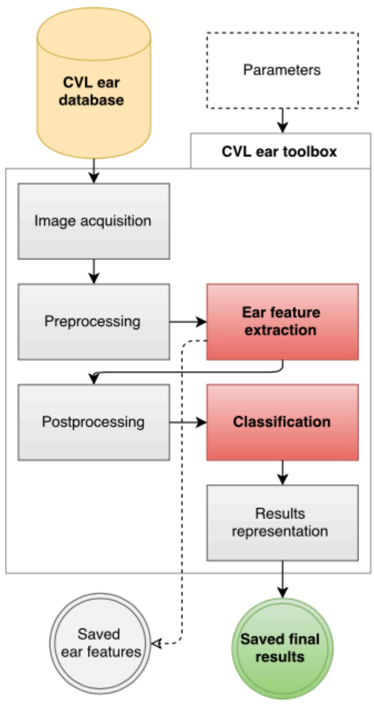




From General to Specific Scheme
... Based on Modality

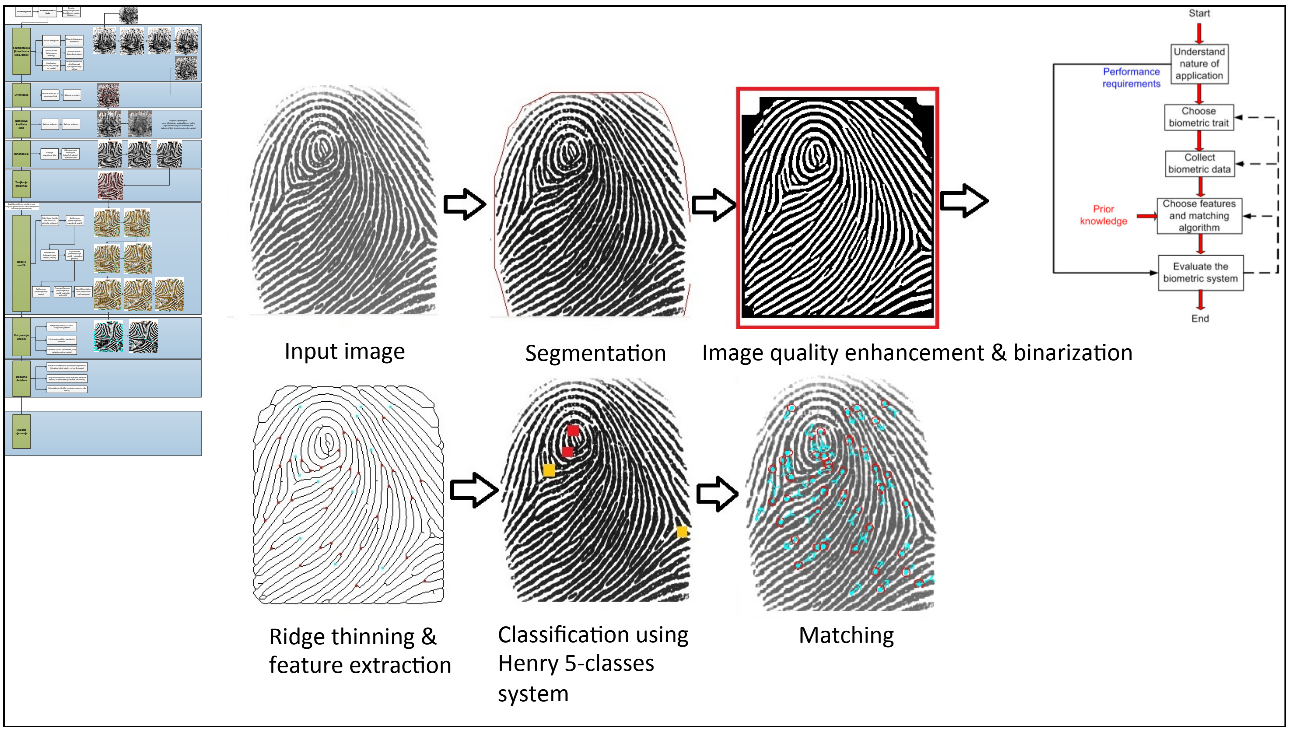


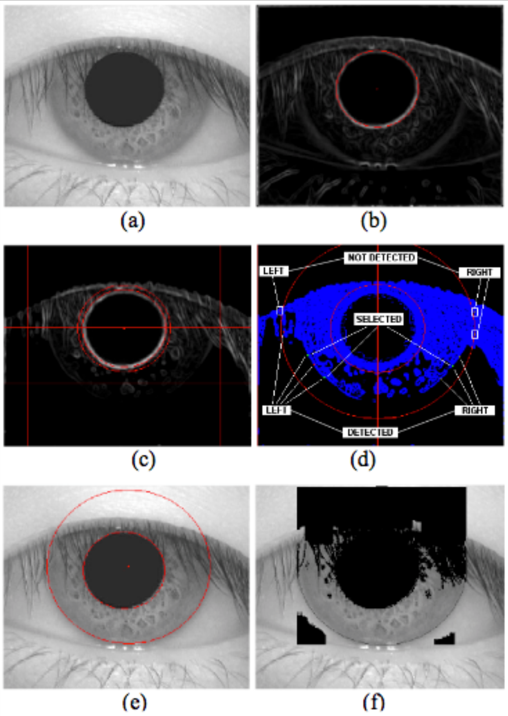

```
% READ DB
% the default path links to
% the included CVL ear database
[db, annotation_data] = database(path);

% PREPROCESSING
db = preprocess(db, annotation_data);

% FEATURE EXTRACTION
% features_extract(image, image_annotation_data)
% is called inside on each db image
features = features_extract_all(db, annotation_data);

% POSTPROCESSING
features = postprocess(features);

% EVALUATION
% learn(X, y, annotation_data)
% and predict(model, X, annotation_data)
% are called inside for each cross-validation step
results = evaluate(features, annotation_data);

% OUTPUT THE RESULTS
visualize_results(results)
```

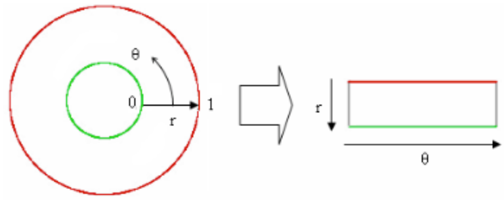


Importance of Protocols

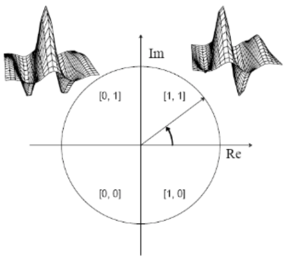
- Needed for **sound comparisons** of techniques / **repeatability**
- Define procedure, metrics, databases, frameworks/toolboxes
- Example:
  - AWE db & toolbox
  - partition the data into a *development set* that contains 60% percent of all images and a *test set* that contains the remaining 40%
  - 5-fold cross validation: means and standard deviations
  - bootstrapping on test set: means and standard deviations
  - files containing lists for the experiments are distributed with the dataset
  - ROC & ERR (VER), CMC & R-1 (ID)

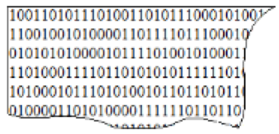


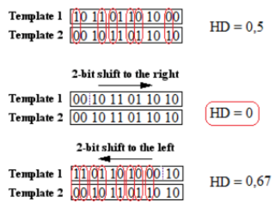




=> 2D Gabor wavelets give a set of complex values from which we calculate the phase angle in each pixel =>

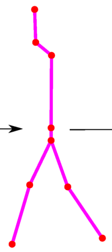




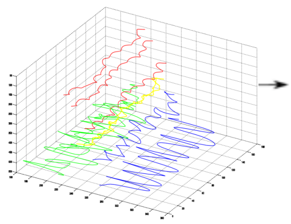


motion segmentation

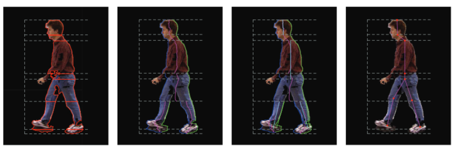
skeleton segmentation



feature extraction (64 signals)

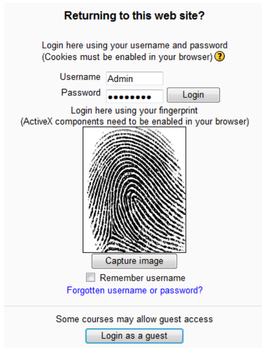


classification:
holistic (gait cycle),
feature fusion based,
frames-based

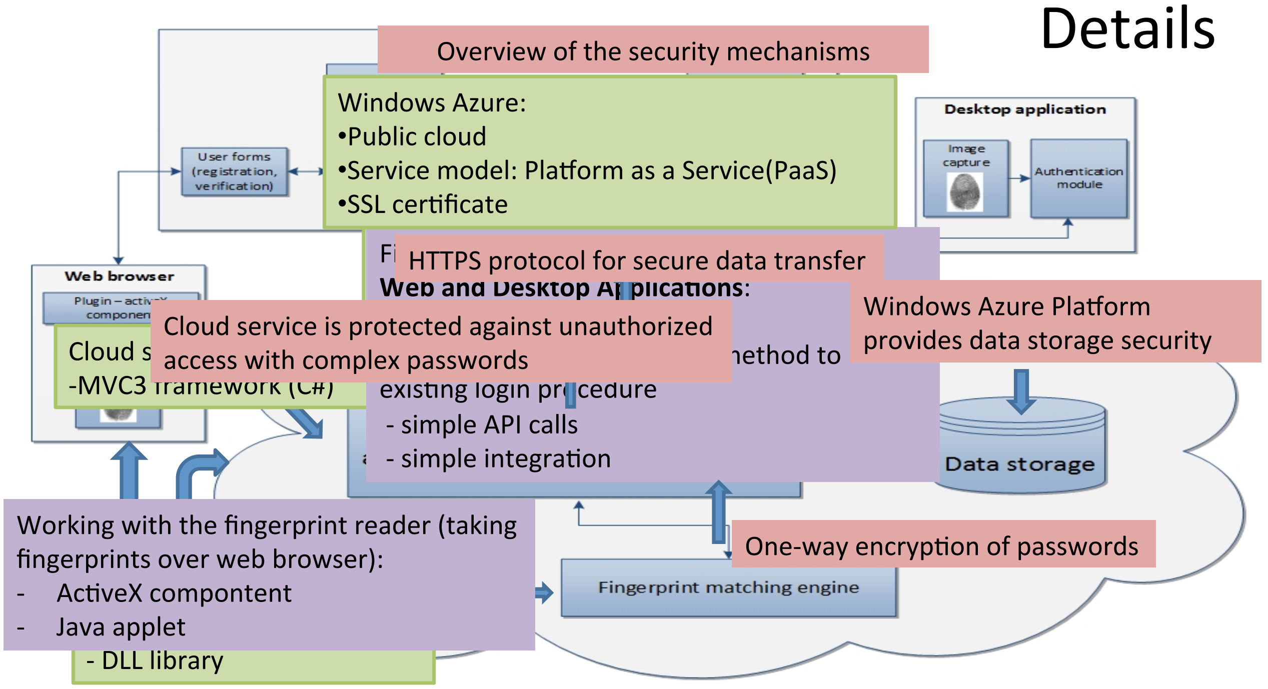


(a) Cloud implementation of face recognition service

(b) Cloud implementation of fingerprint recognition service



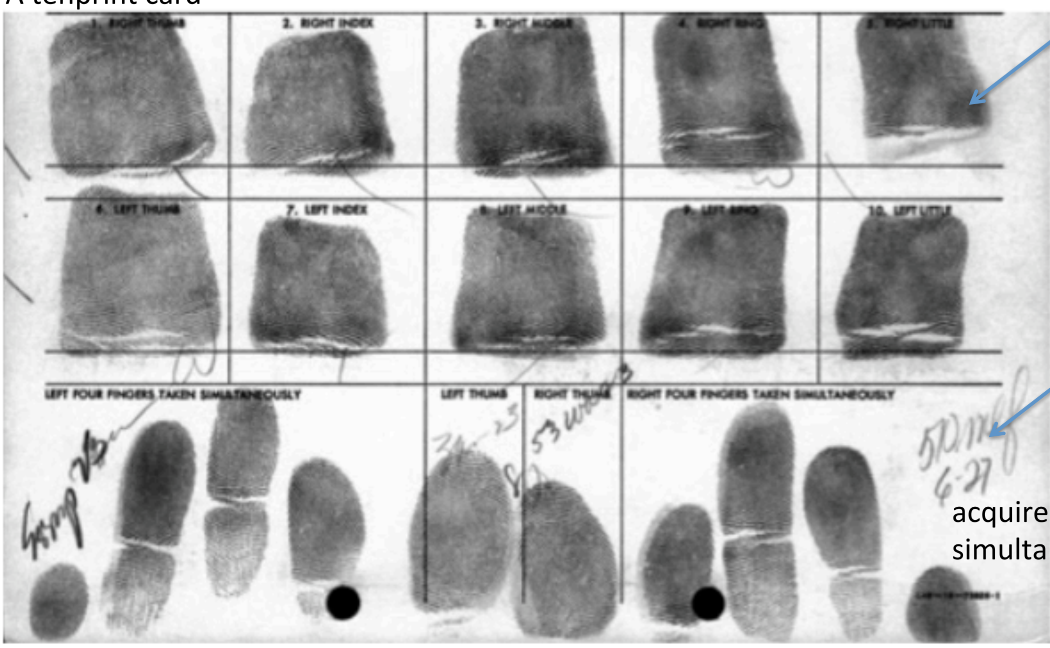
Details



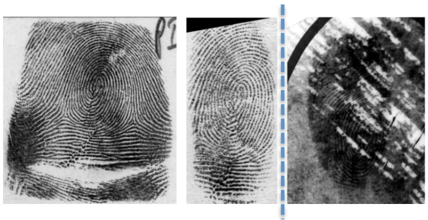
FINGERPRINT

- The ridges increase the friction with the contact surface – grasping made easier ;-)
- A typical young male has, on an average, 20.7 ridges per centimeter, while a female has 23.4 ridges per centimeter
- Is claimed to be unique
- Became synonym for biometric recognition in the minds of the general public

A tenprint card



A latent fingerprint



AFIS

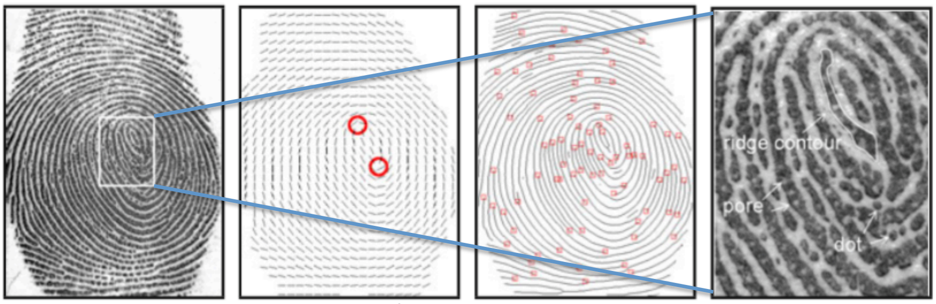
Michigan State Police

3.2 million tenprint cards

Performs 700,000 searches each year

Features

- Features used have a physical interpretation
- Three levels of features (L1-3)



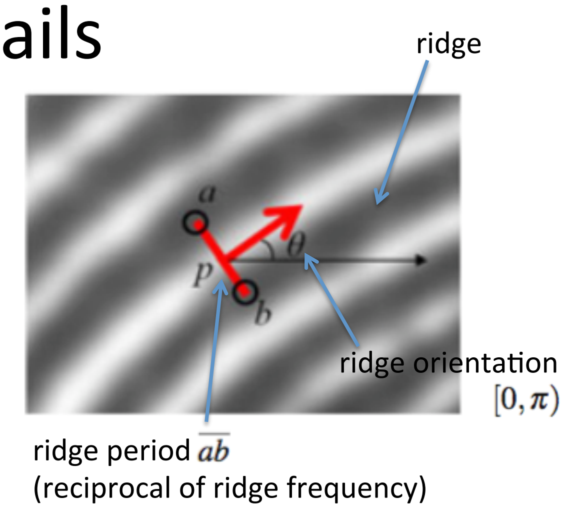
L1: orientation field or ridge flow and singular points
L2: ridge skeleton and minutiae
L3: ridge contour, pore, dot

L1 Details

- Oriented texture pattern – its global shape and structure can be defined by the **orientation** and **frequency** of its ridges
- Only the ridge flow and ridge frequency are observed; the exact location and dimensional details of ridges are **ignored**
- Low-resolution image sensors capable of scanning 250 ppi can be used

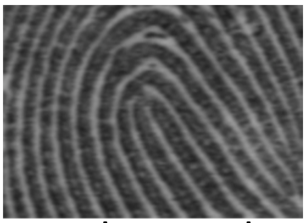
L1 Details

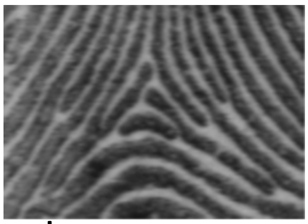
- Coarse features @ pixel *p*
- Ridge orientation information is viewed as being more important



Singular Points

- Salient locations where the ridge orientations change abruptly
- Two basic types: **loop** and **delta**
- Loop can be used as a landmark point to **align** the fingerprint



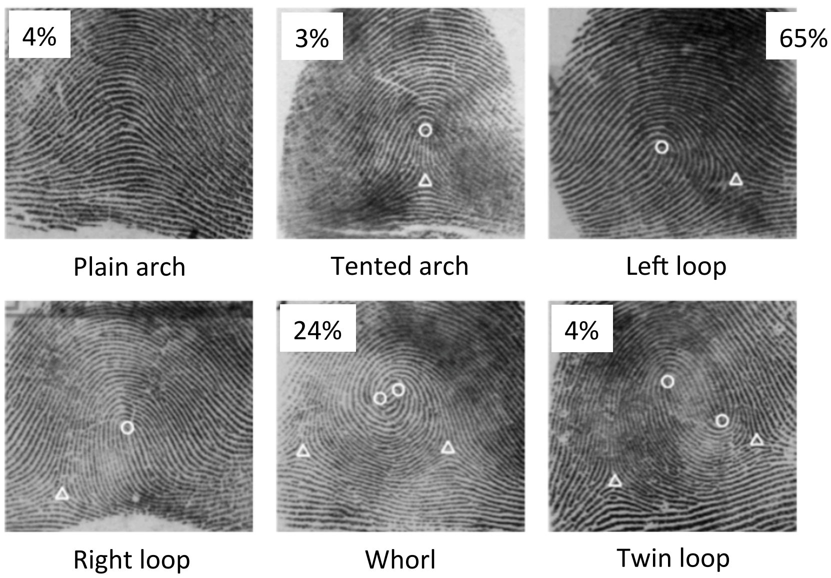


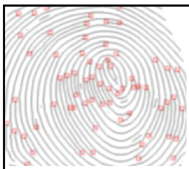
- **Core** point corresponds to the north most loop-type singular point
- No singular points => core usually refers to the point of maximum ridge curvature

Singular Points

- Singular points can be viewed as an abstract representation of the orientation map =>
- Orientation map can be roughly predicted based on the number and location of singular points
- Even more abstract representation is the pattern type (fingerprint class) =>
- Deduced based on the number of loops and deltas and the spatial relationship between them

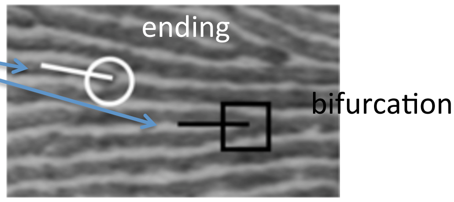
Pattern Types





L2 Details

- Represented as a ridge skeleton image
- **Locations** (a) where a ridge emerges, ends, splits, or merges with another ridge are termed as ridge characteristics or **minutiae**
- Minutia generally has two other properties: **direction** (b) and **type** (c)
- Easily observed at resolution of 500 ppi

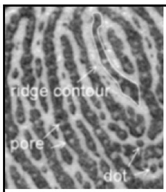


Minutiae & AFIS

- Number varies a lot according to the acquisition method etc.
- Abstract representation of the ridge skeleton
- Capture much of the discriminative information
- Are storage efficient
- Reasonably robust to various sources of degradation

Rolled – 136 Plain – 56 Latent – 18

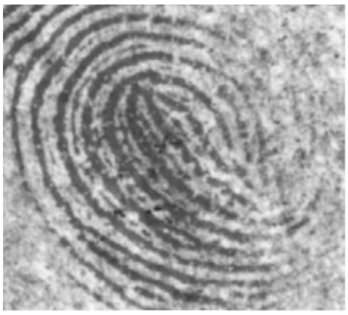
L3 Details



- Ridges are no longer viewed as being simple, one-pixel wide skeletal images
- Using the inner holes (**sweat pores**) and outer **contours** (edges)
- **Incipient** ridges are immature ridges, thinner than **mature** ridges and contain no sweat pores
- **Dot** is a very short ridge containing only a single ridge unit

L3 Details

- 1000 ppi scanning capability is needed to capture L3 details
- importance in matching **latent** fingerprints – contain much fewer minutiae than rolled or plain ones

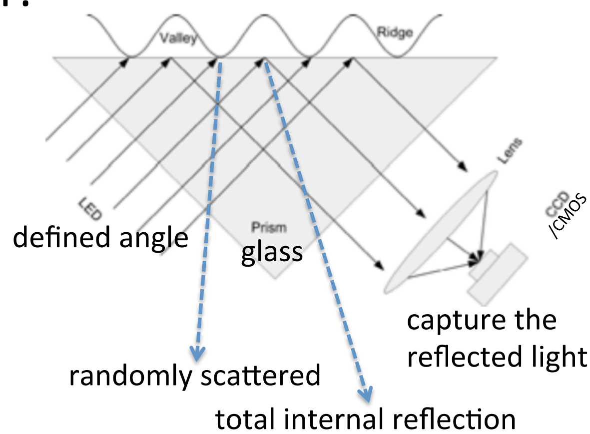


latent

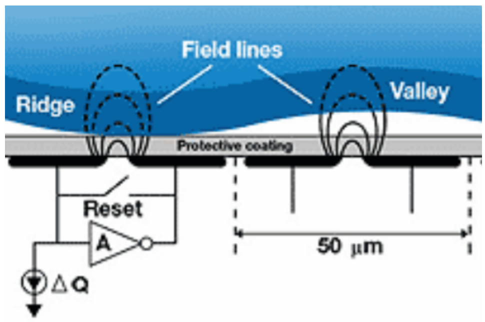
rolled

Live-Scan Sensors

- Optical FTIR-based sensor:
  - Difficult to have this arrangement in a compact form; focal length of small lenses can be very large
  - Image distortions are possible when the reflected light is not focused properly



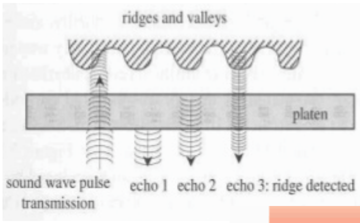
Live-Scan Sensors

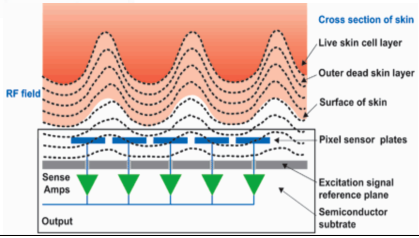


- Capacitance-based sensor:
  - More **commonly used**
  - Very small in size
  - An array of electrodes
  - Fingerprint skin acts as the other electrode, forming a miniature capacitor
  - Magnitude of electrical charges depends on the distance between the fingerprint surface and the capacitance plates
  - Capacitance due to the ridges is higher than those formed by valleys
  - Proper grounding needed due to tip electrostatic discharges

Live-Scan Sensors

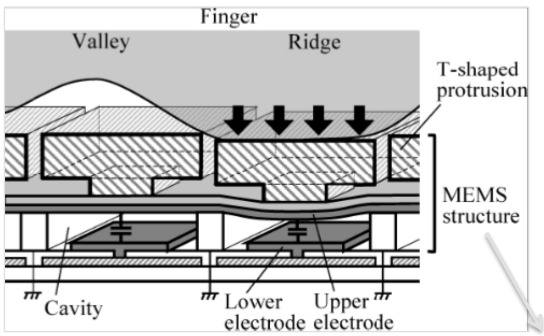
- Ultrasound reflection-based sensor:
  - Sends acoustic signals toward the tip and capture the echo signal
  - Compute the **range image**
  - Images the **sub**surface of the fingerprint =>
  - Resilient to dirt and oil accumulations
  - Expensive





Live-Scan Sensors

- Piezoelectric effect-based sensor:
  - **Pressure**-sensitive sensor
  - Produce an electrical signal when a **mechanical** stress is applied
  - Generates a small amount of **current** => piezoelectric effect
  - Does not capture relief accurately because of its low sensitivity



mechanical devices such as sensors, embedded in semiconductor chips

Live-Scan Sensors

- Temperature differential-based sensor:
  - Made of pyro-electric material that generates a current based on temperature differentials (lasts approx. 100 ms)
  - Sensors are typically maintained at a high temperature by electrically heating them up
  - Only four bit gray scale from sensor (Atmel, 2005) ?!

Image Quality

- Quality factors: image **resolution**, finger **area**, **clarity** of ridge pattern

- 500 ppi (L2) => distance between adjacent ridges is approximately 9 pixels
- Civilian applications => sensors with <500 ppi are often used to reduce the cost

Image Quality – Area

- The finger has to be rolled on the sensor surface to obtain the full fingerprint
- In consumer electronic products => swipe sensors to further reduce the cost

Swipe Sensors

Software stitching process may produce **artifacts** => unacceptable in forensic quality fingerprints

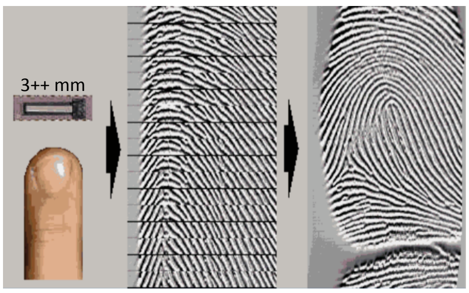


Image Quality – Clarity

- If finger is dry, the ridges may have many breaks
- If moist, adjacent ridges may be joined

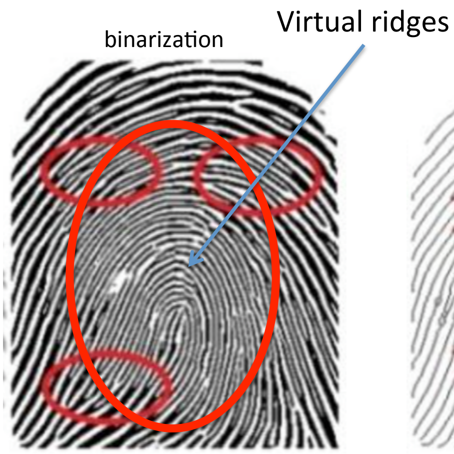
dry finger

wet finger

many creases

How to Evaluate the Quality?

Is this one good or not?



Where do we have a problem?

How to Evaluate the Quality?

- Two main approaches:
  - Use of local properties
    - Block division
    - Groups of quality
    - Calculate the ratio between good and bad blocks
  - Use of global properties
    - Global metric

+ Combination

How to Evaluate the Quality?

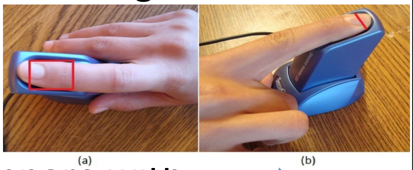
- Local methods:
  - Local ridge orientation
  - Gabor filters
  - Pixels intensity
  - Power spectrum
  - Combination

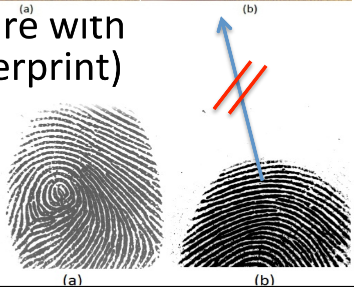
How to Evaluate the Quality?

- Global methods:
  - Field direction continuity
  - Uniformity of the frequency field
  - Features counting
  - Combination

How to Evaluate the Quality?

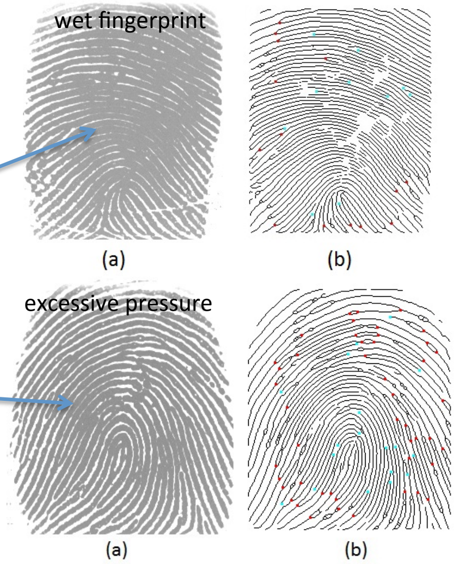
- Pixels intensity method:
  - Use histogram to determine the value of background *b* [0..255] based on decision threshold (peak in the lighter part > 1k)
  - Correct each pixel value (-(255-*b*))
  - Divide image to blocks (9*9)
  - Calculate mean, variance, gradient and compare with decision thresholds (classes: background, fingerprint)
  - If block still unclassified: within the block calculate value based on gradients and subject it to a threshold
  - If *#bgr_blocks* > 50% => reject





How to Evaluate the Quality?

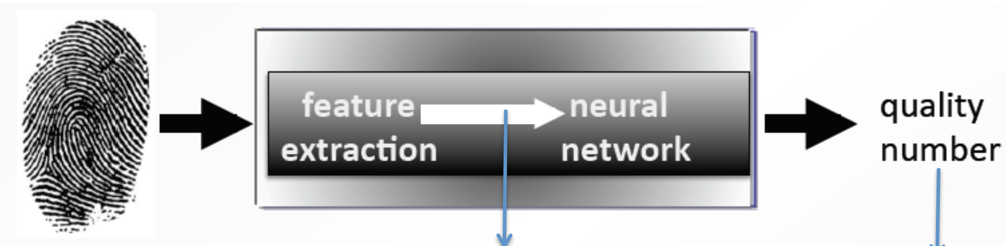
- Features counting method:
  - Bad quality images normally give very small or big number of feature points => reject
  - Secugen Hamster Plus sensor (based on 320 test samples) => ok if 25 ≤ *#minutiae* ≤ 70



How to Evaluate the Quality?

- NFIQ method:
  - Prediction of matching efficiency
  - Combines local and global properties
  - For each **feature** calculate ridge density and clarity, high curvature areas, dominant block orientation, intensity etc.
  - Evaluate each **block** based on feature quality (4 – perfect quality … 0 – background)

How to Evaluate the Quality?



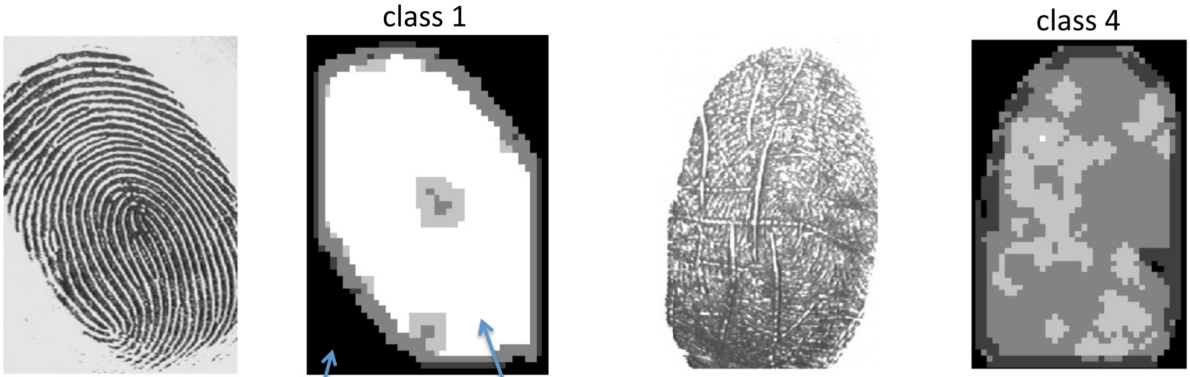
11D vector:
#fingerprint_blocks,
#features,
#features_with_quality_x
(x ≥ 0.5 / 0.6 / 0.7 / 0.8 / 0.9),
%blocks_with_quality_y
(y = 1 / 2 / 3 / 4)

class 1 – perfect quality
…
class 5 – bad quality

How to Evaluate the Quality?

class 1

class 4



(a) (b) (a) (b)

background –
block quality 0

block quality 4

=> Registration with only class 1 and 2 quality is acceptable

Comparison Results

groundtruth: 31% 34% 55% 21%

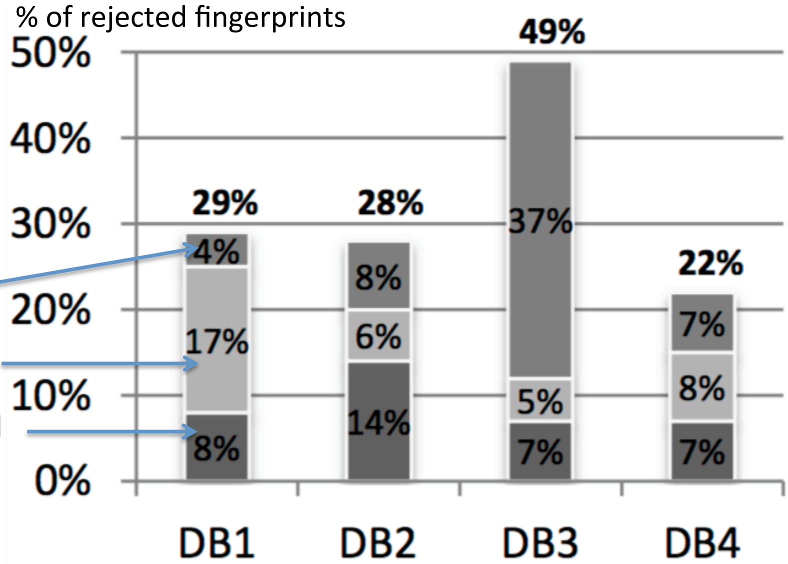
% of rejected fingerprints

FVC 2002 DB1-4
800*4 fingerprints
80 used for thresholding

1) NFIQ method

3) pixels intensity method

2) features counting method



different sensors

synthetic fingerprints

Need more info?
Bule, Tovšak, Peer, Fingerprint Image Quality Assessment, ERK 2011.
Challenge (for a project)?
Foggy images, distortion in only some parts of the image, other methods.

How to **Improve** the Quality?

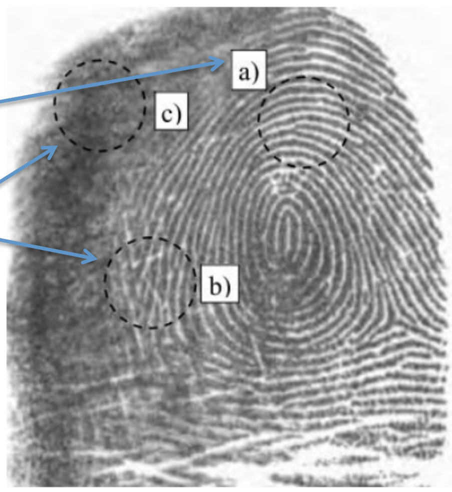
Three categories of regions:

a) Well defined regions

b) Renewable regions

c) Nonrenewable regions

Goal of the methods:
1. Enhance ridge structure
2. Mark nonrenewable regions



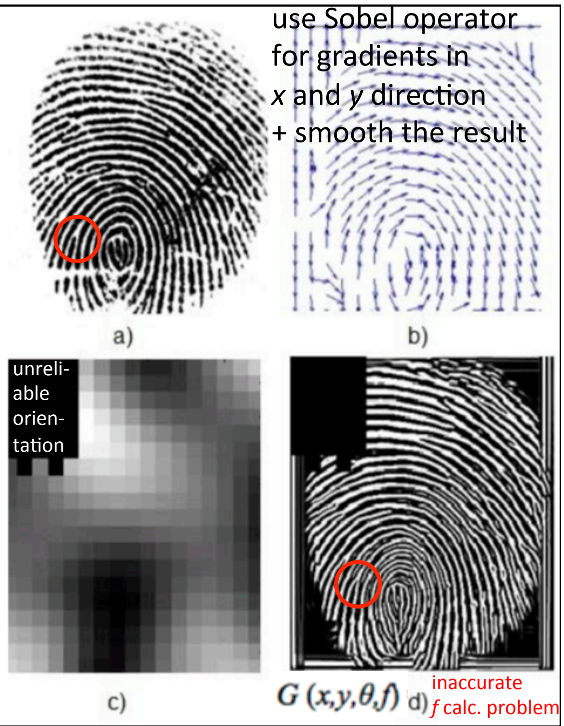
How to Improve the Quality?

- Methods:
  - Global filters are normally not practical
  - Use of contextual information in regions:
    - Local orientation
    - Local frequency
    - Ridge continuity

  Ridge regularity

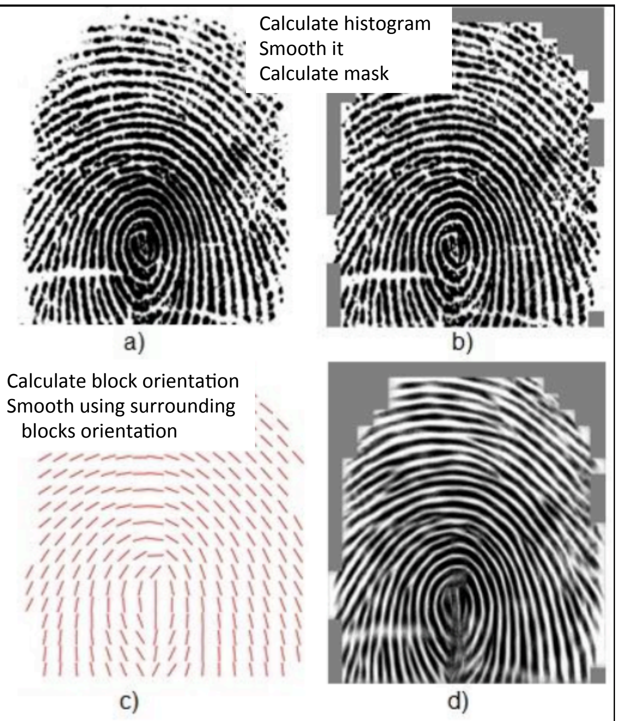
How to Improve the Quality?

- Hong's method:
  - Calculate local ridge **orientation**
  - Calculate **frequency** using sinusoidal curve => visualize with intensity
  - Use 2D Gabor filter with orient. and freq. info



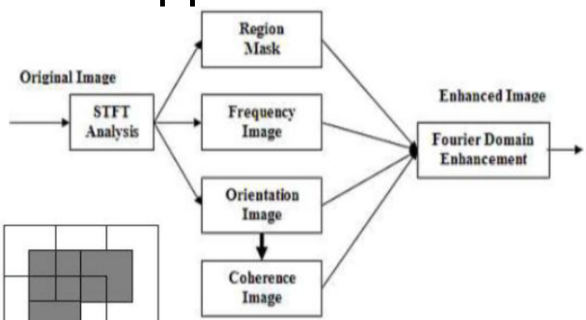
How to Improve the Quality?

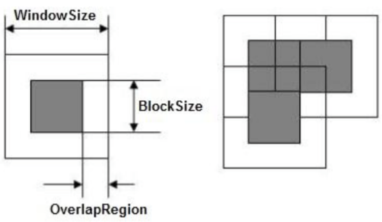
- 1D smoothing method:
  - Input: image (a), blocks (b), segmentation mask (b), orientation field (c)
  - Divide circle with 32 lines, each having 11 points
  - Determine which line best fits the orientation field
  - Use 11 points to smooth the ridges in the block



How to Improve the Quality?

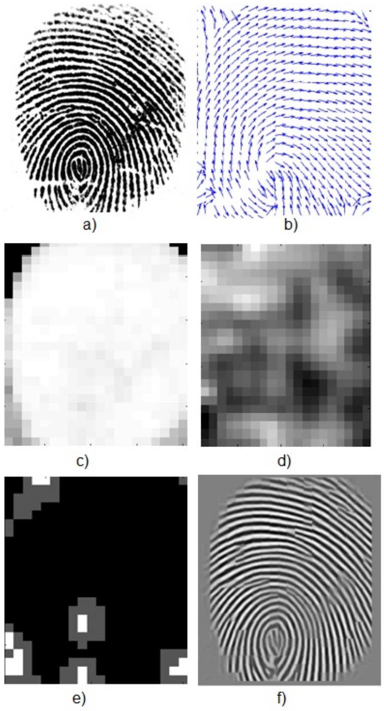
- Short Time Fourier Transform approach:
  - Independence between calculations =>
  - Overlapping windows
  - Block treated as surface wave





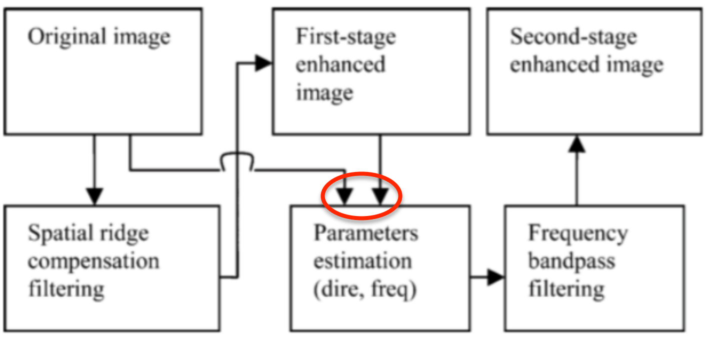
How to Improve the Quality?

- STFT approach:
  - Use block Fourier spectrum to calculate orientation (b) and frequency (d)
  - Use energy image (c) from STFT for masking
  - Calculate angle coherence image (e) using (b)
  - Filter each window in Fourier domain (using obtained info) and combine windows for final result (f)



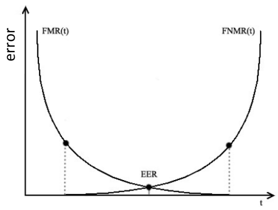
How to Improve the Quality?

- Two-Stage Enhancement Scheme:
  1. S1: Enhancement in spatial domain
     a) Use small window along orientation field with weighted mask to improve ridge quality
     b) Improve ridge contrast along local orientation
  2. S2: Enhancement in frequency domain
     a) Estimate orientation and frequency (similar as Hong)
     b) For each block: FT => filter with obtained parameters => IFT



Comparison Results

FVC 2002 DB3
800 fingerprints



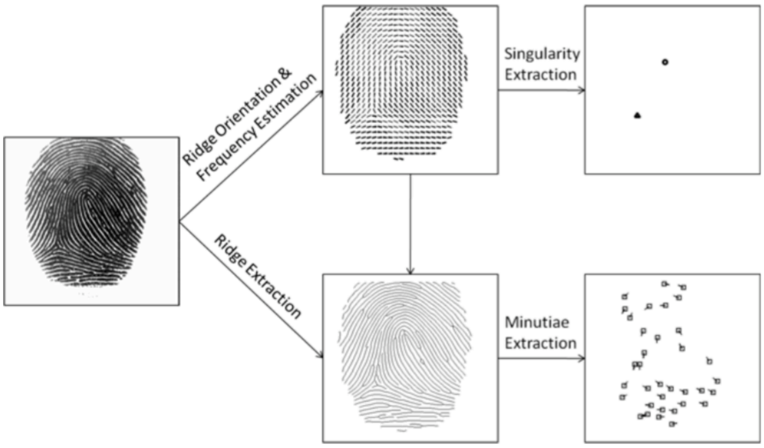
Method	EER	Processing time
non	37.5 %	/
Hong's method	29.5 %	0.444 s
1D smoothing	7.8 %	**0.07 s**
STFT	7.8 %	0.430 s
2SES	7.5 %	10.538 s
STFT + 1D smoothing	*6.1 %*	*0.512 s*
2SES + 1D smoothing	**5.9 %**	10.732 s

@i7 / 2011

Need more info?
Bule, Peer, Fingerprint Image Enhancement, ERK 2012.
Challenge (for a project)?
Other methods.

Feature Extraction

Commercial systems => L1&2 features



Feature Extraction

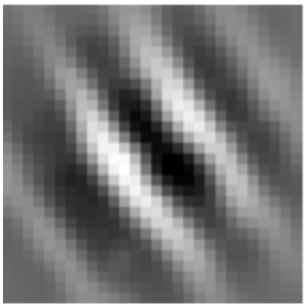
Ridge orientation and frequency estimation

- Approximate ridge pattern in a local area by a cosine wave (defined by amplitude, frequency & orientation – parameters)
- Compute its 2D Fourier transform (using FFT)
- The wave parameters can be obtained by detecting the maximum value of the magnitude spectrum
- But …

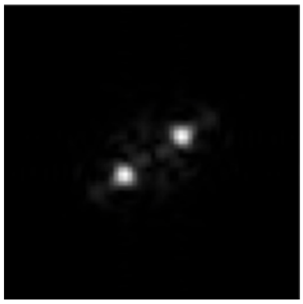
Feature Extraction

Ridge orientation and frequency estimation

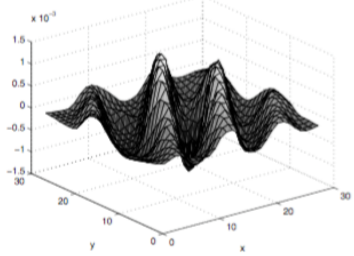
- But … ridge pattern is not exactly a cosine wave
- FT contains a pair of blurred impulses
- Magnitude spectrum is first **smoothed** using a low pass filter and then the maximum value is detected => better parameters obtained



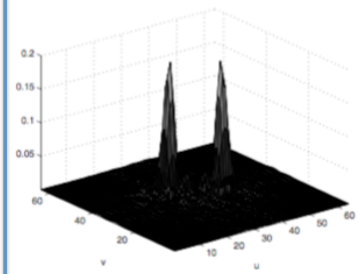
Local ridge



Magnitude spectrum



3D visualization

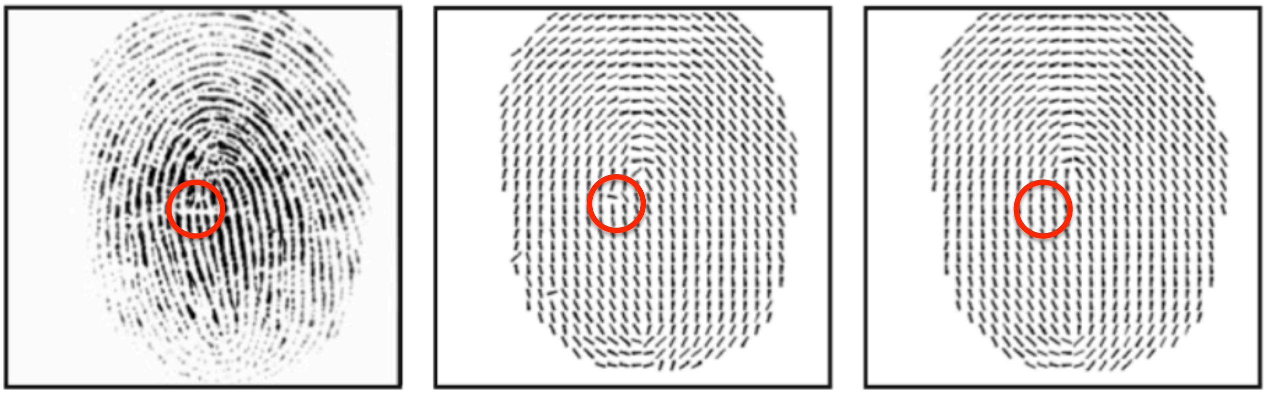


3D visualization

Feature Extraction

Ridge orientation and frequency estimation

- Problem of **creases**? => smoothing
- Recall … ridge orientation of fingerprints in a local area is defined in the range $[0,\pi)$ => $\theta = (\theta + \pi)$ => average between 1° and 179° should be 0° rather than 90°
- Smooth using a vector field $V = (V_x,V_y) = (\cos 2\theta, \sin 2\theta)$



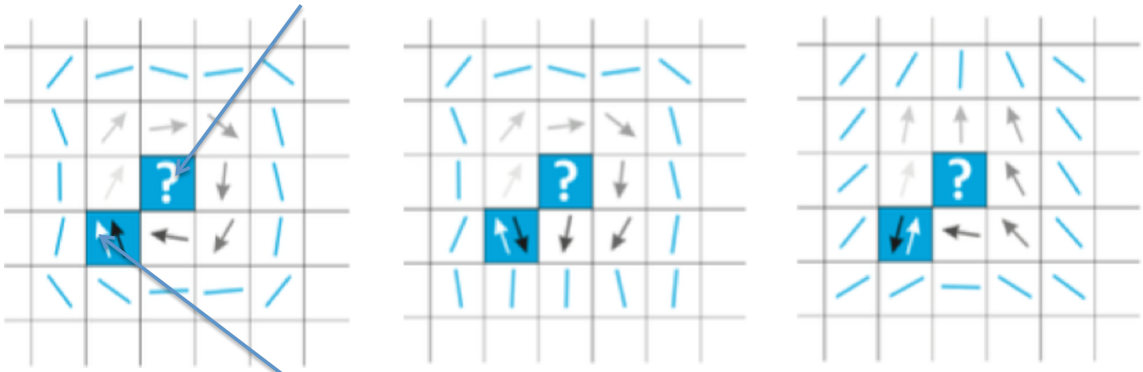
input | initial (noisy) orientation field | smoothed orientation field

Feature Extraction

Singularity extraction

- Poincare´ index method
- = cumulative change of orientations along a closed path in an orientation field
- Generally evaluated using the eight neighbors of a pixel

calculating the Poincaré Index



p = 360◦ (whorl)
(white and black arrow are in the same direction)

p = 180◦ (loop)

p = -180◦ (delta)

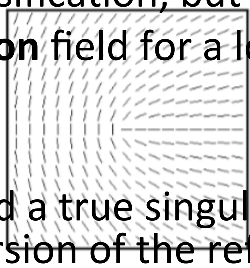
1. Select a random orientation for **one** example element (<u>white</u>) of the directional image
2. For the remaining elements select the **closest** orientation to the neighboring element
3. Travel along the closed curve and sum up the rotations

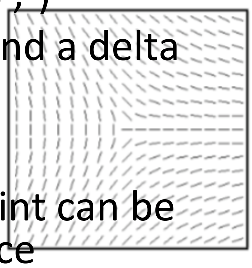
if 0◦ => no singular point

Feature Extraction

Direction of singularity

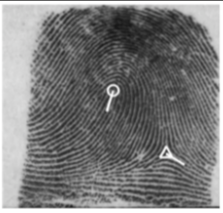
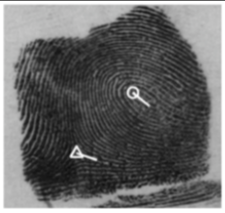
- Not absolutely needed for classification, but helps ;-)
- Define the **reference orientation** field for a loop and a delta





- The orientation field (α) around a true singular point can be **approximated** by a rotated version of the reference orientation field
- Calculate α based on the difference between the local orientation field around a singularity and the reference orientation field

Remember classes?


- Left loop, right loop, twin loop;
  plain arch, tented arch, whorl

- It requires the fingerprint to be **complete** (i.e. a good quality rolled fingerprint)
- Latent fingerprint obtained from crime scenes is typically considered as **unknown** and latents are often searched against all types of fingerprints in the database

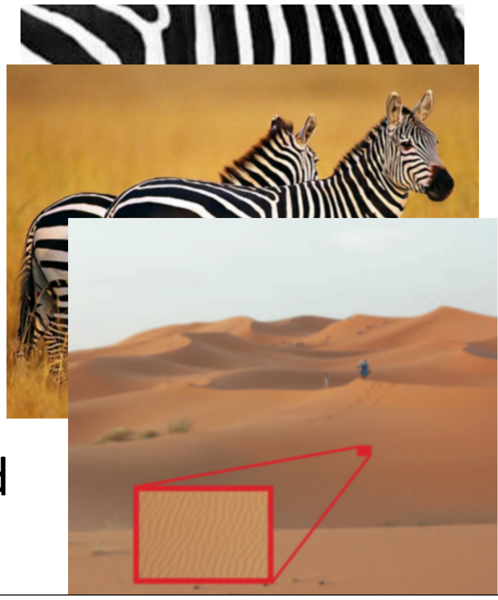
Feature Extraction

Ridge extraction (to get minutiae ;-)

- Enhanced image is converted to a **binary** image by using either a global threshold (Otsu thresholding) or thresholds that are locally computed
- A morphological operation **thinning** is used to reduce the ridge width to one pixel


Feature Extraction

What is this?

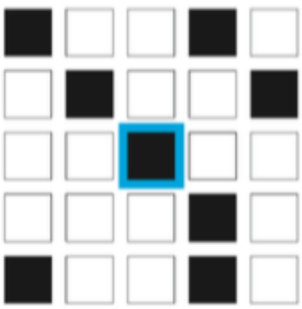


Minutiae extraction

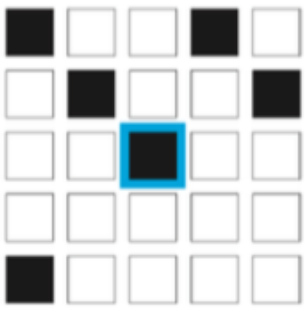
- Ridge pixels with 3 ridge pixel neighbors are identified as ridge **bifurcations**
- Those with only one ridge pixel neighbor are identified as ridge **endings**

Feature Extraction

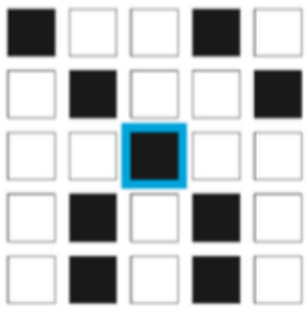
Minutiae extraction



#ridge_neighbors=2 => ridge (no minutiae)



=1 => ridge ending

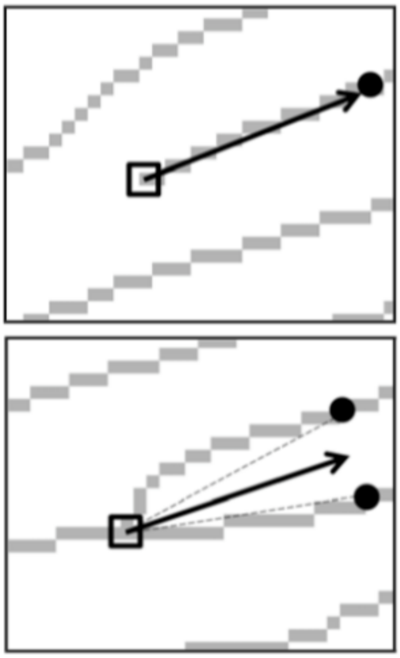


=3 => bifurcation

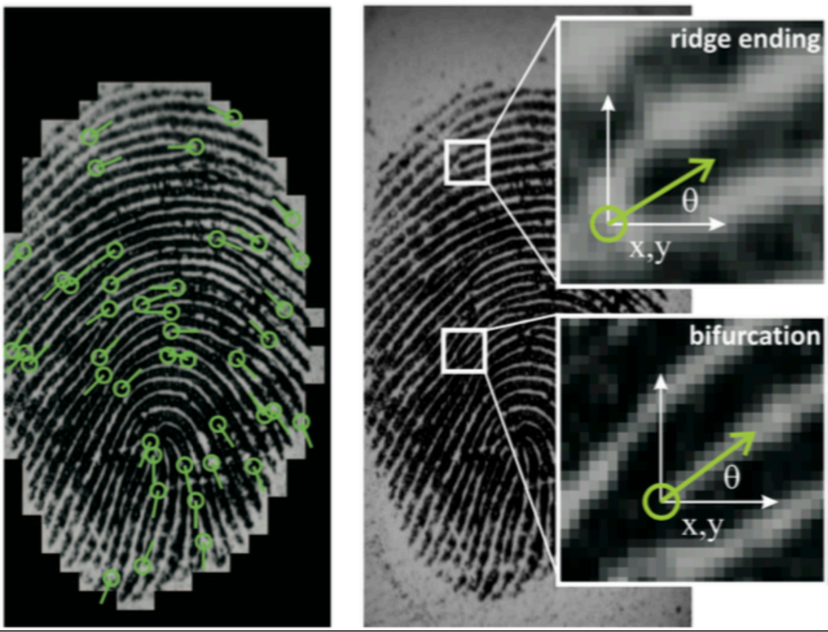
Feature Extraction

Minutiae **direction**

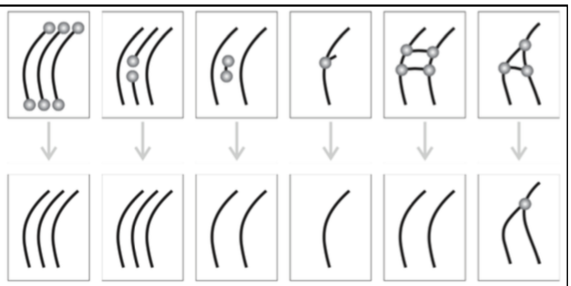
- Ending => **trace** the associated ridge to a fixed distance (say 10 pixels)
- Bifurcation => obtain three points by tracing the ridges to a fixed distance & calculate mean of the two directions whose difference is the **smallest** among the three ridges



Importance of standards:
Minutiae directions compliant to ISO/IEC FDIS 19794-2 (2011)



Feature Extraction

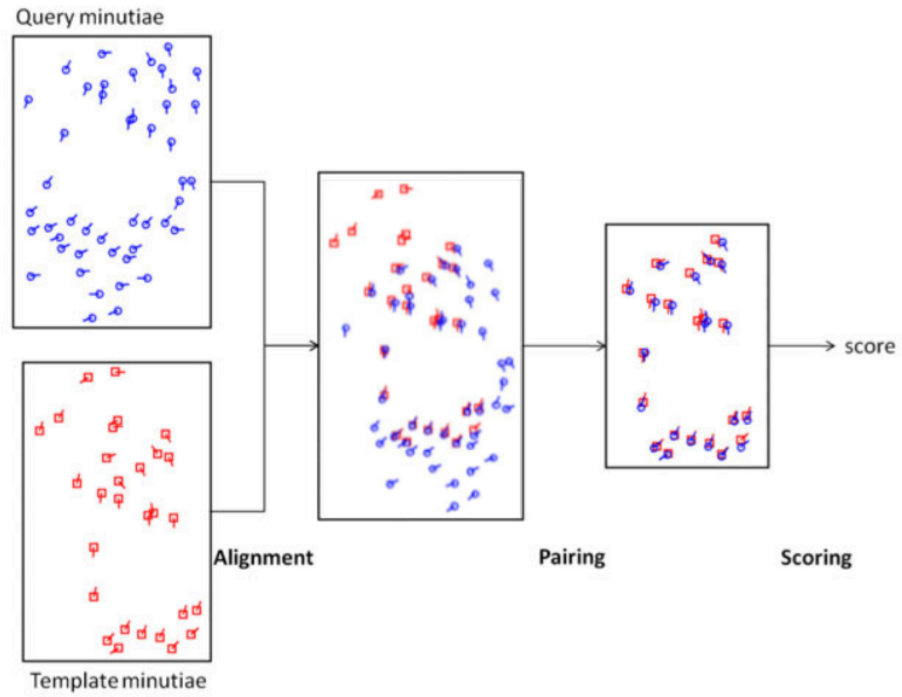


Minutiae **filtering**

- Removal of spurious minutiae
- Heuristic methods:
  remove minutiae located at boundaries,
  that are close in location and almost opposite in direction,
  too many minutiae in a small neighborhood
- Duality of minutiae: simultaneous processing of positive and negative images

Matching

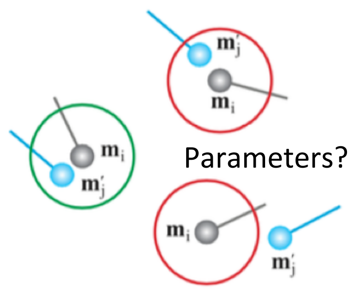
1. Alignment: determine the geometric transformation between the template & query minutiae sets so that they are in **the same coordinate system**
2. Correspondence: form **pairs** of corresponding minutiae
3. Score generation: compute the **match score** based on the corresponding minutiae points



Matching

Alignment/registration

- Transforms one image in such a way that it is geometrically aligned with the other
- Rigid transformation is normally sufficient: **linear mapping** + compensation for non-linear deformations (e.g. use of tolerance box)
- Use **generalized Hough transform** for estimating the spatial transformation between two point sets =>



input : Two minutiae sets $\{x_i^T, y_i^T, \theta_i^T\}_{i=1}^{M}$ and $\{x_j^Q, y_j^Q, \theta_j^Q\}_{j=1}^{N}$

output: Transformation parameters

Initialize accumulator array A to 0

for $i = 1, 2, \cdots, M$ **do**

  for $j = 1, 2, \cdots, N$ **do**

    $\Delta\theta = \theta_i^T - \theta_j^Q$

    $\Delta x = x_i^T - x_j^Q cos(\Delta\theta) - y_j^Q sin(\Delta\theta)$

    $\Delta y = y_i^T + x_j^Q sin(\Delta\theta) - y_j^Q cos(\Delta\theta)$

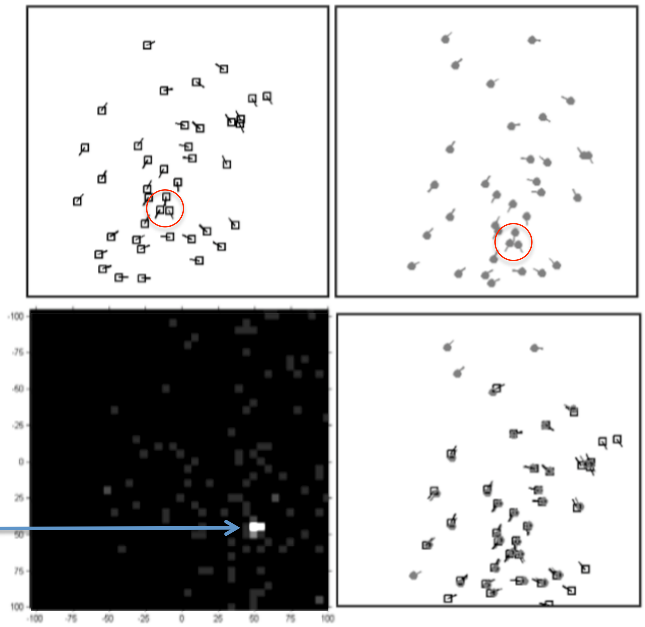
    $A[\Delta\theta][\Delta x][\Delta y] = A[\Delta\theta][\Delta x][\Delta y] + 1$

  end

end

return location of peak in A

Simplified example for visualization purposes when only translation is in question (2D case) =>



receives the most votes

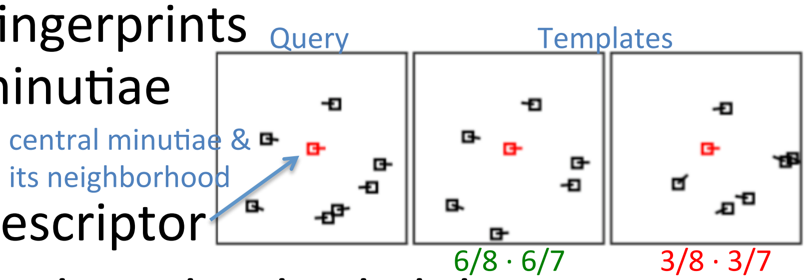
Matching

Useful modifications for HT

- Add fourth dimension to accumulator array for **scale**
- Simultaneous weighted incrementation of location in array and its neighbors
- Parallelization of HT is easy
- Coarse-to-fine approaches

Matching

Alignment … if the overlapping area between two fingerprints contains very **few** minutiae (latents?!)



1. Define minutia descriptor
2. Pairing of neighboring minutiae in it is established (similar as on the next slide)
3. Compute the product of the percentages of matched minutiae

Matching

Pairing minutiae

- Minutia *a* is in correspondence with *b* if their distance is within a predefined **distance** threshold (e.g. 15 pixels)
- And the angle between their directions is within another predefined **angle** threshold (e.g. 20 degrees)
- *a* is allowed to match to **at most one** *b* and vice versa (use of flag array) =>

input : Two minutiae sets $\{x_i^T, y_i^T, \theta_i^T\}_{i=1}^M$ and $\{x_j^Q, y_j^Q, \theta_j^Q\}_{j=1}^N$;
Transformation parameters $(\Delta\theta, \Delta x, \Delta y)$
output: List of paired minutiae
Initialize: set flag arrays f^T, f^Q, and *count* as 0; *list* as empty
for $i = 1, 2, \cdots, M$ **do**
    for $j = 1, 2, \cdots, N$ **do**
        if $f^T[i] == 0$ & $f^Q[j] == 0$ & *distance between minutiae i and j* $< t_d$ & *rotation between them* $< t_r$ **then**
            $f^T[i] = 1$
            $f^Q[j] = 1$
            $count = count + 1$
            $list[count] = \{i, j\}$
        end
    end
end
return *list*

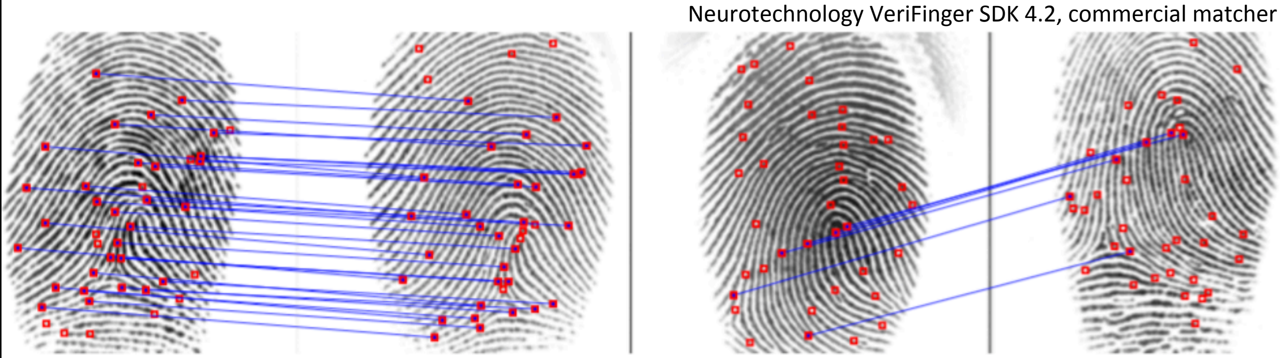
Matching

Match score

- Number of paired minutiae
- Percentage of matched minutiae in the overlap area
- Number of paired minutiae /
  average number of minutiae in both maps

- Compare to a predefined **threshold** to classify the two fingerprints as a **genuine** or an **impostor** match (two-class classification)

Neurotechnology VeriFinger SDK 4.2, commercial matcher



31 matched minutiae

6 matched minutiae

match score is computed as some function of the number of matched minutiae and *some other parameters that are proprietary* to the commercial matcher

match score = 614

match score = 7

More reading material?

Bule, Peer, Fingerprint matching, ROSUS 2012.

Pohar, Bule, Peer, Ridge-based fingerprint matching, ROSUS 2013.

Performance evaluation

Evaluation	Data	Best reported accuracies
NIST FpVTE 2003 (MST)	10,000 plain fingerprints	FNMR = 0.6% at FMR = 0.1%
FVC2006	140 fingers, 12 images per finger Electric field sensor (250 ppi) Optical sensor (569 ppi) Sweep sensor (500 ppi)	 FNMR = 15% at FMR = 0.1% FNMR = 0.02% at FMR = 0.1% FNMR = 3% at FMR = 0.1%
NIST ELFT 2008 (Phase II)	835 latent prints, 100,000 rolled fingerprints	FNMR = 8% at FMR = 1%

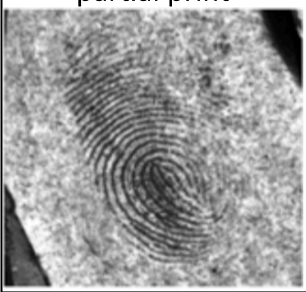
Note: variability due to data characteristics (also number of subjects, demographic distribution; *direct comparison possible only on the same DB!!!*), sensors, environments (including indoor/outdoor, temperature, humidity), user interactions with the sensor, specific application (i.e. Disney World is using low FNMR)

Open Research Questions

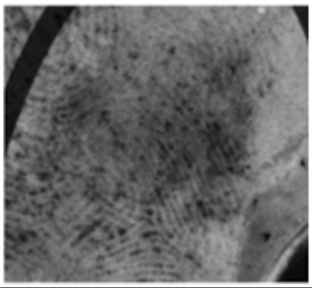
Latent matching

- Poor image quality, small finger area, large nonlinear distortion of most latents

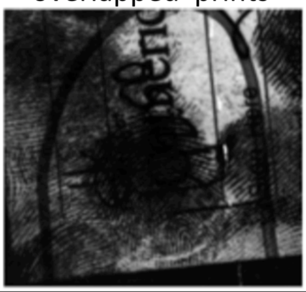
partial print



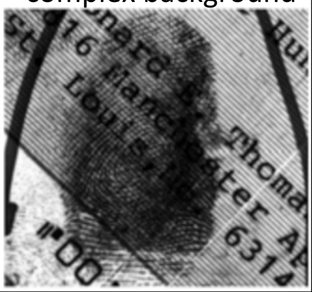
unclear structure



overlapped prints



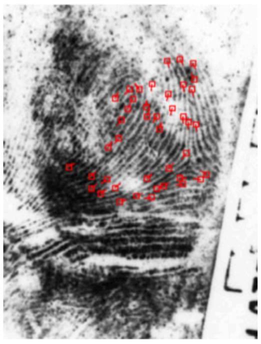
overlapped with complex background

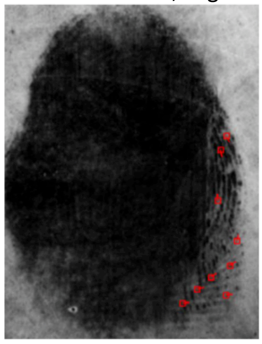


Open Research Questions

NIST SD27 latents DB:
5..82 minutiae, avg=**21**

Latent matching





- State-of-the-art feature extractors do not work well => manual extraction
- Limited number of minutiae => L3 features
- Main challenge is how to reliably encode and match L3 features in poor quality latents

Open Research Questions

Latent matching

- Daubert standard for source of forensic evidence (admissibility of scientific testimony):
  1. underlying scientific basis should be accepted widely
  2. its **error rate** should be known **=>**
- Problem: estimate involves human factors (latent development, minutia extraction,…) for which the error rate is **not** known!
- Solution: keep improving the performance of automated fingerprint systems and ultimately replace human experts with automated systems ;-)

Open Research Questions

Anti-spoofing

Acquisition of 3D fingerprints in a touchless mode

Persistence and uniqueness over time

Age 34 Age 40 Age 42

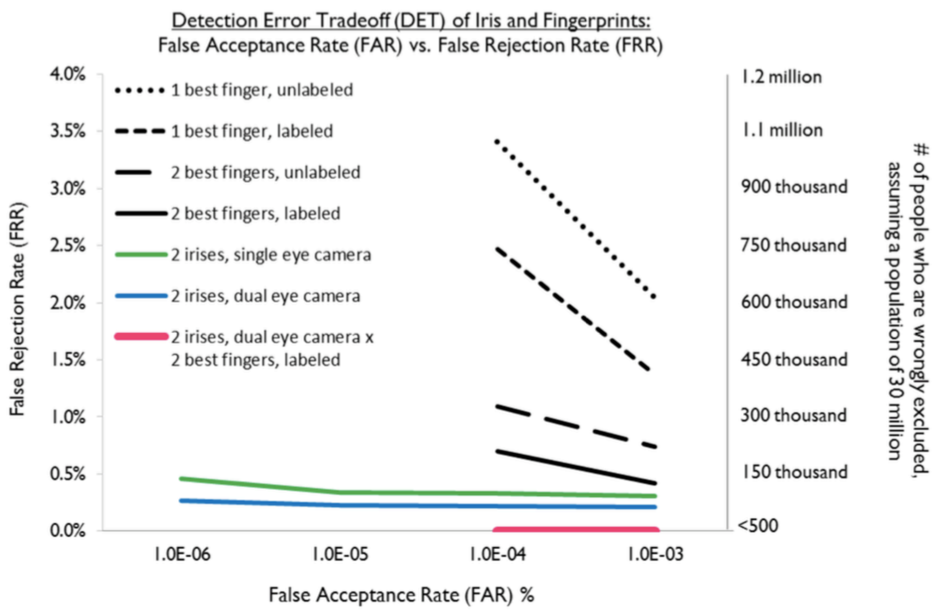
Bigger time interval between prints => decreased genuine match score

As subject's age increases /or/ image quality decreases => decreased GMS

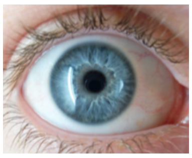
Probability of true acceptance at operational FAR settings remains close to 1

But with poor quality the uncertainty becomes large

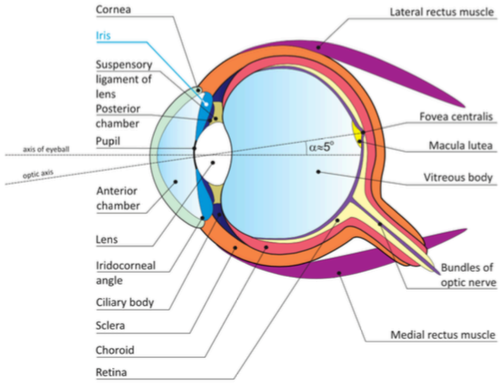
Motivation for the Next Modality ;-)

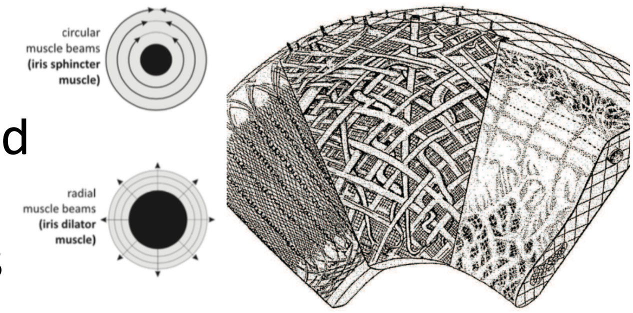


IRIS

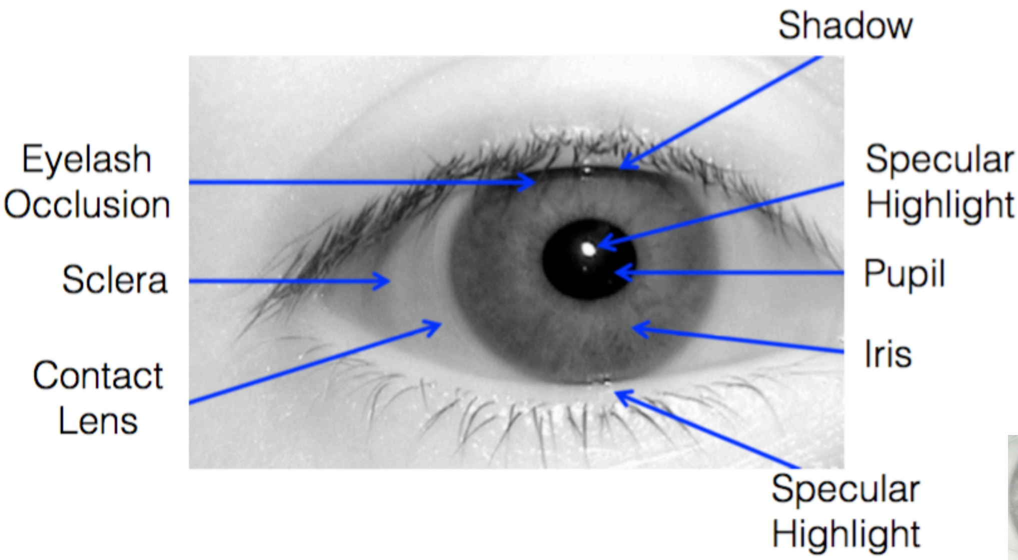


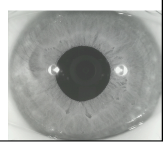
- The **texture** of an iris is the source of an **iris code**
- Iris codes are **unique** to an iris and stable across a range of conditions
- Iris structure: complicated 3D meshwork of muscle beams, blood and nerves



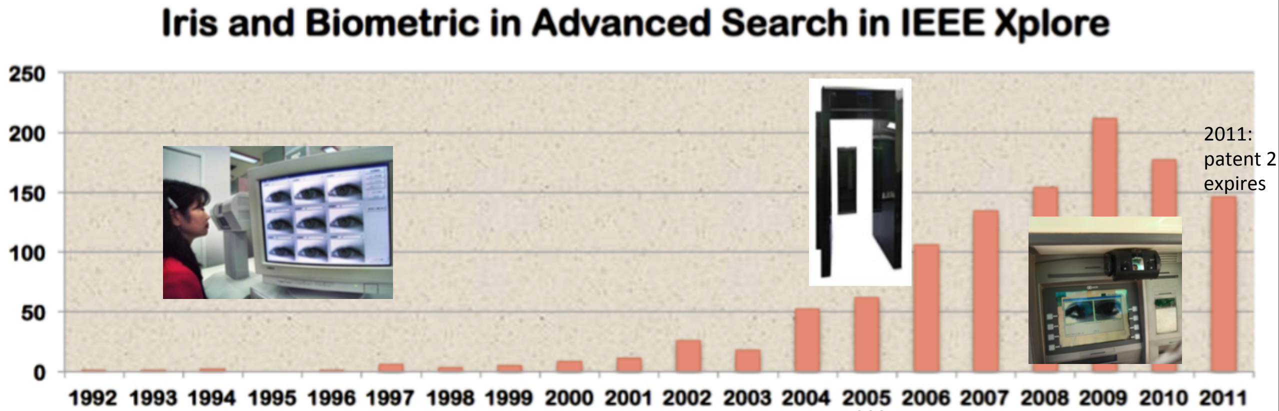


Typical (IR) Image for Iris Recognition





A Brief Historical Timeline



1987: Flom & Safir patent

1994: Daugman patent

1995: IriScan, Oki commercial systems

2003: UAE border control system begins

2005: "Iris on the move" from Sarnoff & patent 1 expires

2005, 2006: NIST Iris Challenge Evaluations

2009: Cairo-Amman Bank single-factor authentication at ATMs

What is Iris Biometrics Today?

- A particular reference implementation developed by Daugman
- An industry with successful applications and strong projected growth
- An active, fast-growing research area with some important open questions and new topics opening up

Do you recognize the image?

1984: titled Afghan Girl

The First World's Third World *Mona Lisa*

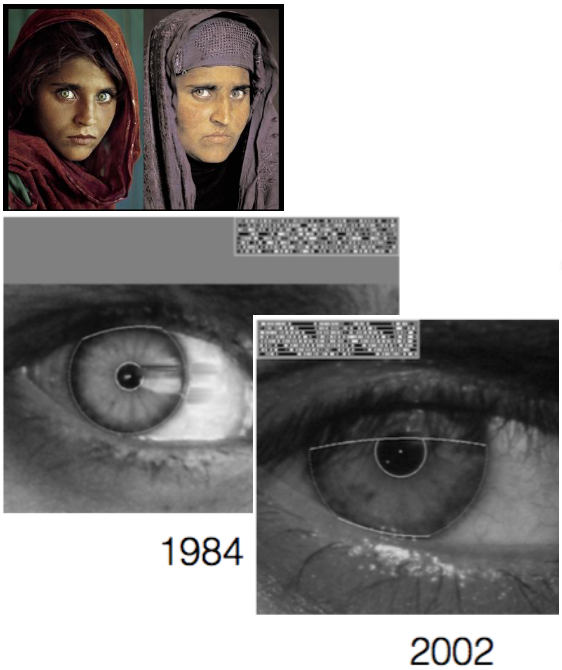
Cover is one of the most famous of the National Geographic

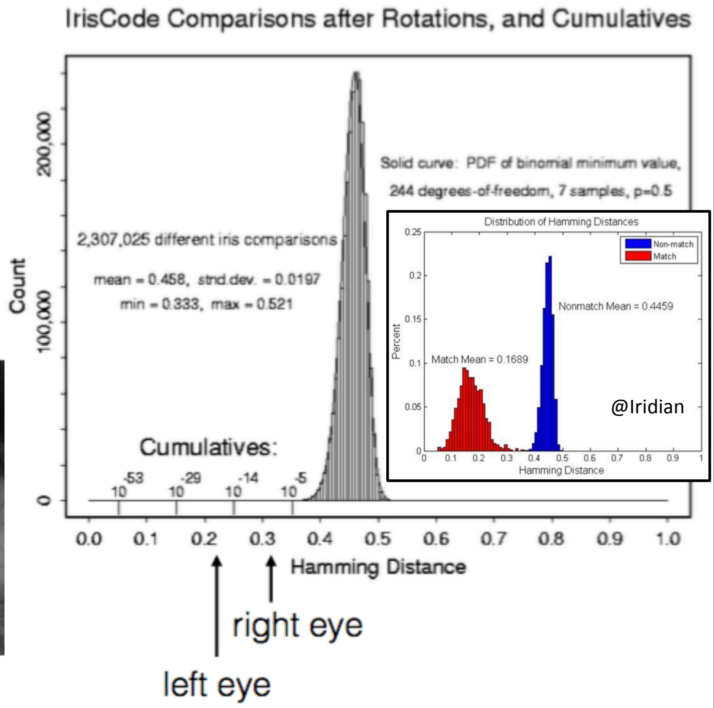
Unsuccessful attempts to locate her afterwards

2002: Sharbat Gula

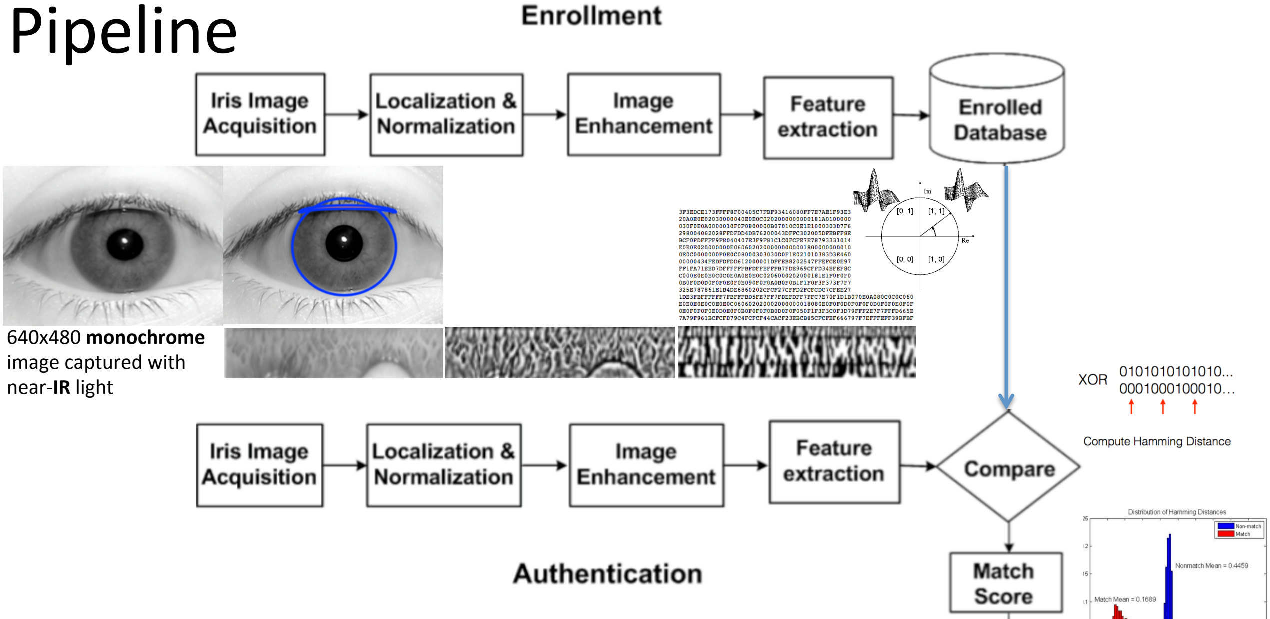
Confirmed by Daugman using iris recognition

Calculation?





Pipeline



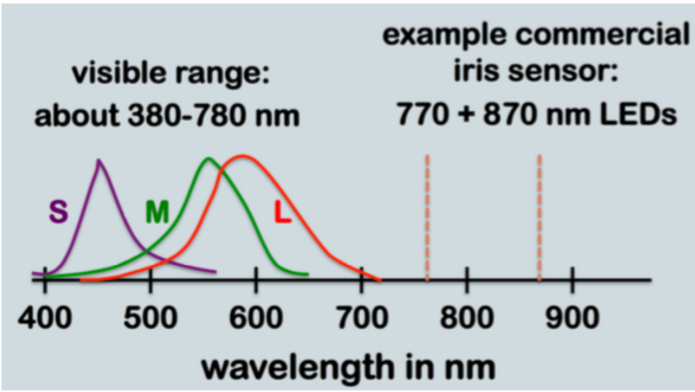
640x480 **monochrome** image captured with near-**IR** light

Most iris recognition systems expect 100 to 200 pixels across the iris in the radial direction for successful processing

Acquisition

The system typically captures a **series** of images of the eye and based on a **quality** evaluation scheme, retains only few (or one) images that are deemed to have sufficient iris texture information for further processing

Why using near-IR spectrum?

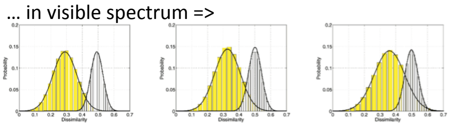



Panasonic BM-ET 330


Oki IrisPass

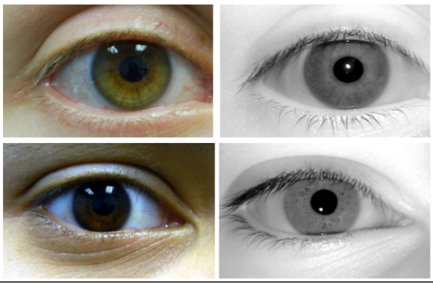

Retica MobileEyes

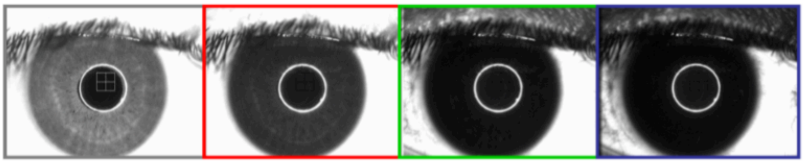
... in visible spectrum =>



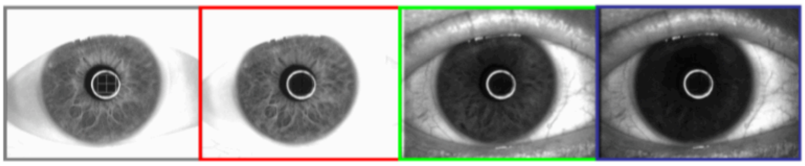
Poor → Very Poor

Because using near-IR illumination allows usable texture imaging even for **dark** irises

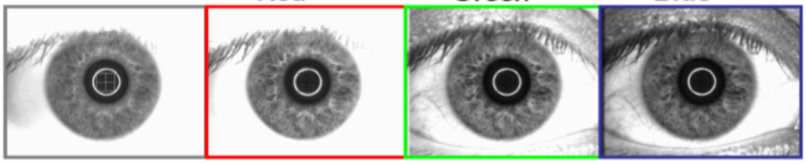




(a) Dark brown colored iris



(b) Green colored iris



(c) Blue colored iris

Negative impacts?

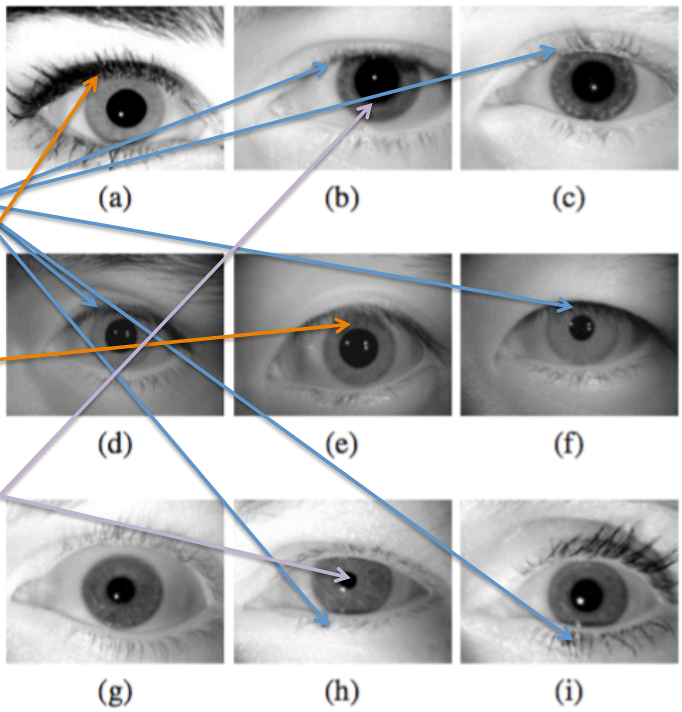
partially closed eyelids

intruding eyelashes

extremely dilated or constricted pupil

harsh or non-uniform illumination

low resolution



ICE db (LG EOU 2200 cam)

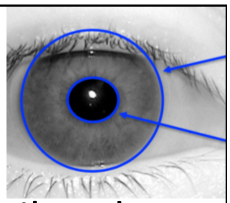
CASIA v3 db (OKI IRISPASS device)

MBGC db (LG EOU 2200 cam)

Segmentation

- To determine pixels in the image that correspond to the iris region
- Challenges:
  - contains numerous “edge” like features that are randomly distributed (no go for appearance-based schemes)
  - boundaries (especially the limbus boundary) may not be very sharp (over-segmentation or under-segmentation problem)
  - irregularly shaped due to eyelids
  - detecting (and masking) eyelashes, eyelids, reflections

Segmentation Assumptions



(for Integro-Differential Operator)

- The contrast in image intensity between the pupil and the iris offers a good cue for identifying the **pupillary** boundary (cataract problem)
- The contrast between the iris and the sclera can be used to detect the **limbus** boundary
- Both these boundaries can be approximated using **circles**
- The magnitude of the edge pixels contributing to these boundaries is **stronger** than those pertaining to other circular contours in the image

Daugman’s Integro-Differential Operator

(or Circular Edge Detector)

compute intensity gradient of pixels lying on the circumference of the circle by:

sum of gradient values is normalized

order statistic

pixel intensity

$$max(r,x_0,y_0)\left|G_\sigma(r) * \frac{\partial}{\partial r}\oint_{r,x_0,y_0}\frac{I(x,y)}{2\pi r}ds\right|$$

pupillary boundary parameters

gradient is computed along the line connecting it to the center of the circle

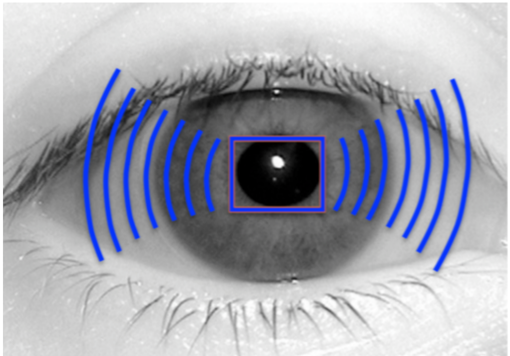
image is convolved with a radial Gaussian filter – to ensure that inner edges are reasonably blurred

What About the Limbus Boundary?

- Arc of integration is constrained to the near-**vertical** pixels on the circumference of the circle
- Necessary since the contour of the limbus may be interrupted by the upper and lower eyelids – the effect of those pixels has to be minimized

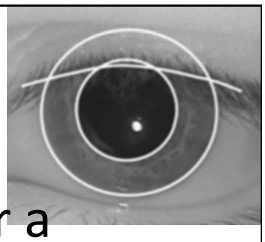
How to Limit the Search Space?

The bounding box of the largest dark region might be a range to search for x_0, y_0 with r in a feasible range



Note:
Typically the iris is not concentric with the pupil

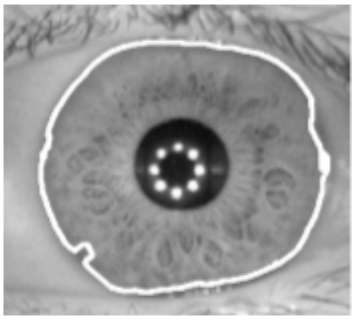
Postprocessing



- Eyelid can be detected by searching for a **parabolic edge** within the region defined by the outer circle – use **spline**-fitting procedure
- Detect eye-lashes infringing into the iris texture by searching for strong near-**vertical edges** in the segmented iris

But Boundaries are Actually Non-Circular !?

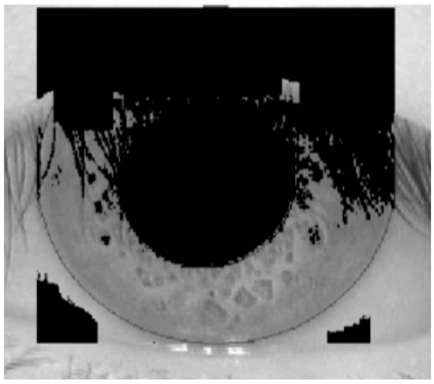
- Fitting ellipses
- Active contours[1] – simultaneously demarcates the iris from the sclera as well as the eyelids/ eyelashes
- Fitting circles and adjusting using active contours[2]



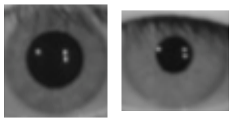
[1]Shah and Ross, IEEE TIFS, 2009 & our primary literature
[2]Park and Savvides, Encyclopedia of Biometrics, 2009

Noise Mask

Records the locations of undesired iris occlusions: eyelids, eyelashes, shadows, specular reflections

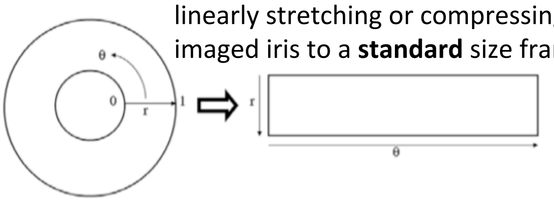


Normalization



- Addresses the problem of variations in pupil size across multiple images, subjects, sensors, imaging distances (main concern: dilation, contraction)
- Daugman's rubber sheet model

linearly stretching or compressing the imaged iris to a **standard** size frame



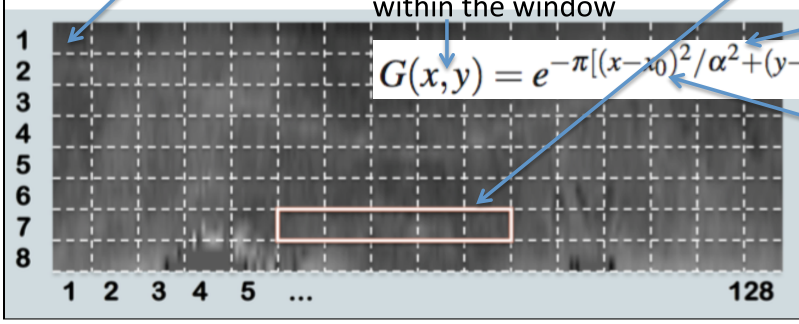
Iris Encoding => Iris Code

- Extracting a numerical feature set
- Typical approach: apply a 2D Gabor filter at *M*N* grid **points** on the image

convolution

within the window

effective width & length



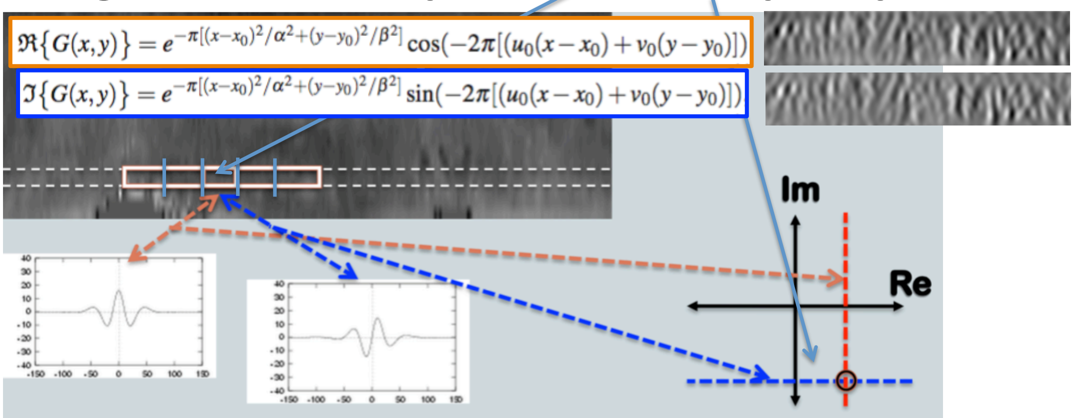
$$G(x,y) = e^{-\pi[(x-x_0)^2/\alpha^2+(y-y_0)^2/\beta^2]}e^{-2\pi i[u_0(x-x_0)+v_0(y-y_0)]}$$

center position

give the wave direction with a spatial frequency $\omega_0 = \sqrt{u_0^2+v_0^2}$

Iris Encoding => Iris Code

Applying the Gabor filter at a **location** on the iris image results in a **point in a complex plane**



How to obtain a point in a complex plane?

- Demodulate Gabor output to compress the data
- By quantizing the **phase** information into four different levels, one for each quadrant of the complex plane

demodulation and phase quantization process:

normalization in polar coordinates => also filter

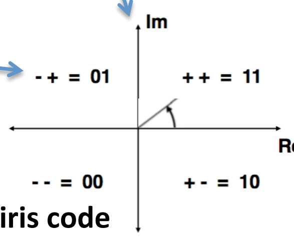
center frequency

$$H(r,\theta) = \underbrace{e^{-i\omega(\theta-\theta_0)}e^{-(r-r_0)^2/\alpha^2}e^{-i(\theta-\theta_0)^2/\beta^2}}$$

$$h_{Re,Im} = sign_{Re,Im}\int_\rho\int_\phi I(\rho,\phi)e^{-i\omega(\theta_0-\phi)}e^{-(r_0-\rho)^2/\alpha^2}e^{-(\theta_0-\phi)^2/\beta^2}\rho d\rho d\phi$$

normalized iris

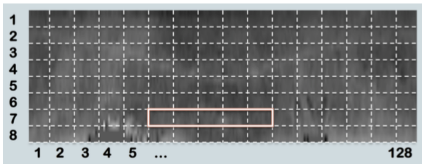
complex valued bit whose real and imaginary components are dependent on the **sign** of the integral

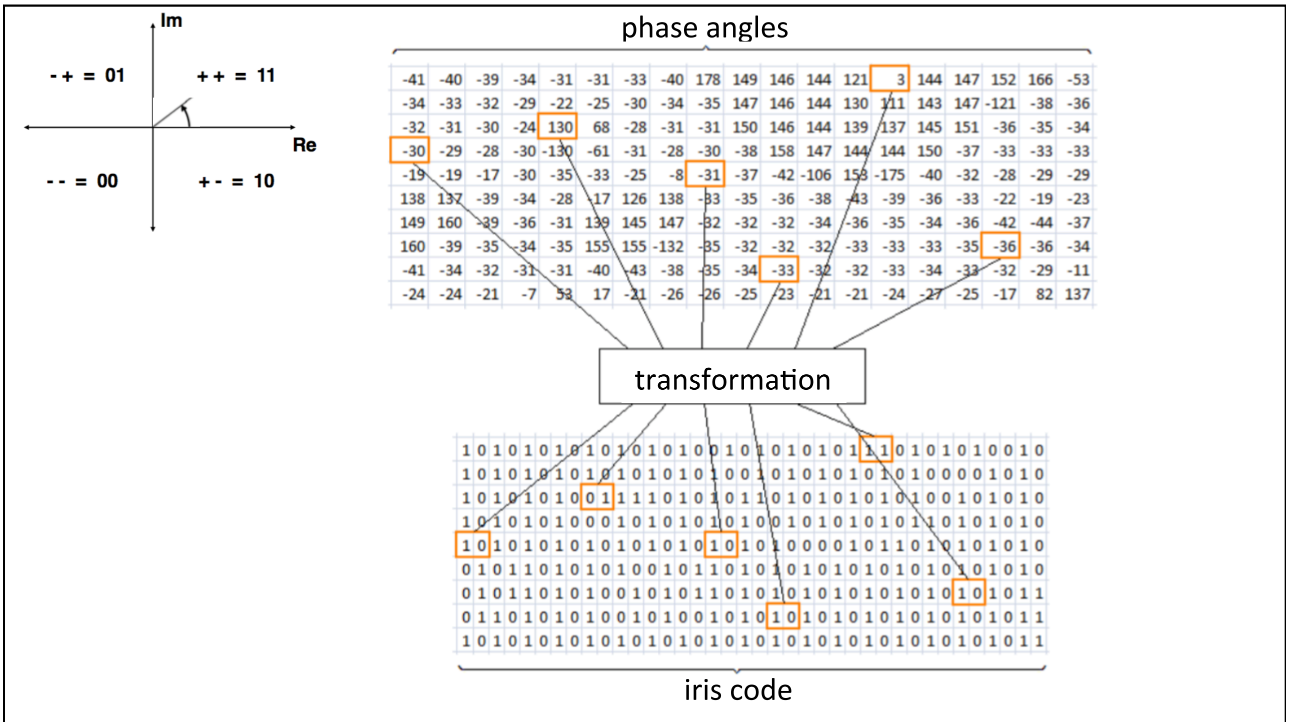


the phase of the complex value makes **two bits in the iris code**

How Big is the Iris Code?

- The standard commercial iris code is **2048** bits (plus 2048-bit mask)
- 2048 / 2 = 1024 sample points in the iris
- 1024 = 8 radii * 128 samples is a plausible sampling configuration



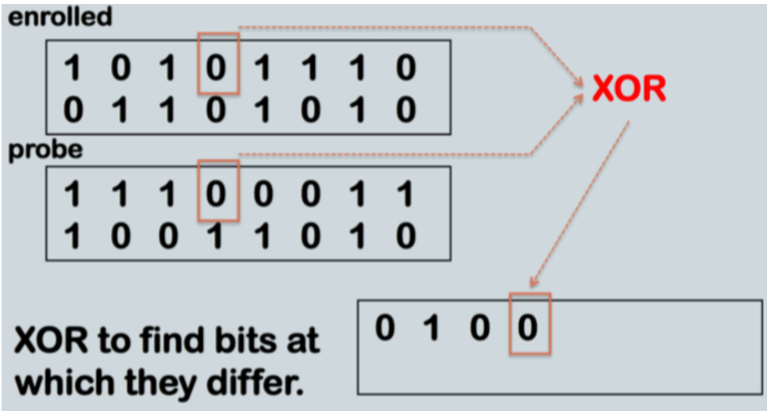


Why Gabor filters?

- Possible relevance to human vision
- Daugman: "Different members of this optimal filter family are an excellent description of the 2D neural receptive fields found in the visual cortex."
- Good theoretical properties

Matching

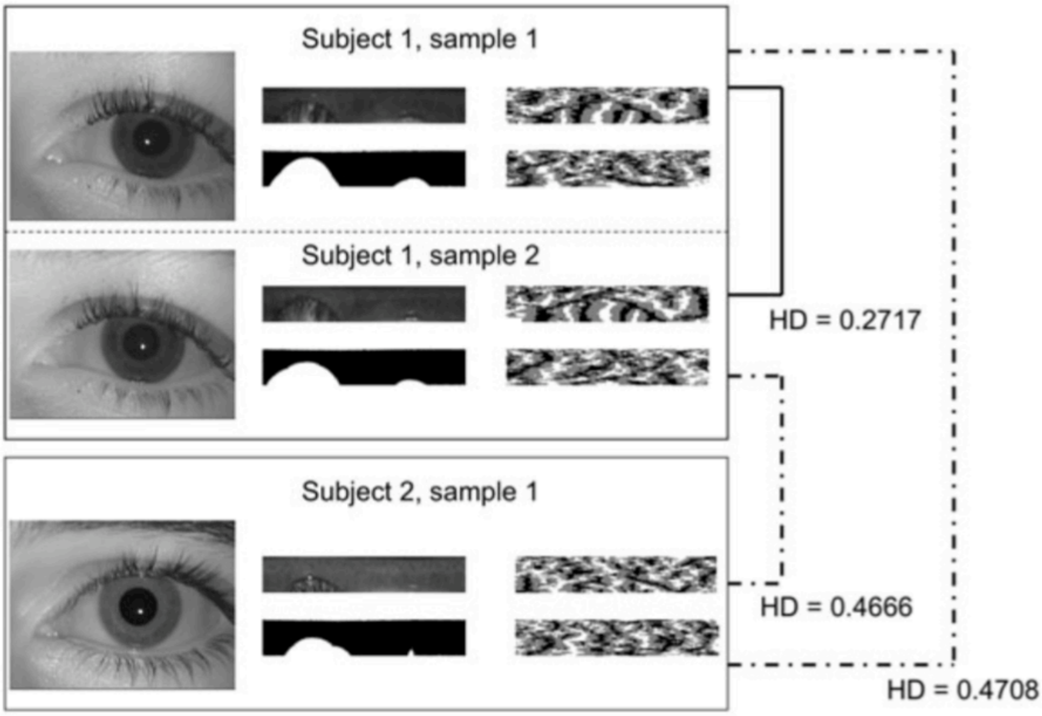
Metric: Hamming distance



Matching

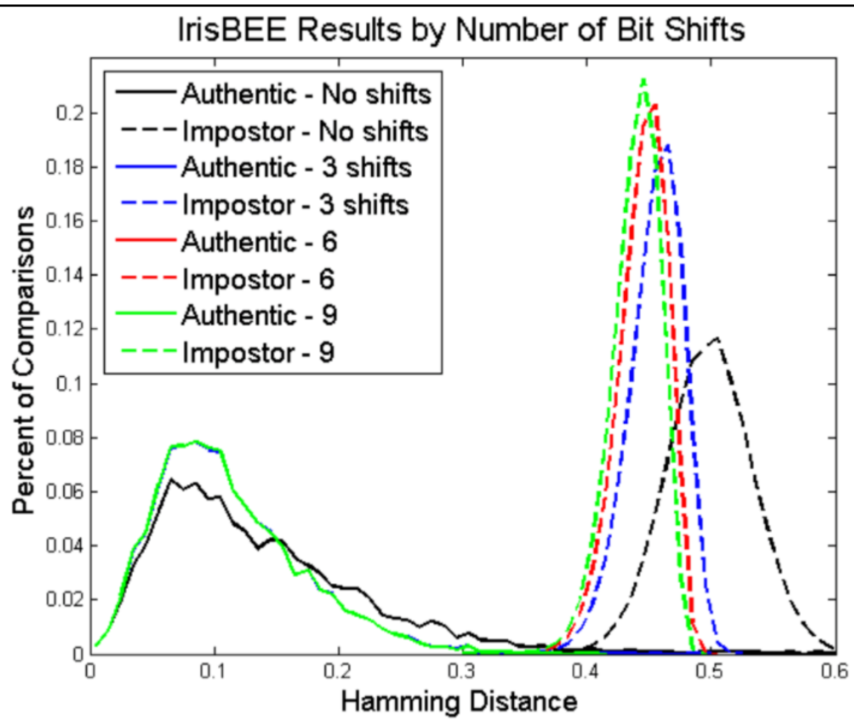
- Normalized Hamming distance – fraction of bits that differ, using **masks** (1 = iris, 0 = occlusion)
- Min. value of 0.0 means **no** difference
- Max. of 1.0 means **totally** different
- 0.5 means **random** agreement
- The denominator helps in normalizing the total number of bits that disagree in the interval [0,1]

$$\frac{|\,(\text{Mask}_1 \text{ AND } \text{Mask}_2) \text{ AND } (\text{Code}_1 \text{ XOR } \text{Code}_2)\,|}{|\,\text{Mask}_1 \text{ AND } \text{Mask}_2\,|}$$



Matching & Rotation of the Iris?

- To account for varying head tilt we **rotate** codes
- Metric is computed for a range of circular **shifts** and the min is kept
- This corrects the authentic distribution but **increases** time for matching and **changes** the impostor distribution =>

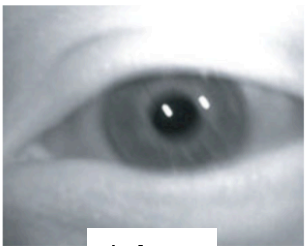


Iris Quality Assessment

Factors influencing the quality:

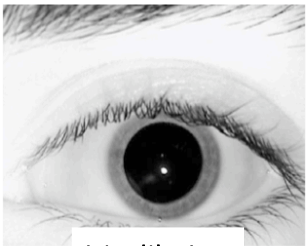
- Occlusion
- Defocus / out-of-focus
- Motion blur
- Non-uniform illumination
- Low resolution / large imaging distance
- Iris dilation
- Off-angled imaging
- Presence of accessories such as fake or printed contact lenses

occlusion

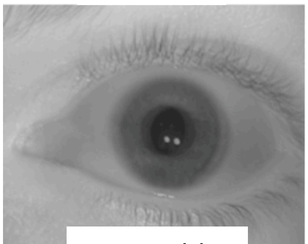


defocus

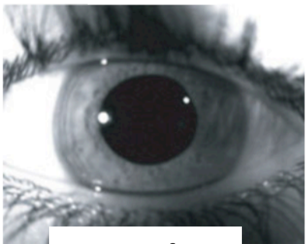
low resolution sensor



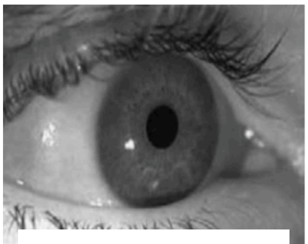
iris dilation



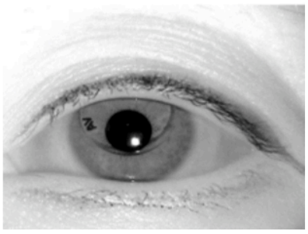
motion blur



non-uniform illumination



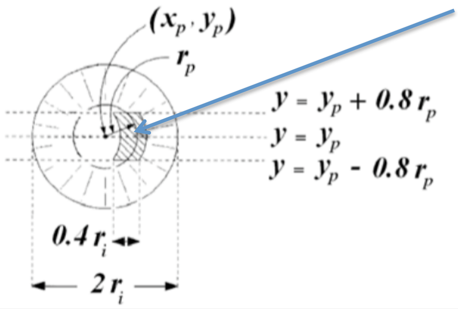
off-angled imaging



presence of a printed contact lens

Sharpness/Focus Evaluation

Sharpness is essentially computed as the normalized magnitude of the intensity gradient **near** the pupillary boundary



ROI (Region of Interest):

$$(y_p - 0.8r_p) < y < (y_p + 0.8r_p)$$

$$-\sqrt{r_p^2 - (y-y_p)^2} + 0.1r_i < x - x_p < -\sqrt{r_p^2 - (y-y_p)^2} + 0.2r_i$$

Sharpness/Focus Evaluation

1. From ROI compute median values:
   - of the pixels falling in the pupil region, M_p
   - of the pixels falling in the iris region, M_i
2. For all the pixels in ROI with intensities in $[M_p..M_i]$ calculate absolute values of their horizontal gradients
3. Calculate the average of the top 20 values = S
4. Calculate sharpness: $S / (M_i - M_p)$
5. If >0.5 => considered to be well focused

Sharpness/Focus Evaluation Using 2D Fourier Transform

- Quantifying the energy of **high** spatial frequencies over the **entire** image

image

$$F(\mu,\nu) = \frac{1}{(2\pi)^2}\int\int I(x,y)\exp(-i(\mu x+\nu y))dxdy$$

frequencies

- Defocused image is related to the FT of the corresponding in-focus image by the following model:

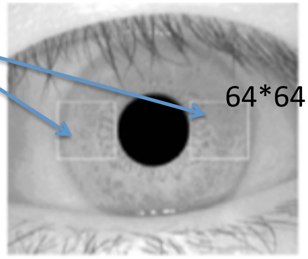
$$D_\sigma(\mu,\nu) = \exp\left(-\frac{\mu^2+\nu^2}{\sigma^2}\right)F(\mu,\nu)$$

blur parameter

Sharpness/Focus Evaluation Using 2D Fourier Transform

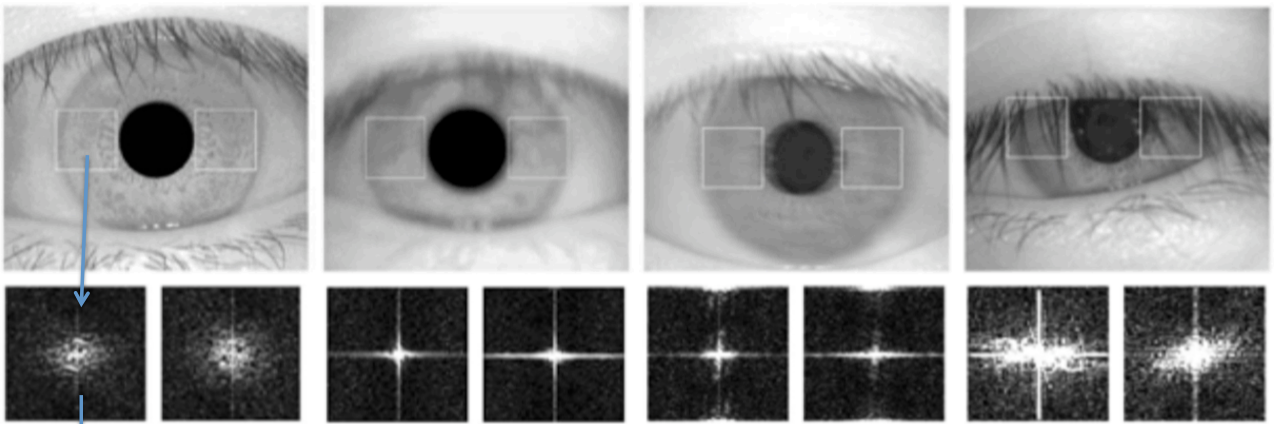
- Defocusing primarily **reduces** the highest frequencies in the image, while the lower frequency components are virtually unaffected
- How to then identify a defocused image?
- **Measure its total power in the Fourier domain at higher spatial frequencies** (as they are most reduced by defocus)

Analyzing the Fourier Spectra of Local Iris Regions



Applicable to:

- out-of-focus blur
  - Fourier spectrum is supposed to be largely dominated by the **low** frequency components
- motion blur
  - lacks **middle** and **high** frequency components and has frequency distribution similar to that of an out-of-focus image
- occlusion due to the eyelashes, eyelids
  - contains significant **middle** and **high** frequency components



Compute quality descriptor values: $D = \left[(F_1 + F_2 + F_3); \frac{F_2}{F_1 + F_3} \right]$

power of the low, middle, and high frequency components

$$F_i = \iint_{\Omega = \{(u,v) | f_1^i < \sqrt{u^2+v^2} <= f_2^i\}} |F(u,v)| \, dudv \quad i = 1,2,3$$

radial frequency pair that form the extrema of the corresponding frequency components

2D Fourier spectrum of the iris region

$$D = \left[(F_1 + F_2 + F_3); \frac{F_2}{F_1 + F_3} \right]$$

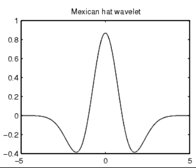
Two discriminating frequency features:

- **Total** spectrum power of the iris region that can effectively discriminate severely occluded iris images from high quality images
- Ratio of the **middle** frequency power to the other frequency powers – for a clearly focused image this ratio is high when compared to out-of-focus or motion blurred images

Mean of the resulting two local quality descriptors is regarded as an appropriate quality measure of the iris image

Measuring the energy from 2D **wavelets** at local concentric bands of segmented iris

- Local quality measure is used as a **weighting** scheme in the matching process
- Method uses **higher** weights for **inner** regions of the iris that are more stable compared to the outer regions that are more prone to occlusions



Continuous Wavelet Transform
(convolution with a **series** of wavelet functions)

Mexican hat wavelet (band pass filter for edge detection at scales *s*)

$$w(s, x_0, y_0) = \frac{1}{\sqrt{s}} \int \int_{R^2} I(x,y) \phi\left(\frac{x - x_0}{s}, \frac{y - y_0}{s}\right) dxdy$$

scale & translation

image

Measuring the energy from 2D **wavelets** at local concentric bands of segmented iris

capture various features at multiple scales:

$$w^{mul}(s_1, s_2, s_3) = w(s_1) \times w(s_2) \times w(s_3)$$

+iris is partitioned into multiple concentric bands of fixed width

energy of the t-th band

$$E_t = \frac{1}{N_t} \sum_{i=1}^{i=N_t} | w_{t,i}^{mul} |^2$$

total number of wavelet coefficients

(high energy value suggests a good quality image)

quality index of the entire iris

$$Q = \frac{1}{T} \sum_{t=1}^{T} (m_t \times \log(E_t))$$

total number of bands

weight

mean radius of the t-th band with respect to l_c

center of the pupil

$$m_t = \exp\{- \| l_t - l_c \|^2 /(2q)\}$$

Measuring the energy from 2D **wavelets** at local concentric bands of segmented iris

Possibility to incorporate local quality measures into the matching scheme =>

Weighted Hamming distance

iris codes

$$HD_w = \frac{1}{B} \frac{\Sigma_{i=1}^{B} \sqrt{E_{g(i)}^X \times E_{g(i)}^Y} \times (X_i \otimes Y_i)}{\Sigma_{i=1}^{B} \sqrt{E_{g(i)}^X \times E_{g(i)}^Y}}$$

total number of bits

local quality measures of the $g(i)$-th band

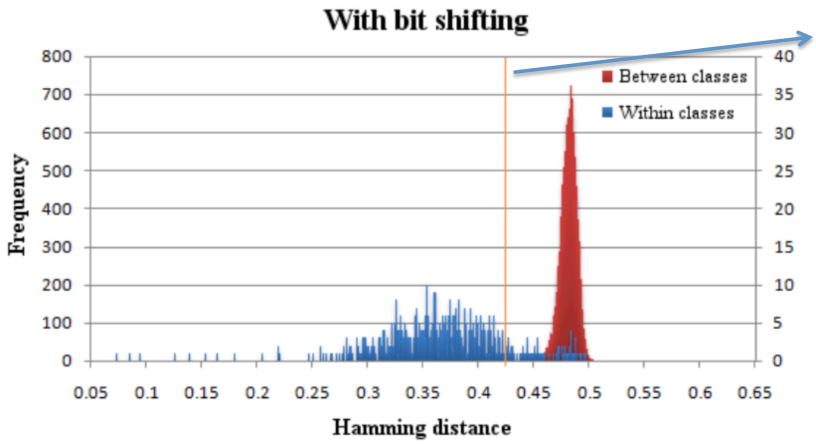
index of the band that contains the i-th bit

Performance Evaluation

- Iris is substantially diverse across the population
- Even monozygotic twins exhibit structural differences
- Daugman's experiment: 632,500 iris images (316,250 persons, 152 nationalities) suggest the possibility of a decision policy that could yield **zero** error rates
- This rate is predicated on the **quality** of the iris image, which must be strictly monitored to ensure reasonable textural clarity
- 2006 NIST Iris Challenge Evaluations:
  - involving a broad range of image quality
  - FNMR=1.1 … 1.4 % @ FMR=0.1 %

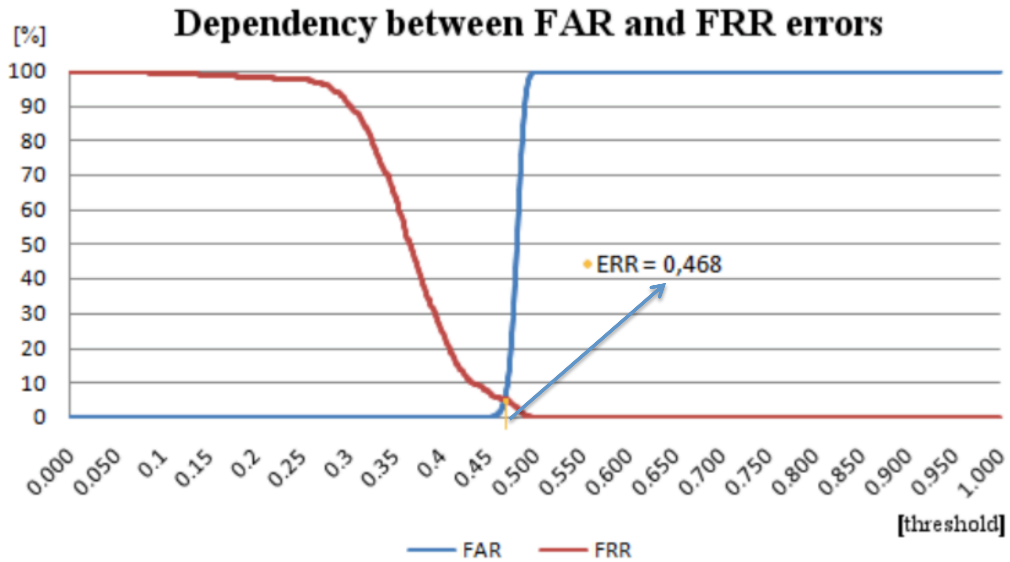
Playing Around with Our Prototype

Histogram of comparisons within classes and between classes on CASIA v1 db:



HD = 0.427 =>
FRR = 11.58 % @ FAR = 0 %

Playing Around with Our Prototype



2006 NIST Iris Challenge Evaluations Details

FRR @ FAR = 0.1 %, where three FRR estimations are reported due to 30 different partitions of the iris database (ICE db) for test purposes

Group	**FRR [%]**		
	Minimum	***Median***	***Maximum***
Sagem-Iridian	0.473	1.22	2.31
Cambridge	1.06	1.93	3.29
Iritech	0.993	2.06	3.84

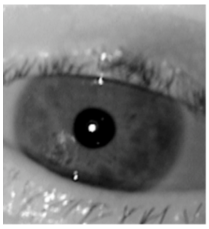
Just to contrast our results (unfortunately on CASIA db, thus not directly comparable):
FRR = 7.7 % (@ FAR = 0.1 %)

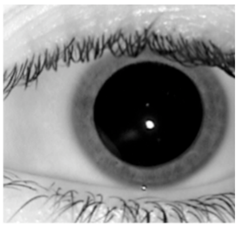
Pupil Dilation Influence

Why it varies?

- Change in ambient lighting
- Emotional state
- Health
- Medications
- Age
- Decision making

$$\text{Pupil Dilation Ratio} = \frac{\text{Pupil Radius}}{\text{Iris Radius}}$$

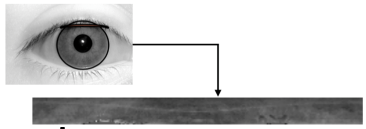




Statements

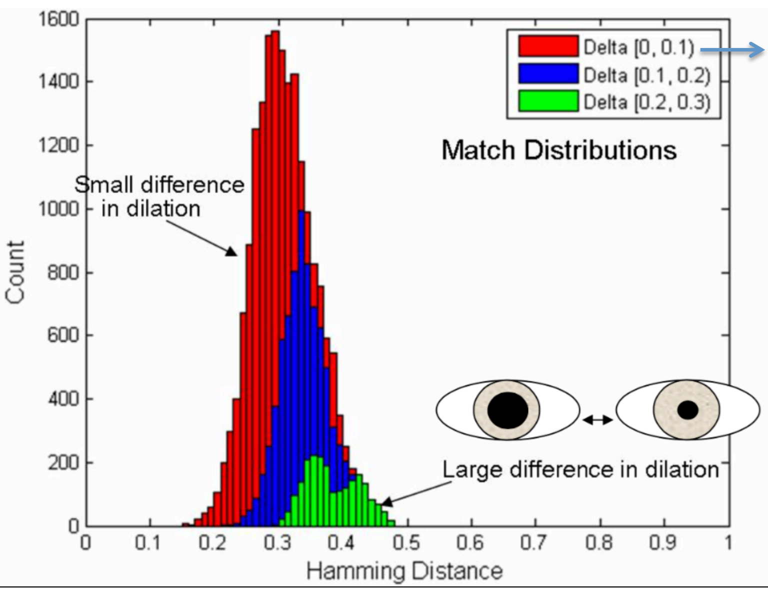
- "Although the iris stretches and contracts ... Such distortions in the texture can readily be reversed mathematically ... to extract and encode an iris signature that remains the same **over a wide range** of dilations.", J. Daugman, U.S. Patent #5291560, 1994
- "Even though the visible portion of the iris changes as a function of pupil dilation, this **does not adversely affect** authentication.", Alfred C. Weaver, How Things Work: Biometric Authentication, IEEE Computer, Feb. 2006

Experiment



- Fact: Dilation info is available at the segmentation stage, but current systems **discard** this info in going to the normalized iris image
- Data: 18 subjects, 632 iris images acquired via LG 2200, 28% of images taken with lights off, to induce normal dilation
- Method:
  - Difference in dilation ratio for each pair of images matched
  - Modified ICE baseline software to compute and match iris codes
  - **Assumption:** authentic and impostor distributions conditioned by dilation

Match Distributions



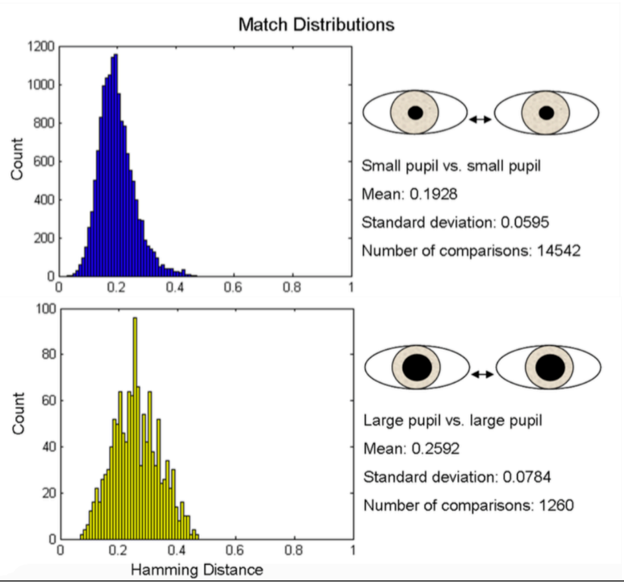
Difference in dilation ratio

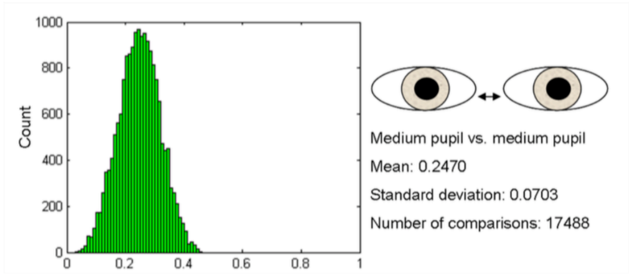
Increased difference in dilation **shifts** the authentic distribution

+No change in impostor distribution

Match Distribution Means

(Using Similar Dilation Ratio)

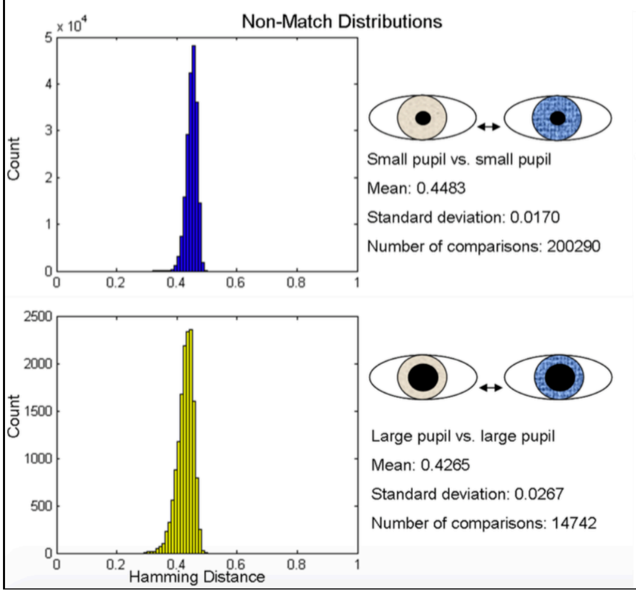


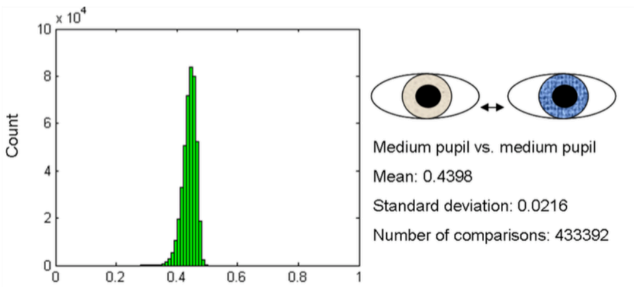


Larger pupil dilation **increases** the mean of the authentic distribution

Non-Match Distribution Means

(Using Similar Dilation Ratio)





Larger pupil dilation **somewhat decreases** the mean of the impostor distribution

This is somewhat surprising; but likely not a problem in typical practice

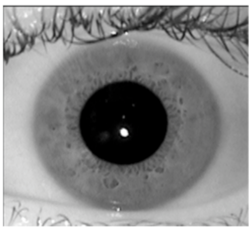
Dilation Influence – Conclusion

- When the difference in pupil dilation between two images is **large**, the **FNMR increases**

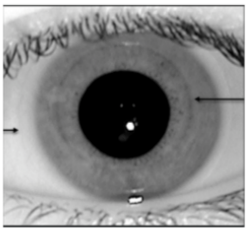
- NIST's IREX 1 report found similar results using different data
- The draft ISO standard for iris data format has a field for the pupil dilation ratio -> evidence that the role of dilation is now acknowledged

Influence of Lenses

- Myth: Wikipedia, "Iris Recognition", Oct. 2010: "Iris recognition efficacy is rarely impeded by glasses or contact lenses."
- Current page no longer contains this language, but instead says: "Iris recognition works with clear contact lenses ..."
- Let's dig into this claim >>>



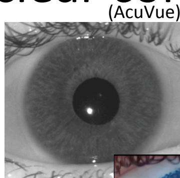
iris **without** contact lens



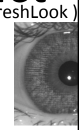
iris **with** contact lens

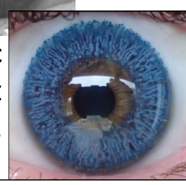
clear contact lenses

(AcuVue)



(FreshLook)





cosmetic contact lenses

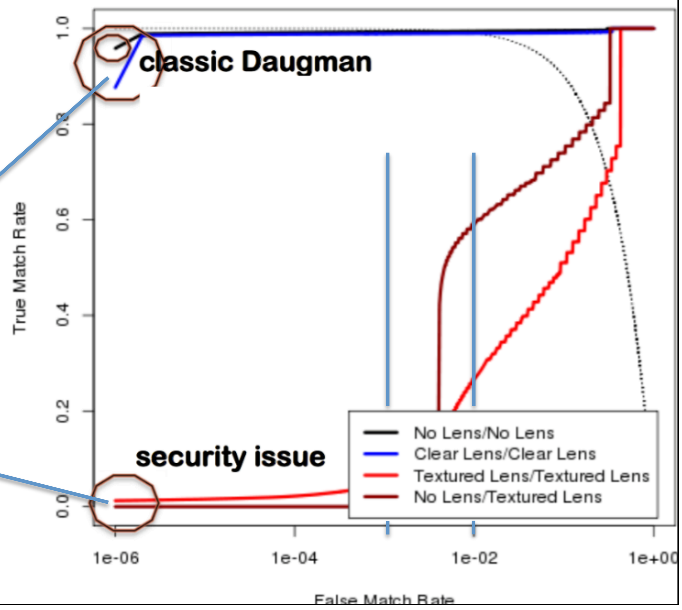
ROC Analysis

NDCLD12 db

VeriEye 4.1 matcher

Clear lenses increase the FNMR slightly;
a minor social impact issue

Textured lenses are a security issue;
automatic detection needed



Variation within One Brand

For just the Freshlook line
there are 4 types
with 3 to 12 colors each

So how many different types & colors of textured lenses are out there?

Malicious Use of Contact Lenses

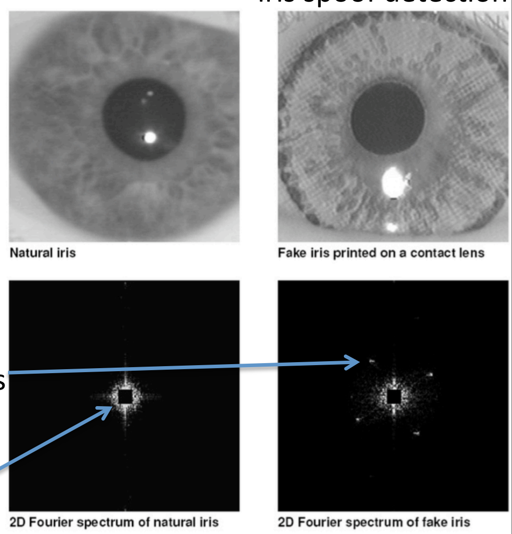
- Evade detection: intentionally create a false non-match (**easy** to do)
- Create a synthetic identity (**maybe**)
- Impersonate a target identity (maybe, but even **harder**)

Detecting Contact Lenses

Daugman's frequency analysis approach:

- Fourier analysis for lens detection
- A **printed** iris typically exhibits some artifacts that can be detected by analyzing the 2D Fourier spectrum of the iris image

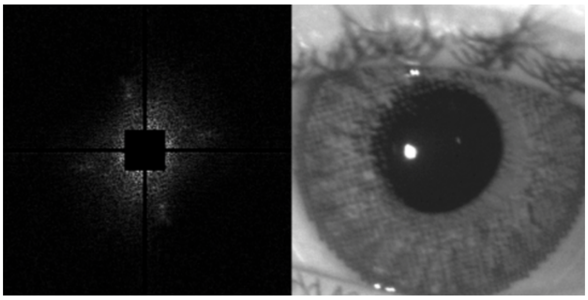
iris spoof detection



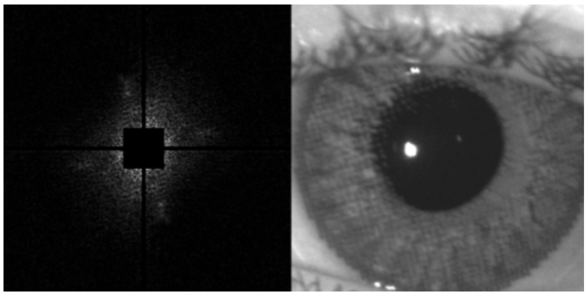
dot matrix printing process generates 4 points of spurious energy

the central square of spectrum has been blanked out to prevent its domination

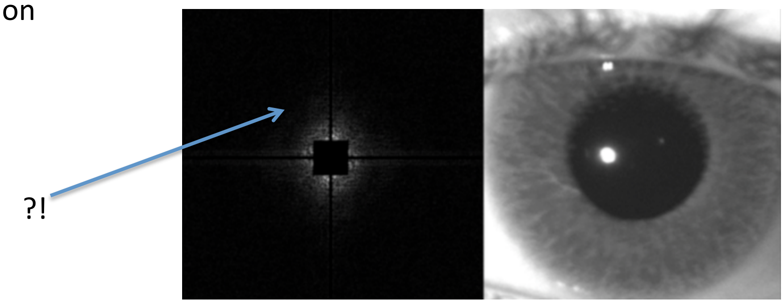
Various Producers



CooperVision



FreshLook

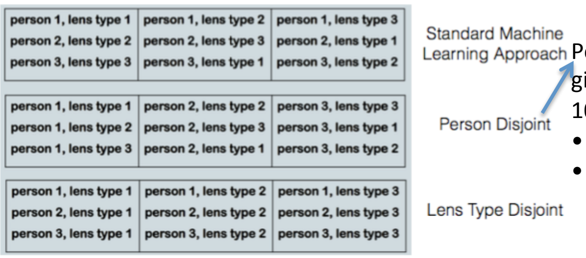


AcuVue

Detecting Contact Lenses

Other approaches:

- Texture analysis (*features*: iris edge sharpness, iris-texton feature, co-occurrence matrix feature,… + *classifiers*)



Person-disjoint 10-fold cross-validation gives estimated performance of 100% correct classification using:
- LBP
- Variety of supervised classifiers

Classifier	All
Naïve Bayes	98.0
Bagging	100
LogitBoost	100
JRip	100
J48	100
Random Forest	100

- Printed iris pattern does not undergo any distortions when the pupil changes in size

Lens Detection – Conclusions

- Detection of textured lenses seen in training data is a solved problem
- Detection of **new textured (Open Set Recognition)** lenses is not a solved problem

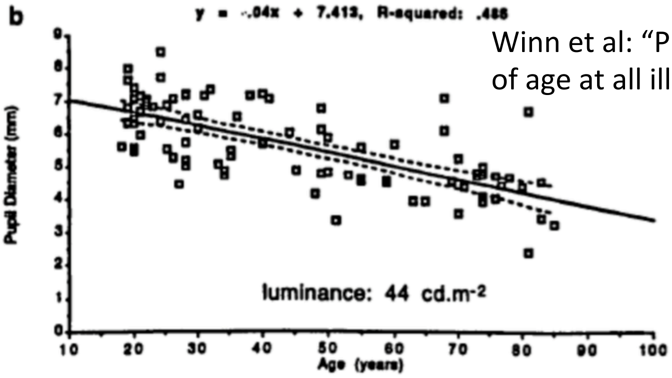
Influence of Aging

ISO/IEC 19795-1:2006 "Information technology - Biometric performance testing and reporting", Section 6.4.6.:

- "Longer time intervals generally make it more difficult to match samples to templates due to the phenomenon known as 'template aging'."
- "This refers to the increase in error rates caused by time-related changes in the biometric pattern, its presentation and the sensor."

What does medicine tell us?

From medical literature we know that the iris changes **functionally** with increased age (1994)



Winn et al: “Pupil size decreased linearly as a function of age at all illuminance levels. …”

Smaller average pupil size with age
=>
Greater time lapse translates into greater average difference in dilation
=>
Assumption: Greater difference in dilation degrades iris match scores

Flom and Safir’s Patent

- 1987 patent allows for the possibility that **re-enrollment** might be needed to maintain performance
- “... the significant features of the iris remain extremely stable and do not change over a period of **many** years.”
- “Even features which do develop over time... usually develop rather **slowly**, so that an updated iris image will permit ID for a substantial period...”

Daugman’s Patent

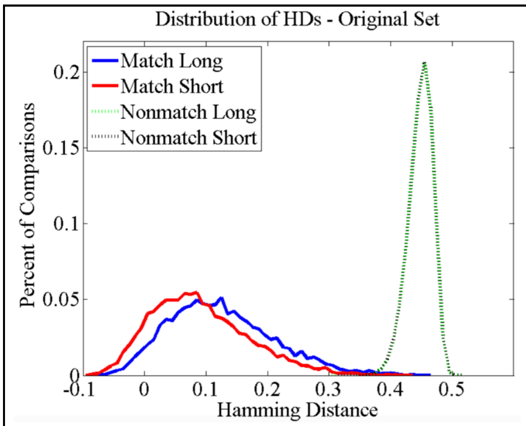
The iris of every human eye has a unique texture of high complexity, which proves to be essentially immutable over a person's life

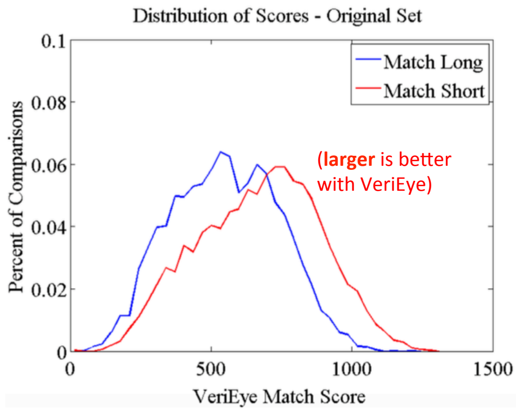
Which narrative won?

A single enrollment can last a lifetime ;-)

Experiments

- 43 subjects, 86 irises
- LG 4000 iris sensor
- Two year time lapse, 2008-2010
- 1830 images
- 9K short-term matches, 5 to 51 days
- 10K long-term matches, about 2 years
- Two matchers: IrisBEE, VeriEye





IrisBEE authentic distribution is clearly degraded

Impostor distribution seems unchanged

VeriEye authentic distribution is also clearly degraded

- 84 of 86 had an increased IrisBEE mean HD with longer time lapse
- 82 of 86 had a decreased VeryEye mean score
- 2 years time-lapse:
  - IrisBEE: 157 % to 305 % increase in FRR at thresholds of 0.28 to 0.34
  - VeriEye: 239 % to 370 % increase in FRR at thresholds of 60 to 120

Iris Template Aging is Controversial

Tome-Gonzales, 2008

Baker, 2009

Fairhurst, 2011

Fenker, 2011 **Sazonova, 2012**

Fenker, 2012 **Czajka, 2013**

Ellavarason, 2013

IREX VI, 2013

NIST released this report, which appears to find near zero “iris aging” on a large “operational” dataset (with some “strange” particularities)

Open Research Questions?

Acquisition:

- How well can iris recognition be performed with little or no explicit user **cooperation**? (on-the-move; the iris is a slightly oscillating object within a moving object (the eye-ball), which is located within yet another moving object (the head) on top of the moving body
  + long focal length needed
  + activating the NIR illuminant ;-)
- How well can iris recognition be performed using **visible** light or cross-wavelength images?

Open Research Questions?

Masking:

- How well can iris texture **occlusions** by specularities, eyelashes, shadows, etc. be found?

Iris code:

- Is there a better way to **compute** an "iris code" from the iris texture? Perhaps higher-res?
- Is there simply a **different** way that could be combined with the iris code?

Open Research Questions?

Quality:

- While initial research has shown the benefits of incorporating iris quality, its **assessment** and **use** in a real-time environment is still an open challenge

Dilation:

- What is the **best way** to deal with the effects of difference in pupil dilation between images?
  - Control acquisition?
  - Enroll multiple samples?
  - Better model of dilation?

Open Research Questions?

Lenses:

- Can the presence of contact lenses be automatically **detected**?
- Can the artifacts created by clear contact lenses be **reversed** or at least **masked** out?

Aging:

- Does template aging **occur**?
- Does the rate of template aging vary with **demographic** factors?
- Can **aging-resistant** algorithms be designed?

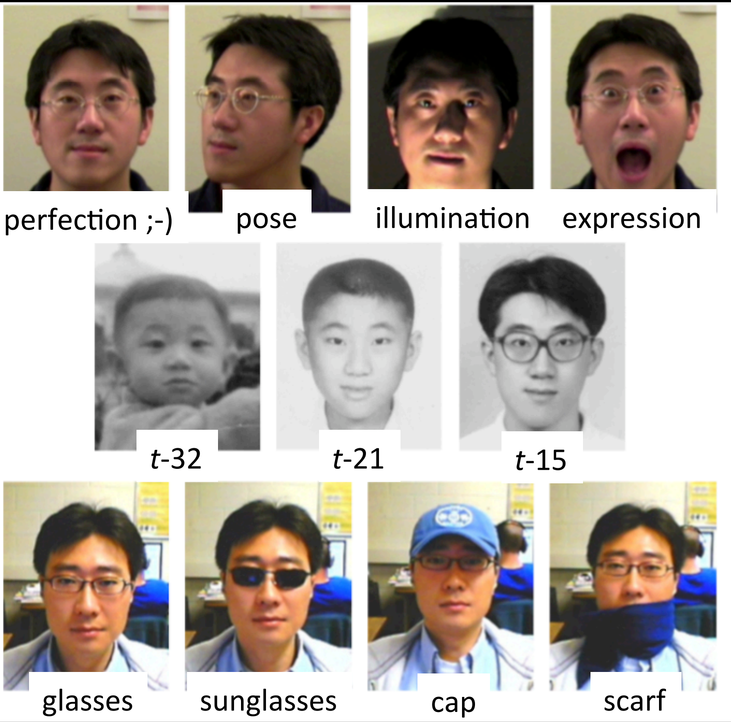
General:

- Although researchers have used match-score distributions and IrisCode statistics to infer the iris biometric's degrees of freedom, no one has yet directly used the iris's biological underpinnings to **ascertain its individuality**
  => This interesting problem has implications for using iris recognition in a court of law in accordance with Daubert's admissibility criteria and Federal Rules of Evidence (remember: as with fingerprints)

MODALITY #3: FACE

- Human face is a specific modality – why?
- Not just that it enables recognition, but gives other attributes like **gender**, **age**, **ethnicity**, and **emotional** state of a person (<u>soft</u> modalities)
- The face is considered to be the most commonly used biometric trait **by humans**

To Many Variations?



\+
make-up,
facial hair,
…

Not Enough **Inter-Class** Variations?

Twins

Family

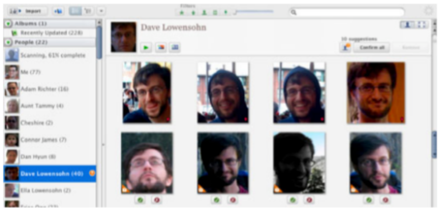
Advantages

1. Face can be captured at a longer stand-off distance using non-contact sensors => surveillance applications
2. Conveys also emotions, gender, age, ethnicity => interactive human-computer interfaces
3. There are large legacy face databases (e.g. U.S. driver's license repositories covers over 95% of the adult population), which enable **large scale** analysis of the face modality in terms of individuality or scalability
4. People are generally more willing to **share** their face images in the public domain (e.g. Facebook)

Applications

- law enforcement
- civilian identification
- surveillance systems
- entertainment/amusement systems

Australia's SmartGate system:
faster immigration clearance



Picasa: automated face tagging

Kinect:
personalizing XBOX

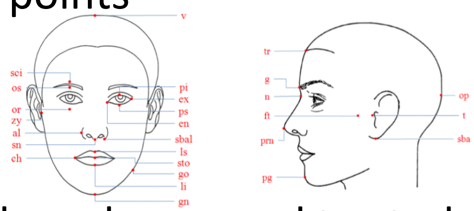
Morpho's driver license solution:
prevent a single person
obtaining multiple licenses

Humans & Face Recognition (*Psychology*)

- Fact: humans find it difficult to detect or recognize faces that are **inverted**
- *Prosopagnosia* – a disorder in which an individual loses his ability to recognize only faces
- Fact: humans perceive the face based on certain higher-level characteristics

Facial Features

- **Anthropometric** studies have attempted to characterize the dimensions of the face based on a set of **anatomically** meaningful **landmark** or fiducial points



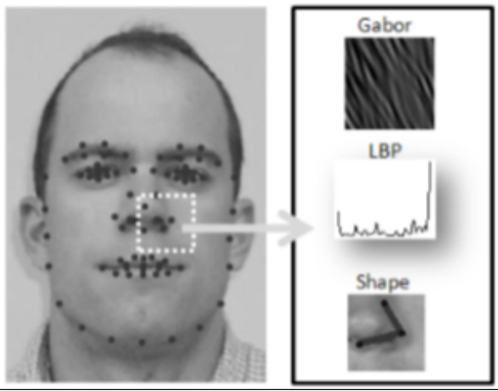
- Anthropometric measurements have been used to study the **growth patterns** in humans as well as understand characteristics of the face as it pertains to **gender** and **ethnicity**
- … but lack of **distinctiveness** for automated FR!

3 Levels of Features (again ;-)

- L1: gross facial characteristics like **geometry** of the face and global skin **color**
  - a short round face and an elongated thin face
  - faces exhibiting predominantly male and female characteristics
  - faces from different races
  - extracted even from low resolution face images (< 30 interpupilary distance (IPD))

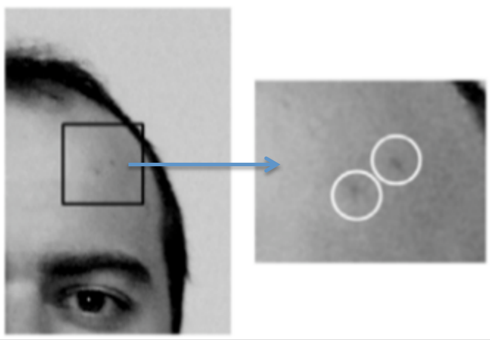
3 Levels of Features

- L2: localized face information such as the structure of the face **components** (e.g. eyes), the **relationship** between facial components, **precise shape** of the face
  - accurate face recognition
  - local regions characteristics can be represented using **geometric** or **texture** descriptors
  - + 30 to 75 IPD



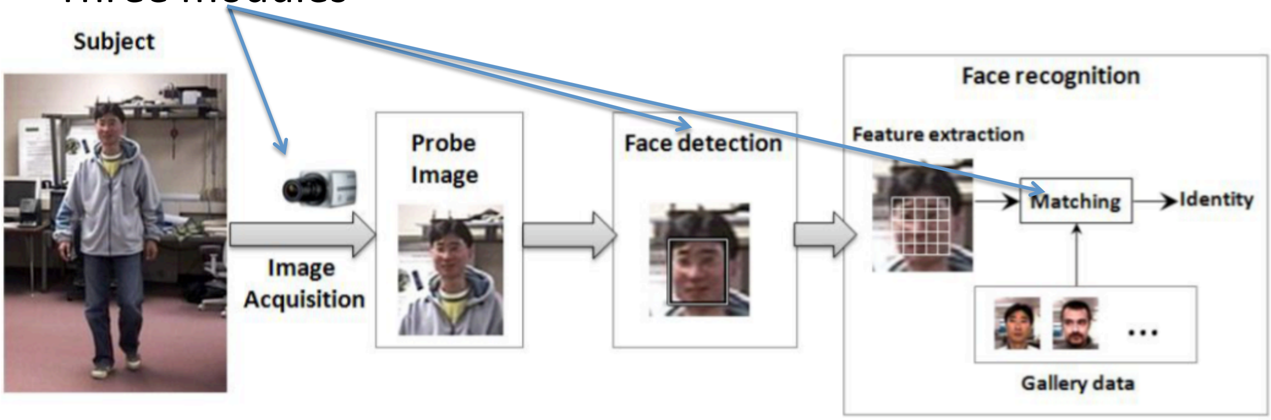
3 Levels of Features

- L3: unstructured, **micro level** features on the face, which includes scars, freckles, skin discoloration, and moles
  - discrimination of identical twins (?!)

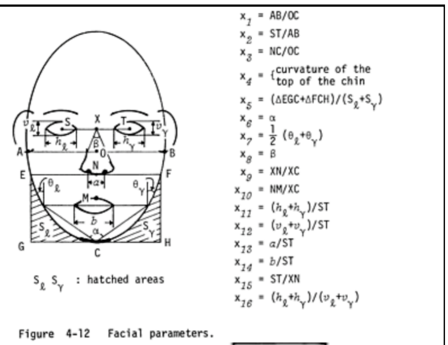


Basic FR Pipeline

- Most of the automated FR systems make use of **2D** images acquired in the **visible** spectrum
- Most commercial FR engines first detect the **two** eyes
- Three modules



From 1973 to the Present



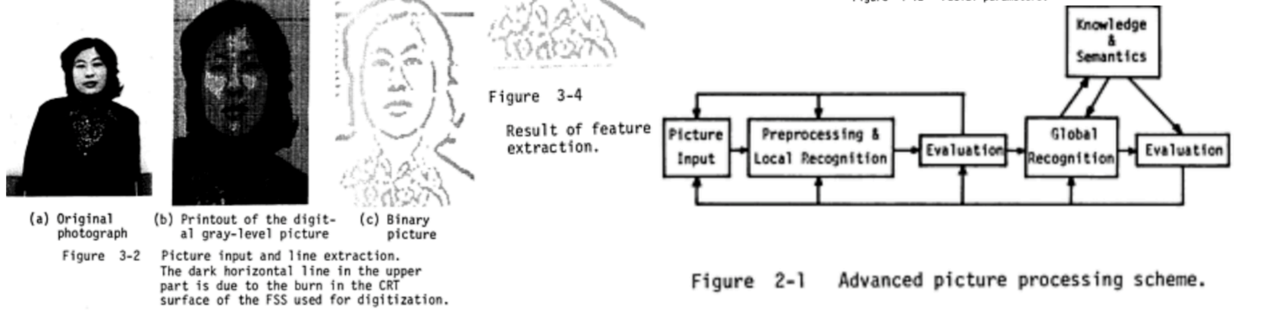


Image Credit: T. Kanade 1973

Acquisition → Normalization → Alignment → Features → Recognition

Acquisition

2D sensors >>>

Multiple 2D cameras

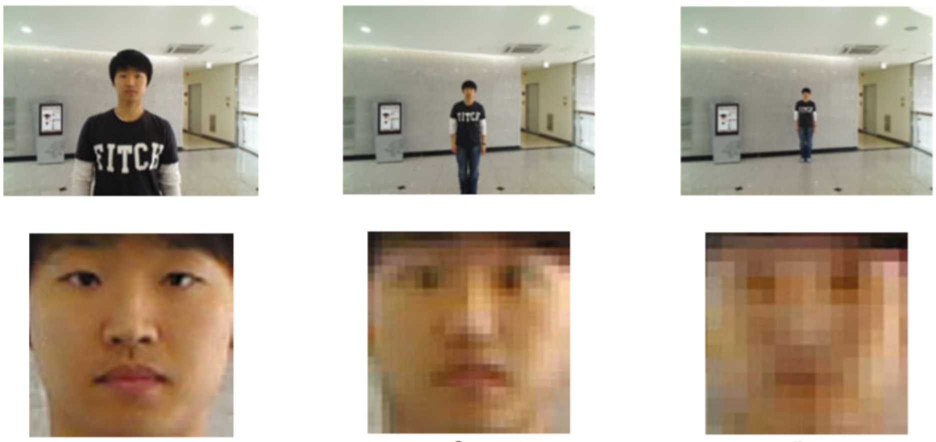
Pan-Tilt-Zoom camera

Long range infrared camera

Why such sensors?
To address:

- spatial resolution
- variations in illumination
- self-occlusion (frontal, profile)
- the pose variation problem

640 × 480 >>>



1m IPD=35

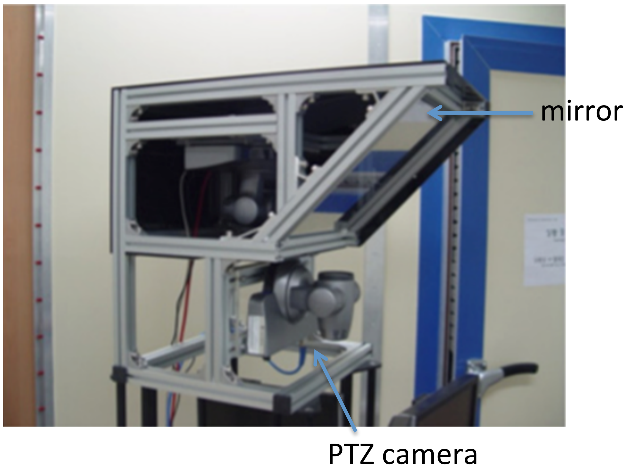
3m IPD=12

5m IPD=7

Surveillance System Setup

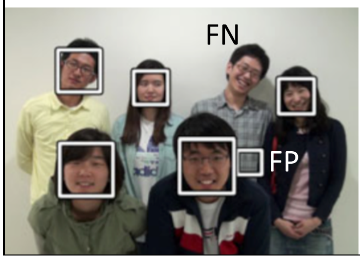
Detect human

Turn the camera and zoom in on the face



Face Detection

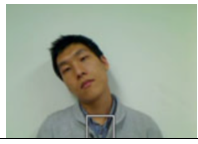
- State-of-the-art face detection methods are typically based on extracting **local texture features**
- This approach follows the seminal work done by **Viola and Jones**



… but is not perfect =>

cluttered background

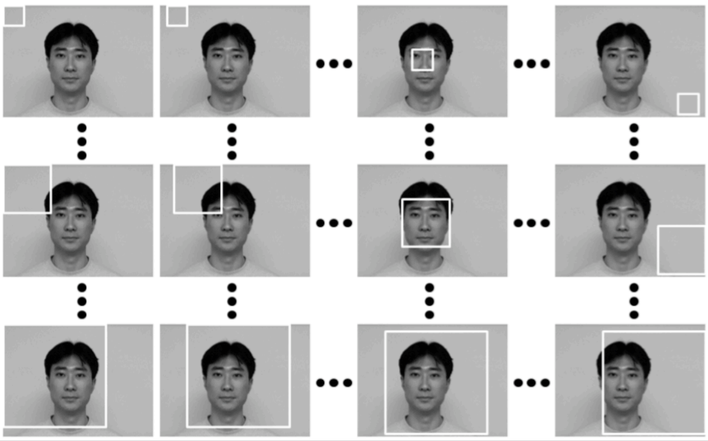
tilted face



inverted face

Viola-Jones Face Detector

Scans through the input image with detection **windows** of different sizes and decides whether each window contains a face or not



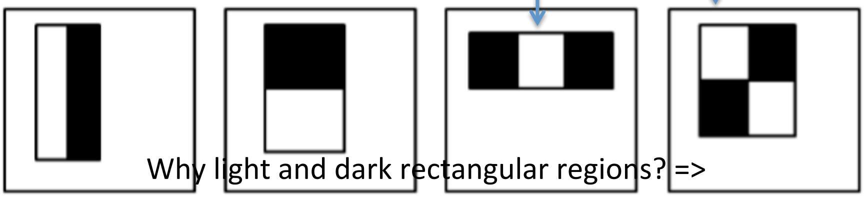
Viola-Jones Face Detector & **Filters**

In each window the existence of a face candidate is decided by applying a classifier to simple **local features** derived using **rectangular filters** (Haar-like)

detection window

… i.e. **two**-rectangle, **three**-rectangle and **four**-rectangle filters >>>

When the **combination** of filter responses (features) in a certain window exceeds a threshold, a face is said to have been detected

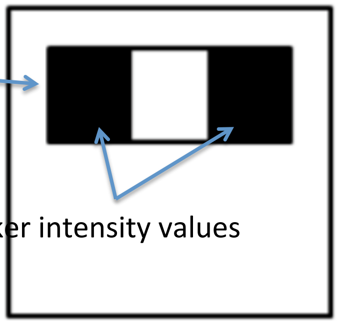


Why light and dark rectangular regions? =>

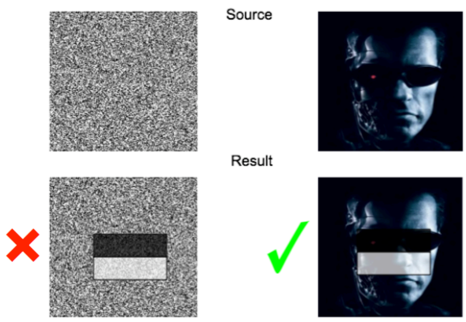
Viola-Jones Face Detector & **Feature Values**

Feature values are obtained by computing the **difference between the sum of the pixel intensities** in the light and dark rectangular regions

filter can be used to detect the two eyes and the bridge of the nose



the eyes typically have darker intensity values

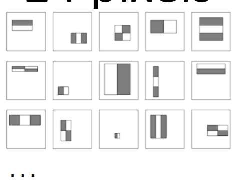


Viola-Jones Face Detector & **Integral Image**

Set of Haar-like filters of **different** sizes need to be applied to each window and the filter responses must be **combined** in an appropriate way in order to detect a face

=> more than **180,000** different Haar-like features can be derived from a detection window of size 24 × 24 pixels

=> to reduce computational burden calculate **integral image**



Input image

1	2	2
2	3	0
1	0	3

⟶

Integral image

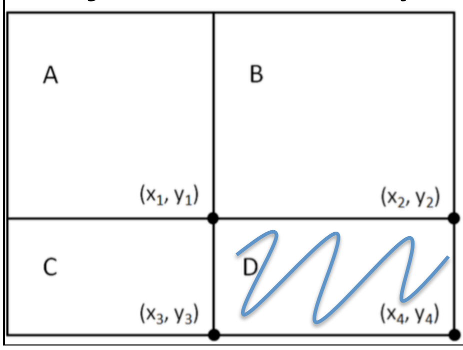
1	3	5
3	8	10
4	9	14

$$S(x,y) = \sum_{1 \leq x' \leq x, 1 \leq y' \leq y} I(x',y')$$

=> the sum of all pixel intensities **above** and to the **left** of (*x*,*y*) in the original image

Viola-Jones Face Detector & Integral Image

Sum of pixel values within any **arbitrary rectangular** region in the original image can be computed based on just **four** array accesses in the integral image =>



$$S(x_4,y_4) - S(x_2,y_2) - S(x_3,y_3) + S(x_1,y_1)$$

Viola-Jones Face Detector & **Adaboost**

Achieving real-time => it is essential to determine a small **subset** of discriminative features from the complete set of features available within each window

=> Variant of the **Adaboost** algorithm can be used to **select the discriminative features** as well as to train the **classifier** function (integrated)

=> Classifier function combines the feature values using appropriate **weights** and the face is found if the combined value is greater than a threshold

VJ FD & Adaboost

Construct a **weak classifier** that predicts whether the window *w* is a face image

sign +/-

feature value obtained using filter *j* in window *w*

$$h_j(w) = \begin{cases} 1 & \text{if } p_j f_j(w) \le p_j \theta_j \\ 0 & \text{otherwise} \end{cases}$$

threshold

L is the total number of filters

Given a training set of *n* example images that includes both faces and non-faces, the classifier can be trained =>

VJ FD & Adaboost & **Selection of Features**

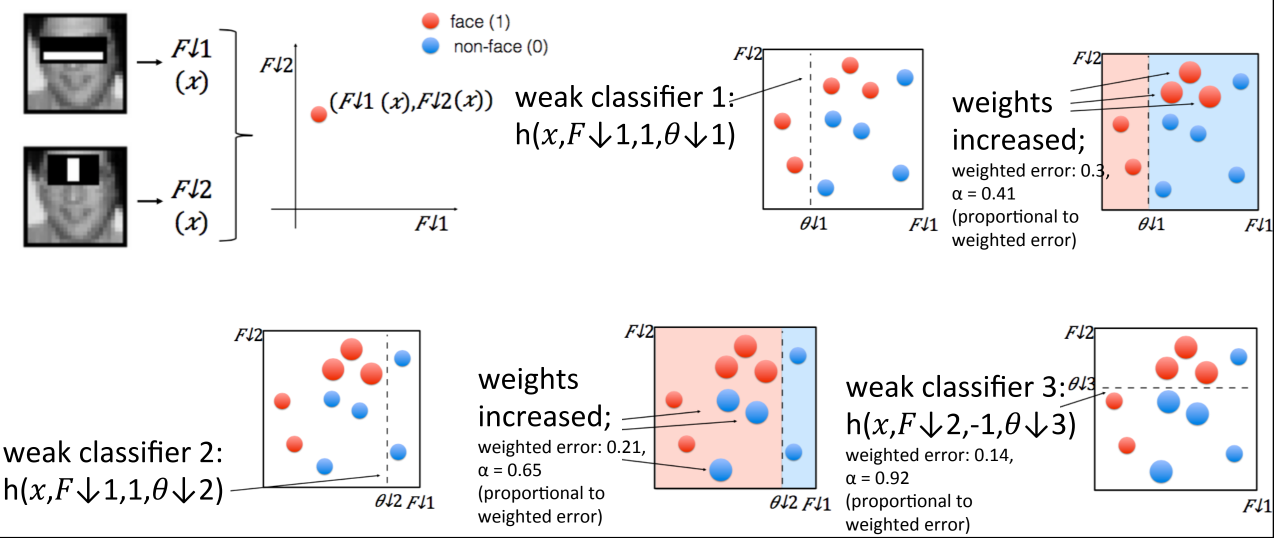
- Boosting is a **supervised learning method** that combines *weak learners* into a more accurate classifier
- A weak learner is a classifier that only needs to exceed **chance** performance
- Training consists of multiple boosting rounds
  - During each round, select a weak learner that does well on examples that were **hard** for the weak learners in previous rounds
  - "Hardness" is captured by **weights** attached to training examples

=>

VJ FD & Adaboost & **Boosting Illustration**

vertical or horizontal half-planes

weak classifier: *h*(window *x*, feature value *F*, sign/polarity *p*, threshold θ)



Given n example image windows $(w_1, y_1), (w_2, y_2), \cdots, (w_n, y_n)$, where $y_i = 0, 1$ for non-face and face examples — training data

Let $l = \sum_{i=1}^{n} y_i$ be the number of face examples and $m = (n - l)$ be the number of non-face examples. Initialize weights $\tau_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ for $y_i = 0, 1$ — weights

for $t = 1, 2, \cdots, T$ **do** — T most discriminative features among the L features (T<<L)

Normalize the weights, $\tau_{t,i} \leftarrow \frac{\tau_{t,i}}{\sum_{k=1}^{n} \tau_{t,k}}$

For each weak classifier $h_j(w)$, compute the weighted error $\varepsilon_j \leftarrow \sum_{i=1}^{n} \tau_{t,i} |h_j(w_i) - y_i|$ — find weak classifier that achieves the lowest weighted training error

Select the best weak classifier with respect to the weighted error, $h_t(w) \leftarrow h_q(w)$ and $\varepsilon_t \leftarrow \varepsilon_q$, where $q \leftarrow \arg\min_j \varepsilon_j$

Update the weights: $\tau_{t+1,i} = \tau_{t,i} \beta_t^{1-e_i}$, where $e_i = 0$ if example w_i is classified correctly, $e_i = 1$ otherwise, and $\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$ — raise the weights of training examples misclassified by current weak classifier

end — $\Rightarrow$ calculate final strong classifier

VJ FD & Adaboost & **Strong Classifier**

Final strong classifier is a **weighted linear combination** of the *T* best weak classifiers based on the selected features …

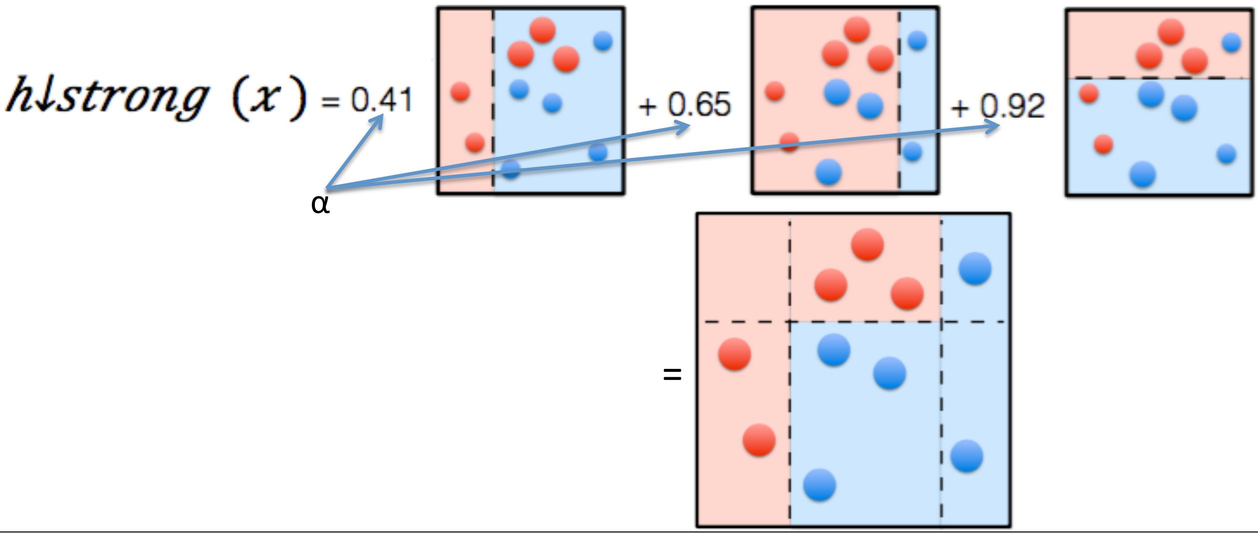
where the weights are proportional to the discriminative power of the features

(inversely proportional to the training **error** rate)

$$H(w) = \begin{cases} 1 & \text{if } \sum_{t=1}^{T} \alpha_t h_t(w) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise,} \end{cases}$$

$$\alpha_t = \log \frac{1}{\beta_t}$$

VJ FD & Adaboost & **Strong Classifier Boosting Illustration**



VJ FD & Adaboost **Detection Stage**

- During the actual face detection stage **only the *T* feature values** need to be computed for each window and the outputs of the weak classifiers based on these features

+ Enables **rapid** categorization

− Original image must still be scanned with **windows** of different sizes to determine the location of the faces

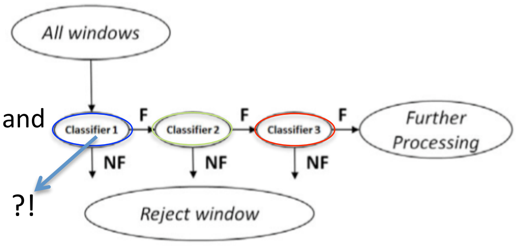
=> speed-up?

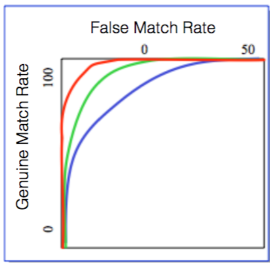
VJ FD & Adaboost & **Cascades**

Classifiers with a simple & smaller set of filters can be first used to screen out a **large** number of **non-face** images in the early stage …
and then **more powerful classifiers** can be used in the later stage

What do we need to reach a GMR of 0.9 and an FMR ≈ 10^{-6}?

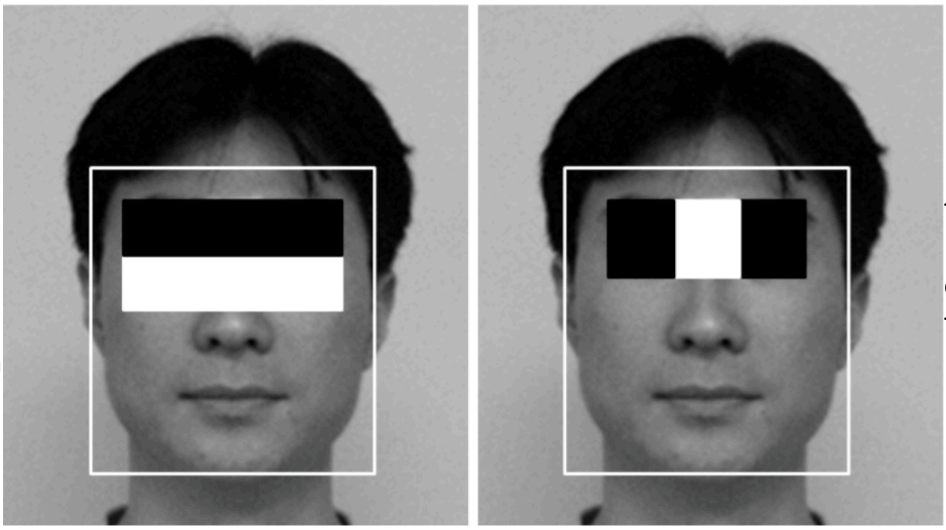
- Assume each stage has a GMR of 0.99 and an FMR of about 0.30
- Need 10 stages: ($0.99^{10} \approx 0.9$) and ($0.3^{10} \approx 6 \times 10^{-6}$)





VJ FD & **2 most effective Haar-like features**

eye region is typically darker than the cheek region

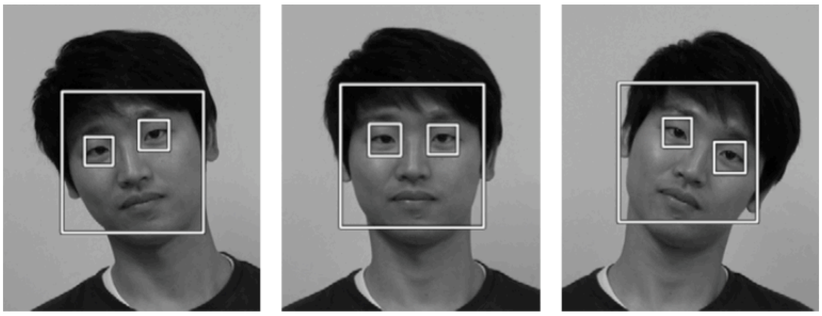


the brighter nose bridge compared to the eye regions

able to reject about 60% of non-faces

Can We Use It for Detection of Eyes,…?

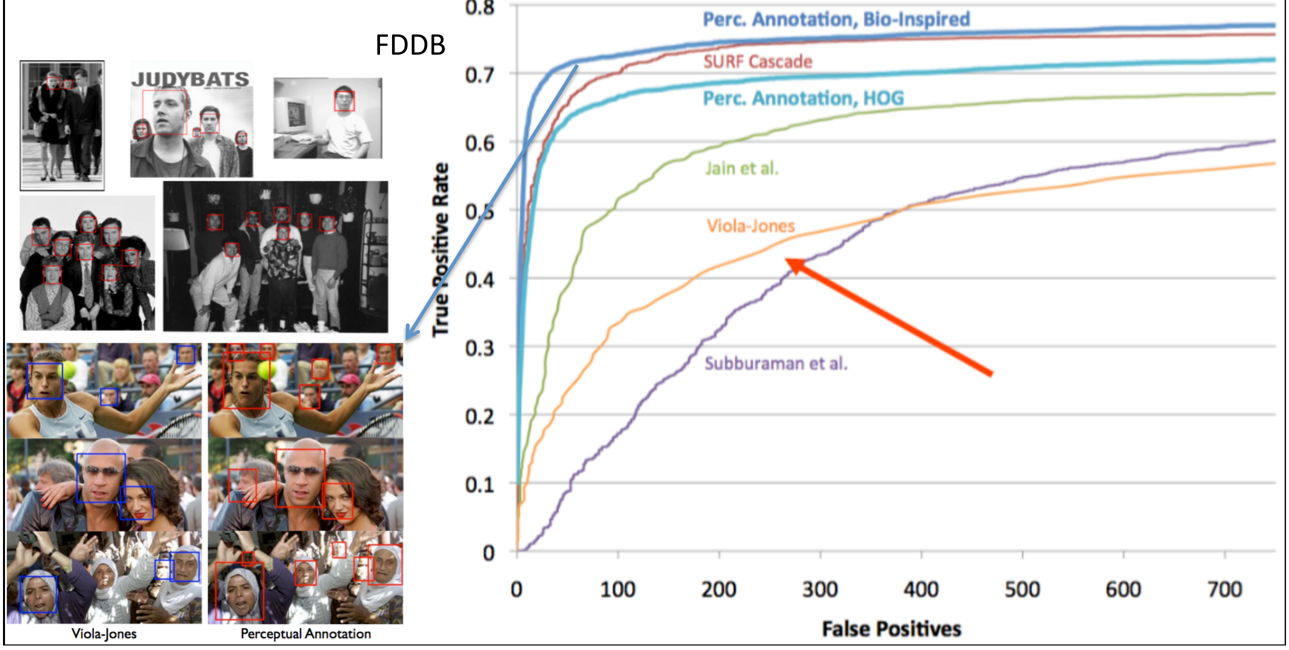
Eye detector can be constructed by simply replacing the training set with examples of eye images and non-eye images



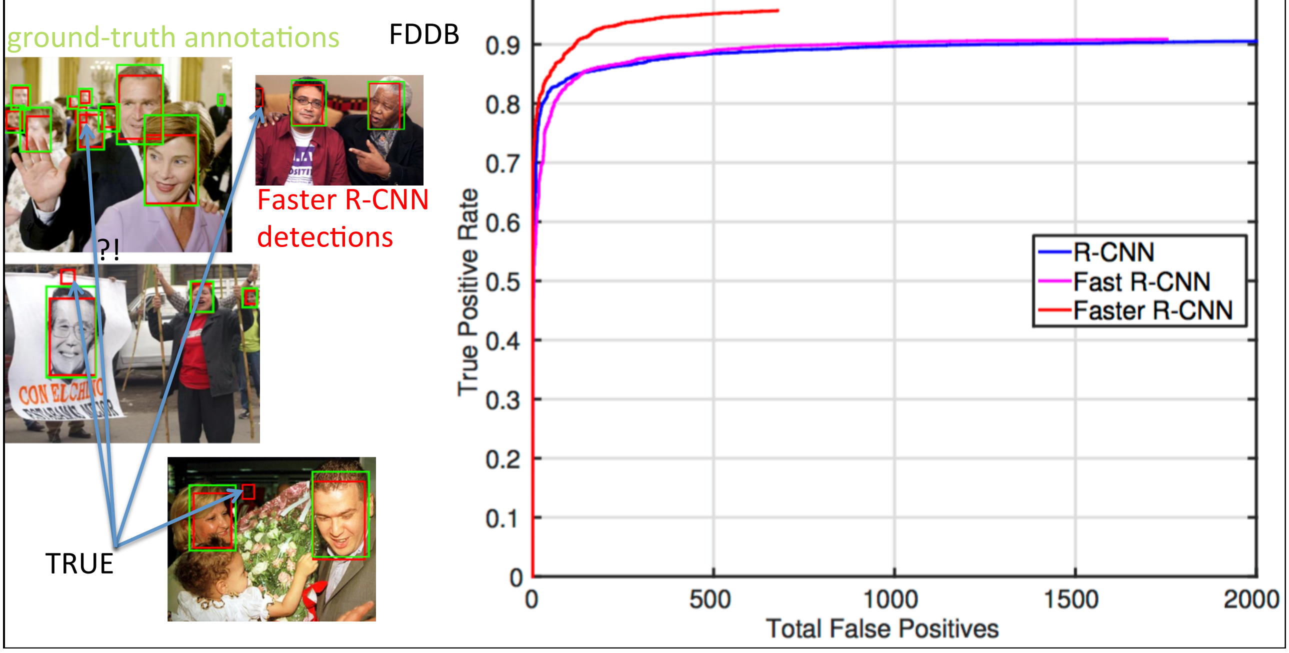
VJ FD & **Practical Considerations**

- VJ FD has a number of open **parameters**, which affect the speed and accuracy
- Factors to consider:
  - Number of overlapping detections
  - Scale factor
  - Minimum and maximum size of faces
  - Number of faces
  - Step size
  - …

FD Evaluation



FD with CNNs



Open Research Questions?

Face detection is not solved

=> Occlusion

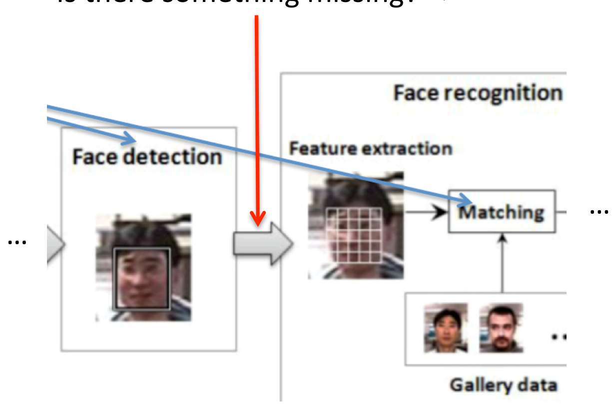
=> Pose

=> Low resolution faces

What Comes Next?

NORMALIZATION:
Pose (landmarking, **alignment**)
Illumination
…

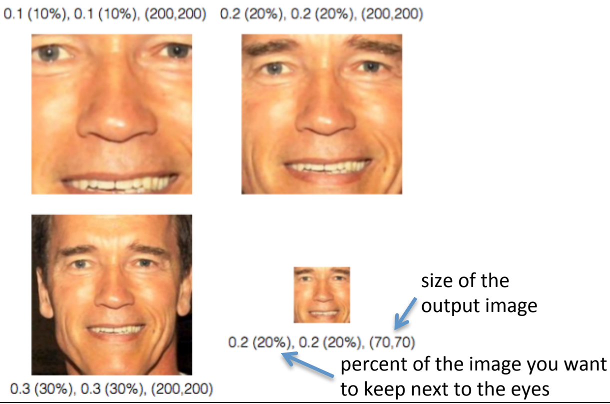
Is there something missing? =>



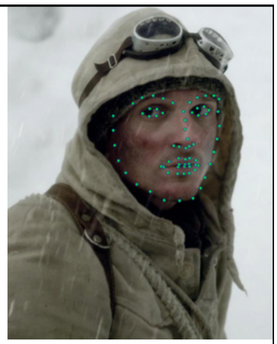
Alignment by Geometric **Warping**

- Input: eyes positions
  & output parameters: horizontal / vertical offset, size
- Output:
  scaled, rotated and
  cropped image
- Problem:
  only *in-plane*
  rotation addressed

basic idea: rotate
the face around
one eye



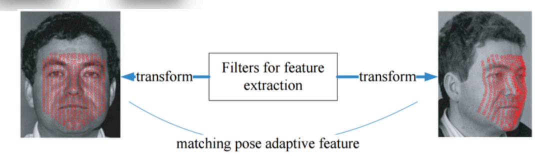
Facial **Landmark Detection**



Find location of facial landmarks to >>>

Align

Use with pose adaptive feature extractors



Morph and synthesize

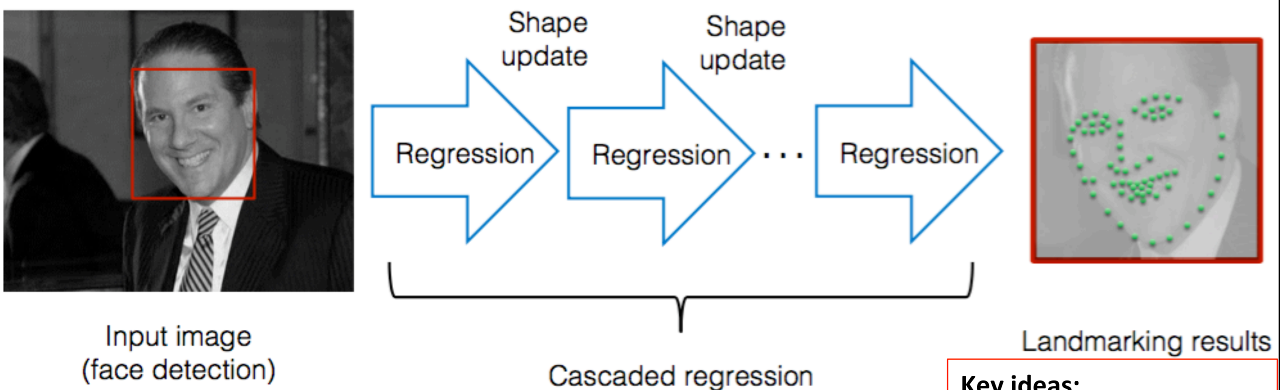
…

Landmark Detection

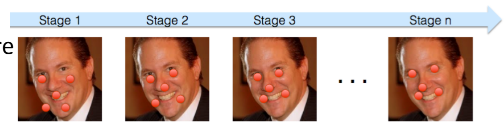
- Input: facial image & face position (detection)
- Output: landmarks (fiducial points)
- Approaches:
  - Active appearance models
  - Constrained local models
  - Parts-based methods
  - **Cascaded regression**
  - …

LD Using **Cascaded Regression**

Framework =>



At each stage calculate feature vector from current shape and perform regression



Key ideas:
Learn regression model**s**
Refine landmark positions by cascading regression models

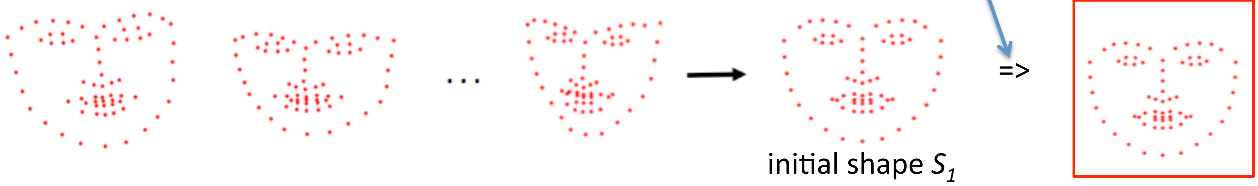
LD & **Training**

- Training data: sample images with annotated landmarks

- Output: regression model (and feature extractor) for **each** stage in cascade

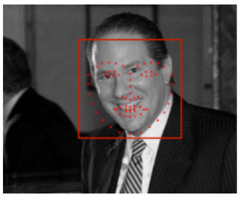
LD & **Training Procedure**

1. Compute initial shape and place it into the image (face detector output)
   - Initial shape is simple **average** of reference annotations



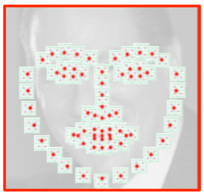
LD & Training Procedure

2. For **stage** t in cascade
   a) Compute feature vector x_i at current shape vertices S_t on image I_i



input image I_i

=>



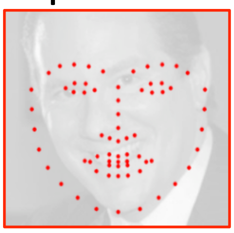
$x_i = F(I_i, S_t)$

Typically used features:
HoG
Pixel differences
Pixel comparisons
...

   b) Repeat for all n training input images and construct regressor matrix: $X_t=[x_1,\ldots,x_i,\ldots,x_n]$

LD & Training Procedure

2. For stage t in cascade
   c) Compute required displacement (shape increment) ΔS



current shape S_t

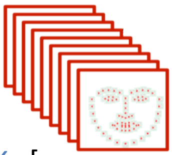
target/reference shape S_r

$\Delta S = S_r - S_t$

   d) Repeat for all n training images and construct regression output matrix: $\Delta_t=[\Delta S_1,\ldots, \Delta S_i,\ldots, \Delta S_n]$

LD & Training Procedure

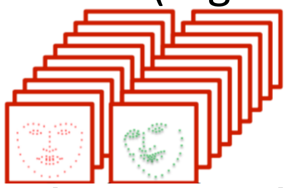
2. For **stage** *t* in cascade
   e) Compute regression model *R* from features (regressor matrix) to displacement (regression output matrix)



$X_t=[x_1,\dots,x_i,\dots,x_n]$

R

=>



$\Delta_t=[\Delta S_1,\dots, \Delta S_i,\dots, \Delta S_n]$

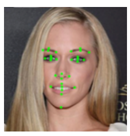
Typically used regression approaches:
Least Squares Regression
Regression Trees
Ridge Regression
Gaussian Process Regression
...

   f) Apply regression model & update current shape estimate: $\Delta_t^* = R(X_t)$ => $S_{t+1} = S_t + \Delta_t^*$

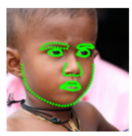
Illustration of the Regression

initial estimate & shape estimates at different stages

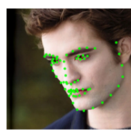
BTW:
different databases,
different number of landmarks



LFPW (29 landmarks)



HELEN (194 landmarks)

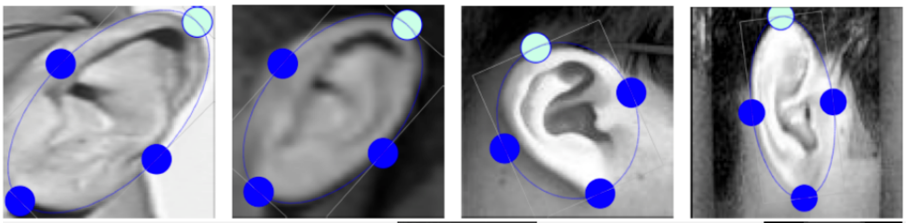


300-W (68 landmarks)

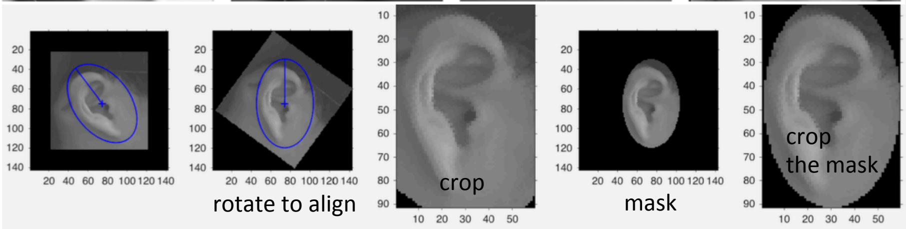
Can We Use CR on Other Modalities?

Cascade Pose Regression on ears:

elliptical model
... on training data



evaluate position and direction of the model



Cascade Pose Regression on Ears

descriptor	unaligned		aligned+cropped		+masked	
	Rank–1 (%)	EER (%)	Rank–1 (%)	EER (%)	Rank–1 (%)	EER (%)
LBP	83.1 ± 7.9	12.9 ± 4.3	94.4 ± 2.1	8.2 ± 2.8	91.9 ± 4.1	9.4 ± 4.3
BSIF	87.8 ± 5.4	11.9 ± 4.9	97.5 ± 1.6	6.0 ± 2.4	85.3 ± 5.4	14.1 ± 5.1
LPQ	84.7 ± 6.9	14.7 ± 4.6	95.0 ± 1.9	7.5 ± 2.8	61.5 ± 6.8	34.8 ± 7.2
RILPQ	86.2 ± 4.7	13.2 ± 4.1	94.7 ± 2.2	6.3 ± 1.9	90.0 ± 3.7	10.2 ± 5.0
POEM	85.0 ± 4.6	13.1 ± 4.1	94.4 ± 2.4	7.1 ± 2.4	92.2 ± 2.6	8.8 ± 3.9
HOG	70.3 ± 7.2	20.7 ± 3.8	84.6 ± 3.3	12.6 ± 5.0	84.7 ± 2.8	14.0 ± 5.6
DSIFT	65.5 ± 9.5	21.4 ± 6.0	83.1 ± 4.4	14.8 ± 5.9	79.1 ± 6.3	16.3 ± 4.8
Gabor	73.7 ± 8.3	19.1 ± 5.9	92.8 ± 3.8	10.7 ± 3.1	87.8 ± 2.9	11.6 ± 3.3

IITD I db –
in-plain images!

	Rank–1 (%)	EER (%)	Rank–1 (%)	EER (%)	Rank–1 (%)	EER (%)
LBP	52.2 ± 10.8	30.4 ± 7.7	46.9 ± 11.0	35.7 ± 6.5	49.9 ± 8.5	32.7 ± 6.6
BSIF	52.2 ± 13.8	30.2 ± 6.7	41.53 ± 8.2	37.9 ± 9.01	34.8 ± 6.1	40.2 ± 8.4
LPQ	48.6 ± 12.6	30.9 ± 6.1	18.4 ± 7.8	39.8 ± 6.2	23.2 ± 6.7	52.0 ± 6.7
RILPQ	51.3 ± 14.9	33.5 ± 9.3	43.3 ± 12.0	32.2 ± 4.2	35.7 ± 4.9	37.7 ± 5.3
POEM	56.6 ± 12.0	28.0 ± 5.9	49.5 ± 11.5	32.2 ± 7.6	48.6 ± 9.2	32.0 ± 4.9
HOG	50.9 ± 6.4	30.0 ± 9.7	48.6 ± 15.7	30.6 ± 8.1	43.3 ± 4.2	32.5 ± 7.1
DSIFT	52.2 ± 10.2	29.9 ± 7.2	49.0 ± 13.3	37.1 ± 11.6	45.9 ± 9.2	32.6 ± 7.7
Gabor	46.4 ± 6.4	30.3 ± 5.1	33.6 ± 7.7	36.5 ± 8.8	37.0 ± 9.4	36.8 ± 9.8

AWE db –
out-of-plain images!!!

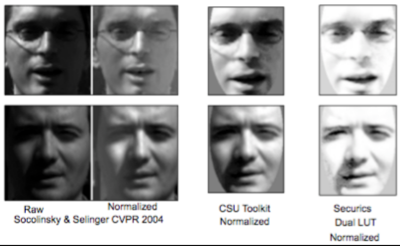
Producing info where there is non-given ... is just not the way to go!

Lighting Normalization

How well will these faces match?

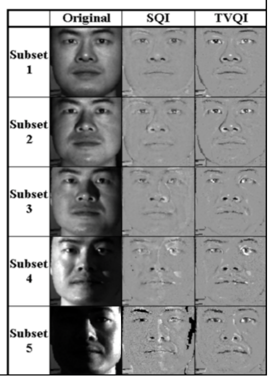
- **Active methods:**
  solve the problem at acquisition
  – Thermal infrared Images
  – Near-infrared Images
  – 3D information acquisition

- **Passive methods:**
  analysis of the acquired images =>



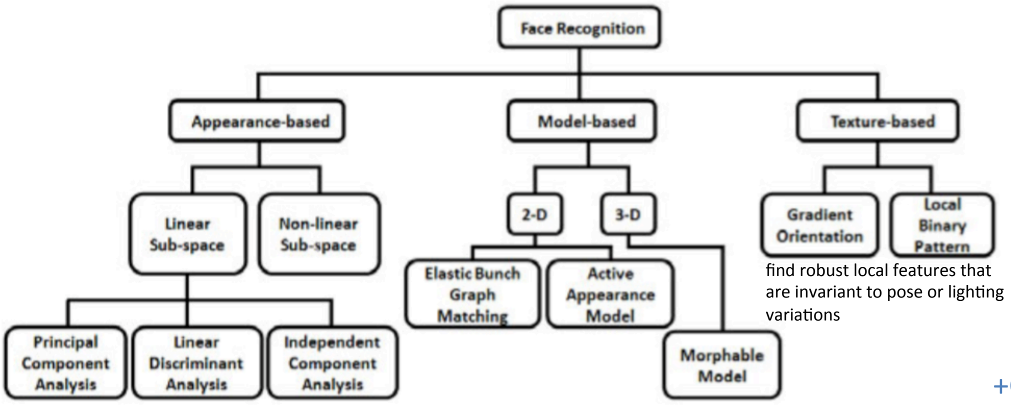
Passive Illumination Invariance Methods

- **Photometric normalization:**
  Histogram normalization, Gamma intensity correction, Local normalization,...
- **Illumination variation modeling:**
  Linear subspaces, Illumination cone, Generalized photometric stereo,...
- **Illumination invariant features:**
  Direction of gradient, Shape from shading, **Q**uotient **i**mage, Eigen-phase, Local bin. pattern,...



Feature Extraction and Matching

Main approaches to match the detected face images:



find robust local features that are invariant to pose or lighting variations

+CNN ;-)

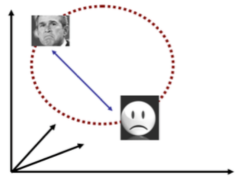
- generate a compact representation of the entire face region
- sub-space is defined by a set of representative basis vectors, which are learned using a training set of images

facilitate matching of face images in the presence of pose variations

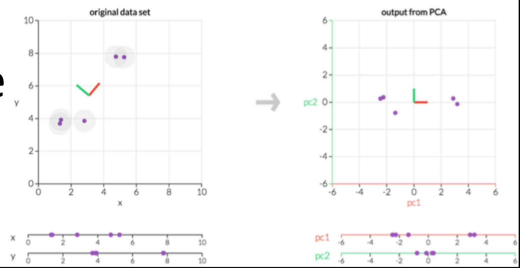
Appearance-based Face Recognition

- **Idea:** the pixel value at location (*x*,*y*) in a face image can be expressed as a weighted sum of pixel values in **all** the training images at (*x*,*y*)
- **Goal:** in linear subspace analysis is to find a **small** set of most representative basis faces
- Images can be **matched** by directly comparing their vector of weights

Subspace Methods



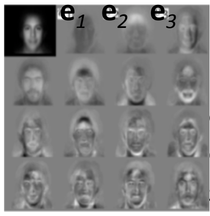
- An image is a point in high dimensional space -> an $N \times M$ image is a point in $\mathbf{R}^{N \times M}$
- **Problem:** faces in the set of all possible images are highly correlated
- **Solution:** compress them to a low-dimensional subspace that captures key appearance features
- How do we do this? >>>

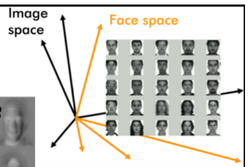


Principal Component Analysis

- Uses the training data to learn a subspace that accounts for as much **variability** in the training data as possible
- By performing an **Eigen** value decomposition of the **covariance** matrix of the data
- Algorithm >>>

1. $\mathbf{x}_1,\mathbf{x}_2,\cdots,\mathbf{x}_N$ is the training set & $\mathbf{x}_i$ represents a d-dimensional column vector;
average of the training is:
$$\boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$



2. Define the data matrix $\mathbf{X} = [(\mathbf{x}_1 - \boldsymbol{\mu})\ (\mathbf{x}_2 - \boldsymbol{\mu})\ \cdots\ (\mathbf{x}_N - \boldsymbol{\mu})]$

each vector is a direction in face space

3. Calculate the data covariance matrix $\mathbf{C} = \mathbf{X}\mathbf{X}^T$

4. Compute the Eigen vectors in $\mathbf{E}$ of the covariance matrix $\mathbf{C}$: $\mathbf{CE} = \lambda\mathbf{E}$, $\mathbf{E} = [\mathbf{e}_1,\mathbf{e}_2,\cdots,\mathbf{e}_d]$

5. Any data vector $\mathbf{x}$ can be represented as a weighted sum of the Eigen vectors
Weights / Eigen coefficients can be computed as $\boldsymbol{\omega}=\mathbf{E}^T\mathbf{x}$
($\boldsymbol{\omega} = [\omega_1,\omega_2,\cdots,\omega_d]^T$ is a d-dimensional column vector, where ω_j is the weight associated with Eigen vector $\mathbf{e}_j$, for $j = 1,2,\cdots,d$)

Matching:
compute Eigen coefficients $\boldsymbol{\omega}^G$ and $\boldsymbol{\omega}^P$ corresponding to the gallery and probe face images >>> Euclidean distance between the two Eigen coefficients can be considered as a measure of dissimilarity between the two face images

Any problems?

Projecting into the face space (using Eigenfaces):

$\mathbf{x}$ $\omega_1\mathbf{e}_1$ $\mathbf{x}\approx\boldsymbol{\mu}+\omega_1\mathbf{e}_1+\ldots$

Dimensionality

- Dimensionality of the Eigen coefficients ($\boldsymbol{\omega}$) is the same as the dimensionality of the original data $\mathbf{x}$, which is d
- Goal of linear subspace analysis is to reduce the dimensionality
- **Solution:** consider a lower dimensional subspace $\mathbf{E}'$, which is spanned by only d' ($d' < d$) Eigen vectors from $\mathbf{E}$ corresponding to the d' **largest** Eigen values
  – Eigen vector corresponding to the largest Eigen value in PCA accounts for maximum variability in the data and is called the **principal axis**
  – It is possible to account for most of the variability in the data by selecting **only a few** Eigen vectors corresponding to the largest Eigen values in the descending order

ORL db & seven largest Eigen values (Eigenfaces) >

What if $N < d$?

- There will be only $N-1$ meaningful Eigen vectors and the remaining Eigen vectors will have associated Eigen values of zero
- Can we speed up the calculation of these $N-1$ Eigen vectors?

$N \times N$ matrix

Calculate $\mathbf{C}^* = \mathbf{X}^T\mathbf{X}$

Compute the Eigen vectors in $\mathbf{E}^*$

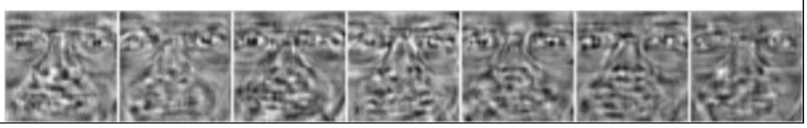
Matrix of meaningful Eigen vectors of $\mathbf{C}$: $\mathbf{V} = \mathbf{X}\mathbf{E}^*$

Eigen coefficients corresponding to a data vector $\mathbf{x}$: $\boldsymbol{\omega} = \mathbf{V}^T\mathbf{x}$

Generalized Version of PCA is … **ICA**

- Independent Component Analysis
- PCA constrains the Eigen vectors to be **orthogonal** to each other and hence, the resulting Eigen coefficients are **uncorrelated**
- However, the Eigen coefficients need **not** be independent =>
- **ICA attempts to find a linear transformation that minimizes the statistical dependence between its components**
  - ICA coefficients are independent or "the most independent possible"
  - Moreover, unlike PCA, there is no relative order between the ICA coefficients

ORL db & seven ICA components >

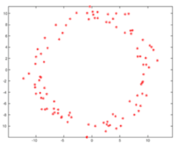


Reconstruction and Errors

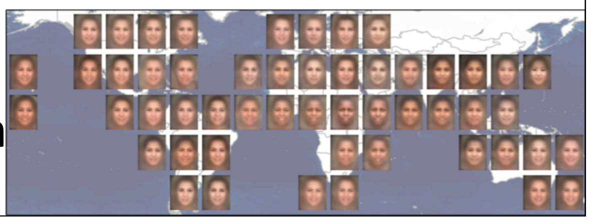
- Only selecting the **top** *P* Eigenfaces reduces the dimensionality
- **Fewer** eigenfaces = more information loss and less discrimination

Other Problems?

- Global appearance method: **not** robust
- PCA assumes that the data has a **Gaussian** distribution (mean, covariance matrix)
  >>> data set can not always be well described by its principal components



- PCA projection is optimal for reconstruction from a low-dimensional basis,
  BUT **PCA may not be optimal for discrimination**



Let's Optimize Something Else …

- **Eigenfaces** attempt to maximize the scatter of the training images in faces space
- Why not try to …
- **Fisherfaces** attempt to maximize the between class scatter, while minimizing the within class scatter
- Method name? >>>

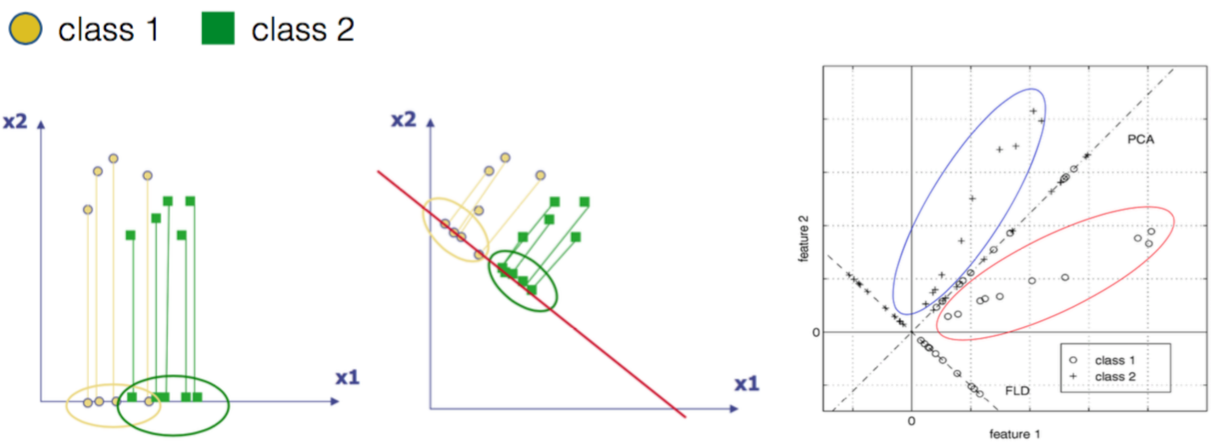
Linear Discriminant Analysis

- PCA is unsupervised learning method – class label (user ID information) is never used during the learning of the basis faces
- Accuracy based on PCA cannot be expected to be very high

- LDA explicitly **uses the class** label of the training data and conducts subspace analysis with the objective of **minimizing** intra-class variations and **maximizing** inter-class variations

Illustration of the Projection

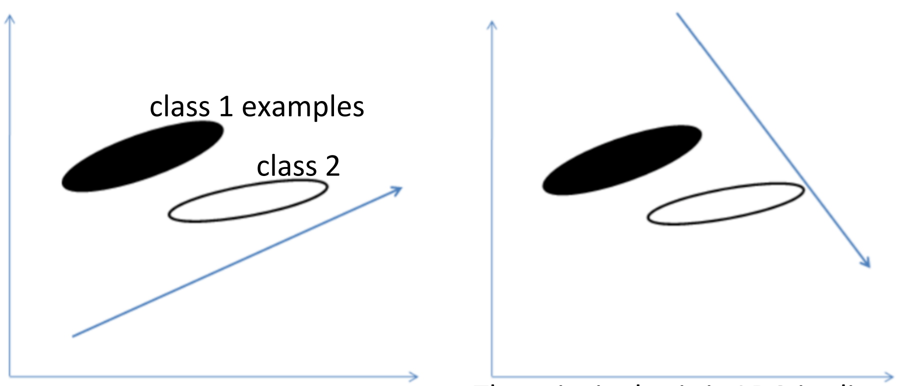
Separability?



Bad Projection

Good Projection

Illustration of the Projection



The principal axis in **PCA** is aligned such that when the **data** is projected onto this axis, the **variance is maximized**

The principal axis in **LDA** is aligned such that when the data is projected onto this axis, the **variance** within each **class** is **minimized** and the **separability** between the two **classes** is **maximized**

Algorithm >>>

1. $(\mathbf{x}_1,y_1),(\mathbf{x}_2,y_2),\ldots,(\mathbf{x}_N,y_N)$ be the training set,
   $\mathbf{x}_i$ represents a *d*-dimensional column vector,
   $y_i \in \{1,2,\cdots,c\}$ (=*j*) is the corresponding class label, *c* is the number of classes
   Compute the mean of each class: $\boldsymbol{\mu}_j = \frac{1}{N_j}\sum_{y_i=j}\mathbf{x}_i$ $\quad \mu = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$

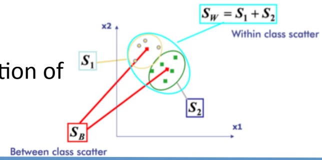
   Scatter of class *j*: S_j

2. Define the within and between class scatter matrixes: $\mathbf{S}_w = \sum_{i=1}^{c}\underbrace{\sum_{y_i=j}(\mathbf{x}_i-\boldsymbol{\mu}_j)(\mathbf{x}_i-\boldsymbol{\mu}_j)^T}_{S_j}$

$$\mathbf{S}_b = \sum_{j=1}^{c} N_j(\boldsymbol{\mu}_j-\boldsymbol{\mu})(\boldsymbol{\mu}_i-\boldsymbol{\mu})^T$$

3. Minimize $\mathbf{S}_w$ and maximize $\mathbf{S}_b$ simultaneously
   by maximizing $\mathbf{S}_w^{-1}\mathbf{S}_b$ >>> solve Eigen system:
   $\mathbf{S}_w^{-1}\mathbf{S}_b\,\mathbf{E} = \lambda\mathbf{E}$
4. Any data vector $\mathbf{x}$ can be represented as a weighted sum of the Eigen vectors
   Weights / Eigen coefficients are computed as $\boldsymbol{\omega}=\mathbf{E}^T\mathbf{x}$

illustration of scatter



Matching:
See PCA ;-)

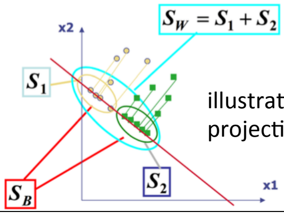
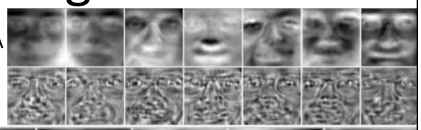


illustration of projection

What if *N* < *d*?

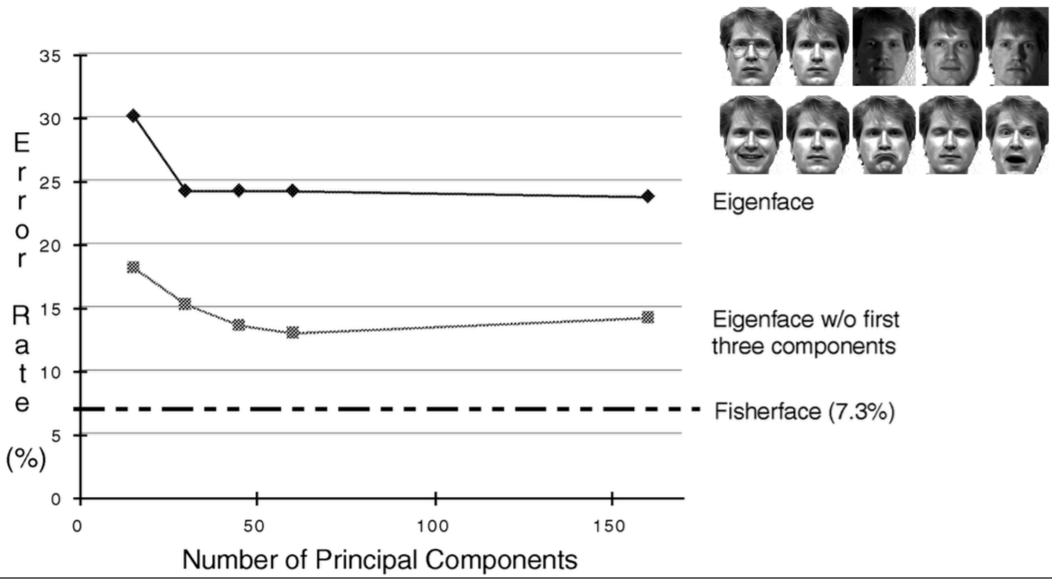
- $\mathbf{S}_w$ often becomes singular
- Solution? >>>
- First apply **PCA** to the training samples to reduce the data dimensionality and then apply **LDA** to the transformed data having lower dimensionality

PCA & ICA



ORL db & seven largest Eigen values (Fisherfaces) >

Eigenfaces vs. Fisherfaces

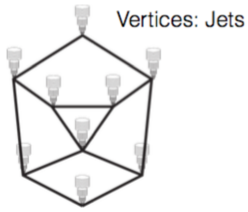


Model-based face recognition

- Try to derive a **pose**-independent representation
- Require the detection of several fiducial/ landmark points
- Increased complexity compared to appearance-based techniques

Elastic Bunch Graph Matching – Face Bunch Graph

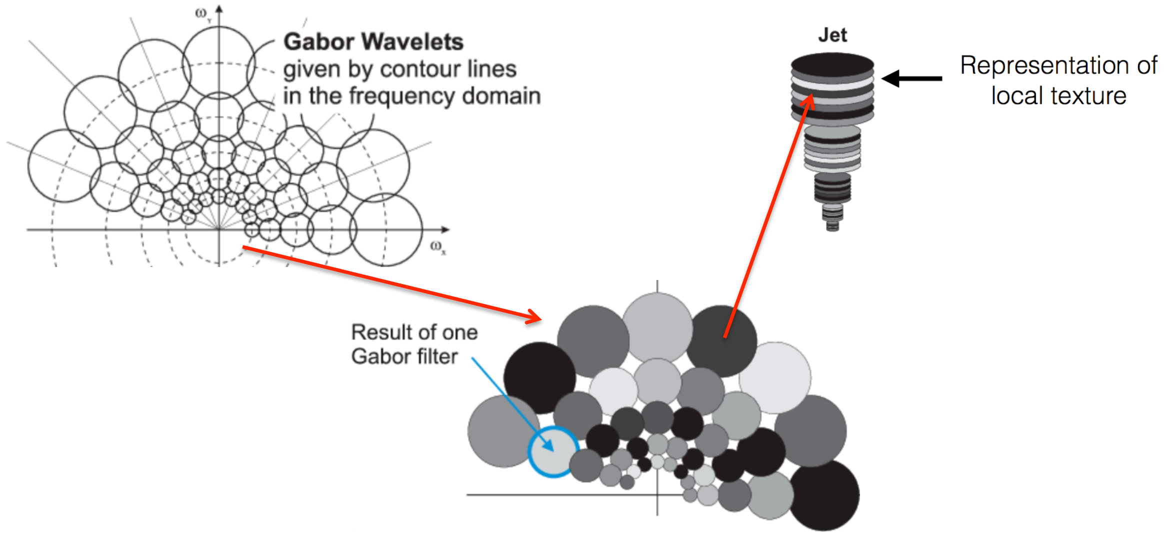
- It represents a face as a **labeled** image **graph** with each **node** being a fiducial point
- Node of the graph is labeled with a set of **Gabor** coefficients (**jet**)
- Jet **characterizes** the local texture information
  - The Gabor coefficient(!) at a location in the image is obtained by **convolving** the image with a complex 2D Gabor filter centered at that location
  - By **varying** the orientation and frequency of the Gabor filter, a set of coefficients or a Gabor **jet** is obtained



- **Connection** of any two nodes of the graph is labeled based on the average distance between the corresponding fiducial points

Jet

Transformation at **one** image location >>>



Construction of **FBG Model** in 2 Stages

From a training set of face images with a **specific** pose (*n* poses)

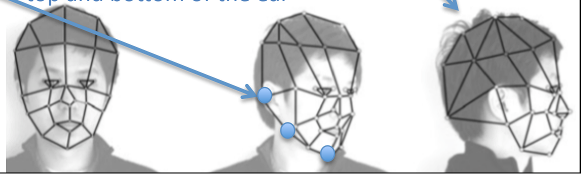
STAGE 1:

a) Manually mark the desired fiducial points and define the geometric structure of the **image graph** for one (or a few) initial image(s)

b) Obtained the rest semi-automatically, by comparing the new images to **model graphs** (images that have been already marked) based on the extracted Gabor jets

chin was not found accurately & top and bottom of the ear

Correspondence between the nodes of bunch graphs belonging to different poses is specified manually



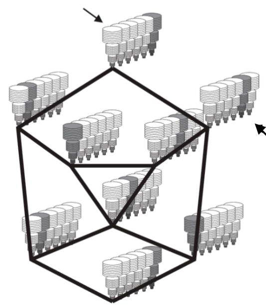
Stage 2

a) Combine a representative set of individual graphs in a stack-like structure

b) Set of jets corresponding to the same fiducial point is called a **bunch**

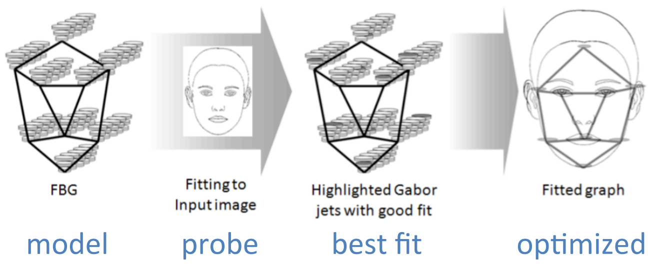
c) Bunch represents **local variations** in the associated fiducial point among the population

– Eye bunch includes jets from open, closed, male, female etc. eyes



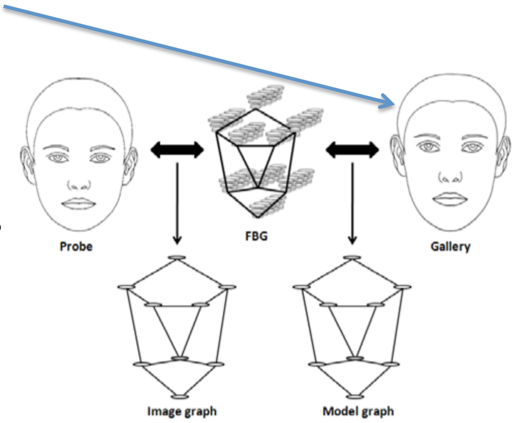
Recognition with FBG Model

1. Execute Elastic Bunch Graph Matching on **probe** image:
   a) Take the **mean** <u>model</u> Gabor jets at each bunch and find the approximate face position at some discrete locations
   b) Refine the position of the face by similarity between a Gabor jet in the given image and a **bunch** of jets in the FBG model
   c) Precisely locate the fiducial points by moving all the nodes locally and relative to each other to **optimize** the graph similarity further

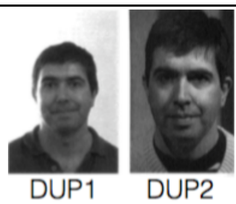


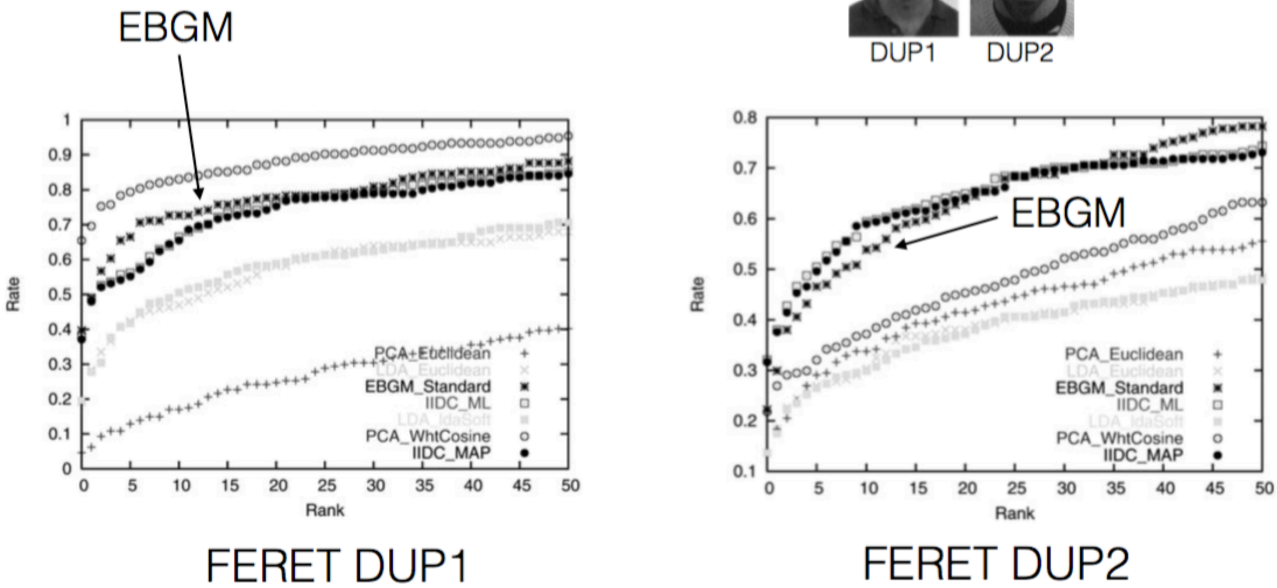
Recognition with FBG Model

2. Comparison with all model graphs
   – Similarity between **image graph** from the probe image and the **model graph** is computed as the **average** similarity between the jets at the corresponding fiducial points
   – Can be matched successfully even with some missing nodes
3. Choose **best similarity score** and apply a **threshold** to determine if it's a valid match



Results





Texture-based Face Recognition

- Raw pixel intensity values are quite sensitive to changes in ambient lighting and facial expressions
- Alternative: use more robust feature representation schemes
- Characterize the texture of an image using the **distribution** of **local** pixel values

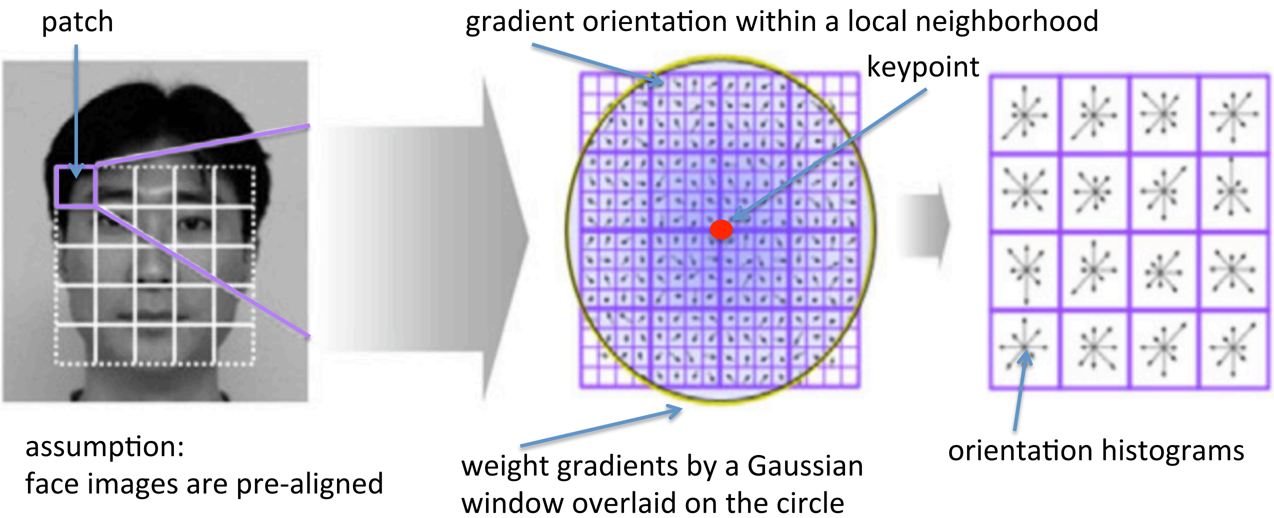
Scale Invariant Feature Transform

- **2 major steps:**
  - key point extraction
  - descriptor calculation in a local neighborhood at each key point
- Achieves tolerance against pose variations
- **Problem:** number of key points in SIFT could be large (in the order of hundreds)
  - Finding the correspondences between the key points from two different images is a challenging task
- **Assumption:** the face images are roughly pre-aligned (e.g. using the eyes)
  - Key point detection process can be bypassed and the descriptor can be constructed directly from the entire face image (Dense SIFT)

Scale Invariant Feature Transform

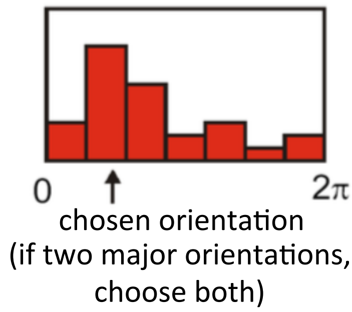
- **Descriptor:** usually a <u>histogram of gradient orientations</u> within a local neighborhood
- Face image is typically divided with multiple **patches** and the SIFT descriptor is constructed from each patch
- Final descriptor is obtained by **concatenating** all the descriptors from all the patches

SIFT Illustration



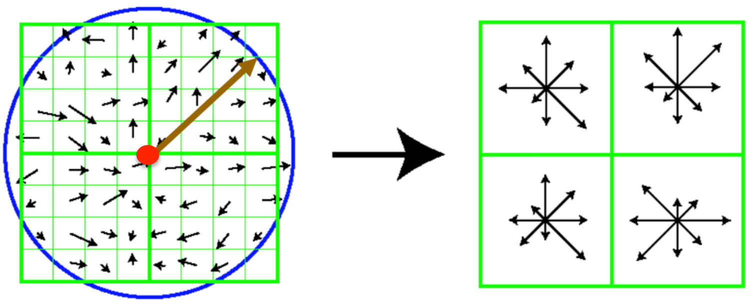
Orientation Assignment

1. Create histogram (36 bins) of local gradients at selected scale
2. Assign canonical orientation (and magnitude) at peak of smoothed histogram



Patch/Keypoint Descriptor

Example: 1 patch = 1 keypoint, 2×2 windows, each sized 4×4 windows, weights



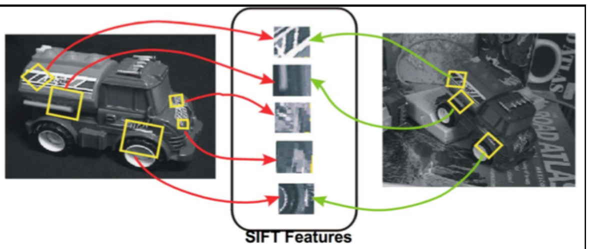
gradients on the image within the patch

orientation histograms for 4 windows (1 patch)

- Orientation histograms have 8 bins (orientations)
- Each patch here is described with 2*2*8=32 features (32D vector)

Normally in practice: 1 patch, **4×4** windows, each sized 4×4 windows, weights

Advantages of SIFT



- **Locality:** features are local
- **Distinctiveness:** features can be matched to a large database
- **Quantity:** many features can be generated for small faces
- **Efficiency:** close to real time performance
- **Extensibility:** can easily be extended to a wide range of feature types

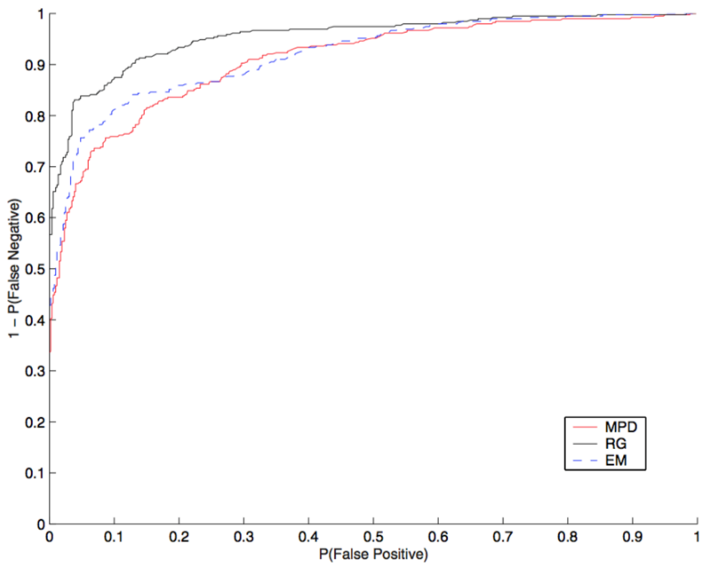
SIFT for Face Recognition

Three SIFT strategies:

1. **Minimum pair distance:** compute distance between all descriptors in each face image
2. **Match eyes and mouth:** compute distance only between descriptors in these regions
3. **Match on a regular grid:** divide face up into a regular grid, match descriptors in subimages

SIFT Results

BANCA db

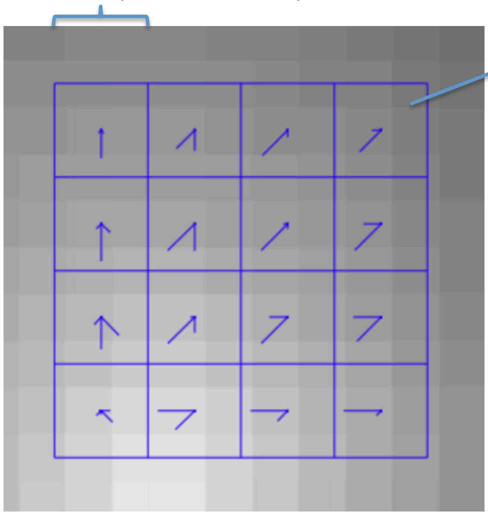


MPD = Min. Pair Distance
EM = Eyes and Mouth
RG = Regular Grid

Histogram of Oriented Gradients

>>> image descriptors invariant to 2D rotation

8 pixels => 8×8 patch => 8×8 HOG descriptor



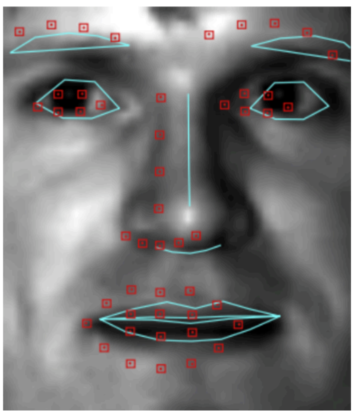
Each cell shows the orientation of the gradients

HOG counts occurrences of edge orientations

Arrange many of these:
in local neighborhoods
or across entire image
(sparse -> dense)

HOG for Face Recognition

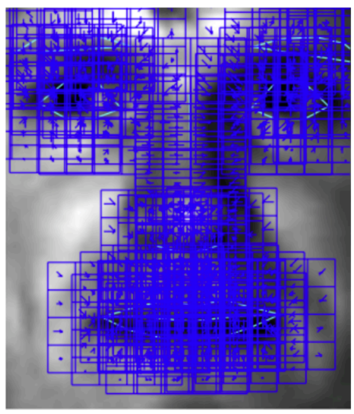
Use of informative local neighborhoods …



initialized landmarks (red boxes) and result for the Active Appearance Model fitting

>>>

24×24 HOG descriptors

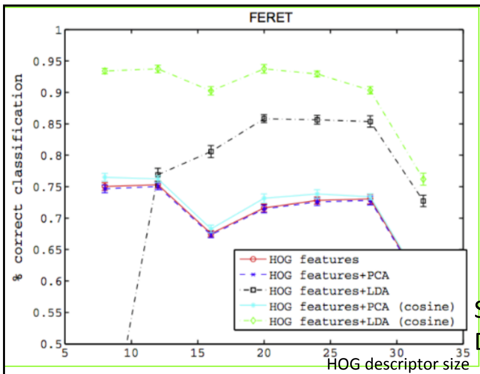


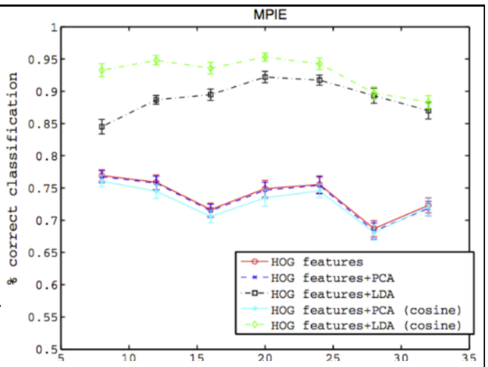
64×64 HOG descriptors

4 databases: FERET, MPIE, AR, Yale =>
only better on FERET than PCA, LDA
(tested robustness to facial feature location)

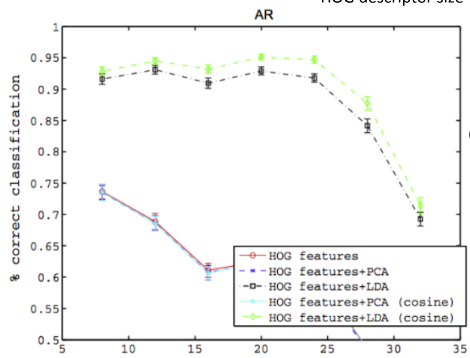
HOG Results

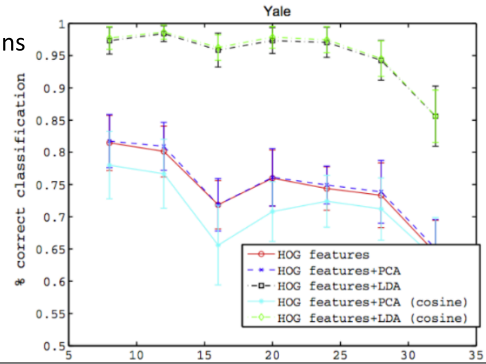
across entire image





- Smaller descriptor sizes better
- Dimensionality reduction doesn't help
- Minimizing intra-class variations and maximizing inter-class variations helps a lot
- Cosine distance better





FERET

		fb (%)	fc (%)	dup1 (%)	dup2 (%)
	PCA Euclidean	74.3	5.6	33.8	14.1
	PCA Mahal. cosine	85.3	65.5	44.3	21.8
	LDA	72.1	41.8	41.3	15.4
	Bayesian	81.7	35.0	50.8	29.9
	Bayesian map	81.7	34.5	51.5	31.2
	Gabor ML	87.3	38.7	42.8	22.7
Hybrid Approach →	HOG-EBGM	**95.5**	81.9	60.1	55.6
	8 × 8 patch	91.4	83.0	70.2	62.0
	12 × 12 patch	93.0	82.0	70.8	63.3
	16 × 16 patch	88.4	68.0	68.7	60.7
	20 × 20 patch	93.7	75.3	70.2	60.3
	24 × 24 patch	94.2	70.1	66.8	56.8
	28 × 28 patch	91.6	42.8	60.0	56.0
Best Approach →	Combination 8 × 8–28 × 28 (product rule)	95.4	**84.0**	**74.6**	**69.2**

decision-level combination of results

Computational Cost

Times in seconds given for all 3540 FERET images

Patch size	From landmarks	From regular grid
8 × 8	203.8	388.3
12 × 12	210.2	200.7
16 × 16	205.6	73.7
20 × 20	211.2	75.1
24 × 24	212.9	75.9
28 × 28	218.2	27.7
Total	1261.9	841.4

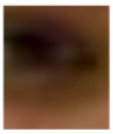
For locating landmarks add about 2.5 s per image

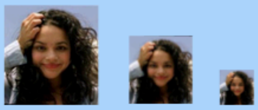
Local Binary Patterns

Motivation:

2-D surface texture is a two-dimensional phenomenon characterized by:

1. Spatial structure (**pattern**)
2. **Contrast** (amount of texture)

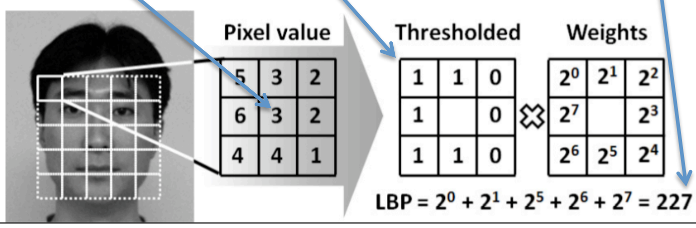


Transformation	Influence?! >>>	Property: Pattern	Property: Contrast
Grayscale 		No effect	Affects
Rotation 		Affects	No effect
Scale 		Affects	?

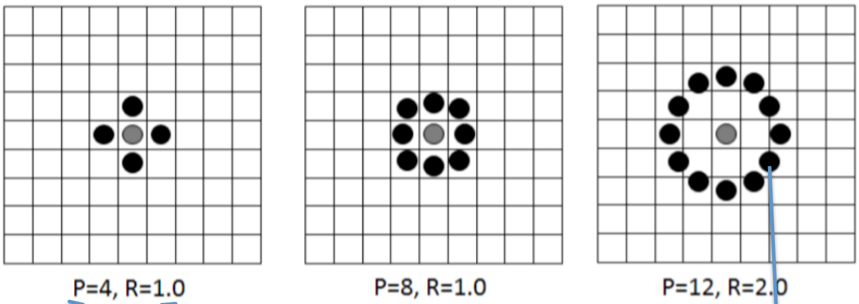
=> we need a grayscale and invariant pattern measure for face recognition =>

Local Binary Patterns

- LBP features are usually obtained from image pixels of a **3×3** neighborhood region
- Compare the 8 neighboring pixel intensity values to the intensity value of the **central** pixel in the region and represents the result as a 8-bit **binary code**
- Binary code can be further converted into a **decimal number** by applying weights to each bit and computing the sum



Multiscale LBP



number of sampling points

radius – the distance of the sampling points from the center pixel

If a sampling point is off the pixel grid, bilinear interpolation of pixel values can be applied to obtain the intensity value of the sampling point

MLBP images encoded at different scales

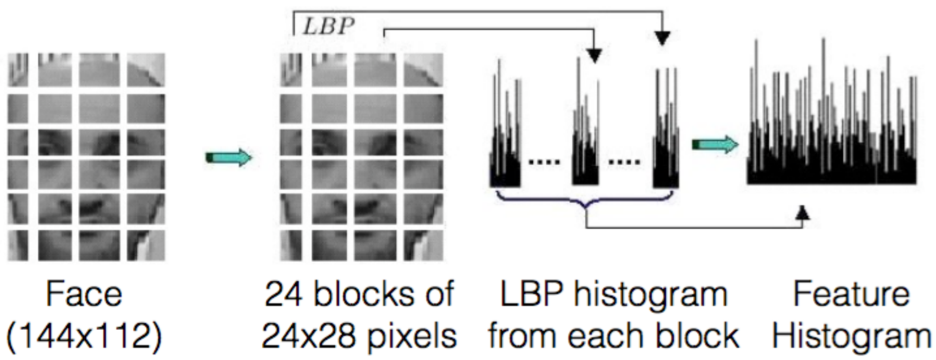
$LBP_{(P=)8,(R=)1}$ $LBP_{8,3}$ $LBP_{8,5}$

detection of **micro** details

highlight the **macro** features

Histogram of LBPs and Matching

- After LBP encoding of each pixel, the face image is divided into several smaller windows and the **histogram** of LBPs **in each window** is computed
- Number of **bins** in the histogram is 8 (LBP) and 2^P MLBP)
- Global feature vector is generated by **concatenating** histograms of all the individual windows and normalizing the final vector
- 2 face images can be matched by computing the **similarity** (or distance) between their feature vectors

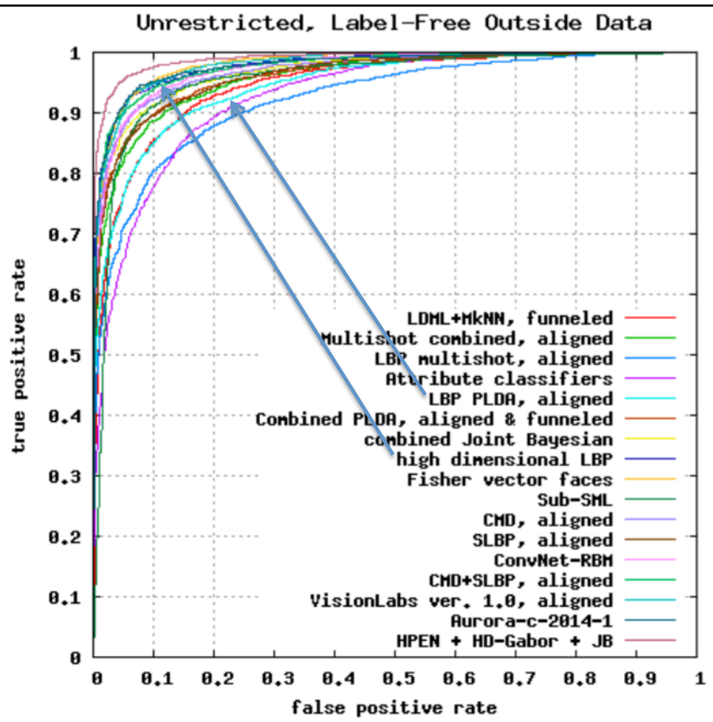


+region weighting:

black squares = 0
dark gray = 1
light gray = 2
white = 4

LBP Results

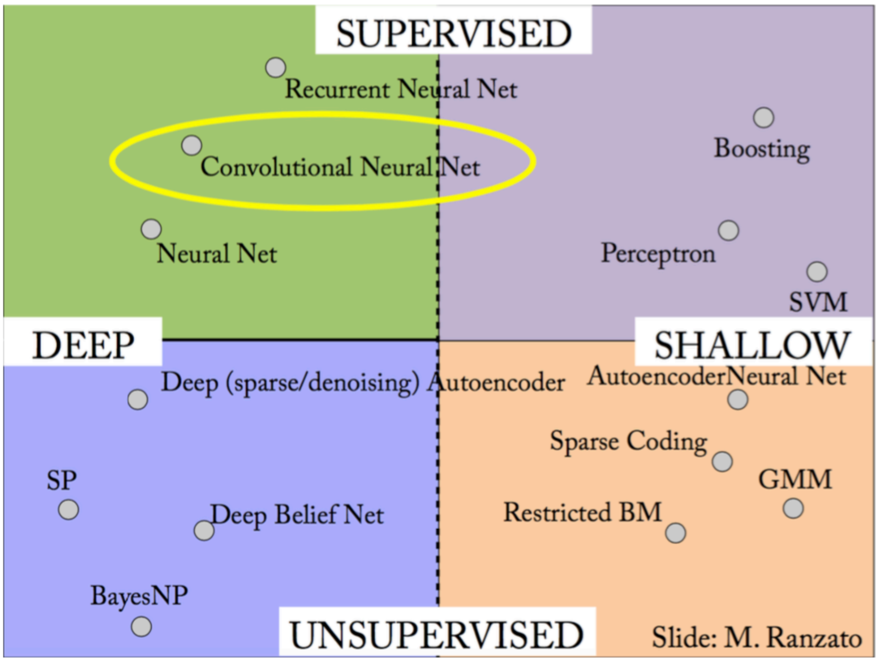
LFW db



Deep Learning Approach

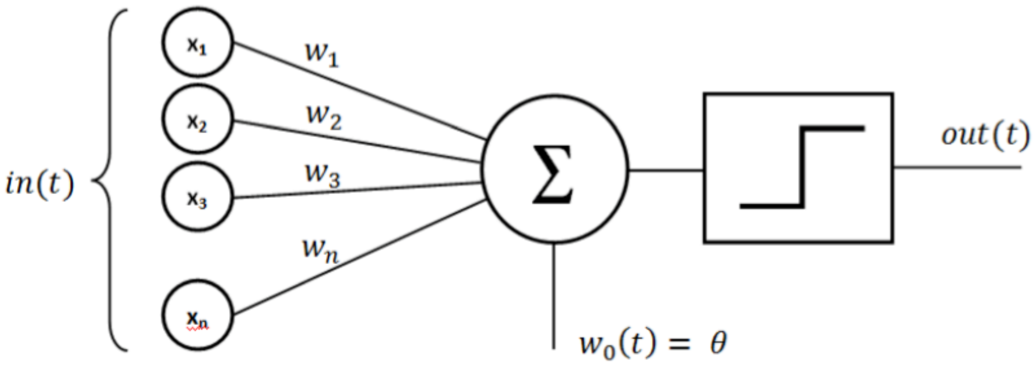
- Representations we have discussed are not strongly invariant!

- **Solution:** multi-layer neural network
- Layers: hierarchy of feature extractors (from simple to complex)



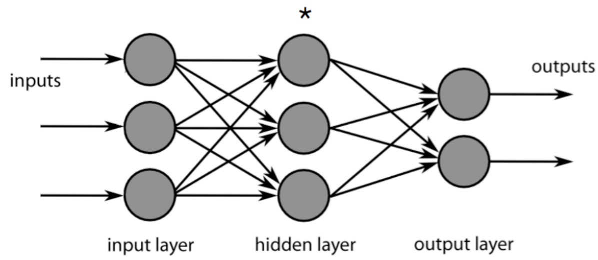
Foundations for CNNs set by Hubel & Wiesel in 1962 by their multi-stage architecture

But it All Started with the ... **Perceptron**



From Percepton…to Layer…to Artificial NN

- Single **layer** perceptrons cannot model non-linear functions
- Adding a **hidden layer** with a non-linear activation function addresses this
- ANN contains sets of adaptive weights
  – Learned during training

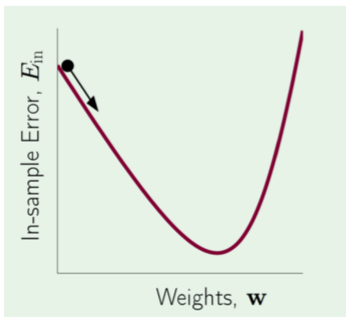


Set the Weights

- We need a **strategy** to set the weights with respect to observed error on the training set >>>

1. Initialize $\mathbf{w}(0)$
2. For $t = 0, 1, 2, \ldots$ [to termination]
3. $\mathbf{w}(t+1) = \mathbf{w}(t) - \alpha \nabla E_{\text{in}}(\mathbf{w}(t))$
4. Return final $\mathbf{w}$

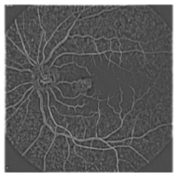
α = learning rate

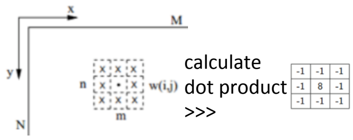


- Use: Stochastic Gradient Descent
- What's an efficient way to calculate the gradients? Backpropagation!

From Deep Learning … to **CNN**

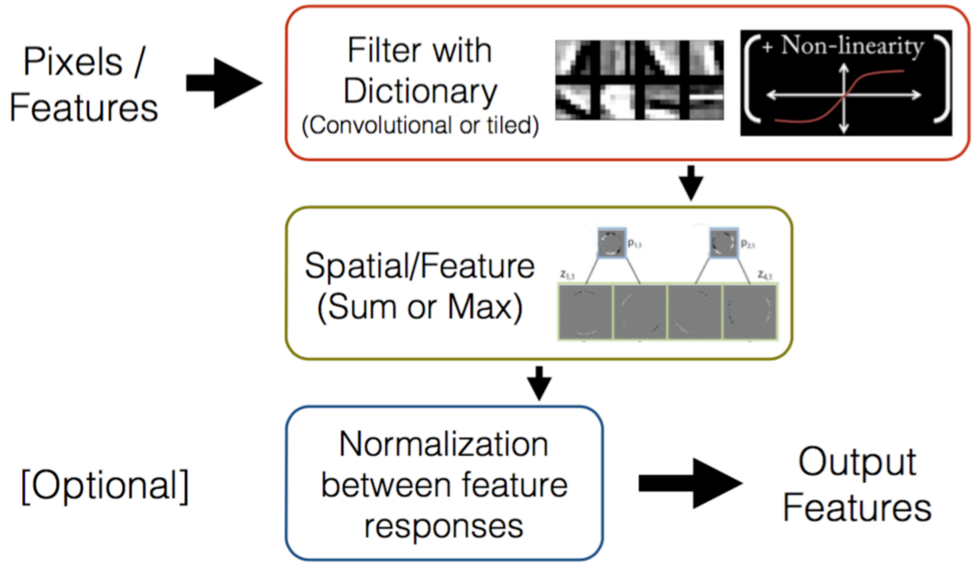
- What is deep learning?
  - The simple answer: **Just multilayer artificial neural networks!**
- What is convolution?
  - The simple answer: **Image filtering!**



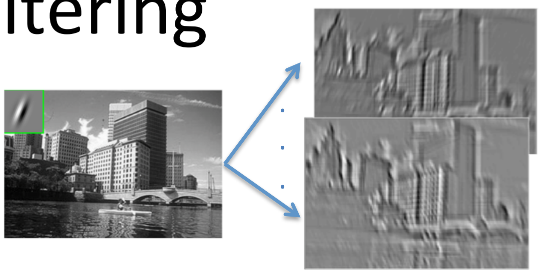


$$\mathbf{w}(x,y) * \mathbf{f}(x,y) = \sum_{i=\lfloor -m/2 \rfloor}^{\lfloor m/2 \rfloor} \sum_{j=\lfloor -n/2 \rfloor}^{\lfloor n/2 \rfloor} w(i,j)\, f(x-i, y-j)$$

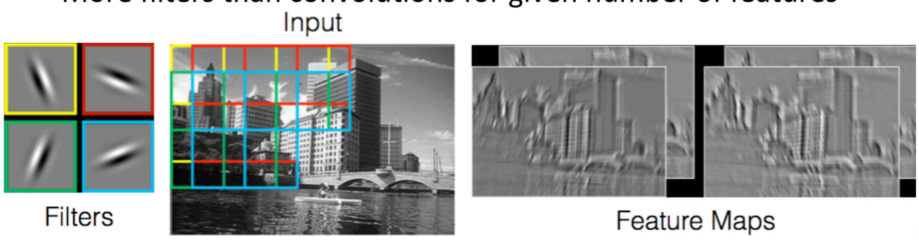
Layer(!) Components



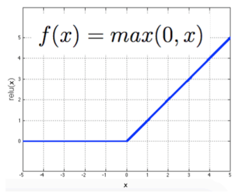
Filtering



- Convolutional
  - Dependencies are local
  - Stride 1, 2,... (faster, less mem.)
- Tiled
  - Filters repeat every *n* pixels
  - More filters than convolutions for given number of features



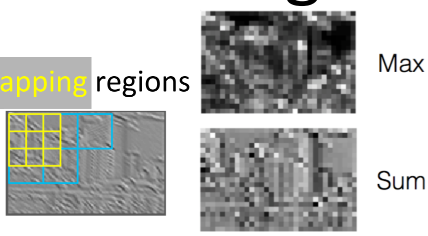
- Non-linearity
  - Use Rectified Linear Units (ReLU) to simplify backpropagation, make learning faster, avoid saturation issues

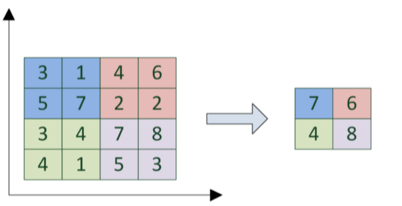


Preferred Option

Pooling

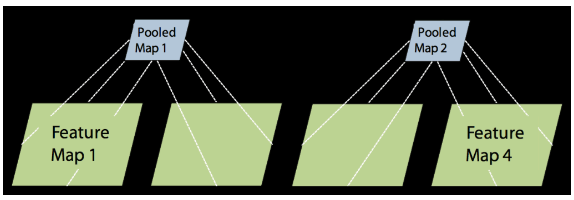
- **Spatial** pooling
  - Non-overlapping / overlapping regions
  - Sum / Max



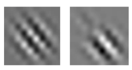


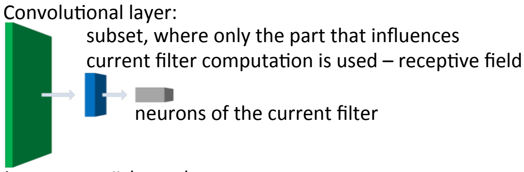
Maximal pooling with a 2×2 filter and stride=2

- Pooling across **feature groups**
  - Additional form of inter-feature pooling
  - MaxOut networks



- Role of (spatial) pooling
  - Invariance to small transformations
  - Larger receptive fields (see more of the input)

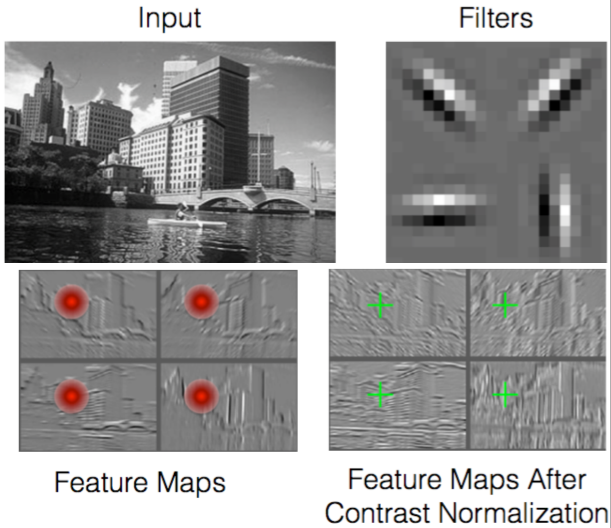




Contrast Normalization

- Between / across **feature maps**
- Local mean = 0, local std. = 1, 7×7 **Gaussian**
- **Equalizes** the feature maps

- Role of Normalization
  - Introduces local competition between features (just like top-down models, but a more local mechanism)
  - helps to scale activations at each layer (each gradient step makes more progress)
  - Empirically, seems to help a bit (1-2% on ImageNet)

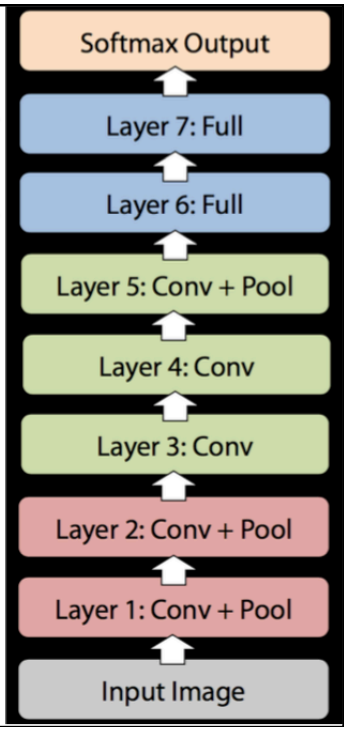


AlexNet Architecture

decision making >>>

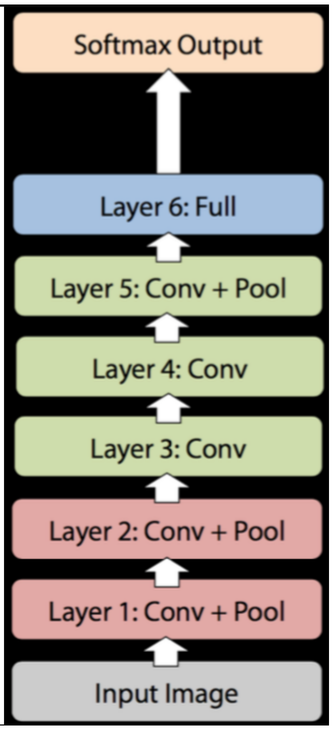
feature transformation

- 8 layers
- Trained on ImageNet
- 18.2% top/rank-5 error (2009?!)



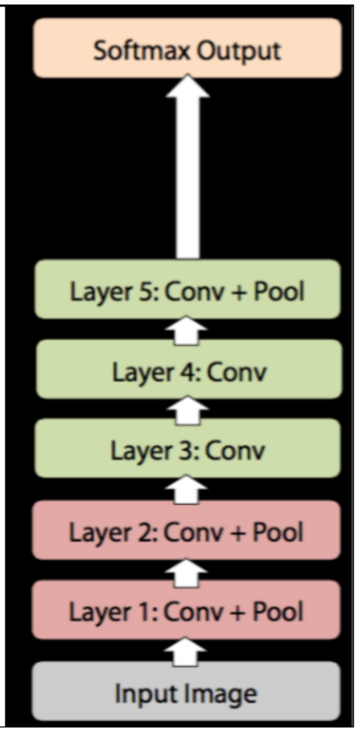
Test by Removal of Layers

- Remove top fully connected layer (Layer 7)
- Drop **16 million** parameters
- **Only** 1.1% drop in performance



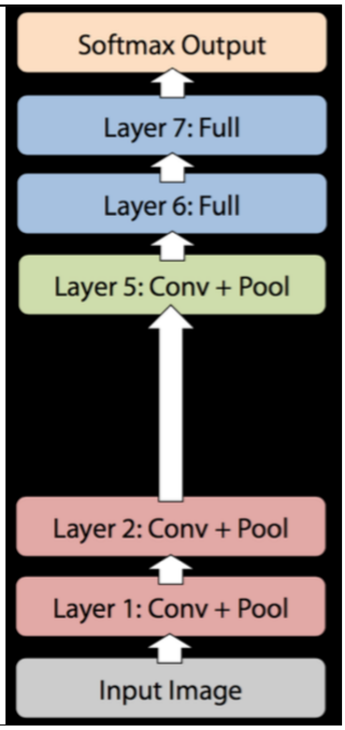
Test by Removal of Layers

- Remove both fully connected layers (Layers 6 & 7)
- Drop ~50 million parameters
- 5.7% drop in performance



Test by Removal of Layers

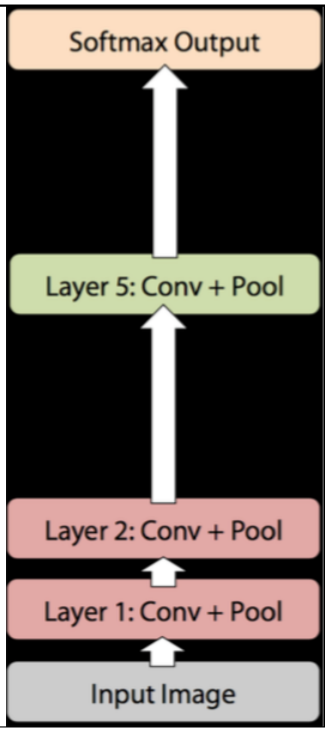
- Now remove the upper feature extractor layers (Layers 3 & 4)
- Drop ~1 million parameters
- 3.0% drop in performance



Test by Removal of Layers

- Now remove the upper feature extractor layers and fully connected layers (Layers 3, 4, 6, 7)
- Now only 4 layers
- 33.5% drop in performance

- **CONCLUSION:**
  Depth of network is key



Analysis of the discriminative information contained in each layer

Take features **from each layer** (ImageNet-pretrained**)** and input them into linear SVM or Softmax

	Cal-101 (30/class)	Cal-256 (60/class)
SVM (1)	44.8 ± 0.7	24.6 ± 0.4
SVM (2)	66.2 ± 0.5	39.6 ± 0.3
SVM (3)	72.3 ± 0.4	46.0 ± 0.3
SVM (4)	76.6 ± 0.4	51.3 ± 0.1
SVM (5)	$\mathbf{86.2 \pm 0.8}$	65.6 ± 0.3
SVM (7)	$\mathbf{85.5 \pm 0.4}$	$\mathbf{71.7 \pm 0.2}$
Softmax (5)	82.9 ± 0.4	65.7 ± 0.5
Softmax (7)	$\mathbf{85.4 \pm 0.4}$	$\mathbf{72.6 \pm 0.1}$

DBs

#training images per class

Higher layers generally produce more discriminative features, i.e.
as the feature hierarchies become deeper, they learn increasingly powerful features

Why does this work so well?
Most probable explanation:
lots of data!

Results on Labeled Faces in the Wild Db

- Google FaceNet (Schroff et al., CVPR 2015): 99.63%±0.09% (22 layers deep model)
- Deep Face: 97.35%
- DeepId2+: 98.7%

- This reduces the error reported for DeepFace **by more than a factor of 7** and the previous state-of-the-art reported for DeepId2+ **by 30%**

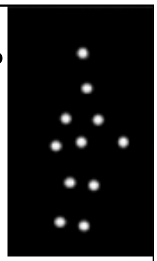
- Jan. 2017: 7 with even better results (see LFW web page), best being Dahua-FaceImage: 99.78%±0.07%, but all commercial recognition systems whose algorithms have not been published and peer-reviewed

Results on **L**abeled **F**aces in the **W**ild Db

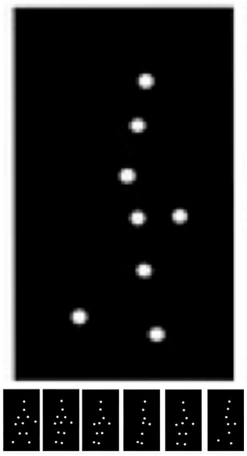
All pairs of images that were incorrectly classified

What is this?

A Xmas tree with lights in the dark? ;-)



What if we animate it? >>>



Psychological studies showed that humans have small but significant ability to recognize people they know well by their **gait**

This ability has encouraged the research for using gait as the means of **biometric identification**

Early studies on **Point Light Displays** (PLD), which enable isolated study of motion by removing all other contexts from observed subjects, confirmed this ability

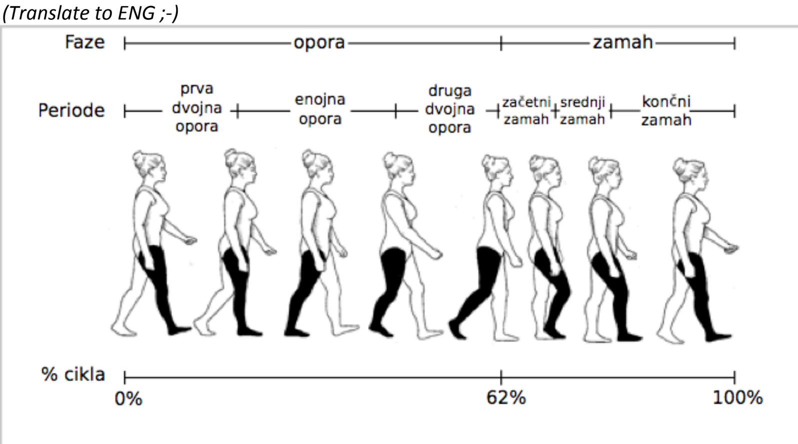
Surveillance Scenario

- Uncontrolled environment
- Low resolution
- What can be observed for identification?
  - Face
  - Ear
  - Iris (perhaps)
  - Gait
  - ?
- The use of gait recognition in real-life application is still **limited**, mostly because of covariate factors that influence individual's gait and therefore make recognition task more difficult (e.g. view changes, walking speed changes, occlusions, etc.)
- Combine them!

What is **Gait**?

Coordinated cyclic combination of gestures that form human motion/walk/run

(Translate to ENG ;-)



+ can be captured with ordinary equipment without individual's awareness or even consent

– unknown level of uniqueness
– covariate factors that change gait characteristics (external: changes of view, direction, speed of movement, illumination conditions, weather, clothing, footwear, terrain, etc.); internal: changes due to illness, injuries, ageing, pregnancy, etc.)
– problems due to uncertain measurements, occlusions, use of noninvasive acquiring techniques (without sensors or markers)

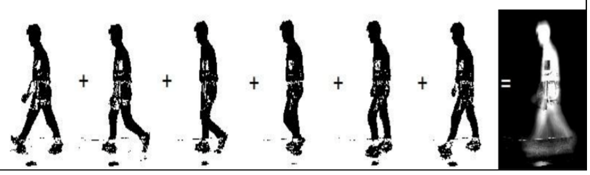
Main Approaches

- Model-free approaches
  - Measurements directly on 2D images
  - Mostly use geometric representations like silhouettes, optical flow, joint trajectories, history of movement, etc.
  - Also take appearance of individual into consideration
  - Less sensitive to covariate factors that result in variations of gait dynamics (e.g. ageing, illness, and walking speed change), but more susceptible to factors that result in changes of appearance (e.g. clothing, obesity, hairstyle, etc.), changes of view, and direction of movement
- Model-based approaches
  - Build the model of human body or its movement and acquire gait parameters from this model: step dimensions, cadence, human skeleton, body dimensions, locations and orientations of body parts, joint kinematics, etc.
  - More resistant to problems like changes of view and scale
  - Computationally demanding and especially susceptible to problems like occlusions

+ CNN

Model-free Approaches – Silhouette Approach

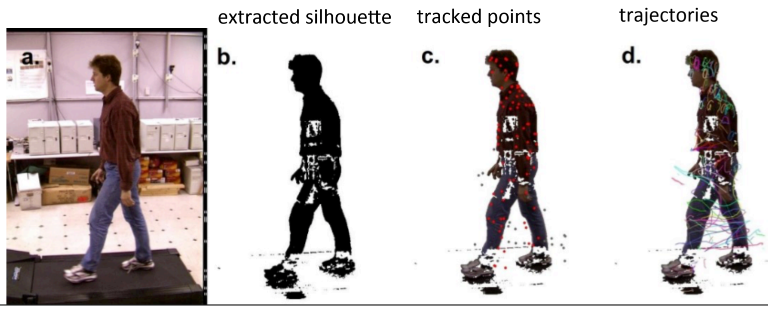
- Based on direct comparison of gallery and probe gait cycles by computing the Euclidean distance of corresponding silhouettes within these two gait cycles
- Upgrade: **averaged silhouette** (**G**ait **E**nergy **I**mage) represents the sum of silhouettes for one gait cycle
  - Such approach eliminates comparison problems of subjects with different cycle period and different choice of initial pose for the gait cycle
  - Form feature vector from GEI
  - Use PCA and LDA for dimensionality reduction and subspace optimization
  - Compare using the Euclidean distance



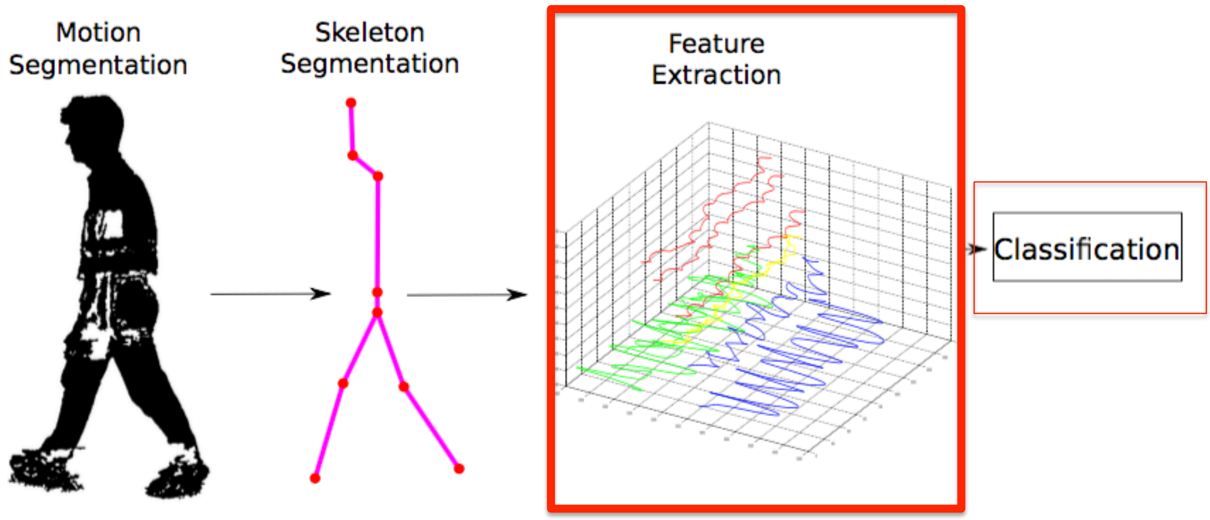
Model-free Approaches – Probabilistic Spatio-Temporal Model

(Model is the result of the method, thus the approach is not model-based ;-)

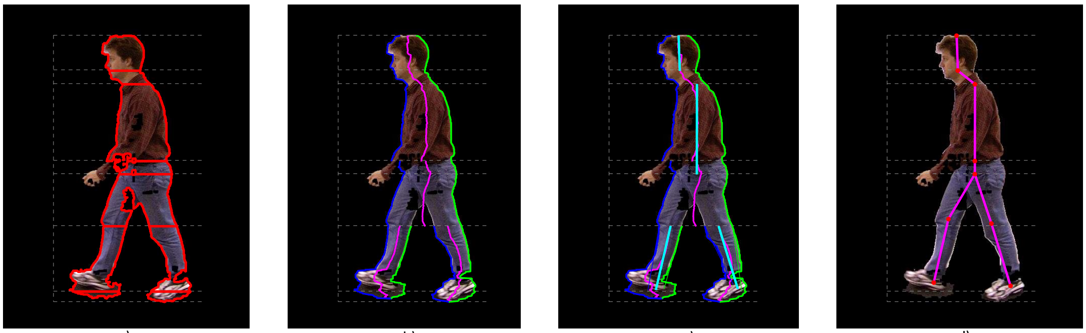
- Based on optical flow tracking of random points in extracted silhouettes; build by Principal Component Analysis (PCA), Expectation Maximization (EM) and Gaussian mixtures (GMM)
- Points are tracked based on texture information of the small area around them
- The points and their movements form point **trajectories** in time
- Method concentrates on using gait dynamics for recognition purposes
- The model with maximal similarity is chosen as a match



Model-based Approach – Skeleton Segmentation-based



Skeleton Segmentation



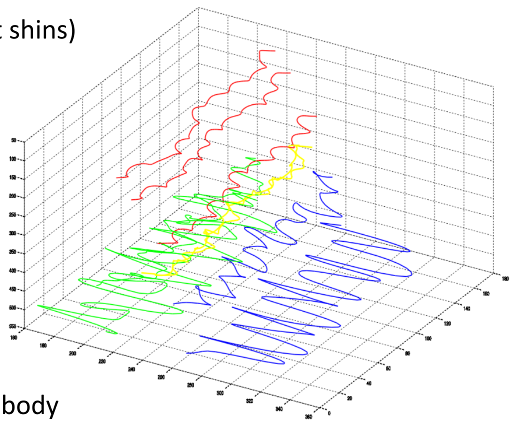
1. Calculate body height and segment lengths based on average anatomical properties
2. Divide silhouette into individual body parts based on calculated segment lengths (a)
3. For each body segment we estimate initial skeleton as segment midpoint (b)
4. Fit lines through this points (c)
5. Deal with lower body part by detecting thigh and shin bones (d)

=> Extracted bones form a simplified skeleton of human body for single silhouette frame

Gait Signal Extraction

Raw signals:

- body part angles
  (head, neck, torso, left and right thighs, left and right shins)
- joint locations (left and right knees and ankles)
- masses and its centres for the whole silhouette and individual body parts
- silhouette width and height



Derived signals:

- phase difference of thighs and shins that describe the correlation among left and right thighs or left and right shins
- body part mass ratio represents the ratio between masses of specific body parts and the mass of entire body
- body part center of mass distances represent the distances of center of mass of specific body parts to the center of mass for the entire body

TOTAL: 64 signals (features)

Classification

Signals of all of the features are serialized into rows of the matrix:

$$\begin{matrix} a_{1,1} & a_{1,2} & \cdots & a_{1,N_o} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,N_o} \\ \vdots & \vdots & & \vdots \\ a_{N_z,1} & a_{N_z,2} & \cdots & a_{N_z,N_o} \end{matrix}$$

cycle descriptor

feature descriptor

frame descriptor

- Holistic approach to classification => we calculate the distances between each probe descriptor (gait **cycle**) and all descriptors of all the models in the gallery
- Feature fusion approach => perform the classification on feature **descriptors** separately for each feature
- **Frame**-based classification => to bypass <u>cycle alignment problems</u>, perform classification in a way that is independent of chosen gait cycle phase (!!!)

Feature Space Transformation

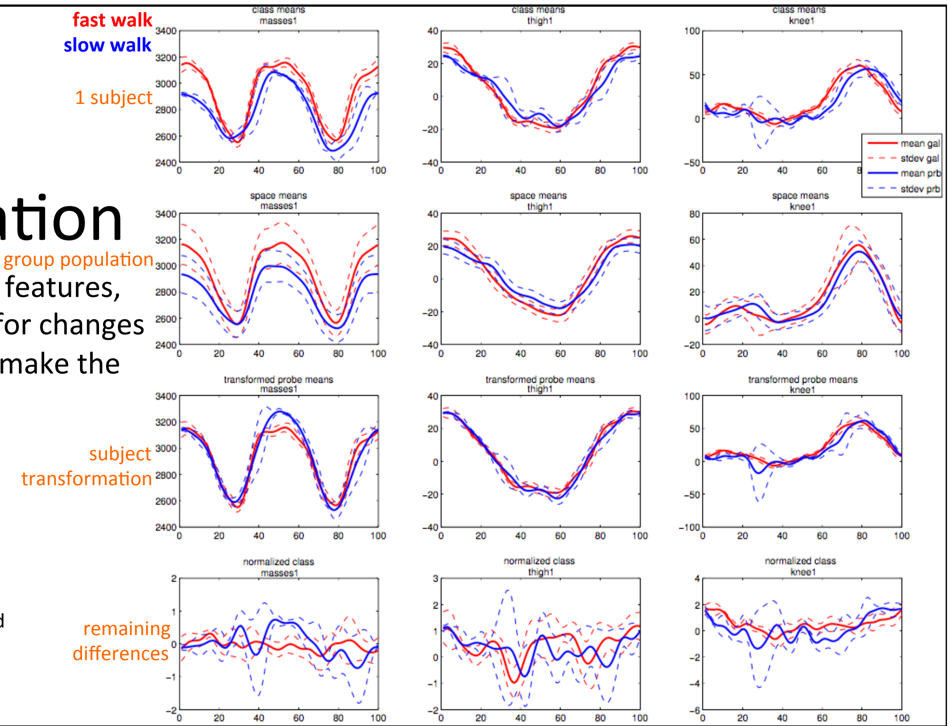
As we deal with dynamic features, we have to compensate for changes due to walking speed => make the transformation:

$$\vec{\mathbf{s}} = (\vec{\mathbf{s}} - \vec{\mu}_{\mathbf{g}})/\vec{\sigma}_{\mathbf{g}}$$

signal vector of single feature of single gait cycle sample

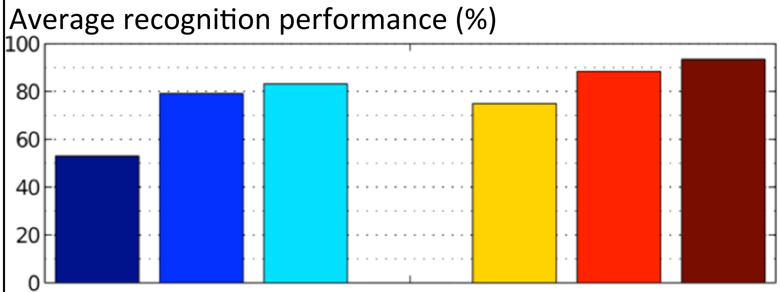
mean vector of the same feature for target group

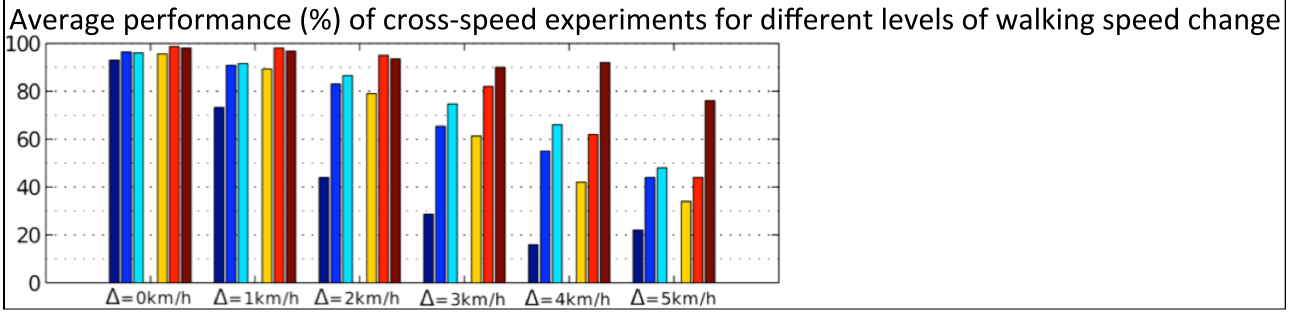
vector of standard deviations for the same group



Results

OU-ISIR/A gait database:
high quality gait silhouette sequences of **34** subjects from **side** view, walking on a treadmill with **speed variation** from 2km/h to 10km/h at 1km/h interval





Comparison to Other Works

Method→ / Experiment↓	Fusion [24]	**Frames**	SN [8]	HMM [14]	SVT [18]	HSC [19]	DCM [20]
Small speed change (3 km/h / 4 km/h)	96	**100**	–	84	90	96	98
Large speed change (2 km/h / 6 km/h)	68	**90**	34	–	58	68	82
All walking speeds (2 km/h – 7 km/h)	88,33	**93,44**	–	–	–	85,67	92,44

(not published yet!)

Multibiometrics

Why?

- Different biometric sources usually **compensate** for the inherent limitations of one another
- Provide a wider population **coverage** (reduce the failure to enroll rate)

How?

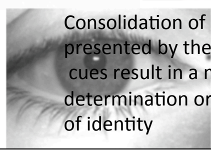
- By consolidating the information given by multiple biometric sources – information fusion

Unibiometric Limitations

Biometric source can become unreliable due to

- sensor or software malfunction
- poor quality of specific biometric trait of the user
- deliberate manipulation
- accuracy requirements that cannot be met

Fusion multibiometric system developed by Cogent Systems



Consolidation of information presented by these multiple cues result in a more accurate determination or verification of identity

Why the Accuracy Improves?

1. Fusion **increases** the dimensionality of the feature space and **reduces** the overlap between the feature distributions of different individuals

   $\Rightarrow$ combination of multiple biometric sources is more unique

2. Noise, imprecision, inherent drift (caused by factors like aging) in a subset of the biometric sources can be compensated by the **discriminatory** information provided by the remaining sources

   $\Rightarrow$ multiple biometric sources provide redundancy and fault-tolerance

Other Advantages

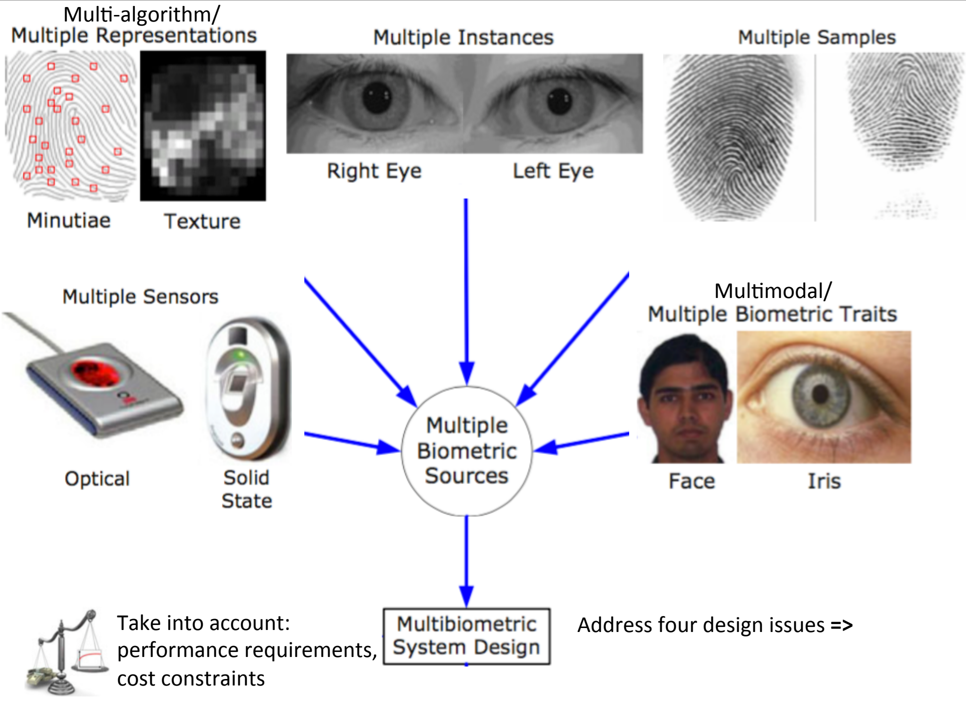
- Flexibility – a **subset** of traits may be acquired based on the nature of the application under consideration and the convenience of the user; continuous monitoring
- Effective search – use a relatively simple but less accurate modality to **prune** the database before using the more complex and accurate modality
- Antispoofing – difficult to circumvent multiple biometric sources simultaneously
  - asking the user to present a random subset of traits for verification of presence of a **live** user (liveness detection)

Disadvantage?

- They are more **expensive**
  - Always think about the tradeoff between the added cost and the benefits

Fact: multibiometric systems are being increasingly **deployed in many** large-scale identification systems involving millions of users (border control or national identity systems, etc.) because of their ability to achieve high recognition accuracy based on existing technologies, which **far outweighs** the additional cost in such applications

Five Classes of Sources of Multiple Evidence =>



Note (intermezzo)

While in principle a large number of sources can be combined to improve the identification accuracy, practical factors such as

- cost of deployment
- enrollment time
- small training sample size
- accuracy requirements
- throughput time
- user acceptance

will **limit** the number of sources used in a particular application

Design Issues

1. What are the various sources of biometric information that should be used in a multibiometric system? (**choice** of biometric sources)
2. Should the data corresponding to multiple biometric sources be acquired simultaneously in a parallel mode or in a sequence? (acquisition **sequence**; similarly for processing)
3. What type of information (i.e. raw data, features, match scores or decisions) is to be fused? (**fusion level**)
4. What fusion scheme should be employed to combine the information presented by multiple biometric sources? (**fusion methodology**)

Dependencies

Between the Design Choices in a Multibiometric System

Multibiometric sources	Type of information fused: Raw data	Features	Scores	Decisions	Acquisition architecture: Serial	Parallel	Processing architecture: Serial	Parallel
Multiple sensors	✓	✓	✓	✓	✓	✓	✓	✓
Multiple representations	×	✓	✓	✓	×	✓	✓	✓
Multiple matchers	×	×	✓	✓	×	✓	✓	✓
Multiple instances	×	✓	✓	✓	✓	✓	✓	✓
Multiple samples	✓	✓	✓	✓	✓	×	✓	✓
Multiple traits	×	✓	✓	✓	✓	✓	✓	✓

✓ Compatible

× Not compatible

Evidences derived from a single biometric trait

Multi-sensors Systems

(Additional Properties)

built in registration

+ also **assist** in the intermediate processing stages of the individual component systems like image segmentation and registration

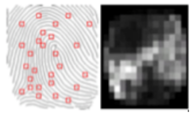
– cost

+ two fingerprint images can be processed independently without spatially registering them

– larger enrollment and verification times

Multi-algorithm Systems

(Additional Properties)

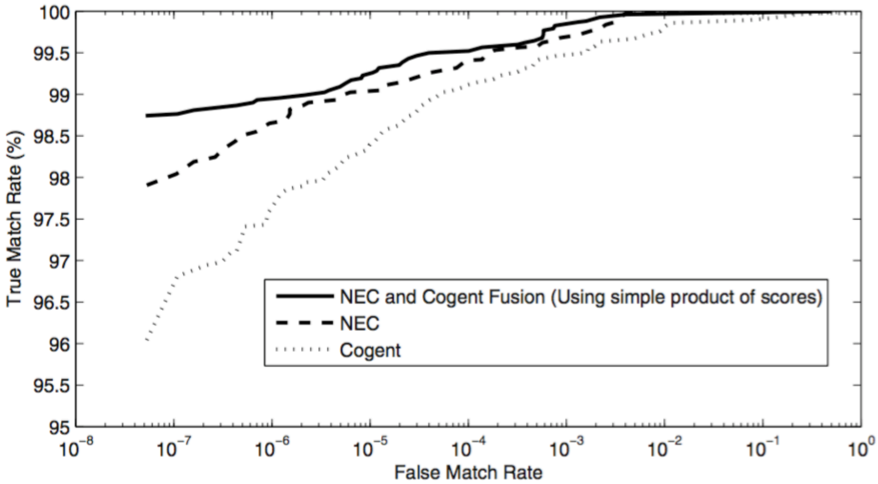


+ cost-effective => does not require the use of new sensors

+ no additional inconvenience for the user

– multiple sources tend to be **correlated** (noisy input will affect both the texture-based and minutiae-based algorithms but to different extents) => limits the possible improvement in the matching accuracy

Accuracy Improvement in a Multi-algorithm Fingerprint Verification System



NIST Fingerprint Vendor Technology Evaluation 2003, two most accurate matchers

Multi-instance Systems
(Additional Properties)

- Also known as multi-unit systems

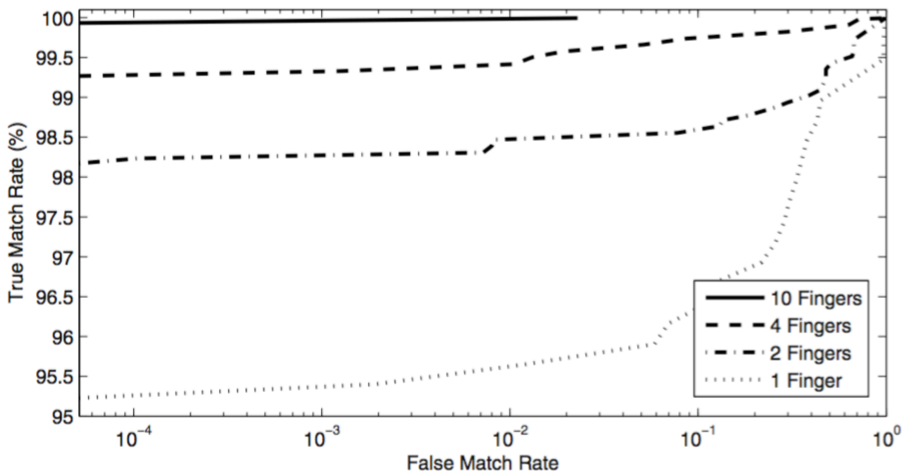
+ Do **not** necessitate the introduction of new sensors nor do they entail the development of new feature extraction or matching algorithms

+ Especially beneficial for users whose biometric traits cannot be reliably captured due to inherent problems (dry finger skin, drooping eyelids, etc.)

– In some cases a new sensor arrangement might be necessary in order to facilitate the simultaneous capture of the various units/instances

rapid acquisition of all ten fingers in three steps (by Identix)

Accuracy Improvement in a Multi-instance Fingerprint Verification System



FpVTE 2003

Multi-sample Systems

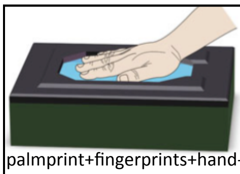
(Additional Properties)

Note: if captured simultaneously it would be considered as a multi-sensor system

- To account for the **variations** that can occur in the trait or to obtain a more **complete** representation of the underlying trait
  - Variations in the facial pose
  - To form a complete fingerprint

Mosaiced Fingerprint Image

- Determine the number of samples that need to be acquired (variability, typicality)
  - Determine sample collection protocol
  - Automatically select the “optimal” subset



palmprint+fingerprints+hand-shape

Multimodal Systems

(Additional Properties)

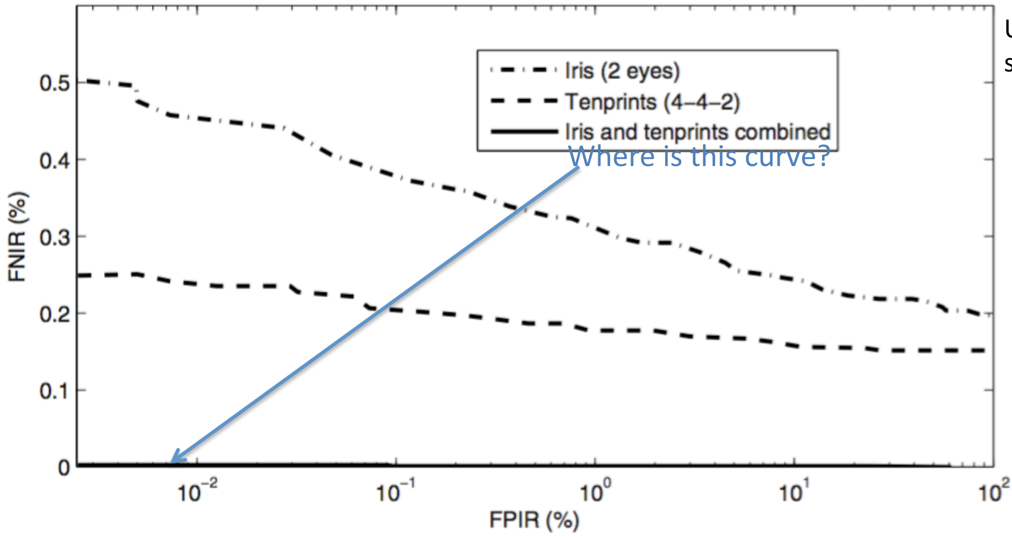
+ Traits are expected to be uncorrelated
  + Leads to **greater improvement** in performance compared to other types of multibiometric systems
  - Voice and lip movement could exhibit significant correlation

– Higher/st cost

– There is a diminishing return after strong biometric traits (e.g. fingerprint and iris) have already been fused

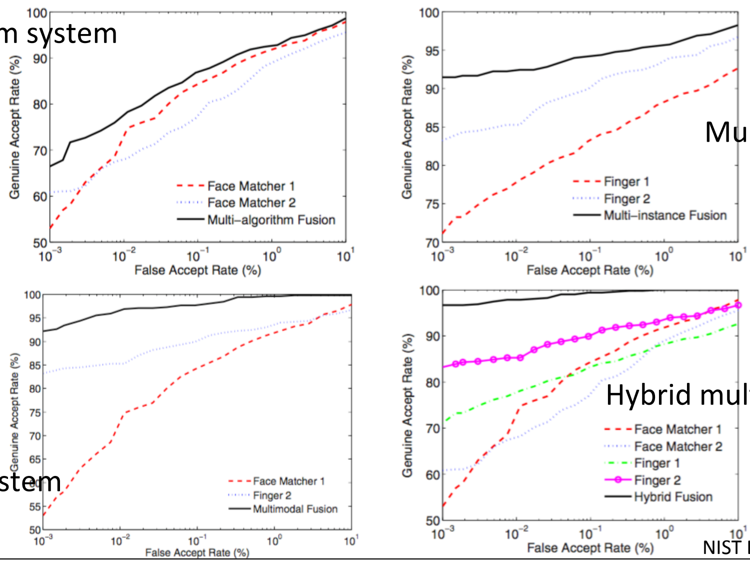
- Depending on the fusion scheme, there may even be a performance **degradation** if many biometric traits with lower accuracy are added

Accuracy Improvement in a Multi-instance and Multimodal Biometric Identification System



UID India,
subset of 20k subjects

Accuracy Improvements due to **Different Sources** in Multibiometric Systems



NIST Biometric Score Set Release 1 DB

Multiple Biometric Sources: Final Remarks

- Multi-instance and multimodal biometric systems are more **popular** than the others
- Use multiple instances or multiple traits or a **combination** of both

- Define: **Multi-factor authentication**
  - Use biometric traits in conjunction with non-biometric authenticators like tokens or password

Acquisition and Processing Architecture

Multibiometric sources	Type of information fused				Acquisition architecture		Processing architecture	
	Raw data	Features	Scores	Decisions	Serial	Parallel	Serial	Parallel

Two main goals:

- Minimal inconvenience to the user
- Minimize throughput time
  - Sequence in which the procured biometric data is processed can significantly impact the throughput time in large-scale identification systems (involving millions of enrolled users)

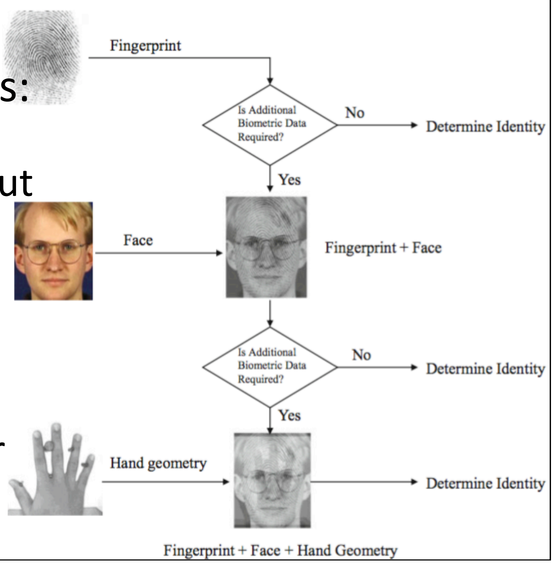
Acquisition Sequence

- Typically, multibiometric systems employ the **serial** acquisition approach, where the evidence is gathered sequentially
  - + Does not require any special sensor arrangement and typically has a lower installation cost
- In some cases the evidence may be acquired **simultaneously**
  - Face, voice, lip movements of a user may be acquired simultaneously

  - Multiple fingerprints can be captured in parallel
  - + Can decrease enrollment and authentication times and improve the usability

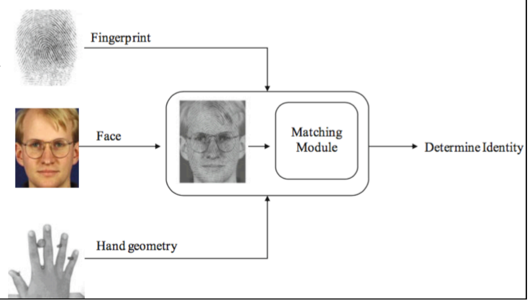
Processing Sequence

- Serial/parallel approach
- **Serial** (cascade) case arrangements:
  - Processing time can be effectively reduced if a decision is made without waiting for the outputs of all the unibiometric systems
  - Allow users to choose which of his biometric traits should be captured first (convenience)
  - Prune the database to search faster and more efficiently



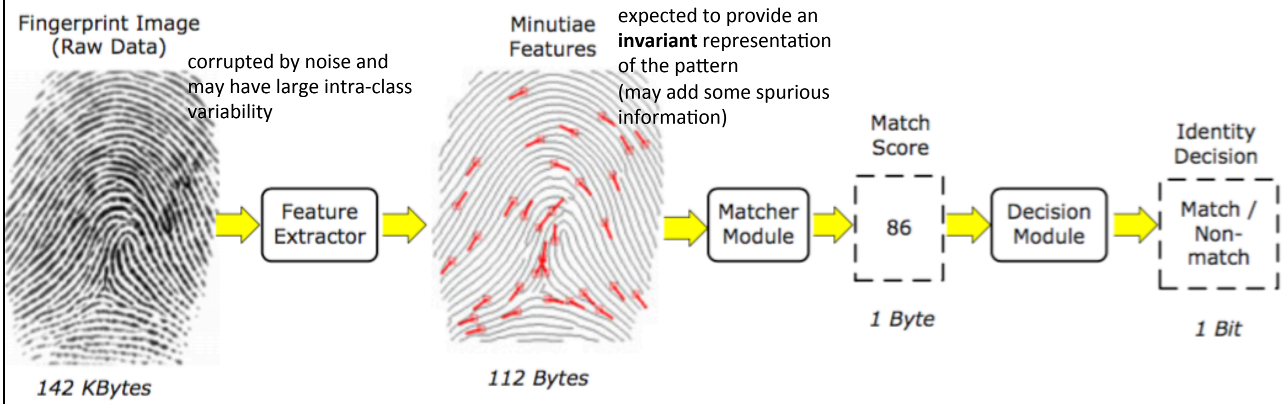
Processing Sequence

- **Parallel** case arrangements:
  – Each unibiometric system processes its information **independently** at the same time and the processed information is combined using an appropriate **fusion** scheme
  – Generally has a higher accuracy because it utilizes more evidence => **preferred** architecture!



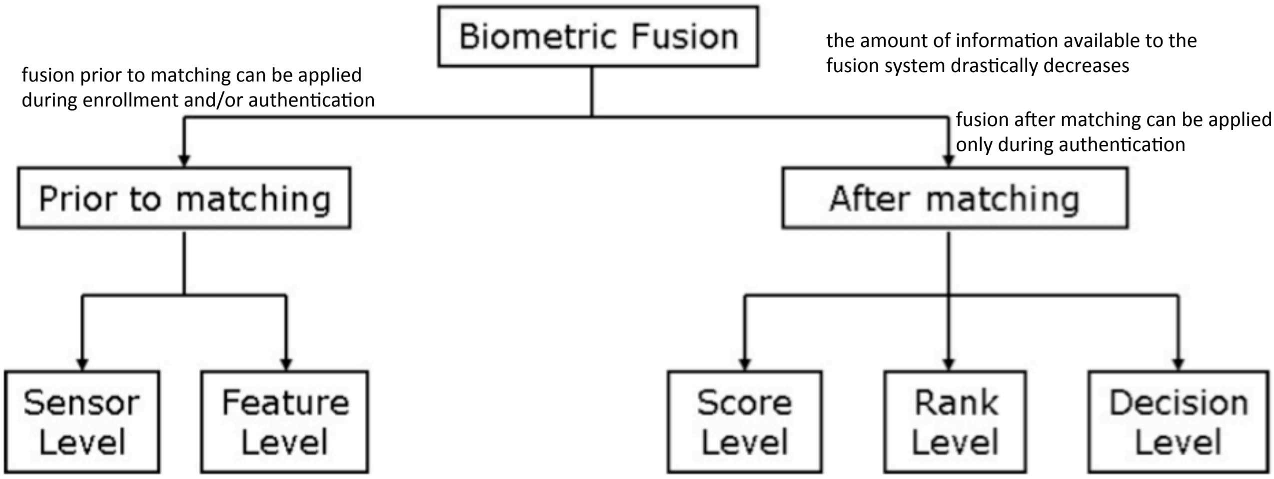
Fusion

A fundamental issue in the design of a multibiometric system is to determine the **type** of information that should be **consolidated** by the fusion module:



The amount of information available for fusion gets reduced as one progresses along the various processing modules of a biometric system &
Fusion can be accomplished by utilizing the information available in any of the biometric modules

Fusion Levels



fusion prior to matching can be applied during enrollment and/or authentication

the amount of information available to the fusion system drastically decreases

fusion after matching can be applied only during authentication

applicable only in the identification mode

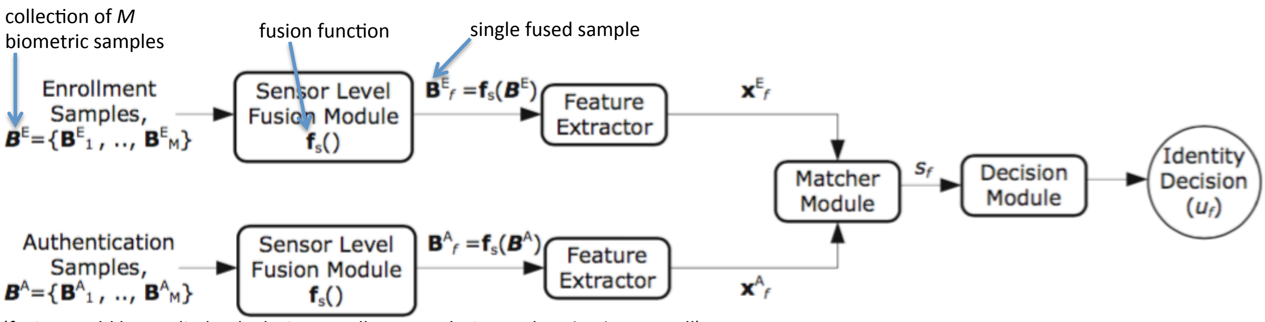
Theory vs. Practice

- Integration at the **feature** level is expected to provide **better** recognition results than **score** or **decision** level fusion
- Not always true since:
  - Fusion process has to deal with the presence of noise in feature sets
  - New matching algorithm may be necessary to compare two fused feature sets
    - Developing efficient matching algorithms is often the most challenging aspect in the design of a biometric system
- Most multibiometric systems fuse information at the **score** level or the **decision** level

Sensor-level Fusion

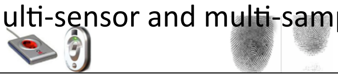
Also known as image-level or pixel-level fusion

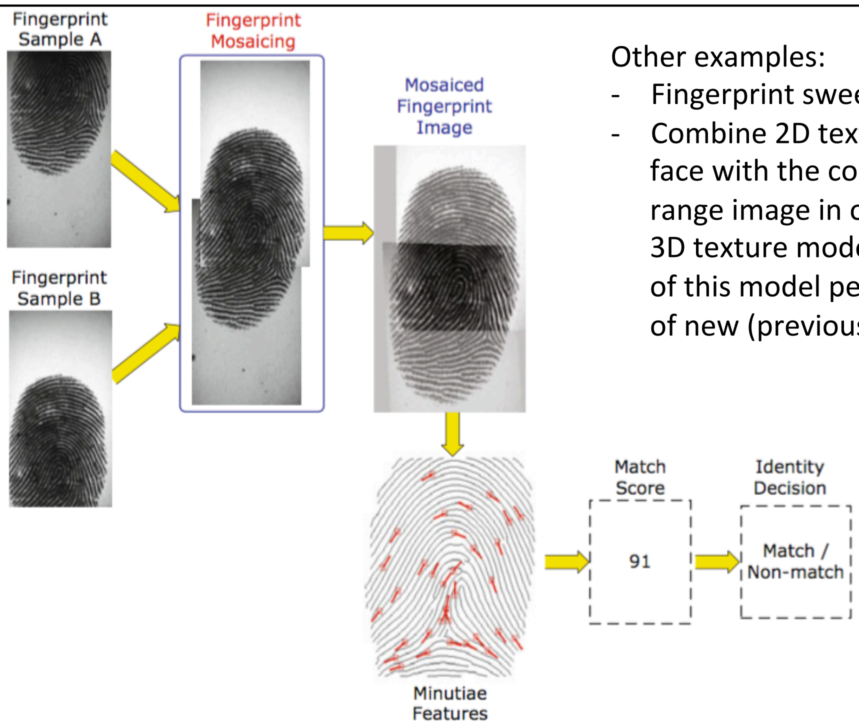
Consolidates evidence presented by multiple sources of raw data **before** they are subjected to feature extraction



(fusion could be applied only during enrollment or during authentication as well)

Applicable only for multi-sensor and multi-sample systems >>>



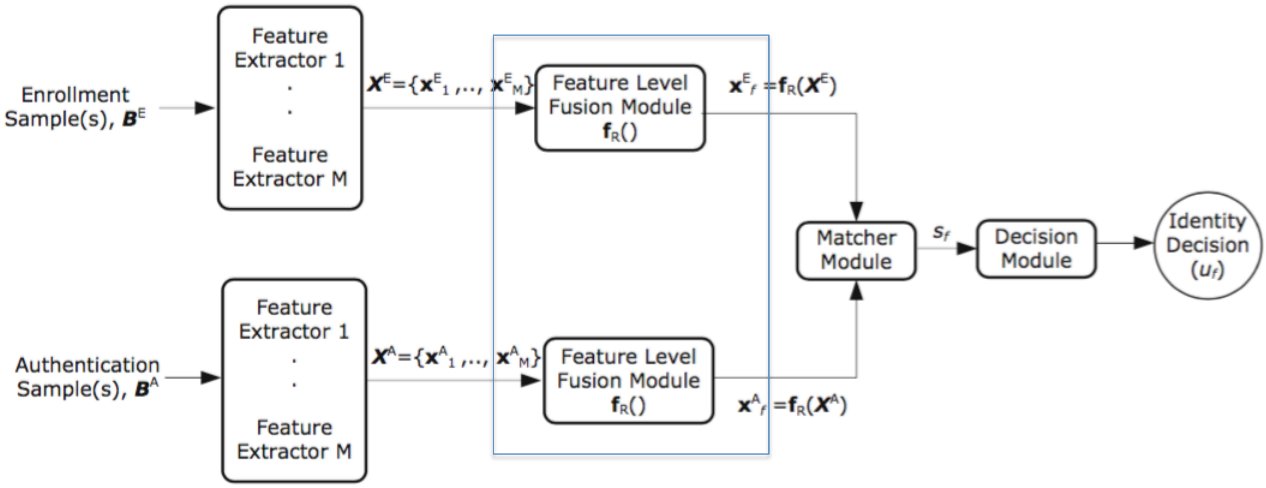


Other examples:

- Fingerprint sweep-sensors
- Combine 2D texture of a person's face with the corresponding 3D range image in order to create a 3D texture model; the availability of this model permits the generation of new (previously unseen) 2D images

Feature-level Fusion

Feature/representation-level fusion **consolidates** evidence presented by two+ different biometric feature sets of the same individual



Categorization of Feature-level Fusion Schemes

- Homogeneous class
  - Feature sets to be combined are obtained by applying **the same** feature extraction algorithm to **multiple** samples of **the same** biometric trait
  - Applicable to multi-sensor and multi-sample systems
- Heterogeneous class
  - Feature sets originate from **different** feature extraction algorithms or from samples of **different** biometric traits (or different instances of the same trait)

Homogeneous Feature Fusion

- Template update:
  - In order to reflect (possibly) permanent **changes**
  - Usually, the template is updated after every successful authentication

Simple template update scheme:

current template new feature vector obtained during authentication

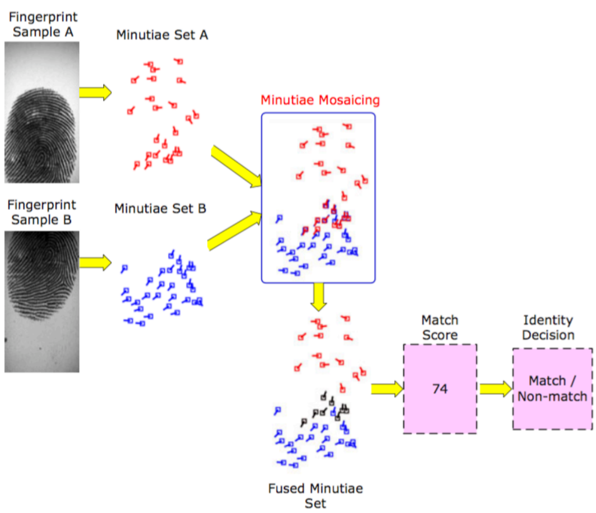
new template

$$\hat{\mathbf{x}}^E = \begin{cases} \dfrac{\mathbf{x}^E + \mathbf{x}^A}{2}, & \text{if } \mathscr{M}(\mathbf{x}^E, \mathbf{x}^A) \geq \tau \\ \mathbf{x}^E, & \text{otherwise} \end{cases}$$

matching function decision threshold

Homogeneous Feature Fusion

- Template improvement:
  - Combine feature sets by appropriately aligning the two sets and then removing duplicate features, thereby generating a **larger** set
  - Used to remove spurious features



Heterogeneous Feature Fusion

Challenges (or why is it difficult to achieve it?):

- Proprietary reasons: most commercial biometric systems do not provide **access** to the feature sets
- **Relationship** between the feature spaces of different biometric systems may not be known
- Incompatible modalities: **variable** length feature set (e.g. minutiae) and **fixed** length feature set (e.g. eigen-faces)
- If both fixed length, then **concatenating** two feature vectors might lead to the curse-of-dimensionality problem

Adopted procedure >>>>>

Fixed-length feature vectors from two biometric sources: $\mathbf{x}_1 \in \mathbb{R}^{d_1}$ $\mathbf{x}_2 \in \mathbb{R}^{d_2}$ $X = \{\mathbf{x}_1, \mathbf{x}_2\}$

Fuse them and get a new feature vector with reduced dimensionality that would better represent the biometric sample: $\mathbf{x}_f$ $d < (d_1 + d_2)$

Sketch of the procedure: **normalize** and **concatenate** vectors, then perform feature **selection** or feature **transformation** on the resultant feature vector in order to reduce its dimensionality

Normalization: $\mathbf{x}_1 = [x_1^1, x_1^2, \cdots, x_1^{d_1}]$ $\mathbf{x}_2 = [x_2^1, x_2^2, \cdots, x_2^{d_2}]$

- may exhibit significant differences in their range as well as form (i.e. distrib.)
- e.g. [0..100] & [0..1] => distance between two augmented feature vectors will be more sensitive to the x^i_1's than the x^i_2's
- map them into a common domain, often with min-max normalization scheme (range [0..1]): $$\hat{x} = \frac{x - \min(h_x)}{\max(h_x) - \min(h_x)}$$ minimum and maximum of all possible *x* values; function that generates *x*
- problem: sensitivity to outliers in the training data

Feature selection or transformation:

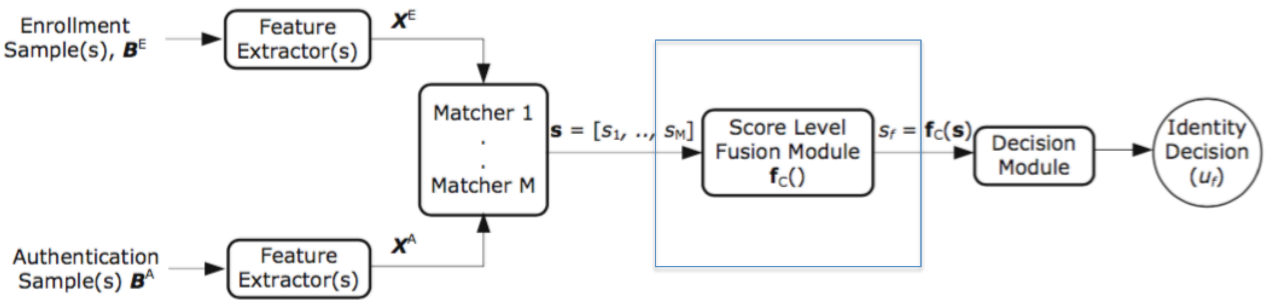
- curse-of-dimensionality dictates that the augmented vector of dimensionality (d_1+d_2) need not necessarily result in an improved matching

- feature **selection** is a dimensionality reduction scheme that chooses a minimal feature set of size d<(d_1+d_2), such that a criterion (objective) function (EER, AUC, average GAR,…) applied to the training set of feature vectors is optimized
    - algorithms: SFS, SBS, SFFS, SBFS,…

- dimensionality reduction may also be accomplished using feature **transformation** methods, where the concatenated vector is subjected to a linear or a non-linear mapping that projects it to a lower dimensional subspace
    - PCA, ICA, MDS, Kohonen maps,…

Score-level Fusion

Also known as fusion at the measurement or confidence level
Match scores output by different biometric matchers are **consolidated**
Relatively easy to access and combine the scores
Most commonly used approach

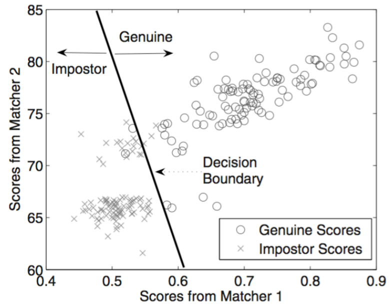


Challenges of non-homogenous match scores:
- distance or similarity measure output
- different numerical ranges of outputs
- different probability distributions of the matchers

Fusion methodologies will vary depending on whether the system operates in the ver./id. mode

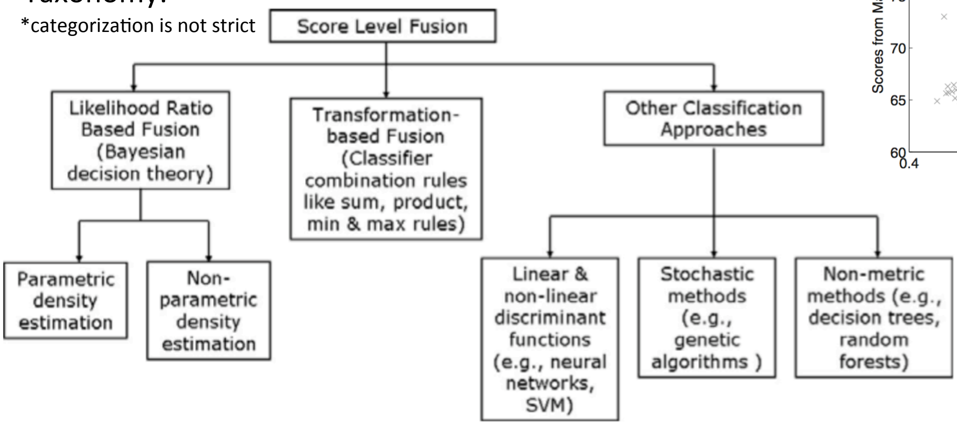
Score-level Fusion in Verification Mode

Based on the training set of match scores from the genuine and impostor classes, a classifier learns a decision boundary between the two classes



*boundary can be quite complex depending on the nature of the classifier

Taxonomy:
*categorization is not strict



Likelihood-ratio-based Score Fusion

Based on the **Bayesian** decision theory &
the **Neyman-Pearson** approach to statistical hypothesis testing

Bayes formula:

prior probabilities of observing the impostor and genuine classes $P(\omega_0)$ $P(\omega_1)$

conditional density function

$$P(\omega_j|\mathbf{s}) = \frac{p(\mathbf{s}|\omega_j)P(\omega_j)}{p(\mathbf{s})}$$

posterior probability

$$p(\mathbf{s}) = \sum_{j=0}^{1} p(\mathbf{s}|\omega_j)P(\omega_j)$$

Decide between the genuine and impostor classes based on the observed match score vector $\mathbf{s}$

>>>>>

Decision rule for minimum error rate is: Decide ω_1 if $P(\omega_1|\mathbf{s}) > P(\omega_0|\mathbf{s})$

use Bayes formula ... to obtain Bayesian rule

$$\text{Decide } \omega_1 \text{ if } P(\omega_1)p(\mathbf{s}|\omega_1) > P(\omega_0)p(\mathbf{s}|\omega_0)$$

if the prior probabilities of the two classes are assumed to be equal

$$\text{Decide } \omega_1 \text{ if } \underbrace{\frac{p(\mathbf{s}|\omega_1)}{p(\mathbf{s}|\omega_0)}}_{\textbf{likelihood ratio}} > 1$$

DONE?

NO! ... Why?

Such decision rule is not widely used in biometric systems because it assumes that both false accept and false reject errors are **equally costly**

In a practical biometric system it is often desirable to have an **upper bound** on the false accept rate (e.g. α=0.1%)

=> modify the decision rule based on the Neyman-Pearson criterion >>>>>

Let's make a statistical test for testing the hypothesis

H_0: *$\mathbf{s}$ corresponds to an impostor*

against the alternative hypothesis

H_1: *$\mathbf{s}$ corresponds to a genuine user*

The probability of correctly rejecting H_0 when H_1 is true gives the genuine accept rate or the *power* of the test

The probability of rejecting H_0 when H_0 is true gives the false accept rate or *level* of the test denoted by α

Neyman-Pearson theorem:

For testing H_0 against H_1 there exists a test Ψ and a constant η such that

decision in favor of H_1

$$P(\Psi(\mathbf{s}) = 1|H_0) = \alpha \qquad \Psi(\mathbf{s}) = \begin{cases} 1, \text{ when } \frac{p(\mathbf{s}|\omega_1)}{p(\mathbf{s}|\omega_0)} \geq \eta \\ 0, \text{ when } \frac{p(\mathbf{s}|\omega_1)}{p(\mathbf{s}|\omega_0)} < \eta \end{cases}$$

If a test satisfies equations for some η, then it is the most powerful test for testing H_0 against H_1 at FAR α >>>>>

Neyman-Pearson theorem (cont.):

According to the theorem, given FAR α, the **optimal** test for deciding whether a match score vector $\mathbf{s}$ corresponds to an impostor or a genuine user is the **likelihood-ratio test**:

$$\text{Decide } \omega_1 \text{ if } \frac{p(\mathbf{s}|\omega_1)}{p(\mathbf{s}|\omega_0)} \geq \eta$$

=> For a specified FAR one can select a threshold η such that the above decision rule maximizes the genuine accept rate (GAR)

DONE?

AGAIN NO! … Why?

Optimality of the likelihood-ratio test is guaranteed only when the underlying match score densities are known

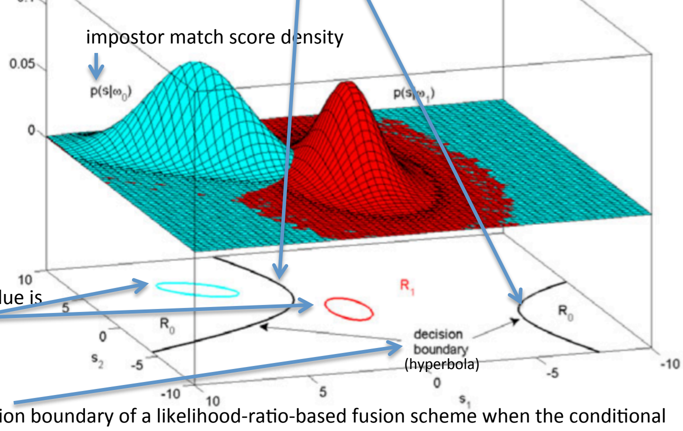
In practice, only a finite set of genuine and impostor match scores are available for training, so the densities must be **reliably estimated** from this training data before applying the likelihood-ratio test >>>>>

Density Estimation

- Parametric density estimation
  - Form of the density function (e.g. Gaussian) is assumed to be known and only the parameters of this density function (e.g. mean and standard deviation) are estimated from the training data

Why not connected boundary?
Because the likelihood-ratio is not monotonic with respect to the match scores (we do not like such anomalies)

impostor match score density



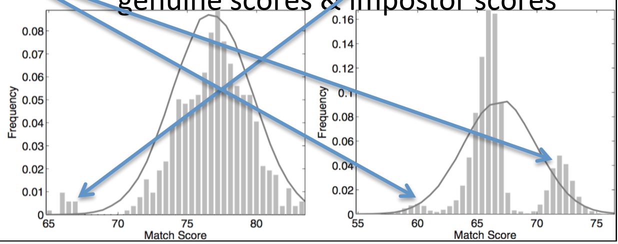
contours of constant density, where the density value is $1/e$ times that at the peak of the distribution

decision boundary of a likelihood-ratio-based fusion scheme when the conditional densities of the two-dimensional score vector $\mathbf{s} = [s_1, s_2]$ are assumed to be Gaussian

Density Estimation

- Parametric density estimation
  - Problems in paradise?
  - Very difficult to choose a specific parametric form as genuine and impostor match score distributions generally have a large **tail** and may have more than one **mode/peak**

Face-G matcher / NIST BSSR1 DB
genuine scores & impostor scores



Density Estimation

- Non-parametric density estimation
- Techniques: density histogram, k-nearest neighbor, kernel density estimator (Parzen window method),… do not assume any standard form for the density function and are essentially data-driven

Density Estimation Simplified

When? If matchers are statistically **independent** under both the hypotheses H_0 and H_1

=> Conditional density can be expressed as the **product** of the marginal conditional densities:

$$p(\mathbf{s}|\omega_j) = p(s_1, s_2, \cdots, s_M|\omega_j) = \prod_{m=1}^{M} p(s_m|\omega_j),\ j = 0, 1$$

New likelihood-ratio-based decision rule: Decide ω_1 if $\prod_{m=1}^{M} \frac{p(s_m|\omega_1)}{p(s_m|\omega_0)} \geq \eta$

In a multimodal biometric system each one of the *M* matchers uses features from a different biometric trait (e.g. face, fingerprint, and hand geometry), which generally tend to be mutually independent

Unless the correlation between matchers is very high (say, over 0.9), degradation in accuracy as a result of independence assumption is not severe => **it is appropriate to use the independence assumption as a rule of thumb!!!**

Transformation-based Score Fusion

- Estimating *M*-dimensional densities is a challenging problem
- Approximate the general minimum error rate of the Bayes decision rule
- Approximations of the posterior probabilities >>>>>

Approx. Achieved Using Different Classifier Combination Rules

- Assumptions:
  - Match scores of different matchers are statistically **independent**
  - Prior probabilities of the genuine and impostor classes are **equal**
- Decision rule: Decide ω_1 if

$$h(P(\omega_1|s_1),P(\omega_1|s_2),\cdots,P(\omega_1|s_M)) > h(P(\omega_0|s_1),P(\omega_0|s_2),\cdots,P(\omega_0|s_M))$$

classifier combination rules function >>>>>

Combination Rule	$h(P(\omega_j\|s_1),P(\omega_j\|s_2),\cdots,P(\omega_j\|s_M)) =$
Product	$\Pi_{m=1}^{M} P(\omega_j\|s_m)$
Sum	$\Sigma_{m=1}^{M} P(\omega_j\|s_m)$
Max	$\max_{m=1}^{M} P(\omega_j\|s_m)$
Min	$\min_{m=1}^{M} P(\omega_j\|s_m)$
Median	$median_{m=1}^{M} P(\omega_j\|s_m)$

Product rule is a direct implication of the assumptions
Limitation: it is sensitive to errors in the estimation of the posterior probabilities

Sum rule is generally more effective
Especially when the estimates of the marginal posterior probabilities are unreliable

The sum rule can be derived from the product rule by assuming that the posterior probabilities do not deviate dramatically from the prior probabilities
The max, min, and median rules can also be obtained by various approximations of the product and sum rules

Approx. Achieved By Transforming the Match Scores into a Common Domain

- Perform **score normalization** (with a monotonic function): min-max, z-score, median, tanh (not sensitive to outliers)
- Once the match scores from different matchers are normalized, classifier combination rules such as the sum, max, and min rules can be applied to obtain the fused match scores
- The corresponding combination rules are referred to as **sum of scores**, **max score**, and **min score** fusion rules, because the normalized match scores may **not** have any direct probabilistic interpretation (normalized scores may not even lie in the interval [0,1]) – product rule is generally not applicable
- Decision rule: Decide ω_1 if $h(g_1(s_1), g_2(s_2), \cdots, g_M(s_M)) > \tau$

Score-level Fusion in Identification Mode

- Key difference:
  - In a verification system the score data is in the form of a **vector** $\mathbf{s} = [s_1, s_2, \cdots, s_M]$
  - Score data in an identification system is in the form of a $N \times M$ **matrix** $\mathbf{S} = [s_{n,m}]$, where $s_{n,m}$ is the match score output by the *m-th* matcher corresponding to the *n-th* identity
- With minor modifications, many of the decision rules designed for the verification scenario can also be extended to the identification mode

Rank-level Fusion

- When a biometric system operates in the **identification** mode, the output of the system can be viewed as a ranking of the enrolled identities
- **Consolidate** all the ranks output by the individual biometric subsystems in order to derive a consensus rank for each identity
  - Ranks reveal less information than match scores
  - Unlike match scores, the rankings output by multiple biometric systems are comparable => no normalization is needed

- Let $\mathbf{R} = [r_{n,m}]$ be the rank matrix in a multibiometric system, where $r_{n,m}$ is the rank assigned to identity I_n by the m-th matcher
- Let $\hat{r}_n$ be a statistic computed for user I_n such that the user with the lowest value of $\hat{r}$ is assigned the highest consensus (or reordered) rank   >>>>>

Highest Rank Method

- Each user is assigned the highest rank (**minimum** r value) as computed by different matchers
- Statistic for user I_n is: $\hat{r}_n = \min_{m=1}^{M} r_{n,m}$
- Useful only when the number of users is large compared to the number of matchers
- It can utilize the strength of each matcher effectively

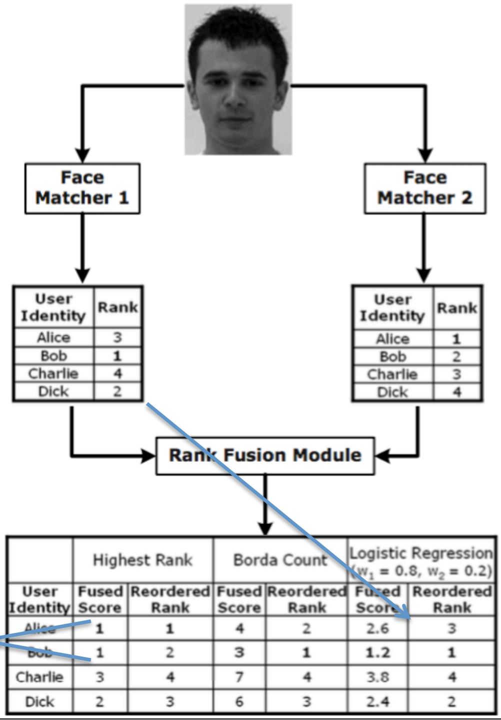
Borda Count Method

- Uses the **sum** of the ranks assigned by the individual matchers
- Statistic for user I_n is: $$\hat{r}_n = \sum_{m=1}^{M} r_{n,m}$$
- Measure of the degree of agreement among the different matchers
- Assumptions:
  - ranks assigned to the users by the matchers are statistically independent
  - all the matchers perform equally well

Logistic Regression Method

- Generalization of the Borda count method where a **weighted sum** of the individual ranks is calculated
- Statistic for user I_n is: $$\hat{r}_n = \sum_{m=1}^{M} w_m r_{n,m}$$
- Weights are determined by logistic regression
- Useful when the different biometric matchers have significant differences in their accuracies
- Requires a training phase

Illustration (two algorithms, four users):



User Identity	Rank
Alice	3
Bob	1
Charlie	4
Dick	2

User Identity	Rank
Alice	1
Bob	2
Charlie	3
Dick	4

	Highest Rank		Borda Count		Logistic Regression ($w_1 = 0.8$, $w_2 = 0.2$)	
User Identity	Fused Score	Reordered Rank	Fused Score	Reordered Rank	Fused Score	Reordered Rank
Alice	1	1	4	2	2.6	3
Bob	1	2	3	1	1.2	1
Charlie	3	4	7	4	3.8	4
Dick	2	3	6	3	2.4	2

breaking the ties randomly ;-)

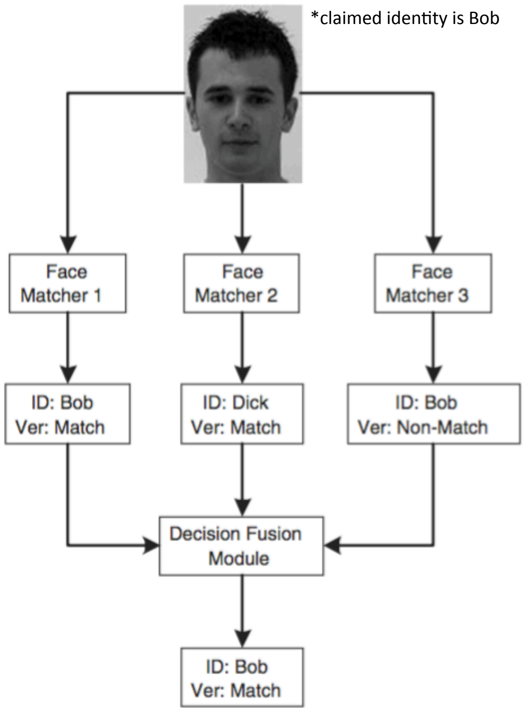
Decision-level Fusion

- Abstract/decision level
- Commercial off-the-shelf (COTS) biometric matchers provide access only to the final recognition decision
- Methods: AND/OR rules (rarely used in practical), majority voting (most common approach), weighted majority voting, Bayesian decision fusion, the Dempster-Shafer theory of evidence, behavior knowledge space,...

Majority Voting

*claimed identity is Bob

- Input biometric sample is assigned to that class on which a **majority of the matchers agree (half)**
- Assumes that all the matchers perform equally well
- No apriori knowledge about the matchers is needed
- No training is required to come up with the final decision



We should stop somewhere …

YOU ARRIVED AT THE END ;-)