Authentication

NISNet Winter School
Finse
23-27 May 2011

Patrick Bours
Norwegian Information Security Laboratory (NISlab)
Norwegian Biometric Laboratory (NBL)
Høgskolen i Gjøvik (HIG)

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
    a. Introduction
    b. Biometric Evaluation
    c. Gait recognition
    d. Keystroke Dynamics
5. Conclusions

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
    a. Introduction
    b. Biometric Evaluation
    c. Gait recognition
    d. Keystroke Dynamics
5. Conclusions

Who am I?

- MSc and PhD in Discrete Mathematics from Eindhoven University of Technology
  - Topic: Discrete Mathematics

- 10 years senior policy maker in cryptology at the Netherlands National Communication Security Agency (NLNCSA)

Who am I?

- Started at HiG in July 2005
  - 7/2005-6/2008: PostDoc
    Authentication in a health service environment
  - 7/2008-now: Ass.Prof.
  - 7/2009-now: Head of NISlab
  - Since 2006 teaching course in Authentication
    First at MSc level
    Now also at PhD level

Who am I?

- Expert in behavioural biometrics, in particular:
  - Gait Recognition
  - Keystroke Dynamics

- Supervision of
  - 1 PhD student in Gait Recognition
  - Many MSc students in Gait Recognition and Keystroke Dynamics
  - Hopefully soon 2 more PhD students (1 in Gait Recognition and 1 in Keystroke Dynamics)

This presentation is...

- … a condensed version of the Authentication Course
- But it is hard to press 6 weeks of teaching into 3 hours
  - Specially if some topics are treated more extensively

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
   a. Introduction
   b. Biometric Evaluation
   c. Gait recognition
   d. Keystroke Dynamics
5. Conclusions

What is authentication?

- Wikipedia:
  Authentication is the act of confirming the truth of an attribute of a datum or entity. This might involve confirming the identity of a person.

What is authentication?

- Webopedia:
  The process of **identifying an individual**, usually based on a **username and password**. In security systems, authentication is distinct from authorization, which is the process of giving individuals access to system objects based on their identity. **Authentication merely ensures that the individual is who he or she claims to be**, but says nothing about the access rights of the individual.

Authentication versus identification

- Authentication:
  - By authentication we mean verification of a claimed identity
  - For example: username (claimed identity) and password (proof of ownership)
- Identification
  - By identification we mean establishing an identity
  - For example recognizing friends by faces or voice

Authentication versus identification

- Authentication:
  - Identity is provided
  - Is he/she really who he/she claims to be?
  - One-on-one verification

- Identification
  - No identity is provided
  - Who is he/she?
  - One-to-many verification

General authentication process

- Two stages:
  1. **Enrollment (1 time):**
     User needs to be “entered” into the system
     UserID and “proof” is stored in a database
  2. **Authentication (many times):**
     User provides UserID and “proof” to verifier
     Verifier can check if “proof” in database and provided “proof” match
     In case of match: Access granted
     In case of no match: Access denied

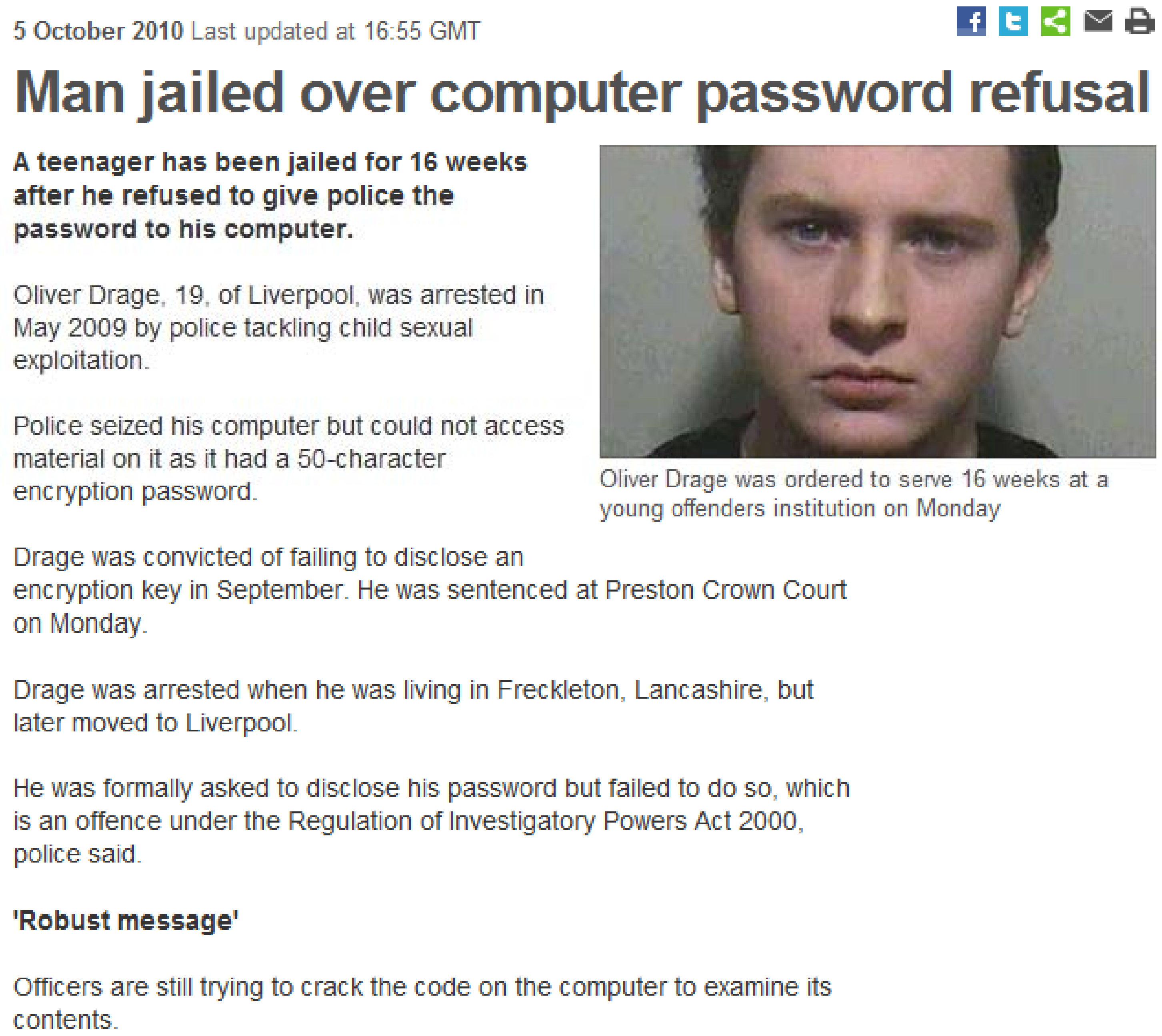
Authentication factors

- Three different authentication factors:

1. Something we **know**
   Secret that only you know, e.g. password, PIN code
2. Something we **have**
   Something only you posses, e.g. token, smartcard, key
3. Something you **are**
   Some biometric property, e.g. fingerprint, signature, face, iris,…

Positive versus negative recognition

- Positive recognition:
  To prevent multiple people from using the same identity
- Negative recognition:
  To prevent one person from using multiple identities
  Only possible with biometrics, we cannot prove that we **do not** know password



5 October 2010 Last updated at 16:55 GMT

Man jailed over computer password refusal

A teenager has been jailed for 16 weeks after he refused to give police the password to his computer.

Oliver Drage was ordered to serve 16 weeks at a young offenders institution on Monday

Oliver Drage, 19, of Liverpool, was arrested in May 2009 by police tackling child sexual exploitation.

Police seized his computer but could not access material on it as it had a 50-character encryption password.

Drage was convicted of failing to disclose an encryption key in September. He was sentenced at Preston Crown Court on Monday.

Drage was arrested when he was living in Freckleton, Lancashire, but later moved to Liverpool.

He was formally asked to disclose his password but failed to do so, which is an offence under the Regulation of Investigatory Powers Act 2000, police said.

'Robust message'

Officers are still trying to crack the code on the computer to examine its contents.

Static versus continuous authentication

- Static Authentication:
  - Authentication process at the beginning of a session
  - Using (mostly) fixed information
  - “Binary” decision: access granted or denied

- Continuous Authentication
  - Checking genuineness of current user during a session
  - Generally not using fixed information
  - Can be Multi Level Security

What authentication method is best?

- Username / password?
  - Easy to use, easy to break if not used correctly, can be forgotten
- Token?
  - Easy to use, expensive in use and for replacement, can be forgotten
- Biometrics?
  - Easy to use, expensive equipment, difficult to replace, can not be forgotten

What authentication method is best?

- Use of password to get access to building?
- Use of DNA to get access to PC?
- Use of retina scan for access to parking lot?

- Use of voice recognition for mobile banking?
- Use of contactless token for access to building?
- Use of fingerprint to get access to PC?

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
   a. Introduction
   b. Biometric Evaluation
   c. Gait recognition
   d. Keystroke Dynamics
5. Conclusions

What is a “password”?

- Password
  - A string of characters: A,B,C,…,d,e,f,…,1,2,3,…,!,”,@,…
- PIN code
  - A string of numbers
- Passphrase:
  - A sentence

What is a “password”?

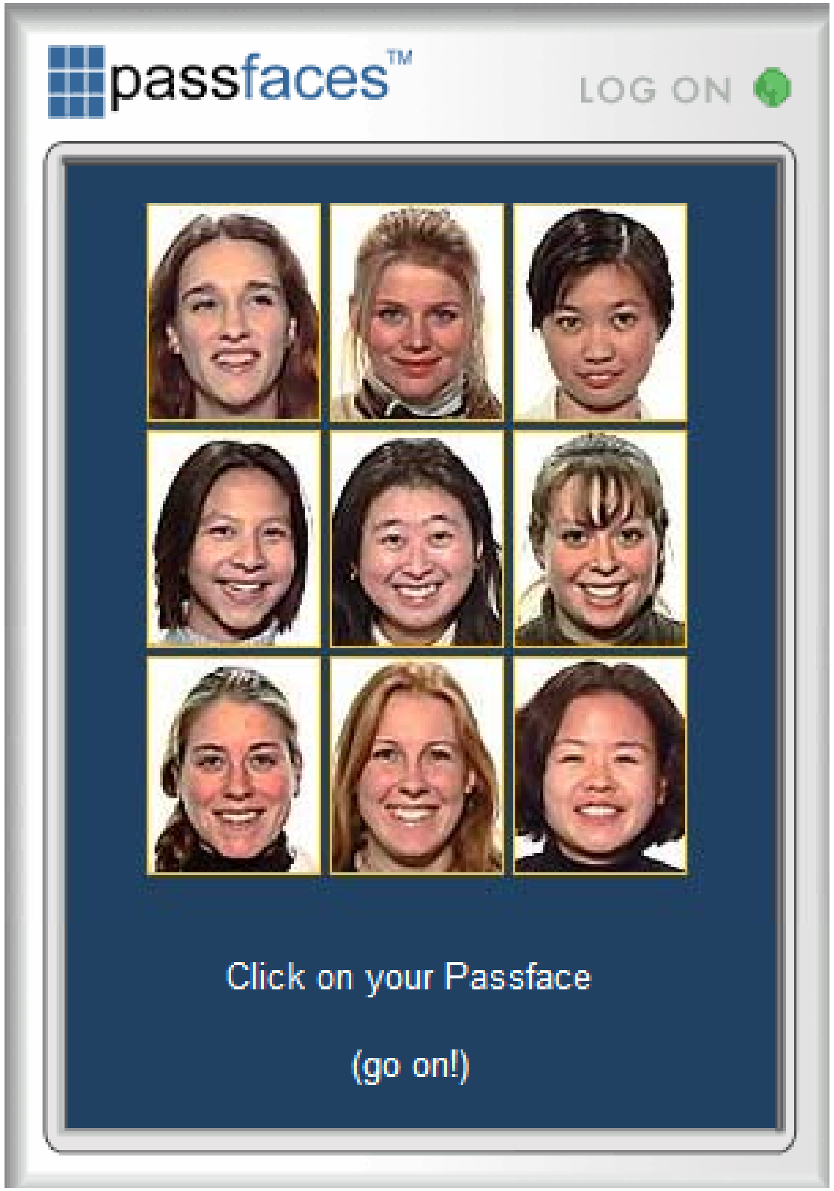
- Associate or cognitive passwords
  - Answers to questions
  - Associate, cue words
    black:white , strawberry:blueberry , dad:mum , day:night
  - Cognitive
    What is your second name? How many cats do you have? Which food do you like best?
- Pass face, pass images
  - It is easier to recognize then to remember

Pass faces / pass images

- **Enrollment:**
  Remember a set of selected or given images
- **Authentication:**
  Enrollment images are hidden within set of images
  Multiple challenges
  Size of challenge can vary
  Number of correct images per challenge can vary

Passfaces

- Online demo
- 3x3 grid, 1 face per grid
- 3 grids
- Faces given, not freely selectable
- Enrolled in 2006
- Authentication once or twice per year
- Never fails!



Password space

- S is set of all passwords
- Size of S is s
- Examples:
  - 4 digit PIN code: $s = 10^4$
  - 6 character password: ??
    - $s = 26^6$
    - $s = 52^6$
    - $s = 62^6$
    - $s = 94^6$
- Need to define question correctly!

Entropy

- Entropy is a measurement of randomness
- Describes the number of bits needed to describe all the members of S
- Entropy directly related to the amount of work in a brute-force attack
- In formula:
  - $h = \log_2(s)$
- Assumption: all passwords are equally likely

Entropy

- Assume $S = \{S_1, S_2, \ldots, S_s\}$
- Probability for password S_i is p_i
- Sum of p_i values equals 1!

- $h = -p_1\log_2(p_1) - p_2\log_2(p_2) - \ldots - p_s\log_2(p_s)$

- Exercise: prove that two definitions of entropy are consistent

Example entropy

- 4 digit PIN code
  - Randomly assigned: h = 13.3
  - Half of people choose a “date like” PIN code
    366 possible choices for half of people
    9634 possible choice for other half
    Entropy: h = 11.9 (exercise)
    Approximately 3 times faster to break

What is a good password?

- **Hard to guess:** do not use names or other personal information
- **Easy to remember:** No need to write down or share

- Are people capable to create good passwords?

Are passwords all that??

- 32M passwords stolen from social networking website RockYou
  - Almost 50% has at most 7 characters
  - Almost 60% uses 1 type of characters
  - Only 3.8% uses special characters
  - Nearly 50% uses names, slang words, dictionary words or trivial passwords
- Password top 3:

  123456

  12345

  123456789

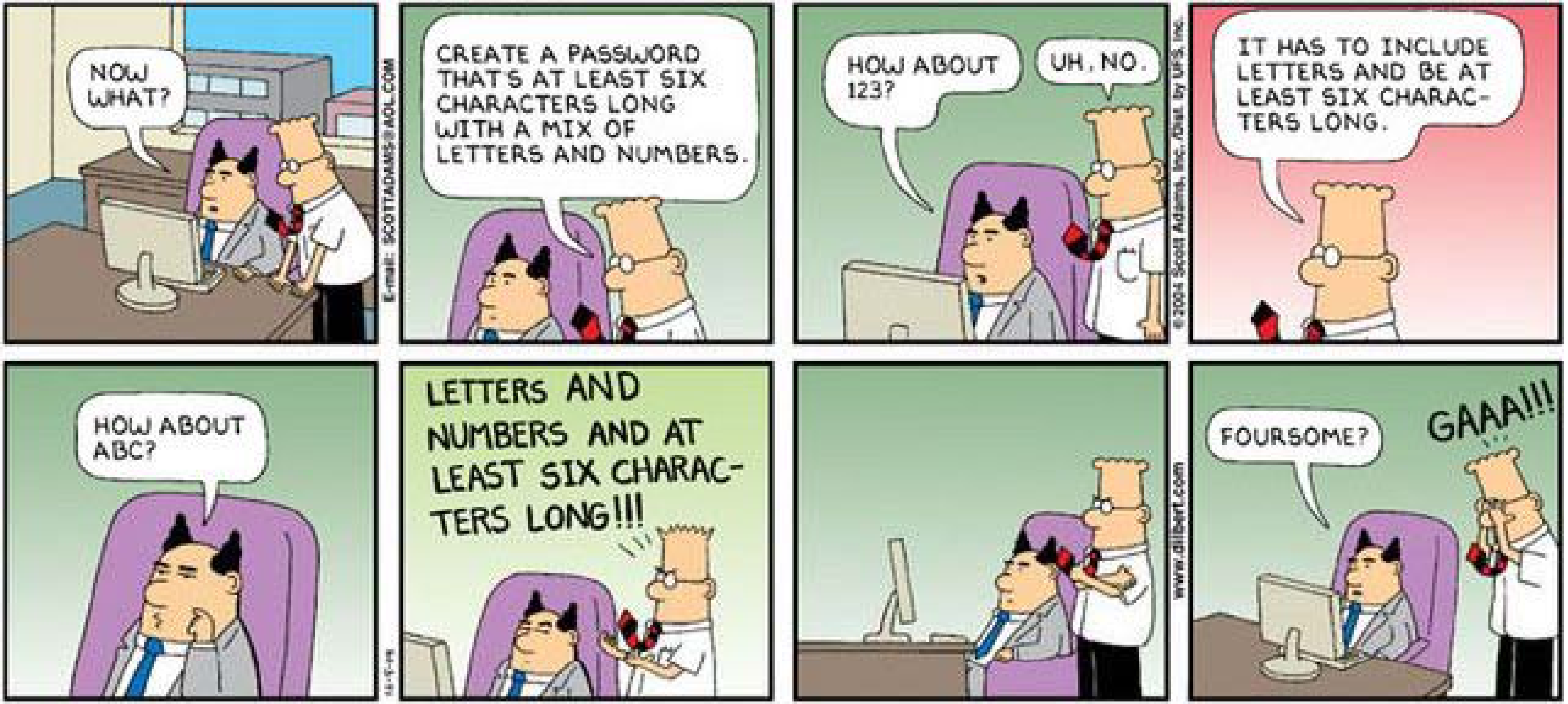
Are passwords all that??

- Password top 10:

  123456, 12345, 123456789, Password, iloveyou, princess, rockyou, 1234567, 12345678, abc123
- Password top 11-20:

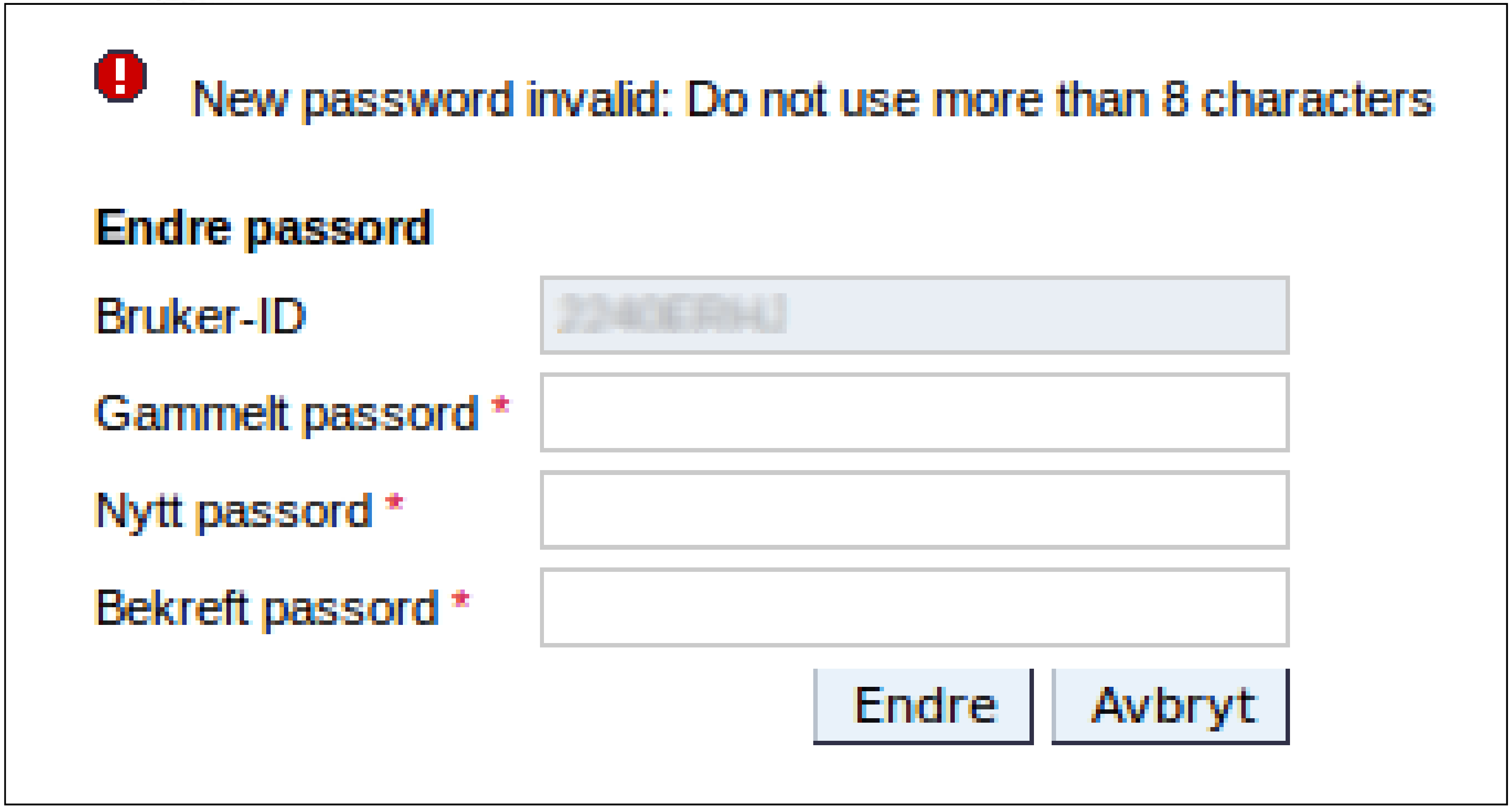
  Nicole, Daniel, babygirl, monkey, Jessica, Lovely, michael, Ashley, 654321, Qwerty

Are passwords all that??


NOW WHAT?
CREATE A PASSWORD THAT'S AT LEAST SIX CHARACTERS LONG WITH A MIX OF LETTERS AND NUMBERS.
HOW ABOUT 123?
UH, NO.
IT HAS TO INCLUDE LETTERS AND BE AT LEAST SIX CHARAC-TERS LONG.
HOW ABOUT ABC?
LETTERS AND NUMBERS AND AT LEAST SIX CHARAC-TERS LONG!!!
FOURSOME?
GAAA!!!
E-mail: SCOTTADAMS@AOL.COM
© 2004 Scott Adams, Inc./Dist. by UFS, Inc.
www.dilbert.com

- People are not capable of creating secure passwords!
  - Or they forget them often

Are passwords all that??



New password invalid: Do not use more than 8 characters

Endre passord

Bruker-ID	[redacted]
Gammelt passord *	
Nytt passord *	
Bekreft passord *	

- Sometimes it is the system!

To NOT do list

- PW based on user account name or personal data
- PW based on dictionary word
  - Even when reversed, replaced letters by numbers or control characters, capitals
  - Even when adding “ing” or “s” or number
- PW which form patterns on keyboard
- PW which consists only of numbers

To do list

- At least 8 characters long
- Mixture of letters, capitals, numbers and/or special characters (at least 3 of these)
- “Welcome42” is
  - Long enough
  - Has characters from three groups
  - And is NOT secure!
- To NOT do list is equally important as To do list
- Use “random” passwords

The password problem

- We have a limited memory:
  - We can only remember 7±2 totally random symbols

- Even more problems:
  - We have many passwords
  - We need to change passwords regularly

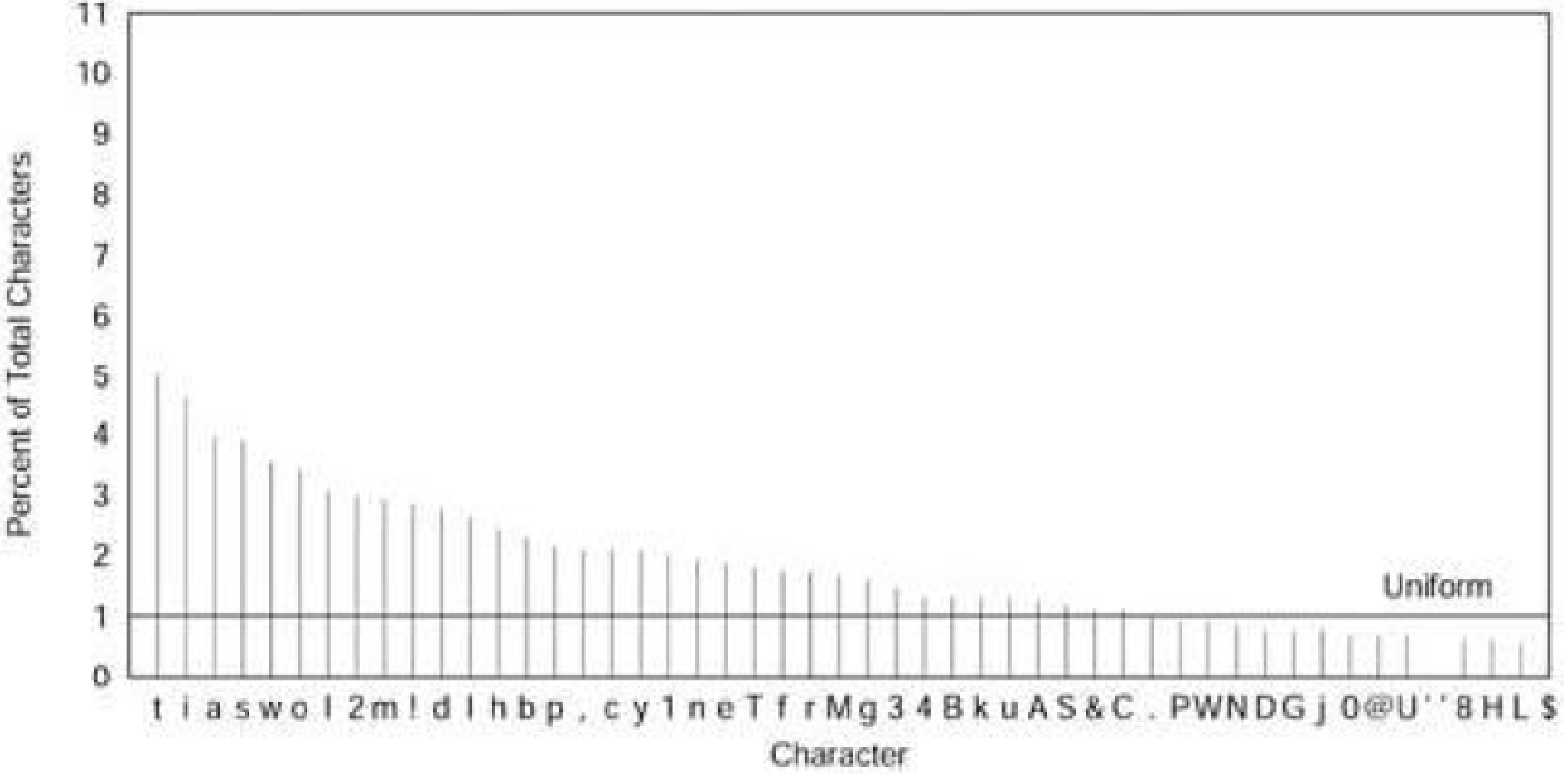
How can we solve the password problem?

- Use of passphrases:
  - Yesterday I watched a nice program on television
  - YIwanpot or Y1w@np0t
- Use events on news or personal events when forced to change regularly

Be aware?

- Don’t use “known” sentences
  - It’s elementary, my dear Watson
  - Luke, I’m your father
  - Beam me up, Scotty
  - Every breath you take and every move you make
  - First, it is slightly cheaper; and secondly it has the words DON’T PANIC inscribed in large friendly letters on its cover
  - The Answer to the Great Question, of Life, the Universe and Everything

Character distribution for known passphrase sentences



What more can we do?

- A good password is easy to remember
- Hard to guess
- Not based on a dictionary word

- What if we use “encryption” of a dictionary word?

What more can we do?

- Shift every character a fixed number of positions in alpabet (Ceasar cipher)
  - a->d , b->e , c-> f , …

- Shift hands on keyboard
  - a->s , b->n , c->v , p->å , m->, , …

What more can we do

- Start with a dictionary word (easy to remember!)
- Find an easy to remember “encryption method”
- Result is reproducible but looks random
  - Welcome42 -> Zhofrph75
  - Welcome42 -> Eøvp,r53

What more can we do

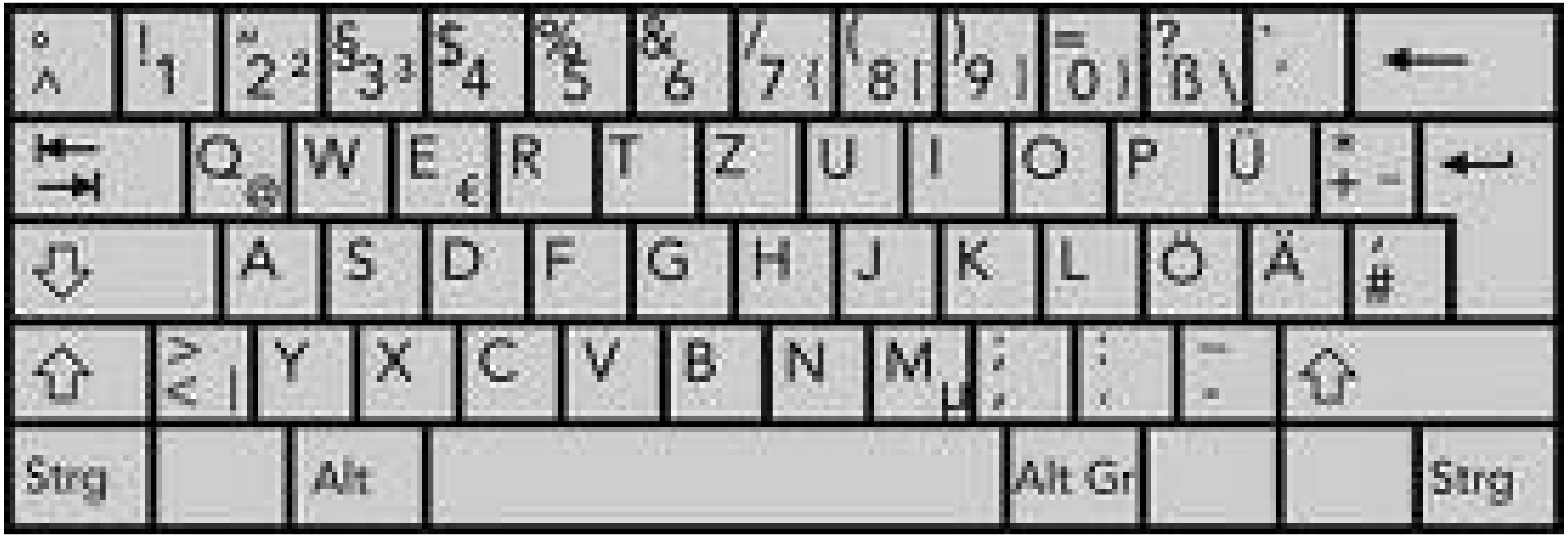
- PlkiuH790 seems pretty random

- Think fishy

  ><>

What more can we do

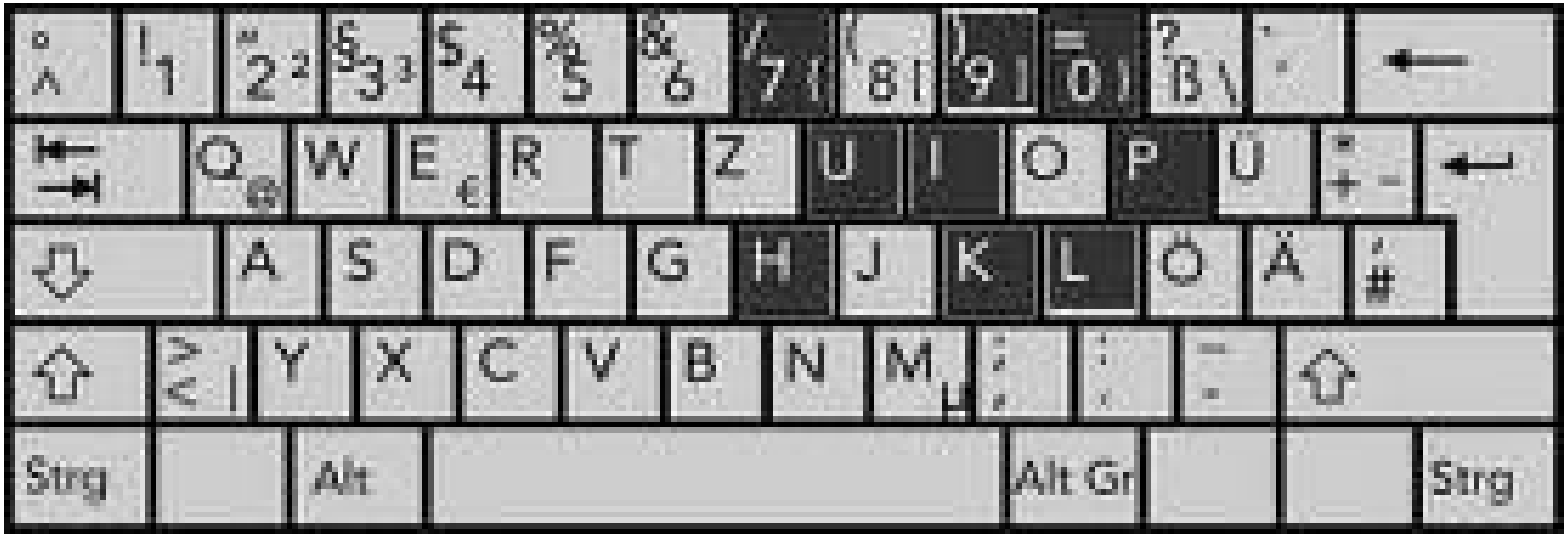
- Remember a pattern and a starting position
- Recreate password



° ^ ! 1 " 2 ² § 3 ³ $ 4 % 5 & 6 / 7 { (8 [ ) 9] = 0 } ? ß \ ` ´ ←
Q @ W E € R T Z U I O P Ü * + ~ ↵
A S D F G H J K L Ö Ä ' #
> < | Y X C V B N M µ ; , : . _ -
Strg Alt Alt Gr Strg

What more can we do

- Remember a pattern and a starting position
  - Fishy P
- Recreate password


° ^ ! 1 " 2 ² § 3 ³ $ 4 % 5 & 6 / 7 { (8 [ ) 9] = 0 } ? ß \ ` ´ ←
Q @ W E € R T Z U I O P Ü * + ~ ↵
A S D F G H J K L Ö Ä ' #
> < | Y X C V B N M µ ; , : . _ -
Strg Alt Alt Gr Strg

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
   1. Introduction
   2. Biometric Evaluation
   3. Gait recognition
   4. Keystroke Dynamics
5. Conclusions

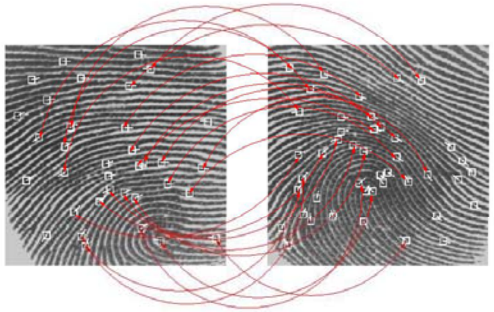
Definition biometrics

- “Biometric Technologies” are **automated** methods of verifying or recognizing the identity of a **living person** based on a **physiological** or **behavioural** characteristic

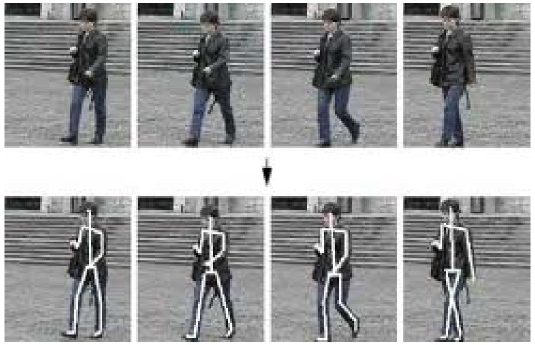
Biometrics

- **Automated:** so not done by humans
- **Living person:** single person and no forensic techniques
- 2 different forms:
  - **Physiological:** Characteristics physically attached to our body
  - **Behavioural:** Things we learned to do in a stable manner

Examples of biometrics



- Physiological:
  - Fingerprint, Face, Hand, Iris, Retina, Thermogram, Vascular Patterns, Ears, DNA, Dental, Odor, Footprints



- Behavioural:
  - Voice, Signature, Gait, Keystroke Dynamics, Mouse Dynamics

More examples of biometrics

- Dental
- DNA
- Ear
- Face
- Fingerprint
- Footprint
- Gait
- Hand geometry
- Iris
- Keystroke Dynamics
- Mouse Dynamics
- Odor
- Palmprint
- Retina
- Signature
- Thermographs
- Veins (finger/palm)
- Voice

Biometrics

- 7 characteristics:
  - Universality
  - Distinctiveness
  - Permanence
  - Collectability
  - Performance
  - Acceptability
  - Circumvention

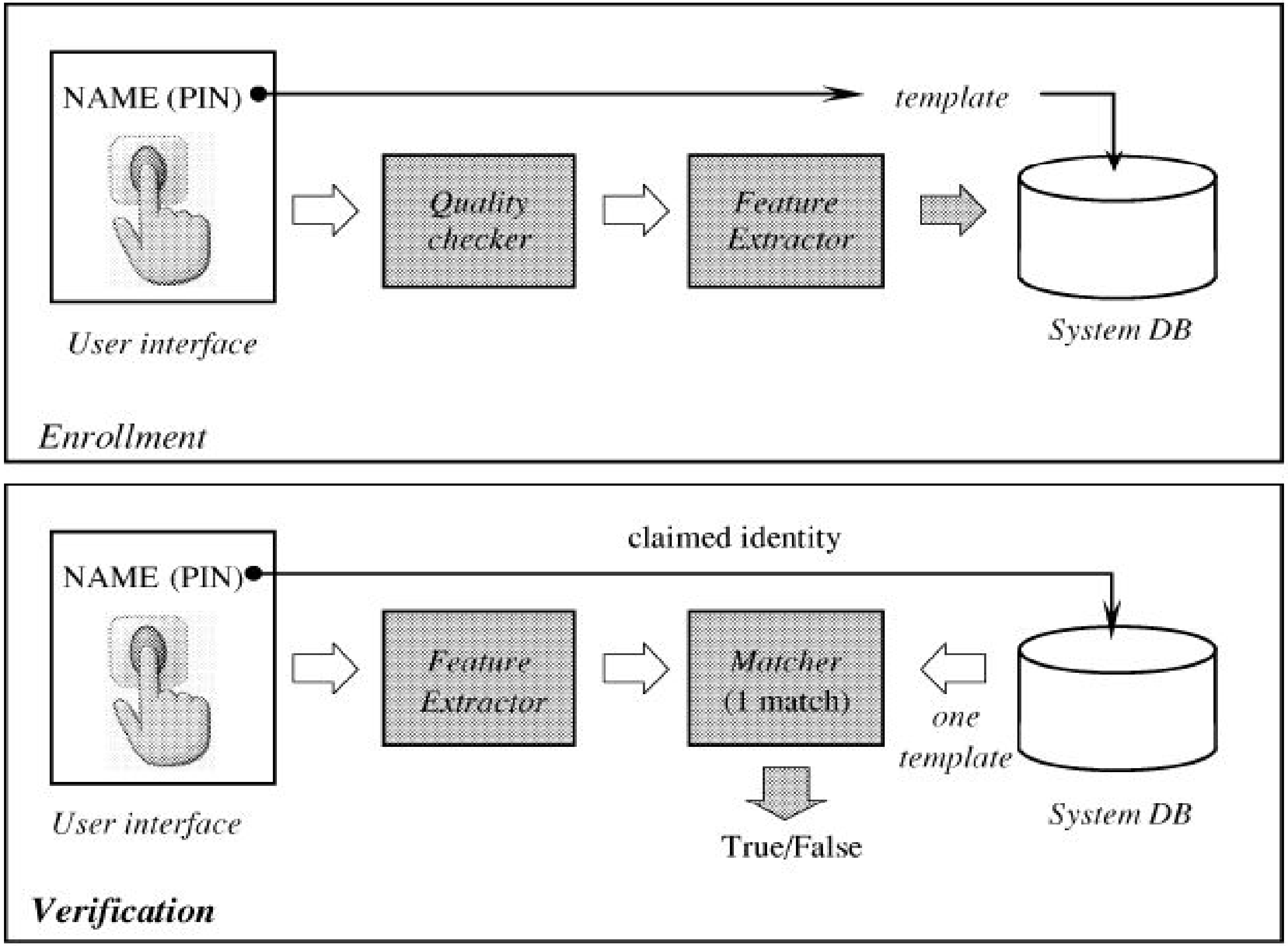
- Which is the most important?

Is biometrics all that?? YES!!

- It binds YOU to your digital identity!!
  - Your own characteristics make YOU to the strongest link!

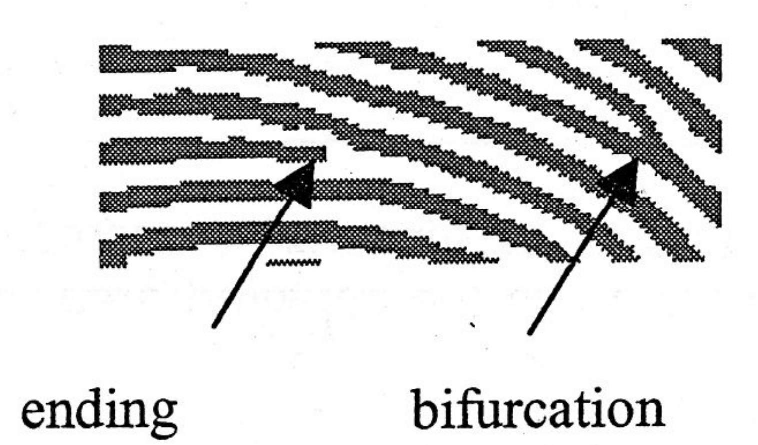
"On the Internet, nobody knows you're a dog."

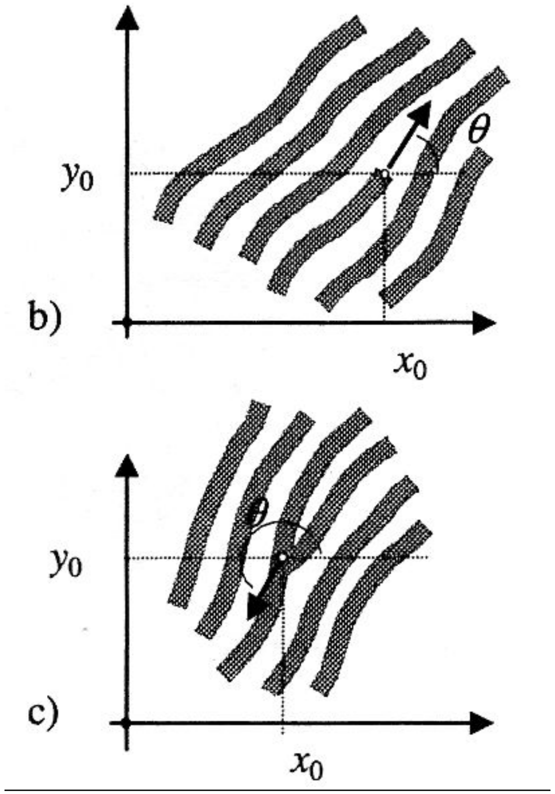
Biometric system



Feature extraction

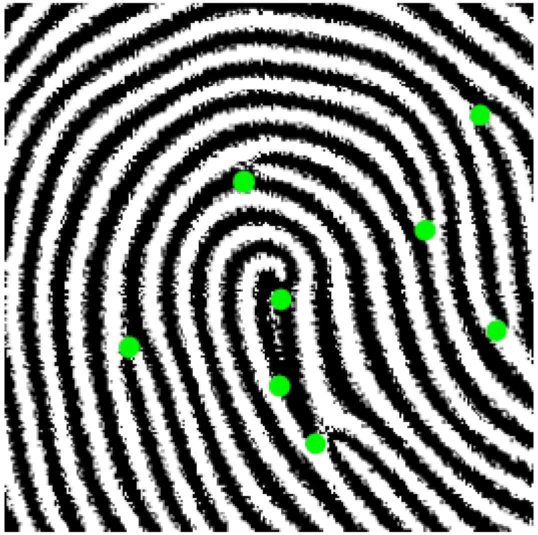
- Captured biometric features are not used directly
- Features are extracted:
  - Fingerprints: minutiae





Feature extraction

- Raw biometric feature is turned into set of numbers
  - Set does not need to have fixed size
  - Set does not need to be ordered

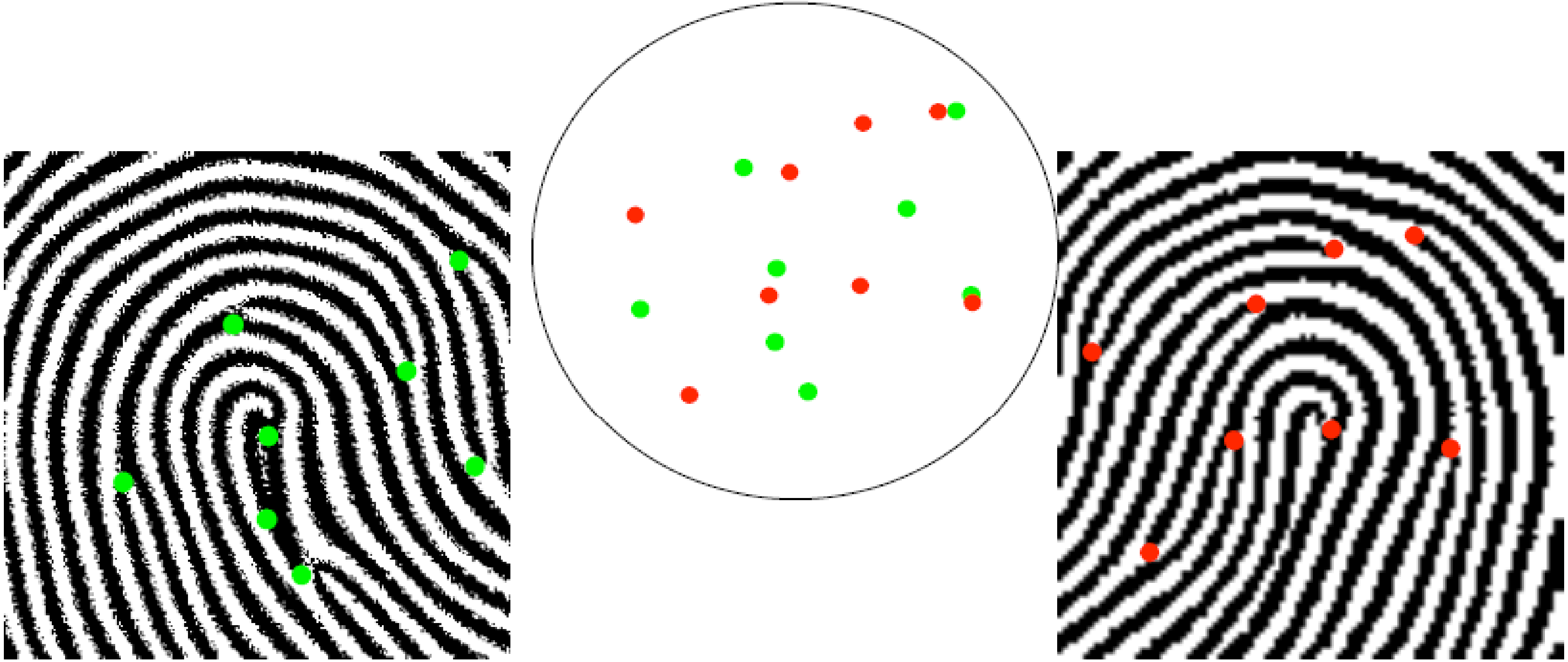


Matcher

- Two biometric features are never exact alike
- Need to find a way to compare template to new input

Matcher

- Red and green dots do “not match”



Distance metric

- Distance metric is function with 2 inputs and 1 output
  - Input: Template and New Input
  - Output: “distance” between Template and New Input
- Small distance:
  Most likely from the same person
- Large distance:
  Most likely from two different persons

Threshold

- What is a large and a small distance?
  - Distance d
- Threshold T
  - Generally T is global, but can be personal

- Small distance: $d \leq T$
- Large distance: $d > T$

- How to calculate d?

Distance metric

- Assumptions:
  x = (x_1, x_2, … , x_n)
  y = (y_1, y_2, … , y_n)

- Note that **x** and **y** have the same length and are ordered
- x_i is comparable to y_i

Distance metric - examples

- Manhattan
  $d_1(\mathbf{x},\mathbf{y}) = \Sigma \mid x_i\text{-}y_i \mid$

- Euclidean
  $d_2(\mathbf{x},\mathbf{y}) = \surd\ (\Sigma (x_i\text{-}y_i)^2)$

- Anything goes
  $d_3(\mathbf{x},\mathbf{y}) = \max \mid x_i\text{-}y_i \mid$
- More later!

Biometrics makes errors

- Sometimes distance d between template and new input from the same person is more than threshold T
- Sometimes it is not possible to capture the biometric features
- Sometimes the quality of the captured biometric features is too low to extract information

Errors

- FMR: False Match Rate
- FNMR: False Non-Match Rate
- FRR: False Reject Rate
- FAR: False Accept Rate
- FTE: Failure to Enroll Rate
- FTC: Failure to Capture Rate
- FTX: Failure to Extract Rate

Errors

- FMR and FNMR are (matching) algorithm errors
- FAR and FRR are system errors

- FAR = FMR*(1-FTA)
- FRR = FTA + FNMR*(1-FTA)
- FTA = FTC + FTX*(1-FTC)

Equal Error Rate

- FMR and FNMR depend on threshold T
  - If T is low: FMR is low and FNMR is high
  - If T is high: FMR is high and FNMR is low

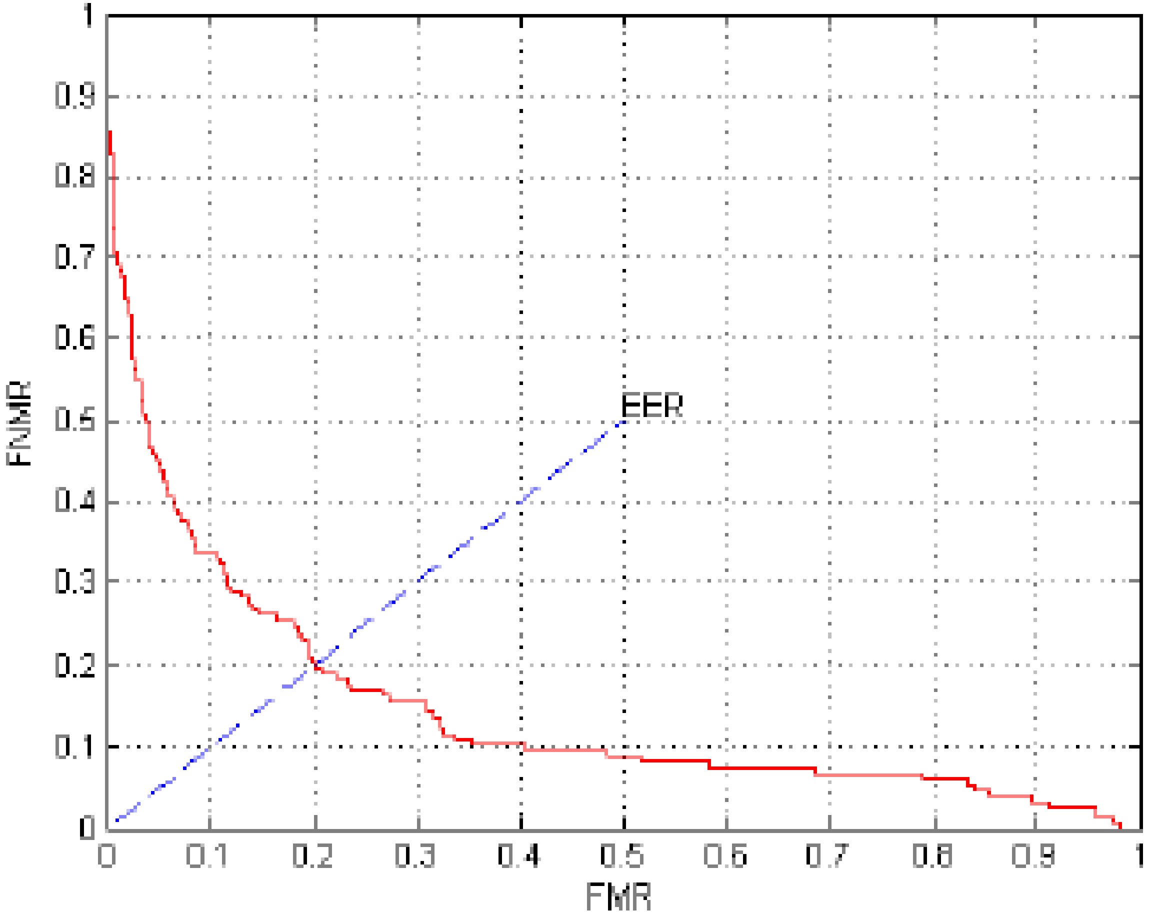
- EER is where FMR and FNMR are equal

Example

	Templ 1	Templ 2	Templ 3	Templ 4	Templ 5
Test 1	**0,182**	0,588	0,435	0,208	0,909
Test 2	0,323	**0,213**	0,286	0,476	0,244
Test 3	0,909	0,625	**0,147**	0,476	1,111
Test 4	0,238	0,294	0,476	**0,256**	0,526
Test 5	0,588	0,454	1,250	0,526	**0,130**

- If T=0.200, then FMR=0/20 and FNMR =2/5
- If T=0.225, then FMR=1/20 and FNMR =1/5
- If T=0.250, then FMR=3/20 and FNMR =1/5

Detection/Decision Error Trade-off (DET) curve



Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
    a. Introduction
    b. Biometric Evaluation
    c. Gait recognition
    d. Keystroke Dynamics
5. Conclusions

What are we going to evaluate?

- Select a biometric feature
- Find an interesting research question
- **Design an experiment**
- Execute the experiment
- **Analyze the data**

Design an experiment

- How to solve the research question?
  - How many participants?
    What is their profile?
    What is their “task”?
  - How much raw biometric data per participant?
    You need data for enrollment and for authentication!
    Check the collected raw data!
    Backup raw data and store in an identifiable way!
  - What extra information from the participants?
  - How do you control the environment?
    Record the environment!

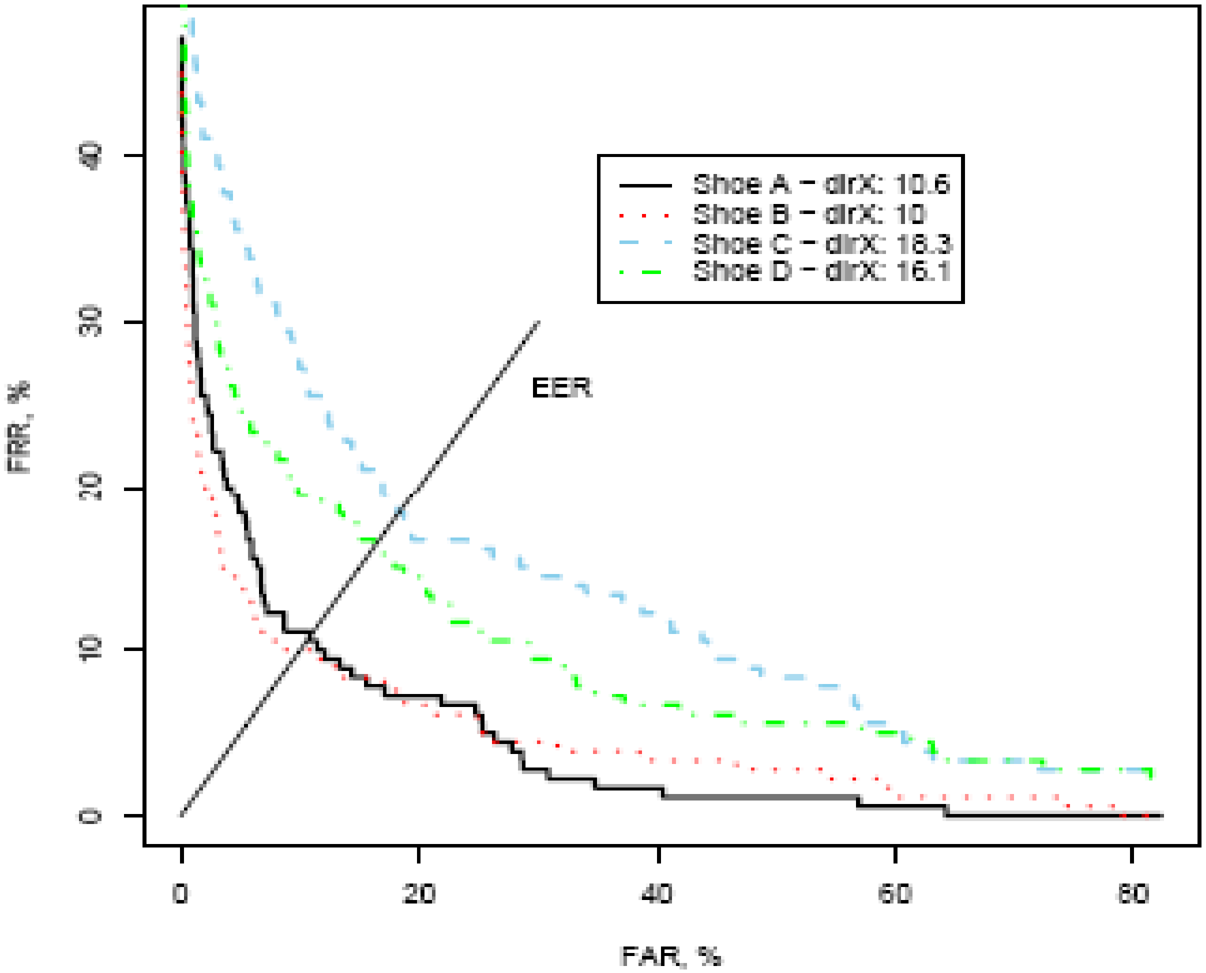
Analyze the data

- Might need pre-processing of raw data
  - BUT ALWAYS KEEP RAW DATA!
- Maybe not all features of the data are relevant
  - Better to leave out then to use too much
- For inspiration “stare” at the data
  - What is “constant” for a person
  - What is “different” between persons
  - Focus both “locally” and “globally”

Analyze the data

- Do NOT expect perfection!!
- Often not predictable if changes will turn out good or bad
  - Change of distance metric, of selected features, of pre-processing steps
  - Play with data and vary settings. Keep good changes and abandon bad changes
- Why do we analyze the data?
  - Find FMR/FNMR, plot DET curve, compare DET curves for various settings, find the best setting

Analyze the data



How to create a DET curve?

- N test persons
- (M+K) inputs per person
- Use K out of (M+K) inputs for template creation (1≤K)
- Use remaining M inputs for testing

How to create a DET curve

- Template: T_i for i=1..N
- Input: $I_{i,j}$ for i=1..N and j=1..M

- “Check” each possible input against each available template
  - N templates
  - N*M inputs
  - N^2*M combinations

False Non Match?

- A person is not recognized as himself
- Test input and template are from the same person!
- N participants and M “own” inputs per participant
  - N*M checks
  - N*M distance values
  - Called: **Genuine Scores**

False Match?

- A person is recognized as somebody else
- Test input and template are from different persons!
- N participants, N-1 “other participants” and M inputs per other participant
  - N*(N-1)*M checks
  - N*(N-1)*M distance values
  - Called: **Impostor Scores**

FMR and FNMR

- FNM: N*M Genuine Scores
- FM: N*(N-1)*M Impostor Scores
- Total: N^2*M!

- FNMR = “number of genuine scores above threshold T” / “total number of genuine scores”

- FMR = “number of impostor scores below threshold T” / “total number of impostor scores”

How to create a DET curve

- Plot FMR against FNMR for various values of the threshold T
- Find EER as the intersection of DET curve and line x=y

Template selection

- Only choose first input from all participants as template and calculate EER
- Choose input 1 from all participants and calculate EER, then input 2 from all participants and calculate EER, etc.
- Choose random input of each participant for EER and calculate EER. Repeat K times

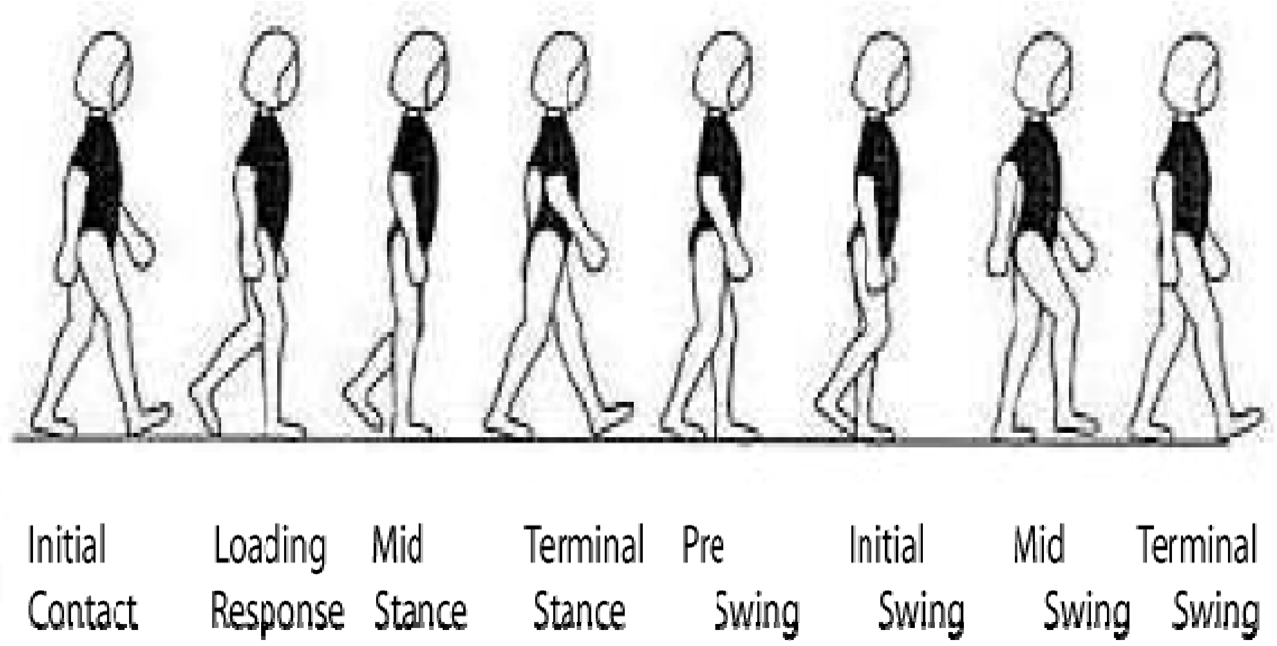
- Average EER, with confidence interval

Program

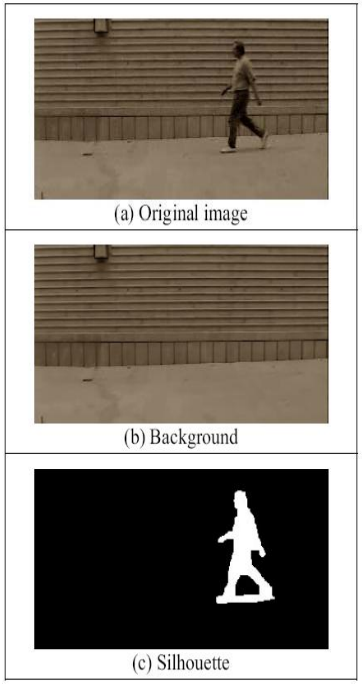
1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
   - a. Introduction
   - b. Biometric Evaluation
   - c. Gait recognition
   - d. Keystroke Dynamics
5. Conclusions

Gait recognition

- Recognizing a person by the way he walks: “Great Juno comes; I know her by her gait” from “The Tempest” by Shakespeare



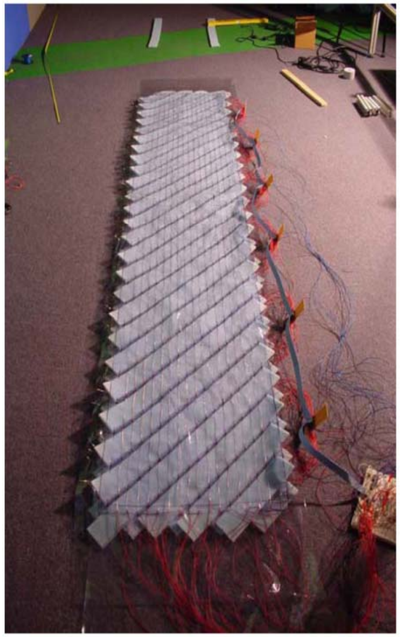
How to do gait recognition?


(a) Original image

(b) Background

(c) Silhouette

- Machine Vision based



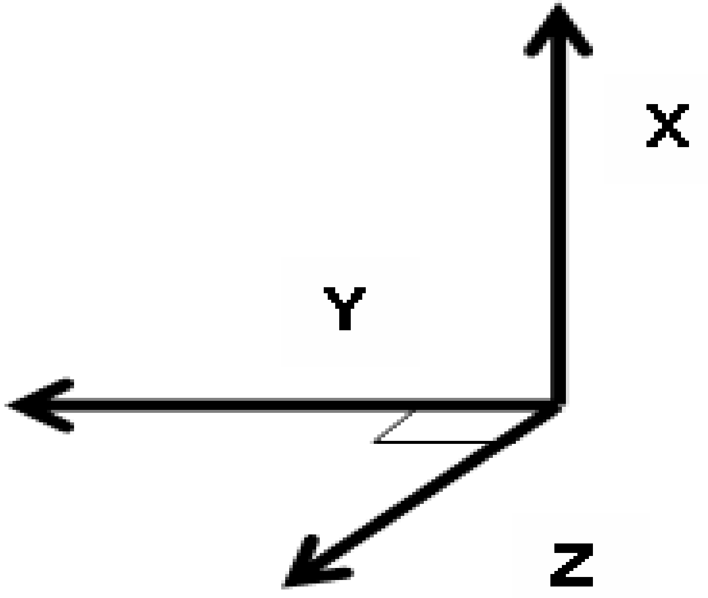
- Floor Sensor based

- Wearable Sensor based

Acceleration data

- Measured in 3 directions
  - Up-down
  - Forward-backward
  - Left-right

- Possible to combine signal
  - $r_i = \sqrt{(x_i^2 + y_i^2 + z_i^2)}$
  - Other combinations are possible



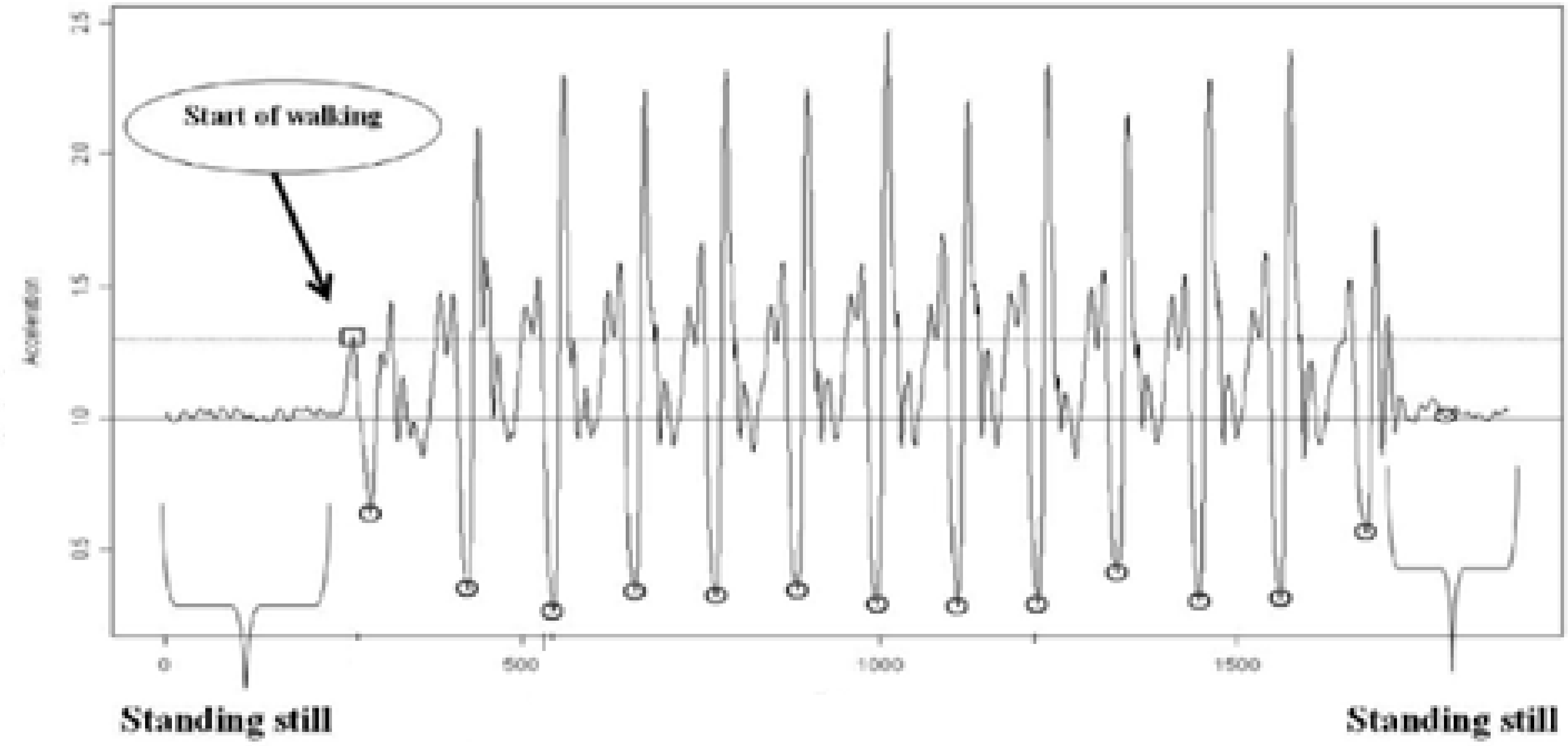
Acceleration data

- Each sample consists of 3 perpendicular directions: (a_x, a_y, a_z)
- Acceleration is measured in m/s^2 or in g
- Generally:
  - 100 samples per second
  - A single “double step” takes approximately 1 second, i.e. 100 samples, or 300 acceleration values

Where to attach the accelerometer?

- Foot/ankle
- Hip
- Hand
- Knee
- Shoulder
- “Free position”, e.g. in trouser pocket

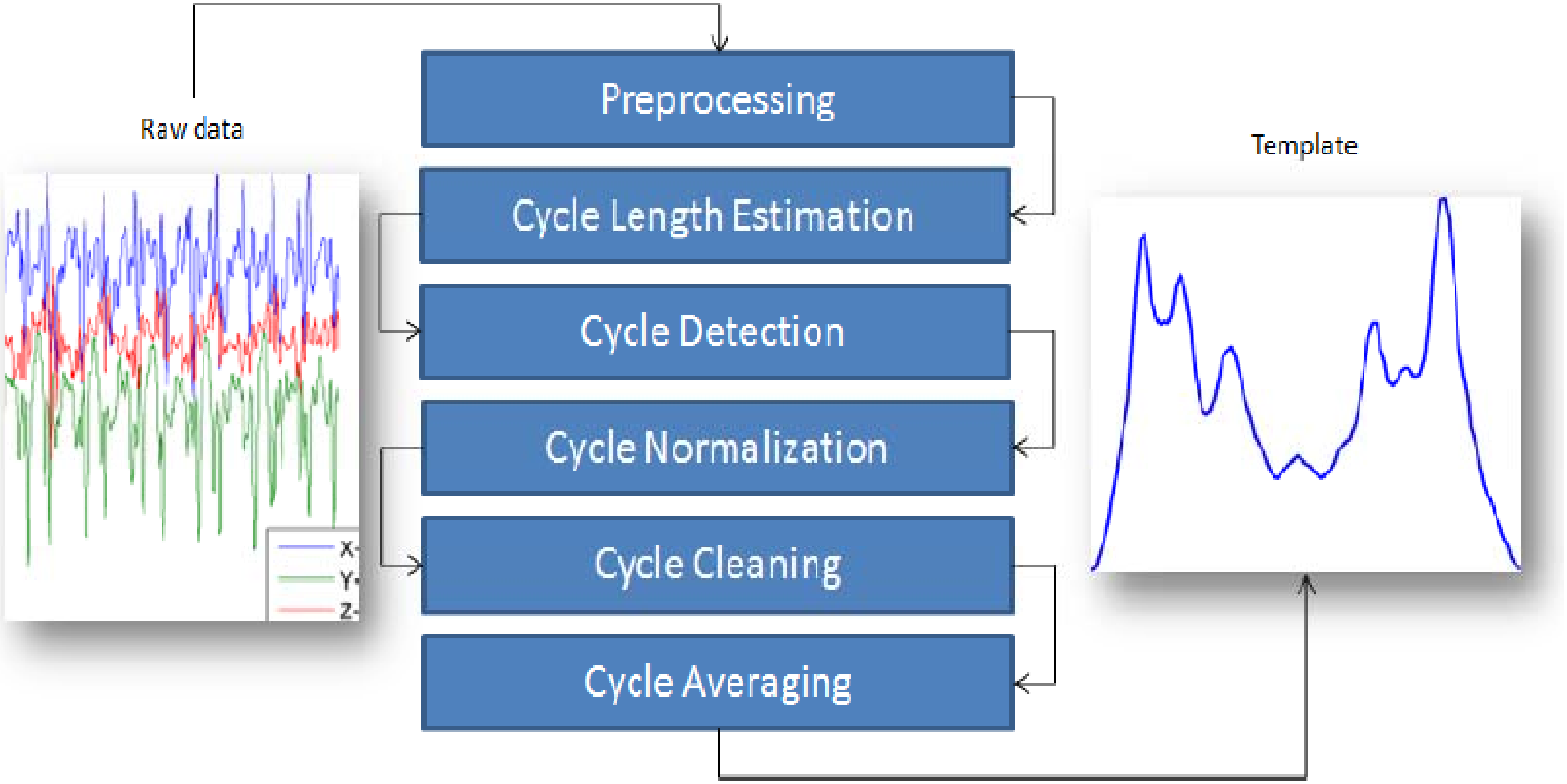
Example of collected data



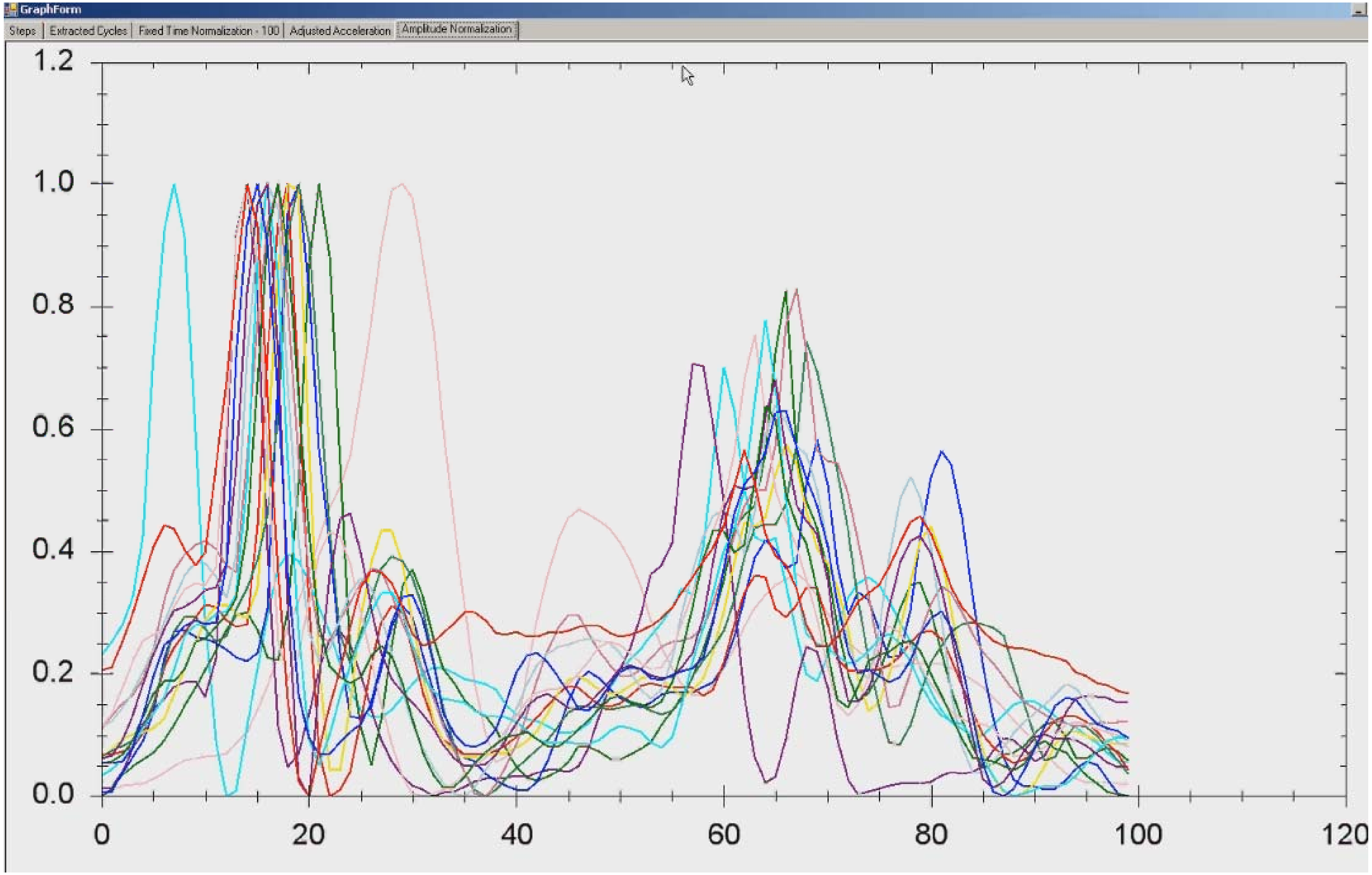
Average Cycle Method

- Often used in analysis
- Consists of 5 steps:
  - Noise reduction
  - Cycle detection
  - Time Normalization
  - Averaging Cycle
  - Comparison

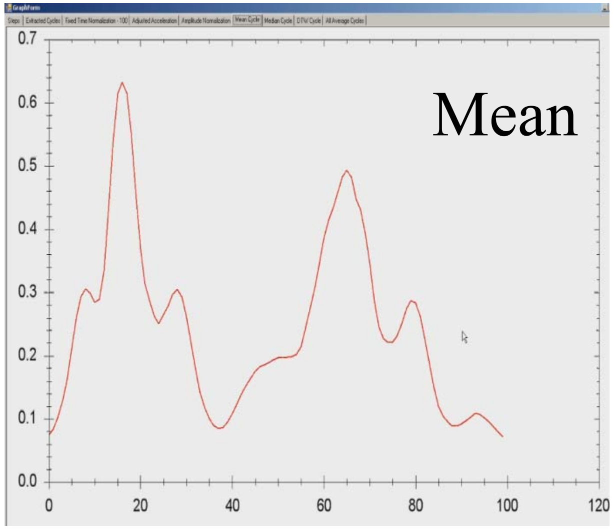
Average Cycle Method

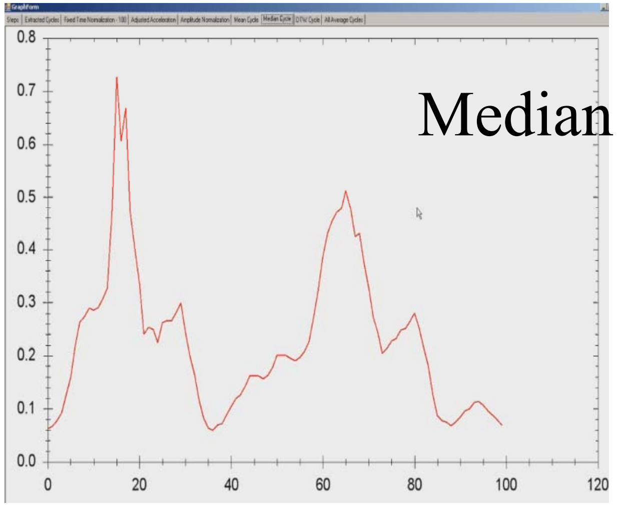


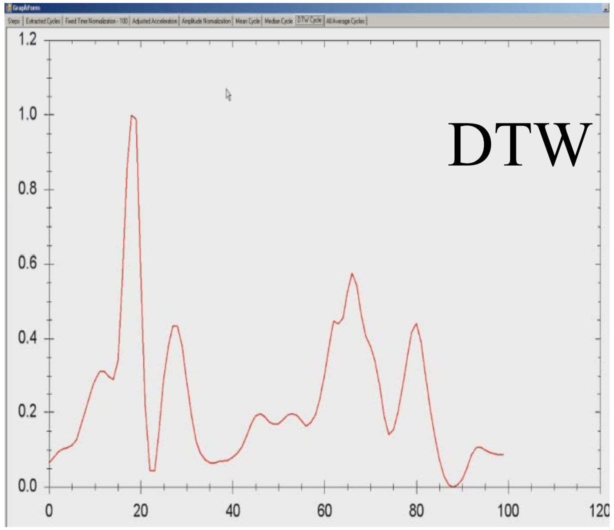
Cycle cleaning

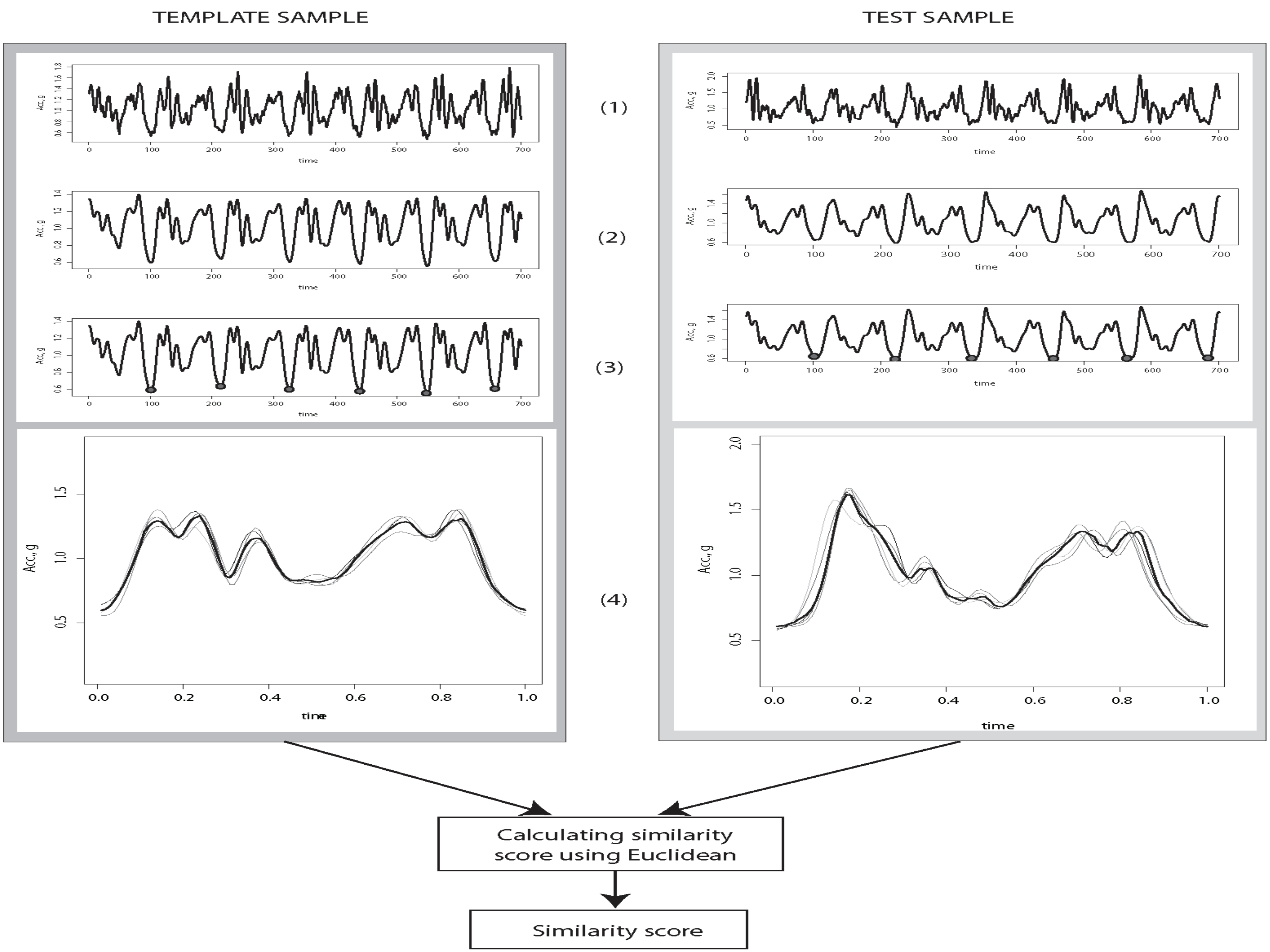


Creating an average cycle









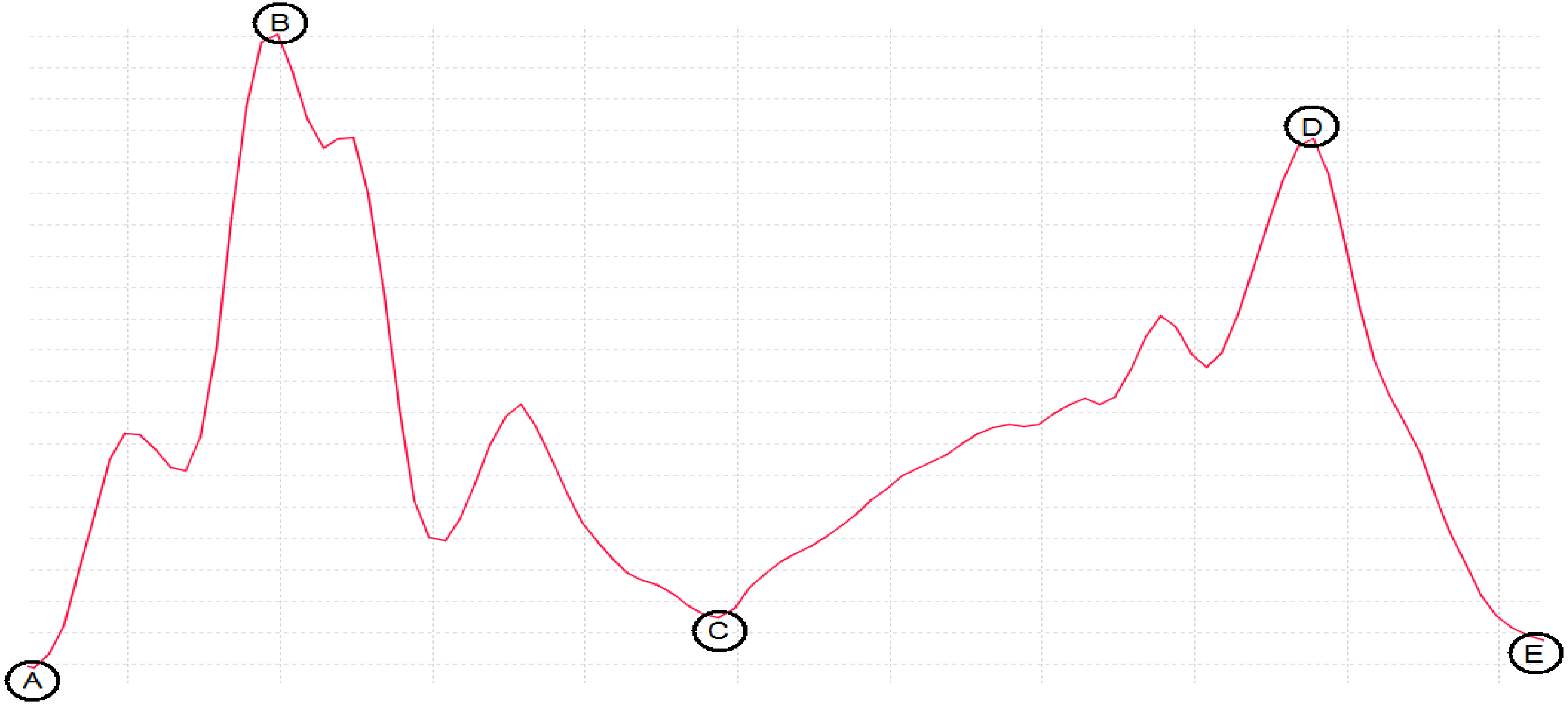
Calculating similarity score using Euclidean

Similarity score

Comparison

- Often used distance metrics:
  - Euclidean and Manhattan distances
  - Dynamic Time Warping distance
    Also called Edit/Levenshtein distance
- Distance metric may include length before cycle normalization
- Mostly on the full signal
  - However some parts might be more interesting than other parts in the signal

Close up of a cycle



Results

Study	% EER	Recognition Rate	Number of participants
Bours and Shrestha	1.68		60
Gafurov	5.0		30
Rong et al.	5.6		21
Derawi et al.	5.7		60
Holien	5.9		60
Heikki et al.	6.4		36
Mantyjarvi et al.	7.0 19.0		36
Vildjiounaite et al.	13.7		31
Huang et al.		96.9	9

Alternative for ACM

- Why average cycles?
- 2 walks w_1 and w_2:
  - $w_1 = (c_1, c_2, \ldots, c_n)$
  - $w_2 = (b_1, b_2, \ldots, c_m)$
- ACM:
  - c = average($c_1, c_2, \ldots, c_n$)
  - b = average($b_1, b_2, \ldots, c_m$)
  - D = distance(w_1,w_2) = distance(b,c)
  - D <= Threshold: w_1 and w_2 from same person
  - D > Threshold: w_1 and w_2 from different persons

Alternative for ACM

- 2 walks w_1 and w_2:
  - $w_1 = (c_1, c_2, \ldots, c_n)$
  - $w_2 = (b_1, b_2, \ldots, c_m)$
  - $d_{i,j}$ = distance(c_i, b_j)
  - D = distance(w_1,w_2) = minimum($d_{i,j}$)
  - D <= Threshold: w_1 and w_2 from same person
  - D > Threshold: w_1 and w_2 from different persons

- Alternatively use maximum, median, mean, 1st quantile, 3rd quantile, …

Future in gait recognition

- Using “commercial” devices
- Different walking styles
- Activity recognition
  - Layered approach
- Use of new techniques (neural networks,…)
- Different age groups (children, elderly,…)
- Development of gait over time

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
    a. Introduction
    b. Biometric Evaluation
    c. Gait recognition
    d. Keystroke Dynamics
5. Conclusions

What is keystroke dynamics?

- Recognition of a person by the way he types on a keyboard
- Known already during World War II – telegraph operators
- Can be easily used for both static and continuous authentication

Static keystroke dynamics

- At the beginning of a session
- Fixed information (mostly username and password)
- Can not detect change of user after log on
- Is unobtrusive
- Relatively easy to process biometric data

Continuous keystroke dynamics

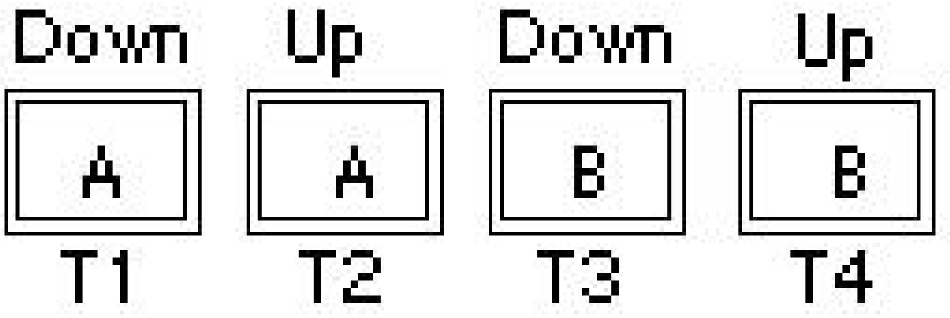
- Monitors typing characteristics during a session
- In general non-fixed text
- Not the same as periodic authentication
- Used to detect change of user
- Can be unobtrusive to the user
- Can provide multi level security

Timing information

- We can measure:
  - When a key is pressed down (Key_down event)
  - When a key is released (Key_up event)

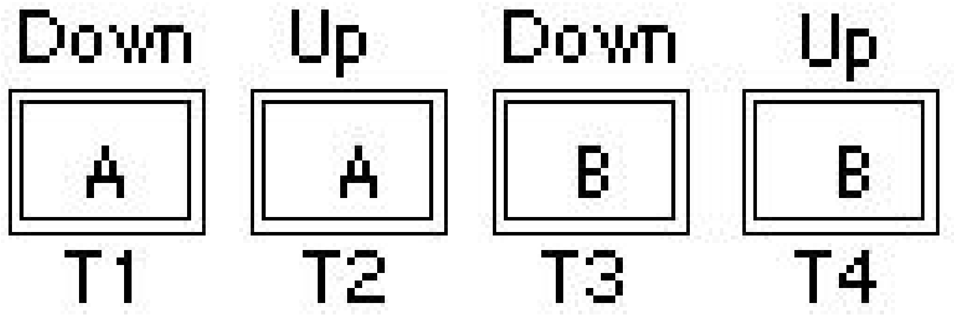
- From this we can calculate:
  - Duration of a key press
  - Latency between two consecutive keys

Duration and Latency



KeyDown	A	T1
KeyUp	A	T2
KeyDown	B	T3
KeyUp	B	T4

Duration and Latency

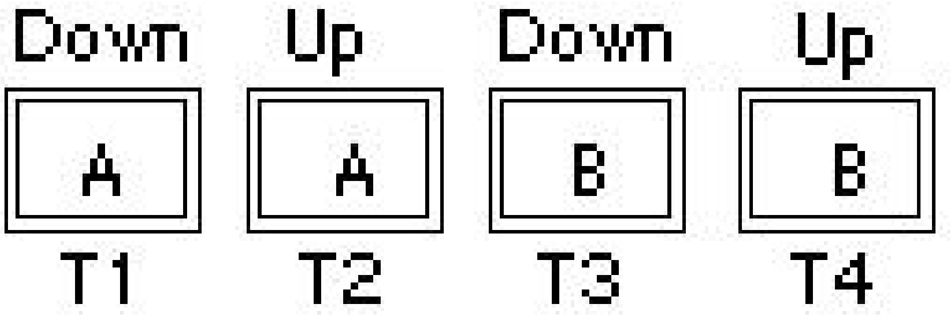


• Duration	A	T2-T1
• Latency	AB	T3-T2
• Duration	B	T4-T3

Static keystroke Dynamic

- Password: K_1 K_2 K_3 … K_N
- Timings (KeyDown and KeyUp) for each key
- Calculate N durations and N-1 latencies
  - Durations: D_1 D_2 D_3 … D_N
  - Latencies: L_1 L_2 L_3 … L_{N-1}
- Note:
  - Durations are always positive
  - Latencies can be negative

Alternative definitions



- Latency =
  T3-T2 = L, our definition, can be negative
  T3-T1 = L + Dur_A
  T4-T2 = L + Dur_B
  T4-T1 = L + Dur_A + Dur_B

Template creation

- New password: .tie5Roanl
  - 11 keys (shift key included) so 11 durations and 10 latencies
  - Kind of hard to type in the beginning
  - Need to learn to type a new password

- For template creation a user types the new password K_1+K_2 times
  - First K_1 are ignored
  - Next K_2 are used for template creation

Template creation

- K_2 measurements of duration D_1
  - $D_{1,1}$, $D_{1,2}$, … $D_{1,K2}$
- Add to template:
  - μ_{D1} = mean ($D_{1,1}$, $D_{1,2}$, … $D_{1,K2}$)
  - σ_{D1} = standard deviation ($D_{1,1}$, $D_{1,2}$, … $D_{1,K2}$)

- Repeat this for all durations and latencies

New input

- We don’t have mean and standard deviation on a single new input
  - Durations: $t_{D(1)}$, $t_{D(2)}$, … , $t_{D(N)}$
  - Latencies: $t_{L(1)}$, $t_{L(2)}$, … , $t_{L(N-1)}$

- Potential distance metric:
  - Manhattan: $d = \Sigma \mid \mu_i - t_i \mid$
  - Scaled Manhattan: $d = \Sigma \mid \mu_i - t_i \mid / \sigma_i$

Alternative distance metrics

- Use only durations or only latencies
- Use only those durations or latencies with a small standard deviation
  - These are typed in a stable way by the user

- How to deal with backspace?
  - Backspace results in missing latency value

Template updating

- Template is created during enrollment
- People will learn to type the password more fluent over time
- Need to update the template with newly typed input
- Need for comparison of update mechanisms

Template updating

- Often the method for updating is mentioned and the performance improvement
- Possible methods
  - Fixed size: delete oldest or “strangest”
  - Increasing size: add new input to list
  - Combination: add new input to list up to maximum size and then switch to fixed size

Template updating

- Important issues:
  - **Data set used and how it is split (ignore, template, testing)**
  - Threshold used
  - Input size
  - Global input order
  - Local input order
  - Update decision
  - **Update mechanism**
  - Performance representation

Template updating

- Public dataset, 51 users, 8 sessions, 50 times typing the password .tie5Roanl per session
- First 20 ignored, next 30 for template creation, remaining 350 for testing
- Threshold = 42.25 (gives EER=21.6% without template update)
- Input size = 350 genuine and 350 impostor
- Performance representation = (FMR ; FNMR)

Template updating

- Global input order =
  - Genuine first
  - Impostor first
  - Random
- Local input order = Randomly chosen impostor
- Update decision = On success
- Update mechanism =
  - Fixed size (30), replace oldest
  - Max size (50), replace oldest

Template updating
Results

- Baseline: EER=21.59, so TER=43.18

	Genuine	First	Impostor	First	Random	
	Fixed	Max	Fixed	Max	Fixed	Max
FNMR	19.72	11.35	22.11	13.87	13.81	6.32
FMR	12.85	18.99	53.60	67.50	23.45	42.23
TER	32.57	30.34	75.71	81.37	37.26	48.55

Continuous keystroke dynamics

- What is the meaning of FMR and FNMR in continuous keystroke dynamics?

- For continuous authentication we need to define the **trust** of the system in the **genuineness** of the current user

Continuous keystroke dynamics

- When a user types as he should
  - Then the trust in the genuineness increases
- When a user does not type as he should
  - Then the trust in the genuineness decreases
- After static log on
  - The trust is maximal (100%)
- When the trust drops below a certain threshold
  - Then the user is logged out

Continuous keystroke dynamics

- We want FMR=0
  - Meaning that an impostor is detected!
- We want FNMR=1
  - Meaning that a genuine user is not locked out
- But more important:
  - We want to detect an impostor as fast as possible
  - Low number of keystrokes before detection is important
  - Experiments show that on average we can detect an impostor after approx 180 keystrokes

Program

1. Introduction
2. Authentication in general
3. Password authentication and security
4. Biometrics
   a. Introduction
   b. Biometric Evaluation
   c. Gait recognition
   d. Keystroke Dynamics
5. Conclusions

Conclusions

- Different authentication mechanisms exist
  - Username/password
  - Token
  - Biometrics
- You heard (learned?) something
  - about password security
  - biometrics in general and biometric evaluation
  - behavioural biometrics: gait and keystroke dynamics

Contact

- Patrick Bours
  - Email: patrick.bours@hig.no
  - Phone: +47 611 35 250
  - Mobile: +47 412 65 872
  - Skype: patrick.bours.norge
  - Twitter: http://twitter.com/PatrickBours