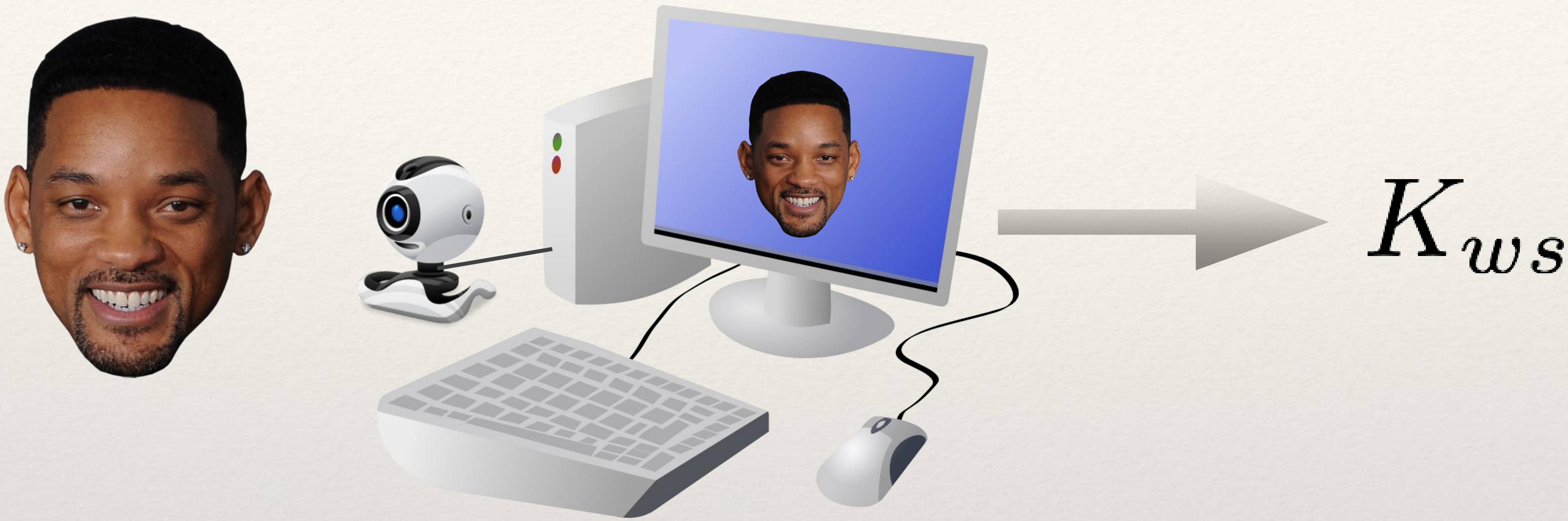


Finding Yourself Is The Key – Biometric Key Derivation that Keeps Your Privacy

Orr Dunkelman,
University of Haifa

Joint work with Mahmood Sharif and Margarita Osadchy

Overview

- Motivation
- Background:
  - The Fuzziness Problem
  - Cryptographic Constructions
  - Previous Work
  - Requirements
- Our System:
  - Feature Extraction
  - Binarization
  - Full System
- Experiments
- Conclusions

Motivation

- Key-Derivation: generating a secret key, from information possessed by the user
- Passwords, the most widely used mean for key derivation, are problematic:

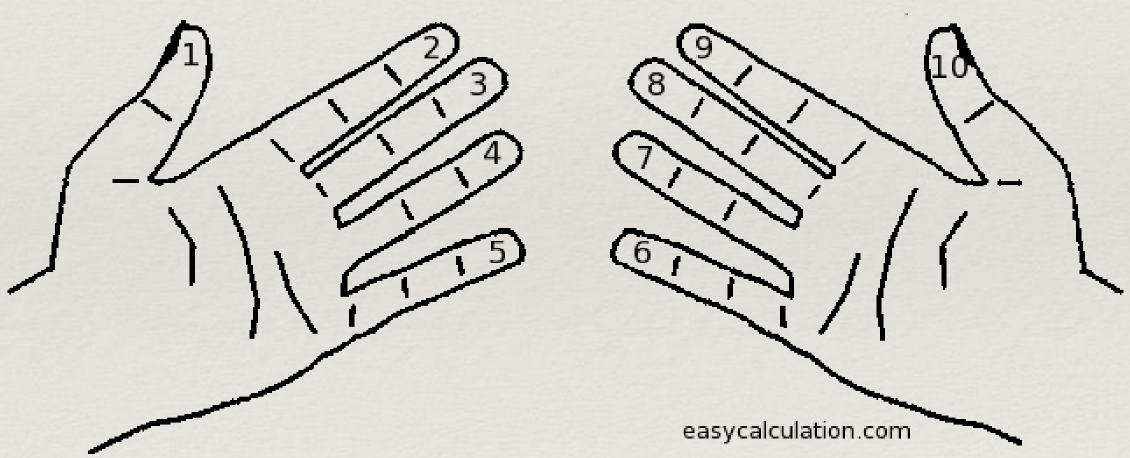
1. Forgettable
2. Easily observable (shoulder-surfing)
3. Low entropy
4. Carried over between systems

Motivation

- **<u>Suggestion</u>: use biometric data for key generation**
- **Problems :**

1. **It is hard/impossible to replace the biometric template in case it gets compromised**
2. **Privacy of the users**

1



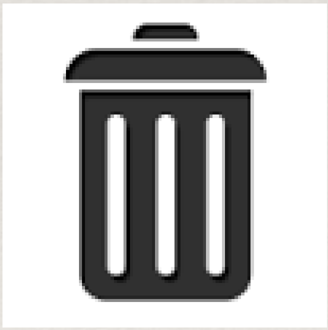
Overview

- Motivation
- Background:
  - The Fuzziness Problem
  - Cryptographic Constructions
  - Previous Work
  - Requirements
- Our System:
  - Feature Extraction
  - Binarization
  - Full System
- Experiments
- Conclusions

Biometric Key Derivation

x

K



The Fuzziness Problem

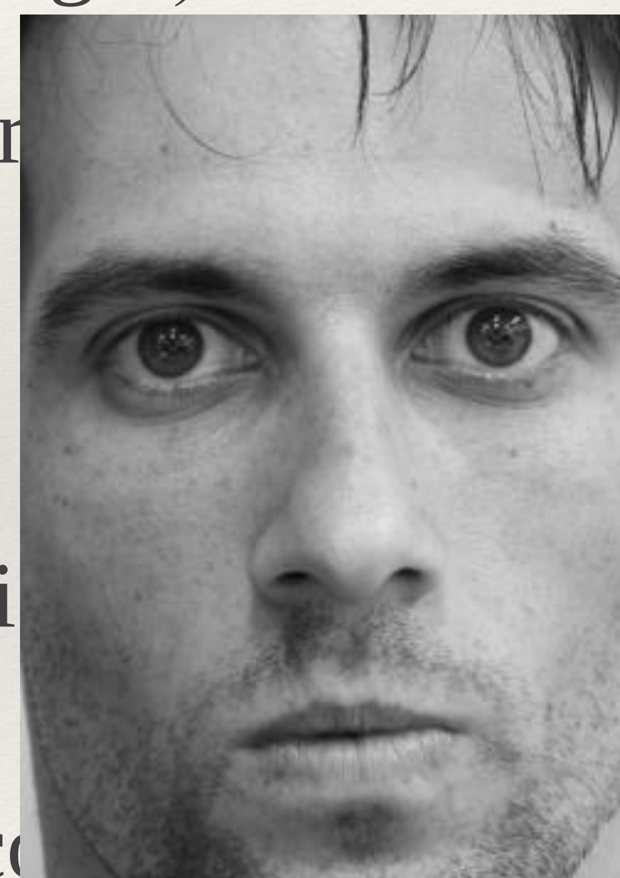
- Two images of the same face are rarely identical (due to lighting, pose, expression changes)

- Yet we wa y cr y for the user every time

- The fuzzin es i

1. Feature e

2. The use o n c data

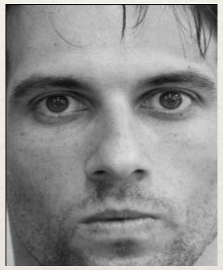
- **Taken one after the other**
- **81689 pixels are different**
- **only 3061 pixels have identical values!**

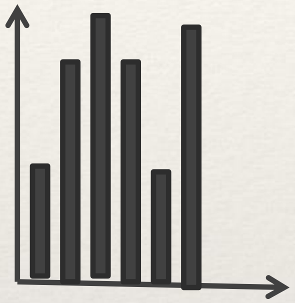
The 3 Step Process

Feature extraction

Binarization

Error correction





$$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

ECC

reduces changes due to viewing conditions and small distortions

converts to binary representation and removes most of the noise

removes the remaining noise

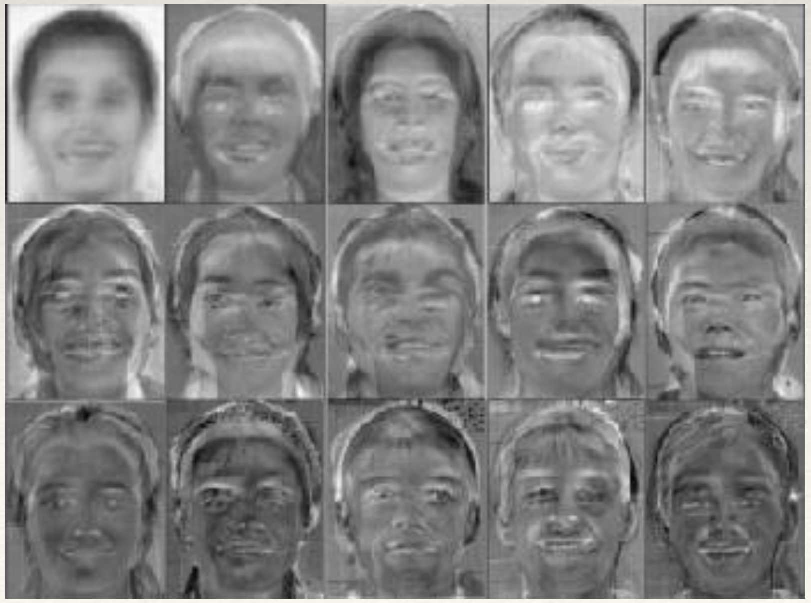
Feature Extraction

User-specific features:

Eigenfaces (PCA)

Fisherfaces (FLD)

training step produces user specific parameters, stored for feature extraction



Generic Features

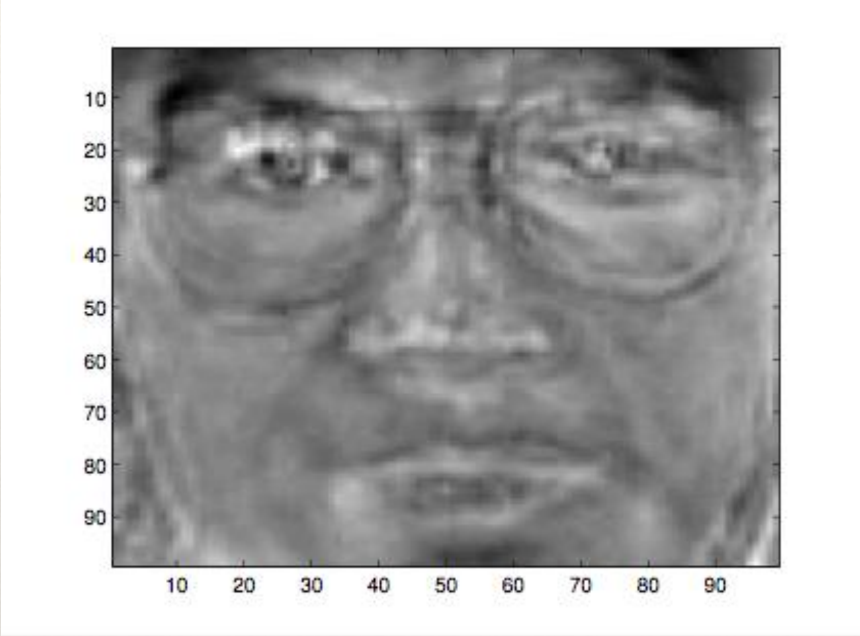
Histograms of low-level features, e.g.: LBPs, SIFT

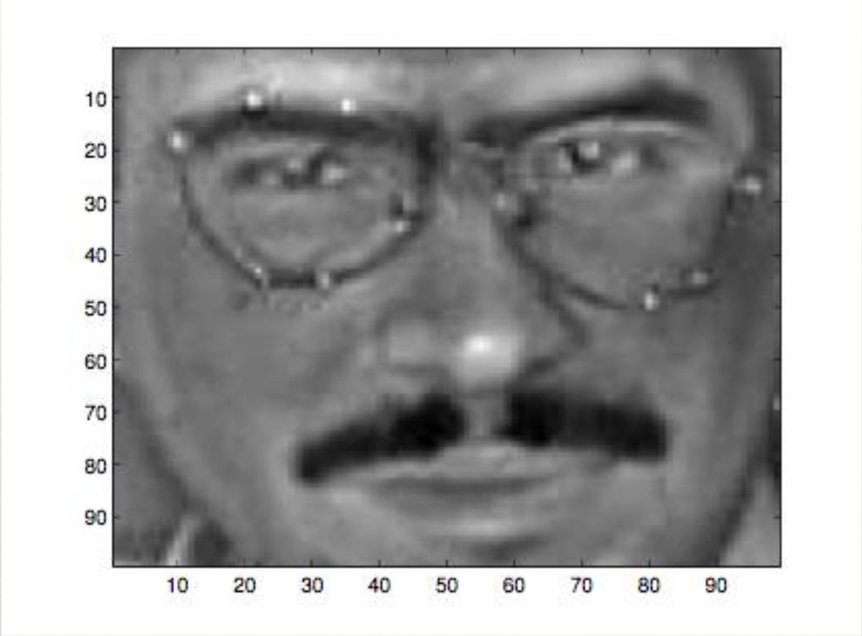
Filters : Gabor features, etc

No training, no user specific information is required

Feature Extraction
Previous Work

- [FYJ10] used Fisherfaces - public data looks like the users:





- Very Discriminative (better recognition)
- But compromises privacy – **<u>cannot be used!</u>**

Feature Extraction
Generic Features?

- Yes, but require caution.
- In [KSVAZ05] high-order dependencies between different channels of the Gabor transform
- ➜ correlations between the bits of the suggested representation

Binarization

- Essential for using the cryptographic constructions
- ~~Some claim: non-invertibile [TGN06]~~

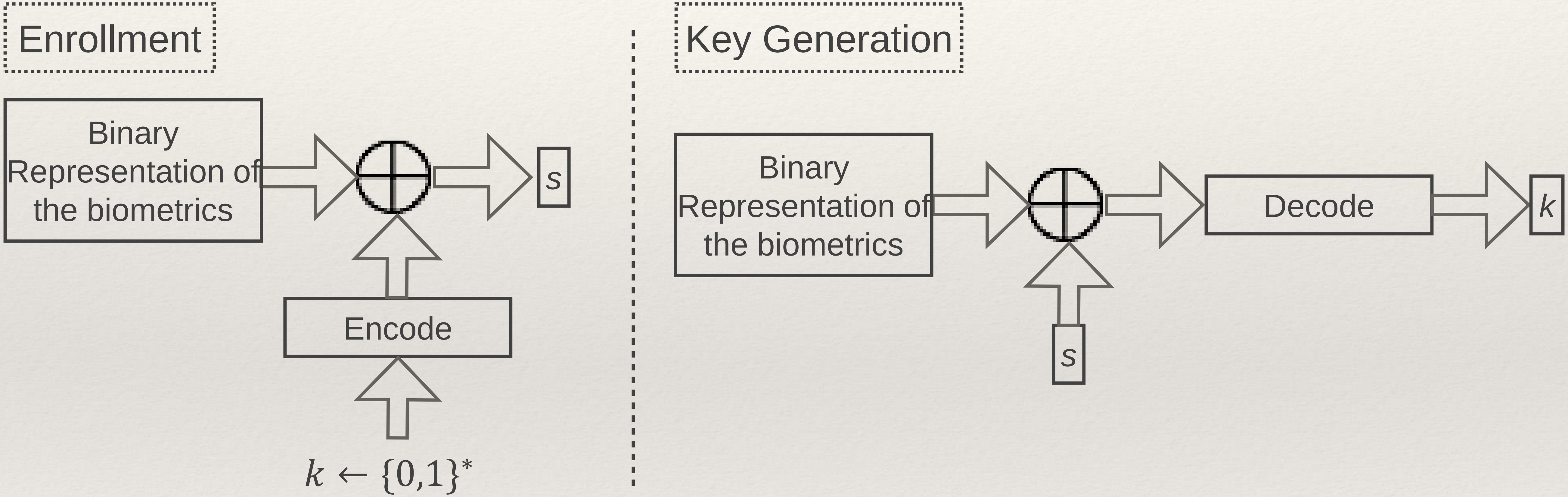
Biometric features can be approximated

- By :
  - Sign of projection
  - Quantization

Quantization is more accurate, but requires storing additional private information.

Cryptographic Noise Tolerant Constructions

- Fuzzy Commitment [JW99]:



- Other constructions: Fuzzy Vault [JS06], Fuzzy Extractors [DORS08]

Previous Work

Problems

1. Short keys
2. Non-uniformly distributed binary strings as an input for the fuzzy commitment scheme
3. Dependency between bits of the biometric samples
4. Auxiliary data leaks personal information
5. No privacy-protection when the adversary gets hold of the cryptographic key (A.K.A. Strong biometric privacy)

Security Requirements

1. Consistency: identify a person as himself (low FRR)
2. Discrimination: impostor cannot impersonate an enrolled user (low FAR)

[BKR08]:

3. Weak Biometric Privacy (REQ-WBP): computationally infeasible to learn the biometric information given the helper data
4. Strong Biometric Privacy (REQ-SBP): computationally infeasible to learn the biometric information given the helper data and the key
5. Key Randomness (REQ-KR): given access to the helper data, the key should be computationally indistinguishable from random

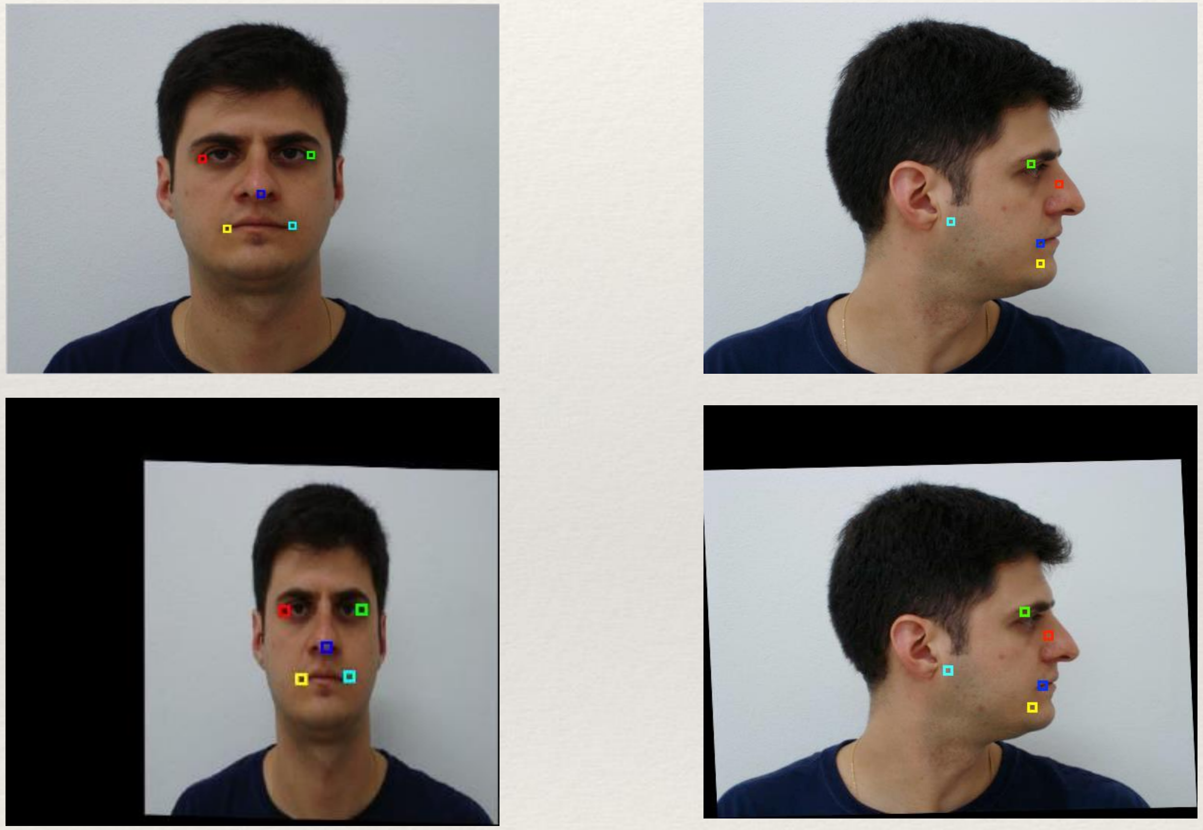
Overview

- Motivation
- Background:
  1. The Fuzziness Problem
  2. Cryptographic Constructions
  3. Previous Work
  4. Requirements
- Our System:
  1. Feature Extraction
  2. Binarization
  3. Full System
- Experiments
- Conclusions

Feature Extraction

1. Landmark Localization and Alignment

- Face landmark localization [ZR12] and affine transformation to a canonical pose:

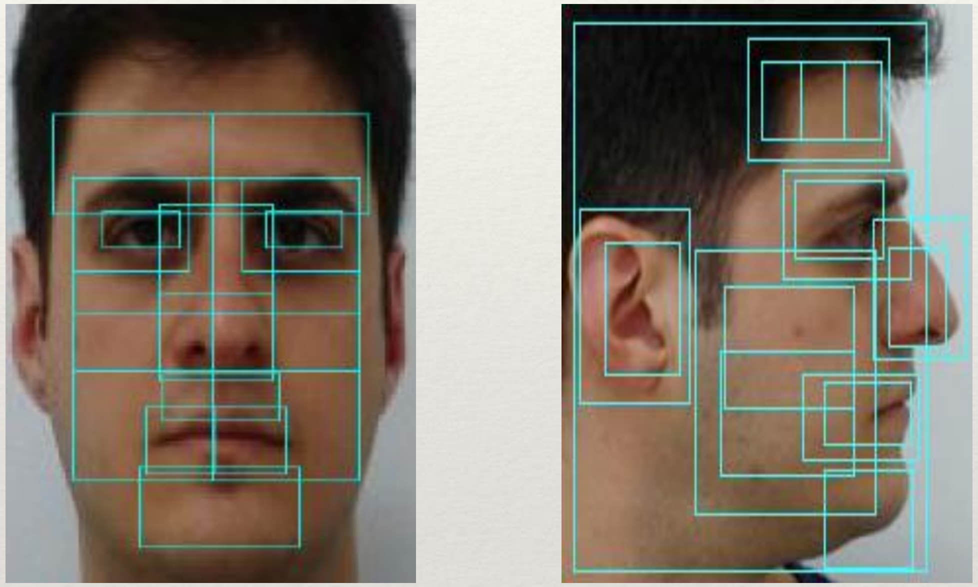


- An essential step, due to the inability to perform alignment between enrolled and newly presented template

Feature Extraction

2. Feature Extraction

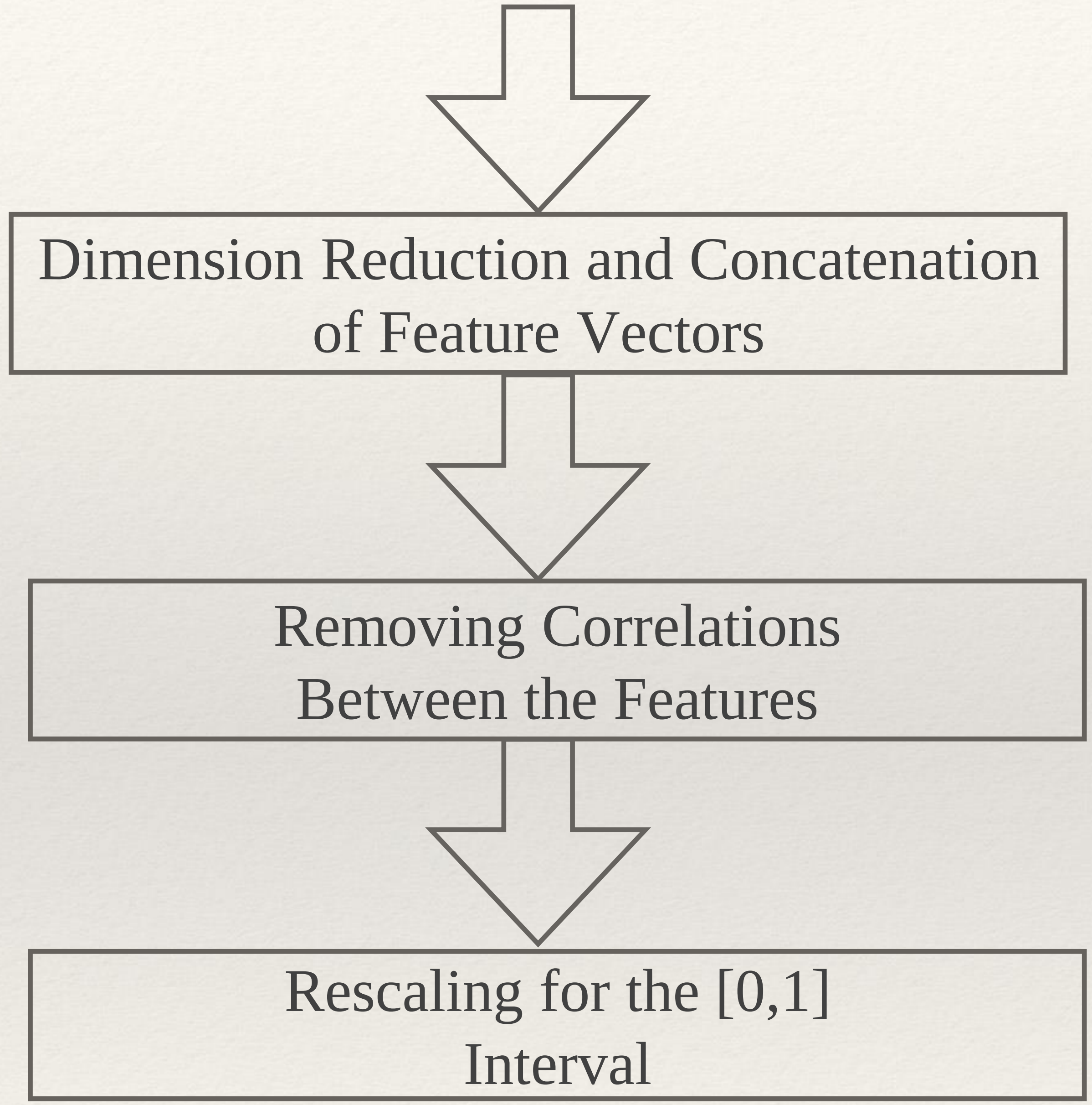
- Local Binary Patterns (LBPs) descriptors are computed from 21 regions defined on the face:



- The same is done with Scale Invariant Feature Transform (SIFT) descriptors
- Histograms of Oriented Gradients (HoGs) are computed on the whole face

Feature Extraction

3. Dimension Reduction and Whitening



Binarization by Projection

$$\mathbb{R}^{n_1} \rightarrow \{0,1\}^{n_2}$$

x

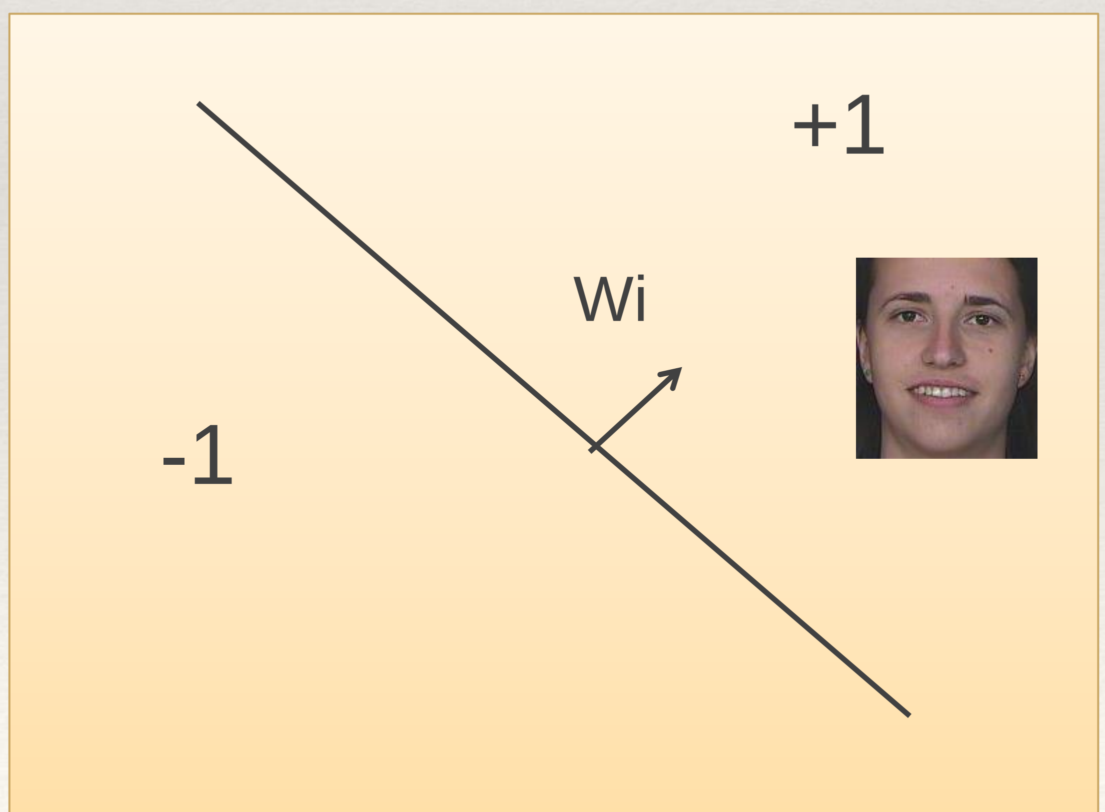
$$h(x) = \frac{1}{2}\left(sign\left(W^T x\right) + 1\right)$$

Binarization by Projection

$$\mathbb{R}^{n_1} \to \{0,1\}^{n_2}$$

x

$$h(x) = \frac{1}{2}\left(sign\left(W^T x\right)+1\right)$$



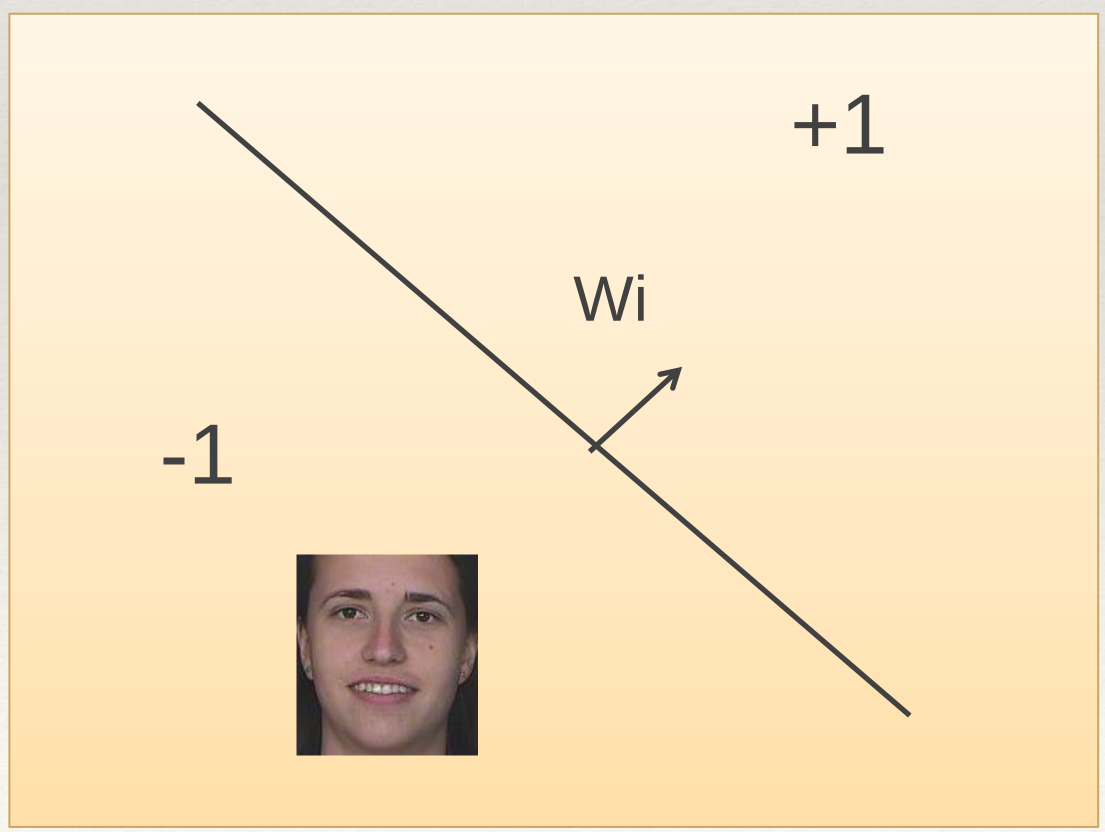
$$h_i(x) = 1$$

Binarization by Projection

$$\mathbb{R}^{n_1} \rightarrow \{0,1\}^{n_2}$$

x

$$h(x) = \frac{1}{2}\left(sign\left(W^T x\right) + 1\right)$$



$$h_i(x) = 0$$

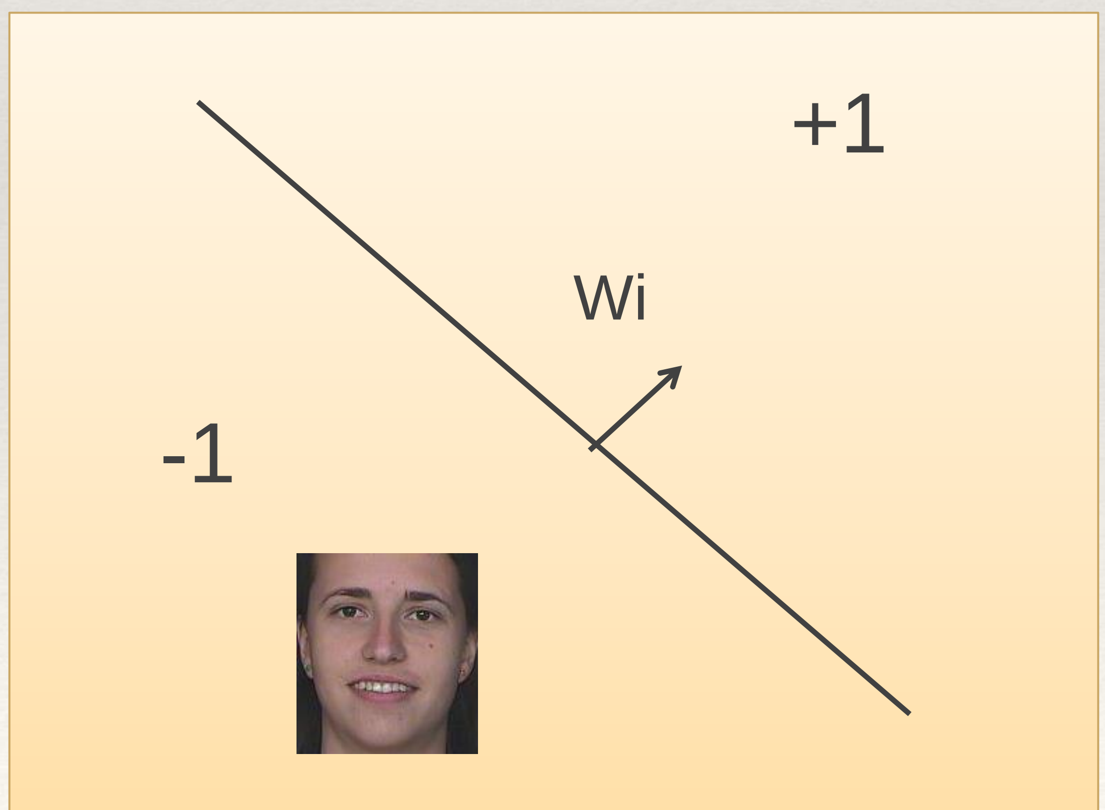
Binarization by Projection

$$\mathbb{R}^{n_1} \rightarrow \{0,1\}^{n_2}$$

x

$$h(x) = \frac{1}{2}\left(sign\left(W^T x\right) + 1\right)$$

h(x') ?



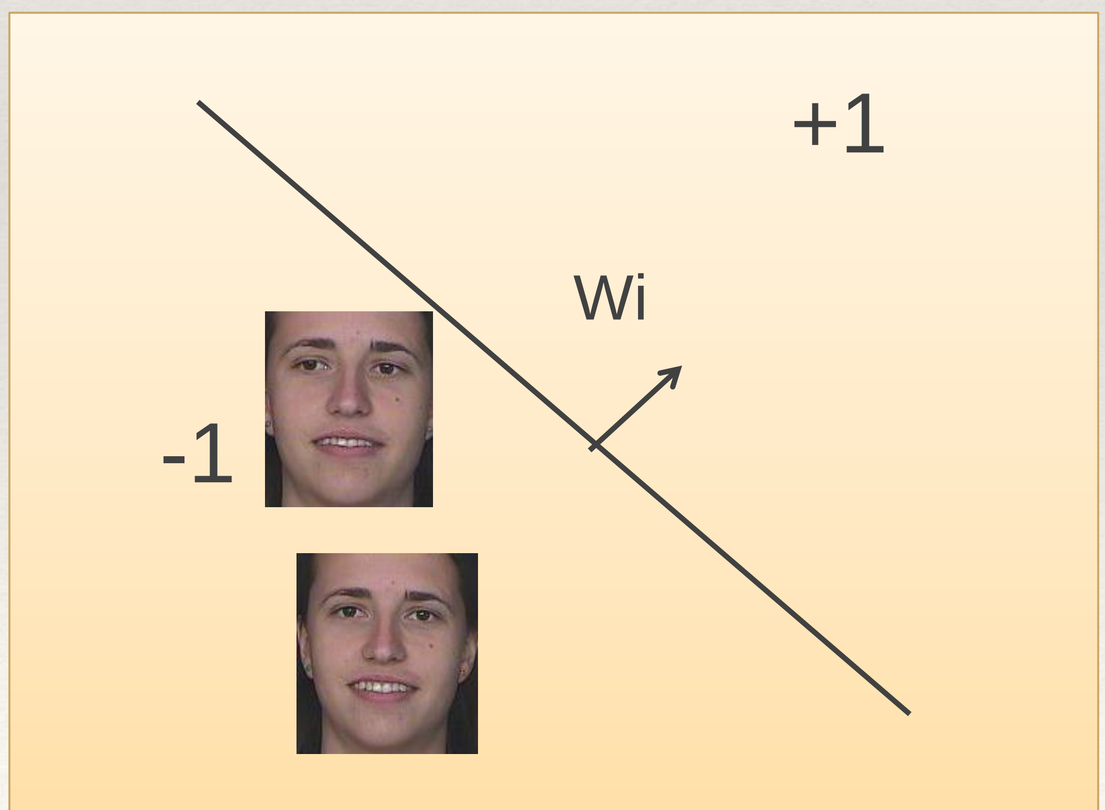
$$h_i(x) = 0$$

Binarization by Projection

$$\mathbb{R}^{n_1} \rightarrow \{0,1\}^{n_2}$$

x

$$h(x) = \frac{1}{2}\left(sign\left(W^T x\right) + 1\right)$$



$$h_i(x) = 0$$

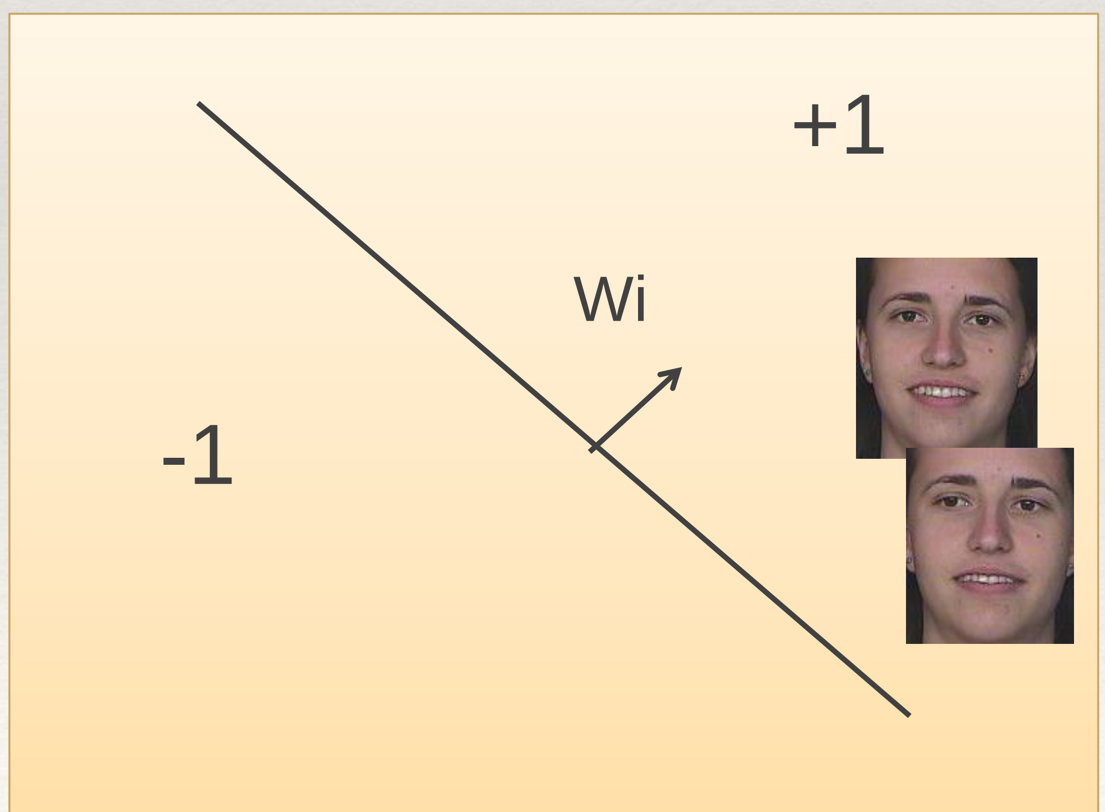
$$h_i(x') = 0$$

Binarization by Projection

$$\mathbb{R}^{n_1} \rightarrow \{0,1\}^{n_2}$$

x

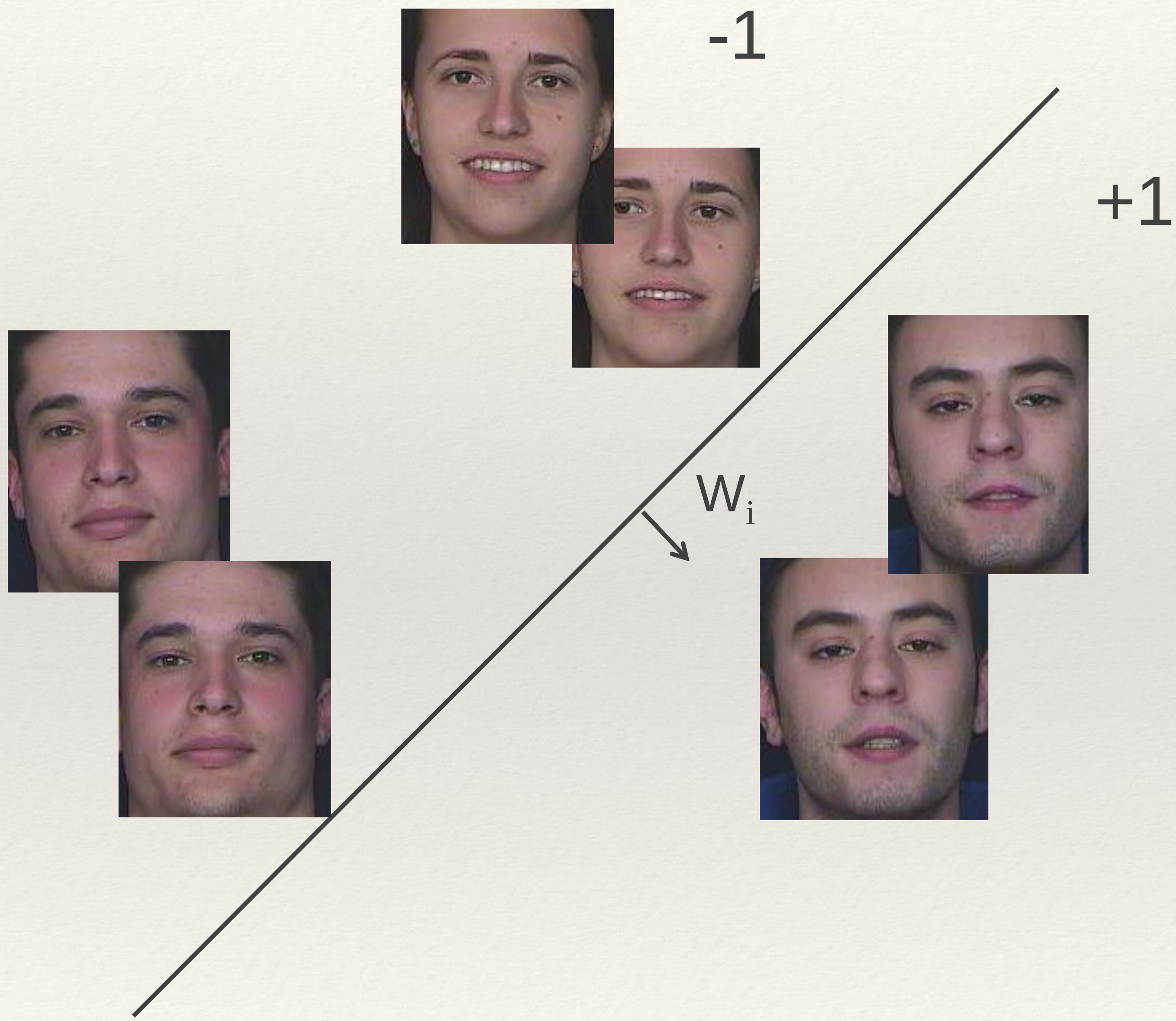
$$h(x) = \frac{1}{2}\left(sign\left(W^T x\right) + 1\right)$$



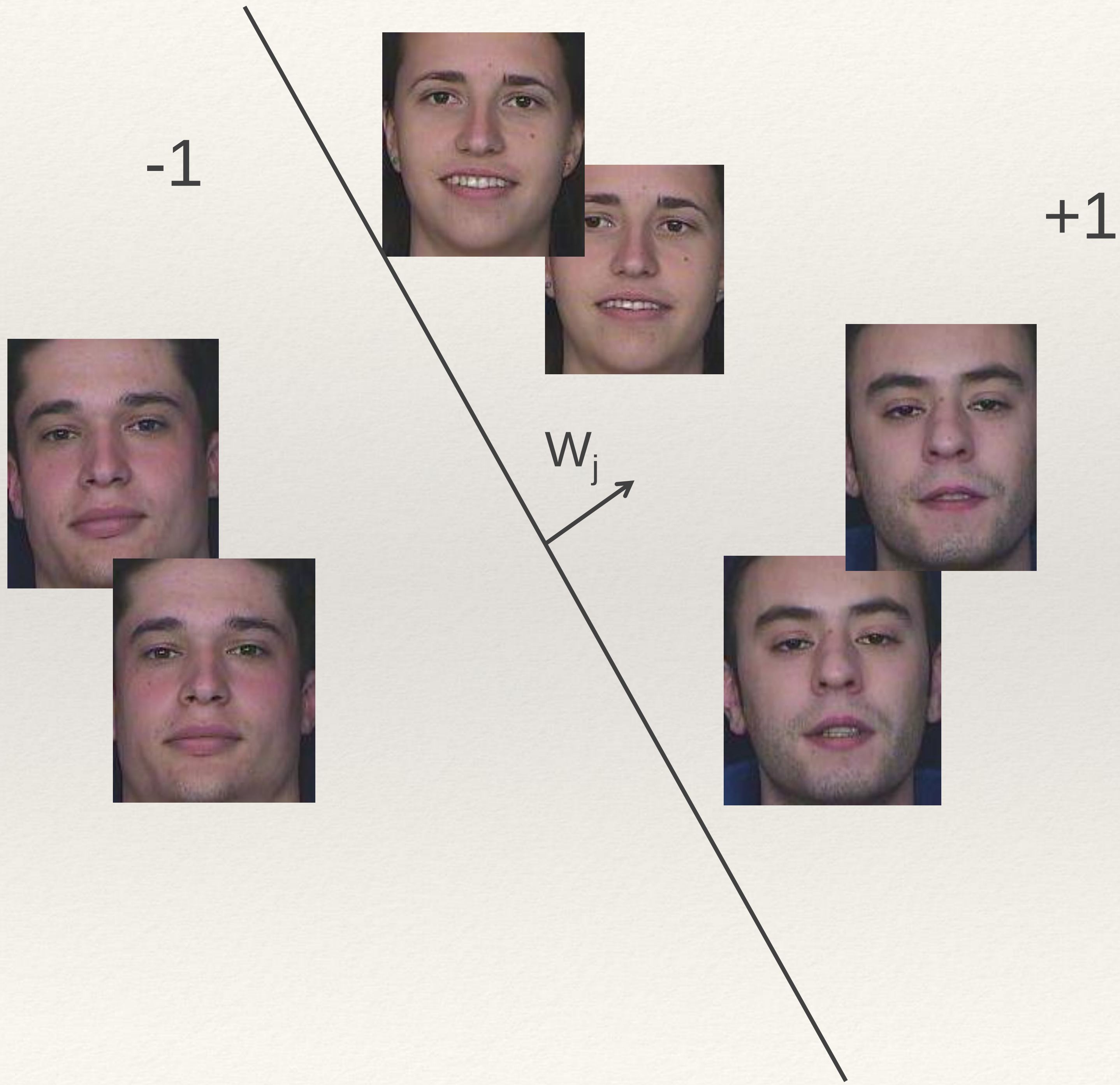
$$h_i(x) = 1$$

$$h_i(x') = 1$$

Embedding in d-dimensional space



Embedding in d-dimensional space



Binarization Alg.

- Requirements from the binary representation:
  1. Consistency and discrimination
  2. No correlations between the bits
  3. High min-entropy
- We find a discriminative projection space *W* by generalizing an algorithm from [WKC10] (for solving ANN problem)
- For: $X = [x_1, x_2, ..., x_n]$ $(x_i, x_j) \in C$ if the pair belongs to the same user

  $(x_i, x_j) \in T$ otherwise
- The aim is to find hyperplanes $[w_1, w_2, ..., w_K]$, s.t. for: $h_k(x) = \mathbf{sgn}(w_k^t x)$

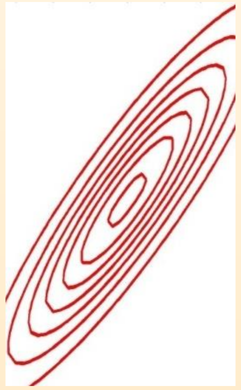
$h_k(x_i) = h_k(x_j)$ if $(x_i, x_j) \in C$

$h_k(x_i) \neq h_k(x_j)$ otherwise

Removing Dependencies between Bits

Dimension Reduction and Concatenation of Feature Vectors

X



Removing Dependencies between Bits

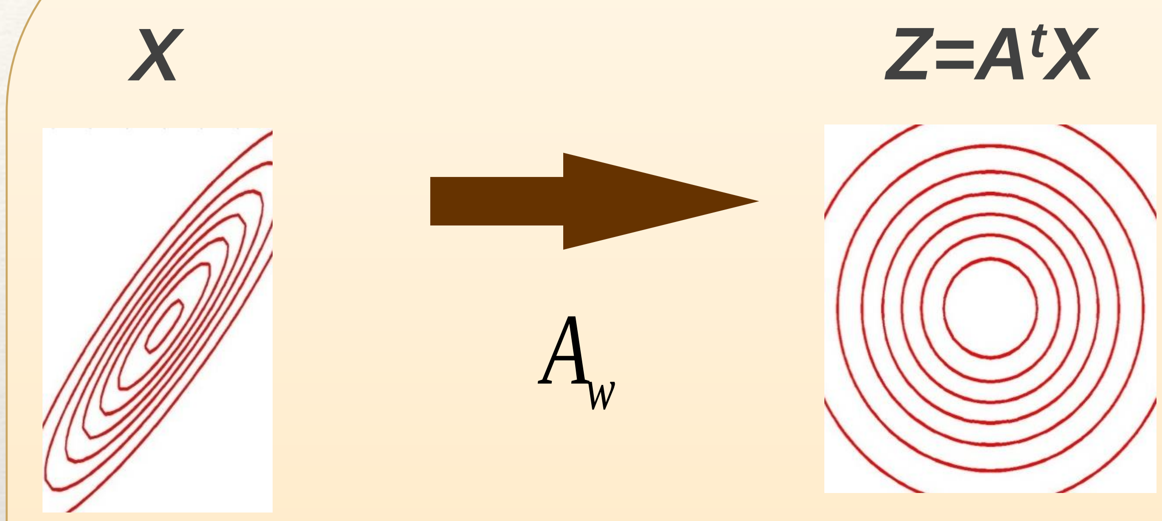
Dimension Reduction and Concatenation of Feature Vectors

↓

Removing Correlations Between the Features

↓

Rescaling for the [0,1] Interval



Removing Dependencies between Bits

Dimension Reduction and Concatenation of Feature Vectors

↓

Removing Correlations Between the Features

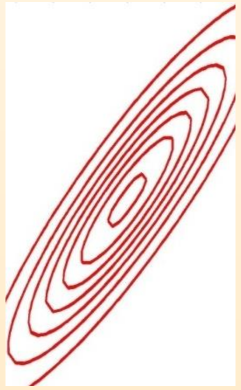
↓

Rescaling for the [0,1] Interval

↓

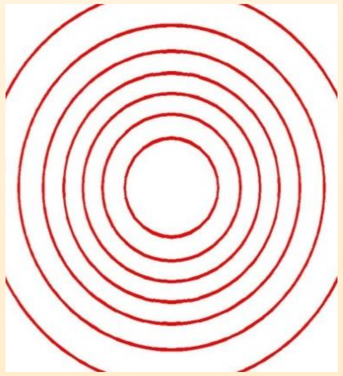
Projection onto orthogonal hypeplanes W

X



A_w

Z=A^tX

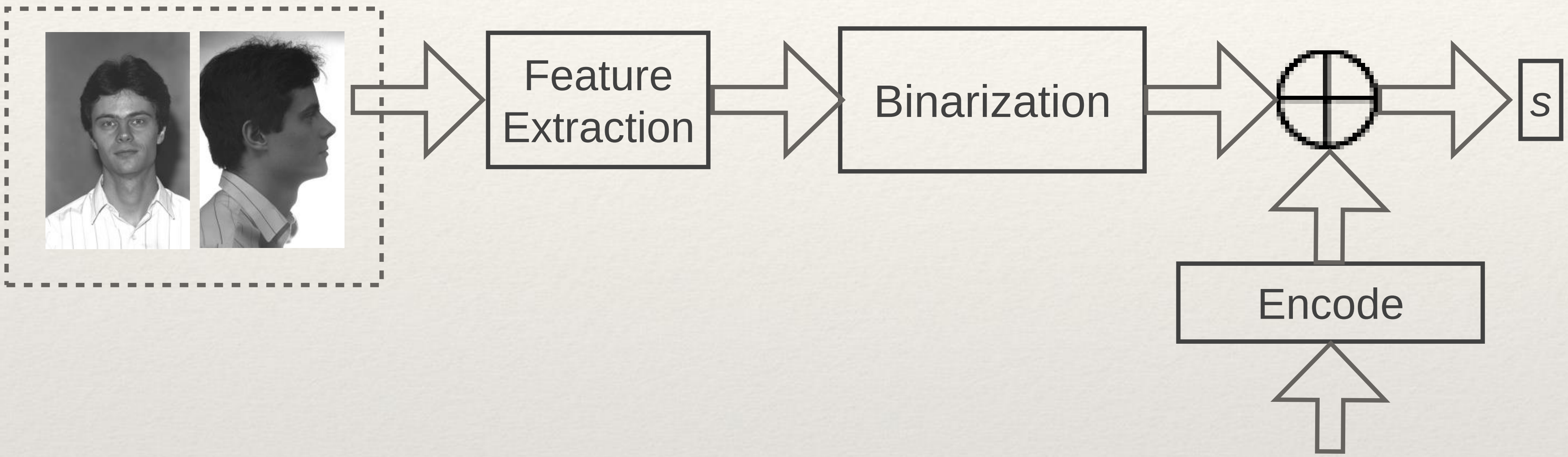


$$h(z) = \frac{1}{2}\left(sign\left(W^T z\right) + 1\right)$$

mutually independent bits

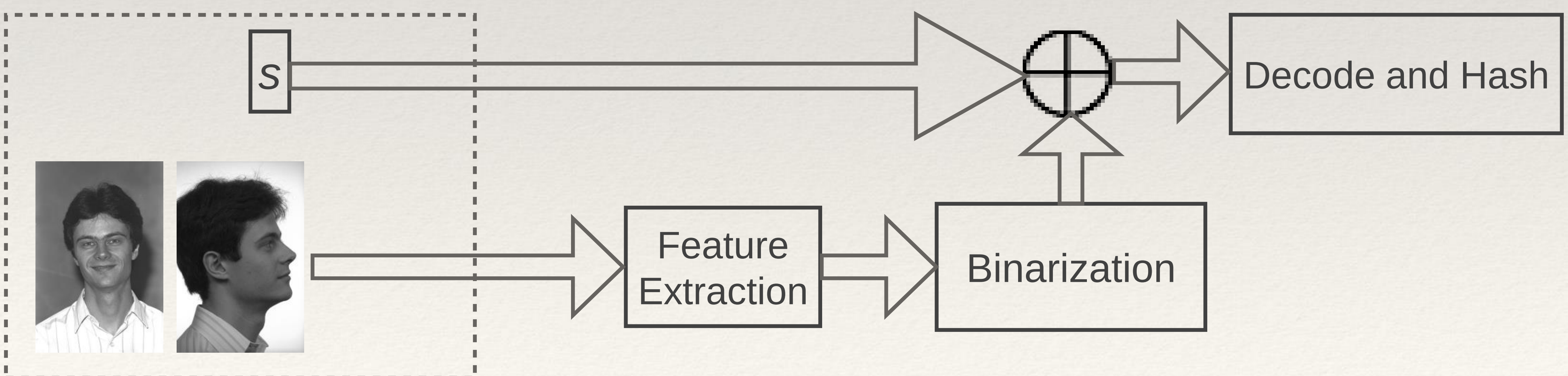
Full System

- Enrollment:



$k \leftarrow \{0,1\}^*$

- Key-Generation:



Transfer Learning of the Embedding

- Learning W is done only **once** using subjects **different from the users of the key derivation system.**
- How is it done?

A more generic question that can be learnt for population.

Overview

- Motivation
- Background:
  - The Fuzziness Problem
  - Cryptographic Constructions
  - Previous Work
  - Requirements
- Our System:
  - Feature Extraction
  - Binarization
  - Full System
- Experiments
- Conclusions

Experiments

Constructing the Embedding

- Performed only once
- Subjects are different than those in testing

View	Number of Subjects	Images Per Subject	Number of Hyperplanes
Frontal	949	3-4	800
Profile	1117	1-8	800

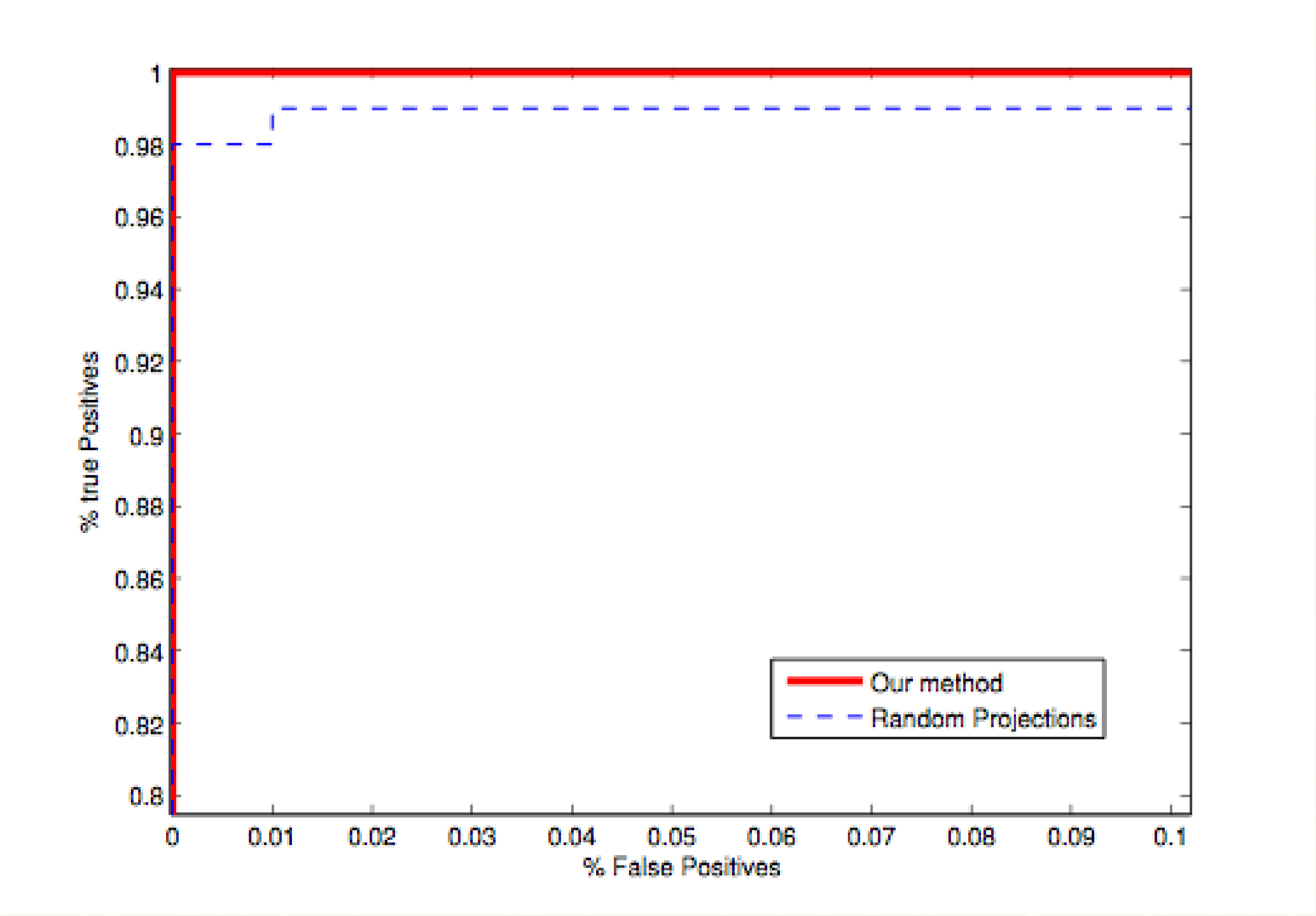
Experiments
Evaluation

- Data:
  - 2 frontal images and 2 profile images of 100 different subjects (not in the training set) were used
- Recognition tests:
  - 5 round cross validation framework was followed to measure TPR-vs-FPR while increasing the threshold (ROC-curves)
- Key generation tests:
  - 100 genuine authentication attempts, and 99*100 impostor authentication attempts

Results

Recognition

ROC curves



Results

Key Generation

- There is a trade-off between the amount of errors that the error-correction code can handle and the length of the produced key
- The Hamming-bound gives the following relation:

$$k \leq \log_2 \left(\frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}} \right)$$

-n: the code length (=*1600* in our case)
-t: the maximal number of corrected errors
-k: the length of the encoded message (produced key, in our case)

Results

Key Generation

For *FAR=:0*

t	*k≥*	FRR our method	FRR Random Projection
595	80	0.30	0.32
609	70	0.16	0.23
624	60	0.12	0.19

Error Correction Code

Reed-Solomon Followed by Concatenation (PUFKY)

Let X be the biometrics

X | 5 bits | 5 bits | … | 5 bits

Reed-Solomon, $GF(2^5)$:

15 symbols over $GF(2^5)$ → 31 symbols over $GF(2^5)$

Probability of error in bit 0.3 → Probability of error in symbol $1-0.7^5 \approx 0.83$

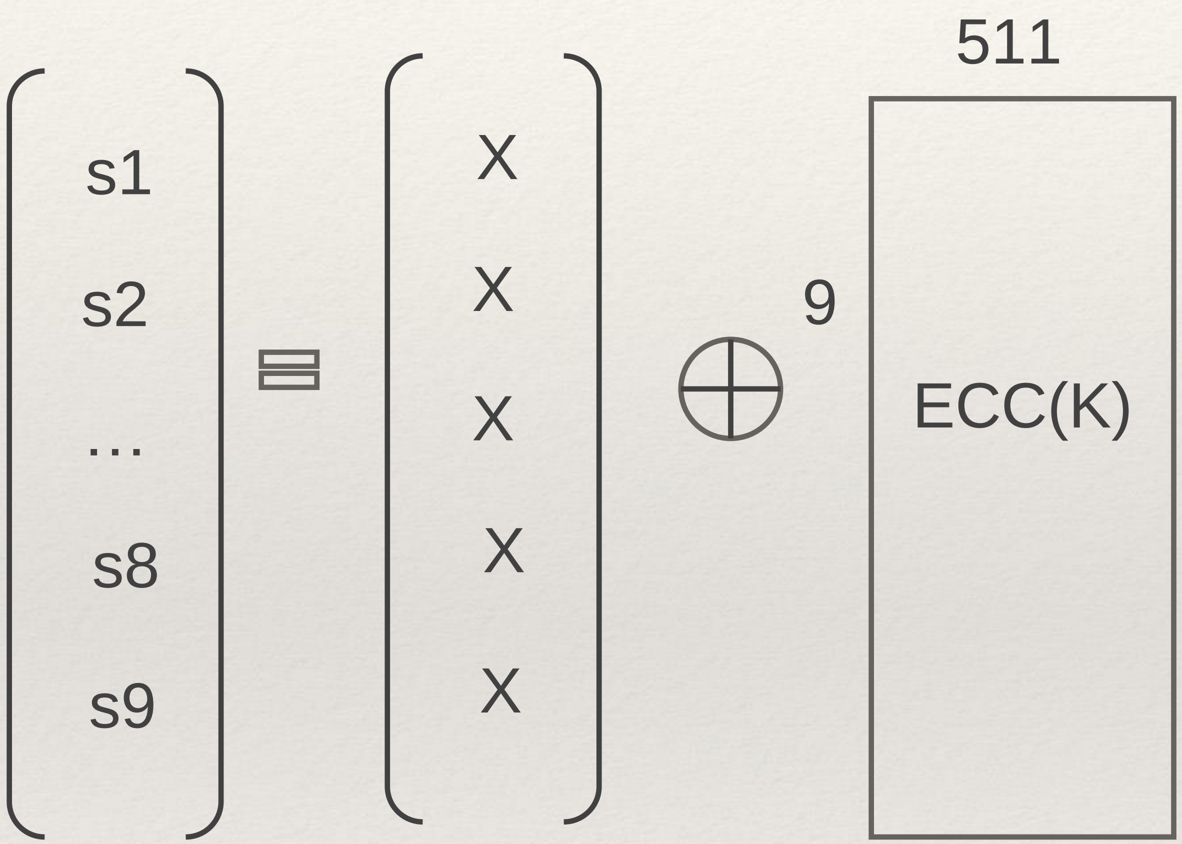
Possible Solution

RS,GF(2^9): 171 Symbols over GF(2^9) $\Rightarrow$ 511 Symbols over GF(2^9)

Probability of error in bit 0.3 $=$ Probability of error in symbol 0.3

Possible Solution

Encoding:



Possible Solution

Decoding:

$$\begin{pmatrix} X' \\ X' \\ \dots \\ X' \\ X' \end{pmatrix} \oplus \begin{pmatrix} s1 \\ s2 \\ \dots \\ s8 \\ s9 \end{pmatrix} = \begin{array}{cc} & 511 \\ 9 & \boxed{} \end{array} \Rightarrow C \xRightarrow{\text{decode(C)}} K$$

Security of Key

Key Length	1539 bits
Security level	171 bits
Biometrics' length	511 bits
Entropy	494.17
FAR (480 subjects)	0
FRR	18.5%

And only a single frontal image needed!

Security Analysis

1. Consistency: FRR = 0.185 (for 1539-bit keys)
2. Discrimination: FAR = 0
3. REQ-WBP: follows from REQ-SBP
4. REQ-SBP: this property is accomplished if the representation is uniformly distributed, as shown in [JW99]

Security Analysis

Uniformity of the Representation

$\boxed{\text{No correlation between the bits + high min-entropy} \Rightarrow \text{uniform distribution}}$

- No correlation between the bits - way :1
  - High degrees-of-freedom $(\gamma = \frac{p(1-p)}{\sigma^2})$: 508.882
    - p: average relative distance between two representation of different persons
    - σ: the standard deviation

Security Analysis

1. Consistency: FRR = 0.16 (for 70-bit key)
2. Discrimination: FAR = 0
3. REQ-WBP: follows from REQ-SBP
4. REQ-SBP: this property is accomplished if the representation is uniformly distributed, as shown in [JW99]
5. REQ-KR: next

Security Analysis

REQ-KR

- Show that $H_\infty(k|s)$ is high

$$k = decode(x \oplus s)$$

- x~U ➜ all possible results of $decode(x \oplus s)$ have an almost equal probability, regardless of s's value
- Thus, $H_\infty(k|s) = H_\infty(decode(x \oplus s)|s) = H_\infty(decode(x \oplus s))$ is high

Overview

- Motivation
- Background:
  - The Fuzziness Problem
  - Cryptographic Constructions
  - Previous Work
  - Requirements
- Our System:
  - Feature Extraction
  - Binarization
  - Full System
- Experiments
- Conclusions

Conclusions

- We showed a system for Key-Derivation that achieves:

1. Consistency and discriminability
2. High min-entropy representation
3. Provable security
4. Provable privacy
5. Fast face-authentication

What this is Good for?

- Key derivation schemes – your face is your key
- Can be easily transformed into a login mechanism
- Can be used in biometric databases (identify double acquisition without hurting honest users’ privacy)

Help Needed

1. We wish to have better training for the vision part
2. Visit our lab – have your photo taken for us (no private information stored)
3. We even pay participants! (not much, still …)

Thank You!