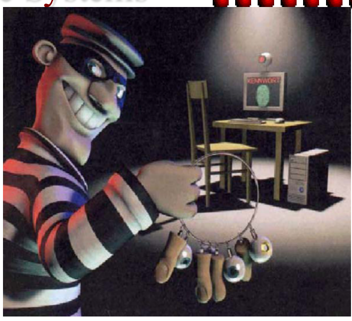
Biometric Systems
Lesson 12– Multibiometric Systems

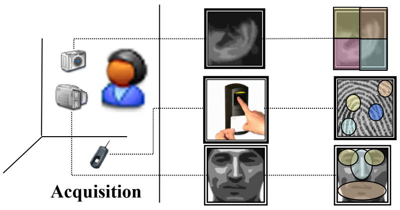
Maria De Marsico
demarsico@di.uniroma1.it

SAPIENZA Università di Roma
Dipartimento di Informatica

Systems with a single biometry vs Multibiometric Systems

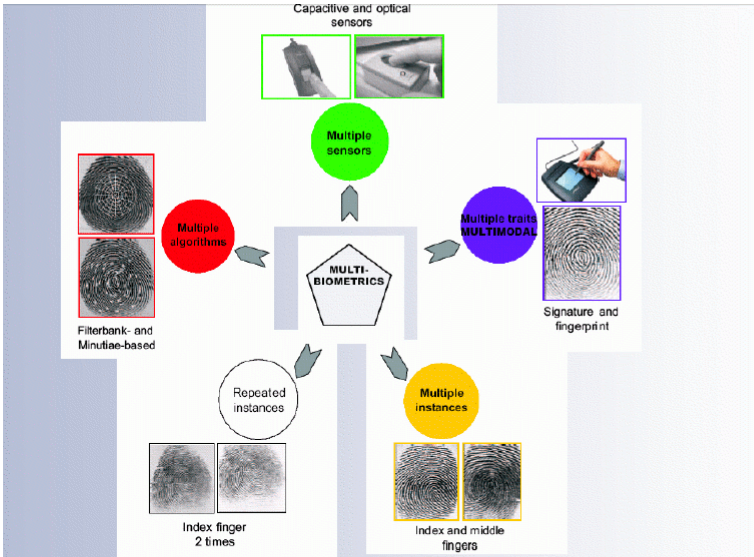
Most present systems are based on a single biometry. This makes them vulnerable to possible attacks, and poorly robust to a number of problems.





A multimodal system provides an effective solution, since the drawbacks of single systems can be counterbalanced thanks to the availability of more biometries.

Kinds of multibiometric systems



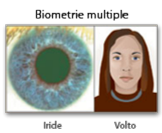
Biometrics, Paulo Lobato Correia, 2007

[Aguilar, J., *Adapted Fusion Schemes for Multimodal Biometric Authentication*, 2006]

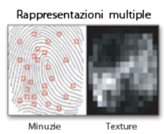
Multimodal, multibiometric and multiexpert (or multiclassifier)

- **Multimodal**:

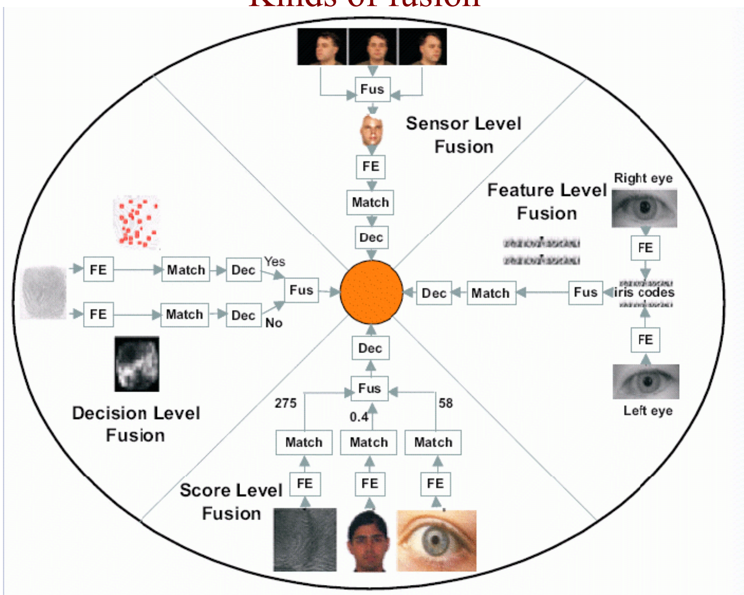
- **Multibiometric**:



- **Multiexpert**:



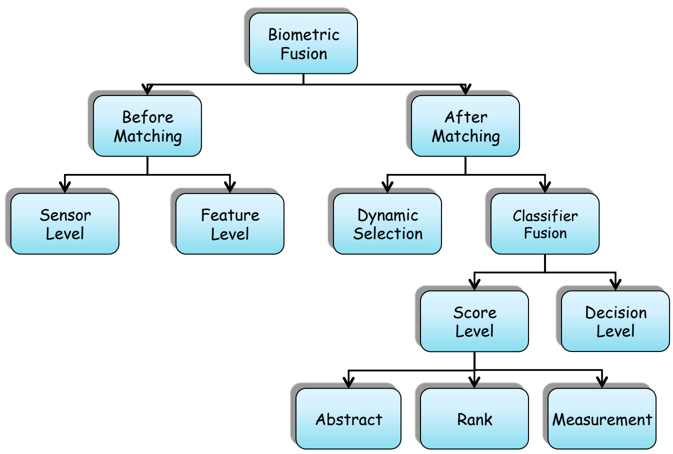
Kinds of fusion



[Aguilar, J., *Adapted Fusion Schemes for Multimodal Biometric Authentication*, 2006]

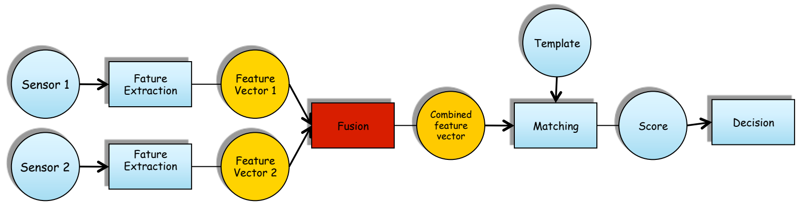
Kinds of fusion

The combination of the different biometries can be peformed in each of the four system modules.



Feature level fusion

Features that were extracted with possibly different techniques can be fused to create a new feature vector to represent the individual.



Better results are expected, since much more information is still present

Possible problems:

•Incompatible feature set.

•Feature vector combination may cause “curse of dimensionality”.

•A more complex matcher may be required.

•Combined vectors may include noisy and/or redundant data.

Feature level fusion - Serial

Feature level data fusion can be based on the simple linking of feature vectors.

$$X = \{x_1, x_2, \ldots, x_m\} \text{ , } Y = \{y_1, y_2, \ldots, y_n\} \Longrightarrow Z = \{x_1, \ldots x_m, y_1, \ldots y_n\}$$

Example: use of SIFT (Scalar Invariant Feature Transform)(Lowe 1999)

Features are computed according to the following phases:

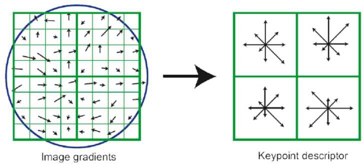
•Feature Extraction: SIFT feature set $s = \{s_1, s_2, \ldots, s_n\}$ with

$s_i = (x, y, \theta, Keydescr)$

x, y -> locazione spaziale

θ -> orientamento locale

$Keydescr$ -> istogrammi di orientamento



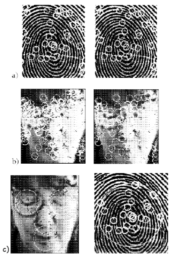
•Feature normalization: it is necessary, due to the possible significant differences in the scale of the vector values.

D.G. Lowe, Object Recognition from Local Scale Invariant Features, In International Conference on Computer Vision, pages 1150-1157, 1999 .

Feature level fusion - Serial

Typical problems to address

- Feature Selection / Reduction : it is more efficient to choose a smaller feature set than the linked vector.
  - K-mean clustering: only maintains the “centroids” of the k clusters formed by the neighbouring points of the linked and normalized vector.
  - Neighborhood elimination: for each point, points laying at a certains distance are removed. It is performed on single vectors before linking.
  - Points belonging to specific regions: only points in specific regions of the biometric trait are maintained.
    eyes, nose, mouth (for the face) – central region (for fingerprints)



- Matching: by *Point pattern matching* technique (Murtagh 1992) wich finds the number of “paired” points between the linked vector of the query image and the one in the database.
  - Two points are paired if their distance is smaller than a predefined theshold.

F. Murtagh, A New Approch to Point-Pattern Matching, Publications of The Astronomical Society of The Pacific 104:301-307, April 1992, http://adsabs.harvard.edu/full/1992PASP..104..301M

Feature level fusion - Parallel

The resulting vector is obtained through the parallel combination of the two feature vectors (Jian et al. 2003)

$$X,\ Y \longrightarrow Z = X + iY \quad \text{where } i \text{ is the imaginary unit}$$

$$X = \{x_1, x_2, \ldots, x_m\} \quad Y = \{y_1, y_2, \ldots, y_m\} \longrightarrow Z = \{x_1 + iy_1, \ldots, x_m + iy_m\}$$

The procedure is the following:

•Vectors normalization : if the vector lenghts are not equal, the shorter vector is extended.

$$X = \{x_1, x_2, x_3\}\ ,\quad Y = \{y_1, y_2\} \ \rangle\ Y = \{y_1, y_2, 0\} \longrightarrow Z = \{x_1 + iy_1, x_2 + iy_2, x_3 + i0\}$$

•Pre-processing of vectors: weighted combination through the combination coefficient θ

- Step 1: transform X, Y vectors into unitary vectors $\overline{X} = X/\|X\|$ and $\overline{Y} = Y/\|Y\|$
- Step 2: if $|X| = |Y|$ then $\theta = 1$ else for $|X| > |Y|$, $\theta = n^2/m^2 \longrightarrow Z = \overline{X} + i\theta\overline{Y}$

•Further feature processing : with well-known linear techniques like PCA, K-L expansion, or LDA.

Y. Jian, Y. Jing-yu and Z. David, et al, "Feature fusion: parallel strategy vs. serial strategy", Pattern Recognition 36 (2003) 1369 – 1381.

Feature level fusion - CCA

- Canonical Correlation Analysis finds a pair of linear transformations, a e b, such to maximaze the correlation coefficient between characteristics (proposed by Pan et al. 2008).

$$u = a^T x \qquad v = b^T y \longrightarrow z = \begin{pmatrix} a \\ b \end{pmatrix}^T \begin{pmatrix} x \\ y \end{pmatrix}$$

- Vectors are first reduced in dimension because CCS suffers from "Small sample size"
- For instance, fusion of ear and face fully exploits correlation due to physiological position

Correlation coefficient

between x_i and x_j $\quad \rho_{ij} = \frac{\sigma_{ij}}{\sqrt{\sigma_{ii}\sigma_{jj}}}$

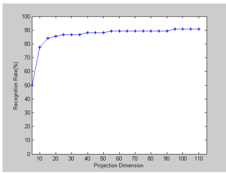
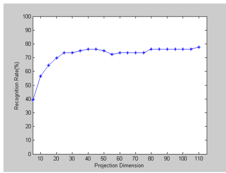
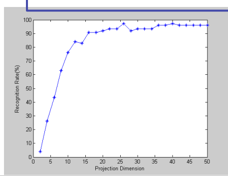
$1 > \rho_{ij} > -1$

Covariance

$$\sigma_{ij} = E\{(x_i - \bar{x}_i)(x_j - \bar{x}_j)\}$$

Variance σ_{ii} and σ_{jj}

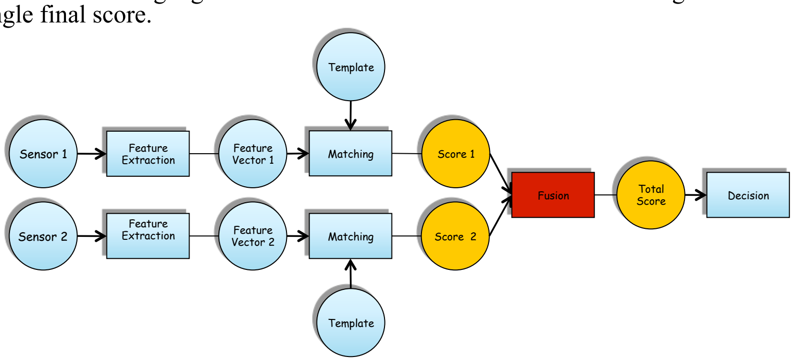
x_i, x_j are positively correlated if $|\rho_{ij}| > 0$

X. Pan, Y. Cao, X. Xu, Y. Lu, Y. Zhao, The Study of Multimodal Recognition Based on Ear and Face, IEEE 2008

Score level fusion

Different matching algorithms return a set of scores that are fused to generate a single final score.



•***Transformation-based*** : the scores from different matchers are first normalized (transformed) in a common domain and then combined using fusion rules.

•***Classifier-based:*** the scores from different classifiers are considered as features and are included into a feature vector. A binary classifier is trained to discriminate between genuine and impostor score vectors (NN-Neural Networks, SVM – Support Vector Machine).

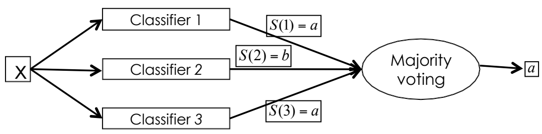
Score level fusion – Fusion Rules

Abstract:

Each classifier outputs its assignment of a *class label* to the input pattern.

•Majority vote:

– *each classifier votes for a class*, the pattern is assigned to the most voted class. Moreover, reliability of the multi-classifier is computed by averaging the single confidences.



Score level fusion – Fusion Rules

Rank:

Each classifier outputs its class rank.

$$\longrightarrow \begin{pmatrix} p_{c_1} = 0.10 \\ p_{c_2} = 0.75 \\ p_{c_3} = 0.15 \end{pmatrix} \longrightarrow \begin{pmatrix} r_{c_1} = 1 \\ r_{c_2} = 3 \\ r_{c_3} = 2 \end{pmatrix}$$

•Borda count:

– *each classifier produces a class ranking ogni classificatore* according to the probability of the pattern belonging to each of them. Ranking are then converted in scores that are summed up; the class with the highest final score is the one chosen by the multi-classifier.

Rank Value	C1	C2	C3
	c	a	b
	b	b	a
	d	d	c
	a	c	d

$$r_a = r_a^{(1)} + r_a^{(2)} + r_a^{(3)} = 1 + 4 + 3 = 8$$

$$\boxed{r_b = r_b^{(1)} + r_b^{(2)} + r_b^{(3)} = 3 + 3 + 4 = 10}$$

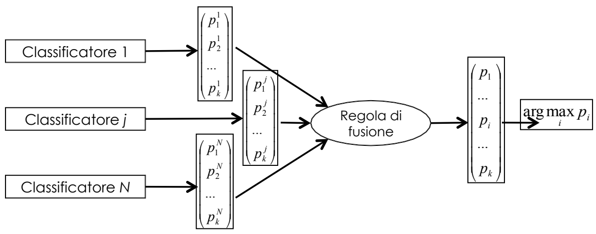
$$r_c = r_c^{(1)} + r_c^{(2)} + r_c^{(3)} = 4 + 1 + 2 = 7$$

$$r_d = r_d^{(1)} + r_d^{(2)} + r_d^{(3)} = 2 + 2 + 1 = 5$$

Score level fusion – Fusion Rules

Measurement:

Each classifier outputs its *classification score* for the pattern in comparison with each class.



Different methods are possible, including sum, weighted sum, mean, product, weighted product, max, min, ecc.

•Sum :

- the sum of the returned confidence vectors is computed, and the pattern is classified according to the highest obtained value

Score level fusion - Normalization

- Scores from different matchers are typically *unhomogeneous*:
  - Similarity/distance
  - Different ranges (eg. [0,1] o [0,100])
  - Different distributions
- To support a consistent score level fusion it is possible to exploit some score transformations (*normalization*), with particular attention to those laying in the overlap region between genuine and impostor.
- Issues to consider when choosing a normalization method:
  - *Robustness*: the transformation should not be influenced by outliers.
  - *Effectiveness*: estimated parameters for the score distribution should best approximate the real values.

Reliability

Due to the possible different quality of input data for the different subsystems, as well as to the possible different accuracy of the adopted recognition procedures, it would be desirable to define a *reliability measure* for each single response of each single subsystem before fusing them in a final response.

•A possible solution to reliability estimate is represented by *confidence margins*.

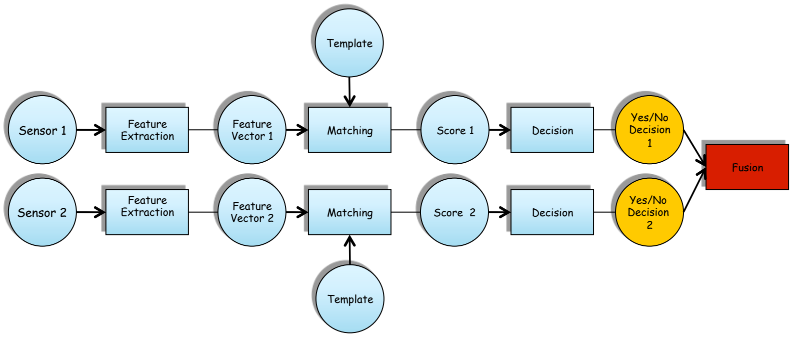
•Among the most popular ones (Poh e Bengio 2004):

$$\mathrm{M}(\Delta) = |FAR(\Delta) - FRR(\Delta)|$$

based on FAR e FRR estimates.

N. Poh, S. Bengio, Improving Fusion with Margin-Derived Confidence In Biometric Authentication Tasks, IDIAP-RR 04-63, November 2004.

Decision level fusion



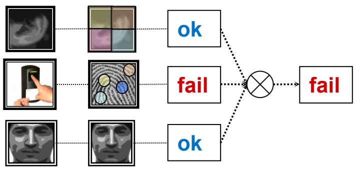
- Each classifier outputs its decision (accept/reject for verification or identity for identification). The final decision is taken by combining the single decisions according to a fusion rule.

Decision level fusion

Different combination strategies are possible. The simplest ones imply a simple logical combination

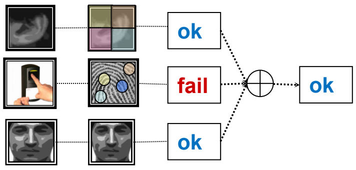
•Serial combination *AND*

global authentication requires all positive decisions.
This improves FAR.



•Parallel combination *OR*

the user may be authenticated even by a single biometric modality.
This improves FRR.



•A further important fusion rule at decision level is *Majority Voting*.

Template Updating – Co-Update Method

"Highly genuine" templates that are classified by one identifier are added to the gallery togheter with the sample corresponding to the other trait (Roli 2007).

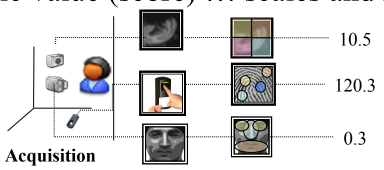
•Benefit: if identifiers are complementary they help each other in identifying "difficult" patterns, by capturing intra-class variations in input data without lowering the acceptance threshold.

A. Rattani, G. L. Marcialis, F. Roli, Capturing large intra-class variations of biometric data by template co-updating, IEEE Workshop on Biometrics, Int. Conf. on Vision and Pattern Recognition CVPR08, Anchorage, Alaska, USA (2008).

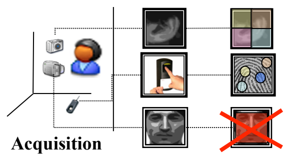
Critical Aspects of Multibiometric Systems

Let us return to some critical aspects:

• When each subsystem assigns a label to each subject with a numeric value (score) … scales and ranges can be different.



•It may happen that responses are not equally reliable.



What about data normalization?

- A number of different solutions have been proposed in literature to solve this problem.

Normalization Functions	
Min/Max	$s'_k = \frac{s_k - \min}{\max - \min}$
Z-score	$s'_k = \frac{s_k - \mu}{\sigma}$
Median/Mad	$s'_k = \frac{s_k - median}{MAD}$
Sigmoid	$s'_k = \frac{1}{1 + ce^{-ks_k}}$
Tanh	$s'_k = \frac{1}{2}\left[tanh\left(0.01\frac{(s_k - E[s_k])}{\sigma(s_k)}\right) + 1\right]$

- When minimum and maximum values are known, the normalization process is trivial.
- For this reason, we assumed to miss an exact estimate of the maximum value
- We chose the average value in its place, in order to stress normalization functions even more.

Testing the existing normalization functions

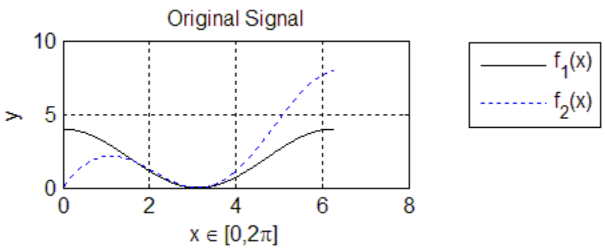
- we chose the two following test functions:

$$f_1(x) = 2 \cdot (\cos(x) + 1)$$

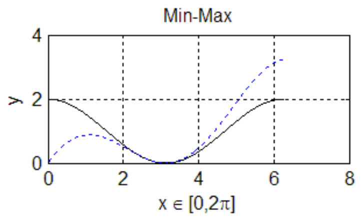
and

$$f_2(x) = 2 \cdot \log(x+1) \cdot (\cos(x) + 1)$$

in [0, 2π] interval.



The Min/Max Function

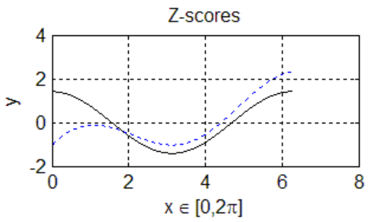


The Min-max normalization technique performs a "mapping" (shifting + compression/dilation) of the interval between the minimum and maximum values in the interval between 0 and 1

Such technique assumes that the minimum and maximum ever generated by a matching module are known.

Normalization Functions	
Min/Max	$s'_k = \frac{s_k - \min}{\max - \min}$
Z-score	$s'_k = \frac{s_k - \mu}{\sigma}$
Median/Mad	$s'_k = \frac{s_k - median}{MAD}$
Sigmoid	$s'_k = \frac{1}{1 + ce^{-ks_k}}$
Tanh	$s'_k = \frac{1}{2}\left[tanh\left(0.01 \frac{(s_k - E[s_k])}{\sigma(s_k)}\right) + 1 \right]$

The Z-Score function



The Z-score technique is the most widespread and uses arithmetic average and standard deviation of scores returned by the single subsystem.

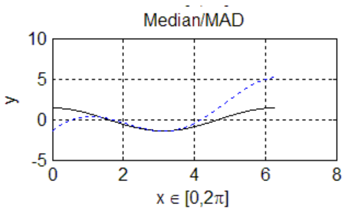
μ represents the arithmetic average of scores and σ is the standard deviation.

Z-score does not guarantee a common interval for normalized values coming from different subsystems.

Normalization Functions

Min/Max	$s'_k = \frac{s_k - \min}{\max - \min}$
Z-score	$s'_k = \frac{s_k - \mu}{\sigma}$
Median/Mad	$s'_k = \frac{s_k - median}{MAD}$
Sigmoid	$s'_k = \frac{1}{1 + ce^{-ks_k}}$
Tanh	$s'_k = \frac{1}{2}\left[tanh\left(0.01\frac{(s_k - E[s_k])}{\sigma(s_k)}\right)+1\right]$

The Median/MAD function



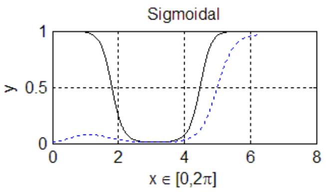
The Median/MAD technique uses the median and the MAD (median of absolute values) .

Median/MAD is less effective, most of all when values have a non-Gaussian distribution; in such cases it neither preserves the original value distribution nor transforms the values in a common numeric interval.

Normalization Functions

Min/Max	$s'_k = \frac{s_k - \min}{\max - \min}$
Z-score	$s'_k = \frac{s_k - \mu}{\sigma}$
Median/Mad	$s'_k = \frac{s_k - median}{MAD}$
Sigmoid	$s'_k = \frac{1}{1 + ce^{-ks_k}}$
Tanh	$s'_k = \frac{1}{2}\left[tanh\left(0.01\frac{(s_k - E[s_k])}{\sigma(s_k)}\right)+1\right]$

The Sigmoidal function



A Sigmoid function has the open interval (0,1) as codomain.

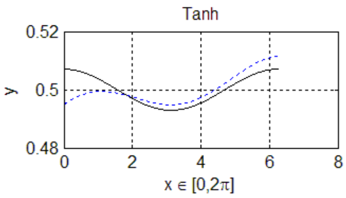
It has two drawbacks:

a) the distortion introduced by the function when x tends to the extremes of the interval is excessive;
b) the shape of the function depends on the two parameters c and k that in turn strongly depend on the domain of x parameter.

Normalization Functions

Min/Max	$s'_k = \frac{s_k - \min}{\max - \min}$
Z-score	$s'_k = \frac{s_k - \mu}{\sigma}$
Median/Mad	$s'_k = \frac{s_k - median}{MAD}$
Sigmoid	$s'_k = \frac{1}{1 + ce^{-ks_k}}$
Tanh	$s'_k = \frac{1}{2}\left[tanh\left(0.01\frac{(s_k - E[s_k])}{\sigma(s_k)}\right) + 1\right]$

The Tanh function



The Tanh function guarantees data to be projected in the open interval (0,1).

It excessively concentrates values around the centre of the interval (0.5).

Normalization Functions

Min/Max	$s'_k = \frac{s_k - \min}{\max - \min}$
Z-score	$s'_k = \frac{s_k - \mu}{\sigma}$
Median/Mad	$s'_k = \frac{s_k - median}{MAD}$
Sigmoid	$s'_k = \frac{1}{1 + ce^{-ks_k}}$
Tanh	$s'_k = \frac{1}{2}\left[tanh\left(0.01\frac{(s_k - E[s_k])}{\sigma(s_k)}\right) + 1\right]$

Some readings

- Ross, A., & Jain, A. (2003). Information fusion in biometrics. *Pattern recognition letters*, *24*(13), 2115-2125.
- Nandakumar, K., Chen, Y., Jain, A. K., & Dass, S. C. (2006, August). Quality-based score level fusion in multibiometric systems. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on* (Vol. 4, pp. 473-476). IEEE.
- Jain, A. K., & Ross, A. (2004). Multibiometric systems. *Communications of the ACM*, *47*(1), 34-40.
- He, M., Horng, S. J., Fan, P., Run, R. S., Chen, R. J., Lai, J. L., ... & Sentosa, K. O. (2010). Performance evaluation of score level fusion in multimodal biometric systems. *Pattern Recognition*, *43*(5), 1789-1800.