Text-independent Speaker Verification Using Support Vector Machines (SVM)

Jamal Kharroubi
Dijana Petrovska-Delacrétaz
Gérard Chollet

(kharroub, petrovsk,chollet)@tsi.enst.fr

ENST/CNRS-LTCI, 46 rue Barrault
75634 PARIS cedex 13

Odyssey 2001 Workshop, 18-22 June 2001

Overview

1 Introduction and motivations
2 SVM principles
3 SVM and speaker recognition
    Identification
    Verification
4 SVM Theory
5 Combining GMM and SVM for speaker verification
6 Database
7 Experimental protocol
8 Results
9 Conclusions and perspectives

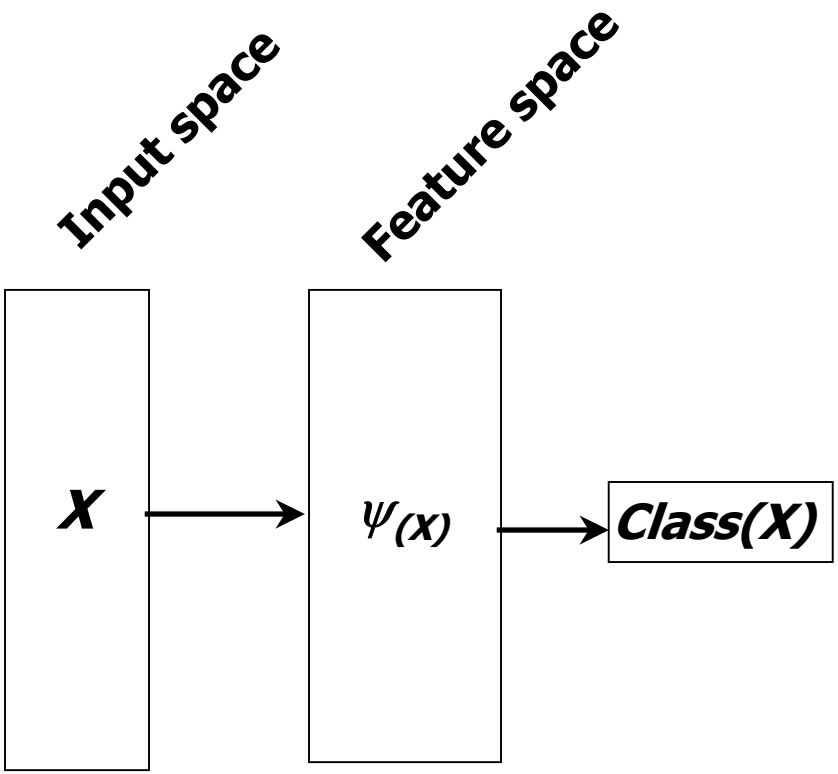
1 Introduction and Motivations

- Gaussian Mixture Models (GMM)
  - ➔State of the art for speaker verification
- Support Vector Machines (SVM)
  - ➔New and promising technique in statistical learning theory
  - ➔Discriminative method
  - ➔Good performance in image processing and multi-modal authentication
- Combine GMM and SVM for Speaker Verification

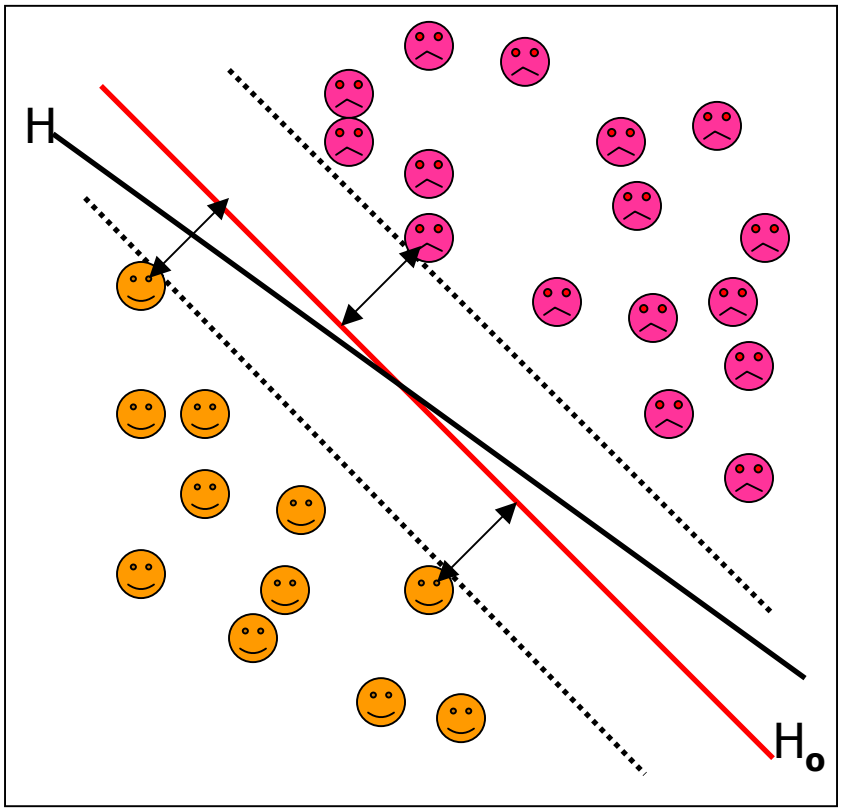
2 SVM Principles

- Pattern classification problem :
  given a set of labelled training data, learn to classify unlabelled test data
- Solution : find decision boundaries that separate the classes, minimising the number of classification errors
- SVM are :
  - ➔ Binary classifiers
  - ➔ Capable of determining automatically the complexity of the decision boundary

2.2 SVM principles



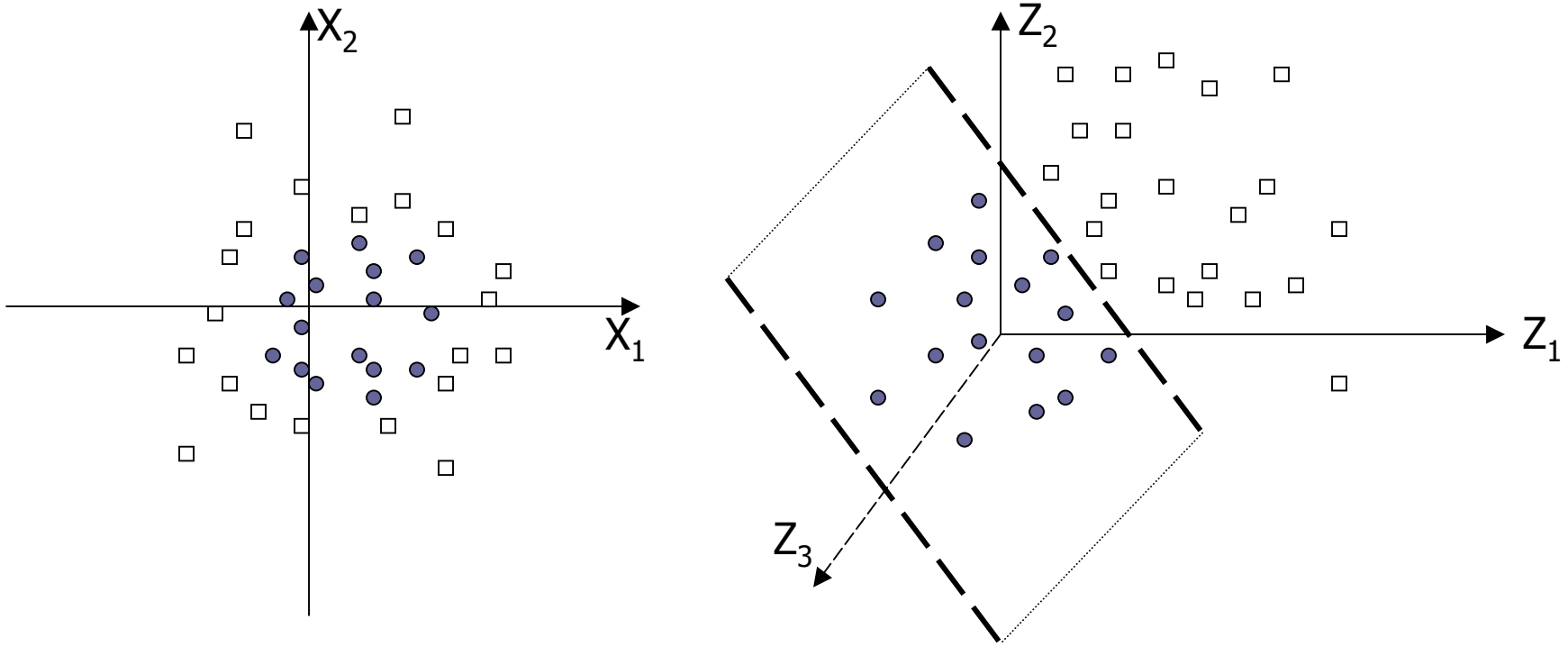
Separating hyperplane H ,
with the optimal hyperplane H_o



2.3 Example

$$\Phi: \ \Re^2 \longrightarrow \Re^3$$

$$(x_1, x_2) \rightarrow (x_1^2, \sqrt{2}\, x_1 x_2, x_2^2)$$



3 SVM and Speaker Recognition

Speaker Identification with SVM : Schmidt and Gish, 1996

- Goal : identify one among a given closed set of speakers
- Methods used : one vs. other speakers or
  pairwize classifier (N(N-1)/2 = 325 for N = 26)
- The input vectors of the SVM's are spectral parameters
- Database : Switchboard, 26 mixed sex speakers,
  15 s for train, 5 s for tests
- Baseline comparison with Bayesian (GMM) modeling

- Results => slightly better performance with SVM's, with the pairwize classifier

- Why these disapointing results ?
  - ➔Too short train/test durations
  - ➔GMM's perhaps better suited to model the data
  - ➔GMM's perhaps more robust to channel variation

3.2 SVM and Speaker Verification

- Not done before
- Difficulty : mismatch of the quantity of labelled data, more data available for impostor access than true target
- Our preliminary test, with speech frames as input to SVM => no satisfactory results
- Present approach :
  model globally the client-client against client-impostor access

4. SVM Theory

Input Space

$$D = \{(x_i, y_i) | x_i \in E; y_i \in \{1, -1\}; i = 1,..m\}$$

Feature Space

$$D = \{(\Psi(x_i), y_i) | x_i \in E; y_i \in \{1, -1\}; i = 1,..m\}$$

Classification Function

$$class(x) = sign[\sum_{SV} a_o y_i (\Psi(x_i) \times \Psi(x)) + b_o]$$

$$K(x_i, x)$$

4.2 SVM – usual kernels used

- Linear

$$K(x, y) = x \times y$$

- Polynomial

$$K(x, y) = [(x \times y) + 1]^d$$

- Radial Basis Function (RBF)

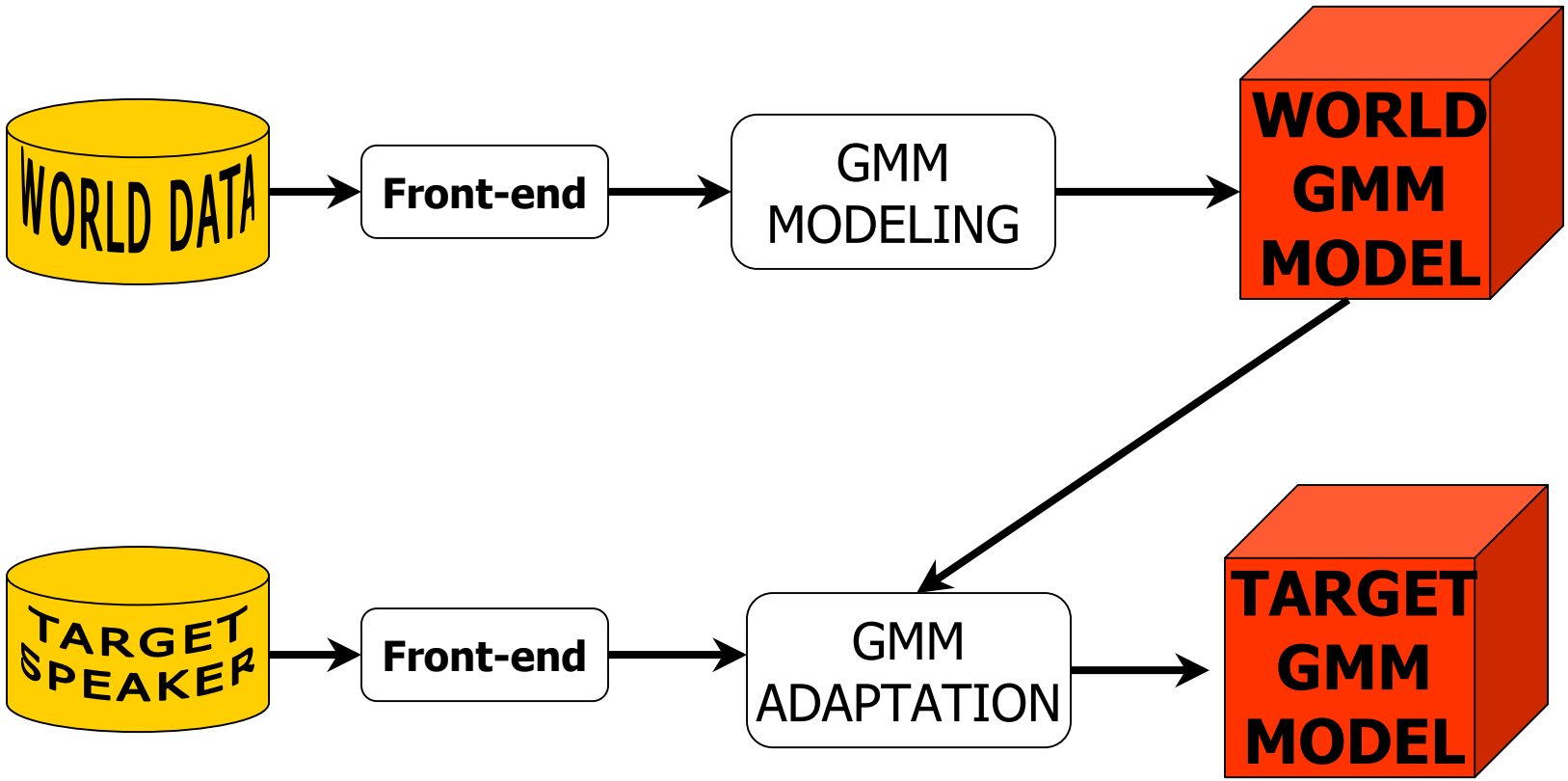
$$K(x, y) = \exp(-\gamma |x - y|^2)$$

5 Combining GMM and SVM for Speaker Verification

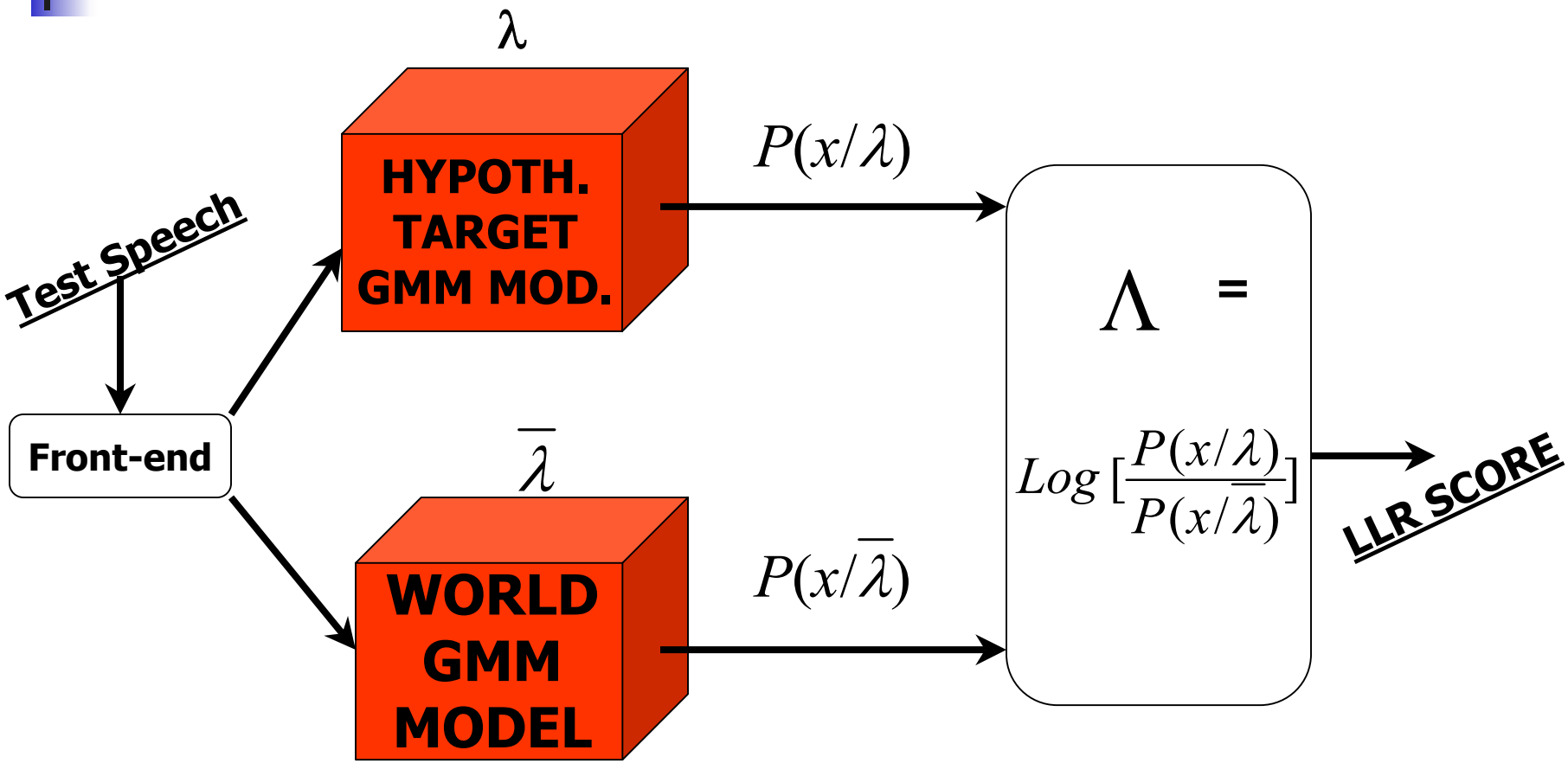
- Reminder : GMM speaker modeling and Log Likelihood Ratio Scoring, referred as LLR

- SVM classifier
  - ➔construction of the SVM input vector
  - ➔SVM train/test procedure

5.1 GMM speaker modeling



5.2 LLR Scoring



5.3 Construction of the SVM input vectors

Additionnal labelled development data, with T frames

$$T = t_1 \ldots t_j \ldots t_T$$

For each frame t_j , the score S_{t_j} is computed as follows :

$$S_{t_j} = \underset{g_i \in \lambda, \overline{\lambda}}{Max} \left[Log[P(t_j / g_i)] \right]$$

Two vectors $V_\lambda^X(\lambda)$, $V_\lambda^X(\overline{\lambda})$ are constructed as follows:

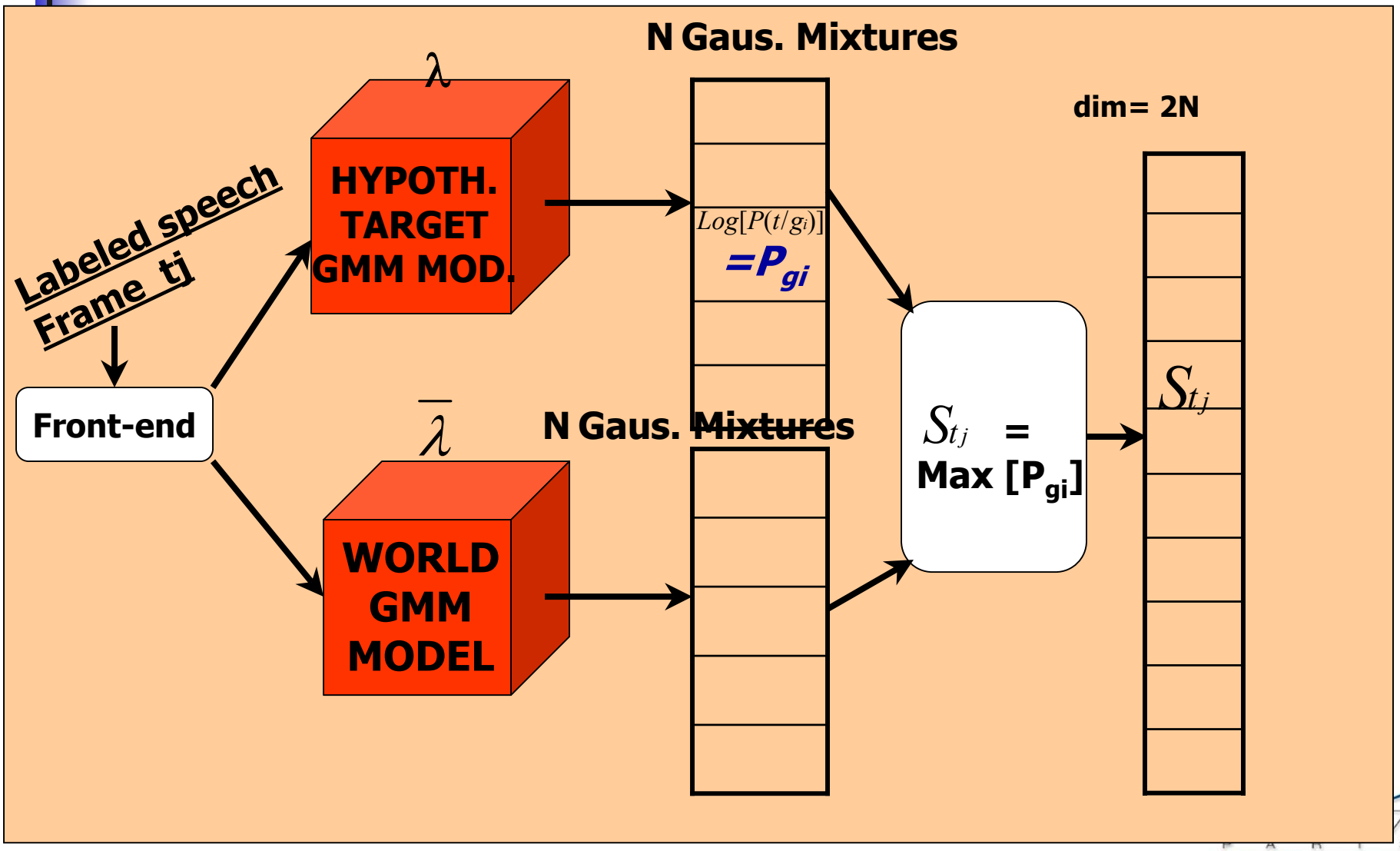
➔ First, all the components of the vectors are initialized to zero

➔ If S_{tj} is given by g_i belonging to λ , the i^{th} component of the vector $V_{\lambda}^{X}(\lambda)$ is incremented by the frame score. If S_{tj} is given by g_j belonging to $\overline{\lambda}$, the j^{th} component of the vector $V_{\lambda}^{X}(\overline{\lambda})$ is incremented by the frame score .

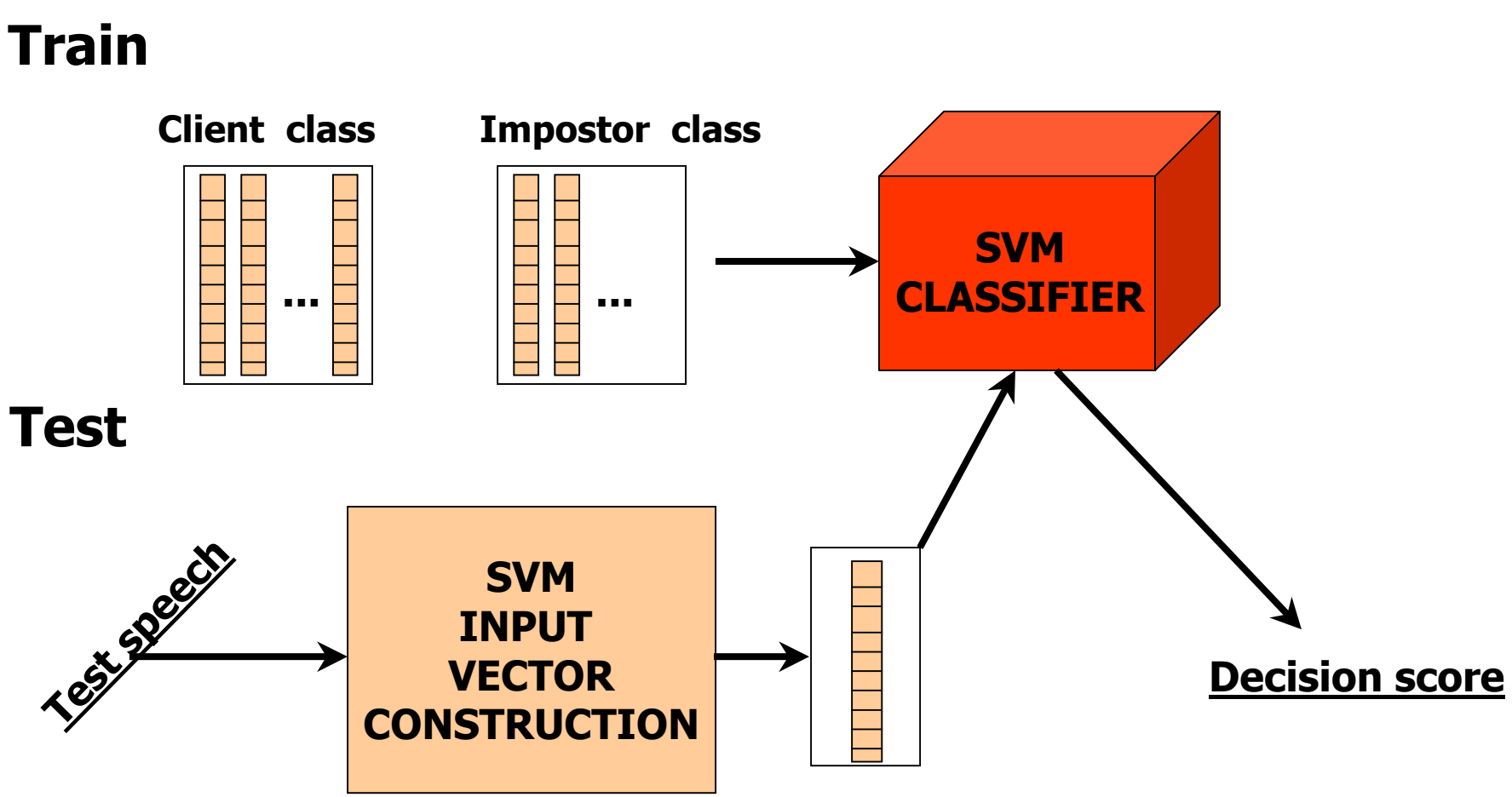
- The input SVM vector is the concatenation of $V_{\lambda}^{X}(\lambda)$ $V_{\lambda}^{X}(\overline{\lambda})$
- Summation and normalization of the SVM input vector by the number of frames of the test segment T

$$S_T = \left[\sum_{j=1}^{T} S_{tj} \right] / T$$

5.3 SVM Input Vector Construction



5.4 SVM : Train /Test



6. Database

Complete Nist'99 evaluation data splitted in :

- Development data = 100 speakers
  - ➔2min GMM model
  - ➔Corresponding test data to train the SVM classifier (519 true and 5190 impostor accesses)
- World data = 200 speakers
  - ➔4 sex/handset dependent world models
- Pseudo-impostors = 190 sp. used for the h-norm
- Evaluation data = 100 speakers = 449 true and 4490 impostor accesses

7. Experimental Protocol: 7.1 Feature Extraction

- LFCC parametrization (32.5 ms windows every 10 ms)
- Cepstral mean substraction for channel compensation
- Feature vector dimension is 33 (16 cep, 16 dcep, Δ log E) (Delta cepstral features on 5-frames windows)
- Frame removal algorithm applied on feature vectors to discard non significant frames (bimodal energy distributions)

7.2 GMM Modeling

- Speaker and background models
  - ➔GMM's with 128 mixtures
  - ➔Diagonal covariance matrix
  - ➔Standard EM algorithm with a max. of 20 iterations

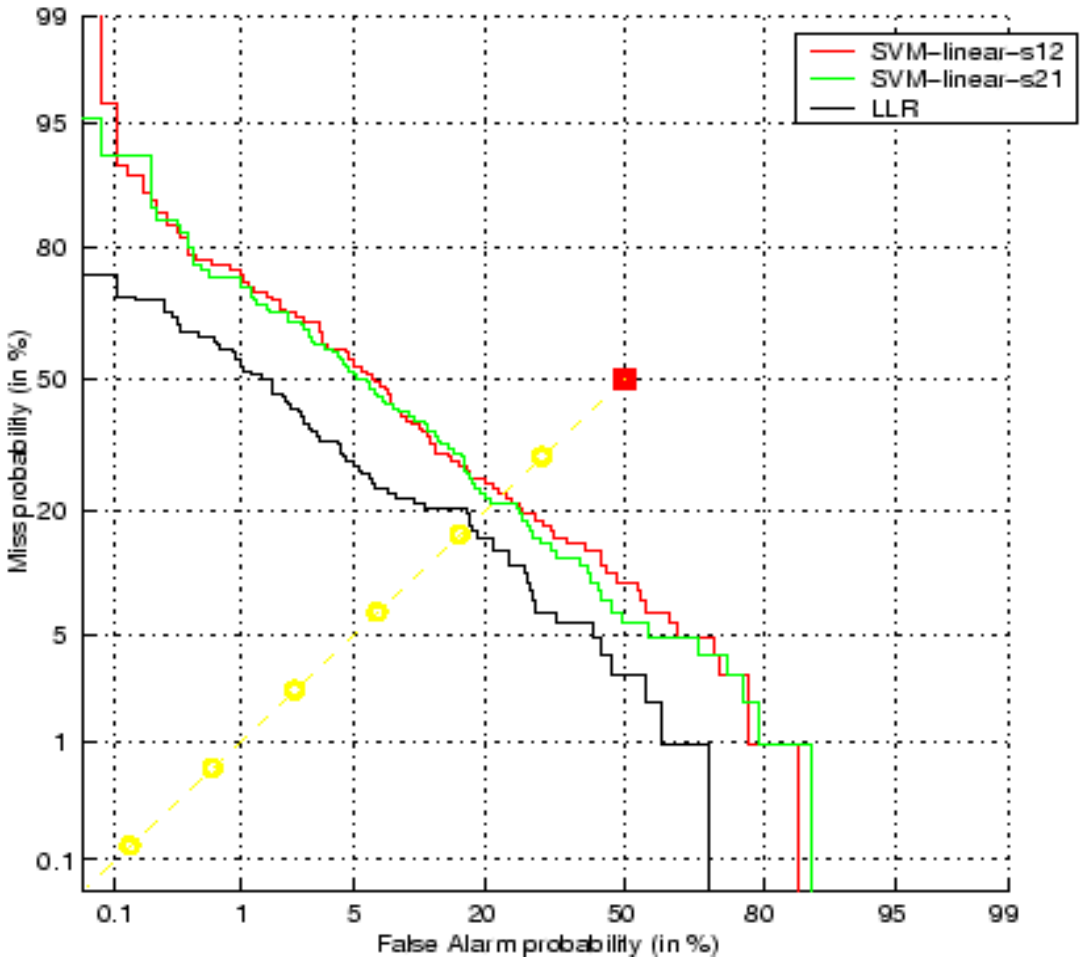
=> Four speaker-independent, gender and handset dependent background (world) models

7.3 SVM Scoring

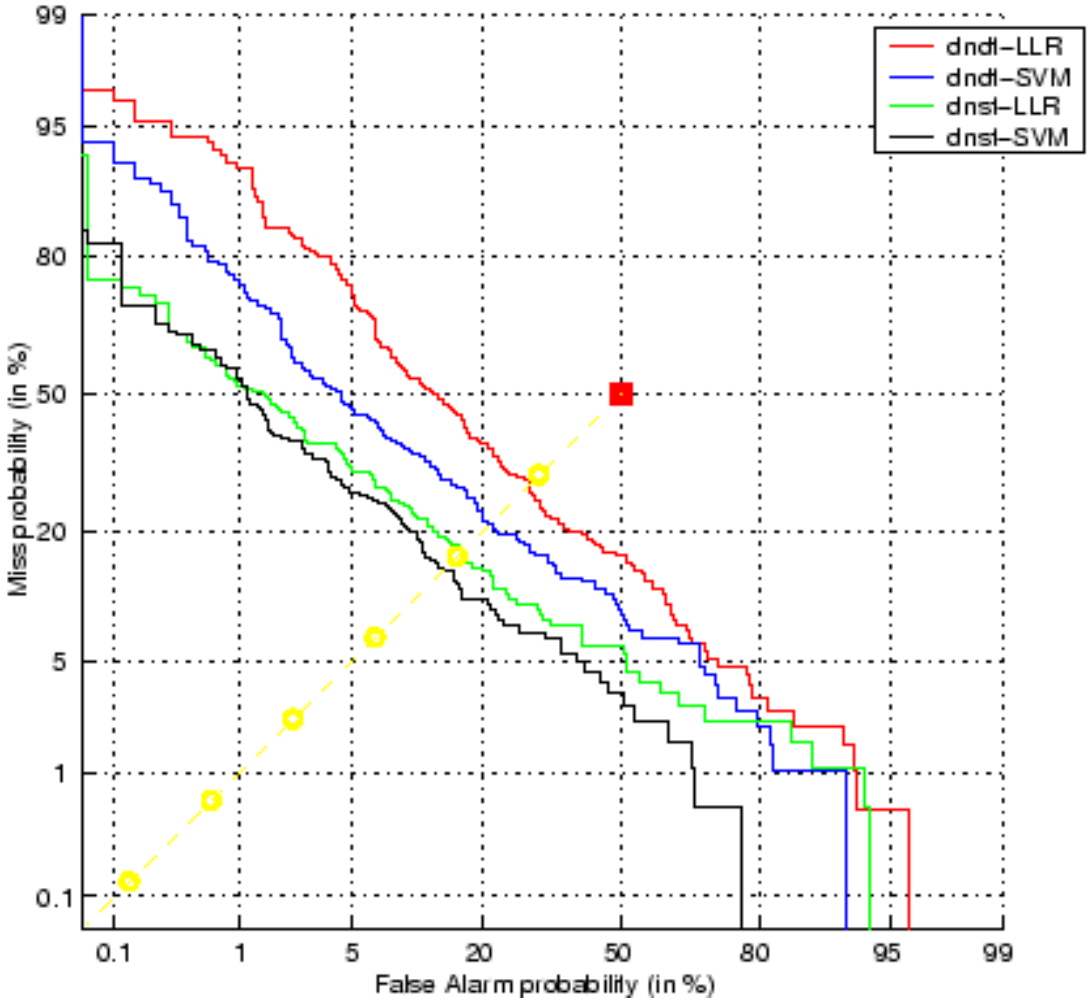
- SVM model was trained using a development corpus (coming from the NIST'99 database)
- Linear kernel is used
- There are 519 true-target speakers accesses and 5190 impostors accesses
- 5489 tests on the evaluation corpus (449 true-target speakers accesses and 4490 impostors accesses)

8.1 Results – preliminary results

SVM trained with feature vectors used as input vectors – condition all



8.2 SVM and LLR scoring

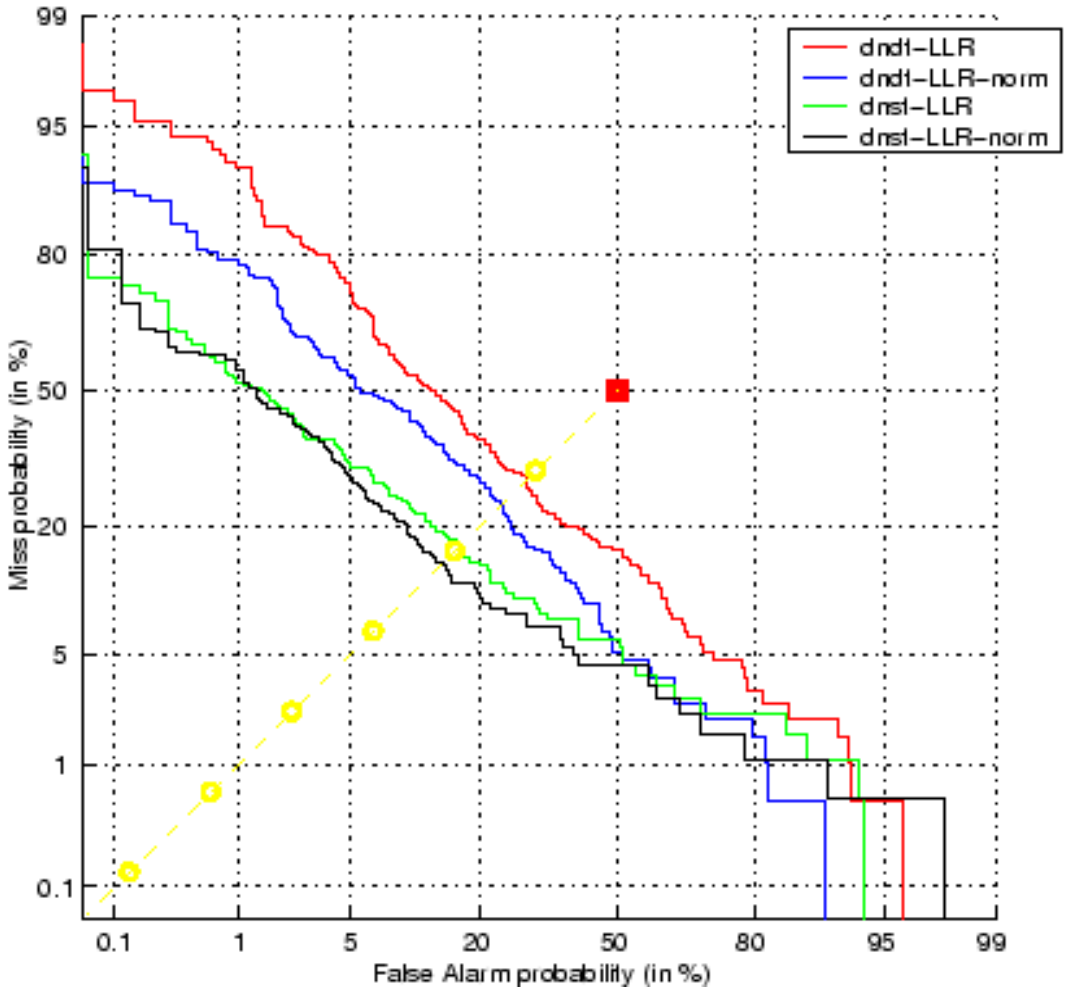


dndt =
different Nu,
different type,

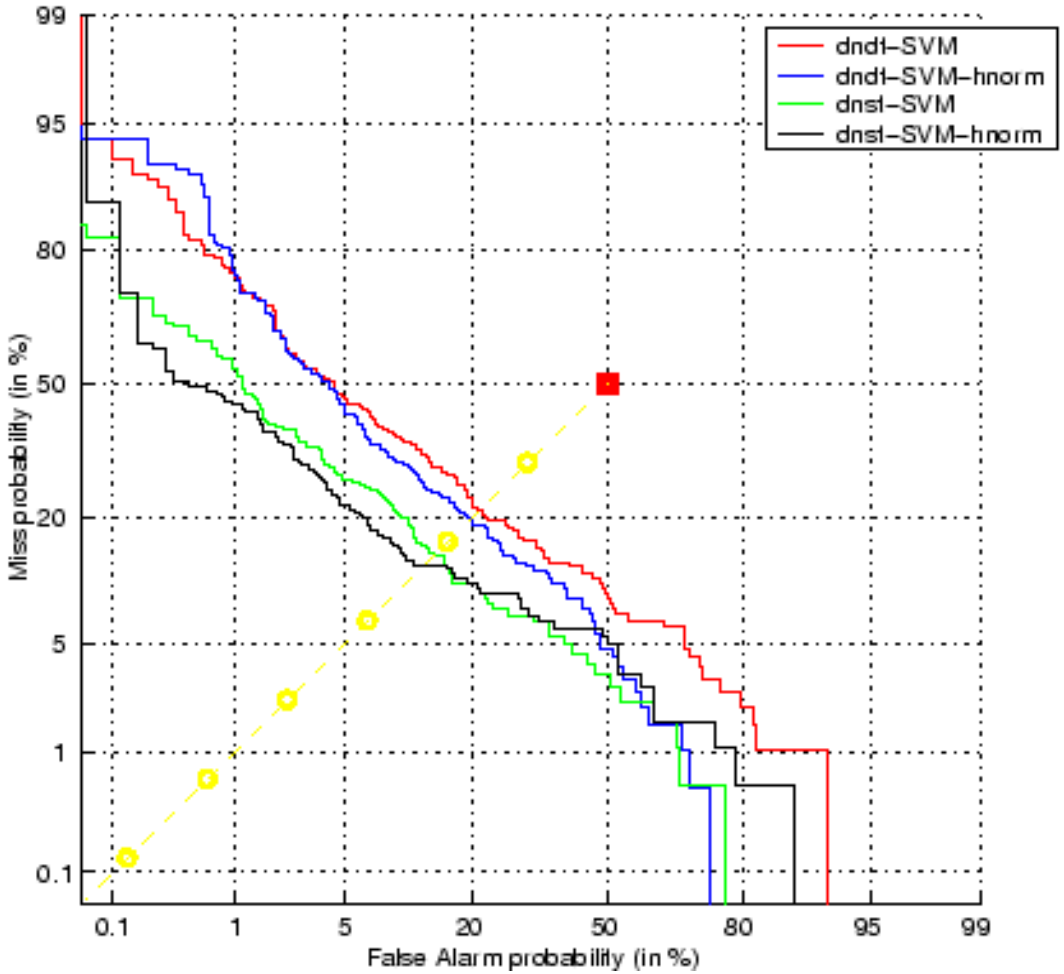
dnst =
different Nu,
same type

no normalization

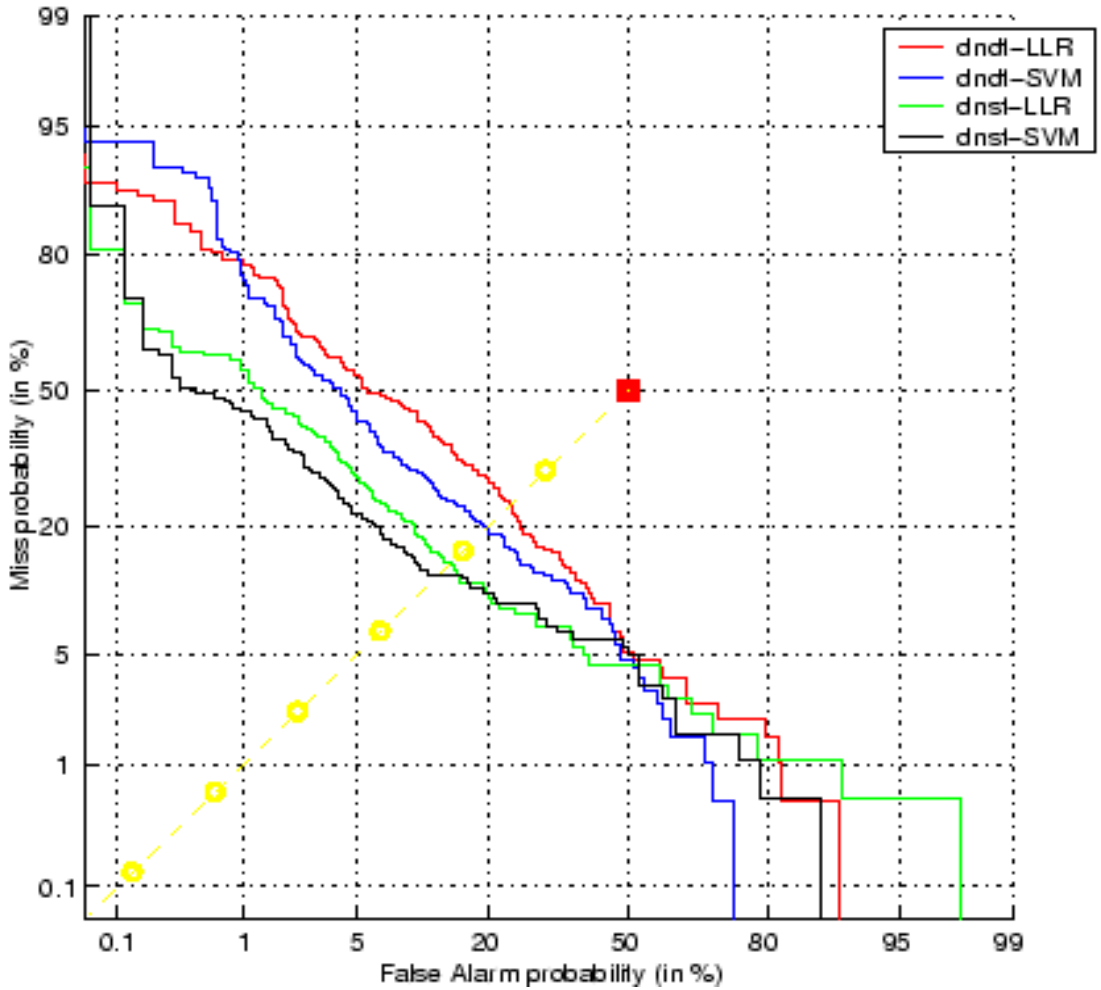
8.3 LLR - Influence of h-horm



8.3 SVM - Influence of h-horm



8.3 SVM – LLR comparison



8.4 Results table at EER

	DNST		*DNDT*	
	LLR	***SVM***	***LLR***	***SVM***
no normalization	17.6 %	15.8 %	27.8 %	21.6 %
h-norm	15.2 %	14.0 %	23.3 %	20.5 %

9. Conclusions

- Better results with GMM-SVM method in all the experimental conditions tested
- Proposed method seems to be more robust to channel variations

10. Perspectives

- Different kernel types and features will be experimented
- Other normalization techniques
- Another feature representation will be experimented to use the SVM in SV:

$$V_{\lambda}^{X}(\lambda) = [P(X/g_1^{\lambda}), .., P(X/g_n^{\lambda})]$$

$$V_{\lambda}^{X}(\overline{\lambda}) = [P(X/g_1^{\overline{\lambda}}), .., P(X/g_n^{\overline{\lambda}})]$$