IDIAP RESEARCH REPORT

Experimental Protocol on the BANCA Database

Samy Bengio [1] Frédéric Bimbot [2]
Johnny Mariéthoz [3] Vlad Popovici [4]
Fabienne Porée [5] Enrique Bailly-Baillière [6]
George Matas [7] Belen Ruiz [8]

IDIAP–RR 02-05

March 27, 2002

Dalle Molle Institute for Perceptual Artificial Intelligence • P.O.Box 592 • Martigny • Valais • Switzerland

phone +41 − 27 − 721 77 11
fax +41 − 27 − 721 77 12
e-mail secretariat@idiap.ch
internet http://www.idiap.ch

1 IDIAP, CP 592, 1920 Martigny, Switzerland, bengio@idiap.ch
2 IRISA - Campus Beaulieu, 35042 Rennes, France, bimbot@irisa.fr
3 IDIAP, CP 592, 1920 Martigny, Switzerland, marietho@idiap.ch
4 EPFL, LTS, 1915 Lausanne, Switzerland, Vlad.Popovici@epfl.ch
5 IRISA - Campus Beaulieu, 35042 Rennes, France, fporee@irisa.fr
6 University Carlos III,Madrid, Spain, ebbg@terra.es
7 University of Surrey, g.matas@eim.surrey.ac.uk
8 University Carlos III,Madrid, Spain, bruiz@inf.uc3m.es

IDIAP Research Report 02-05

Experimental Protocol on the BANCA Database

Samy Bengio Frédéric Bimbot Johnny Mariéthoz Vlad Popovici
Fabienne Porée Enrique Bailly-Baillière George Matas Belen Ruiz

March 27, 2002

1 Presentation

This document describes the experimental set up used in the BANCA project in conducting experiments in *Identity Verification* (IV) on the BANCA multi-modal database. This protocol was designed by the research partners for assessing algorithms and methods used for uni-modal IV (voice verification and face verification) as well as for evaluating procedures for the fusion of the two modalities (voice and face).

2 Task and System Overview

IV can be defined as the task that consists in verifying the identity X claimed (explicitly or implicitly) by a person U, using a *sample* y from this person, for instance an image of the face of U, a speech signal produced by U... By comparing the sample to some *template* (or *model*) of the claimed identity X, the IV system outputs a decision of *acceptance* or *rejection*.

The process can be viewed as a hypothesis testing scheme, where the system has to decide within the following alternative:

- U is the *true client* (acceptance, denoted $\hat{X}$),
- U is an *impostor* (rejection, denoted $\hat{\bar{X}}$).

In practice, an IV system can produce 2 types of errors:

- False Acceptance (FA) if the system has wrongly accepted an impostor,
- False Rejection (FR) if a true client has in fact been rejected by the system.

In practical applications, these 2 types of error have an associated cost, which will be denoted as C_{FA} and C_{FR} respectively.

IV approaches are usually based on the characterisation of hypotheses X and $\bar{X}$ by a client template and a non-client template respectively, which are learned during a *training* (or enrolment) phase[1]. Once the template for client X has been created, the system becomes operational for verifying identity claims on X. In the context of performance evaluation, this is referred to as the *test* phase.

Conventionally, the procedure used by an IV system during the test phase can be decomposed as follows:

- *feature* extraction, i.e. transformation of the raw sample into a (usually) more compact representation,
- *score* computation, i.e. output of a numerical value $S_X(y)$ based on a (normalized) resemblance of Y with the templates for X (and $\bar{X}$),
- *decision* by comparing the score $S_X(y)$ to a threshold Θ, independent of X.

3 Structure of the BANCA database

The BANCA database was designed in order to test multi-modal IV with various acquisition devices (2 cameras and 2 microphones) and under several scenarios (controlled, degraded and adverse).

For 5 different languages[2], video and speech data were collected for 52 subjects (26 males and 26 females), i.e. a total of 260 subjects. Each language - and gender - specific population was itself subdivided into 2 groups of 13 subjects (denoted $g1$ and $g2$). Table 1 below summarizes the structure of the database.

[1]The non-client model may even be trained during a preliminary phase, also called *installation* phase.

[2]English, french, german, italian and spanish

Total (260)	Language (52)		Gender (26)		Group (13)
	English	L_1	Female	$p = 1$	Group $g1$
					Group $g2$
			Male	$p = 2$	Group $g1$
					Group $g2$
	French	L_2	Female	$p = 3$	Group $g1$
					Group $g2$
			Male	$p = 4$	Group $g1$
					Group $g2$
	German	L_3	Female	$p = 5$	Group $g1$
					Group $g2$
			Male	$p = 6$	Group $g1$
					Group $g2$
	Italian	L_4	Female	$p = 7$	Group $g1$
					Group $g2$
			Male	$p = 8$	Group $g1$
					Group $g2$
	Spanish	L_5	Female	$p = 9$	Group $g1$
					Group $g2$
			Male	$p = 10$	Group $g1$
					Group $g2$

Table 1: BANCA Database.

Each subject participated to 12 recording sessions, each of these sessions containing 2 records: 1 true *client access* (T) and 1 informed[3] *impostor attack* (I). The 12 sessions were separated into 3 different scenarios:

- *controlled* (c) for sessions 1-4,
- *degraded* (d) for sessions 5-8,
- *adverse* (a) for sessions 9-12.

Two cameras were used, a cheap one and an expensive one. The cheap camera was used in the degraded scenario, while the expensive camera was used for controlled and adverse scenarios. Two microphones, a cheap one and an expensive one, were used simultaneously in each of the three scenarios. During the recordings, the camera was placed on the top of the screen and the two microphones were placed in front of the monitor and below the subject chin.

Table 2 contains a description of the 12-session structure for a given subject.

Scenario									
Scenario c (controlled)	Session	1		2		3		4	
	Records	T	I	T	I	T	I	T	I
Scenario d (degraded)	Session	5		6		7		8	
	Records	T	I	T	I	T	I	T	I
Scenario a (adverse)	Session	9		10		11		12	
	Records	T	I	T	I	T	I	T	I

Table 2: Session description for the 3 scenarios. For each user (in both development and test sets), the following sequence of sessions/records are available (a "T" record is a genuine client access, while an "I" record is an impostor access).

In a given session, the impostor accesses by subject X were successively made with a claimed identity corresponding to each other subject from the *same group* (as X). In other words, all the subjects in group g recorded one (and only one) impostor attempt against each other subject in g and each subject in group g was attacked once (and only once) by each other subject in g. Moreover, the sequence of impostor attacks was designed so as to make sure that each identity was attacked exactly 4 times in the 3 different conditions (hence 12 attacks in total).

In the rest of the document (cf Annexe A), we shall use the following notations:

X_i^g	:	subject i in group g	$g \in \{g1, g2\}, \quad i \in [1, 13]$
$y_k(X)$	:	true client record from session k by subject X	$k \in [1, 12]$
$z_l(X)$	:	impostor record (from a subject X') claiming identity X during a session l (with $X' \neq X$)	$l \in [1, 12]$

For each language, an additional set of 30 other subjects, 15 males and 15 females, recorded one session (audio and video). This set of data is referred to as *world data*. These individuals claimed two different identities, recorded by both microphones. World data were collected for each language in conditions which are specified in the document describing the database.

Finally, any data outside the BANCA database will be referred to as *external data*.

[3]The actual speaker knew the text that the claimed identity speaker was supposed to utter.

4 File naming convention

All the records of the BANCA database follow a unique naming convention which should simplify the use of scripts to apply different protocols. The convention goes as follows:

- For audio files of records:

  `<id>_<gender>_<group>_<session>_<claimed_id>_<lang>_<mic>.wav.gz`

- For video files of records:

  `<id>_<gender>_<group>_<session>_<claimed_id>_<lang>_<shot>.ppm.gz`

where:

`<id>` uniquely identifies the subject (4 characters),

`<gender>` is either 'm' (male) or 'f' (female),

`<group>` is either 'wm' (world model), 'g1' (odd-numbered group), or 'g2' (even-numbered group),

`<session>` identifies the session: s01 to s12,

`<claimed_id>` identifies the identity claimed during the access (4 characters),

`<lang>` is the language: 'en'=English, 'fr'=French, 'it'=Italian, 'sp'=Spanish, 'ge'=German,

`<mic>` is 1 for the high quality microphone, 2 for the low quality one,

`<shot>` identify the video frame (1-5).

5 Experimental requirements

For defining an experimental protocol, it is first necessary to define a set of evaluation data (or *evaluation set*), and to specify, within this set, which are to be used for the training phase and which are to be used for the test phase.

Moreover, before becoming operational, the development of an IV system requires usually the adjustment of a number of configuration parameters (model size, normalization parameters, decision thresholds, etc.). It is therefore necessary to define a *development set*, on which the system can be calibrated and adjusted, and for which it is permitted to use the knowledge of the actual subject identity during the test phase. Once the development phase is finished, the system performance can then be assessed on the evaluation set, without using the knowledge of the actual subject identity during the test phase. To avoid any methodological flaw, it is essential that the development set is composed of a distinct subject population from the one of the evaluation set.

In order to carry realistic (and unbiased experiments), it is necessary to use different speaker populations and data sets for development and for evaluation. We distinguish further between 2 circumstances: single-modality evaluation experiments and multi-modality evaluation experiments.

In the case of single-modality experiments, we need to distinguish only between two data sets: the development set, and the evaluation set. However, in the case of multi-modality experiments, it is necessary to introduce a third set of data: the *(fusion) tuning set* used for tuning the fusion parameters, i.e. the way to combine the outputs of each modality. If the tuning set is identical to the development set, this may introduce a bias in the estimation of the tuning parameters (*biased* case). An other solution is to use three distinct sets for development, tuning and evaluation (*unbiased* case).

Table 3 describes the BANCA protocol for the 3 cases mentioned above: in the unbiased case, we prescribe the use of data from other languages as development data.

	Option	Development Set	Fusion Tuning Set	Evaluation Set
One-modality Experiment	A	$g1$	N/A	$g2$
	B	$g2$	N/A	$g1$
Biased Multi-modality Experiment	A	$g1$	$g1$	$g2$
	B	$g2$	$g2$	$g1$
Unbiased Multi-modality Experiment	A	any data except $\{g1, g2\}$	$g1$	$g2$
	B		$g2$	$g1$

Table 3: Description of the development set for the evaluation on a given couple of groups ($g1$, $g2$), $\forall\ 1 \leq p \leq 10$ as defined in Table 1.

6 Experimental configurations

In the BANCA protocol, we consider that the true client records for the first session of each condition is reserved as training material, i.e. record T from sessions 1, 5 and 9. In all our experiments, the client model training (or template learning) is done on at most these 3 records.

We then consider 7 distinct training-test configurations, depending on the actual conditions corresponding to the training and to the testing conditions.

- Matched controlled (Mc):
    - client training from 1 controlled session
    - client and impostor testing from the other controlled sessions (within the same group)
- Matched degraded (Md):
    - client training from 1 degraded session
    - client and impostor testing from the other degraded sessions (within the same group)
- Matched adverse (Ma):
    - client training from 1 adverse session
    - client and impostor testing from the other adverse sessions (within the same group)
- Unmatched degraded (Ud):
    - client training from 1 controlled session
    - client and impostor testing from degraded sessions (within the same group)
- Unmatched adverse (Ua):

- client training from 1 controlled session
- client and impostor testing from adverse sessions (within the same group)

- Pooled test (P):

  - client training from 1 controlled session
  - client and impostor testing from all conditions sessions (within the same group)

  Note that the scores $S_X(y)$ necessary for this experiment can be obtained directly from experiments Mc, Ud and Ua.

- Grand test (G):

  - client training from 1 controlled, 1 degraded and 1 adverse sessions
  - client and impostor testing from all conditions sessions (within the same group)

From the comparison of these various performances, it is possible to measure:

- the intrinsic performance in a given condition
- the degradation from a mismatch between controlled training and uncontrolled test
- the performance in varied conditions with only one (controlled) training session
- the potential gain that can be expected from more representative training conditions.

Different protocols are presented in Table 4. Annex A presents each protocol in a more detailed way.

Test / Train	C: 2,3,4	C: 6,7,8	C: 10,11,12	C: 2,3,4,6,7,8,10,11,12
	I: 1,2,3,4	I: 5,6,7,8	I: 9,10,11,12	I: 1,2,3,4,5,6,7,8,9,10,11,12
1	Mc	Ud	Ua	P
5		Md		
9			Ma	
1,5,9				G

Table 4: Description of all protocols in the same table in fonction of train and test session numbers.

Let us note that:

1. These configurations are applicable to each type of microphone and to each type of experiment A and B (see Table 3).

2. We can define protocols P and G as *primary protocols, and the others as secondary protocols.*

3. *In P, the client training has already been performed during protocols M or U.*

4. *In G, for client training, there is a solution which does not need new computation.*

7 File format for scores

Score values (i.e. $S_X(y)$, the resemblance measure between the template for client X and the sample y, before it is used in the decision module) appears as a canonical quantity: some scores are common to various experimental configuration in single-modality experiments. Moreover, the one-modality scores are used as input values for the fusion modules.

It is therefore desirable to store them as intermediate values.

All the scores of a given set (such as development set of Language L, or test set of Language L) shoud be given in one ASCII file containing one score per line. The syntax should be:

```
<id> <claimed_id> <access_file_name> <score>
```

where:

`<id>` is the real `<id>` of the access,

`<claimed_id>` is the claimed `<id>` of the access,

`<access_file_name>` is the name of the access file,

`<score>` is the score given by the verification algorithm.

8 Performance measures

In order to visualise the performance of the system, irrespectively of its operating condition, we use the conventional DET curve[4], which plots on a log-deviate scale the *False Rejection Rate* P_{FR} as a function of the *False Acceptance Rate* P_{FA}.

Traditionally, the point on the DET curve corresponding to $P_{FR} = P_{FA}$ is called EER (Equal Error Rate) and is used to measure the closeness of the DET curve to the origin. The EER value of an experiment is reported on the DET curve, to comply with this tradition.

We also measure the performance of the system for 3 specific operating conditions, corresponding to 3 different values of the Cost Ratio $R = C_{FA}/C_{FR}$, namely $R = 0.1$, $R = 1$, $R = 10$. Assuming equal *a priori* probabilities of genuine clients and impostor, these situations correspond to 3 quite distinct cases:

$R = 0.1$ $\rightarrow$ a FA is an order of magnitude less harmful than a FR
$R = 1$ $\rightarrow$ a FA and a FR are equally harmful
$R = 10$ $\rightarrow$ a FA is an order of magnitude more harmful than a FR.

When R is fixed and when P_{FR} and P_{FA} are given, we define the Weighted Error Rate (WER) as:

$$WER(R) = \frac{P_{FR} + R\, P_{FA}}{1 + R} \tag{1}$$

P_{FR} and P_{FA} (and thus WER) vary with the value of the decision threshold Θ, and Θ is usually optimised so as to minimise WER on the development set D:

$$\hat{\Theta}_R = \arg\min_{D} \; WER(R) \tag{2}$$

The *a priori threshold* thus obtained is always less efficient than the *a posteriori threshold* that optimises the WER on the evaluation set E itself:

$$\Theta^*_R = \arg\min_{E} \; WER(R) \tag{3}$$

[4]A. Martin et al., The DET Curve in Assessment of Detection Task Performance, In *EuroSpeech'97*, vol. 4, p. 1895-1898.

The latter case does not correspond to a realistic situation, as the system is being optimised with the knowledge of the actual test subjects identity on the evaluation set. However, it is interesting to compare the performance obtained with *a priori* and *a posteriori* thresholds in order to assess the reliability of the threshold setting procedure.

9 Result presentation

9.0.1 Baseline results: english, mic 1, protocol P

Name Expe 1

Language english

Modality speech mic 1

Protocol P $g1$, $g2$, males, females

Preprocessing spro (33 lfcc), bi-gaussian sil/speech models

World models Gmm (200), Ml(vfloor=0.6*V_g), gender dependant, language dependant

Client models Gmm, Map(0.5)

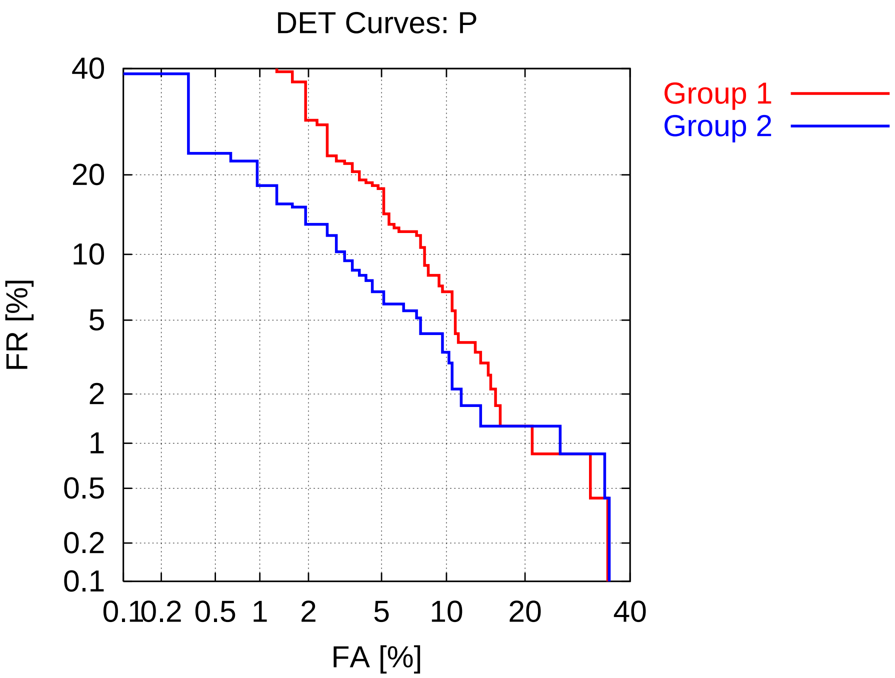

Figure 1: Det curves for experiments A and B.

			$R = 0.1$		$R = 1$		$R = 10$		EER	
A posteriori threshold	P_{FR}	P_{FA}	1.28	16.3	3.85	11.2	23.1	2.56	8.55	8.33
	WER		2.65		7.53		4.43		8.44	
A priori threshold	P_{FR}	P_{FA}	2.56	15.1	8.12	8.97	28.2	2.56	8.12	9.29
	WER		3.7		8.55		4.89		8.71	

Table 5: Group 1 (Experiment A).

			$R = 0.1$		$R = 1$		$R = 10$		EER	
A posteriori threshold	P_{FR}	P_{FA}	1.28	13.8	5.98	5.13	23.5	0.321	5.98	6.09
	WER		2.42		5.56		2.43		6.04	
A priori threshold	P_{FR}	P_{FA}	1.28	16	4.27	8.97	19.2	0.962	6.84	4.49
	WER		2.62		6.62		2.62		5.66	

Table 6: Group 2 (Experiment B).

A Detailed description of the different protocols

Tables 7, 8 and 9 describe more formally the 7 training-test configurations.

		Mc (use only controlled data)	Md (use only degraded data)	Ma (use only adverse data)
Client training	$\forall\ X_i \quad i = 1, ..., 13$ train with:	$y_k(X_i), \quad k = 1$	$y_k(X_i), \quad k = 5$	$y_k(X_i), \quad k = 9$
Non client training	All World data + any external data (it is forbidden to use other client data from the same group)			
Client testing	$\forall\ X_i \quad i = 1, ..., 13$ test with:	$y_k(X_i), \quad k = 2, 3, 4$	$y_k(X_i), \quad k = 6, 7, 8$	$y_k(X_i), \quad k = 10, 11, 12$
Impostor testing	$\forall\ X_i \quad i = 1, ..., 13$ test with:	$z_l(X_i), \quad l \in \{1, 2, 3, 4\}$	$z_l(X_i), \quad l \in \{5, 6, 7, 8\}$	$z_l(X_i), \quad l \in \{9, 10, 11, 12\}$
Number of tests per experiment		client: $13 \times 3 = 39$ impostor: $13 \times 4 = 52$	client: 39 impostor: 52	client: 39 impostor: 52
Total number of tests		client: $2 \times 10 \times 2 \times 39 = 1560$ impostor: $2 \times 10 \times 2 \times 52 = 2080$	client: 1560 impostor: 2080	client: 1560 impostor: 2080

Table 7: Description of protocols Mc, Md and Ma.

		Ud (use controlled data for training and degraded data for testing)	Ua (use controlled data for training and adverse data for testing)
Client training	$\forall\ X_i \quad i = 1, ..., 13$ train with:	$y_k(X_i), \quad k = 1$	$y_k(X_i), \quad k = 1$
Non client training	All World data + any external data (it is forbidden to use other client data from the same group)		
Client testing	$\forall\ X_i \quad i = 1, ..., 13$ test with:	$y_k(X_i), \quad k = 6, 7, 8$	$y_k(X_i), \quad k = 10, 11, 12$
Impostor testing	$\forall\ X_i \quad i = 1, ..., 13$ test with:	$z_l(X_i), \quad l \in \{5, 6, 7, 8\}$	$z_l(X_i), \quad l \in \{9, 10, 11, 12\}$
Number of tests per experiment		client: $13 \times 3 = 39$ impostor: $13 \times 4 = 52$	client: 39 impostor: 52
Total number of tests		client: $2 \times 10 \times 2 \times 39 = 1560$ impostor: $2 \times 10 \times 2 \times 52 = 2080$	client: 1560 impostor: 2080

Table 8: Description of protocols Ud and Ua.

		P (use controlled data for training and all data for testing)	G (use all data for training and all data for testing)
Client training	$\forall\ X_i \quad i = 1, ..., 13$ train with:	$y_k(X_i), \quad k = 1$	$y_k(X_i), \quad k = 1, 5, 9$
Non client training	All World data + any external data (it is forbidden to use other client data from the same group)		
Client testing	$\forall\ X_i \quad i = 1, ..., 13$ test with:	$y_k(X_i), \quad k = 2, 3, 4, 6, 7, 8, 10, 11, 12$	$y_k(X_i), \quad k = 2, 3, 4, 6, 7, 8, 10, 11, 12$
Impostor testing	$\forall\ X_i \quad i = 1, ..., 13$ test with:	$z_l(X_i), \quad l \in \{1, ..., 12\}$	$z_l(X_i), \quad l \in \{1, ..., 12\}$
Number of tests per experiment		client: $13 \times 9 = 117$ impostor: $13 \times 12 = 156$	client: 117 impostor: 156
Total number of tests		client: $2 \times 10 \times 2 \times 117 = 4680$ impostor: $2 \times 10 \times 2 \times 156 = 6240$	client: 4680 impostor: 6240

Table 9: Description of protocols P and G.