

# On-line signature verification resilience to packet loss in IP networks

Jonas Richiardi<sup>†</sup>, Julian Fierrez-Aguilar<sup>‡</sup>, Javier Ortega-Garcia<sup>‡</sup>,  
Andrzej Drygajlo<sup>†</sup>

<sup>†</sup> Speech Processing and Biometrics Group  
Signal Processing Institute  
Swiss Federal Institute of Technology  
Lausanne, Switzerland  
jonas.richiardi@epfl.ch

<sup>‡</sup> Biometrics Research Lab.  
Dept. Ing. Audiovisual y Comunicaciones  
Universidad Politecnica de Madrid  
Madrid, Spain  
jfierrez@diac.upm.es

## Abstract

This paper reports on experiments investigating the effects of packet loss for two online signature verification systems distributed over the Internet. The effects on verification performance of using different signature data recovery strategies are explored. Justifications are given taking into account the statistical nature of the signature models used for the verification task. Under realistic packet loss conditions, it is found that marginalising the lost feature vectors at the verification stage outperforms other feature-based data imputation methods such as packet repetition.

## 1. Introduction

Distributing software and hardware components of a biometric authentication system over hosts linked by a network introduces additional challenges compared to developing a centralised system. Among these, fault tolerance and network transfer resilience are of particular importance. Specifically, transmission of data over IP networks is not without flaws [1]: overflowing router queues, parallelised data streams, variability in traffic load, non-constant delay between packet arrivals, packet corruption, and many other causes can lead to data loss, packet reordering and transmission delays, depending on the transport layer protocol used.

It is thus crucial to the successful deployment of distributed biometric authentication systems that biometric data transmission problems be addressed and mechanisms for ensuring successful authentication in the event of transmission problems be built in.

An area that has seen much research in recent years, and from which knowledge can be transferred to the case of distributed biometrics, is that of streaming audio over IP networks. A large amount of data recovery methods have been developed to cope with network pathologies such as packet loss [2] for streaming audio. While many such methods (for instance silence substitution) primarily target minimisation of psychoacoustic distortion in the audio signal and are not necessarily applicable to biometrics, some others (such as packet repetition) are generic enough to be applied to many modalities.

The effect of packet loss on different biometric modalities is a current ongoing subject of investigation, notably for speech [3], and also for face [4]. However, the effects of IP network transmission on on-line signature verification

performance have not yet been tested. This contribution explores the effects of, and possible ways of coping with, data loss for distributed on-line signature verification systems.

The rest of this paper is organised as follows: Section 2 describes the signature verification systems used in our tests. Section 3 explains the network modelling process. Section 4 explains the different data recovery strategies implemented. Section 5 presents the experimental protocol and the error rates obtained with the different recovery strategies.

## 2. Signature verification systems

For the experiments reported in this paper, the HMM-based on-line signature verification system from Universidad Politecnica de Madrid competing in the First Intl. Signature Verification Competition (SVC 2004) has been used<sup>1</sup>. Below we briefly describe the basics; for more details about the use of HMM and GMM in signature verification, we refer the reader to [5, 6, 7].

**Feature extraction.** Coordinate trajectories  $(x_n, y_n)$  and pressure signal  $p_n$  are the components of the unprocessed feature vectors  $\mathbf{u}_n = [x_n, y_n, p_n]^T$  extracted from the signature signal, where  $n = 1, \dots, N_s$  and  $N_s$  is the duration of the signature in time samples. Signature trajectories are then preprocessed by subtracting the center of mass followed by a rotation alignment based on the average path tangent angle. An extended set of discrete-time functions are derived from the preprocessed trajectories consisting of sample by sample estimations of various dynamic properties. As a result, the parameterised signature  $\mathbf{O}$  consist in the sequence of feature vectors  $\mathbf{o}_n = [x_n, y_n, p_n, \theta_n, v_n, \rho_n, a_n, \dot{x}_n, \dot{y}_n, \dot{p}_n, \dot{\theta}_n, \dot{v}_n, \dot{\rho}_n, \dot{a}_n]^T$ ,  $n = 1, \dots, N_s$ , where the upper dot notation represents an approximation to the first order time derivative and  $\theta$ ,  $v$ ,  $\rho$ ,  $a$  stand respectively for path tangent angle, path velocity magnitude, log curvature radius and total acceleration magnitude. A whitening linear transformation is finally applied to each discrete-time function so as to obtain zero mean and unit standard deviation function values.

<sup>1</sup>See <http://www.cs.ust.hk/svc2004/>

**Similarity computation.** Given the parameterised enrollment set of signatures of a client  $\mathcal{C}$ , a left-to-right Hidden Markov Model (HMM)  $\lambda^{\mathcal{C}}$  is estimated by using the Baum-Welch iterative algorithm. No transition skips between states are permitted and multivariate Gaussian Mixture Model (GMM) state observation distributions are used. Given a test signature parameterised as  $\mathbf{O}$  (with a duration of  $N_s$  time samples) and the claimed identity  $\mathcal{C}$  represented by its signature model  $\lambda^{\mathcal{C}}$ , the matching score  $s$

$$s = \frac{1}{N_s} \log p(\mathbf{O}|\lambda^{\mathcal{C}}) \quad (1)$$

is computed by using the Viterbi algorithm. Matching scores are finally aligned between clients by using the a posteriori target-dependent score normalization technique based on individual EERs described in [8].

For the experiments carried out in this work two configurations of the above system are used, namely: *i*) High performance HMM verification system, where 2 states and 32 mixtures per state are considered; and *ii*) Robust GMM system, where 1 state and 64 mixtures are used.

### 3. Packet loss simulation

Simulating IP transmission based on a network pathology model rather than actually transmitting data over the Internet ensures careful control of experimental conditions and makes for relatively easier implementation, but is limited by the accuracy of the model. To enable different recognition systems to be tested, several degraded versions of the original on-line signature corpus were produced with built-in packet recovery strategies, at two different mean packet loss rates.

#### 3.1. Building packets for distributed on-line signature verification

The simulated transport protocol is the User Datagram Protocol (UDP) [9]. This transport protocol does not guarantee delivery or delivery order, but has the advantage over the Transmission Control Protocol (TCP) [10] of using smaller headers, at 8 bytes vs. 20 bytes (without options), and lower complexity. It is suited to real-time data delivery applications because it does not retransmit lost packets or adjust transmission rate according to network congestion. UDP transmission can be made more reliable by the use of a higher-level protocol such as the Real Time protocol (RTP) [11]. This is typically used in real-time speech applications as it facilitates the implementation of higher-level functionality such as sequence numbering and timestamping. An example of the IP/UDP/RTP combination is found in the H.323 video conferencing standard [12].

In the simulated distributed signature verification system, unprocessed signature feature vectors  $\mathbf{u}_n$  are packetised in groups of 10 per packet, leading to a payload of 200 bytes per packet. The simulated IP and UDP headers add respectively 20 and 8 bytes. In addition, the sending of a 60-byte proprietary file header containing the total signature data payload size, packet sequence

number, and other information with every packet is simulated. While this is certainly suboptimal and the file header could be much simplified, at 288 bytes, the overall packet size is below the minimum reassembly buffer size of 576 bytes mandated by IPv4 [13], thus ensuring that any conforming implementation will accept the packet. Furthermore, at 288 bytes the packet size stays well below the 1500 bytes Maximum Transmission Unit (MTU) of Ethernet links, which is often equivalent to the path MTU between two hosts [14], ensuring that packets would not be fragmented and thus ensuring more accurate packet loss modelling. This assumption could however prove wrong depending on the particular routing conditions, as Internet routing is often asymmetric [15]. Lastly, in real applications the payload may have to be encrypted to prevent covert acquisition.

#### 3.2. IP network modelling

A large body of literature exists on packet dynamics in IP networks. A model that is simple yet has been shown to capture essential properties of IP traffic, such as loss burstiness, is the two-state Markov model (also called Gilbert model). Higher order models [16], or segmented two-state Markov models [17] are better descriptors of IP traffic, at the cost of higher complexity. The experiments presented here use a simple two-state Markov model to simulate various amounts of mean packet loss rates over a UDP connection, using realistic model values derived from large-scale Internet packet traces [18, 19], with the assumptions that:

1. These model parameters apply to a wider range of users than those using high-speed internet connections found in academic institutions. The route used in [18] could be atypical for non-academic users.
2. The signature packet sizes are small enough that network parameters estimated with smaller packets can apply.

The two-state  $s = \{loss, noloss\}$  Markov chain is entirely defined by the two parameters  $gl$ , the probability of transitioning to the *loss* state from the *no loss* state, and  $sl$ , the probability of staying in the loss state (also called *clp* for conditional loss probability), which controls the burstiness of the loss process. Thus the transition matrix  $\mathbf{T}$  for the Markov chain is:

$$\mathbf{T} = \begin{pmatrix} 1 - gl & 1 - sl \\ gl & sl \end{pmatrix} \quad (2)$$

The mean packet loss rate, or steady-state probability of being in the *loss* state  $\bar{P}(s = loss)$ , also called unconditional loss probability (*ulp*), can be computed as:

$$\bar{P}(s = loss) = ulp = \frac{gl}{1 - sl + gl} \quad (3)$$

It was chosen to experiment with “medium” and “high” packet loss with parameters as shown in Table 1. It can be observed that these values are in accordance with the observation made in [18] that the conditional loss probability  $sl$  is higher than the unconditional loss probability  $ulp$ . This can be attributed to the fact that neighbouring packets are more likely to see correlated

router buffer states, which is one of the causes of loss burstiness.

loss condition name	<i>ulp</i>	<i>gl</i>	<i>sl</i>
“medium”	12.5%	0.1	0.3
“high”	25.6%	~0.14	0.6

Table 1: *Two-state Markov model parameters*

## 4. Signature data recovery strategies

Assuming packets have been lost, different strategies can be put in place to minimise the impact of data loss. A useful framework for dealing with missing or unreliable data is based on missing feature theory [20], where the approaches can be roughly divided into two: Either ignore the missing features (marginalisation), or try to recover them with estimated values (imputation). Packet loss can be construed as an extreme case of missing features, where all components of a feature vector are lost simultaneously.

From a statistical pattern recognition standpoint, packet loss causes a random number of consecutive feature vectors in the original sequence of unprocessed feature vectors  $\mathbf{U}$  to be lost. If no data recovery strategy is implemented, the received sequence of observation vectors  $\tilde{\mathbf{U}}$  will be shorter than the original. Because of the random nature of the loss process, the lost feature vectors could be ones that represent a significant part of their state’s observation distribution. Therefore, the corrupted received signature  $\tilde{\mathbf{O}}$  may have a lower likelihood given the estimated probability density function (PDF) for each state.

Additionally, the segmentation of  $\tilde{\mathbf{O}}$  effected by the Viterbi algorithm could be sub-optimal compared to the case where no feature vectors are missing, leading to decreased likelihoods: the observation probability of non-lost observations following lost observations will be computed with respect to the emission probability of earlier HMM states, e.g.  $P(\mathbf{o}_5|state = j)$  could be computed as  $P(\tilde{\mathbf{o}}_5|state = j - 1)$  for the corrupted observation sequence.

The effect of sub-optimal segmentation is not expected to be very pronounced for HMMs with a low number of states (1 or 2 for those described in Section 2): it is expected that statistical distribution of features will be more important than timewise alignment.

Finally, it is probable that different recovery strategies are best suited to different signature features. For instance, for latin script signatures, which generally progress horizontally from left to right, the  $x$  coordinate on average increases with time. Thus, linear or low-order polynomial interpolation could be a good approximation in case of packet loss. The pressure  $p$  however does not behave linearly and thus linear interpolation will not necessarily be a good strategy for this feature.

### 4.1. No recovery

With the “no recovery” strategy, lost packets are not replaced. The signature length is shortened by transmission. In the verification process, feature vectors contained in lost packets are marginalised.

### 4.2. Zero-substitution and corpus mean substitution

A classic and straightforward technique used in streaming speech applications is to replace lost packets with zero values [2], which play out as silence. The suitability of this approach for signature verification is tested. Except for the pen pressure value, zeros do not normally occur in the training or testing corpora. Thus, while the length of the signature is preserved, many outliers are introduced. This will introduce a severe bias in the feature normalisation stage.

An enhancement on zero-substitution can be made by estimating means across all users for each feature from the training corpus. This is similar to unconditional mean replacement [21].

### 4.3. Packet repetition

Packet repetition consists of replacing lost packets by the last received packet before the loss. In case the first packet is lost, the next received packet is used to fill the gap instead. This technique has the advantage of not introducing outliers to the feature distributions, but will bias the mean and variance estimates of each state’s observation distribution by over-representing the values contained in the repeated packet. In preliminary experiments [22], this recovery strategy has been shown to be substantially more effective than zero-substitution for a “medium” amount of packet loss.

### 4.4. Linear interpolation

The linear interpolation recovery strategy replaces lost packets with a linear interpolation between the last feature vector  $\tilde{\mathbf{u}}_l$  of the last packet received before the loss gap and the first feature vector  $\tilde{\mathbf{u}}_f$  of the first packet received after the loss gap. The  $n$ th estimated feature vector  $\hat{\mathbf{u}}_n$  for a total number of lost feature vectors  $N_l$  is computed according to

$$\hat{\mathbf{u}}_n = \tilde{\mathbf{u}}_l + n \frac{\tilde{\mathbf{u}}_f - \tilde{\mathbf{u}}_l}{N_l} \quad (4)$$

where  $n = 1, \dots, N_l$ . If the first or last packet in the signature are lost,  $\tilde{\mathbf{u}}_l$ , respectively  $\tilde{\mathbf{u}}_f$ , are set to be equal to a training corpus mean vector computed as per Sec. 4.2.

## 5. Experiments

### 5.1. Signature database and experimental protocol

50 users have been randomly selected from the UPM contribution to the MCYT Bimodal Database [23]. The following training and testing strategy is used:

**Training:** Each signature is modelled with 5 samples from a single set. Each contributor in the MCYT Database provides 5 sets of 5 signatures each and he/she is asked to forge other clients between sets, so inter-set variability is obtained [7].

**Testing:** *i*) Clients: the remaining 20 samples from each client are considered as client trials; *ii*) Impostors: 5 different impostors for each client are considered and, for each impostor, 5 signature samples are used. Skilled forgeries with natural dynamics are

considered for the experiments (i.e., contributors are requested to sign naturally without breaks or slowdowns).

Some signature examples from uncorrupted and degraded MCYT corpus are given respectively in Fig. 1 and Fig. 2.

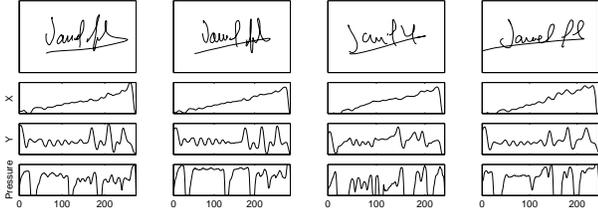


Figure 1: *Signature examples from the MCYT corpus. Two genuine signatures (left) and two skilled forgeries (right) are given for a random client.*

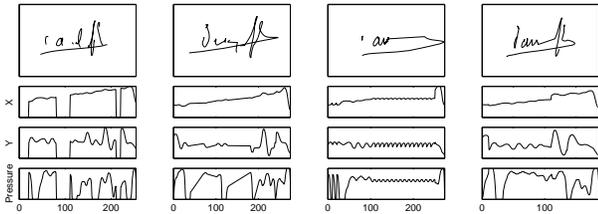


Figure 2: *Signature examples from degraded MCYT corpus under “high” loss. Recovery strategies from left to right: zero-substitution, linear interpolation, packet repetition and no recovery.*

As a result, each verification experiment described in the following consists of  $50 \times 20 = 1000$  client, and  $50 \times 5 \times 5 = 1250$  impostor trials. The baseline verification performance for the HMM system is an Equal Error Rate (EER) of 0.44%, while the GMM system has an EER of 0.89%. The results of the experiments are provided using Detection Error Tradeoff (DET) curves [24].

## 5.2. Recovery strategies evaluation

### 5.2.1. Zero-substitution and corpus mean substitution

As can be seen in Fig. 3, these strategies perform poorly compared to others. This can be expected because they essentially replace a dynamic signal with constant values, introducing many outliers in all replaced features; corpus mean substitution only minimally improves on zero-substitutions because the values are more probable. With the MCYT database used, corpus mean substitution for raw feature data is further made inappropriate because the signatures are acquired in sequence on a rectangular grid, making  $x$  and  $y$  features inconsistent across users or signature realisations. This underlines the need for recovery strategies to take into account acquisition conditions at every stage.

### 5.2.2. Linear interpolation

Linear interpolation performs substantially better than substitution of a fixed value. This is probably because

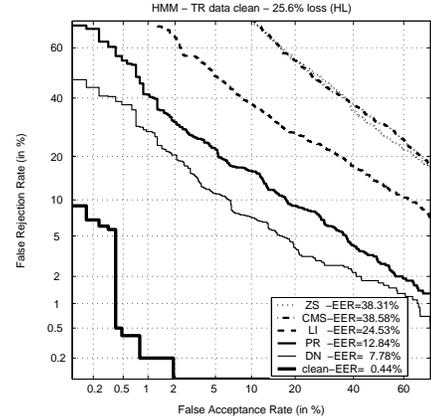
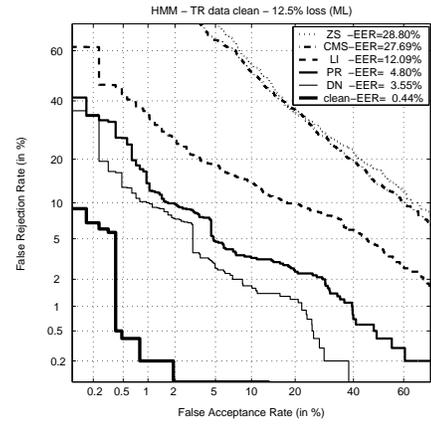


Figure 3: (top) *Verification performance of HMM system trained on clean data for different recovery strategies under “medium” packet loss conditions. (bottom) same system under “high” packet loss conditions. ZS: zero-substitution, CMS: corpus mean substitution, LI: linear interpolation, PR: packet repetition, DN: do nothing (no recovery), clean: baseline, ML: “medium” loss, HL: “high” loss*

the values imputed by Eq. 4 are more likely to appear in the uncorrupted signature (at least for  $x$  and  $y$ ). However, depending on the position of the lost packets, very discriminative information can be lost and replaced with poor approximations. Furthermore, the pressure  $p$  is subject to sudden drops and increases, making linear interpolation a very poor choice. This method introduces too many outliers, which cause the mean and variance of the corrupted signature to be strongly biased.

### 5.2.3. Packet repetition and no recovery

Packet repetition performs the best of all the imputation methods. By replicating existing feature vectors, no underlying data is added. However, the discontinuities caused by the packet boundaries are likely to modify first-order time derivatives.

The best method of all is the no recovery strategy, where lost packets are simply marginalised during ver-

ification. Remarkably, this result also holds for “high” packet loss rates, suggesting that unless the underlying individual features can be modelled for a general user population with good accuracy, imputing data is a risky process. This agrees with results for speaker verification in [25], namely, imputation often performs worse than marginalisation. This result however is largely dependent on the nature of the features extracted from the signal, their modelling, and correlation.

In light of these results, a strategy which would seem likely to provide appropriate values for replacement is user-dependent imputation with state sampling, where missing feature vectors would be replaced by a feature vector sampled from a given state’s observation distribution in a particular user’s model. However, for biometric applications this may not be desirable because parts of forged signatures could be replaced with data drawn from an authentic user’s model, thus potentially increasing the likelihood of the signature and the number of false acceptances. It is likely that many user model-based imputation techniques should be applied only with caution to the biometric case.

### 5.3. Training conditions evaluation

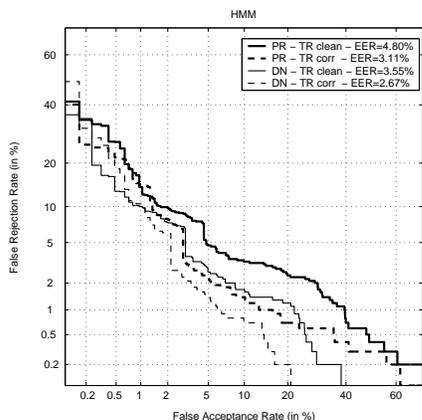


Figure 4: Impact of training conditions (clean data or corrupted data) on verification performance for HMM system under “medium” loss. Legend as per Fig. 3.

To evaluate other potential robustness techniques, the user models were trained with corrupted signatures. Matching training and testing conditions is a common technique in speech processing [26]. The results presented in Fig. 4 indicate that, for verifying signatures subjected to a similar amount of packet loss, this results in decreased error rates. However, it is likely that the verification performance of the corrupted models with clean test data would be decreased. Additionally, this result should be taken with care, as fixed network model parameters used to train user signature models cannot match the diversity of real, time-varying Internet conditions; it is likely that robustness should be achieved by other means.

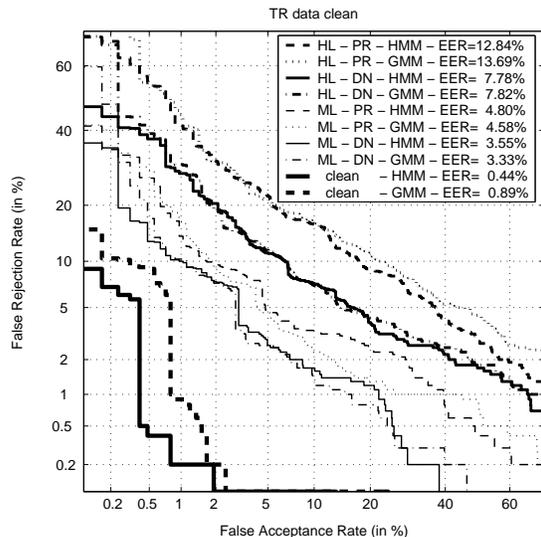


Figure 5: Verification performance comparing HMM and GMM systems with two strategies under “medium” and “high” loss. Legend as per Fig. 3.

### 5.4. Signature Models evaluation

As can be seen in Fig. 5, at EER the GMM and HMM systems perform very similarly for both “medium” and “high” loss conditions, using both packet repetition and no recovery. This is likely to be because, with packets lost, the HMM system loses the modelling advantage given by segmentation in the clean case. It can also be noted that, with good recovery strategies, the GMM system suffers from less performance drop (relative to no packet loss) than the HMM system.

## 6. Conclusions

The effects of packet loss in IP networks on distributed on-line signature verification have been investigated. It was found that even moderate amounts of loss can lead to serious degradations in error rates (one order of magnitude) unless corrective measures are taken. Even with the best data recovery strategy (“no recovery”) it was found that the error rate is significantly worse than in the no loss condition.

Therefore, it is important that distributed on-line signature verification systems ensure that packets are safely delivered. This can be achieved by using TCP “out of the box”, or by adding features to UDP to provide retransmission (for instance by using RTP fields to implement application-level retransmission). This would avoid overheads associated with TCP: connection establishment increases latency, transmission can be slow under high network congestion because TCP retransmits lost packets and adapts the rate of transmission to take into account network load<sup>2</sup>, and streaming of feature vectors for immediate use is made more difficult because of the re-ordering buffer.

Because it is also expected that out-of-order packet

<sup>2</sup>note that one full signature in the ATVS system weighs only about 13 KB on average

delivery would lower performance for HMM signature verification systems using more than one state, reordering mechanisms may be necessary. The trade-off between response times and application-layer complexity needs to be more fully evaluated in the context of real applications, taking into account system response time, usability factors, and modality under use.

## 7. Acknowledgements

This work has been supported by the Spanish Ministry of Science and Technology under project TIC2003-08382-C05-01 and by the COST 275 action in the framework of a short-term scientific mission. Julian Fierrez-Aguilar and Jonas Richiardi also thank Consejeria de Educacion de la Comunidad de Madrid and Fondo Social Europeo, respectively the Swiss Federal Office for Education and Science for supporting their doctoral research.

## 8. References

- [1] V. Paxson, "End-to-end Internet packet dynamics," *IEEE/ACM Trans. on networking*, vol. 7, pp. 277–292, June 1999.
- [2] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE network*, vol. 12, pp. 40–48, Sept.–Oct. 1998.
- [3] N. Evans, J. Mason, R. Auckenthaler, and R. Stappert, "Assessment of speaker verification degradation due to packet loss in the context of wireless mobile devices," in *Proc. COST275 Workshop on The Advent of Biometrics on the Internet*, (Rome, Italy), Nov. 2002.
- [4] I. Fratric, "Degradation of the XM2VTS face images database," COST275 STSM Report, Sept. 2003.
- [5] J. Ortega-Garcia, J. Fierrez-Aguilar, J. Martin-Rello, and J. Gonzalez-Rodriguez, "Complete signal modeling and score normalization for function-based dynamic signature verification," in *Proc. International Conference on Audio- and Video-Based Biometric Person Authentication 2003*, (Guildford, UK), pp. 658–667, 2003.
- [6] J. Richiardi and A. Drygajlo, "Gaussian mixture models for on-line signature verification," in *Proc. ACM Multimedia 2003 Workshop on Biometric Methods and Applications*, (Berkeley, USA), Nov. 2003.
- [7] J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez, "A function-based on-line signature verification system exploiting statistical signal modeling," *Intl. Journal on Image and Graphics*, 2004. (accepted).
- [8] J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez, "Target dependent score normalization techniques and their application to signature verification," in *Proc. of Intl. Conf. on Biometric Authentication 2004*, (Hong Kong), 2004. (accepted).
- [9] J. Postel, *RFC 768: User Datagram Protocol*, Aug. 1980.
- [10] J. Postel, *RFC 793: Transmission Control Protocol*, Sept. 1981.
- [11] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, *RFC 1889: RTP: a transport protocol for real-time applications*, Jan. 1996.
- [12] G. Thom, "H.323: the multimedia communications standard for local area networks," *IEEE Communications Magazine*, vol. 34, pp. 52–56, Dec. 1996.
- [13] J. Postel, *RFC 791: Internet Protocol - DARPA Internet program protocol specification*, Sept. 1981.
- [14] W. Stevens, *Unix network programming vol.1*. Prentice Hall, second ed., 1998.
- [15] V. Paxson, "End-to-end routing behavior in the internet," *IEEE/ACM Trans. on Networking*, vol. 5, pp. 601–615, Oct. 1997.
- [16] M. Yajnik, J. Kurose, and D. Towsley, "Packet loss correlation in the Mbone multicast network: experimental measurements and Markov chain models," UMMASS CMPSCI Technical Report 95-115, University of Massachusetts at Amherst, Dept. of Computer Science, 1996.
- [17] G. Nguyen, R. Katz, B. Noble, and M. Satyanarayanan, "A trace-based approach for modeling wireless channel behaviour," in *Proc. Winter Simulation Conference '96*, IEEE, 1996.
- [18] J.-C. Bolot, "Characterizing end-to-end packet delay and loss in the Internet," *J. High-Speed Networks*, vol. 2, pp. 305–323, Dec. 1993.
- [19] J.-C. Bolot, S. Fosse-Parisis, and D. Towsley, "Adaptive FEC-based error control for Internet telephony," in *Proc. Infocom '99*, pp. 1453–1460, 1999.
- [20] A. Drygajlo and M. El-Maliki, "Integration and imputation methods for unreliable feature compensation in GMM based speaker verification," in *Proc. "2001: A Speaker Odyssey"*, (Crete, Greece), pp. 107–112, June 2001.
- [21] M. Cooke, A. Morris, and P. Green, "Missing data techniques for robust speech recognition," in *Proc. IEEE Int'l Conf. on Acoustics, Speech, and Signal Processing 1997*, vol. 2, pp. 863–866, Apr. 1997.
- [22] J. Richiardi, "Resilience of on-line signature verification to packet loss on IP networks: preliminary experiments," COST275 STSM Report, Sept. 2003.
- [23] J. Ortega-Garcia, J. Fierrez-Aguilar, D. Simon, J. Gonzalez, M. Faundez-Zanuy, V. Espinosa, A. Satue, I. Hernaez, J.-J. Igarza, C. Vivaracho, C. Escudero, and Q.-I. Moro, "MCYT baseline corpus: a bimodal biometric database," *IEE Proc. Vision, Image and Signal Processing*, vol. 150, pp. 395–401, December 2003.
- [24] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of decision task performance," in *Proc. Eurospeech 1997*, pp. 1895–1898, 1997.
- [25] M. El-Maliki, *Speaker verification with missing features in noisy environments*. PhD thesis, Swiss Federal Institute of Technology, 2000.
- [26] G. Davis, *Noise reduction in speech applications*, ch. 10, pp. 245–275. Boca Raton, Florida, USA: CRC Press, 2002.