ITU - Telecommunication Standardization Sector

Study Period 1997-2000

Study Group **12**

Delayed Contribution **D.xxx**

Geneva, 24 November – 4 December 1998

Text available only in **E**

Question(s): **Q17/12, “Noise Aspects in Evolving Networks**”

SOURCE*: **McGill University - Canada**

TITLE: **Frame-Level Noise Classification in Mobile Environments**

Abstract

Background environmental noises degrade the performance of speech-processing systems (e.g. speech coding, speech recognition). By modifying the processing according to the type of background noise, the performance can be enhanced. This requires noise classification. In this work, four pattern-recognition frameworks have been used to design noise classification algorithms. Classification is done on a frame-by-frame basis (e.g. once every 20 ms). Five commonly encountered noises in mobile telephony (i.e. car, street, babble, factory, and bus) have been considered in our study. Our experimental results show that the Line Spectral Frequencies (LSFs), derived from the linear prediction coefficients, are robust features in distinguishing different classes of noises.

1. Introduction

In daily life, we encounter different types and levels of background acoustical noises (e.g. traffic noise, car noise, office noise etc.). Speech-processing systems (e.g. speech coding, speech recognition, speaker verification) pick up those “unwanted” signals along with speech. These noise signals result in performance degradation of those systems. For example, the accuracy of a speech recognition device might severely be affected if the level of noise is high and there is a mismatch between training and operation conditions [1]. In speech coding, background noises can be coded with

*** Contact:** Khaled El-Maleh and Peter Kabal
Department of Electrical & Computer Engineering
McGill University, Montreal, Quebec
H3A 2A7, Canada
Tel: (514) 398-7130
Fax: (514) 398-4470
E-mail: {khaled,kabal}@tsp.ece.mcgill.ca

annoying artifacts [2]. Noise classification can be used to reduce the effect of environmental noises on speech processing tasks. As an example, in variable bit rate speech coders, the lowest rate is used to encode background noises in non-active speech periods. As environmental noises vary in texture and dynamics, using a simple coding scheme has proven to be not adequate for many common types of noises. Noise classification can be used to design natural-quality multi-mode noise coding algorithms. Similarly, multi-mode comfort noise generators can be designed to remedy the noise contrast problem reported in discontinuous transmission-based cellular systems (i.e. GSM). A fuzzy-logic noise classifier was proposed in [3], as part of an object-oriented audio coder proposal to the MPEG-4. Recognition of the noise type was used in [4] to design a noise-independent speech recognition system.

Noise classification has been used in other applications. For example, in programmable hearing-aid devices, a classification algorithm automatically can match a program mode with the listening environment of the user [5]. In noise monitoring systems, classification of environmental noises can be done to help in controlling noise pollution [6].

In this report we present the results of our work in designing noise classification algorithms to be used as part of speech-processing systems in mobile environments.

The report is organized in five parts. Section 2 discusses the feature extraction module and the classification algorithms that have been used for our study. In Section 3, we review the performance evaluation tools we have used. Classification results from different tests of the classification algorithms are given in Section 4. Finally conclusions are presented in Section 5.

2. Frame-Level Noise Classification

2.1 Feature Extraction

The choice of signal features is usually based on *a priori* knowledge of the nature of the signals to be classified. Features that capture the temporal and spectral structure of the input signal are used. Examples of such features are zero crossing rate, root-mean-square energy, critical bands energies, and correlation coefficients. The classifier operates on a frame-by-frame basis using short segments of the signal, e.g. 20 ms.

Linear Prediction (LP) analysis is a major part of many modern speech-processing systems. Transformations of linear prediction coefficients (LPC) (e.g. cepstral, log-area ratio coefficients, line spectral frequencies) have been used successfully in many pattern-recognition problems (e.g. speech recognition, speaker recognition) [7].

We have experimented with different sets of features derived from both the LP coefficients and the LP residual (e.g. residual critical band energies, zero crossing rates). The line spectral frequencies (LSFs) gave the best class separability for the noises we considered. Moreover, a Gaussian fit to each LSF histogram was found to be quite good. Thus, we have selected the LSFs as our features for noise classification.

A 10th order LP analysis is performed every 20 ms using the autocorrelation method. A Hamming window of length 240 samples is used. The LP coefficients are calculated using the Levinson-Durbin algorithm and then bandwidth expanded using a factor η= 0.994. The LP coefficients are then converted to the LSF domain.

2.2 Classification Algorithms

Four pattern-recognition techniques have been chosen for our noise classification problem: Quadratic Gaussian Classifier (QGC), Least-Square Linear Classifier (LS-LC) [8], Nearest-Neighbor Classifier (NNC) [9], and Learning Vector Quantization (LVQ) [10]. A brief description of these classification algorithms is presented below.

A Gaussian classifier is based on the assumption that feature vectors of each class obey a multivariate Gaussian distribution. Estimates of the parameters of the Gaussian PDF of each class (mean and covariance) using the labelled training data are computed. In the classification stage, an input vector is mapped to the class with the largest likelihood. In linear classifiers, a linear discriminant function is optimized to maximize class separability. Least-square optimization algorithm is used to compute the coefficients (weights) of the linear function.

In nearest neighbor-type classifiers, for each input feature vector, a search is done to find the label of the vector in the dictionary of stored training vectors with the minimum distance. Euclidean distance is commonly used as the metric to measure neighborhood. In k-NN decision rule, the input feature vector is assigned the label most frequently represented among the k nearest patterns in the training dictionary. One major disadvantage of NN classifiers is the need to store large number of training vectors resulting in a large amount of computations. As a remedy to this problem, only prototype vectors from the training data are computed and stored (prototype nearest-neighbor classifier).

Learning vector quantization is an example of prototype nearest-neighbor classification. A set of L vectors (prototypes) is selected from the labelled training data to minimize the misclassification errors using nearest-neighbor decision rule. An initial set of L vectors is chosen from the training set. An iterative update rule is used to modify the vectors in such a way that achieves a better classification of the training set by the 1-NN rule based on the selected vectors. The final set of the L vectors defines the LVQ codebook to be used in the testing mode. The size of the codebook (L) and the distribution of the vectors amongst the classes are two free parameters to choose in the design process. For more details about LVQ-based classification see the excellent book by Kohonen [10].

3. Performance Evaluation

Five commonly encountered noise types were considered: car, voice babble, street, bus, and factory. A total of 56,250 frames (18.75 minutes), equally distributed between the 5 classes, were used for training. We have recorded street noise (traffic noise, pedestrians walking and talking, and noise from a nearby work area) and bus noise (background music, background speech, bus engine noise, and other external transient noises such as passing cars). The other noises are from the NOISEX-92 database [1]. We have used the **Tooldiag** pattern recognition software developed by Rauber [12] in designing and testing the QGC, LS-LC, and the k-NN classifiers. To design our LVQ-based classifier, we used the **LVQ_PAK** software package (version 3.1) developed by Kohonen *et al.* [11]. A codebook of size 256 vectors, equally distributed among the 5 noise classes, has been designed using both the OLVQ1 and the LVQ1 Algorithms [10].

To measure the discriminating power of the LSFs as features, we estimated the Bayes error rate. A lower bound on the Bayes error rate P_{Bayes} is a function of the asymptotic error rate of the nearest-neighbor decision rule P_{NN} [9], given as

$$P_{Bayes} \times \frac{M \,.\, 1}{M}\left(1 \,.\, \sqrt{1 \,.\, \frac{M}{M \,.\, 1} P_{NN}}\right) , \qquad (1)$$

where M is the number of classes.

This lower bound will be used as our reference point for the performance evaluation of the different designed classifiers. For each classification algorithm, we used a cross-validation testing methodology to evaluate the classification performance. Of the labelled frames, 30% selected at random were used as test vectors while the remaining vectors were used for classifier training. Five iterations of the Hold-out cross-validation method [8] were used to compute the empirical error rate for each classifier.

4. Classification Results

In some speech-processing tasks, we need to discriminate speech from noise. For example, a voice activity detection is used in some wireless communications systems to enhance systems capacity and prolong the battery life of portable units [13]. Thus, in this work we have considered two cases: noise-only classification (5 noise classes), and noise-and-speech classification (5 noise classes and speech class). A total of 750,000 frames, equally distributed between the 5 noise classes and speech, have been used for training in the noise-and-speech case. The classification algorithms were tested using 500 frames (different from the training data) for each class, and with other new noises.

4.1 Noise-only Classification

Table 1 gives the empirical error rate evaluated with the hold-out procedure for the various classifiers. Using Eq.(1) and the empirical error rate of the 1-NN classifier (19.8%), the Bayes error rate was estimated at 10.6%. This means that independent of the classifier structure, the best frame-level classification accuracy for the 5 selected noises (car, street, babble, bus, and factory), and with the 10 LSFs as features is around 87%. From Table 1, the quadratic Gaussian classifier (QGC) outperforms the other classifiers with 13.6% error rate. The linear and the nearest-neighbor classifiers are less accurate. For the remainder of the report, we will present results from the Gaussian classifier and compare its performance with the LVQ classifer, as an example of nearest neighbor classifiers.

Classifier	*Error Rate* *%*
Optimal Bayes	10.6
Quadratic Gaussian	13.6
3-Nearest Neighbor	17.5
Learning Vector Quantiziation	19.2
1-Nearest Neighbor	19.8
Linear (least-squares method)	21.9

Table 1 Empirical error rate for the different classifiers (noise-only)

A detailed presentation of the classification results for each class is given in the form of a classification matrix. Tables 2 and 3 show that the classification accuracy is different for each class. For example, accuracy ranging from 90–100% were obtained for car noise and factory noise. Street, babble, and bus noises are more often misclassified with error rates ranging from 20–40%. The Gaussian classifier is more robust to new test vectors than the LVQ classifier. This is mainly due to the parametric nature of the QGC and its ability to model well the LSFs feature vectors.

	Babble *%*	*Car* *%*	*Bus* *%*	*Factory* *%*	*Street* *%*
Babble	**79.8**	0.0	12.8	2.0	5.4
Car	0.0	**99.6**	0.2	0.2	0.0
Bus	8.8	0.0	**85.2**	2.2	3.8
Factory	1.0	0.0	5.6	**93.2**	0.2
Street	1.8	0.0	24.8	2.0	**71.4**

Table 2 Classification matrix (QGC) (noise-only)

	Babble ***%***	***Car*** ***%***	***Bus*** ***%***	***Factory*** ***%***	***Street*** ***%***
Babble	**73.8**	0.2	11.2	5.8	9.0
Car	0.0	**100.0**	0.0	0.0	0.0
Bus	7.8	0.0	**77.2**	8.6	6.4
Factory	1.4	0.0	6.2	**92.0**	0.4
Street	3.8	0.0	32.0	2.8	**61.4**

Table 3 Classification matrix (LVQ) (noise-only)

4.2 Classification of New Noises

In practical applications of noise classification, the input noise signals are not constrained to belong to one of the 5 pre-selected noise classes. Thus, a “good” noise classifier should have the ability to map an input feature vector from a new noise class to the closest pre-selected classes. We have tested the QGC on 5 new noise signals (restaurant, shopping mall, sports, subway, and traffic). The results are presented in Table 4. It is interesting to observe that the Gaussian classifier maps the new noises to the noise classes with the same noise events. As an example, a restaurant noise is composed of simultaneous conversations (babble noise), background music, and other ambient noises. Thus, restaurant noise was classified 82.4% as babble noise and 10.8% as bus noise (which has babble, and background music components).

	Babble ***%***	***Car*** ***%***	***Bus*** ***%***	***Factory*** ***%***	***Street*** ***%***
Resturant	**82.4**	0.0	**10.8**	4.0	2.8
Shop. Mall	**52.4**	0.0	2.4	0.0	**45.2**
Sports	**22.8**	0.0	6.2	0.0	**71.0**
Subway	**53.0**	0.0	**21.0**	4.0	**22.0**
Traffic	2.7	0.0	**15.2**	0.0	**82.1**

Table 4 Classification of new noises (QGC) (noise-only)

4.3 Noise-and-Speech Classification

In Tables 5–7, we present the classification results for the noise-and-speech case. Similar results to the noise-only case were obtained for the noises. The QGC outperforms the LVQ classifier in classifying speech, with more than 30% difference in accuracy. Speech signal is 91% accurately discriminated from the noises using the QGC. This suggests that the QGC classifier using LSFs as the features provides robust voice activity detection at the frame level.

Classifier	***Error Rate %***
Optimal Bayes	10.1
Quadratic Gaussian	13.6
3-Nearest Neighbor	16.2
1-Nearest Neighbor	18.9
Learning Vector Quantiziation	20.9
Linear (least-squares method)	33.8

Table 5 Empirical error rate for the different classifiers (noise-and-speech)

	Speech %	***Babble %***	***Car %***	***Bus %***	***Factory %***	***Street %***
Speech	**91.0**	7.4	0.0	0.8	0.2	0.6
Babble	4.2	**76.0**	0.0	12.4	2.0	5.4
Car	0.2	0.0	**99.6**	0.2	0.0	0.0
Bus	2.4	8.0	0.0	**84.0**	2.2	3.4
Factory	0.2	1.0	0.0	5.6	**93.0**	0.2
Street	0.2	1.8	0.0	24.6	2.0	**71.4**

Table 6 Classification matrix (QGC) (noise-and-speech)

	Speech ***%***	***Babble*** ***%***	***Car*** ***%***	***Bus*** ***%***	***Factory*** ***%***	***Street*** ***%***
Speech	**57.0**	30.0	2.2	4.2	1.8	4.8
Babble	0.2	**76.4**	0.0	8.6	3.8	11.0
Car	0.2	0.0	**99.8**	0.0	0.0	0.0
Bus	0.4	7.6	0.0	**78.0**	7.2	6.8
Factory	0.0	1.8	0.0	2.0	**96.0**	0.2
Street	0.0	3.0	0.0	26.0	3.6	**67.4**

Table 7 Classification matrix (LVQ) (noise-and-speech)

4.4 Classification of Human Speech-Like Noise

Human speech-like noise (HSLN) is a kind of babble noise generated by superimposing independent speech signals. HSLN of various number of superpositions (*N)* (1, 2, 4, …, 1024, 4096) were used in [14] to investigate perceptual discrimination of speech from noise. For example, for low number of superpositions (below 10), the resulting signal is perceived as speech-like. For *N* between 10 and 200 superpositions, the noise is perceived as babble-like. As *N* increases further, the noise starts to sound like stationary Gaussian-like noise (Central Limit Theorem). We have used this set of signals (150 frames each) to test our classifiers. The frame-level classification results are shown in Table 8 for the noise-only case and in Table 9 for the noise-and-speech case. For the noise-only case, the HSLN signals were classified as babble noise most of the time or as bus noise (note that bus noise has babble as one of its noise events). However, for the noise-and-speech case, the results are more interesting. For example, for *N*=1, the signal is classified as speech (86.8%) and as babble (9.3%). On the other hand, for 128 superpositions, the HSLN signal is classified as babble noise (88.7%) and as speech (5.3%). As *N* increases, the HSLN signal is classified more as babble than speech. These classification results clearly illustrate the robustness of the Gaussian classifier.

N	*Car* *%*	*Factory* *%*	*Street* *%*	*Bus* *%*	*Babble* *%*
1	0.7	6.6	0.0	0.7	92.0
4	0.0	0.0	0.7	4.6	94.7
8	0.0	0.0	3.3	11.3	85.4
16	0.0	0.0	3.3	7.3	89.4
32	0.0	0.0	2.0	4.0	94.0
128	0.0	0.0	0.7	5.3	94.0
512	0.0	0.0	3.3	8.0	88.7
1024	0.0	0.0	5.3	3.3	91.4
4096	0.0	0.0	4.6	10.6	84.8

Table 8 Classification of HSLN signals (QGC) (noise-only)

N	*Speech* *%*	*Car* *%*	*Factory* *%*	*Street* *%*	*Bus* *%*	*Babble* *%*
1	86.8	0.7	3.4	0.0	0.0	9.3
4	81.5	0.0	0.0	0.7	1.3	16.6
8	72.9	0.0	0.0	3.3	5.9	17.9
16	51.0	0.0	0.0	3.3	4.0	41.7
32	29.8	0.0	0.0	2.0	3.3	64.9
128	5.3	0.0	0.0	0.7	5.3	88.7
512	0.7	0.0	0.0	3.3	8.0	88.0
1024	1.3	0.0	0.0	5.3	3.3	90.1
4096	0.7	0.0	0.0	4.6	10.6	84.1

Table 9 Classification of HSLN signals (QGC) (noise-and-speech)

5. Conclusion

Frame-level noise classification results have been presented using four pattern-recognition frameworks. The line spectral frequencies have been used as the features. The quadratic Gaussian classifier (QGC) outperforms the other classifiers tested. The classification accuracy is different for each class, with accuracy ranging from 90–100% for car and factory noise, and with error rates ranging from 20–40% for street, babble, and bus noise. Speech signal is 91% accurately discriminated from the noises using the QGC. This suggests that the QGC classifier using LSFs as the features provides robust voice activity detection at the frame level. The accuracy can be improved substantially by postprocessing the temporal sequence of decisions (for instance with a Viterbi type algorithm), however this comes at the expense of further delay.

6. References

[1] A. Varga, and H. M. Steeneken, "Assessment for automatic speech recognition:II NOISEX-92: A database and an experiment to study the effect of additive noise on automatic speech recognition systems," *Speech Communication*, 12 (1993), pp. 247–251.

[2] T. Wigren, A. Bergstrom, S. Harrysson, F. Jansson, and H. Nilsson, "Improvements of background sound coding in linear predictive speech coders," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Detroit, MI), pp. 25–29, May 1995.

[3] F. Beritelli, S. Casale, and M. Russo, "A pattern classification proposal for object-oriented audio coding in MPEG-4, to appear in the *International Journal on Telecommunication Systems*, Special Issue on Multimedia.

[4] W. C. Treurniet and Y. Gong, "Noise independent speech recognition for a variety of noise types," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing* (Adelaide, Australia), pp. 437–440, April 1994.

[5] J. M. Kates, "Classification of background noises for hearing-aid applications," *J. Acoust. Soc. Amer*., vol. 97, pp. 461–470, Jan. 1995.

[6] C. Couvreur and Y. Bresle, "A statistical pattern recognition framework for noise recognition in an intelligent noise monitoring system," *Proc. EURONOISE'95* (Lyon, France), pp. 1007–1012, Mar. 1995.

[7] C.-S. Liu and M.-T. Lin, "Study of line spectrum pair frequencies for speaker recognition," *Proc. IEEE Int. Conf. on Acoustics, Speech, Signal Processing*, (Albuquerque, NM), pp. 277–280, Apr. 1990.

[8] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press, 1990.

[9] T. M. Cover and P. E. Hart, "Nearest-neighbor pattern classification," *IEEE Trans Inform. Theory*, vol. 13, pp. 21–27, Jan. 1967.

[10] T. Kohonen, *Self-Organizing Maps*, Springer Series in Information Sciences, 2nd ed., 1997.

[11] *LVQ_PAK: The Learning Vector Quantization Package*, Version 3.1, 1995, Helsinki University of Technology, Finland.

[12] T. Rauber, *Inductive Pattern Classification: Methods-Features-Sensors*, Ph.D. Thesis, Universidade Nova de Lisboa, 1994.

[13] K. El-Maleh and P. Kabal, “Comparison of voice activity detection algorithms for wireless personal communications systems,” *Proc. of the IEEE Canadian Conference in Electrical and Computer Engineering*, (St. John’s Nfld, Canada), pp. 470–475, May 1997

[14] D. Kobayashi, S. Kajita, K. Takeda, and F. Itakura, “Extracting speech features from human speech like noise,” *Proc. Int. Conf. on Spoken Language Processing*, pp. 418–421, Oct. 1996