# A VECTOR QUANTIZER CLASSIFIER FOR BLIND SIGNAL TO NOISE RATIO ESTIMATION OF SPEECH SIGNALS

*Russell Ondusko*[1], *Matthew Marbach*[2], *Ravi P. Ramachandran*[3], *Linda M. Head*[4] *and Mark C. Huggins*[5]

1. Rowan University, ondusk13@students.rowan.edu
2. Rowan University, marbac50@students.rowan.edu
3. Rowan University, ravi@rowan.edu
4. Rowan University, head@rowan.edu
5. Airforce Research Laboratory, Mark.Huggins@rl.af.mil

## ABSTRACT

A blind approach for estimating the signal to noise ratio (SNR) of a speech signal corrupted by additive noise is proposed. The method is based on a pattern recognition paradigm using various linear predictive based features and a vector quantizer classifier. Blind SNR estimation is very useful in speaker identification systems in which a confidence metric is determined along with the speaker identity. The confidence metric is partially based on the mismatch between the training and testing conditions of the speaker identification system and SNR estimation is very important in evaluating the degree of this mismatch. The aim is to correctly estimate SNR values from 0 to 30 dB, a range that is both practical and crucial for speaker identification systems. Additive white Gaussian noise is investigated. The best features are the line spectral frequencies, reflection coefficients and the log area ratios. The linear predictive cepstrum also shows great promise. The average SNR estimation error is 1.6 dB.

## 1. INTRODUCTION

Consider a speech signal corrupted by additive noise that is statistically independent of the signal. This noisy signal is characterized by a signal to noise ratio (SNR) calculated over the entire duration of the signal. In this paper, a pattern recognition approach using various linear predictive (LP) [1] derived features is used to blindly estimate the SNR of the noisy speech signal. Blind estimation of the SNR is very useful in closed set speaker identification systems. The training of a speaker identification system involves the configuration of $M$ models each representing a different speaker. During closed set testing, the features of an utterance are compared to the $M$ models to render a decision of the speaker identity as being one of the $M$ speakers [2][3]. Recent research has been done to develop techniques to calculate a confidence metric to accompany the decision of the speaker identity [4][5]. The confidence metric is calculated based on the mismatch between training and testing conditions, amount of training and testing data, and number of speakers (value of $M$). As $M$ increases, there is usually more model overlap. The more the difference between the SNR of the training and testing speech, the more the mismatch between the two and the lower the confidence metric. An automatic and blind method of SNR estimation of the training and testing speech is an integral part of the technique of finding the confidence metric of a speaker identification system.

The method proposed for blind SNR estimation is based on a pattern recognition paradigm just like what is used for speaker identification. Features based on LP analysis that would not be robust to noise are highly useful candidates for SNR estimation as they show differences for varying noise levels. The overall system consists of three components, namely, (1) Linear predictive (LP) analysis, (2) Feature extraction for ensuring SNR discrimination and (3) Vector quantizer (VQ) classifier and decision logic for computing the SNR. During training, a VQ codebook is trained for each distinct SNR value using feature vectors obtained from noisy speech corresponding to that particular SNR. During testing, the input to the system will be a noisy speech signal with an unknown SNR. After LP analysis and feature extraction, the set of feature vectors will be passed through each VQ codebook to get an overall distance for each codebook. Based on these distances, the output will be an estimated SNR value. A comparison of different LP based features is done with respect to the average absolute error between the actual and estimated SNR. The features considered [1][6][7] include the line spectral frequencies (LSFs), reflection coefficients (REFL), log area ratios (LAR), linear predictive cepstrum (CEP), adaptive component weighted cepstrum (ACW) and the postfilter cepstrum (PFL).

## 2. FEATURE EXTRACTION

Linear predictive analysis results in a stable all-pole model $1/A(z)$ of order $p$ where

$$A(z) = 1 - \sum_{n=1}^{p} a(n) z^{-n} \tag{1}$$

The autocorrelation method of LP analysis gives rise to the predictor coefficients $a(n)$ and the REFL feature $refl(n)$ for $n = 1$ to $p$. The LAR feature is found as

$$lar(n) = \log\left[\frac{1 - refl(n)}{1 + refl(n)}\right] \tag{2}$$

for $n = 1$ to $p$. The LSF feature $lsf(n)$ are the angles (between 0 and $\pi$) of the alternating unit circle roots of $F(z)$ and $G(z)$ [1] where

$$\begin{aligned} F(z) &= A(z) + z^{-(p+1)} A(z^{-1}) \\ G(z) &= A(z) - z^{-(p+1)} A(z^{-1}) \end{aligned} \tag{3}$$

The predictor coefficients $a(n)$ are converted to the LP cepstrum $clp(n)$ $(n \geq 1)$ by an efficient recursive relation [1]

$$clp(n) = a(n) + \sum_{i=1}^{n-1} (\frac{i}{n}) clp(i) a(n-i) \tag{4}$$

Since $clp(n)$ is of infinite duration, the CEP feature vector of dimension $p$ consists of the components $clp(1)$ to $clp(p)$ which are the most significant due to the decay of the sequence with increasing $n$.

The first step in developing the ACW cepstrum [6] is to perform a partial fraction expansion of the LP function $1/A(z)$ to get

$$\frac{1}{A(z)} = \sum_{n=1}^{p} \frac{r_n}{1 - p_n z^{-1}} \tag{5}$$

where $p_n$ are the poles of $A(z)$ and $r_n$ are the corresponding residues. The variations in $r_n$ were removed by forcing $r_n = 1$ for every $n$. Hence, the resulting transfer function is a pole-zero type of the form

$$\begin{aligned} \frac{N(z)}{A(z)} &= \sum_{n=1}^{p} \frac{1}{1 - p_n z^{-1}} \\ &= \frac{1}{A(z)} \sum_{n=1}^{p} \prod_{i=1 \neq n}^{p} (1 - p_n z^{-1}) \\ &= p\left[\frac{1 - \sum_{n=1}^{p-1} b(n) z^{-n}}{1 - \sum_{n=1}^{p} a(n) z^{-n}}\right] \end{aligned} \tag{6}$$

Applying the recursion in Eq. (4) to $b(n)$ and $a(n)$ results in two cepstrum sequences $cb(n)$ and $clp(n)$ respectively. The ACW cepstrum is $cacw(n) = clp(n) - cb(n)$ [6].

The postfilter is obtained from $A(z)$ and its transfer function is given by

$$H_{pfl}(z) = \frac{A(z/\beta)}{A(z/\alpha)} \tag{7}$$

where $0 < \beta < \alpha \leq 1$. The cepstrum of $H_{pfl}(z)$ is the PFL cepstrum which is equivalent to weighting the LP cepstrum as $cpfl(n) = clp(n)[\alpha^n - \beta^n]$ [7]. The ACW feature $cacw(n)$ and PFL feature $cpfl(n)$ are taken from $n = 1$ to $p$.

The CEP, REFL and LAR feature vectors decrease in norm as the SNR decreases while the LSF vector components become more equally spaced between 0 and $\pi$. Figure 1 shows the 12 dimensional REFL feature vector for SNRs of 30, 15 and 0 dB. The decrease in norm is mainly due to the amplitude shrinkage of the first few components. The ACW and PFL features were originally formulated to be robust to channel and noise effects for application to speaker recognition [6][7]. However, the plan is to compare these two features to the other LP features that vary more with noise to understand their role in SNR estimation.

### 3. VQ CLASSIFIER AND DECISION LOGIC

A vector quantizer (VQ) classifier is used to generate a score for each candidate SNR value. During training, speech is corrupted by additive noise with a particular SNR and a corresponding set of feature vectors are computed. The feature vectors are used to design a VQ codebook for the particular SNR based on the Linde-Buzo-Gray algorithm [8]. The squared Euclidean distance is the distance measure. There will be $N$ codebooks, one pertaining to each candidate SNR value.

During testing or score determination, a test noisy speech utterance of a particular SNR is converted to a set of test feature vectors. Consider a particular test feature vector. This is quantized

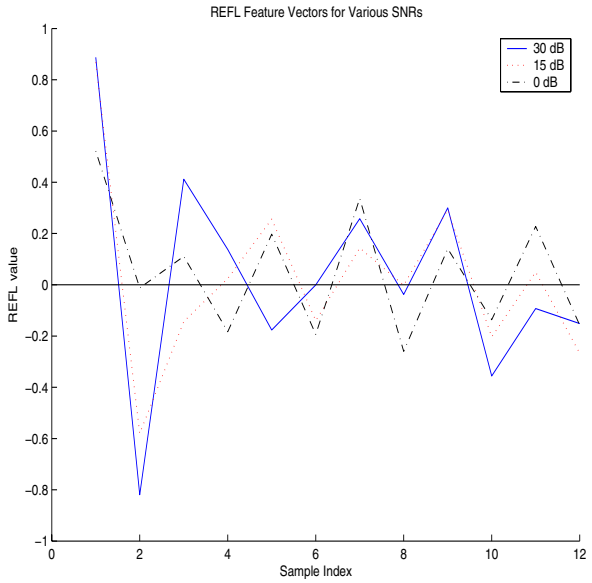


Figure 1: REFL feature vectors for SNRs of 30, 15 and 0 dB

by each of the $N$ codebooks. The quantized vector is that which is closest with respect to the squared Euclidean distance measure to the test feature vector. Hence, $N$ different distances are recorded, one for each codebook. This process is repeated for every test feature vector. The distances are accumulated over the entire set of feature vectors. This accumulated distance is the score for each codebook.

Two methods of implementing the decision logic are investigated. A hard decision approach estimates the SNR to correspond to the codebook which renders the smallest accumulated distance. This smallest distance is the best score. In the soft decision approach, the scores from a subset of the $N$ codebooks are used to estimate the SNR. Consider the $i$th codebook trained for the value SNR$(i)$ and rendering a score (accumulated distance) Score$(i)$. Let Ind$(i)$ denote the indicator function which equals 1 if codebook $i$ is used for SNR computation. Otherwise, Ind$(i)$ equals 0. The number of codebooks used which is also the number of times that Ind$(i)$ equals 1 is denoted by $C$. A probability Prob$(i)$ is derived from Score$(i)$ by the equations

$$\begin{aligned} \text{Total} &= \sum_{j=1}^{N} \text{Ind}(j)\text{Score}(j) \\ \text{Prob}(i) &= \text{Ind}(i)\left[\frac{\text{Total} - \text{Score}(i)}{(C-1)\text{Total}}\right] \end{aligned} \tag{8}$$

For the considered codebooks, smaller distances are converted to higher probabilities. If a codebook is not used, the probability assumes a value of 0. The probabilities add up to 1. The experiments revealed that using the three codebooks ($C = 3$) with the smallest accumulated distances (best scores) led to good results. From the probabilities, the SNR is estimated as

$$\text{SNR} = \sum_{j=1}^{N} \text{Prob}(j)\text{SNR}(j) \tag{9}$$

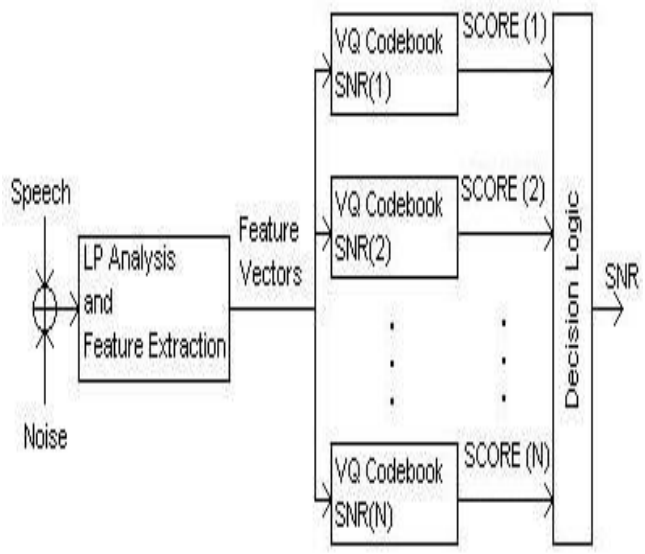


Figure 2: Block diagram for SNR Estimation

| Feature | Codebook Increment | Hard Decision | Soft Decision |
|---|---|---|---|
| LSF | 5 dB | 2.02 dB | 2.35 dB |
| LSF | 3 dB | 1.87 dB | 1.78 dB |
| LSF | 1 dB | 1.76 dB | 1.61 dB |
| CEP | 5 dB | 2.09 dB | 2.33 dB |
| CEP | 3 dB | 1.96 dB | 1.82 dB |
| CEP | 1 dB | 1.85 dB | 1.68 dB |
| REFL | 5 dB | 2.07 dB | 2.21 dB |
| REFL | 3 dB | 1.92 dB | 1.72 dB |
| REFL | 1 dB | 1.84 dB | 1.62 dB |
| LAR | 5 dB | 2.08 dB | 2.23 dB |
| LAR | 3 dB | 1.94 dB | 1.71 dB |
| LAR | 1 dB | 1.83 dB | 1.59 dB |
| ACW | 5 dB | 2.35 dB | 2.33 dB |
| ACW | 3 dB | 2.23 dB | 1.94 dB |
| ACW | 1 dB | 2.09 dB | 1.85 dB |
| PFL | 5 dB | 2.30 dB | 2.32 dB |
| PFL | 3 dB | 2.15 dB | 1.89 dB |
| PFL | 1 dB | 2.01 dB | 1.78 dB |

Table 1: Hard and soft decision OAAE values

For each test utterance, an absolute error between the true SNR and the estimated SNR is found. The performance measure is a mean value of this absolute error taken over the total number of test speech utterances. Figure 2 shows the block diagram for score and SNR determination.

## 4. EXPERIMENTAL PROTOCOL

Ten sentences from each of the 38 speakers from the New England dialect of the TIMIT database are used for the experiments. The speech in this database is clean and first downsampled from 16 kHz to 8 kHz. For both training and testing, white Gaussian noise is added to correspond to a particular SNR. The noisy speech is preemphasized by using a nonrecursive filter $1 - 0.95z^{-1}$. For the LP analysis, the autocorrelation method [1] is used to get a 12th order LP polynomial $A(z)$. The LP analysis is done over frames of 30 ms duration. The overlap between frames is 20 ms. The LP coefficients are converted into 12 dimensional LSF, REFL, LAR, CEP, ACW and PFL feature vectors. For the PFL feature, $\alpha = 1$ and $\beta = 0.9$ (see Eq. (7)). The feature vectors are computed only in voiced frames that are selected based on energy thresholding. The VQ classifier (as described earlier) is trained using the 12 dimensional feature vectors. A separate classifier is used for each feature. A codebook of size 256 for each SNR is designed using the Linde-Buzo-Gray algorithm [8].

For each speaker in the database, there are 10 sentences. The first five are used for training the VQ classifier. The remaining five sentences are individually used for testing thereby giving 190 test cases. The roles of the training and testing speech are then reversed to get an additional 190 test cases bringing the total to 380. The goal is to correctly estimate SNR values between 0 and 30 dB (inclusive). This is a significant range for practical speaker identification systems. For each SNR value tested, there are 380 utterances over which an average absolute error (AAE) is obtained.

## 5. RESULTS

An average absolute error (AAE) is computed for test speech having SNR values between 0 and 30 dB in 1 dB increments. There are a total of 31 AAE values and an average of these values result in an overall average absolute error (OAAE). Table 1 depicts these OAAE values for each of the six features. Three different VQ classifier systems are attempted. First, the codebooks are trained for SNR values in 5 dB increments starting at 0 dB. Second, the codebooks are trained for SNR values in 3 dB increments starting at 0 dB. Third, the codebooks are trained in 1 dB increments. Even though test speech of various SNRs (like 22 dB) will definitely show some error when codebooks are designed using 3 and 5 dB increments, the purpose is to observe how the increments influence the OAAE.

For the hard decision, there can be a zero error for a particular test speech utterance especially if its SNR corresponds to that of a trained codebook. This is more likely when the codebooks are trained in 5 and 3 dB increments than when the codebooks are trained in 1 dB increments. For example, when using the LSF feature for test speech at 15 dB SNR, a zero error is achieved for about 77, 53 and 21 percent of the 380 test utterances for codebooks trained in 5, 3 and 1 dB increments. respectively. However, if an error is made, it is relatively higher for codebooks trained in 5 and 3 dB increments. When the SNR of the test speech does not correspond to a codebook trained in 5 and 3 dB increments, there is no chance of a zero error. For example, when using the REFL feature for test speech at 11 dB SNR, a zero error is achieved for about 19 percent of the 380 test utterances for codebooks trained in 1 dB increments. When the error is not zero, the SNR estimates are usually 9, 10, 12 and 13 dB with 10 and 12 dB being more common. Absolute errors of 3 dB or more occur as statistical outliers. For codebooks trained in 5 dB increments, the error is usually either 1 or 4 dB that correspond to SNR estimates of 10 or 15 dB, respectively. For codebooks trained in 3 dB increments, the error is usually either 1 or 2 dB that correspond to SNR estimates

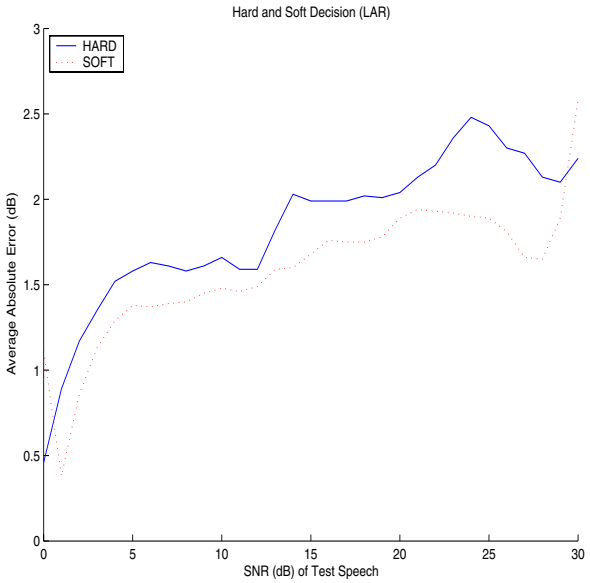


Figure 3: The AAE for the hard and soft decision methods for the LAR feature. The codebooks are trained in 1 dB increments.

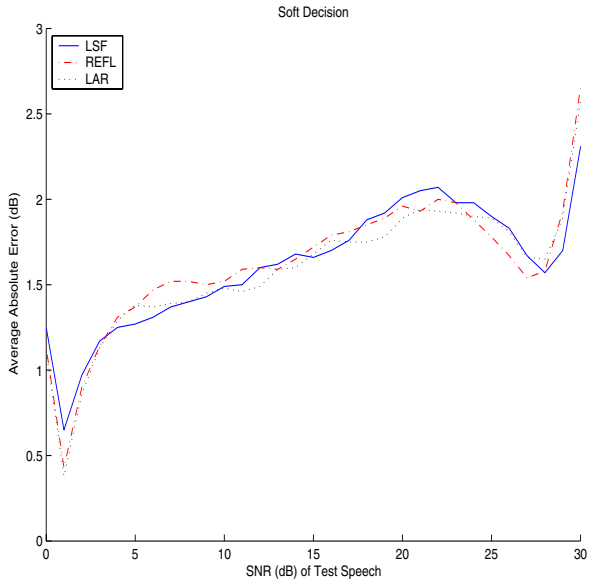


Figure 4: The AAE for the soft decision methods for the LSF, REFL and LAR features. The codebooks are trained in 1 dB increments.

of 12 or 9 dB, respectively. Again, higher errors occur as outliers. Using more codebooks trained at 1 dB increments brings down the OAAE for all the six features.

The score values and resulting probabilities are numerically close for codebooks trained in the neighborhood of the SNR of the test speech. For the soft decision, the codebooks with the best 3 scores are used. The use of 3 codebooks can compensate for the error made by a hard decision. However, the utilization of more than 3 codebooks for the soft decision will include the influence of the scores of the codebooks trained on SNR values that are appreciably different from the SNR of the test speech. Although the corresponding probabilities of these scores are relatively low, the resulting terms in Eq. (9) leads to a higher absolute error than using 3 codebooks. The soft decision approach diminishes the OAAE for codebooks trained in 1, 3 and 5 dB increments when compared to the hard decision method. This is more apparent for codebooks trained in 5 dB increments. Figure 3 shows the AAE for the hard and soft decision methods for the LAR feature with the codebooks trained in 1 dB increments. The AAE is taken for 380 test speech utterances at each SNR between 0 and 30 dB in steps of 1 dB.

The three best features are the LSF, REFL and LAR features. The CEP feature shows only a slightly higher OAAE. Figure 4 shows the AAE for the soft decision method for the LSF, REFL and LAR features with the codebooks trained in 1 dB increments.

## 6. SUMMARY AND CONCLUSIONS

The VQ based pattern recognition approach to blind SNR estimation has given very good results when the codebooks are trained in 1 dB increments. Using a soft decision approach improves the performance with an OAAE value of about 1.6 dB for the LAR, REFL and LSF features. The CEP feature gives about a 1.7 dB OAAE.

## 7. ACKNOWLEDGEMENT

This work was supported by the U.S. Air Force Research Laboratory, Rome NY, under contract FA8750-05-C-0029.

## 8. REFERENCES

1. T. F. Quatieri, *Discrete Time Speech Signal Processing Principles and Practice* Prentice Hall PTR, 2002.
2. J. P. Campbell, "Speaker recognition: A tutorial", *Proc. IEEE*, vol. 85, pp. 1437–1462, September 1997.
3. H. A. Murthy, F. Beaufays, L. P. Heck and M. Weintraub, "Robust text-independent speaker identification over telephone channels", *IEEE Trans. on Speech and Audio Proc.*, vol. 7, pp. 554–568, Sept. 1999.
4. M. C. Huggins and J. J. Grieco, "Confidence Metrics For Speaker Identification", *Int. Conf. on Spoken Language Processing*, Denver, Colorado, 2002.
5. M. C. Huggins and J. J. Grieco, "Speaker Identification Confidence Metrics For Heterogeneous Model Spaces", *Proc. of the 8th World Multiconference on Systemics, Cybernetics and Informatics*, Orlando, Florida, pp. 440–443, July 2004.
6. K. T. Assaleh and R. J. Mammone, "New LP-derived features for speaker identification", *IEEE Trans. on Speech and Audio Proc.*, vol. 2, pp. 630–638, Oct. 1994.
7. M. S. Zilovic, R. P. Ramachandran and R. J. Mammone, "Speaker identification based on the use of robust cepstral features obtained from pole-zero transfer functions", *IEEE Trans. on Speech and Audio Proc.*. vol. 6, pp. 260–267, May 1998.
8. Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. on Comm.*, vol. COM-28, pp. 84–95, Jan. 1980.