Marsland Press
Journal of American Science 2009:5(2) 1-12

Using the Sound Recognition Techniques to Reduce the Electricity Consumption in Highways

[1] Khalid T. Al-Sarayreh, [2] Rafa E. Al-Qutaish, [3] Basil M. Al-Kasasbeh

[1] School of Higher Technology (ÉTS), University of Québec, Montréal, Québec H3C 1K3, Canada

[2] Faculty of IT, Alzaytoonah University of Jordan, Airport Street, PO Box: 130, Amman 11733, Jordan

[3] Faculty of IT, Applied Science University, PO Box: 926296, Amman 11931, Jordan

Abstract:
The lighting is available for the highways to avoid accidents and to make the driving safe and easy, but turning the lights on all the nights will consume a lot of energy which it might be used in another important issues. This paper aims at using the sound recognition techniques in order to turn the lights on only when there are cars on the highway and only for some period of time. In more details, Linear Predictive Coding (LPC) method and feature extraction will be used to apply the sound recognition. Furthermore, the Vector Quantization (VQ) will be used to map the sounds into groups in order to compare the tested sounds. [Journal of American Science 2009:5(2) 1-12] (ISSN: 1545-1003)

Key word: Linear Predictive Analysis; Sound Recognition; Speaker Verification; Electricity Consumption

1. Introduction

Conserving Energy is one of the most important issues in many countries since they have limited resources of fuel to depend on, and they may be import all their need of energy from other countries. Therefore many conferences have been held urge the people to conduct the consumption of energy. This paper will introduce a system to control the lighting of lamps in highways. The system will turn the lights on only if there is a car in the highway for a pre-defined period of time, and will keep the lights off for any other sound.

Conserving energy of highways lights system could be used to reduce the power invoice by controlling the lights of lamps in the highways and will save a lot of energy. The algorithms that define the Conserving Energy of Street Lights system use the Database which consists of 250 sounds of cars and a lot of sounds from other domains.

2. An Overview of the Related Techniques

2.1 Voice Recognition

Voice recognition consists of two major tasks, that is, Feature Extraction and Pattern Recognition. Feature extraction attempts to discover characteristics of the sound signal, while pattern recognition refers to the matching of features in such a way as to determine, within probabilistic limits, whether two sets of features are from the same or different domain [Rabiner and Juang, 1993]. In general, speaker recognition can be subdivided into speaker identification, and speaker verification. Speaker verification will be used in this paper to recognize the sound of cars.

2.2 Linear Predictive Coding (LPC)

Linear predictive coding (LPC) is defined as a digital method for encoding an analogue signal in which a particular value is predicted by a linear function of the past values of the signal. It was first proposed as a method for encoding human speech by the United States Department of Defence (DoD) in federal standard, published in 1984. The LPC model is based on a mathematical approximation of the vocal tract. The most important aspect of LPC is the linear predictive filter which allows determining the current sample by a linear combination of P previous samples. Where, the linear combination weights are the linear prediction coefficient.

The LPC based feature extraction is the most widely used method by developers of speech recognition. The main reason is that speech production can be modelled completely by using linear predictive analysis, beside, LPC based feature extraction can also be used in speaker recognition system where the main purpose is to extract the vocal tract [Nelson and Gailly, 1995].

2.3 Vector Quantization (VQ)

The quantization is the process of representing a large set of values with a much smaller set [Sayood, 2005]. Whereas, the Vector Quantization (VQ) is the process of taking a large set of feature vectors, and producing a smaller set of feature vectors, that represent the centroids of the distribution, i.e. points spaced so as to minimize the average distance to every other point.

However, optimization of the system is achieved by using vector quantization in order to compress and subsequently reduce the variability among the feature vectors derived from the frames. In vector quantization, a reproduction vector in a pre-designed set of K vectors approximates each feature vector of the input signal. The feature vector space is divided into K regions, and all subsequent feature vectors are classified into one of the corresponding codebook-elements (i.e. the centroids of the K regions), according to the least distance criterion (Euclidian distance) [Kinnunen and Frunti, 2001].

2.4 Digital Signal Processing (DSP)

The Digital Signal Processing (DSP) is the study of signals in a digital representation and the processing methods of these signals [Huo and Gan, 2004]. The DSP and analogue signal processing are subfields of signal processing. Furthermore, the DSP includes subfields such as audio signal processing, control engineering, digital image processing, and speech processing. RADAR Signal processing, and communications signal processing are two other important subfields of DSP [Lyons, 1996].

2.5 Frequency Domain

The signals are converted from time or space domain to the frequency domain usually through the Fourier transform. The Fourier transform converts the signal information to a magnitude and phase component of each frequency. Often the Fourier transform is converted to the power spectrum, which is the magnitude of each frequency component squared. This is one of the features that we have depended on in our analysis [Fukunaga, 1990].

2.6 Time Domain

The most common processing approach in the time or space domain is enhancement of the input signal through a method called filtering. Filtering generally consists of some transformation of a number of surrounding samples around the current sample of the input or output signal. There are various ways to characterize filters [Smith, 2001].

Most filters can be described in Z-domain (a superset of the frequency domain) by their transfer functions. A filter may also be described as a difference equation, a collection of zeroes and poles or, if it is an FIR filter, an impulse response or step response. The output of an FIR filter to any given input may be calculated by convolving the input signal with the impulse response. Filters can also be represented by block diagrams which can then be used to derive a sample processing algorithm to implement the filter using hardware instructions [Garg, 1998].

3. The Methodology

In this paper, first we have collect many samples for cars sound from many areas. Then the feature extraction was applied on the sound. The sound was passed through a high-pass filter to eliminate the noise. The extraction of the LPC coefficient, the magnitude of the signal, and the pitch of the signal were made. These features were normalized and clustered into codebooks using vector quantization and the Linde-Buzo-Gray (LBG) algorithm for clustering which based on the k-mean algorithm. Finally a comparison with the template database that we have built before was made.

3.1 Database

The database which was used in this system was built from the recorded sounds which we record from different places, also from sounds for rain, thunder, and plane which we have brought them from internet, also from different human sounds. For the cars, rain, and plane groups, vector quantization method is used for clustering based on LBG algorithm and k-mean algorithm, and the Euclidian distance for matching. Statistical analyses were used for the human group, since the sounds of the human are very different and can't be bounded. Statistical analyses were based on the power spectrum of the sound then the mean and slandered deviation was taken to make the comparison.

3.2 Collecting Samples

We have collected about 250 sample of car's sound from different places beside the highways. These samples were taken after the mid night to assure that we have taken the pure sound of the car with the least possible noise. A microphone connected to a laptop was used; it was at a high place to assure to collect all the sound, since the proposed hardware should be beside the light of the highway, which is about 5 to 50 meters above the cars. We have used a program called Sound Forge for recording the sounds. Most of the sounds were recorded at a sample frequency of 44Khz to make sure that the sound has a high quality, and all the component of the sound will be shown when converting the sound to the frequency domain.

3.3 Feature Extraction

In order to recognize the sound of the car among other sounds we need to extract the parameters from the sound signal, these parameters help us to distinguish the sounds domain from others (car, plane, weather, and human sounds). Feature extraction consists of choosing those features which are most effective for preserving class separately [Fukunaga, 1990]. The main features that we have chosen which most effectively describe the sounds are LPC analysis, magnitude of the signal, and pitch of the signal.

3.4 Pitch Extraction

The harmonic-peak-based method has been used to extract pitch from the wave sound. Since harmonic peaks occur at integer multiples of the pitch frequency, then we compared peak frequencies at each time (t) to locate the fundamental frequency in order to find the highest three magnitude peaks for each frame. Therefore, the differences between them computed. Since the peaks should be found at multiples of the fundamental, we know that their differences should represent multiples as well. Thus, the differences should be integer multiples of one another. Using the differences, we can derive our estimate for the fundamental frequency.

The peak vector consists of the largest three peaks in each frame. This forms a track of the pitch for the signal [Ayuso-Rubio and Lopez-Soler, 1995]. First we have found the spectrogram of the signal; spectrogram computes the windowed discrete-time Fourier transform of a signal using a sliding window.

The spectrogram is the magnitude of this function which shows the areas where the energy is mostly appear, after that we have take the largest three peaks in each frame. A major advantage to this method is its very noise-resistive. Even as noise increases, the peak frequencies should still be detectable above the noise.

3.5 Feature Comparison

After the feature extraction, the similarity between the parameters derived from the collected sound and the reference parameters need to be computed. The three most commonly encountered algorithms in the literature are Dynamic Time Warping (DTW), Hidden Markov Modelling (HMM) and Vector Quantization (VQ). In this paper, we use the VQ to compare the parameter matrices.

3.6 Decision Function

There are usually three approaches to construct the decision rules [Gonzales and Woods, 2002], that is; Geometric, Topological, or Probabilistic rules.

If the probabilities are perfectly estimated, then the Bayes Decision theory is the optimal decision. Unfortunately, this is usually not the case. In that case, the Bayes Decision might not be the optimal solution, and we should thus explore other forms of decision rules. In this paper, we will discuss two types of decision rules, which are based either on linear functions or on more complex functions such as Support Vector Machines (SVM).

4. Theoretical Implementation

4.1 LPC Analysis

LPC based feature extraction is the most widely used method by developers of speech recognition. The main reason is that speech production can be modelled completely by using linear predictive analysis, beside, LPC based feature extraction can also be used in speaker recognition system where the main purpose is to extract the vocal tract parameters from a given sound, in speech synthesis, linear prediction coefficient are the coefficient of the FIR filter representing a vocal tract transfer function, therefore linear prediction coefficient are suitable to use as a feature set in speaker verification system. The general idea of LPC is to determine the current sample by a linear combination of P previous samples where the linear combination weights are the linear prediction coefficient. Since LPC is one of the most powerful speech analysis techniques for extracting good quality features and hence encoding the speech. The LPC coefficients (ai) is the coefficients of the all pass transfer function H(z) modelling the vocal tract, and the order of the LPC (P) is also the order of H(z), which has been defined to be 10 in this paper.

Linear predictive coding (LPC) offers a powerful and simple method to exactly provide this type of information. Basically, the LPC algorithm produces a vector of coefficients that represent a smooth spectral envelope of the DFT magnitude of a temporal input signal. These coefficients are found by modelling each temporal sample as a linear combination of the previous P samples.

To be noted that the order of the LPC which used in this paper is 10. The LPC filter is given by:

$$H(Z) = \frac{1}{1 + a_1 Z^{-1} + a_2 Z^{-2} + \ldots + a_{a10} Z^{-10}}$$

This is equivalent to saying that the input-output relationship of the filter is given by the linear difference equation:

$$u(n) = s(n) + \sum_{i=1}^{10} a_i s(n-i)$$

Where u(n) is the innovation of the signal, s(n) is the original signal, H(Z) is LPC filter, and ai are the coefficient of the filter.

Another important equation that is used to predicate the next output from previous samples is:

$$\hat{s}[n] = G \cdot u[n] - \sum_{k=1}^{10} a_k s[n-k] \cong -\sum_{k=1}^{10} a_k s[n-k]$$

Where ŝ[n] (the prediction for the next output value) is a function of the current input and previous outputs, G is the gain.

The optimal values of the filter coefficients are gotten by minimizing the Mean Square Error (MSE) of the estimate, that is:

$$e[n] = s[n] - \hat{s}[n] \rightarrow \min\left(E\left(e^2[n]\right)\right)$$

Where E[n] is the mean square error.

A popular method to get a Minimum Mean Square Error (MMSE) is called the autocorrelation method, where the minimum is found by applying the principle of orthogonality. To find the LPC parameters, the Toeplitz autocorrelation matrix is used:

$$\begin{bmatrix} R(0) & R(1) & R(2) & R(3) & R(4) & R(5) & R(6) & R(7) & R(8) & R(9) \\ R(1) & R(0) & R(1) & R(2) & R(3) & R(4) & R(5) & R(6) & R(7) & R(8) \\ R(2) & R(1) & R(0) & R(1) & R(2) & R(3) & R(4) & R(5) & R(6) & R(7) \\ R(3) & R(2) & R(1) & R(0) & R(1) & R(2) & R(3) & R(4) & R(5) & R(6) \\ R(4) & R(3) & R(2) & R(1) & R(0) & R(1) & R(2) & R(3) & R(4) & R(5) \\ R(5) & R(4) & R(3) & R(2) & R(1) & R(0) & R(1) & R(2) & R(3) & R(4) \\ R(6) & R(5) & R(4) & R(3) & R(2) & R(1) & R(0) & R(1) & R(2) & R(3) \\ R(7) & R(6) & R(5) & R(4) & R(3) & R(2) & R(1) & R(0) & R(1) & R(2) \\ R(8) & R(7) & R(6) & R(5) & R(4) & R(3) & R(2) & R(1) & R(0) & R(1) \\ R(9) & R(8) & R(7) & R(6) & R(5) & R(4) & R(3) & R(2) & R(1) & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ a_7 \\ a_8 \\ a_9 \\ a_{10} \end{bmatrix} = \begin{bmatrix} -R(1) \\ -R(2) \\ -R(3) \\ -R(4) \\ -R(5) \\ -R(6) \\ -R(7) \\ -R(8) \\ -R(9) \\ -R(10) \end{bmatrix}$$

Where:

$$R(k) = \sum_{n=0}^{159-k} s(n)s(n+k)$$

and R(k) is the autocorrelation of the signal.

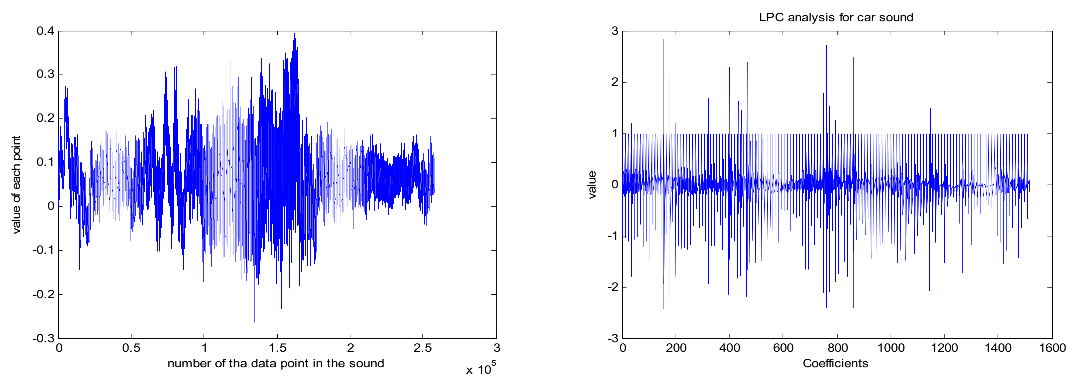
The above matrix equation could be solved using the Gaussian elimination method. Any matrix inversion method or The Levinson-Durbin recursion (described below). To compute this vector, the recursive Levinson-Durbin Algorithm (LDR) was used.

4.2 Pre-emphasis

In general, the digitized speech waveform has a high dynamic range and suffers from additive noise. In order to reduce this range pre-emphasis is applied. By pre-emphasis [Robiner and Juang, 1993], we imply the application of a high pass filter, which is usually a first-order FIR of the form:

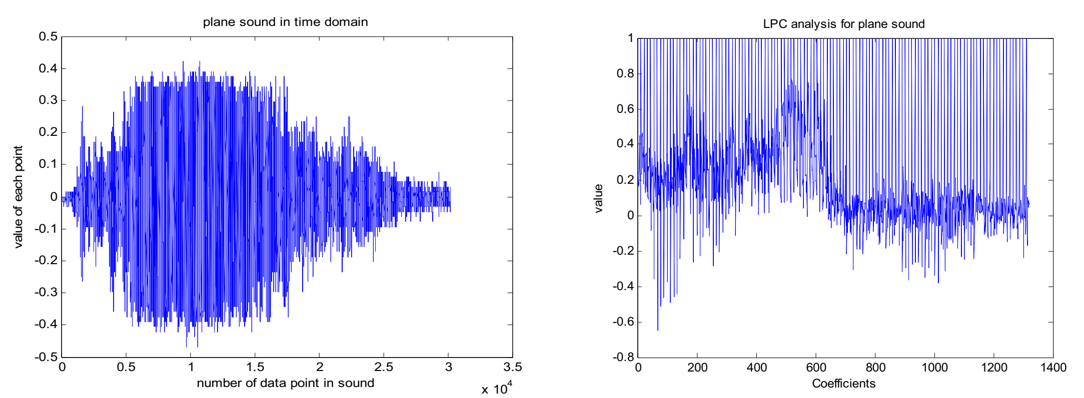
$$H\ (z) = 1 - az^{-1},\ 9 \leq a \leq 1.0$$

The pre-emphasis is implemented as a fixed-coefficient filter or as an adaptive one, where the coefficient ai is adjusted with time according to the autocorrelation values of the speech. The pre-emphasize has the effect of spectral flattening which renders the signal less susceptible to finite precision effects (such as overflow and underflow) in any subsequent processing of the signal. The selected value for a in our work was 0.9375. Fig.1 and Fig.2 below represent the process of LPC analysis.



(a) (b)

Fig.1 Presents (a) the Car Sound Wave, (b) the Coefficient of LPC Analysis for Same Sound



(a) (b)

Fig.2 Presents (a) the Plane Sound Wave, (b) the Coefficient of LPC Analysis for Same Sound

4.3 Factors Affecting Linear Predictive Analysis

The main factors which the LPC equations depend on are the number of predictor parameters [Goldberg and Riek, 2000]. Depends upon the order, in this paper P is chosen to be 10, and the frame length N the choice of P depends primarily on the sampling rate and is essentially independent of the LPC method and we can summarize the dependency as the following:

1. The prediction error decrease steadily as P increase.
2. For P in the order of 13-14, the error has essentially flattened off showing only small decrease as P increases further [Klevans and Rodman, 1997].

There are a number of clustering algorithms available for use; however it has been shown that the one chosen does not matter as long as it is computationally efficient and works properly. A clustering algorithm is typically used for vector quantization, Therefore, One popular approach to designing vector quantization is a clustering procedure known as the k-means algorithm, which was developed for pattern recognition applications and LBG algorithm proposed by Linde, Buzo, and Gray.

4.4 The k-means algorithm

The K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other.

The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early group is done. At this point we need to re-calculate k new centroids as bar centers of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more [MacQueen, 1997].

Finally, this algorithm aims at minimizing an objective function, in this case a squared error function. The objective function:

$$j = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\| x_i^{(j)} - C_j \right\|^{(2)}$$

Where $\left\| x_i^{(j)} - C_j \right\|^{(2)}$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre is an indicator of the distance of the n data points from their respective cluster canters [Zha et al, 2001]. The algorithm is composed of the following steps:

1. Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.
2. Assign each object to the group that has the closest centroid.
3. When all objects have been assigned, recalculate the positions of the K centroids.

Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated [Moore, 2007].

4.5 Distance Measure

After quantizing a sound into its codebook, we need a way to measure the similarity/dissimilarity between and two sound domains. It is common in the field to use a simple Euclidean distance measure, and so this is what we have used.

4.5.1 Euclidian distance

Euclidean metric is the distance between two points that one would measure with a ruler, The Euclidean distance between two points P=[p_1 p_2 p_3 p_n]^T and Q=[q_1 q_2 q_3 q_n]^T and, in Euclidean n-space, is defined as:

$$\sqrt{\sum_{i=1}^{n}(p_i - q_i)^2}$$

4.5.2 Distance Weighting Coefficients

We follow the algorithm proposed above to weight distances so as to increase the likelihood of choosing the true sound domain. This algorithm, basically, reflects the greater importance of unique codewords as opposed to similar ones. This is a very important portion of our system. The training system architecture we created consists of two main parts. The first part consists of processing each sound input voice sample to condense and summaries the characteristics of the sound features. The second part involves pulling each sound's data together into a single, easily manipulated, three dimensional matrix.

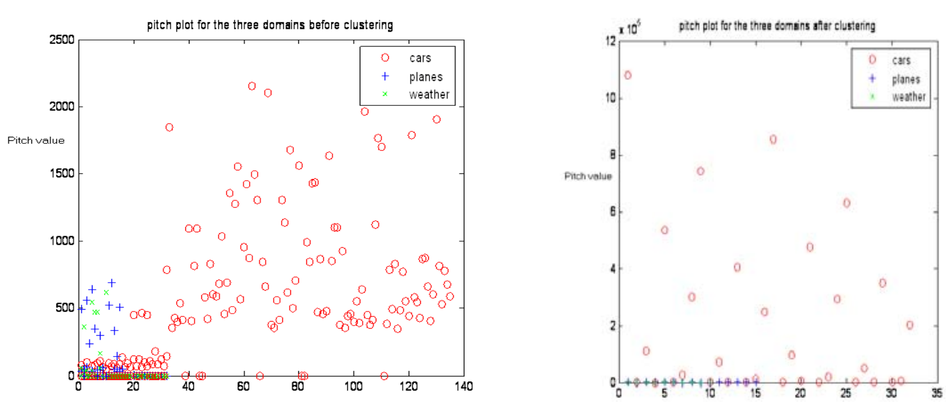
5. Practical Implementation

First we have high-pass filtered the signal because the more important information for speech processing lies in the higher frequencies according to Kinnunen and Franti [2004]. Then we split the signal into frames, each about 30ms long. By breaking the signal into frames, we approximate these distinct sounds in our analysis. For each frame we calculate the LPC coefficient. We also calculated the Magnitude and the pitch of the sound. These coefficients characterize each sound domain. The next step is to map these data. This is called vector quantization (VQ) and is accomplished by a clustering algorithm.

However, the clustering algorithm takes a number of random vectors and condenses the vectors that are nearest to it, iterating until the least mean error between the vectors is reached. We clustered the data into vectors, and each of these vectors is called a codeword. This set of vectors, or codewords is created for each sound. The codewords for a given sound are then stored together in a codebook for that sound domain. Each speaker's codebook (sound group) is then stored together in a master codebook which is compared to the test sample during the testing phase to determine the sound domain.

Suppose there is a region of space where codeword vectors from several different sounds were laid. If a test vector also falls in this region, the codewords do not help determine the identity of the sound domain because the errors between the test vector and the various codewords will be roughly equal.

Kinnunen and Franti [2004] present an algorithm for discriminating between code vectors during the testing phase to help solve this problem. Their idea is to give a higher precedence to code vectors which give a more definite idea of a domain's identity by weighting them. We used the algorithm they presented and computed weighting coefficients for my codebook data during the training phase. In parts (a) and (b) of Fig.3, we present the features of the sounds for the different domains before and after clustering [MacQueen, 1997].



(a) (b)

Fig.3 (a) the pitch feature for all sounds in all domains before clustering, (b) the pitch feature for all sounds in all domains after clustering

The figures above depict code vectors before and after clustering, and each shape represents a different group (car, plane, weather). You can see that the distribution of the vectors before and after clustering is different. Weighting the vectors does make a difference.

6. Testing, Results, and Discussion

6.1 Testing

For all of the following tables, denote "recognizing the sound as a car" to be T, "not recognizing the sound as a car" to be F, and we assume that the acceptable interval is between 0.607 and 1.0622. However, we have conducted two types of testing as the following:

1. Using the first method in testing (Feature Extraction based on Statistical analysis), the percentage of recognizing the cars sounds as a car is 92.5% and the percentage of recognizing the plane and rain sounds as a car is 6%; see Tables 1 and 2 below for more details.

Table 1: The performance of the first method in recognition for cars samples

Number of Sample Cars	Recognized	Value of the mean calculation for the sound
1	T	0.70
5	F	0.45
9	F	0.59
10	T	0.77
13	T	0.70
20	T	0.69
35	T	0.80
49	T	0.88
57	T	0.90
62	T	0.96
77	T	0.92
99	T	0.85

Table 2: The performance of the first method in recognition for planes samples

Number of sample Planes and rains	Recognized	Value of the mean calculation for the sound
1	T	0.650
5	F	0.005
9	F	0.020
10	F	0.100
13	F	0.217
15	F	0.168
19	F	0.020
22	F	0.180
25	F	0.054
29	T	0.600
33	F	0.322

2. Second Method in Testing (Vector Quantization); see Tables 3 and 4.

Table 3: The performance of the second method in recognition for cars samples

Number of Sample Cars	Recognized
1	T
5	T
9	T
10	T
13	T
20	T
35	T
49	T
57	T
62	T
77	T
99	T

Table 4: The performance of the second method in recognition for planes and rains samples

Number of sample Planes and rains	Recognized
1	F
5	F
9	F
10	F
13	F
15	F
19	F
22	F
25	F
29	F
33	F

6.2 Results

The voice recognition system that we have built identifies the three main sound domains which are planes, cars, and weather. The recognition for these domains was 100% according to the vector quantization method but the vector quantization method divided all the sound into three mainly regions so any new sound will be approximated into one of these region so to get more accuracy and to avoid any similar sound to the car sound we have developed a new code called feature extraction based on statistical analysis to discard any sound that is similar to the car sound but really not car sound this method with the vector quantization method get accuracy 100% and we assure to make energy conserving with probability reached 100% since the lights will turn only for cars .the feature extraction based on statistical analysis method gain accuracy of 92.5% when work alone.

6.3 Discussion

The idea of the feature selection based on statistical hypothesis testing is to set two classes let's say x1 and x2 and each class has its own shape, distribution, mean, and standard deviation then for a specific sample we will try to investigate whether the values which it takes differ significantly or belongs to these sets. For the feature extraction, we have take the power spectral density for each sound and the mean of all the samples and their standard deviation, then we have make dot multiplication between the mean (power spectral density), of all the samples and the new test sound and upon this number(the result of multiplication) we have decide if this test sound lie in the pre-determined classes or not and the range of comparison was (μ - 2*σ , μ + 2*σ) so the probability that the data will lie within this range is 95% according to statistical analysis and rules. However, the results of this method were as the following three choices:

1. When executing the code as it is (with

acceptable interval [av-2*std, av+2*std]), the results were as the following:

- Average = 0.8346,
- STD = 0.1138
- The probability of data that will lie here = 95%.
- The correct recognition of car sound = 92.5%.
- Recognizing sound of planes as car= 4%
- Recognizing sound of rain as car= 2%
- Recognizing sound of animals as car= 0%

2. When executing the code and taking the acceptable interval to be [av-sd , av+sd], the results were as the following:
   - Average = 0.8346
   - Std = 0.1138
   - The probability of data that will lie here = 67%.
   - The correct recognition of car sound= 87%
   - Recognizing sound of planes as car= 0%
   - Recognizing sound of rain as car= 0%
   - Recognizing sound of animals as car =0%
3. When executing the code and taking the FFT(sound, 11000) and summing the area under the curve from 0-5500 and interval to be [av-2*sd, av+2*sd] the results were as the following:
   - Average = 0.8324
   - Std = 0.1155
   - Acceptable interval [av-2*std, av+2*std].
   - The correct recognition of car sound =91%
   - Recognizing sound of planes as car= 0%
   - Recognizing sound of rain as car= 8%
   - Recognizing sound of animals as car= 0%.

Based on the results of this paper, we can note that the choice number one is the best

7. Conclusion

The lighting is available for the highways to avoid accidents and to make the driving more safety and more easy, but turning the lights on all the nights will consume a lot of energy which it might be used in another important issues. This paper presented a methodology of using the sound recognition techniques in order to turn the lights on only when there are cars on the highway and only for some period of time. In more details, Linear Predictive Coding (LPC) method and feature extraction have been used to apply the sound recognition. Furthermore, the Vector Quantization (VQ) has been used to map the sounds into groups in order to compare the tested sounds.

However, this paper made the following contributions:

1. Designing a new system for conserving energy based on the voice recognition of the car sound.
2. This system is the first application of this type that concern the street lights.
3. This paper also demonstrates that the weighted Euclidian distance with the LBG algorithm was very helpful and achieved high accuracy.

This paper shows that the feature of the sound that we have extracted is very valuable and really can distinguish between different sounds, so it is differentiate sounds from other or which we can call that it makes speaker identification with high accuracy.

References

Ayuso-Rubio, A. J., Lopez-Soler, J. M., 1995, Speech Recognition and Coding: New Advances and Trends, Springer, Berlin, Germany.

Fukunaga, K., 1990, Introduction to Statistical Pattern Recognition, 2nd edition, Academic Press, London, UK.

Goldberg, R., Riek, L., 2000, A Practical Handbook of Speech Coders, CRC Press, Boca Raton, FL, USA.

Gonzales, R., Woods, R., 2002, Digital image processing, 2nd edition, Prentice-Hall.

Garg, H. K., 1998, Digital Signal Processing Algorithms: Number Theory, Convolution, Fast Fourier Transforms, and Applications, CRC Press, Boca Raton, FL, USA.

Kinnunen, T., Frunti, P., 2001, Speaker Discriminative Weighting Method for VQ-Based Speaker Identification, In proceedings of the. 3rd International Conference on Audio-and Video-Based Biometric Person Authentication (AVBPA'01), Halmstad, Sweden, pp. 150-156.

Klevans, R., Rodman, R., 1997, Voice Recognition, Artech House Inc., Northfield, MN, USA.

Kuo, S. M., Gan, W.-S., 2004, Digital Signal Processors: Architectures, Implementations, and Applications, Prentice Hall, Englewood Cliffs, New Jersey, USA.

Lyons, R. G., 1996, Understanding Digital Signal Processing, Pearson Education, London, UK.

MacQueen, J., 1967, Some Methods for Classification and Analysis of Multivariate Observations, In proceedings of the 5th Berkeley Symposium on Mathematical statistics and probability, Berkeley, California, University of California Press, pp. 281-297.

Moore, A., 2007, K-means and Hierarchical Clustering, Tutorial Slides, School of Computer Science, Carnegie Mellon University, Online: http://web.cecs.pdx.edu/~york/cs510w04/kmeans09.pdf, Accessed on Dec. 3, 2007.

Nelson, M., Gailly, J.-L., 1995, The Data Compression Book, M&T Books.

Rabiner, L. R. and Juang, H., 1993, Fundamentals of Speech Recognition, Prentice Hall, Englewood Cliffs, New Jersey, USA.

Sayood, K., 2005, Introduction to Data Compression, Morgan Kaufmann, San Francisco, CA, USA.

Smith, D., 2001, Digital Signal Processing Technology: Essentials of the Communications Revolution, American Radio Relay League, Newington, CT, USA.

Zha, H., Ding, C., Gu, M., He, X., Simon, H. D., 2001, Spectral Relaxation for K-means Clustering, Advances in Neural Information Processing Systems, Vol. 14, pp. 1057-1064.