Suggested Algorithm For Speech Signal Coding

Dr. khalil Al saif **Saja Jasem mohammed**
Department of Computer Science
College of Computer Science and Mathematics
University of Mosul

Received 11 / 04 / 2010 **Accepted** 03 / 10 / 2010

الخلاصة

ناقش هذا البحث طريقة مقترحة لنوع جديد من كبس وترميز بيانات إشارة الكلام مبنية على أساس تحويل KL (Karhunen-Loève transform) الذي يستخدم عادة مع الصور الرقمية.

يتم إعادة ترتيب إشارة الكلام (والتي تكون عادة على شكل تسلسل أحادى) بصيغة مصفوفة ذات بعدين لتكون ملائمة لتحويل (KL transform)، ثم تجرى عليها عملية الترميز والكبس باختيار عدد مختلف من eigen values و eigenvectors. تم قياس كفاءة أداء الخوارزمية بحساب عامل SNR وعامل PSNR لعدد من التجارب على إشارة كلام وبقيم مختلفة من eigenvalues و eigenvectors.

حققت الخوارزمية نسبة عالية من الكبس مع نسبة جيدة من عامل SNR و PSNR واسترجعت الإشارة وكانت قريبة جدا من الإشارة الأصلية (وقد تم قياس الجودة باستخدام عامل الارتباط الذي يعطي قيم قريبة من الـ١) بالإضافة الى زمن قليل للكبس.

Abstract

This paper discusses a suggested approach for new kind of Speech Signal Compression Algorithm based on KL transform (Karhunen-Loève transform) which normally used with digital image.

Speech signal (which is single sequence) rearranged in form of two dimension square matrix to be suitable for KL transformation. Then to be compressed using different no. of eigen values and eigenvectors. Measuring the performance of applied KL transformation (by evaluating the PSNR and SNR factors) with different numbers of eigen values and vectors studied.

High rate of compression with high SNR, PSNR got with so closed speech signal to the original one measured using correlation factor which gives values near to 1. In addition to short time needed for compression operation.

1- Introduction:

Speech coding is the application of data compression of digital audio signals containing speech **[1].** A variety of techniques have been developed to efficiently represent speech signals in digital form for either transmission or storage [2]. The goal in speech coding is to minimize the distortion at a given bit rate, or minimize the bit rate to reach a given distortion [3].

The recent improvements both in speech coding algorithms and DSP hardware have resulted in an increased use of low rate codes in communication systems.[4] The two most important applications of speech coding are mobile telephony and Voice over IP **[1].**

2- Speech coding:

Speech coding uses speech-specific parameter estimation using audio signal processing techniques to model the speech signal, combined with generic data compression algorithms to represent the resulting modeled parameters in a compact bit stream[5][1].

The central problem in speech coding is to represent the speech signal using as little bits as possible so that quality and intelligibility get damage as little as possible[6]. Speech coding is important in digital mobile phone systems and that's why speech coding methods have advanced considerably in 10 recent years. Thinking commercially, speech coding is the most important application of speech processing field[5][7]

The requirements for a good speech codec (codec = coder-decoder) can be: quality of speech suffers as little as possible the speech is compressed in a small amount of bits coding-decoding yields only small delay codec is not sensitive to errors in transmission of bits coding/decoding is computationally fast the codec should perform well with noisy speech (and if possible with other musical signals etc.) several consequent encodings should not impair the quality too much[7].

There are no perfect codecs satisfying all the requirements because part of the requirements are contradictory. However by making different compromises, a large number of coding standards for different applications have been developed. For instance in the speech codec of a mobile phone all the requirements above are essential, whereas in recording of speech in databases delay, computational load and error resiliency are inessential, only the quality and good compression ratio counts. [7]

There are plenty of coding methods but they can be divided in roughly two main classes[7]:

- waveform coding .
- source coding (also known as vocoders).

In waveform coding an effort is made to retain the waveform of the original signal and the coding is based on quantization and removal of redundancies in the waveform.[8].

Waveform encoders typically use Time Domain or Frequency Domain coding and attempt to accurately reproduce the original signal. These general encoders do not assume any previous knowledge about the signal. The decoder output waveform is very similar to the signal input to the coder[9].

In source coding the parameters of speech (the type of excitation, model of vocal tract, formant frequencies...) are coded enabling reconstruction in the decoder.[7].

The advantages of the former algorithms are: simplicity, high quality of reestablished Signal and anti—noise. But its compression ratio is low, it need high transmissibility. This kind of algorithm can not satisfy the requirement when transmissibility is not high[8][10].

Speech Parameter Coding Algorithm can achieve high compression ratio and low bit rate. But there are many shortages, such as: bad quality of reestablished Signal, loss of the nature of the speaker, bad naturality etc. And the parameter coding algorithms is sensitive to the environment noise. So this kind of algorithm can not satisfy the requirement of high quality [5] [8][10], this is shown in fig (1).

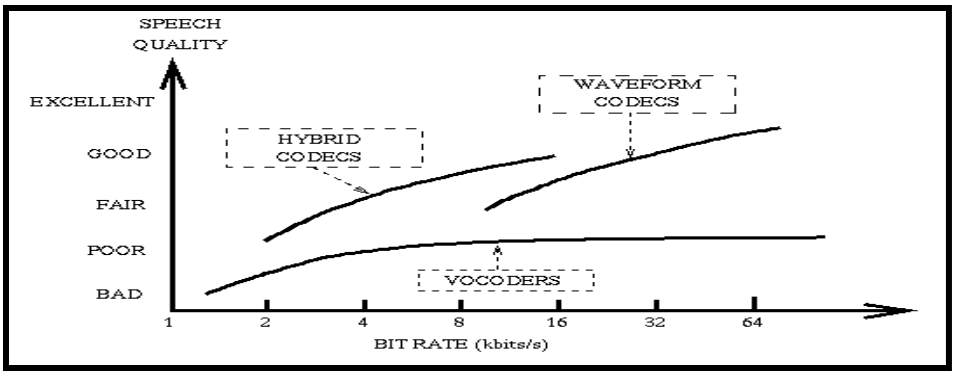


fig (1): Mean opinion scores for various types of speech coders

3- Compression:

Compression is used just about everywhere. All the images you get on the web are compressed, most modems use compression, HDTV will be compressed using MPEG-2, and several file systems automatically compress files when stored, and the rest of us do it by hand.

We must distinguish between "lossless algorithms", which can reconstruct the original message exactly from the compressed message, and "lossy algorithms", which can only reconstruct an approximation of the original message. Lossless algorithms are typically used for text, and lossy for images and sound[11].

Lossy Compression techniques are compression in which some of the information from the original message sequence is lost. This means the original sequences cannot be regenerated from the compressed sequence. Just because information is lost doesn't mean the quality of the output is reduced. The certain losses in images or sound might be completely imperceptible to a human viewer[11], Transform Coding is one of techniques used in this type of compression**.**

Transform Coding is a way of data encoding that is used in many compression schemes. The data is transformed to an other domain before encoding. The transform should be chosen such that it removes the correlation (or dependence) from the source representation.[12]

The idea of transform coding is to transform the input into a different form which can then either be compressed better, or for which we can more easily drop certain terms without as much qualitative loss in the output[11].

In this paper KL transformation was achieved for speech coding/compression. Karhaunen-Loeve Transform, or Principal Component Analysis (PCA) has been a popular technique for many image processing and pattern recognition applications. This transform which is also known as Hotelling Transform is based on the concepts of statistical properties of image pixels or pattern features [13][14]. The KL transformation is also knows as the principal component transformation, the eigenvector transformation or the Hotelling transformation [15].

In 1983, the work of Hemon and Mace was extended by a group of researchers at the University of British Columbia in Canada which culminated in the work of Jones and Levy (1987) [15].

In 1988 Freire and Ulrych applied the KL transformation in a somewhat different manner to the processing of vertical seismic pro_ling data[15].

In Signal processing, signal dependent transforms are those transforms that generate their transform vectors based on either apriori knowledge of the signal or by analysis of the signal [16].

The advantage of such schemes is that they can adjust themselves to the characteristics of the signal. The main disadvantage is that they tend to be more computationally intensive. This is caused by two factors. Firstly, if the transform vectors are not predefined then a fast algorithm for their use cannot be generated. Secondly, there will be a computational overhead for the generation of the vectors in the first place, which can be very large. There is also the problem of storage of the transform vectors. As these vectors are unique for the data set under analysis they must all be stored. This overhead can become relatively large if the number of waveforms in the set is small, or if the length of each waveform is large[16][14].

4- THE KARHUNEN LOEVE TRANSFORM (KLT):

KLT can be used on any signal that comprises a set of correlated data sequences[signal], KLT is the optimal transform in that:

• It completely decorrelates the original signal. (the transform coefficients are statistically independent for a Gaussian signal).

• It optimizes the repacking of the signal **energy**, such that *most of the signal energy is contained in the fewest transform coefficients.*

• The total entropy of the signal is minimized.

• For any amount of compression the MSE (Mean Square Error) in the reconstruction is minimized.

Given these abilities, the KLT should be in widespread use. However, there are several disadvantages to using the KLT, the greatest being the computational overhead required to generate the transform vectors. The transform vectors for the KLT are the eigenvectors of the auto covariance matrix formed from the data set[**16**].

To generate the KLT vectors the procedure is as follows[**16**]:

1. Take N sequences each of L data points, $X_{[N][L]}$.
2. Construct an average signal M

$$\mu_j = \frac{1}{N}\sum_{i=0}^{N-1} X_{[i][j]} \qquad j = 0....L-1 \qquad (1)$$

3. Subtract the average signal from the original ensemble

$$Z_{[i]} = X_{[i]} - \mu \qquad i = 0......N-1 \qquad (2)$$

4. Construct a covariance matrix.

$$\begin{bmatrix} C_{1,1} & C_{1,2} & \cdots & \cdots & C_{1,L} \\ C_{2,1} & C_{2,2} & & & \\ \vdots & & \ddots & & \\ \vdots & & & \ddots & \\ C_{L,1} & C_{L,2} & & & C_{L,L} \end{bmatrix}$$

where each value $C_{i,j}$ is given by:

$$C_{i,j} = \sum_{p=0}^{N-1} \frac{Z_{[p][i]} \times Z_{[p][j]}}{N-1} \qquad i = 1...L,\ j = 1....L \qquad (3)$$

5. Evaluating eigen values(λ) & eigenvectors(e) of matrix C, that is possible because C matrix is real and symmetric.
6. Sort the resulted matrix descending order(i.e. $\lambda_i \geq \lambda_{i+1} \geq \lambda_{i+2}$)
7. Using the sorted eigen value to calculate eigen vectors .
8. Select k of the highest eigen vectors which is belong to highest eigen value to generate (A) matrix which will be of dimension [k,n].
9. Now last the coding/compressed speech signal is obtain from equation(4) below:

$$Y = A(X-\mu) \qquad (4)$$

To get the approximately speech signal (□)we just apply equation(5) below:

$$□ = A^T(Y+ \mu) \qquad (5)$$

That is because $A^{-1} = A^T$.

5- The suggested algorithm:

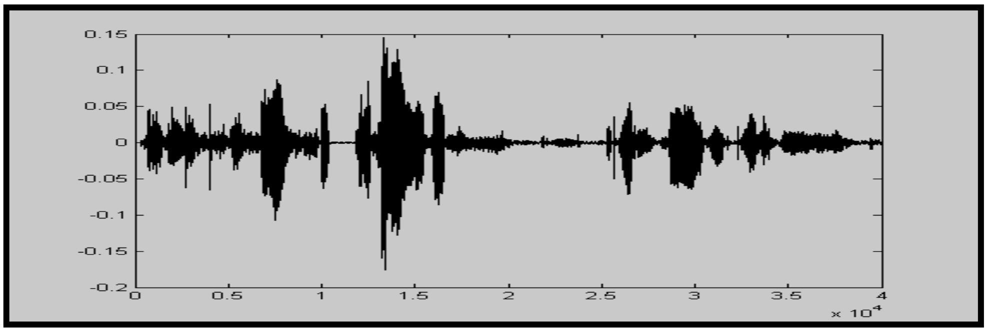
In this paper, KL transform is used as compression/coding method for speech signal by the following steps:

- At the begin the speech signal which was recorded before needed to be read, the used sampling rate is 8000, with length of signal N samples.
- The signal dimension speech signal of length N will be reconstructed in two dimensional array of dimension n*n (n will be integer of square root of N).
- Apply K-L transform on this matrix, Y will be the output of the KL transformation process which will be obtained from equation.(4).

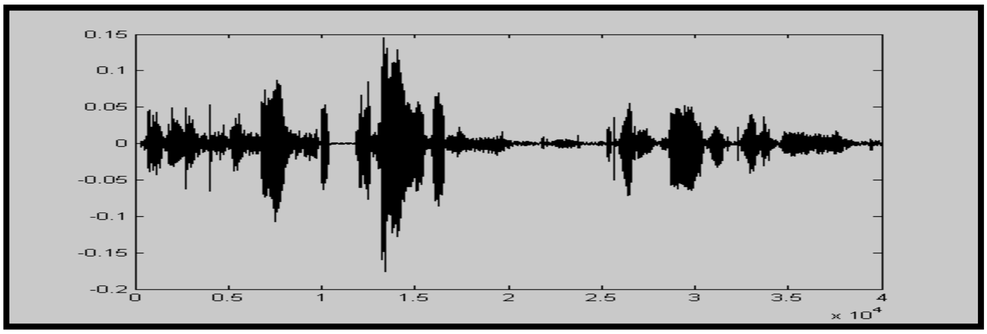
Now reconstructed speech signal □ can be regenerated from equation (5), □ now is the approximately speech.

6- Applied Example:

The suggested algorithm was applied to some examples of speech signal. The figures below show this application on man speech signal with 40000 sample length, therefore new speech matrix will be of (200*200) dimension. As well as length of eigenvector is: 200. Fig(2) shows original speech signal. where fig(3) explains the reconstructed one without compression (i.e. 200 eigenvectors in matrix A).

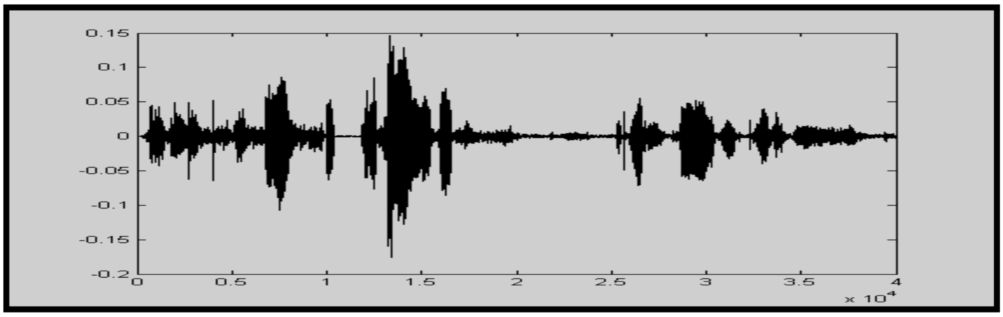


Fig(2):original speech signal for man sound

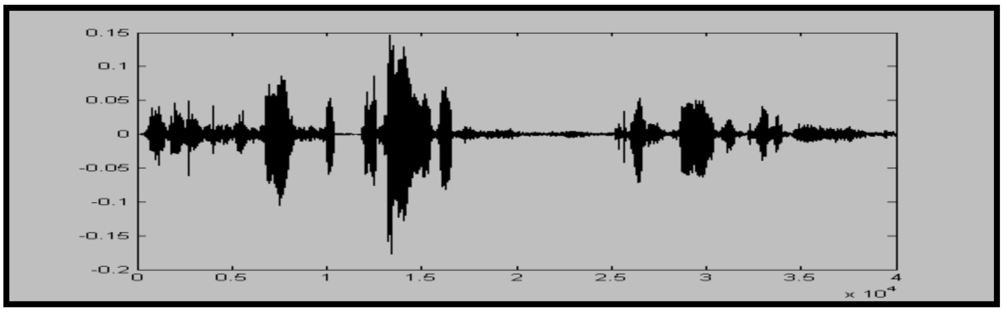


Fig(3):reconstructed signal with 200 eigenvectors

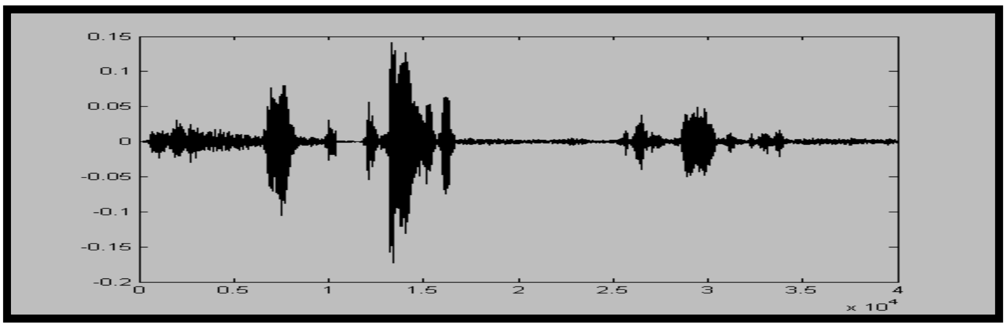
When compression is applied by this suggested approach in addison to coding the no of eigen values and vectors will be reduced until reaching suitable ratio. This is explained in the figures below, fig (4) shows same previous speech signal with compression to half of total size (i.e. the no of eigenvectors: 100 vectors). Where fig(5) explain the reconstructed signal with 50 eigenvectors. Also fig.(6) clear the reconstructed signal with 20 eigenvectors, and last fig(7) shows the reconstructed signal with 1 eigenvector only.



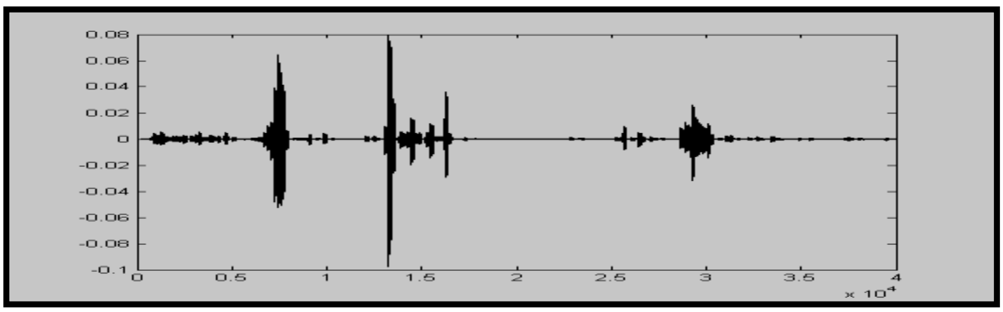
Fig(4): reconstructed signal with 100 eigenvectors



Fig(5): reconstructed signal with 50 eigenvectors



Fig(6): reconstructed signal with 20 eigenvectors



Fig(7): reconstructed signal with 1 eigenvector

7- Some standard measures[1]:

a- Signal-to-Noise Ratio (SNR or S/N):

It is a measure used in science and engineering to quantify how much a signal has been corrupted by noise. It is defined as the ratio of signal power to the noise power corrupting the signal. A ratio higher than 1:1 indicates more signal than noise[1].

Because many signals have a very wide dynamic range, SNRs are often expressed using the logarithmic decibel scale. In decibels, the SNR is defined as equation(6):

$$SNR_{dB} = 10\log_{10}\left(\frac{P_{signal}}{P_{noise}}\right) = P_{signal,dB} - P_{noise,dB} \qquad (6)$$

where P is average power.

which may equivalently be written as equation(7):

$$SNR_{dB} = 10\log_{10}\left(\frac{A_{signal}}{A_{noise}}\right)^2 = 20\log_{10}\left(\frac{A_{signal}}{A_{noise}}\right) \qquad (7)$$

where A is root mean square (RMS) amplitude.

When a measurement is digitized, the number of bits used to represent the measurement determines the maximum possible signal-to-noise ratio. This is because the minimum possible noise level is the error caused by the quantization of the signal, sometimes called Quantization noise[1]

b- Mean Square Error (MSE) and Peak signal-to-noise ratio (PSNR):

The PSNR block computes the peak signal-to-noise ratio, in decibels, between two signals. This ratio is often used as a quality measurement between the original and a reconstructed one. The higher the PSNR, the better the quality of the signal.

The Mean Square Error and the Peak Signal to Noise Ratio (PSNR) are the two error metrics used to compare signal compression quality. The Mean Square Error (MSE) represents the cumulative squared error between the reconstructed and the original signal, whereas PSNR represents a measure of the peak error. The lower the value of MSE, the lower the error.

To compute the PSNR, the block first calculates the mean-squared error using equation(8):

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n} er_i^2 \qquad (8)$$

Where n : Signal length

er : error between original signal and reconstructed one.

Then the block computes the PSNR using equation(9):

$$PSNR = 10\log_{10}[\frac{R^2}{MSE}] \qquad (9)$$

In the previous equation, R is the maximum fluctuation in the input signal data type. For example, if the input signal has a double-precision floating-point data type, then R is 1. If it has an 8-bit unsigned integer data type, R is 255, etc.[1]

c- Compression Ratio:

Data compression ratio, also known as compression power, is a computer-science term used to quantify the reduction in data-representation size produced by a data compression algorithm. The data compression ratio is analogous to the physical compression ratio used to measure physical compression of substances, and is defined in the same way, as the ratio between the uncompressed size and the compressed size[1][15]this explained in equation(10):

$$\text{Compressed ratio} = \frac{\text{Compressed size}}{\text{uncompressed size}} \qquad (10)$$

d- Pearson's Correlation Coefficient

The most familiar measure of dependence between two quantities is the Pearson product-moment Correlation Coefficient, or "Pearson's Correlation." It is obtained by dividing the covariance of the two variables by the product of their standard deviations. Karl Pearson developed the coefficient from a similar but slightly different idea by Francis Galton[17].

The population correlation coefficient $\rho X,Y$ between two random variables X and Y with expected values μX and μY and standard deviations σX and σY is defined as:

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\mu_X)(Y-\mu_Y)]}{\sigma_X \sigma_Y}, \qquad (11)$$

Where E : the expected value operator
cov : means covariance.

The Pearson correlation is 1 in the case of an increasing linear relationship, −1 in the case of a decreasing linear relationship, and some value between -1 and 1 in all other cases, indicating the degree of linear dependence between the variables. The closer the coefficient is to either −1 or 1, the stronger the correlation between the variables[17][1].

If the variables are independent, Pearson's correlation coefficient is 0, but the converse is not true because the correlation coefficient detects only linear dependencies between two variables. However, in the special case when X and Y are jointly normal, uncorrelatedness is equivalent to independence.

If we have a series of n measurements of X and Y written as xi and yi where i = 1, 2, ..., n, then the sample correlation coefficient, can be used to estimate the population Pearson correlation between X and Y. The sample correlation coefficient is written

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}, \qquad (12)$$

where x and y are the sample means of X and Y, sx and sy are the sample standard deviations of X and Y.[17][1]

8- Result Discussion:

The proposed algorithm applied on multi-numbers of speech files with different size and it also in different number of eigen vectors. The results written in table(1) which shows the measured compression ratio in addition to the speech quality represented by SNR, PSNR, MSE & Correlation Factor.

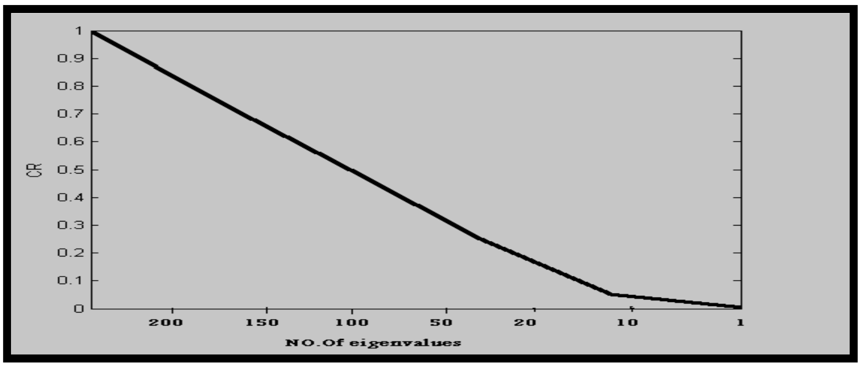
Table(1): Result of testing suggested method on the speech signal in variable no. of eigenvectors.

no. of eigenvectors	Compression Ratio	SNR	PSNR	MSE	Correlation Coefficient
200	1	64.184	67.1002	1.9498e-007	0.9992
150	0.75	63.5909	66.8426	2.0689e-007	0.9991
100	0.5	51.1962	61.4597	7.1455e-007	0.9970
50	0.25	28.5174	51.6104	6.9018e-006	0.9707
10	0.05	7.5141	42.4888	5.6380e-005	0.7268
1	0.005	0.9529	39.6392	1.0866e-004	0.3015

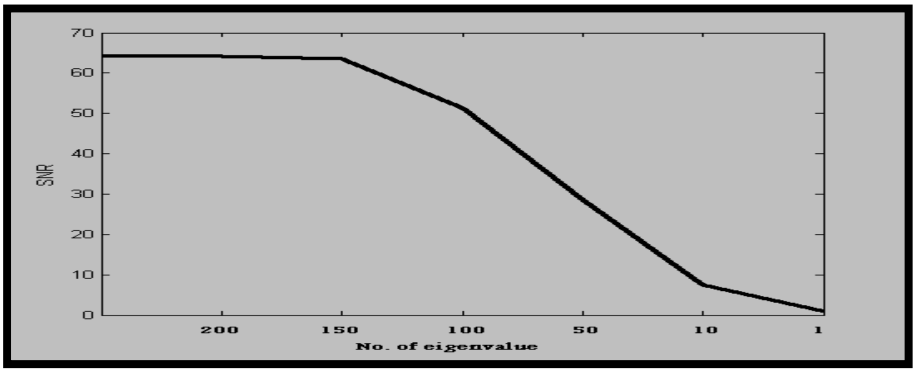
The curve in fig(8) gives clear result that the compression ratio is rapidly proportional increased with no. of eigenvectors adopted in compression operation. Also the SNR clearly seen that low effect at high no. of eigenvectors and rapidly decrease with low no. of eigenvectors, that is shown in fig(9).

Same things can be seen in fig(10) which represents that the proposed algorithm is stable when the no. of eigenvectors increased, in addition to that, fig(11) show that the error vanished when the number of eigenvectors increased.

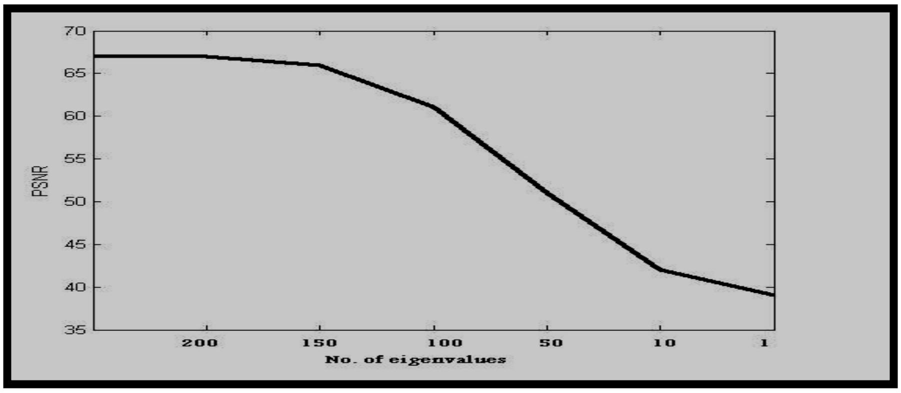
Fig(12) shows logically seen that when the no. of eigenvectors rise up the correlation factor goes near to 1 which mean the retreaved signal is so close to the original.



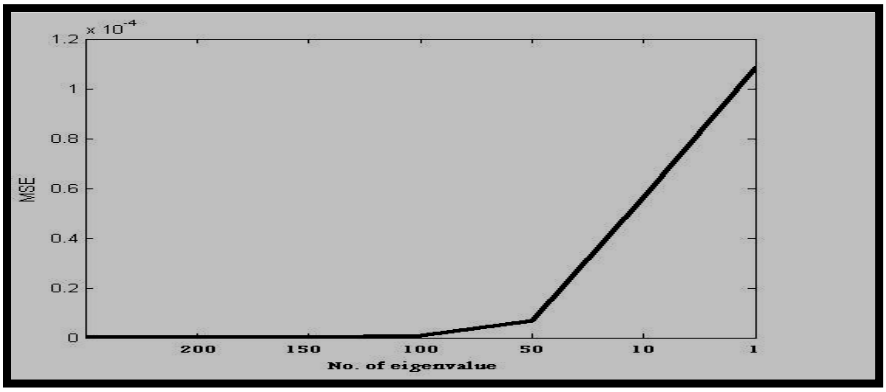
Fig(8): The relationship between CR and eigenvectors



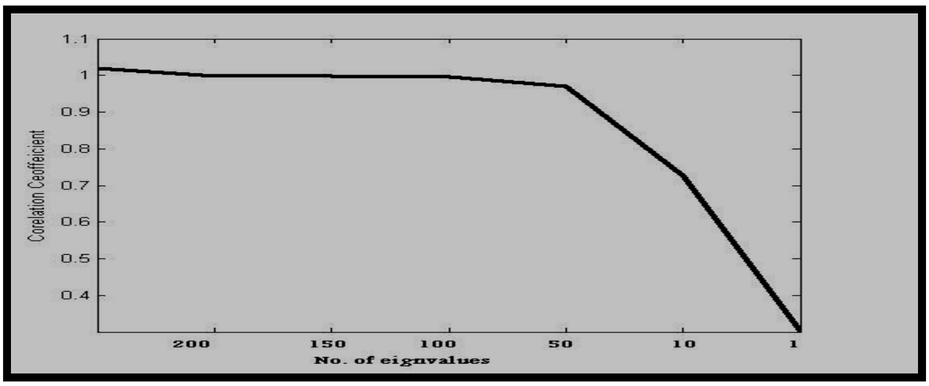
Fig(9): The relationship between SNR and eigenvectors



Fig(10): The relationship between PSNR and eigenvectors

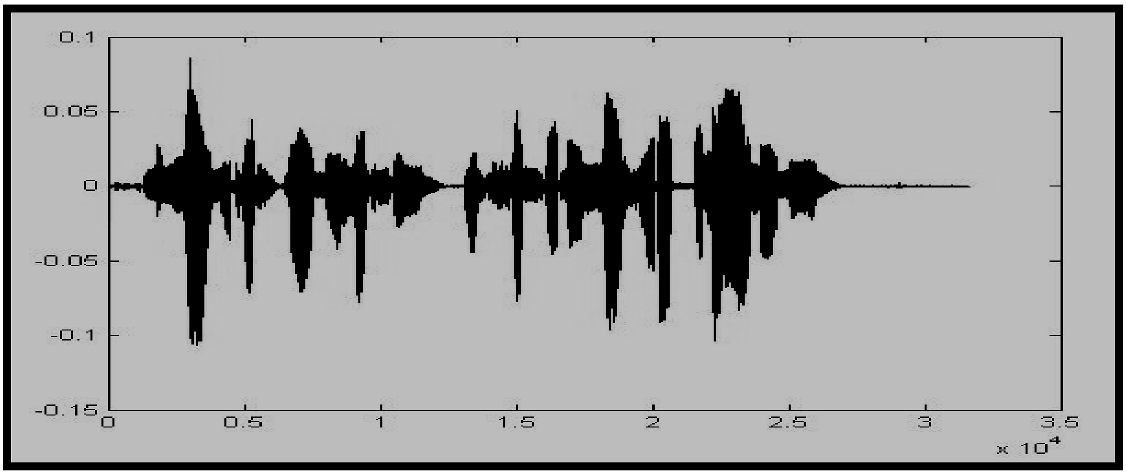


Fig(11): The relationship between MSE and eigenvectors



Fig(12) The relationship between the correlation factor and no. of eigenvectors

The proposed algorithm applied on another sound file that contains female speech signal in (4sec.) period which clear in fig(13) and the results are clarify in table(2) below.



Fig(13) origin female speech signal

Table(2): Result of testing suggested method on another speech signal in variable no. of eigenvectors.

no. of eigenvectors	Compression Ratio	PSNR	MSE	SNR	Correlation Coefficient
178	1	31.8688	6.50E-04	1.8201	0.9998
120	0.67	31.8687	6.50E-04	1.8198	0.9997
100	0.56	31.8678	6.50E-04	1.8177	0.9992
60	0.33	31.8536	6.53E-04	1.7851	0.991
1	0.0056	31.1508	7.67E-04	0.1668	0.3152

Comparing the proposed algorithm with some familiar methods can be seen in table (3) which shows the clear comparison between two familiar methods and proposed one on same file (3sec.peired) .Note that the proposed algorithm use third of eigenvectors no. in compress operation.

Table(3): Result of comparison between two familiar methods and proposed one

Methods	Compression Ratio	SNR	PSNR	MSE	Correlation Coefficient
Proposed one	0.322	0.9759	31.6948	6.7689e-004	0.9942
LPC	0.331	0.2769	30.9538	8.0282e-004	0.0288
DPCM	0.368	0.692	69.888	5.9429e-007	0.9962

The parameter of table (3) shows that the proposed technique give better compression ratio than the others with flexibility between the quality and the compression ratio, and also the retrieved signal was closer to the original one, than LPC method.

Also the table shows that the LPC technique has a lower compression ratio and the correlated coefficient seems to be small because of the difficulty of the predication code of the retrieve way. And practically the licenser face that the retrieved voice with high noise.

9- Conclusions:

Due to the practical application on the suggested algorithm, the following can be seen as conclusion for the suggested approach:

- High rate of compression can be got with big size of speech file.
- Retrieved speech still with high quality low noise and so closed to the original one .
- Difficult for listener to recognize the effect of the difference between the original and retrieved speech.

10- feature work:

- The suggested algorithm can be adopted to compress speech over multi- channels.
- The algorithm can be modified in real time compression .
- With multiplexer the algorithm can be simulated using simulink matlab to be checked correctly.

Reference

1) Wikipedia the free encyclopedia, http://en.wikipedia.org, (2009).

2) Seth, Siddharth, "Sub-band Coding of Speech Signals using Multirate Signal Processing", Third Year Undergraduate Student, Deptt. of E & ECE, Indian Institute of Technology Kharagpur, (2005).

3) Wang, Yao; "Source Coding Basics and Speech Coding", Polytechnic University, Brooklyn, NY11201, ©Yao Wang, (2006).

4) Euler, S.; Zinke, J; "The influence of speech coding algorithms on automatic speech recognition", Proc. ICASSP-94, IEEE, (1994).

5) Islam, Tamanna; "Interpolation of Linear Prediction Coefficients for Speech Coding", Department of Electrical Engineering, McGill University, Montreal, Canada, (2000).

6) Gibson, Jerry D.; "Speech Coding Methods, Standards, and Applications", Department of Electrical & Computer Engineering University of California, Santa Barbara, Santa Barbara, CA 93106-6065, (2005).

7) E-Book, "Speech encoding", www.toodoc.com, (2009).

8) Yang, Xingjun, Chi, Huisheng; "Digital Processing of Speech Signal", chinese, electrical industrial press, (1995).

9) Douglas, Bryan; "Voice Encoding Methods for Digital Wireless Communications Systems", EE6302 Section 324, Fall, Southern Methodist University, (1997).

10) Yantao, Song; Jianlai, Zhou; Tiecheng, Yu; "A speech compression coding algorthim based on half-waveform", Speech Recognition Lab, Chinese Academy of Sciences, Institute of Acoustics, (2005).

11) Blelloch, Guy E.; "Introduction to Data Compression", Computer Science Department Carnegie Mellon University, (2001).

12) Schaaf, A. van der; "Data compression for mobile Communications", Ubicom- Technical Report, Ubiquitous Communications, Delft University of Technology, P. O. Box 5031, 2600 GA Delft, The Netherlands, (1998).

13) Acharya, Tinku;. Ray ,Ajoy K.;, "Image Processing Principles and Applications", Copyright by John Wiley & Sons, Inc, (2005).

14) Gautschi, Walter; "computing the kontorovich-lebedev integral transforms and their inverses", Department of Computer Sciences, Purdue University, West Lafayett, Vol., No., pp.1–14, c Swets & Zeitlinger, (2000).

15) Sacchi, M. D; "Statical and transform methods for geophysical signal processing", © Sacchi, (2001).

16) Bigan, Cristin; "Signal Dependent KL Transform for ECG Signals Representation", Politehnica University of Bucharest, Acta Universitatis Apulensis, (2008).

17) Rodgers, J. L. and Nicewander, W. A.; "Thirteen ways to look at the correlation coefficient". The American Statistician 42:59–66. doi:10.2307/2685263, (1988).