Silence Encoding Technique for Compressing Digital Speech Signal

By
Falah Mahdi Abdullah

ABSTRACT

Conventional methods of coding the time waveform of a speech signal are used to reduce the data rate. However, it is to be expected that these simple coding techniques will be superseded by more complex coding methods that may give much more substantial reduction in the data rate for accepted level of degradation in speech quality.

Within the framework of this paper, a new perceptive silence encoding technique for compressing the digital speech signal is introduced. First, an overview of some basic concepts involved with natural of digital speech and digital speech coding paradigm are presented. Then, a quantitative analysis for the digital speech waveform signal is conducted to emphasis the powerful utilization of the proposed coders. An efficient preprocessing process is established to perform the deletion of the nonessential acoustic material from the speech waveform . The major aim of designing the silence encoding system is to satisfy the following:

1. Smoothing the output speech signal by determine the noise within the waveform and substitute it with a value that do not have any relation with the noise value.
2. Reduce the cost that encountered in the speech compression process and this is can be established by compressing the voiced speech only and left the unvoiced speech without any processing.

تقنية الترميز الصامتة لضغط اشارة الكلام الرقمية

المستخلص:-

تستخدم الأساليب التقليدية للترميز الموجي الوقت المستغرق للكلام كمؤشر على خفض معدل البيانات. ومع ذلك ، فأن هذه التقنيات البسيطة للترميز تكون أكثر تعقيدا لذلك حلت محلها أساليب الترميز التي قد تعطي انخفاض كبير في معدل البيانات أكثر من ذلك بكثير.

في هذا البحث ، عرض جديد لتقنية ضغط اشارات الكلام الرقمية . وترد أولا ، لمحة عامة عن بعض المفاهيم الأساسية التي ينطوي عليها مع نموذج الترميز. حيث يتم إجراء التحليل الكمي للإشارة إلى موجة رقمية مع التركيز على الاستغلال القوي للتقنية المقترحة. حيث يتم التجهيز للقيام بعملية حذف المواد الصوتية غير الاساسية من الكلام الموجي . والهدف الرئيسي من تصميم نظام الترميز الصامت هو لتلبية ما يلي :

1.تحسين الاشارة بتحديد الضوضاء داخل الموجة واستبدالها بقيمة ليس لديها أي علاقة مع قيمة الضوضاء.

2.تقليل التكلفة التي نواجهها في عملية ضغط الكلام ، وهذه يمكن أن تنشأ عن طريق ضغط اشارة الكلام فقط وترك الاشارة الصامتة دون أية معالجة.

1 Introduction

The simplest speech-specific compression technique is the *silence encoding*. When people speak, there are many pauses. Some are short pauses between words and phrases; others are long pauses between sentences or when changing speakers. You can often compress speech data by as much as 50 percent by identifying these silences and replacing them with compact duration codes [1].

A physical implementation of digital-analog communication might consist of the following steps. The information source may first be filtered to ensure that the signal is band limited and to reduce high-frequency noise. Next, the signal is sampled and quantized to provide a pulse code modulated signal. Certain encoding may then be performed to reduce the statistical redundancy. The signal is then encoded for reduction the channel errors and transformed by the modulator into a form suitable for transmission over the channel. The signal received form the channel is demodulated into base band signals, decoded for channel errors and the statistical redundancy is restored. Finally, an equalization or enhancement may be performed before the signal is received by the user [2].

A modern coder (compressor), acts in similar way, it takes digital PCM speech signal and apply further compression processes resampling , mapping, requantization, and appropriate encoding algorithms to emphasize the reduction of the bit rate efficiently to produce a more compressed representation, and finally a decoding procedure (expander) is performed at the receiver to reconstruct the original speech signal.[3].

2. WAVE File Format

Because the WAVE is the native sound format used by Microsoft Windows, it is naturally the most popular sound formats around. The overall structure is based on the general file format called the Resource Interchange File Format (RIFF).

WAVE files are RIFF files in which the outermost chunk is a RIFF container with a WAVE container type. Most WAVE files contain both an fmt chunk and a data chunk, as shown in figure (1).[4]

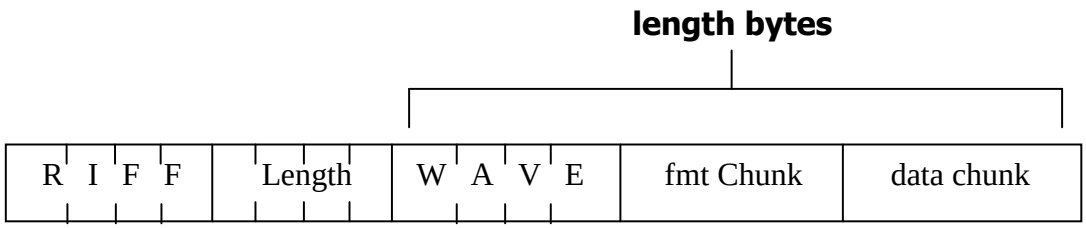


FIGURE 1: ***RIFF Chunk Format.***

3. Proposed System

In this paper, we are mainly concerned with source compressor and source expander. That it, we show source diagram of a source compressor and expander in figure (2). As shown in figure (2)(a), there are many components in source compressor: preprocessing process (noise canceling), segmenting machine, run-length coding, transform coding, sub-band coding, quantization, and codeword assignment, whereas figure (2)(b) shows the source expander which has the inverse of the source compressor components. An illustration of these components or processes will be introduced in the later sections.

The silence encoding technique was the system proposed in the current paper work. In simple words, it splits the digital speech signal into two parts (voiced and unvoiced parts), the voiced part will only compressed by the compressor body and then send a compressed part and referenced to the unvoiced part only. The expander body decompresses the compressed part and restores the unvoiced part in its appropriate places.

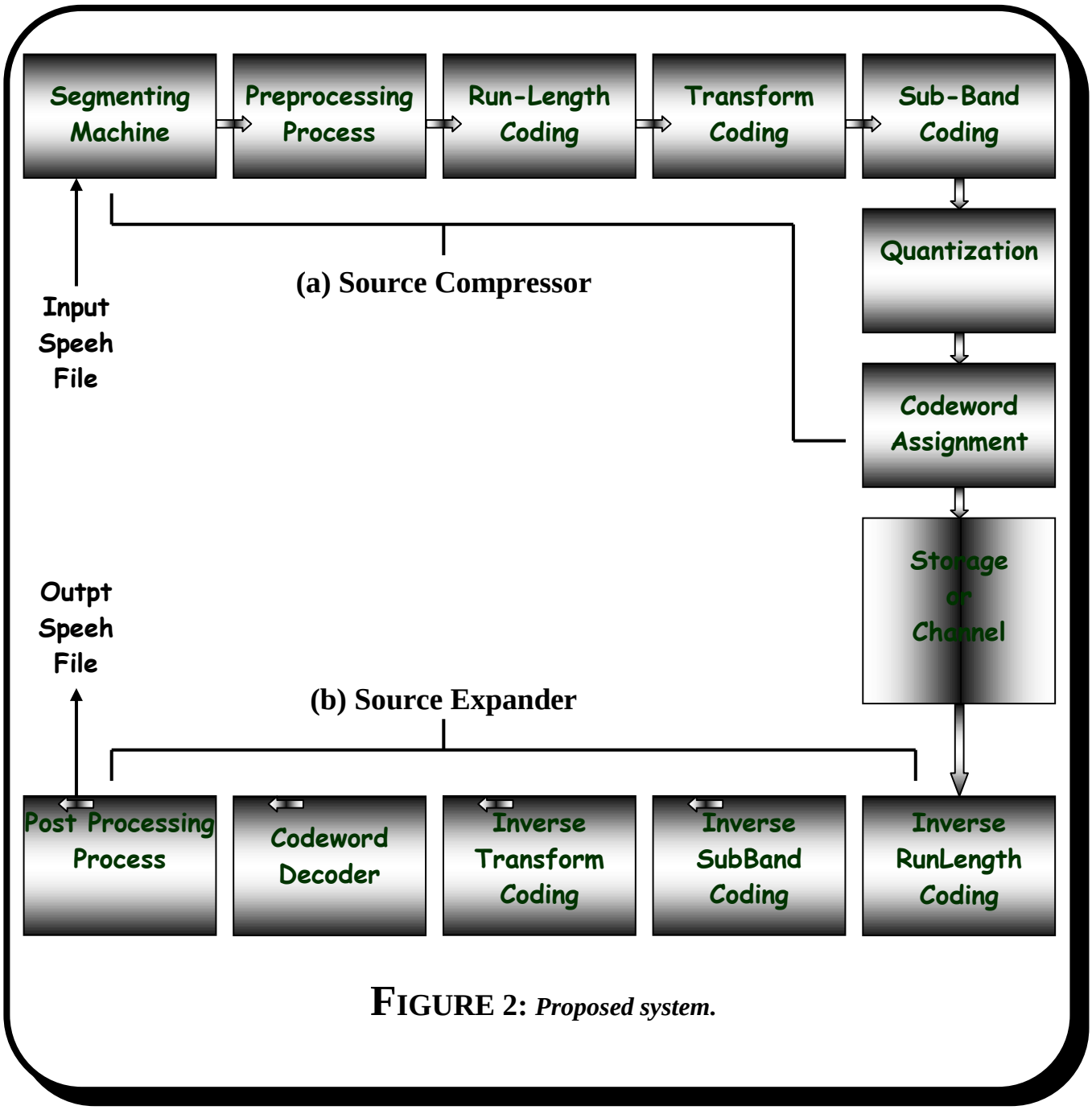


FIGURE 2: ***Proposed system.***

3.1 Segmenting Machine

The proposed coding system is based on segmenting the voiced portion of the sound (WAVE file) into fixed sized frames, and each frame will be coded separately by using the DCT coding scheme.

The first process of the system is to segment the voiced part of the input speech file into blocks (all have the same size), and then these blocks sent to the next process of the system.

3.2 Preprocessing Process

Because when we talk, we have to make a pause interval between successive words. Speech recordings may convey portions of unvoiced materials which vary in size from a talker to another (depending on the talking habits of that talker). Canceling these materials can reduce the storage requirements of the assumed output file format substantially and may reduce the processing efforts.

The unvoiced block or frame has either low power or amplitude and high zero-crossing. Thus depending on these two attributes, and consider them as two criteria, we can determine the segmented frame voiced or unvoiced and this is the second process in the suggested preprocessing process, and this can be done according to the following condition:

If (FramePower < PowerThreshold) or
(FrameZeroCrossing < ZeroCrossingThreshold)
Then Frame is voiced
Else Frame is unvoiced.

3.3 Run-Length Coding

Run-length coding (RLC) is a compression method, in our speech system, it works by counting the number of adjacent frames of voiced or unvoiced. This count, called the run-length, is then coded and stored. The efficiency of the run-length coding depends on the number of adjacent voiced or unvoiced frames. The method is also sensitive to error, since a single bit error could change the length of the run and thus offset of the entire speech.

The format of the run-length file can be illustrated in figure (3). The run-length file begins with one bit 0 or 1, depending on the first adjacent voiced or unvoiced frames of the speech file, if the first adjacent frames are voiced, then the file starts with 1 else it starts with 0, thus the first bit determines the speech file begins with voiced or unvoiced frame. Let we assume that the run-length file stars with 1 then the first number represents the number of adjacent voiced frames, the second represents the number of adjacent unvoiced frames, while the third represents the number of adjacent voiced frames, and so on.

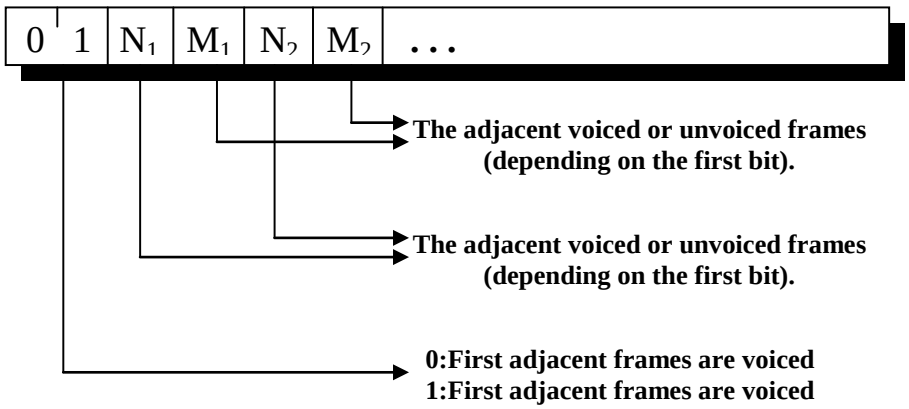


FIGURE 3: ***Run-Length File Format.***

3.4 Transform Coding

A transform is a process which takes information in one domain and express it in another. The representation of audio signal is in the time domain: their voltages (or sample values) change as a function of time. Such signals are often transformed to the frequency domain for the purpose of compression [3].

The Fast Discrete Cosine Transform (FDCT) was implemented in our system. The DCT is a time consuming transform because the number of multiplication operations involved in the transformation, and this may affect the performance of the system. In this correspondence we propose an additional algorithm, which not only reduce the number of multiplication operations but also has a simple structure. This algorithm is the FDCT. The FDCT is based on pre-computed the kernel of the DCT and can be defined as,

$$C(u) = \alpha(u)\sum_{x=0}^{N-1} f(x)\,\text{ker}\,nal[(2x+1)*u\ MOD(N*4)] \quad \ldots(1)$$

for u=0, 1, 2, …, N-1. Similarly, the inverse FDCT is defined as,

$$f(u) = \sum_{u=0}^{N-1} \alpha(u)C(u)\,\text{ker}\,nal[(2x+1)*u\ MOD(N*4)] \quad \ldots(2)$$

for x=0, 1, 2, …, N-1. In both equations (1) and (2), α is as defined in equation (3) and the kernel is an array of N*4 dimension and can be pre-computed as,

$$\ker nal(x) = Cos\left[\frac{x\pi}{2N}\right] \quad \ldots(3)$$

for x=0, 1, 2, …, N*4-1.

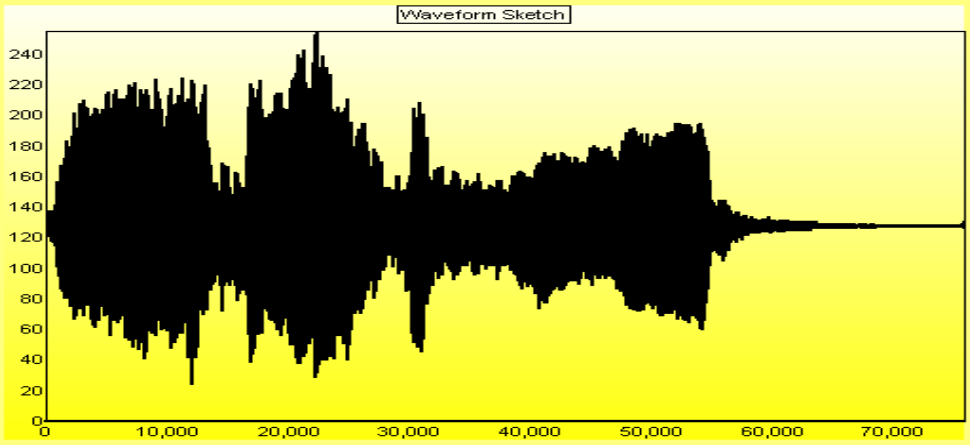
3.5 Sub-Band Coding

Sub-band coding mimics the frequency analysis mechanism of the ear (this fact realize that the speech signals do not have uniform spectral energy), it splits the speech spectrum into a large number of frequency bands.

This transform is more commonly implemented in DSP systems because it produces a smaller amount of coefficients by reiteratively filtering the original signal using high and low pass filters to remove redundant information. This process is carried out by using the high and low pass filters to split the signal into two distinct types; approximations and details. To minimize the amount of data produced, the details are sacrificed, while approximations are maintained. This process is known as *two-channel sub-band coding* [6].

The DCT converts the digital speech from the spatial domain to the frequency domain, low-frequency accumulated in the first half of the output of the DCT and high-frequency in the second half. According to the sub-band coding and the frequency analysis mechanism of the ear in which the ear is less sensitive to the high-frequency of the speech, we can discard the high-frequency part of the output of the DCT without significant distortion on the reconstructed speech, as illustrated in figure (4), in such case a reduction in the storage will be realized.

(a)



(b)

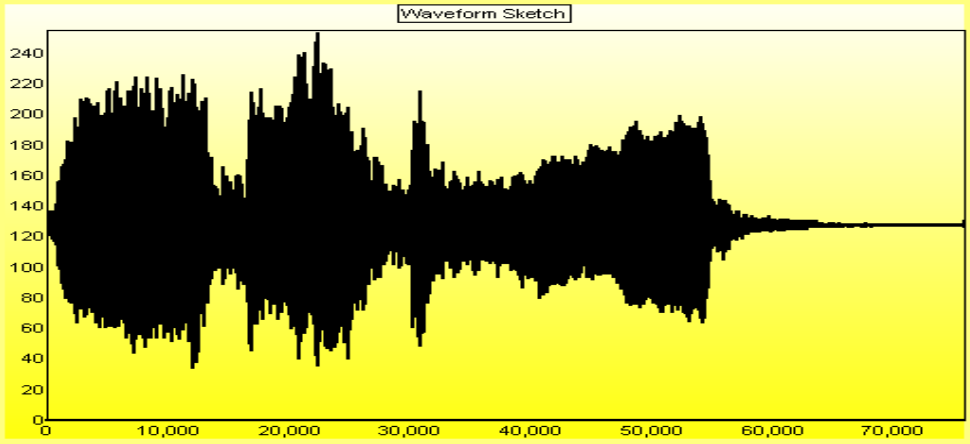


FIGURE 4: ***(a) Original Speech (b) Reconstructed Speech After Discarding the High-Frequency of the Output of the DCT.***

4. Bit Allocation

The input speech is first divided into a sequence of voiced and unvoiced frames (each frame consists of 16 bytes). The collected unvoiced frames are transmitted as zero-amplitude frames, while the collected voiced frames are divided into blocks (each block consists of 64 bytes), then a linear transform is applied to each block. The transformed blocks go through *truncation*, *quantization*, and *codeword assignment*. The last three functions: truncation, quantization, and codeword assignment are combined and called bit allocation.

As we know, the applied transform de-correlates sub-blocks of the collected voiced frames. Moreover, it redistributes speech energy in the transform domain in such a way that most of the energy is compacted into a small fraction of coefficients. Therefore, it is possible to discard the majority of transform coefficients without introducing significant distortion [6].

In the current work, several speech samples have been tested, and it is found that introducing the first half of the frequency coefficient while ignoring the second half coefficient will always does not introduce a subjective error in the speech sample signal.

Before starting the truncation process, a *weighted mapping process* was applied. The suggested mapping process maps the output coefficients of the DCT such that the first half coefficients will be enlarged, while the second part will be reduced. The weighting process was adopted due to the fact that the human hearing system is less the variations occurred in high frequency components in comparison with those variations associated with the low frequency components. Thus, the weighting mechanism will exploit this fact beside to offering the opportunity to coarsely quantization the high frequency coefficients, which will in turn leads to improve the compression performance. Due to discarding of the second half part of the output coefficients of the DCT, a reduction in the storage will be satisfied. The foreword weighted mapping process can be defined as,

$$R(x)=\frac{C(x)}{QF(x)} \qquad \textbf{...(4)}$$

for x=0, 1, 2, …, N-1. Where QF is the quality factor and can be defined as,

$$QF(x)=1+\frac{Q-1}{N-1}*x \qquad \textbf{…(5)}$$

for x=0, 1, 2, …, N-1. Clearly, the inverse weighted mapping process can be defined as,

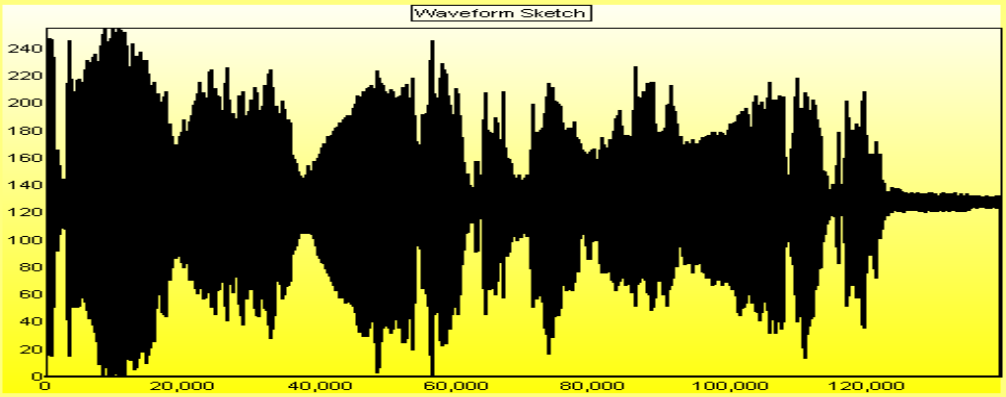
$$C(x)=R(x)*QF(x) \qquad \textbf{…(6)}$$

for x=0, 1, 2, …, N-1.

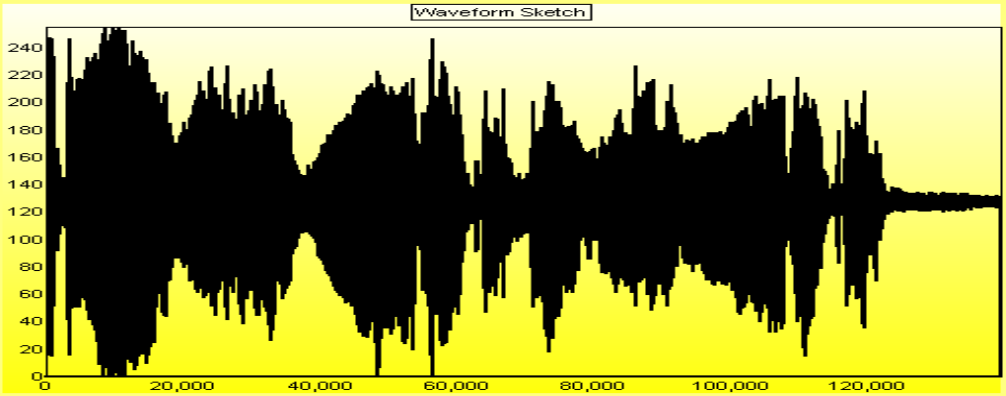
The constant Q appears in equation (5) is not chosen arbitrarily. The decision of assigning value to the constant Q stands on an experimental analysis, thus we sought that the best value of Q which dose not significantly distort the reconstructed speech and lead to good compression performance is 5. This can be illustrated in figure (5), where (a) is the original speech, (b) is the reconstructed speech after applying the weighted mapping process on the output coefficients of the DCT with Q=20, and (c)

is the reconstructed speech also after applying the weighted map After employing the mapping process, we apply the truncation (i. e., quantization) process. Table (1) shows the output coefficients of the DCT before and after the mapping and truncation processes.

(a)



(b)



(c)

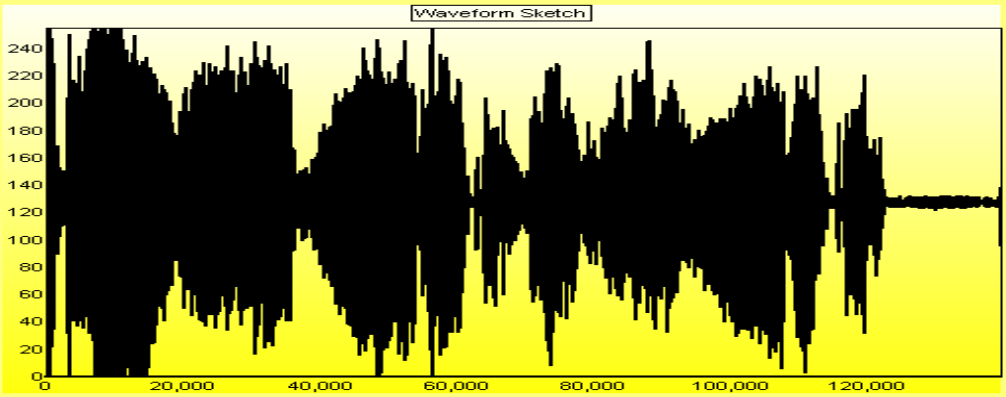


FIGURE 5: ***(a) Original Speech, (b) Reconstructed Speech with Q=20, and (c) Reconstructed Speech with Q=5.***

TABLE 1: DCT's Coefficients Before and After the Resizing Process and After the Truncation Process.

Coefficient Number	*Before Resizing*	*After Resizing*	After Truncation
0	1.00862500000000E+0003	1.00862500000000E+0003	1009
1	4.06231322426902E+0000	3.81978706162609E+0000	4
2	1.01838437265724E+0000	9.03636837709943E-0001	1
3	-1.23269009266551E+0000	-1.03545967783903E+0000	-1
4	3.98638943440233E-0002	3.17901942237148E-0002	0
5	8.31669398134181E-0001	6.31267133523535E-0001	1
6	-1.08013988157290E+0000	-7.82170259070034E-0001	-1
7	1.39122390459988E-0001	9.63155010876842E-0002	0
8	1.05311189643544E+0000	6.98379468162449E-0001	1
9	-9.80528788165429E-0001	-6.23972865196182E-0001	-1
10	-9.80528788165429E-0001	4.19398461728073E-0002	0
11	4.89255539578153E-0001	2.88066345732931E-0001	0
12	2.92647034947549E-0001	1.66096965781042E-0001	0
13	8.04298153524542E-0001	4.40615510191705E-0001	0
14	6.13219427038530E-0001	3.24645579020398E-0001	0
15	-5.28011192923259E-0001	-2.70444757350938E-0001	0
16	1.18287076122033E+0000	5.86778409109297E-0001	1
17	7.49659975752365E-0001	3.60523499789306E-0001	0
18	-1.41843761285236E+0000	-6.61937552664434E-0001	-1
19	-1.38935761718244E-0002	-6.29708848075495E-0003	0
20	-3.79479459384584E-0001	-1.67183258330271E-0001	0
21	4.05947720850236E-0001	1.73977594650101E-0001	0
22	3.01367984647641E-0001	1.25736311475506E-0001	0
23	-4.91984286933985E-0001	-1.99967806947361E-0001	0
24	-3.52677067614877E-0001	-1.39739970187027E-0001	0
25	-8.62388573707449E-0002	-3.33315829101652E-0002	0
26	2.03609367751142E-0001	7.68107195707900E-0002	0
27	-3.39013183872339E-0001	-1.24899594058230E-0001	0
28	-8.94733003027795E-0001	-3.22103881090006E-0001	0
29	-1.39568487918041E+0000	-4.91218700493663E-0001	0
30	6.25667138183417E-0001	2.15393604948389E-0001	0
31	7.78453473116315E-0001	2.62259726237047E-0001	0
32	-1.24999999999091E-0001	-4.12303664918466E-0002	0
33	4.75109978925957E-0001	1.53497070114540E-0001	0
34	2.52185536230627E-0002	7.98376320730125E-0003	0
35	-1.88700988824849E-0001	-5.85623758421944E-0002	0
36	-2.04020105388963E-0001	-6.20930755531627E-0002	0
37	-3.05184290883062E-0001	-9.11213759508669E-0002	0
38	1.02831559599690E+0000	3.01320383943278E-0001	0
39	4.08901870999216E-0001	1.17629305355939E-0001	0
40	3.92275429782103E-0002	1.10822206620056E-0002	0
41	-1.82933158878086E-0001	-5.07699956357684E-0002	0
42	1.11622973363637E-0001	3.04426290991738E-0002	0
43	2.51091139393793E-0001	6.73137948162084E-0002	0
44	9.24732016259441E-0001	2.43757811817342E-0001	0
45	-1.20302342404466E+0000	-3.11894961789356E-0001	0
46	5.80684309735261E-0001	1.48109763211828E-0001	0
47	-5.17931909556864E-0001	-1.29998845825030E-0001	0
48	-8.40640367022161E-0002	-2.07687620087828E-0002	0
49	-4.60417639080788E-0001	-1.11993479776408E-0001	0
50	4.04519831346988E-0001	9.69001877371112E-0002	0
51	3.03726061219550E-0001	7.16656998383207E-0002	0
52	-8.09012018564317E-0001	-1.88072904684694E-0001	0
53	4.05951643298522E-0001	9.29998310102068E-0002	0
54	4.27383790321983E-0001	9.65060171694800E-0002	0
55	-9.85208464814605E-0001	-2.19322025736113E-0001	0
56	-2.78451825296088E-0001	-6.11235714064583E-0002	0
57	2.86276628632550E-0001	6.19774144462221E-0002	0
58	9.00391925373469E-0002	1.92287089147554E-0002	0
59	6.63872583207194E-0001	1.39879507498506E-0001	0
60	-7.36181244314139E-0001	-1.53067387431653E-0001	0
61	1.29550491523560E-0001	2.65852800194927E-0002	0
62	6.06320219459121E-0001	1.22823710051205E-0001	0
63	8.26077043482655E-0001	1.65215408696531E-0001	0

The masking process is essential to excluded the unwanted coefficients. It is an important step in the lossy compression of digital speech. We apply masking in our system after the truncation process. The selection of masking pattern was done after examining many samples (about 22 speech files were examined) to choose the pattern and the minimum and maximum values of the 64 output coefficients of the DCT. Not all the scale of the output of the DCT were takes, this will satisfy a reduction in the storage. Table (2) shows the minimum and maximum values of the 64 output coefficients of the DCT and the number the bits required for each coefficient. The number of bits are then used in the line of the codeword assignment.

The number of bits required can be computed after knowing the minimum and maximum for each coefficient as,

$$N1 = \left\lceil \frac{Ln|\min|}{Ln(2)} + 0.9 \right\rceil \quad \textbf{…(7)}$$

$$N2 = \left\lceil \frac{Ln|\max|}{Ln(2)} + 0.9 \right\rceil \quad \textbf{…(8)}$$

$$N = MAX\{N1, N2\} + 1 \quad \textbf{…(9)}$$

Where *min*, *max*, and *N* are the minimum coefficient, maximum coefficient, and the number of required bits, respectively.

5. Compression Ratio

Speech compression involves with the reducing the size of the speech data file, while retaining necessary information. The recording file is called the *compressed speech file* and is used to reconstruct the speech resulting in the *decompressed speech*. The original speech file before any compression is performed, is called the *uncompressed speech file*. The ratio of the original, uncompressed speech file and the compressed speech file is referred to as the *compression ratio*. The compression ratio is denoted by:

$$\text{Compression Ratio} = \frac{\text{Uncompressed Size}}{\text{Compressed Size}} \quad \textbf{…(10)}$$

Table (3) presents the compression ratio of the 64 output coefficients of the DCT.

Figure (6) shows the reconstructed speech after applying the suggested compression process with attained compression ratio 0.13, 0.25, and 0.34.

TABLE 2: The Minimum and Maximum Values of the 64 Output Coefficients of the DCT and its Number of Required Bits.

Coefficient Number	Min. Coefficient	Max. Coefficient	Number of Bits
0	508	1207	9
1	-61	83	6
2	-64	50	6
3	-54	76	6
4	-65	80	6
5	-59	81	6
6	-120	114	7
7	-193	219	7
8	-295	231	7
9	-167	124	7
10	-168	185	7
11	-146	136	7
12	-92	99	6
13	-63	58	6
14	-70	64	6
15	-66	63	6
16	-95	86	6
17	-82	72	6
18	-97	65	6
19	-50	52	6
20	-21	32	5
21	-30	28	5
22	-28	24	5
23	-9	10	4
24	-11	10	4
25	-13	12	4
26	-14	10	4
27	-13	12	4
28	-10	11	4
29	-16	17	5
30	-33	34	5
31	-43	39	6
32	-27	22	5
33	-19	18	5
34	-12	13	4
35	-10	10	4
36	-9	9	4
37	-9	6	4
38	-7	7	4
39	-12	8	4
40	-12	11	4
41	-12	12	4
42	-10	14	4
43	-9	10	4
44	-6	5	4
45	-3	3	3
46	-2	2	2
47	-1	2	2
48	-1	1	2
49	-2	2	2
50	-1	1	2
51	-2	1	2
52	-1	1	2
53	-1	1	2
54	-1	1	2
55	-1	1	2
56	-1	2	2
57	-2	1	2
58	-1	1	2
59	-1	1	2
60	-1	1	2
61	-1	1	2
62	-1	1	2
63	-1	1	2

TABLE 3: The Attained Compression Ratio of the DCT with QF=5.

Coefficient Number	Compression Ratio
0	0.016
1	0.029
2	0.041
3	0.053
4	0.064
5	0.076
6	0.090
7	0.104
8	0.117
9	0.131
10	0.145
11	0.158
12	0.170
13	0.182
14	0.193
15	0.205
16	0.217
17	0.229
18	0.240
19	0.251
20	0.262
21	0.271
22	0.281
23	0.290
24	0.297
25	0.305
26	0.313
27	0.320
28	0.328
29	0.338
30	0.348
31	0.360
32	0.369
33	0.380
34	0.387
35	0.395
36	0.402
37	0.410
38	0.418
39	0.426
40	0.433
41	0.441
42	0.450
43	0.457
44	0.465
45	0.471
46	0.475
47	0.479
48	0.482
49	0.486
50	0.490
51	0.494
52	0.498
53	0.502
54	0.506
55	0.510
56	0.514
57	0.518
58	0.521
59	0.525
60	0.529
61	0.533
62	0.537
63	0.541

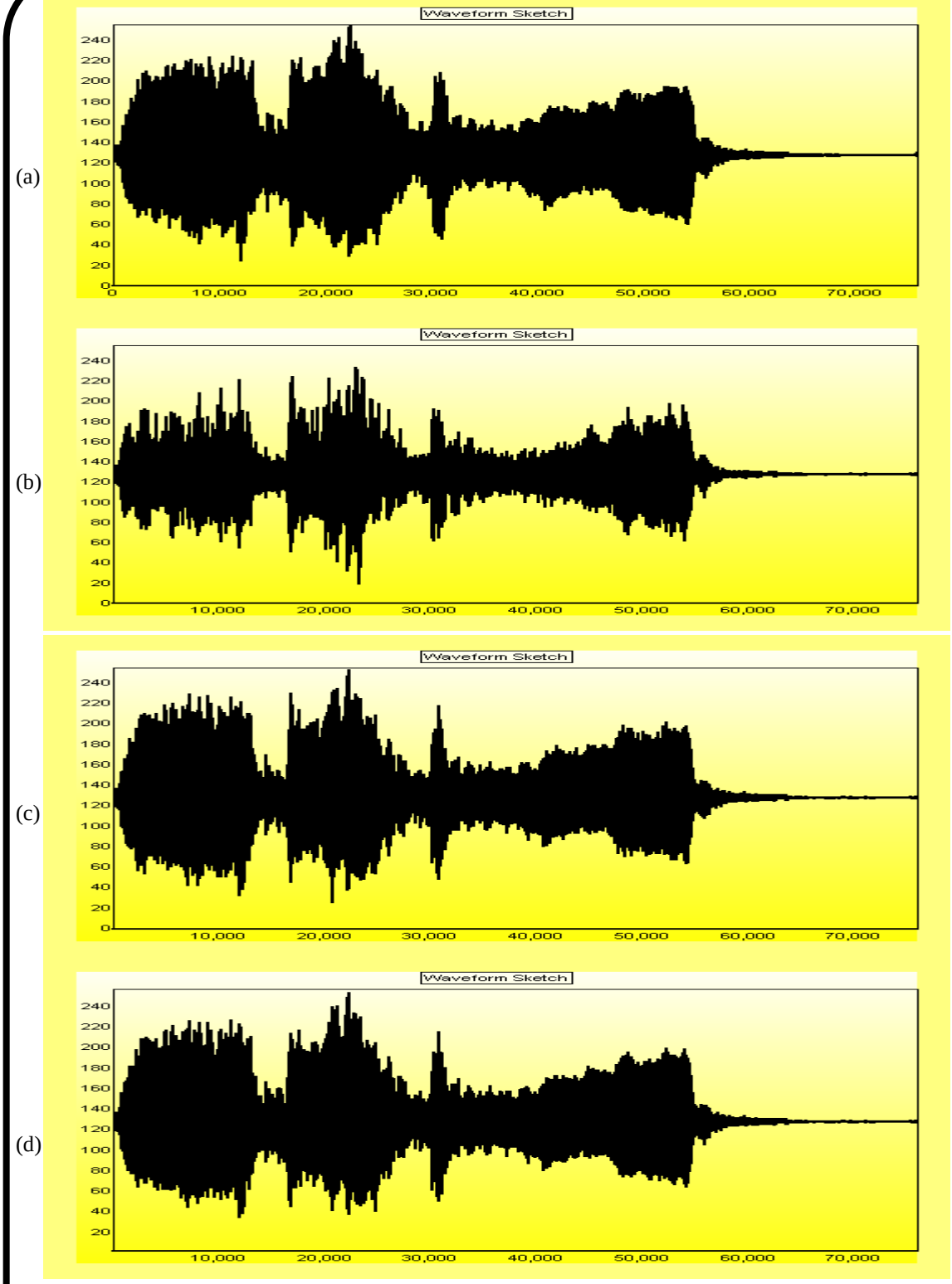


FIGURE 6: ***(a) Original Speech, (b), (c), and (d) Reconstructed of (a) with Compression Ratio 0.13, 0.25, and 0.34 Respectively.***

It is important to mention here, that the attained compression ratio mentioned in this section is not the final compression ratio of the speech file. The final compression ratio of a given speech file depends also on the number of the unvoiced frames may exist in the speech file.

6. Output File Format

In order to record all the information required by the expander (decoder) to perform the reconstruction of the coded voiced frames; a single output file was constructed to hold all the information. From the size of this output file we can determined the ultimate compression ratio gained by our proposed compression system. The output file consists of the *data*, *run-length code*, and *compressed data* sections.

The data section consists of the compression ratio used to compress the output coefficients of the DCT and the flag (0 or 1) to assign whether the first collected frames are voiced or unvoiced.

The compressed data section is the quantized indices of the coefficients of the DCT, this section which put after the run-length of the collected voiced frames. Figure 7 shows the output file format sections.

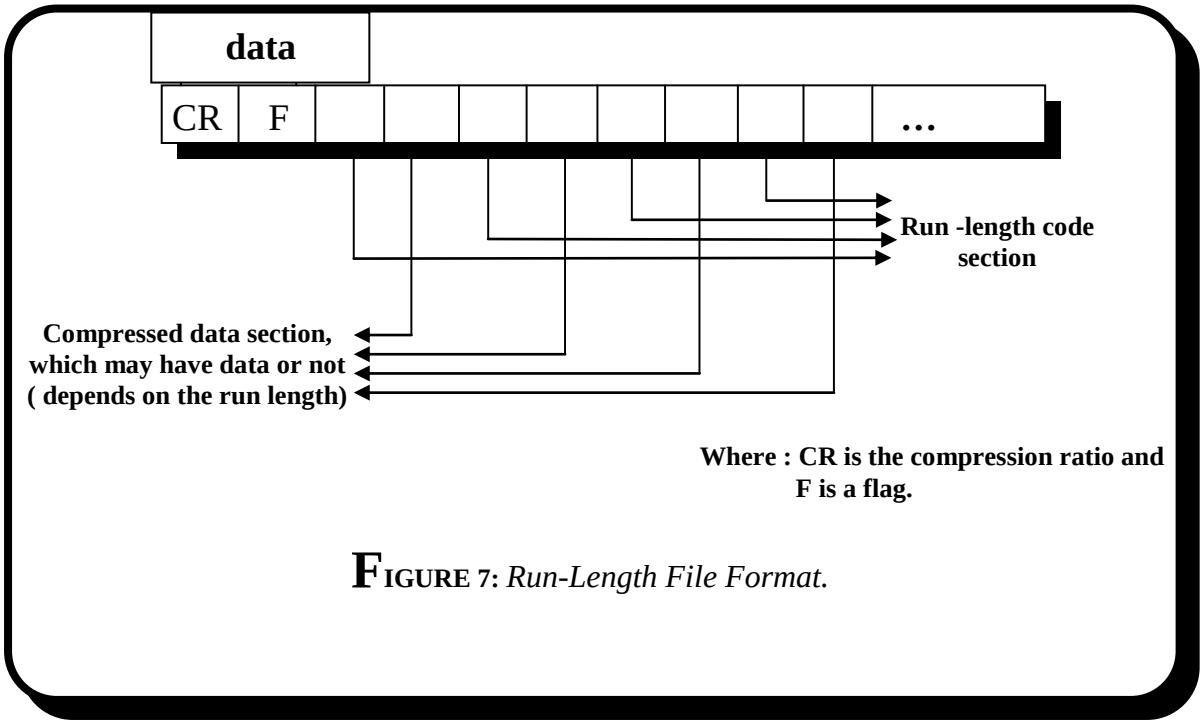


FIGURE 7: *Run-Length File Format.*

7.Conclusions

Data compression is a technique for reducing the redundancy in the data system in order to improve the performance of the transmission to transmit and process true information. Compression methods are playing an increasingly important role in speech storage and transmission. Many techniques are now available and many efforts are being expended in determining the optimum techniques to use different type of data.

Although, compression techniques themselves are complex, there are simple rules that can be used to avoid disappointment. Used wisely, compression has a number of advantages. Used in an inappropriate manner, disappointment is almost inevitable and the technology could get a bad name.

References

[1]. Tim Kientzle **"A Programmer's Guide to Sound"** Addison –Wesely Developer Press.1998

[2]. Ernest L. Hall **"Computer Image Processing and Recognition"** Academic Press.1979

[3]. John Watkinson **"Compression in Video and Audio"** Hartnolls Limited.1997

[4]. Kruti Dangarwala and Jigar Shah **"Implementation and Comparison of Companding and Silence audio Compression Techniques"**.2010

[5]. M. A. Shaalan **"New High Synthetic Coding Methods for Compressing digital Speech Signals"** M. Sc. Thesis, University of Baghdad.2000

[6]. Yon Q. Shi and Huifang Sun **"Image and Video Compression for Multimedia Engineering"** CRC Press.2000

[7]. Daniel Gent **"Speech and Music Compression"** 2008