NATIONAL OPEN UNIVERSITY OF NIGERIA

# CIT 742

## Multimedia Technology

## Module 4

# CIT 742 Multimedia Technology  Module 4

**Course Developer/Writer**
Vivian Nwaocha, National Open University of Nigeria

**Course Coordinator**
Vivian Nwaocha, National Open University of Nigeria

**Programme Leader**
Dr. B. Abiola, National Open University of Nigeria

Credits of cover-photo: Henry Ude, National Open University of Nigeria

**National Open University of Nigeria  -** 91, Cadastral Zone, Nnamdi Azikiwe Express Way, Jabi, Abuja, Nigeria

www.nou.edu.ng centralinfo@nou.edu.ng
oer.nou.edu.ng oerunit@noun.edu.ng OER repository

Published in 2021 by the National Open University of Nigeria



## How to re-use and attribute this content

# Module 4 Multimedia Compression

# Unit1 Rudiments of Multimedia Compression

## 1.0 Introduction

Essentially, video and audio files are very large. Unless we develop and maintain very high bandwidth networks (Gigabytes per second or more) we have to compress these files. However, relying on higher bandwidths is not a good option.

Thus, compression has become part of the representation or *coding* scheme. In this unit, we will learn about basic compression algorithms and then go on to study some actual coding formats.

## 2.0 Objectives

At the end of this unit, you should be able to:

- explain what transform coding is
- describe the notion of lossless compression
- identify typical applications of lossy and lossless compression
- discover how images are represented in the different forms of encoding
- Identify the basic lossy compression schemes.

## 3.0 Main Content

### 3.1 Overview of Multimedia Compression?

**Multimedia compression** is a broad term that refers to the compression of any type of multimedia (i.e. combination of media and content forms), most notably graphics, audio, and video.

Multimedia actually derives from data sampled by a device such as a camera or a microphone. Such data contains large amounts of random noise, thus, traditional lossless compression algorithms tend to do a poor job compressing multimedia. Multimedia compression algorithms, traditionally known as ***codecs***, work in a lossy fashion, the entire process is known as transform coding.

The term **Transform coding** is a technique for compressing signals such as audio signals (1-D) or images (2-D). In transform coding, a frequency transform or other basic transformation is applied before entropy coding. The inverse transformation is applied after decoding. This has a considerable benefit since it produces coefficients that have a statistically significant distribution which can be modeled and compressed more easily.

This implies that, after transformation, some coefficients are predictably greater, others smaller. Thus, some coefficients can be neglected or quantized (lossy compression) and/or entropy encoded (lossless compression). We shall see these categories of compression in the ensuing unit.

## Self-Assessment Exercise

Give a concise description of the term 'multimedia compression'

## 3.2 Categories of Multimedia Compression

Multimedia compression can be broadly classified as Lossless and Lossy compression.

### 3.2.1 Lossy Compression

In information technology, "***lossy***" compression is a data encoding method which discards some of the data, in order to achieve its goal, with the result that decompressing the data yields content that is different from the original, though similar enough to be useful in some way.

Lossy compression is most commonly used to compress multimedia data (audio, video, still images), especially in applications such as streaming media and internet telephony. Lossy compression formats suffer from generation loss: repeatedly compressing and decompressing the file will cause it to progressively lose quality. Information-theoretical foundations for lossy data compression are provided by rate-distortion theory.

**Types of Lossy Compression Schemes**

There are two basic lossy compression schemes:

- In *lossy transform codecs*, samples of picture or sound are taken, chopped into small segments, transformed into a new basis space, and quantized. The resulting quantized values are then entropy coded.

- In *lossy predictive codecs*, previous and/or subsequent decoded data is used to predict the current sound sample or image frame. The error between the predicted data and the real data, together with any extra information needed to reproduce the prediction, is then quantized and coded.

In some systems, the two techniques are combined, with transform codecs being used to compress the error signals generated by the predictive stage.

### 3.2.2 Lossless Compression

**Lossless compression** is a compression technique that does not lose any data in the compression process. This compression "packs data" into a smaller file size by using a kind of internal shorthand to signify redundant data. If an original file is 1.5MB, lossless compression can reduce it to about half that size, depending on the type of file being compressed. This makes lossless compression convenient for transferring files across the Internet, as smaller files transfer faster. Lossless compression is also handy for storing files as they take up less room.

The zip convention, used in programs like WinZip, uses lossless compression. For this reason zip software is popular for compressing program and data files. That's because when these files are decompressed, all bytes must be present to ensure their integrity. If bytes are missing from a program, it won't run. If bytes are missing from a data file, it will be incomplete and garbled. GIF image files also use lossless compression.

Lossless compression has advantages and disadvantages. The advantage is that the compressed file will decompress to an exact duplicate of the original file, mirroring its quality. The disadvantage is that the compression ratio is not all that high, precisely because no data is lost.

**Typical Lossless Compression File Formats**

**Audio**

- Waveform audio format (WAV)
- Free Lossless Audio Codec (FLAC)
- Apple Lossless Audio Codec (ALAC)
- ATRAC Advanced Lossless
- Audio Lossless Coding
- MPEG-4 SLS
- Direct Stream Transfer (DST)
- DTS-HD Master Audio
- Meridian Lossless Packing (MLP)
- Monkey's Audio APE
- RealPlayer – RealAudio Lossless
- Shorten – SHN,TTA – True Audio Lossless
- WMA Lossless

**Graphics**

- Adaptive Binary Optimization (ABO)
- JPEG XR
- Progressive Graphics File (PGF)
- Portable Network Graphics (PNG)
- Tagged Image File Format (TIFF)

**Video**

- Animation codec
- CorePNG, Dirac
- FFV1
- JPEG 2000
- Huffyuv
- Lagarith
- MSU Lossless Video Codec
- SheerVideo

## 3.3 Lossy versus Lossless

Lossless and lossy compressions have become part of our every day vocabulary largely due to the popularity of MP3 music files. A standard sound file in WAV format, converted to an MP3 file will loose much data as MP3 employs a lossy, high-compression algorithm that tosses much of the data out. This makes the resulting file much smaller so that several dozen MP3 files can fit, for example, on a single compact disk, verses a handful of WAV files. However the sound quality of the MP3 file will be slightly lower than the original WAV.

The advantage of lossy methods over lossless methods is that in some cases a lossy method can produce a much smaller compressed file than any lossless method, while still meeting the requirements of the application.

Lossy methods are most often used for compressing sound, images or videos. This is because these types of data are intended for human interpretation where the mind can easily "fill in the blanks" or see past very minor errors or inconsistencies – ideally lossy compression is transparent (imperceptible), which can be verified via an ABX test.

## 3.4 Mathematical and Wavelet Transformation

Mathematical and wavelet transformation is the process through which images are converted to mathematical functions. Discreet Cosine Transformation uses series of cosine functions to approximate image.  This technique is used with JPEG, MPEG1 and MPEG 2 formats. A wavelet function is used to approximate the image. This can be used with the JPEG 2000 and MPEG 4 formats.

### 3.4.1 JPEG Encoding

In this encoding, an image is represented by a two dimensional array of pixels. A Grayscale picture of 307*200 pixels is represented by 2,457,600 bits and a color picture is represented by 7,372,800 bits. Due to the number of calculations to be had in a JPEG format of a grayscale picture, it is divided into blocks of 8*8 pixels. The number of the units' id equal to the number of mathematical equations of each picture. The whole idea of JPEG is to change the picture into a linear set of numbers that reveals the redundancies. In addition to those techniques, MPEG is also a Lossy Compression technique. It is a way to encode the moving images and audio included in it. It supports many video formats from mobile phone to HD TV.

### 3.4.2 H.261, H.263, H.264

H.261 is designed for video telephony and video conferencing applications. It was developed in 1988-1990. Data rate is a multiplication of a 64 kb/s. H.263 is a video coding technique for low bit rate communication. In addition, a 30% of bit saving can be done by this technique when it is compared to the MPEG-1. H.264 is a joint project of ITU-Ts Video Experts Group and the ISO/IEC MPEG group. All those three methods use different methods of reducing redundant data. There for the output differs from bit rate, quality and latency.

## 4.0 Conclusion

In this unit, we learnt that **Transform coding** is a technique for compressing signals such as audio signals (1-D) or images (2-D). Multimedia compression can be broadly classified as Lossless and Lossy compression. Discreet Cosine Transformation uses series of cosine functions to approximate image. This technique is used with JPEG, MPEG1 and MPEG2 formats. A wavelet function is used to approximate the image. We equally discovered that wavelet transform techniques will be a good implementation technique for the next generation.

## 5.0 Summary

In this unit, we considered multimedia compression and the categories of multimedia compression (lossy and lossless). We equally learnt about the specific applications of these categories of multimedia. You may now proceed to the tutor marked assignment below.

## 6.0 Self-Assessment Exercise

1. What is Transform coding?
2. Describe the notion of lossless compression
3. Give 3 applications of lossy compression
4. Describe JPEG encoding
5. State 2 basic lossy compression schemes

## 7.0 References/Further Reading

A.M. Tekalp. (1995). Digital video processing, Prentice Hall PTR.

Boyle. (1998).Design for Multimedia Learning, Prentice Hall, (ISBN 0-13-242155-8)

Brook and Wynne, Hodder and Stoughton (2001).Signal Processing: Principles and Applications.

Encyclopedia of Graphics File Formats, Second Edition by James D. Murray and William vanRyper, O'Reilly & Associates, 1996 (ISBN: 1-56592-161-5)

Fluckiger. (1994). Understanding Networked Multimedia, Prentice Hall.

H. Maurer, Addison Wesley, (1996). Hyperwave: The Next Generation Web Solution, (ISBn 0-201-40346).

Intro. to Computer Pictures, http://ac.dal.ca:80/ dong/image.htm from Allison Zhang at the School of Library and Information Studies, Dalhousie University, Halifax, N.S., Canada

J. G. Shuman, (2002). Multimedia Elements. *Multimedia In Action.* Vikas Publishing House Pvt Ltd.

J. Vince, Addison Wesley, (1995).Virtual Reality Systems, (ISBN 0-201-87687-6)

J.F.K, Buford. (1994). Multimedia Systems, ACM Press, 1994 (ISBN 0-201-53258-1).

James D. Murray and William vanRyper, (1996). Encyclopedia of Graphics File Formats, Second Edition, O'Reilly & Associates.

Lowe and W. Hall, J. (1999). Hypermedia and the Web: An Engineering Approach, 1999 (ISBN 0-471-98312-8).

P.W. Agnew and A.S. Kellerman. (1996). Distributed Multimedia: Technologies, Applications, and Opportunities in the Digital Information Industry (1st Edition) Addison Wesley.

Sloane, McGraw Hill. (2002). Multimedia Communication, (ISBN 0-077092228)

T. Kientzle, Addison Wesley, 1997. A programmer's Guide to Sound, (ISBN 0-201-41972-6)

Vaughan, Tay, 1993, Multimedia: Making It Work (first edition, ISBN 0-07-881869-9), Osborne/McGraw-Hill, Berkeley, pg. 3.

Watkinson, (2004). The Art of Digital Audio, -Heinmann. Synthesizer Basics, GPI Publications.

# Unit 2 Source Coding Techniques

## 1.0 Introduction

In this unit, you'll gain knowledge of fourier transform. You would also learn about the common types of transforms and how signals are transformed from one domain to another. Do take note of these key points.

## 2.0 Objectives

At the end of this unit, you should be able to:

- explain the concept of the Discreet Fourier Transform
- identify the common types of transforms
- describe how signals are transformed from one domain to the other
- identify the different classes of signals
- Discover the common applications of Fourier transforms in multimedia processing.

## 3.0 Main Content

### 3.1 Discrete

The Discrete Fourier Transform (DFT) is a specific form of Fourier analysis for converting one function (often in the time or spatial domain) into another (frequency domain). DFT is widely employed in signal processing and related fields to analyze frequencies contained in a sample signal, to solve partial differential equations, and to perform other operations such as convolutions. Fast Fourier Transform (FFT) is an efficient implementation of DFT and is used, apart from other fields, in digital image processing. Fast Fourier Transform is applied to convert an image from the image (spatial) domain to the frequency domain. Applying filters to images in frequency domain is computationally faster than to do the same in the image domain.

Mathematically, Suppose

f = [ $f_{0,} f_{1,} f_{2,} \ldots, f_{N-1}$ ]

is a sequence of length N. Then the *discrete Fourier transform* can be defined as:

f = [ $F_{0,} F_{1,} F_{2,} \ldots, F_{N-1}$ ] 3.1

Where

$$Fu = \frac{1}{N}\sum_{x=0}^{N-1}\exp\left[-2\pi i\frac{xu}{N}\right]f_x \qquad 3.2$$

The formula for the inverse DFT is very similar to the forward transform:

$$x_u = \sum_{x=0}^{N-1}\exp\left[2\pi i\frac{xu}{N}\right]f_u \qquad 3.3$$

When you try to compare equations 3.2 and 3.3., you will notice that there are really only two differences:

1 there is no scaling factor 1/N

2 the sign inside the exponential function is now positive instead of negative

## 3.2 The Fast Fourier Transform.

One of the many aspects which make the DFT so attractive for image processing is the existence of very fast algorithm to compute it. There are a number of extremely fast and efficient algorithms for computing a DFT; any of such algorithms is called a fast Fourier transform, or FFT. When an FFT is used, it reduces vastly the time needed to compute a DFT.

A particular FFT approach works recursively by separating the original vector into two halves as represented in equation 3.4 and 3.5, computing the FFT of each half, and then putting the result together. This means that the FFT is most different when the vector length is a power of 2.

$$F(u) = \sum_{x=0}^{M-1} f(x)\exp\left[-2\pi i\frac{xu}{M}\right] \qquad 3.4$$

$$f(x) = \frac{1}{M}\sum_{v=0}^{M-1} F(u)\exp\left[2\pi\frac{xu}{M}\right] \qquad 3.5$$

Table 3.1 is used to depict the benefits of using the FFT algorithm as opposed to the direct arithmetic definition of equation 3.4 and 3.5 by comparing the number of multiplication required for each method. For a vector of length $2^n$, the direct method takes $(2^n)^2 = 2^{2n}$ multiplications; while the FFT takes only $n2^n$. Here the saving with respect to time is of an order of $2^n$/n. Obviously, it becomes more attractive to use FFT algorithm as the size of the vector increases.

Because of this computational advantage, it is advisable for any implementation of the DFT to use an FFT algorithm.

| $2^n$ | Direct Arithmetic | FFT | Increase in speed |
|---|---|---|---|
| 4 | 16 | 8 | 2.0 |
| 8 | 84 | 24 | 2.67 |
| 16 | 256 | 64 | 4.0 |
| 32 | 1024 | 160 | 6.4 |
| 64 | 4096 | 384 | 10.67 |
| 128 | 16384 | 896 | 18.3 |
| 256 | 65536 | 2048 | 32.0 |
| 512 | 262144 | 2406 | 56.9 |
| 1024 | 1048576 | 10240 | 102.4 |

**Table 3.1: Comparison of FFT and direct arithmetic**

## 3.3 Two-Dimensional Discrete Fourier Transform (DFT)

In two dimensions, the DFT takes a matrix as input, and returns another matrix, of the same size as output. If the original matrix values are $f(x,y)$, where $x$ and $y$ are the indices, then the output matrix values are $F(u,v)$. We call the matrix $F$ the Fourier transform $f$ and write

$$F = F(f).$$

Then the original matrix $f$ is the inverse Fourier transforms of F, and we write

$$f = F^{-1}(F)$$

We have seen that a (one-dimensional) function can be written as a sum of sines and cosines. Given that an image may be considered as a two dimensional function, it seems reasonable to assume that F can be expressed as sums of “corrugations” functions which have the general form

$$z = a \sin(bx+cy)$$

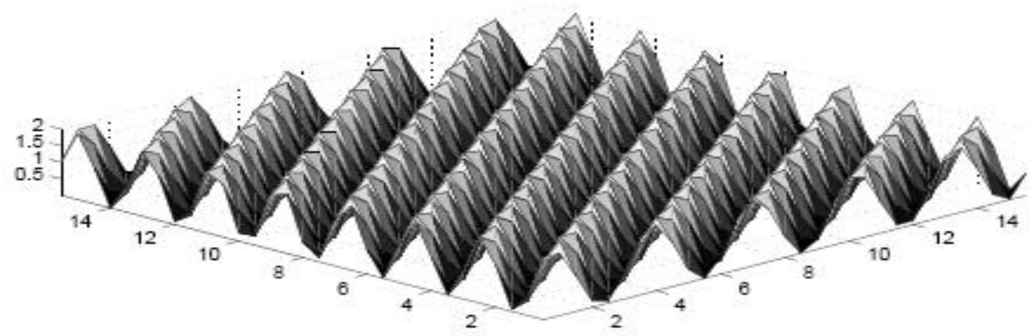



Figure 3.2: Corrugate Function

A sample of such function is depicted in figure 3.2. And this is in fact exactly what the two dimensional Fourier transforms does: it rewrites the original matrix in terms of sums of corrugation.

The definition of the two-dimensional discrete Fourier transform is very similar to that for one dimension. The forward and inverse transforms for an *M x N* matrix where for notational convenience we assume that the x indices are from 0 to *M-1* and the *y* indices are from 0 to *N-1* are:

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\exp\left[-2\pi i\left(\frac{xu}{M}+\frac{yv}{N}\right)\right] \qquad 3.6$$

$$F(x,y) = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F(u,v)\exp\left[-2\pi i\left(\frac{xu}{M}+\frac{yv}{N}\right)\right] \qquad 3.7$$

You may need to revise your mathematics to fully comprehend the formulas. However, they are not as difficult as they look.

## Self-Assessment Exercise

What is the factor that makes the DFT so attractive for image processing?

## 3.4 Properties of Two-Dimensional Fourier Transform

All the properties of the one-dimensional DFT transfer into two dimensions. We shall briefly consider some which are of particular use for image processing.

**Similarity**. A close study of the formulae for the forward and inverse transforms reveals some similarity except for the scale factor *1/M N* in the inverse transform and the negative sign in the exponent of the forward transform. This means that the same algorithm, only very slightly adjusted, can be used for both the forward and inverse transform. The DFT can thus be used as a spatial Filter

**Linearity -** An important property of the DFT is its linearity; the DFT of a sum is equal to the sum of the individual DFT's, and the same goes for scalar multiplication:

Thus

F (f+g) = F(f) + F (g)

F *(kf)* = k F*(f)*

Where *k* ? is a scalar product and *f* and *g* are matrices. This follows directly from the definition given in equation 3.6

This property is of great use in dealing with image degradation such as noise which can be

modeled as a sum: *d=f + n*

where *f* is the original image, *n* is the noise, and *d* is the degraded image. Since

F (*d*) = F(*f*) + F (*n*)

We may be able to remove or reduce *n* by modifying the transform. And we shall see some noise appear on the DFT in a way which makes it particularly easy to remove

## 3.5 The Discrete-time Fourier Transform (DTFT)

The **Discrete-time Fourier transform (DTFT)** is one of the specific forms of Fourier analysis. As such, it transforms one function into another, which is called the frequency domain representation, or simply the "DTFT", of the original function (which is often a function in the time-domain). But the DTFT requires an input function that is discrete. Such inputs are often created by sampling a continuous function, like a person's voice.

The DTFT frequency-domain representation is always a periodic function. Since one period of the function contains all of the unique information, it is sometimes convenient to say that the DTFT is a transform to a "finite" frequency-domain (the length of one period), rather than to the entire real line.

Given a discrete set of real or complex numbers: x[n], n $\in Z$ (integer), the discrete-time Fourier transform (DTFT) is written as:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-i\omega n}$$

The following inverse transforms recovers the discrete-time sequence

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(\omega).e^{i\omega n} d\omega$$

$$= T \int_{-\frac{1}{2T}}^{\frac{1}{2T}} X_T(f).e^{i2\pi fnT} df$$

Since the DTFT involves infinite summations and integrals, it cannot be calculated with a digital computer. Its main use is in theoretical problems as an alternative to the DFT. For instance, suppose you want to find the frequency response of a system from its impulse response. If the impulse response is known as an array of numbers, such as might be obtained from an experimental measurement or computer simulation, a DFT program is run on a computer. This provides the frequency spectrum as another array of numbers, equally spaced between, for example, 0 and 0.6 of the sampling rate. In other cases, the impulse response might be given as an equation, such as a sine function or an exponentially decaying sinusoid. The DTFT is used here to mathematically calculate the frequency domain as another equation, specifying the entire continuous curve between 0 and 0.6. While the DFT could also be used for this calculation, it would only provide an equation for samples of the frequency response, not the entire curve.

## 3.6 Discrete Cosine Transform (DCT)

A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. In particular, a DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers. DCTs are equivalent to DFTs of roughly twice the length, operating on real data with even symmetry (since the Fourier transform of a real and even function is real and even), where in some variants the input and/or output data are shifted by half a sample. The cosine transform, like Fourier Transform, uses sinusoidal basis functions. The difference is that the cosine transform basis functions are not complex; they use only cosine functions, and not sine functions. The two-dimensional discrete cosine transform (DCT) equation for an N x N image for an example is as given by:

$$F(u,v) = C(u)C(v)\sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f[m.n]\cos\left(\frac{(2m+1)u\pi}{2N}\right)\cos\left(\frac{(2n+1)v\pi}{2N}\right)$$

for $0 \le u,v < N$

with $C(u) = \begin{cases} \sqrt{1/N} \; for u=0 \\ \sqrt{2/N} \; for u \neq 0 \end{cases}$

We can interpret this as the projection of *f* [*m*,*n*] onto basis functions of the form:

$$e_{u,v}[m,n] = C(u,v)\cos\left(\frac{(2m+1)u\pi}{2N}\right)\cos\left(\frac{(2n+1)v\pi}{2N}\right)$$

Since this transform uses only the cosine function it can be calculated using only real arithmetic, instead of complex arithmetic as the DFT requires. The cosine transform can be derived from the Fourier transform by assuming that the function (the image) is mirrored about the origin, thus making it an even function. Thus, it is symmetric about the origin. This has the effect of canceling the odd terms, which correspond to the sine term (imaginary term) in Fourier transform. This also affects the implied symmetry of the transform, where we now have a function that is implied to be 2N x 2N.

Some sample basis functions are shown in Figure. 3.5, for a value of *N*=8.

Based on the preceding discussions, we can represent an image as a superposition of weighted basis functions (using the inverse DCT):

$$f[m,n] = \sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C(u)C(v)F[u,v]\cos\left(\frac{(2m+1)u\pi}{2N}\right)\cos\left(\frac{(2n+1)v\pi}{2N}\right)$$

for $0 \le m,n < N$

with $C(u) = \begin{cases} \sqrt{1/N} \; for u=0 \\ \sqrt{2/N} \; for u\neq 0 \end{cases}$

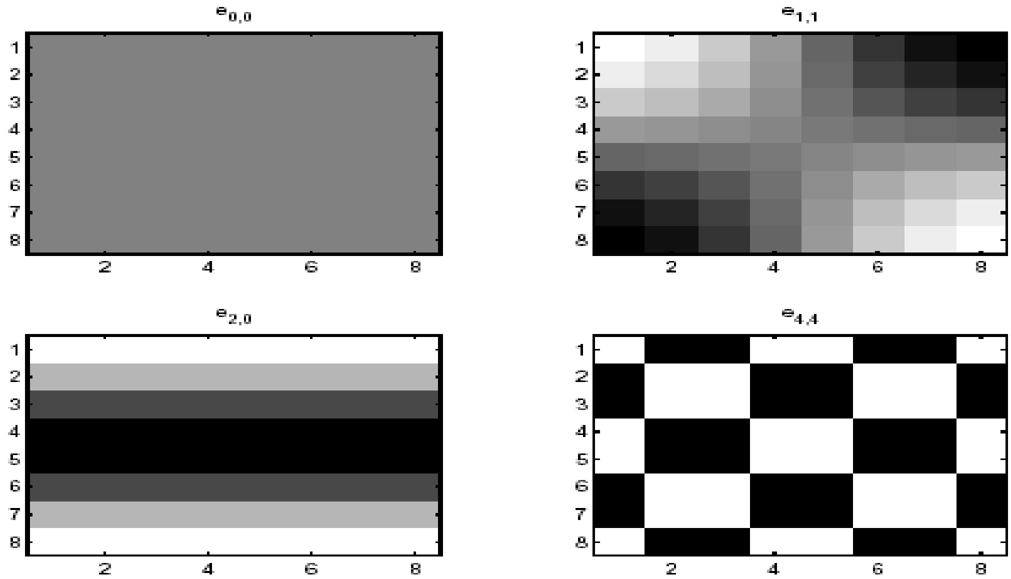



**Figure 3.5:** Sample basis functions for an 8x8 block of pixels.

The above four have been chosen out of a possible set of 64 basis functions.

This goes to show that DCT coefficients are similar to Fourier series coefficients in that they provide a mechanism for reconstructing the target function from the given set of basis functions. In itself, this is not particularly useful, since there are as many DCT coefficients as there were pixels in the original block. However, it turns out that most real images (natural images) have most of their energy concentrated in the lowest DCT coefficients. This is explained graphically in Figure 3.6 where we show a 32 x 32 pixel version of the test image, and its DCT coefficients. It can be shown that most of the energy is around the (0,0) point in the DCT coefficient plot. This is the motivation for compression – since the components for high values of *u* and *v* are small compared to the others, why not drop them, and simply transmit a subset of DCT coefficients, and reconstruct the image based on these. This is further illustrated in Figure 3.7, where we give the reconstructed 32 x 32 image using a small 10x10 subset of DCT coefficients. As you can see there is little difference between the overall picture of Figure 3.6(a) and Figure 3.7, so little information has been lost. However, instead of transmitting 32x32=1024 pixels, we only transmitted 10x10 =100 coefficients, which is a compression ratio of 10.24 to 1.

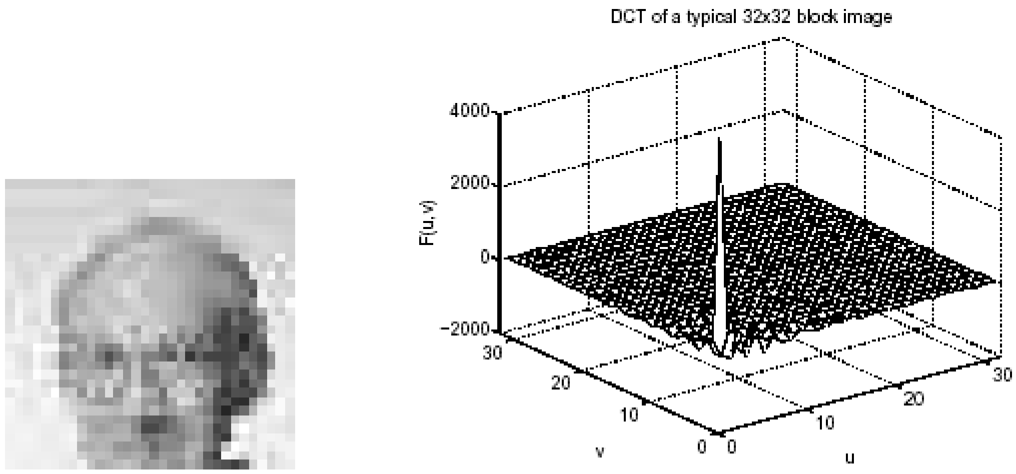



**Figure 3.6**: (a) 32 x 32 pixel version of our standard test image. (b) The DCT of this image.

**Figure 3.7**: 32 x 32 pixel image reconstructed from 10x10 subset of DCT coefficients. Overall information has been retained, but some detail has been lost.

An optimal transform for compression would maximise the "energy-compressing" feature of the transform; that is the transform of the image would have most of its energy in the fewest number of coefficients. The DCT is not the optimal transform from this perspective; it can be shown mathematically that a Karhunen-Loeve transform,  will provide the best basis for compression. However, this optimal basis is image-dependent and computationally intensive to find, so it is not commonly used in image compression systems.

DCTs are important to numerous applications in science and engineering, from lossy compression of audio and images (where small high-frequency components can be discarded), to spectral methods for the numerical solution of partial differential equations. The use of cosine rather than sine functions is critical in these applications: for compression, it turns out that cosine functions are much more efficient whereas for differential equations the cosines express a particular choice of boundary conditions. The DCT is the basis of many widespread image-coding standards: specifically, JPEG, MPEG, and H.26X which are respectively still image, video-playback, and video telephony standards.

## 4.0 Conclusion

To wrap up, we discovered that 'Fourier Transform' is of fundamental importance to image processing. It allows us to perform tasks which would be impossible to perform by any other way; its efficiency allows us to perform most compression tasks more quickly. The Fourier transform is a very useful mathematical tool for multimedia processing. The Fourier Transform and the inverse Fourier transforms are the mathematical tools that can be used to switch from one domain to the other.

## 5.0 Summary

In this unit, we covered the definition of Fourier transforms, types of Fourier transform and its application in digital image processing

## 6.0 Self-Assessment Exercise

a. What is the significance of fourier transform?

b. State the link between fourier series coefficient and discreet cosine transform coefficient

c. Give a concise description of the discrete time fourier transform

d. List 3 common image-coding standards

e. Explain two (2) properties of the two dimensional Fourier transform

## 7.0 References/Further Reading

McAndrew (2004), An Introduction to Digital Image Processing with Matlab, School of Computer Science and Mathematics, Victoria University of Technology

J. S. Lim(1990), Two-Dimensional Signal and Image Processing. Prentice Hall, 1990.

William K. Pratt (1991), Digital Image Processing. John Wiley and Sons, second edition,

J. D. Gibson (Eds) (2001), Multimedia Communications: Directions and Innovations, Academic Press, San-Diego, USA

Schuler, M. Chugani (2005), Digital Signal Processing, A Hands-on Approach, McGraw Hills, USA

# Unit 3 Video and Audio Compression

## 1.0 Introduction

In this unit, you have a chance to learn another aspect of Multimedia Technology. We will study about video compression. We'll equally learn about different audio compressions.

## 2.0 Objectives

At the end of this unit, you should be able to:

- explain the principle of video compression
- describe the JPEG algorithm approach
- explain the application of video compression
- discover the sensitivity of human hearing
- describe the notion of psychoacoustics.

## 3.0 Main Content

### 3.1 Principle of Video Compression

The principle of still image compression is very similar to that of video compression. Video is simply sequence of digitized picture. Video can also be referred to as moving picture. The terms "frame" and "pictures" are used interchangeably in relation to video. However, we shall use the term frame in relation to videos except where particular standard uses the term picture.

In principle, one way to compress video source is to apply any of the common algorithms such as JPEG algorithm independently to each frame that makes up a video. This approach is also known as moving JPEG or MPEG.  For now typical compression ratios of about 29:1 obtained with JPEG are not large enough to produce the compression ratio needed for multimedia applications.

In practice, in addition to the spatial redundancy present in each frame considerable redundancy is often present between a set of frame since, in general, only a small portion  of each frame is involved with any motion that is generally, only a small portion of each frame is involved with any motion that is taking place. For an example, consider the movement of a person's lip or eye in a video telephony application.

### 3.2 Application of Video Compression

Haven studied the theory of encoding now let us see how this is applied in practice.

Video (and audio) need to be compressed in practice for the following reasons:

1 Uncompressed video (and audio) data are huge. In HDTV, the bit rate easily exceeds 1 Gbps. -- big problems for storage and network communications. For example: One of the formats defined for HDTV broadcasting within the United States is 1920 pixels horizontally by 1080 lines vertically, at 30 frames per second. If these numbers are all multiplied together, along with 8 bits for each of the three primary colors, the total data rate required would be approximately 1.5 Gb/sec. Because of the 6 MHz. channel bandwidth allocated, each channel will only support a data rate of 19.2 Mb/sec, which is further reduced to 18 Mb/sec by the fact that the channel must also support audio, transport, and ancillary data information. As can be seen, this restriction in data rate means that the original signal must be compressed by a figure of approximately 83:1. This number seems all the more impressive when it is realized that the intent is to deliver very high quality video to the end user, with as few visible artifacts as possible.

2 Lossy methods have to be employed since the *compression ratio* of lossless methods (e.g., Huffman, Arithmetic, LZW) is not high enough for image and video compression, especially when distribution of pixel values is relatively flat.

The following compression types are commonly used in Video compression:

Spatial Redundancy Removal - Intraframe coding (JPEG)

- Spatial and Temporal Redundancy Removal - Intraframe and Interframe coding (H.261, MPEG)

## Self-Assessment Exercise

Which compression method is preferable in the image and video compression context?

## 3.2 Audio Compression

As with video a number of compression techniques have been applied to audio. We shall consider the common ones in the subsequent units.

### 3.2.1 Simple Audio Compression Methods

The following are some of the Lossy methods applied to audio compression:

- Silence Compression - detect the "silence", similar to run-length coding
- Adaptive Differential Pulse Code Modulation (ADPCM) e.g., in CCITT G.721 - 16 or 32 Kbits/sec.
- encodes the difference between two consecutive signals,
- adapts at quantization so fewer bits are used when the value is smaller.
- It is necessary to predict where the waveform is headed -> difficult
- Apple has proprietary scheme called ACE/MACE. Lossy scheme that tries to predict where wave will go in next sample. About 2:1 compression.

- Linear Predictive Coding (LPC) fits signal to speech model and then transmits parameters of model. Sounds like a computer talking, 2.4 kbits/sec.
- Code Excited Linear Predictor (CELP) does LPC, but also transmits error term - audio conferencing quality at 4.8 kbits/sec.

### 3.2.2 Psychoacoustics

These methods are related to how humans actually hear sounds

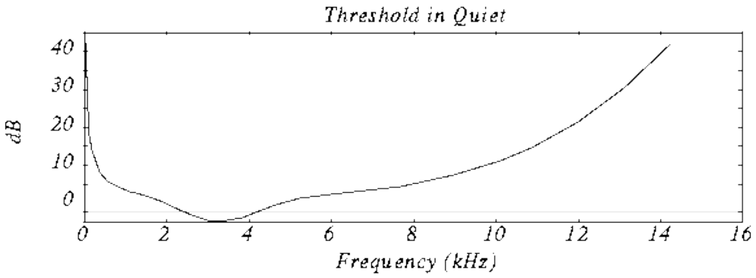
## 3.3 Human Hearing and Voice

These methods are related to how humans actually hear sounds

- Range is about 20 Hz to 20 kHz, most sensitive at 2 to 4 KHz.
- Dynamic range (quietest to loudest) is about 96 dB
- Normal voice range is about 500 Hz to 2 kHz
- Low frequencies are vowels and bass
- High frequencies are consonants

**Question: How sensitive is human hearing?**

- Experiment: Put a person in a quiet room. Raise level of 1 kHz tone until just barely audible. Vary the frequency and plot




In sum,

- If we have a loud tone at, say, 1 kHz, then nearby quieter tones are masked.
- Best compared on critical band scale - range of masking is about 1 critical band
- Two factors for masking - frequency masking and temporal masking

## 3.4 Streaming Audio (and Video)

This is the popular delivery medium for the Web and other Multimedia networks

Examples of streamed audio (and video) (and video)

- Real Audio
- Shockwave
- wav files (not video obviously)

If you hYou could try The file was originally recorded at CD Quality (44 Khz, 16-bit Stereo) and is nearly minutes in length.

The original uncompressed file is about 80 Mb.

The compressed file (at 33.3) is only 1.7 Mb in total and is still of very good quality.

The file is downloaded to browser and not steamed above. Whilst real audio players and encoders are freely available ( see {\em http://www.realaudio.com/}). Real audio servers {\bf cost money}.

- Buffered Data:
- Trick get data to destination before it's needed
- Temporarily store in memory (Buffer)
- Server keeps feeding the buffer
- Client Application reads buffer
- Needs Reliable Connection, moderately fast too.
- Specialised client, Steaming Audio Protocol (PNM for real audio).

## 4.0 Conclusion

In conclusion, video is simply sequence of digitized picture, they can be ***compressed.*** One way to compress video source is to apply any of the common algorithms such as JPEG algorithm independently to each frame that makes up a video. Normal voice range is about 500 Hz to 2 kHz. Low frequencies are vowels and bass, while high frequencies are consonants.

## 5.0 Summary

In sum, we learnt about the principle of video compression. We equally discovered the common types of audio compression as well as the phenomenon of human hearing and voice. Some examples of streamed audio (and video) are; real audio, shockwave, .wav files. Hope you grasped these key points? You can now attempt the questions below.

## 6.0 Self-Assessment Exercise

1 List the common compression techniques used in audio

2 Give 2 common examples of streamed audio

3 State the lossy method applied to audio compression

## 7.0 References/Further Reading

A.M. Tekalp. (1995). Digital video processing, Prentice Hall PTR.

Boyle. (1998).Design for Multimedia Learning, Prentice Hall, (ISBN 0-13-242155-8)

Brook and Wynne, Hodder and Stoughton (2001).Signal Processing: Principles and Applications.

Encyclopedia of Graphics File Formats, Second Edition by James D. Murray and William vanRyper, O'Reilly & Associates, 1996 (ISBN: 1-56592-161-5)

Fluckiger. (1994). Understanding Networked Multimedia, Prentice Hall.

H. Maurer, Addison Wesley, (1996). Hyperwave: The Next Generation Web Solution, (ISBn 0-201-40346).

Intro. to Computer Pictures, http://ac.dal.ca:80/ dong/image.htm from Allison Zhang at the School of Library and Information Studies, Dalhousie University, Halifax, N.S., Canada

J. G. Shuman, (2002). Multimedia Elements. *Multimedia In Action.* Vikas Publishing House Pvt Ltd.

J. Vince, Addison Wesley, (1995).Virtual Reality Systems, (ISBN 0-201-87687-6)

J.F.K, Buford. (1994). Multimedia Systems, ACM Press, 1994 (ISBN 0-201-53258-1).

James D. Murray and William vanRyper, (1996). Encyclopedia of Graphics File Formats, Second Edition, O'Reilly & Associates.

Lowe and W. Hall, J. (1999). Hypermedia and the Web: An Engineering Approach, 1999 (ISBN 0-471-98312-8).

P.W. Agnew and A.S. Kellerman. (1996). Distributed Multimedia: Technologies, Applications, and Opportunities in the Digital Information Industry (1st Edition) Addison Wesley.

Sloane, McGraw Hill. (2002). Multimedia Communication, (ISBN 0-077092228)

T. Kientzle, Addison Wesley, 1997. A programmer's Guide to Sound, (ISBN 0-201-41972-6)

Vaughan, Tay, 1993, Multimedia: Making It Work (first edition, ISBN 0-07-881869-9), Osborne/McGraw-Hill, Berkeley, pg. 3.

Watkinson, (2004). The Art of Digital Audio, -Heinmann. Synthesizer Basics, GPI Publications.

# Unit 4 Image Histogram and Processing

## 1.0 Introduction

Image processing is a very important aspect of digital signal processing (DSP). In this unit, we shall explore the application of DSP techniques in the enhancement of images, by applying histogram analysis. We shall equally consider image restoration which is a main relevance of image processing.

## 2.0 Objectives

At the end of this unit, you should be able to:

- describe the concept of the histogram of an image
- give an overview of image analysis
- identify a bi-modal image
- state 2 main image enhancement operators
- explain the notion of  image restoration
- Define the term 'noise'.

## 3.0 Main Content

### 3.1 Histogram of an Image

In an image processing context, the ***histogram of an image*** normally refers to a histogram of the pixel intensity values. This ***histogram*** is a graph showing the number of pixels in an image at each different intensity value found in that image. For an 8-bit grayscale image there are 256 different possible intensities, and so the histogram will graphically display 256 numbers showing the distribution of pixels amongst those grayscale values.

Histograms can also be taken of color images --- either individual histogram of red, green and blue channels can be taken, or a 3-D histogram can be produced, with the three axes representing the red, blue and green channels, and brightness at each point representing the pixel count. The exact output from the operation depends upon the implementation --- it may simply be a picture of the required histogram in a suitable image format, or it may be a data file of some sort representing the histogram statistics.

### Self-Assessment Exercise

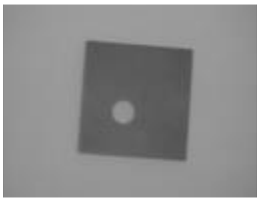
Describe the histogram of an image

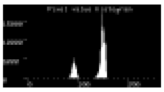
## 3.2 Image Analysis

The process of image analysis is very simple. The image is scanned in a single pass and a running count of the number of pixels found at each intensity value is recorded. This is then used to construct a suitable histogram.

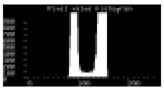
Histograms have many applications. One of the more common is to decide what value of threshold to employ when converting a grayscale image to a binary one through thresholding. If the image is suitable for thresholding then the histogram will be *bi-modal, i.e.* the pixel intensities will be clustered around two well-separated values. A suitable threshold for separating these two groups will be found somewhere in between the two peaks in the histogram. If the distribution is not like this then it is unlikely that a good segmentation can be produced by thresholding.

The intensity histogram for the input is the same, but with the *y*-axis expanded to show more detail. It is clear that a threshold value of around 120 should segment the picture nicely, as can be seen in the histogram of image.

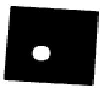


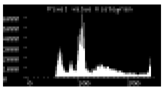


The object being viewed is dark in color and it is placed on a light background, and so the histogram exhibits a good bi-modal distribution. One peak represents the object pixels, one represents the background. The histogram

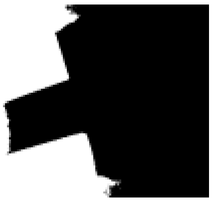


is





This time there is a significant incident illumination gradient across the image, and this blurs out the histogram. The bi-modal distribution has been destroyed and it is no longer possible to select a single global threshold that will neatly segment the object from its background. Two failed thresholding segmentations are shown in and using thresholds of 80 and 120, respectively.



It is often helpful to be able to adjust the scale on the *y*-axis of the histogram manually. If the scaling is simply done automatically, then very large peaks may force a scale that makes smaller features indiscernible.

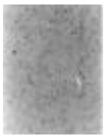
## 3.3 Image Enhancement Operators

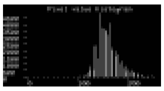
The histogram is used and altered by many image enhancement operators. Two operators which are closely connected to the histogram are ***contrast stretching*** and ***histogram equalization***. They are based on the assumption that an image has to use the full intensity range to display the maximum contrast. Contrast stretching takes an image in which the intensity values don't span the full intensity range and stretches its values linearly. This can be illustrated with

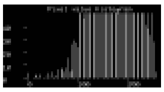
Its histogram,

shows that most of the pixels have rather high intensity values. Contrast stretching the image yields which has a clearly improved contrast. The corresponding histogram is

If we expand the *y*-axis, as was done in we can see that now the pixel values are distributed over the entire intensity range. Due to the discrete character of the pixel values, we can't increase the number of distinct intensity values. That is the reason why the stretched histogram shows the gaps between the single values.

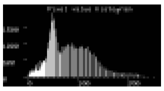






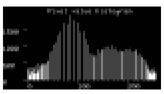
The image

also has low contrast. However, if we look at its histogram,



we see that the entire intensity range is used and we therefore cannot apply contrast stretching. On the other hand, the histogram also shows that most of the pixels values are clustered in a rather small area, whereas the top half of the intensity values is used by only a few pixels. The idea of ***histogram equalization*** is that the pixels should be distributed evenly over the whole intensity range, *i.e.* the aim is to transform the image so that the output image has a *flat* histogram. The image

results from the histogram equalization and



is the corresponding histogram. Due to the discrete character of the intensity values, the histogram is not entirely flat. However, the values are much more evenly distributed than in the original histogram and the contrast in the image was essentially increased.

### 3.4 Image Restoration

Image restoration focuses on the removal or reduction of degradations which happened during the acquisition of an image data. The degradations may include noise, which are errors in the pixel values, or optical effects such as out of focus blurring, or blurring due to camera motion. While neighbourhood operations can be used as a dependable technique for image restoration, other techniques require the use of frequency domain processes.

### 3.2.1 Noise

In image digital signal processing systems, the term noise refers to the degradation in the image signal, caused by external disturbance. If an image is being sent electronically from one place to another, via satellite or through networked cable or other forms of channels we may observe some errors at destination points.

These errors will appear on the image output in different ways depending on the type of disturbance or distortions in the image acquisition and transmission processed. This gives a clue to what type of errors to expect, and hence the type of noise on the image; hence we can choose the most appropriate method for reducing the effects. Cleaning an image corrupted by noise is thus an important aspect of image restoration.

Some of the standard noise forms include:

- Salt and Pepper Noise
- Gaussain Noise and provide some details on the different approaches to eliminating or reducing their effects on the image.

### 3.2.2 Noise Reduction

Now that we have identified the sources of noise in digital signals and some types of noise, we shall describe some of the techniques of reducing or eliminating noise in the image processing. On a general note filters can be used to remove or eliminate noise in an image. The energy of a typical image is primarily in the low frequency region; therefore, a (two-dimensional) low-pass filtering will be good enough in removing a substantial amount of uniform random noise though not without removing some details of the image. On the other hand, the edges that exist in an image usually produce high frequency components. If these components are removed or reduced in energy, the edges will become fuzzier. Median filter are ideal in removing impulse noise while preserving the edges.

They are non-linear filters however, and therefore the process cannot be reversed. In median filtering, a window or mask slides along the image. This window defines a local area around the pixel being processed. The median intensity value of the pixel within that window becomes the new intensity value of the pixel being processed.

## 4.0 Conclusion

To wrap up, recall the following: In an image processing context, the ***histogram of an image*** normally refers to a histogram of the pixel intensity values. Two operators which are closely connected to the histogram are ***contrast stretching*** and ***histogram equalization***.

***Image restoration*** focuses on the removal or reduction of degradations which happened during the acquisition of an image data. Remember that with more practise, you will acquire skills for advanced Multimedia Technology. All the best!

## 5.0 Summary

In this unit we discovered that image processing is a very important aspect of digital signal processing (DSP). We also explored the application of DSP techniques in the enhancement of images, by applying histogram analysis as well as image restoration. Let us now attempt the questions below.

1 Give an overview of image analysis

2 When is an image said to be bi-modal?

3 State 2 main image enhancement operators

4 Define the term 'noise'

## 7.0 References/Further Reading

A.M. Tekalp. (1995). Digital video processing, Prentice Hall PTR.

Boyle. (1998).Design for Multimedia Learning, Prentice Hall, (ISBN 0-13-242155-8)

Brook and Wynne, Hodder and Stoughton (2001).Signal Processing: Principles and Applications.

Encyclopedia of Graphics File Formats, Second Edition by James D. Murray and William vanRyper, O'Reilly & Associates, 1996 (ISBN: 1-56592-161-5)

Fluckiger. (1994). Understanding Networked Multimedia, Prentice Hall.

H. Maurer, Addison Wesley, (1996). Hyperwave: The Next Generation Web Solution, (ISBn 0-201-40346).

Intro. to Computer Pictures, http://ac.dal.ca:80/ dong/image.htm from Allison Zhang at the School of Library and Information Studies, Dalhousie University, Halifax, N.S., Canada

J. G. Shuman, (2002). Multimedia Elements. *Multimedia In Action.* Vikas Publishing House Pvt Ltd.

J. Vince, Addison Wesley, (1995).Virtual Reality Systems, (ISBN 0-201-87687-6)

J.F.K, Buford. (1994). Multimedia Systems, ACM Press, 1994 (ISBN 0-201-53258-1).

James D. Murray and William vanRyper, (1996). Encyclopedia of Graphics File Formats, Second Edition, O'Reilly & Associates.

Lowe and W. Hall, J. (1999). Hypermedia and the Web: An Engineering Approach, 1999 (ISBN 0-471-98312-8).

P.W. Agnew and A.S. Kellerman. (1996). Distributed Multimedia: Technologies, Applications, and Opportunities in the Digital Information Industry (1st Edition) Addison Wesley.

Sloane, McGraw Hill. (2002). Multimedia Communication, (ISBN 0-077092228)

T. Kientzle, Addison Wesley, 1997. A programmer's Guide to Sound, (ISBN 0-201-41972-6)

Vaughan, Tay, 1993, Multimedia: Making It Work (first edition, ISBN 0-07-881869-9), Osborne/McGraw-Hill, Berkeley, pg. 3.

Watkinson, (2004). The Art of Digital Audio, -Heinmann. Synthesizer Basics, GPI Publications.