



NATIONAL OPEN UNIVERSITY OF NIGERIA

SCHOOL OF SCIENCE AND TECHNOLOGY

COURSE CODE: CIT 891

COURSE TITLE: ADVANCED MULTIMEDIA TECHNOLOGIES



CIT 891: ADVANCED MULTIMEDIA TECHNOLOGIES

Multimedia Integrated Services: Voice, Data, Video, Facsimile, Graphics and their characterization, use of various modeling techniques, detection techniques for multimedia signals, multimedia representation for service requirements. Principles of multimedia compression: Overview of current techniques in image/video compression, image/video compression standards JPEG, MPEG and H.263. Image Processing and Computer Sampling, and Filtering, 2D data transform with DTFT, DFT, DCT, KLT, the human visual system and image perception, image enhancement with histogram analysis, linear and morphological operators and image restoration and reconstruction from projection, image analysis, feature detection and recognition, image coding with DCT and wavelet technologies, JPEG and JPEG2000, video coding with motion estimation, H.263 and MPEG

Course Developer

Course Writer Dr. Ikhu-Omoregbe, Nicholas

Course Adapter

National Open University of Nigeria

Lagos.

Course Editor

Course Co-ordinator National Open University of Nigeria

Lagos.



NATIONAL OPEN UNIVERSITY OF NIGERIA

National Open University of Nigeria
Headquarters
14/16 Ahmadu Bello Way
Victoria Island
Lagos

Abuja Annex
245 Samuel Adesujo Ademulegun Street
Central Business District
Opposite Arewa Suites
Abuja

e-mail: centralinfo@nou.edu.ng

URL: www.nou.edu.ng

National Open University of Nigeria 2009

First Printed 2009

ISBN

All Rights Reserved

Printed by ..

For

National Open University of Nigeria

TABLE OF CONTENTS	PAGE
Introduction.....	
What you will learn in this Course.....	
Course Aims.....	
Course Objectives.....	
Working through this Course.....	
Course Materials.....	
Study Units	
Textbooks and References	
Assignment File.....	
Presentation Schedule.....	
Assessment.....	
Tutor Marked Assignments (TMAs)	
Examination and Grading.....	
Course Marking Scheme.....	
Course Overview	
How to Get the Best from This Course	
Tutors and Tutorials	

Introduction

CIT 891 Advanced Multimedia Technologies is a three [3] credit unit course of twelve units. It deals with multimedia integrated services / elements such as voice, graphics, images, video and animation. Multimedia data representation and compression techniques are also covered. In addition image processing with special focus on image enhancement with histogram analysis, image restoration from projection and feature detection were discussed. Finally, you will find this material useful in understanding topics such as image coding with DCT and wavelet technologies, video coding with motion estimation, and JPEG/JPEG2000, H.263 and MPEG standards. The course material is made up of four modules.

Module 1 provides a foundation for the course. In this module we described the components of a multimedia system; explained some desirable features for a multimedia system, and provided other details that will help you understand the remaining parts of the course. Also covered in the module are the various elements of multimedia and their representation within the computer system.

Module 2: in this module we discussed some compression techniques, explained how a compression systems work, stated some advantages and disadvantage of data compression, discussed how signals are transformed from one domain to the other, discussed the different classes of signals, explained the meaning and the different types of transforms and the applications of Fourier transforms in multimedia processing

Module 3: In this module, we described histograms and their applications in image enhancement, we also considered how filters are used for the removal of noise, explained the applications of morphological operations in image processing, explained image degradation model and the principles of object detection

Module 4 has its focus image compression with discrete cosine transform (DCT) and wavelet technologies, video compression with motion estimation and image and video compression standards. In this module MPEG, JPEG and H.363 standards were discussed.

The aim of this course is to equip you with the basic skills in creating, storing, compressing, retrieving, and managing multimedia data as well as providing the required knowledge for you to become a multimedia system expert. By the end of the course, you should be able to confidently work in any multimedia service oriented organization such as entertainment, broadcasting, IT, cable, satellite, mobile and computer organizations.

This Course Guide gives you a brief overview of the course content, course duration, and course materials.

What you will learn in this course

The main purpose of this course is to provide the necessary skills for understanding the creation, storage, compression, transmission and management of multimedia data.

Course Aims

- i. Introduce concepts in the field of multimedia technologies;
- ii. Provide the necessary details on how multimedia data are represented, stored, compressed and used;
- iii. Expose readers to the mathematics of multimedia data transformation and manipulations;
- iv. Acquaint readers with existing standards that will allow interoperable transfer of multimedia data from one system to another.

Course Objectives

Certain objectives have been set out to ensure that the course achieves its aims. Apart from the course objectives, every unit of this course has set objectives. In the course of the study, you will need to confirm, at the end of each unit, if you have met the objectives set at the beginning of each unit. By the end of this course you should be able to:

- understand various concept associated with multimedia technology
- describe the component of multimedia systems
- explain some desirable features for multimedia systems
- explain the basic concepts of multimedia element representation
- explain the principles of digitization
- discuss some compression techniques
- explain how a compression systems work
- state the advantages and disadvantages of data compression
- discuss how signals are transformed from one domain to the other
- discuss the different classes of signals
- explain the meaning and the different types of transforms
- explain the applications of Fourier transforms in multimedia processing
- consider how histograms are used for image enhancement
- consider how filters are used for the removal of noise
- explain the applications of morphological operations in image processing
- explain image degradation model
- explain the principles of object detection
- explain the meaning of motion estimations
- explain the different types of frame
- explain the principles behind MPEG-1 Video Coding
- provide an overview of different image / Video Standards

- explain important features of some common standards used in multimedia applications
- highlight the areas of applications of the standards

Working through this Course

In order to have a thorough understanding of the course units, you will need to read and understand the contents of this course and explore the usage of some multimedia applications.

This course is designed to be covered in approximately sixteen weeks, and it will require your devoted attention. You should do the exercises in the Tutor-Marked Assignments and submit to your tutors.

Course Materials

These include:

1. Course Guide
2. Study Units
3. Recommended Texts
4. A file for your assignments and for records to monitor your progress.

Study Units

There are twelve study units in this course:

Module 1

Unit 1: Multimedia Systems and Requirements

Unit 2: Elements of Multimedia

Unit 3: Multimedia Signal Representation and Processing

Module 2

Unit 1: Overview of Current Techniques in Image/Video Compression

Unit 2: Image Processing and Human Visual System

Unit 3: 2D Data Transform with DTFT, DFT, DCT

Module 3

Unit 1: Image Perception, Image Enhancement with Histogram Analysis

Unit 2: Morphological Operators

Unit 3: Image Restoration, Feature Detection and Pattern Matching

Module 4

Unit 1: Image Coding With DCT and Wavelet Technologies

Unit 2: Video Coding With Motion Estimation

Unit 3: Image / Video Compression Standards JPEG, MPEG and H.263.

Make use of the course materials, do the exercises to enhance your learning.

Textbooks and References

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

D. Jankerson, G. A. Harris and P. D. Johnson, Jr (2003), Introduction to Information Theory and Data Compression, Second Edition, Chapman and Hall / CRC, Florida, USA

A. Gersho and R. Gray (1992), Vector Quantization and Signal Compression, Boston, MA

S. Mitra and Tinkuacharya (2003), Data Mining Multimedia, Soft Computing, and Bioinformatics, John Wiley & Sons, Inc, Hoboken, New Jersey, Canada.

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

B. Furht, Stephen W. Smoliar, H. Zhang (1995), Video and Image processing in multimedia systems, Kluwer Academic Publisher

T. Acharya, P. Sing Tsai (2005), JPEG2000 Standard for Image Compression: Concept, algorithm and VLSI

B. A. Forouzan, S. C. Fegan (2003), Data Communication and Networking, McGraw Hill Higher Education, Singapore

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

D. Strandedby, W. Walker(2004), Digital Signal Processing and Applications, Newnes, An imprint of Elsevier, Jordan Hill, Oxford

H. Benoit(1997), Digital Television, MPEG-1, MPEG-2 and Principles of the DVB system, Focal Press, An imprint of Elsevier, Jordan Hill, Oxford

S. Heath (1996) Multimedia and Communications Technology, An imprint of Butterworth-Heinemann, Jordan Hill, Oxford

- J. D. Gibson (Ed.) (2001), *Multimedia Communications, Directions and Innovation*, Academic Press, San Diego, USA
- F. Halsa (2001), *Multimedia communications, Applications, Networks, Protocols and Standards*, Pearson Education
- A. N. Netravali and B. Haskell (1988), *Digital Pictures*. New York: Plenum Press
- A. N. Netravali and B. Haskell (1988.),*Digital Pictures*. New York: Plenum Press,
- A. K. Jain (1989.) *Fundamentals of Image Processing*. Englewood Cliffs, NJ: Prentice-Hall,
- W. B. Pennebaker and J. L. Mitchell,(1993) *JPEG: Still Image Data Compression Standard*. New York: Chapman & Hall
- R. Hunter and A. H. Robinson,(1980) "International digital facsimile standard," *Proceedings of IEEE*, vol. 68, pp. 854-867.
- D. A. Huffman, " A method for the construction of minimum redundancy codes," *Proceedings of the IRE*, vol. 40, pp. 1098-1101, 1952.
- R. J. Clarke, *Transform Coding of Images* (1985) New York: Academic Press,
- I. T. Jolliffe, *Principal Component Analysis* (1986) New York: Springer-Verlag,
12. D. Hand, H. Mannila and P. Smyth, (, 2001) *Principles of Data Mining*. Cambridge, MA: The MIT Press
- K. R. Rao and P. Yip (1990), *Discrete Cosine Transform - Algorithms, Advantages, Applications*. San Diego, CA: Academic Press
- M. Ghanbari, *Video Coding (1999) : An Introduction to Standard Codecs*, vol. 42 of *Telecommunications Series*. London, United Kingdom: IEEE
- K. R. Castleman(1996) *Digital Image Processing*. Prentice Hall.
- R. Gonzalez and R. E. Woods (2000), *Digital Image Processing*. Addison-Wesley, second
- R. M. Haralick and L. G. Shapiro(1993), *Computer and Robot Vision*. Addison-Wesley
- 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,

M. Rabbani and P. W. Jones(1991). Digital Image Compression Techniques. SPIE Optical Engineering Press

S. Roman(1997), Introduction to Coding and Information Theory. Springer-Verlag, 1997.

A. Rosenfeld and A. C. Kak (1982) Digital Picture Processing. Academic Press, second edition

J. P. Serra. Image analysis and mathematical morphology. Academic Press, 1982.

M. P. Siedband (1998) Medical imaging systems. In John G. Webster, editor, Medical instrumentation: application and design, pages 518_576. John Wiley and Sons, 1998.

M. S.onka, V. Hlavac, and R. Boyle (1999), Image Processing, Analysis and Machine Vision. PWS Publishing, second edition

S. E. Umbaugh (1998) Computer Vision and Image Processing: A Practical Approach Using CVIPTools. Prentice-Hall.

<http://encyclopedia.jrank.org/articles/pages/6922/Video-Coding-Techniques-and-Standards.html>

<http://www.mpeg.org/>

<http://www.jpeg.org/>

S. Image Compression - from DCT to Wavelets : A Review,
<http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html>

http://upload.wikimedia.org/wikipedia/commons/6/6e/Sampled_signal.png

Assignments File

These are of two types: the self-assessment exercises and the Tutor-Marked Assignments. The self-assessment exercises will enable you monitor your performance by yourself, while the Tutor-Marked Assignment is a supervised assignment. The assignments take a certain percentage of your total score in this course. The Tutor-Marked Assignments will be assessed by your tutor within a specified period. The examination at the end of this course will aim at determining the level of mastery of the subject matter. This course includes twelve Tutor-Marked Assignments and each must be done and submitted accordingly. Your best

scores however, will be recorded for you. Be sure to send these assignments to your tutor before the deadline to avoid loss of marks.

Presentation Schedule

The Presentation Schedule included in your course materials gives you the important dates for the completion of tutor marked assignments and attending tutorials. Remember, you are required to submit all your assignments by the due date. You should guard against lagging behind in your work.

Assessment

There are two aspects to the assessment of the course. First are the tutor marked assignments; second, is a written examination.

In tackling the assignments, you are expected to apply information and knowledge acquired during this course. The assignments must be submitted to your tutor for formal assessment in accordance with the deadlines stated in the Assignment File. The work you submit to your tutor for assessment will count for 30% of your total course mark.

At the end of the course, you will need to sit for a final three-hour examination. This will also count for 70% of your total course mark.

Tutor Marked Assignments (TMAS)

There are twelve tutor marked assignments in this course. You need to submit all the assignments. The total marks for the best four (4) assignments will be 30% of your total course mark.

Assignment questions for the units in this course are contained in the Assignment File. You should be able to complete your assignments from the information and materials contained in your set textbooks, reading and study units. However, you may wish to use other references to broaden your viewpoint and provide a deeper understanding of the subject.

When you have completed each assignment, send it together with form to your tutor. Make sure that each assignment reaches your tutor on or before the deadline given. If, however, you cannot complete your work on time, contact your tutor before the assignment is done to discuss the possibility of an extension.

Examination and Grading

The final examination for the course will carry 70% of the total marks available for this course. The examination will cover every aspect of the course, so you are advised to revise all your corrected assignments before the examination.

This course endows you with the status of a teacher and that of a learner. This means that you teach yourself and that you learn, as your learning capabilities would allow. It also means that you are in a better position to determine and to ascertain the what, the how, and the when of your course learning. No teacher imposes any method of learning on you.

The course units are similarly designed with the introduction following the table of contents, then a set of objectives and then the concepts and so on.

The objectives guide you as you go through the units to ascertain your knowledge of the required terms and expressions.

Course Marking Scheme

This table shows how the actual course marking is broken down.

Assessment	Marks
Assignment 1- 4	Four assignments, best three marks of the four count at 30% of course marks
Final Examination	70% of overall course marks
Total	100% of course marks

Table 1: Course Marking Scheme

Course Overview

Unit	Title of Work	Weeks Activity	Assessment (End of Unit)
	Course Guide	Week 1	
	Module1		
1	Multimedia Systems and Requirements	Week 1	Assignment 1
2	Elements of Multimedia	Week 2	Assignment 2
3	Multimedia Signal Representation and Processing	Week 3-4	Assignment 3
	Module2		
1	Overview of Current Techniques in Image/Video Compression	Week 5	Assignment 4
2	Image Processing and Human Visual System	Week 6	Assignment 5
3	2D Data Transform with DTFT, DFT, DCT	Week 7-8	Assignment 6
	Module3		
1	Image Perception, Image Enhancement with Histogram Analysis	Week 9	Assignment 7
2	Morphological Operators	Week 10-11	Assignment 8
3	Image Restoration, Feature Detection and Pattern Matching	Week 12	Assignment 9
	Module4		
1	Image Coding with DCT and Wavelet Technologies	Week 13	Assignment 10
2	Video Coding with Motion Estimation	Week 14	Assignment 11
3	Image / Video Compression Standards JPEG, MPEG and H. 263.	Week 15-16	Assignment 12
	Revision	Week 16	
	Examination	Week 17	
Total		17 weeks	

How to get the best from this course

In distance learning the study units replace the university lecturer. This is one of the great advantages of distance learning; you can read and work through specially designed study materials at your own pace, and at a time and place that suit you best. Think of it as reading the lecture instead of listening to a lecturer. In the same way that a lecturer might set you some reading to do, the study units tell you when to read your set books or other material. Just as a lecturer might give you an in-class exercise, your study units provide exercises for you to do at appropriate points.

Each of the study units follows a common format. The first item is an introduction to the subject matter of the unit and how a particular unit is integrated with the other units and the course as a whole. Next is a set of learning objectives. These objectives enable you know what you should be able to do by the time you have completed the unit. You should use these objectives to guide your study. When you have finished the units you must go back and check whether you have achieved the objectives. If you make a habit of doing this you will significantly improve your chances of passing the course.

Remember that your tutor's job is to assist you. When you need help, do not hesitate to call and ask your tutor to provide it.

1. Read this Course Guide thoroughly.
2. Organize a study schedule. Refer to the Course Overview for more details. Note the time you are expected to spend on each unit and how the assignments relate to the units. Whatever method you chose to use, you should decide on it and write in your own dates for working on each unit.
3. Once you have created your own study schedule, do everything you can to stick to it. The major reason that students fail is that they lag behind in their course work.
4. Turn to Unit 1 and read the introduction and the objectives for the unit.
5. Assemble the study materials. Information about what you need for a unit is given in the Overview at the beginning of each unit. You will almost always need both the study unit you are working on and one of your set of books on your desk at the same time.

6. Work through the unit. The content of the unit itself has been arranged to provide a sequence for you to follow. As you work through the unit you will be instructed to read sections from your set books or other articles. Use the unit to guide your reading.
7. Review the objectives for each study unit to confirm that you have achieved them. If you feel unsure about any of the objectives, review the study material or consult your tutor.
8. When you are confident that you have achieved a unit's objectives, you can then start on the next unit. Proceed unit by unit through the course and try to pace your study so that you keep yourself on schedule.
9. When you have submitted an assignment to your tutor for marking, do not wait for its return before starting on the next unit. Keep to your schedule. When the assignment is returned, pay particular attention to your tutor's comments, both on the tutor-marked assignment form and also written on the assignment. Consult your tutor as soon as possible if you have any questions or problems.
10. After completing the last unit, review the course and prepare yourself for the final examination. Check that you have achieved the unit objectives (listed at the beginning of each unit) and the course objectives (listed in this *Course Guide*).

Tutors and Tutorials

There are 12 hours of tutorials provided in support of this course. You will be notified of the dates, times and location of these tutorials, together with the name and phone number of your tutor, as soon as you are allocated a tutorial group.

Your tutor will mark and comment on your assignments, keep a close watch on your progress and on any difficulties you might encounter and provide assistance to you during the course. You must mail or submit your tutor-marked assignments to your tutor well before the due date (at least two working days are required). They will be marked by your tutor and returned to you as soon as possible.

Do not hesitate to contact your tutor by telephone, or e-mail if you need help. The following might be circumstances in which you would find help necessary. Contact your tutor if:

- you do not understand any part of the study units or the assigned readings,
- you have difficulty with the self-tests or exercises,

- you have a question or problem with an assignment, with your tutor's comments on an assignment or with the grading of an assignment.

You should try your best to attend the tutorials. This is the only chance to have face to face contact with your tutor and to ask questions which are answered instantly. You can raise any problem encountered in the course of your study. To gain the maximum benefit from course tutorials, prepare a question list before attending them. You will learn a lot from participating in discussions actively.

Summary

This course has introduced you to the basic principles and concepts in the domain of multimedia technologies. The skills you need to understand the various elements of multimedia data, their representation within the computer, compression techniques, new and emerging services / applications, etc. are intended to be acquired in this course. The content of the course material was planned and written to ensure that you acquire the proper knowledge and skills for you to become an expert in multimedia technologies. We wish you success with the course and hope that you will find it both interesting and useful.

Course Code CIT 891

Course Title ADVANCED MULTIMEDIA TECHNOLOGI ES

Course Developer

Course Writer Dr. Ikhu-Omoregbe, Nicholas

Course Adapter
National Open University of Nigeria
Lagos.

Course Editor

Course Co-ordinator National Open University of Nigeria
Lagos.



NATIONAL OPEN UNIVERSITY OF NIGERIA

National Open University of Nigeria
Headquarters
14/16 Ahmadu Bello Way
Victoria Island
Lagos

Abuja Annex
245 Samuel Adesujo Ademulegun Street
Central Business District
Opposite Arewa Suites
Abuja

e-mail: centralinfo@nou.edu.ng

URL: www.nou.edu.ng

National Open University of Nigeria 2009

First Printed 2009

ISBN

All Rights Reserved

Printed by ..

For

National Open University of Nigeria

MODULE 1

UNIT 1: MULTIMEDIA SYSTEMS AND REQUIREMENTS

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Overview of Multimedia Technology
 - 3.2 Definition of multimedia
 - 3.3 Multimedia Applications
 - 3.4 Multimedia Systems
 - 3.5 Components of Multimedia Systems
 - 3.5.1 Input Devices / Output Devices
 - 3.5.2 Systems Devices
 - 3.5.3 Storage Devices Hard disks, CD-ROMs, DVD-ROM, etc
 - 3.5.4 Communication devices
 - 3.5.5 Additional Hardware
 - 3.6 Multimedia Workstations
 - 3.7 Desirable Features for a Multimedia Computer
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

In this unit, we shall provide a foundation for this course on multimedia technologies. In achieving this, we attempt to refresh your skills / knowledge of some basic concepts in computer systems and related courses / topics such as signal processing, computer hardware and software, data communications, computer network, multimedia applications etc. Specifically, we shall describe the components of a multimedia system; explain some desirable features for a multimedia system, and provide other details that will help you understand the remaining parts of this course.

2.0 OBJECTIVES

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

- Explain the meaning of multimedia technology
- Describe the component of multimedia systems
- Explain some desirable features for multimedia systems

3.0 MAIN CONTENT

3.1 Overview of Multimedia Technology

Multimedia technology has emerged in the last few years as a major area of research. Multimedia computer systems have opened a wide range of potential applications by

combining a variety of information sources such as voice, graphics, animation, images audio, and full motion video. Multimedia technology refers to both the hardware and software, and techniques used to create and run multimedia systems. Multimedia technology has its background in the merging of three industries; computer, communication, and broadcasting industries. Many applications exist today as a result of the advances in multimedia technology. The mode of delivery for the applications depends on the amount of information that must be stored, the privacy desired, and the potential expertise of the users. Applications that require large amounts of data are usually distributed on Digital Versatile Disk - Read Only Memory (DVD-ROM), while personal presentations might be made directly from a computer using an attached projector. Also some of them can be viewed on the internet from a server or bundled into portable consumer electronics such as Personal digital assistants (PDAs), iPods, iPhones, mp3 players, etc.

3.2 Definition of Multimedia

Multimedia simply means multiple forms of media integrated together. Media can be text, graphics, audio, video, animation, data etc. An example of multimedia is a blog that has text regarding an owner along with an audio file of some of his music and can include selected videos of its owner. Besides multiple types of media being integrated with one another, multimedia can also stand for interactive types of media such as games, Digital Versatile Disk (DVD) or Compact Disk - Read Only Memory (CD-ROM) containing computer-aided learning instructions, animations or movies. Other terms that are sometimes used for multimedia include hypermedia, media, etc.

3.3 Multimedia Applications

We have seen a revolution in computer and communication technologies in the twentieth century. The telecommunications industry has experienced some dramatic innovations that allowed analog to digital networking that enabled today's very powerful internet technology. Transition from the analog to the digital world has offered many opportunities in the way we do things. Telecommunications, the Internet, digital entertainment, and computing in general are becoming part of our daily lives. Today, we are talking about digital networks, digital representation of images, movies and video, TV, voice, digital library all because digital representation of signal is more robust than analog counterpart for processing, manipulations, storage, recovery and transmission over long distances, even across the globe through communication networks. In recent years, there have been significant advancement in processing of still images, video, graphics, speech, and audio signals through digital computers in order to accomplish different applications challenges. As a result, multimedia information comprising image, video, audio, speech, text, and other data types has the potential to become just another data type. Telecommunications is no longer a platform for peer-to-peer voice communications between two people. Demand for communication of multimedia data through the telecommunications network and accessing the multimedia data through Internet is growing explosively. In order to handle this pervasive multimedia data it is essential that the data representation and encoding of multimedia data be standard across different platforms and applications. As more portable consumer electronic devices continue to

emerge, still images and video data comprise a significant portion of the multimedia data and they occupy the lion share of the communication bandwidth for multimedia communications. As a result, development of efficient image compression technique continues to be an important challenge in multimedia technology research.

With the increasing usage of multimedia systems, it is not uncommon to find them exist as standalone / workstations with associated software systems and tools, such as music composition, computer-aided learning, and interactive video or as distributed systems. The combination of multimedia computing with distributed systems in recent times, have offered greater potentials; new applications based on distributed multimedia systems including multimedia information systems, collaboration and conferencing systems, on-demand multimedia services and distance learning are all made possible today.

Generally, multimedia applications use a collection of multiple media sources e.g. text, graphics, images, sound/audio, animation and/or video. Examples of multimedia applications include:

- World Wide Web(WWW)
- Hypermedia courseware
- Video- on-demand
- Interactive TV
- Computer Games
- Virtual reality
- Digital video editing and production systems
- Multimedia Database systems
- Video conferencing and Teleconferencing
- Groupware
- Home shopping
- Games

3.4 Multimedia Systems

The word multimedia in a computer environment implies that many media are under computer control. Thus a multimedia computer is a system capable of processing multimedia data and applications. In its loosest possible sense, a multimedia computer should support more than one of the following media types: text, images, video, audio and animation. However, that means that a computer which manipulates only text and images would qualify as a multimedia computer. A Multimedia System is characterized by its capability to process, store, generate, manipulate and render multimedia information.

3.5 Components of Multimedia Systems

A multimedia systems is not too different from any other type of computer system except for its ability to process multimedia data. Thus, it should have features that can process

audio, video, graphics and animation. Where these data would need to be transmitted, it should have enough memory and support adequate bandwidth / data compression features to minimize delays. Now let us consider the Components (Hardware and Software) required for a multimedia system:

3.5.1 Input Devices / Output Devices

An input device is any piece of hardware used to accept any type of multimedia data for processing by the computer while an output device is any piece of computer hardware equipment used to communicate the results of multimedia processing carried out by a multimedia processing system (such as a computer) to the outside world. In computing, input/output, or I/O, refers to the communication between a multimedia processing system (such as a computer), and the outside world. Inputs are the signals or data sent to the system, and outputs are the signals or data sent by the system to the outside world.

The most common input devices used by the computer are the keyboard and mouse. The keyboard allows the entry of textual information while the mouse allows the selection of a point on the screen by moving a screen cursor to the point and pressing a mouse button. The most common outputs are monitors and speakers. Microphone is another input device that can interpret dictation and also enable us to input sound like the keyboard is used for text. A digital camera records and stores photographic images in digital form that can be fed to a computer as the impressions are recorded or stored in the camera for later loading into a computer. The digital cameras are available for still as well as motion pictures. Other capture devices include video recorder, graphics tablets, 3D input devices, tactile sensors, VR devices, etc.

Output devices exist in different forms. A printer for example, receives the signal from computer and transfers the information to paper. Printers operate in different ways, for example, the Dot-matrix printer strikes the paper a line at a time while inkjet sprays ink. The laser printer uses a laser beam to attract ink. Another example of an output device is the monitor. It is a device for display. It is just like a television set and diagonally from two opposing corners of the picture tube. The standard monitor size is 14 inches. Very large monitors can measure 21 inches diagonal or greater. Another, common device that may be seen with a multimedia system is the amplifier. An amplifier is an electronic device that increases the power of a signal. Amplifiers are used in audio equipments. They are also called power equipments. Speakers with built-in amplifiers have become an integral part of the computers today and are important for any multimedia project.

3.5.2 Systems Devices

These are the devices that are the essential components for a computer. These include microprocessor, motherboard and memory. Microprocessor is basically the heart of the computer. It is a computer processor on a small microchip. When you turn your computer

on, it is the microprocessor, which performs some operations. The microprocessor gets the first instruction from the Basic Input/Output System (BIOS), which is a part of its memory. BIOS actually load the operating system into random access memory (RAM). The motherboard contains computer components like microprocessor, memory, basic input/output system (BIOS), expansion slots and interconnecting circuitry. You can enhance the performance of your computer system by additional components to a motherboard through its expansion slot. RAM also called primary memory, locates the operating system, application programs, and data in current use so that the computer's processor reaches them quickly. RAM is called "random access" because any storage location can be accessed randomly or directly. RAM is much faster than the hard disk, the floppy disk, the CD-ROM and any other secondary storage device. But might get slow when used to its limit. That is the reason you need more memory to support multimedia applications.

3.5.3 Storage Devices Hard disks, CD-ROMs, DVD-ROM, etc

Storage device provides access to large amounts of data on an electro magnetically charged surface. Most personal computers typically come with a hard disk that contains several billion bytes (gigabytes) of storage. The popular ones are 40 GB and above. Hard disk contains a part called which is responsible for improving the time it takes to read or write to a hard disk. The disk cache holds data that has recently been read. The other type of hardware cache inside your computer is cache memory. Cache is the memory that temporarily stores something temporarily. For example, Temporary Internet files are saved in Cache. On the other hand Compact Disc, read-only memory can store computer data in the form of text, graphics and sound. To record data into a CD, you need a writer. Normally this type of CD is either (CD-R) or (CD-RW). For the latter you can use the CD as a floppy disk write, erase and again write data into the same disk. In the end, the data recording is completed, it becomes a CD-ROM and nothing can be deleted. Photo CD: is a standard for storing high-resolution photographic images which can either be a pre-mastered disc or a CD-WO disc. In the latter the images can be added to it.

A CD-ROM is a read-only, digital medium, whose mastering is expensive, but whose replication is relatively cheap. Its current capacity is over 700 MB, its access time is less than 400 m/sec, and its transfer rate is 300 Kbs. A newer technology, the digital versatile disc (DVD), stores much more in the same space and is used for playing back DVDs. DVD was originally said to stand for digital video disc, and later for digital versatile disc. DVD is an optical disc technology with a 4.7 gigabyte storage capacity on a single-sided, one-layered disk, which is enough for a 133-minute movie. DVDs can be single- or double-sided, and can have two layers on each side; a double-sided, two-layered DVD will hold up to 17 gigabytes of video, audio, or other information. This compares to 650 megabytes (.65 gigabyte) of storage for a CD-ROM disk. DVD uses the MPEG-2 file and compression standard. MPEG-2 images have four times the resolution of MPEG-1 images and can be delivered at 60 interlaced fields per second where two fields constitute one image frame. (MPEG-1 can deliver 30 noninterlaced frames per second.) Audio quality on DVD is comparable to that of current audio compact discs.

3.5.4 Communication devices

A modem (modulator-demodulator) modulates digital signals going out from a computer or other digital devices to analog signals for a telephone line and demodulates the analog signal to convert it to a digital signal to be inputted in a computer. Some personal computers come with 56 Kilobits per seconds modems. Modems help your computer to connect to a network, communication networks such as local network, Intranets, Internet, Multimedia server, or other special high speed networks.

3.5.5 Additional Hardware

Having discussed the basic components that you will find on a standard computer system we shall now proceed to mention some additional devices you should expect to see on a system dedicated for multimedia processing. One of such devices is the video capture device. Video capture is the process of converting an analog video signal such as that produced by a video camera or DVD player to digital form. The resulting digital data are referred to as a digital video stream, or more often, simply video stream.

Video capture analog devices like video camera requires a special video capture card that converts the analog signals into digital form and compresses the data.

Video capture card use various components of the computer to pass frames to the processor and hard disk. Video-capture results will depend on the performance and capacity of all of the components of your system working together. For good quality video, a video-capture card must be able to capture full-screen video at a good rate. For example for a full-motion video, the card must be capable of capturing about 35 frames per second at 720 by 480 pixels for digital video and 640 by 480 for analog video. To determine what settings will produce the best results for your projects, you must be careful in defining these parameters. A video adapter provides extended capability to a computer in terms of video. The better the video adapter, the better is the quality of the picture you see. A high quality video adapter is a must for you while designing your developing multimedia applications.

Another device to mention here is the sound card which is a device that attaches to the motherboard to enable the computer to input, process, and deliver sound. The sound card generates sounds; records sound from analog devices by converting them to digital mode and reproduce sound for a speaker by reconvertng them to analog mode. A standard example of this is Creative Labs Sound Blaster.

3.6 Multimedia Workstations

A multimedia workstation can be defined as a computer system capable of handling a variety of information format; text, voice, graphics, image, audio, and full motion video. Advances in several technologies are making multimedia systems technically and economically feasible. These technologies include powerful workstations, high capacity storage devices, high-speed networks, advances in image and video processing (such as animation, graphics, still and full-motion video compression algorithm), advances in audio processing such as music synthesis, compression and sound effects and speech processing (speaker recognition, text-to-speech conversion and compression algorithms), asynchronous and ATM networks.

The main subsystem that could differentiate a multimedia workstation from traditional (non-multimedia) workstation include CD-ROM device, video and audio subsystem and multimedia related hardware (including image, audio and video capture storage and output equipment).

Video Subsystem

A video subsystem for multimedia system is usually composed of a video codec which provides compression and decompression of images and video data. This also performs video capture of TV-type signal (NTSC, PAL, SECAM-these are defined latter in this unit) from camera, VCR and laserdisc, as well as playback of full-motion video. The playback part of the system should include logic to decode the compressed video stream and place the result in the display buffer depending on the functionalities supported. Where a video subsystem exists for a PC-based multimedia system, it can be connected to a central system to receive data in real-time and allows for the scheduling.

An advanced video subsystem may include additional components for image and video processing. For example, the system can contain output connection for attachment to a monitor, which will allow the user to view live images during the capturing process.

An additional function of the video subsystem may be the mixing of real-time video images with video graphics array (VGA) computers graphics origination from the computer system.

Audio Subsystem

An audio subsystem provides recording, music synthesis and playback of audio data. Audio data is typically presented in one of three forms:

- Analog waveform
- digital waveforms
- Musical Instrument Digital Interface (MIDI)

Analog waveform: audio is represented by an analog electrical signal whose amplitude specifies the loudness of the sound. This form is used in microphone, cassette tapes, records, audio amplifiers and speakers.

Digital waveform is represented using digital data. The digital audio has considerable advantage over analog audio, such as less indifference to noise and distortion. However, it involves larger processing and storage capacities. Digital devices which use digital waveforms audio format are compact disc, the digital audio tape (DAT) and the digital compact disc (DCD).

MIDI (Musical Instrument Digital Interface) refers to digital encoding of musical information where the sound data is not stored, and only the commands that describe how the music should be played are generated. MIDI gives the highest data compression, is easy for editing, and is widely accepted as a musical data standard. However, it requires additional hardware (music synthesizer) to generate music.

An audio system could capture and perform digitization of external audio signal through an Analog/Digital converter and generation of audio signal through a Digital/Analog converter. The digital signal processor (DSP) performs data compression and some additional processing functions, such as mixing and volume controls. Some advanced multimedia systems combine both video and audio subsystems into a unified audio / video subsystem.

Self Assessment Test/Practice

- 1) List Five (5) Multimedia applications
- 2) Describe the three main forms of representing audio data

Multimedia Related Hardware

Multimedia related hardware includes video and audio equipment required at multimedia production and or presentation stages; these equipment (some already mentioned in the previous section) can be divided into:

- Image and Video capture equipment; still and video camera, scanner and video recorders
- Image and video storage equipment: laserdisc, videotapes and optical disks
- Image and video output equipment: displays, interactive display, TV projectors and printers
- Audio equipment: microphones, audio tape, recorders, video tapes recorder, audio mixers, headphones and speakers

There are several home TV distribution standards. These include PAL, SECAM, NTSC

There are three main analog color coding systems: Phase Alternation Line (PAL), Sequential Couleur Avec Memoire (SECAM) and National Television System Committee (NTSC). They differ mainly in the way they calculate the luminance/chrominance components from the Red Green Blue (RGB) components.

PAL - It is a European standard which uses a TV scan rate of 25 frames (50 half-frames) per seconds and a frequency of 625 lines / frames. Other countries where it is used are Australia and South America.

SECAM -It is French standard similar to PAL, but it uses different internal / video and io frequencies. Besides France, SECAM is also used in Eastern Europe.

NTSC -It is the USA standard which is very different from PAL and SECAM standards. The frame rate in NTSC is 30 frames (60 half-frame per seconds and the frequency is 525 lines per frame. Other countries where this standard is used are Canada, Japan and Korea.

3.7 Desirable Features for a Multimedia Computer

By definition, a multimedia computer processes at least one media type that is either discrete or continuous in nature. Text and images are example of discrete media (i.e., they are time-independent), whereas video and audio are time-dependent, and consequently, continuous. The processing of time-independent media is meant to happen as fast as possible, but this processing is not time critical because the validity of the data does not depend on any time condition. However, in the case of time-dependent media, their values change over time - and, in fact, processing values in the wrong sequence can invalidate (part of) the data. In addition, multimedia systems are presently being faced with some of the following challenges:

- How to render different data at the same time continuously;
- Sequencing within the media;
- Data retrieval from distributed networks;
- How to strictly maintain the temporal relationships on playback ;
- Retrieving and playing frames in correct order / time frame in video;
- Synchronization ; inter-media scheduling-E. g. Video and Audio
- How to represent and store temporal information. ;
- Multimedia data compression, etc

Given the above challenges the following features are desirable (if not a prerequisite) for a Multimedia System:

- Very high processing power. This is needed to manage large amount of data and real-time delivery of media

- **Efficient and High Input / Output devices** interfaced with the file systems. This is needed to allow for real-time recording as well as play back of data to users.
- **Special Operating System** to allow access to file system, quick and efficient processing of data. It is needed to support direct transfers to disk, real-time scheduling, fast interrupt, processing, input/output streaming, etc.
- **Storage and Memory** large storage units (of the order of hundreds of Gigabytes if not more) and large memory (several gigabytes or more). Large Caches should be provided and high speed buses for efficient management.
- **Network Support** Required for inter-process controls and Client-server communications in distributed systems
- **Software Tools** user-friendly tools needed to handle media, design and develop multimedia applications.
- **Bandwidth:** This is probably the most critical area for multimedia system, without sufficient bandwidth, multimedia applications are simply not practical. The challenge is not simply in providing X megabits per second of bandwidth. The bandwidth has to have the right characteristics to ensure the desired quality of service at all times.

4.0 CONCLUSION

As technology advances, so we expect new multimedia applications / products to be in the market. No doubt, a good number of new media technologies are being used to create complete multimedia experience. For instance, virtual reality integrates the sense of touch with video and audio media to immerse an individual into a virtual world. Other technologies being developed include the sense of smell that can be transmitted via the Internet from one individual to another. Lots of multimedia entertainment software is available on the internet while others are bundled into portable consumer electronics. No doubt, as computers increase their power, new ways of integrating media will make the multimedia experience extremely exciting

5.0 SUMMARY

In this unit we have covered multimedia systems and technology by learning basic definitions. In addition, the features of the hardware, software and network devices required for a multimedia system to function were discussed. Furthermore the desirable features for a multimedia computer system and some multimedia applications were covered. In the next unit, you shall learn about the elements of multimedia.

6.0 TUTOR MARKED ASSIGNMENTS

- 1a.) What is the meaning of the term multimedia ?
 - b.) What do you consider the main requirements for multimedia systems?
- 2a) List three international television standard and state their features
 - b.) List some additional devices that you expect to find in a multimedia computer system

7.0 REFERENCES / FURTHER READINGS

B. Furht, Stephen W. Smoliar, H. Zhang (1995), Video and Image processing in multimedia systems, Kluwer Academic Publisher

T. Acharya, P. -Sing Tsai (2005), JPEG2000 Standard for Image Compression: Concept, algorithm and VLSI

S. Heath (1996) Multimedia and Communications Technology, An imprint of Butterworth-Heinemann, Jordan Hill, Oxford

B. A. Forouzan, S. C. Fegan (2003), Data Communication and Networking, McGraw Hill Higher Education, Singapore

UNIT 2: ELEMENTS OF MULTIMEDIA

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Text
 - 3.2 Graphic / Images
 - 3.3 Audio / Sound
 - 3.4 Video
 - 3.5 Animation
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

In the unit, you have been introduced to multimedia systems and the basic hardware and software requirements of multimedia systems. You should not forget also that the fundamental characteristics of multimedia systems are that they incorporate and process multimedia such as text, colour, audio / voice, video, and animated graphics. In this unit you will study, the main features of these elements.

2.0 OBJECTIVES

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

- Describe the elements of multimedia
- Describe some digital video formats

3.0 MAIN CONTENT

3.1 Text

There are three types of text that are processed by a multimedia computer and these are: **Unformatted text, formatted text and hypertext.** Texts are captured from the keyboard and hypertext can be followed by a mouse, keyboard, stylus or other devices when being run on a computer.

a) Unformatted text

Unformatted text, also referred to as plain text consists of strings of fixed sized characters from limited character set. An example of a character set that is widely used in computing

is ASCII which stands for American Standard Code for Information Interchange. Normal alphabetic, numeric, punctuation and control characters are represented by ASCII character set and constitute the unformatted text.

b) Formatted text

This is also known as rich text. It enables documents comprising of strings of characters of different size, styles, and shape with tables, images and graphics to be inserted at some points within a document. An example of a formatted text is this course material which is produced by a word processing package. Most word processing packages such as Microsoft Word (MS-WORD) has features that allow a document to be created which consist of characters of different styles and of variable size and shape, each of which can be plain, bold, or italicized. To prepare this course material some features of MS-WORD were used in formatting the texts, preparing tables, inserting graphics, etc, at appropriate positions to make the material more interesting for the reader.

c) Hypertext

This refers to documents that contain unformatted or formatted text as well as links to other parts of the document, or other documents. The user can move to any section of the document accessible by selecting the link. The linked document may be on a single system or physically distributed across different systems. Hypertext that includes multimedia information such as sound, graphics, and video, is sometimes referred to as hypermedia.

3.2 Graphics / Images

Graphics are visual presentations on some surface, such as a canvas, wall, computer paper, or stone to brand, inform, illustrate, or entertain. Examples are photographs, Line Art, graphs, diagrams, drawings, typography, numbers, symbols, geometric designs, maps, engineering drawings, or other images. Graphics often combine text, illustration, and color. Graphics are usually generated by a graphics editor program (e.g. Illustrator) or automatically by a program (e.g. Postscript). Graphics files usually store the primitive assembly and do not take up a very high storage overhead. Graphics are usually editable or revisable (unlike Images). Input devices for capturing graphics include keyboard (for text and cursor control), mouse, trackball or graphics tablet.

Images

Images may be two-dimensional, such as a photograph, screen display, and as well as three-dimensional, such as a statue. They can be captured by scanner, digital camera for processing by a multimedia computer. In a broader sense, an image can be seen as two-dimensional figure such as a map, a graph, a pie chart, or an abstract painting.

In this

wider sense, images can also be rendered manually, such as by drawing, painting, carving, rendered automatically by printing or computer graphics technology, or developed by a combination of methods, especially in a pseudo-photograph. To be more specific a still image is a single static image, as distinguished from a moving image. This

phrase is used in photography, visual media and the computer industry to emphasize that one is not talking about movies.

Images are displayed (and printed) in the form of a two-dimensional matrix of individual picture elements known as pixels or sometimes pels. A computer display screen can be considered as being made up of a two-dimensional matrix of individual picture elements (pixel) each of which can have a range of colours associated with it. For example, VGA (Video graphic array) is a common type of display and consist of 640 horizontal pixels by 480 vertical pixels.

3.3 Audio / Sound

Voice and music, for example are by nature analog, so when we record voice or video have created an analog electric signal. They can be captured into the computer for processing via microphones and then digitized and stored. If we want to store the recording in the computer or send it digitally, we need to change it through a process called sampling. The term sampling means measuring the amplitude of the signal at equal intervals. After the analog signals is sampled, we can store the binary data in the computer or use line coding (or a combination of block coding and line coding) to further change the signal to a digital one so it can be transmitted digitally. Digital signals are prone to noise and distortion. A small change in an analog signal can change the received voice substantially, but it takes a considerably change to convert a 0 to 1 or a 1 to 0. Two popular methods of Analog-to-digital conversion are the Pulse amplitude modulation (PAM) and the Pulse code modulation (PCM). A CD Quality Audio requires 16-bit sampling at 44.1 KHz.

3.4 Video

A still image is a spatial distribution of intensity that is constant with respect to time. Video, on the other hand, is a spatial intensity pattern that changes with time. Another common term for video is image sequence, since video can be represented by a time sequence of still images. Video has traditionally been captured, stored and transmitted in analog form. The term analog video signal refers to a one-dimensional (1-D) electrical signal of time that is obtained by sampling the video intensity pattern in the vertical and temporal coordinates and converting intensity to electrical representation. This sampling process is known as scanning.

A typical example of scanning is the Raster scanning. This begins at the top-left corner and progresses horizontally, with a slight slope vertically, across the image. When it reaches the right-hand edge it snaps back to the left edge (horizontal retrace) to start a new scan line. On reaching the bottom-right corner, a complete frame has been scanned and scanning snaps back to the top-left corner (vertical retrace) to begin a new frame. During retrace, blanking (black) and synchronization pulses are inserted.



Figure 3.4: Raster Scan

The aspect ratio, vertical resolution, frame rate, and refresh rate are important parameters of the video signal. The aspect ratio is the ratio of the width to the height of the frame. The vertical resolution is related to the number of scan lines per frame (including the blanking intervals). The frame rate is the number of frames scanned by second. The effect of smooth motion can be achieved using a frame rate of about 25-30 frames per second. However, at these frame rates the human eye picks up the flicker produced by refreshing the display between frames. To avoid this, the display refresh rate must be above 50 Hz.

3.5 Animation

Video may be generated by computer program rather than a video camera. This type of video content is normally referred to as computer animation or, sometimes, because of the way it is generated, animated graphics. Animation is the rapid display of a sequence of images of 2-D or 3-D artwork or model positions in order to create an illusion of movement. It is an optical illusion of motion due to the phenomenon of persistence of vision, and can be created and demonstrated in a number of ways. The most common method of presenting animation is as a motion picture or video program, although several other forms of presenting animation also exist. The typical frame rate required for animation is 15-19 frames per second.

4.0 CONCLUSION

In this unit, you have been exposed to the fundamental media elements i.e text, images/graphics, audio, and video/animation.

5.0 SUMMARY

Computer systems have capacities to manipulate multimedia elements. In the next unit, you shall be exposed to how these elements are represented within the computer system

6.0 TUTOR MARKED ASSIGNMENT

- 1) Explain the following terms:
 - a) Frame rate
 - b) Pixel
 - c) Animation
 - d) Aspect ratio



7.0 REFERENCES / FURTHER READINGS

F. Halsall (2001), *Multimedia communications, Applications, Networks, Protocols and Standards*, Pearson Education

C. Bohren (2006). *Fundamentals of Atmospheric Radiation: An Introduction with 400 Problems*. Wiley-VCH.

M. E. Al-Mualla, C. N. Canagarajah, D. R. Bull (2002), *Video Coding for Mobile Communications*, Elsevier Science, Academic Press, California, USA



UNIT 3: MULTIMEDIA SIGNAL REPRESENTATION AND PROCESSING

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Principles of Digitization
 - 3.2 Sampling Rate
 - 3.3 Nyquist's Theorem
 - 3.4 Encoders and Decoders for Multimedia Applications
 - 3.5 Colour Representations
 - 3.6 Colour Principles:
 - 3.7 Colour models
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

In the preceding unit we covered the elements of multimedia. Multimedia technology has in the last few years led to the development of wide range of applications by combining a variety of information sources such as text, images, graphics, colors, animation, images, audio, and full motion video. In order to effectively manipulate the varied data elements, digitization process is employed. This entails the conversion from other forms of storage, which are analog, to represent media as a set of binary data which can be manipulated by the machine. The time to access text or image-based data is usually short compared to that of audio and video data. Audio and video signals vary continuously with time as the amplitude of the speech, audio and video varies. This type of signal is called analog signal. The duration of applications involving audio / video can be relatively long. If an application requires a single type of media say images, the basic form of representing the media elements (i.e. pixel) is used. However, in application that involves all media elements (text, picture/images, audio, video/animation) integrated together in some way; the four media type would be represented in digital form for ease of processing by the computer system.

2.0 OBJECTIVES

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

- Explain the basic concepts of multimedia element representation
- Explain the principles of digitization
- Describe the techniques of analog - digital conversion
- Explain the concept of color representations



3.1 Principles of Digitization

Both sounds and images can be considered as signals, in one or two dimensions, respectively. Sound can be described as a fluctuation of the acoustic pressure in time, while images are spatial distributions of values of luminance or color. Images can be described in their RGB (Red Green Blue) or HSB (Hue Saturation Value) components as will be discussed later. Any signal, in order to be processed by numerical computing devices, have to be reduced to a sequence of discrete samples, and each sample must be represented using a finite number of bits. The first operation is called sampling, and the second operation is called quantization of the domain of real numbers.

The general properties relating to any time-varying analog signal are represented in Figure 3.1. As shown in part (a) of the figure, the amplitude of such signal varies continuously with time. The highest and lowest frequency components of the signal shown in Fig 3.1 (a) may be those shown in Figure 3.1 (b). Our assumption is based on a mathematical technique known as Fourier analysis. As we can use this theory to show that any time-varying analog signal is made up of a possibly infinite number of single-frequency sinusoidal signals whose amplitude and phase vary continuously with time relative to each other. You shall learn more about this theory in Module Two of this course.

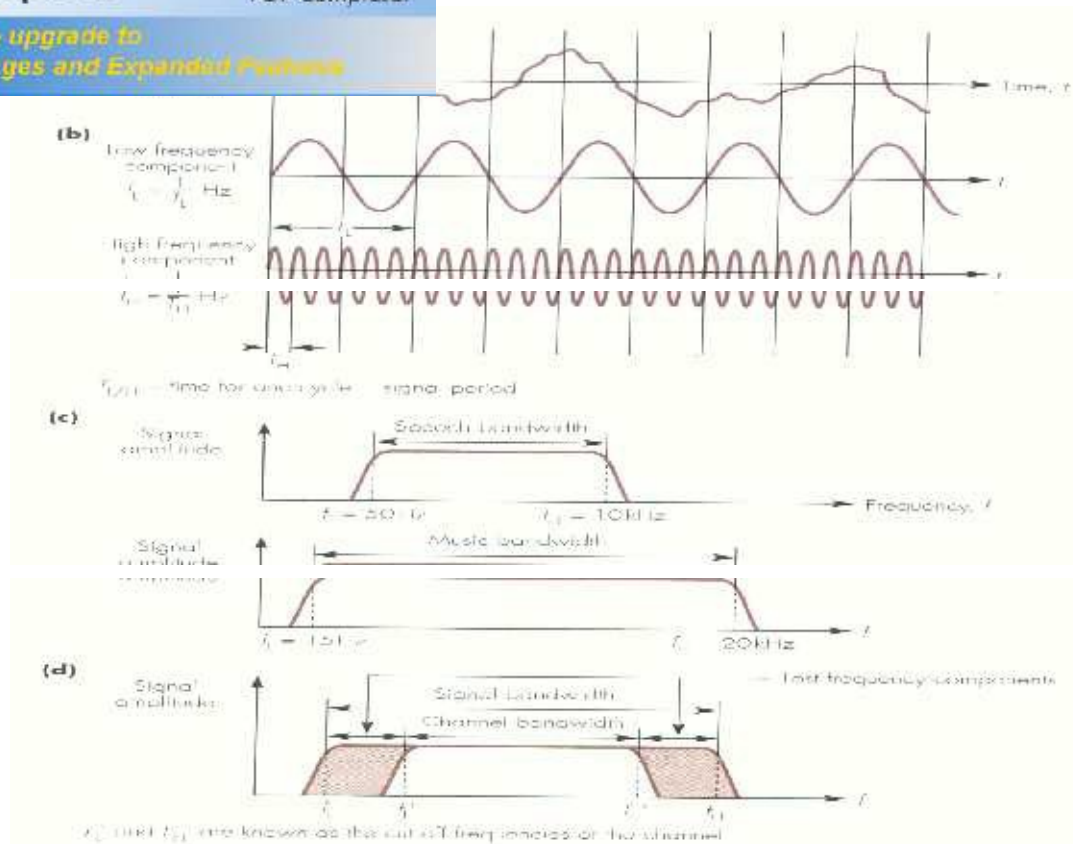


Figure 3.1 (a) (d): Source [F. Halsall, 2001]

The range of frequencies of the sinusoidal components that make up a signal is called the **signal bandwidth**. We consider two examples as shown in Figure 3.1 (c). Here we are considering audio signals. Our first example is a speech signal and the second signal produced by say an orchestral.

Recall that in unit one, we stated that the microphone can be used as an input device in a multimedia system. When human speech is captured by a microphone, what it does is to convert it into electrical signal that are made up of sinusoidal signal varying in frequency say between 50 Hz and 10kHz. In the case of music signal, however, the range of signal is wider and varies between 15 Hz and 20 kHz, this being comparable with the limits of the sensitivity of the ear.

From Data communication background, we should recall that when an analog signal is being transmitted through a network the bandwidth of the transmission channel that is the range of frequencies the channel will pass should be equal to or greater than the bandwidth of the signal. If the bandwidth of the channel is less than this, then some of the low and/ or high frequency components will be lost thereby degrading the quality of the received signal. This type of transmission channel is called **bandlimiting channel**.

3.2 Sampling Rate

The sampling rate or sample rate is a term that defines the number of samples per second (or per other unit) taken from a continuous signal to make a discrete signal. For time domain signals, it can be measured in samples per second (S/s), or hertz (Hz).

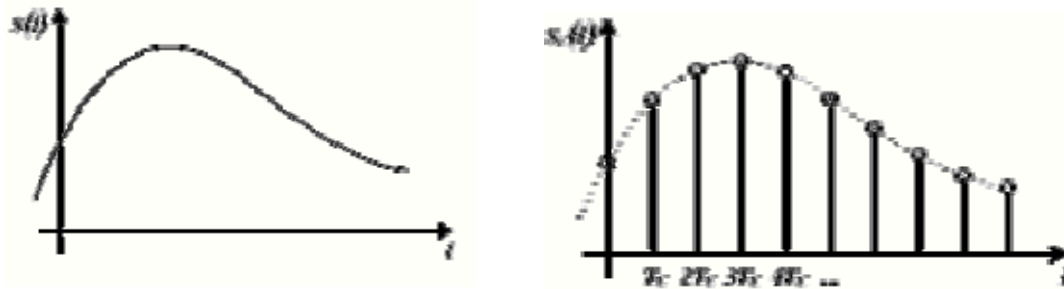


Figure 3.2 a: Analog signal Figure 3.2b: Resulting Sampled signal

In relation to the sampling rate, the Nyquist sampling theorem states that: in order to obtain an accurate representation of a time-varying analog signal, its amplitude must be sampled at a minimum rate that is equal to or greater than twice the highest frequency component that is present in the signal.

This is known as the Nyquist rate and is normally represented as either Hz or more correctly, samples per second (sps). Sampling a signal at a rate which is lower than Nyquist rate results in additional frequency components being generated that are not present in the original signal which in turn, cause the original signal to become distorted.

The distortion caused by sampling a signal at a rate lower than the Nyquist rate is illustrated by considering the effect of undersampling a single frequency sinusoidal signal as shown in Figure 3.3.

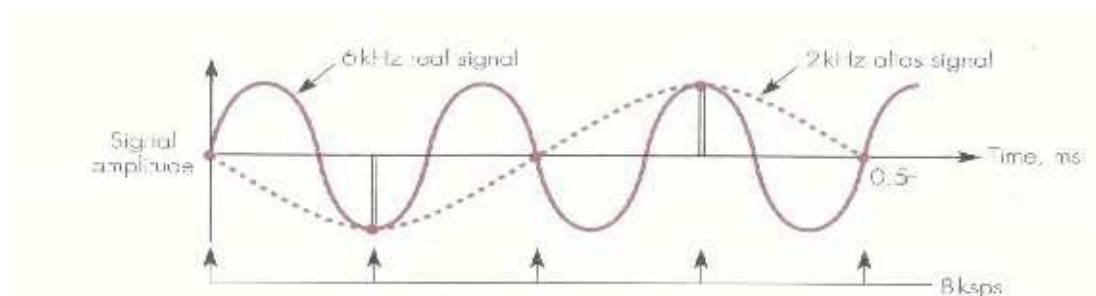


Figure 3.3: Alias Signal Generation due to undersampling Source [F. Halsall, 2001]



In the example, the original signal is assumed to be a 6kHz sinewave which is sampled at a rate of 8 kilo samples per second. Clearly, this is lower than the Nyquist rate of 12ksps (2×6 kHz) and, as we can see, results in a lower-frequency 2 kHz signal being created in place of the original 6 kHz signal. Because of this, such signals are called alias signals. They replace the corresponding original signals.

In general, this means that all frequency components present in the original signal that are higher in frequency than half the sampling frequency being used (in Hz), will generate related low-frequency alias signals which will add to those making up the original source thereby causing it to become distorted. However, by first passing the source signal through a bandlimiting filter which is designed to pass only those frequency components up to that determined by the Nyquist rate, any higher-frequency components in the signal which are higher than this are removed before the signal is sampled. Due to this function the bandlimiting filter is also known as an antialiasing filter.

3.3 Nyquist's Theorem

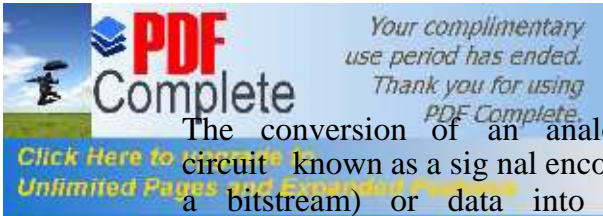
From the preceding discussions, the sample rate must be chosen carefully when considering the maximum frequency of the analog signal being converted. Nyquist's theorem states that the minimum sampling rate frequency should be twice the maximum frequency of the analog signal. A 4kHz analogue signal would need to be sampled at twice that frequency to convert it digitally. For example, a hi-fi audio signal with a frequency range of 20 Hz to 20 kHz would need a minimum sampling rate of 40 kHz. High frequency sampling introduces a frequency component which has to be filtered out using an analog filter.

In the process of digitizing video, if the frequency content of the input analog signal exceeds half the sampling frequency, aliasing artifacts will occur. Thus, a filtering operation is used to bandlimit the input signal and conditions it for the following sampling operation. The amplitude of the filtered analog signal is then sampled at specific time instants to generate a discrete-time signal. The minimum sampling rate is known as the Nyquist rate and is equal to twice the signal bandwidth.

The resulting discrete-time samples have continuous amplitudes. Thus, it would require infinite precision to represent them. The quantization operation is used to map such values onto a finite set of discrete amplitudes that can be represented by a finite number of bits.

3.4 Encoders and Decoders for Multimedia Applications

a. Signal Encoders



The conversion of an analog signal into digital form is carried out using an electrical circuit known as a signal encoder. An encoder is a device used to change a signal (such as a bitstream) or data into a code. Encoders serve any of a number of purposes such as compressing information for transmission or storage, encrypting or adding redundancies to the input code, or translating from one code to another. This is usually done by means of a programmed algorithm, especially if any part is digital, while most analog encoding is done with analog circuitry.

b. Signal Decoder

Similarly, the conversion of the stored digitized sample relating to a particular media type into their corresponding time-varying analog form is performed by an electrical circuit known as signal decoder. A decoder is a device which does the reverse of an encoder, decoding the encoding so that the original information can be retrieved. The same method used to encode is usually just reversed in order to decode. In digital electronics, a decoder can take the form of a multiple-input, multiple-output logic circuit that converts coded inputs into coded outputs, where the input and output codes are different.

3.4.1 Encoder Design

The conversion of a time-varying analog signal such as an audio signal into digital form is carried out using an electronic circuit known as signal encoders. The principles of an encoder are shown in figure 3.4.1 and, as we can see in part (a), it consists of two circuits; a bandlimiting filter and an analog-to-digital converter (ADC), the latter comprising a sample-and-hold and a quantizer. A typical waveform set for a signal encoder is shown in part (b) of the figure. The function of the bandlimiting filter is to remove selected higher-frequency components from the source signal (A). The output of the filter (B) is often then fed to the sample-and-hold circuit which, as its name implies is used to sample the amplitude of the filtered signal at regular time intervals (C) and hold the sample amplitude constant sample (D). After which, it is fed to the quantizer which converts each sample amplitude into binary value known as a codeword (E).

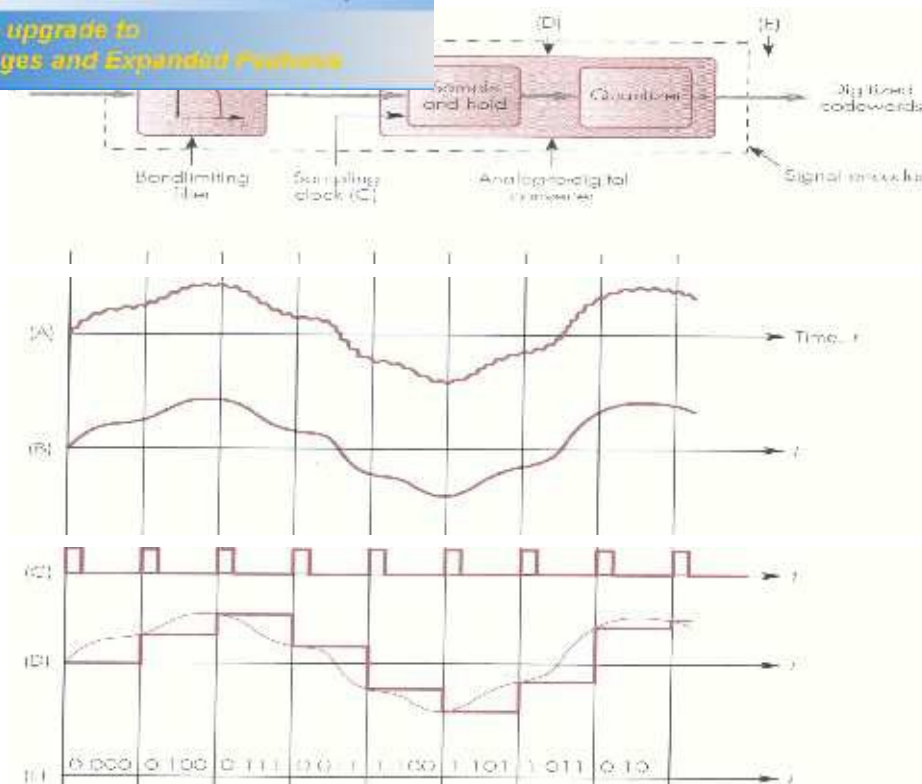


Figure 3.4. 1: Analog to digital signal conversion Source [F. Halsall, 2001]

3.4.2 Decoder Design

Analog signals stored, processed and transmitted in a digital form, normally, prior to their output must be converted back again into their analog form. The loudspeakers, for example, are driven by an analog current signal. The electronic circuit that performs this conversion is known as a (signal) decoder, the principles of which are shown in figure 3.4.2. As depicted in the diagram, first, each digital codeword (A) is converted into an equivalent analog sample using a circuit called a digital-to-analog converter (DAC).

This produces the signal shown in (B), the amplitude of each level being determined by the corresponding codeword.

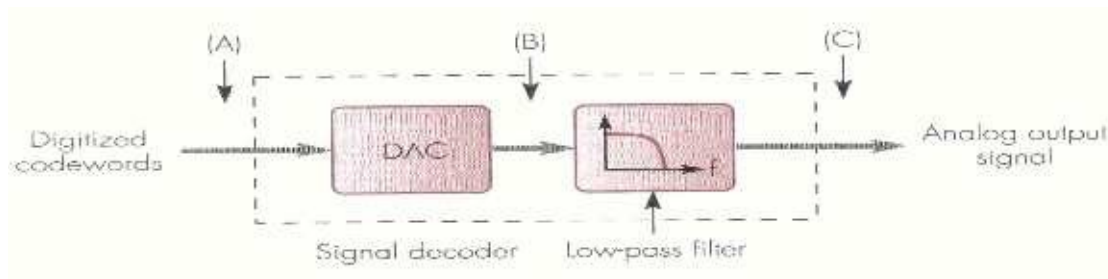


Figure 3.4. 2 Source [F. Halsall, 2001]



In order to reproduce the original signal, the output of the DAC is transmitted through a low-pass filter which, as its name implies, only passes those frequency components that made up the original filtered signal (C). Usually, the high-frequency cut-off of the low-pass filter is made the same as that used in the band limiting filter of the encoder. In this case, the main function of the filter is for reconstruction. Hence, the low-pass filter is known as a recovery or reconstruction filter.

CODEC

Many multimedia applications involve audio and video in which the communication channel is usually two-way simultaneous. Thus any terminal equipment used should support both input and output simultaneously. This often informs the need to have the audio / video signal encoder in each of the terminal equipment to be combined into a single unit called audio / video encoder-decoder or simply an audio/video codec.

3.5 Colour Representations

Colour is the visual perceptual property corresponding in humans to the categories called *red, yellow, blue and others*. Colour derives from the spectrum of distribution of light energy versus wavelength interacting in the eye with the spectral sensitivities of the light receptors. Electromagnetic radiation is characterized by its wavelength (or frequency) and its intensity. When the wavelength is within the visible spectrum (the range of wavelengths humans can perceive, approximately from 380 nm to 740 nm), it is known as "visible light".

Colour	Wavelength interval	Frequency interval
Red	~700-635 nm	~430-480 THz
Orange	~635-590 nm	~480-510 THz
Yellow	~590-560 nm	~510-540 THz
Green	~560-490 nm	~540-610 THz
Blue	~490-450 nm	~610-670 THz
Violet	~450-400 nm	~670-750 THz

Table 3.5: Colours

Colour categories and physical specifications of colour are also associated with objects, materials, light sources, etc., based on their physical properties such as light absorption, reflection, or emission spectra. Colour space can be used as a model to identify colours; for example, a colour can be specified by their unique RGB and HSV values.

3.6 Colour Principles:

It is a known fact, that human eyes see a single colour when a particular set of primary colours are mixed and displayed simultaneously. The truth is that, a whole spectrum of colours known as a colour gamut can be reproduced by using different

proportions of the three primary colours red (R), green (G) and blue (B). This principle is shown in figure 3.6 together with some example of colours that can be produced.

The missing technique used in part (a) is known as additive colour mixing which, since black is produced when all three primary colours are zero, is particularly useful for producing a colour image on a black surface as is the case in display application. It is also possible to perform complementary subtraction colour mixing operation to produce a similar range of colours. This is shown in part (b) of the figure and as we see, with subtractive mixing white is produced when the three chosen primary colours cyan (C), magenta (M), and yellow (Y) are all zero. Hence this choice of colours is particularly producing a colour image on white surface as the case in printing application.

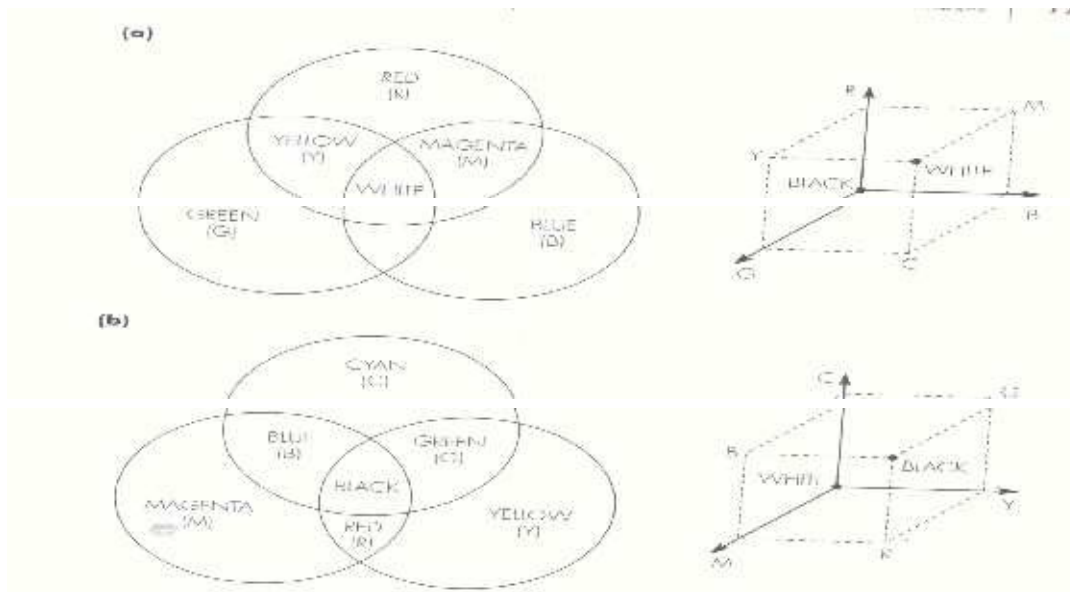


Figure 3.6: Colour derivative principles a) additive colour mixing b) Subtractive colour missing
Source [F. Halsall, 2001]

This is the principles used in the picture tubes associated with colour television set and most computer monitors in the formation of images. The three main properties of a colour source that the eye makes use of are:

- **Brightness:** this represents the amount of energy that stimulates the eye and varies on a gray scale from black (lowest) through to white (highest). It is thus independent of the colour of the source;
- **Hue:** this represents the actual colour of the source, each colour has a different frequency / wavelength and the eye determines the colour from this;



- Saturation: this represents the strength or clarity of the colour, a pastel colour has a lower level of saturation than a colour such as red. Also a saturated colour such as red has no white light in it.

Another term we must define is luminance. This term refers to the brightness of a source.

As we saw in section 3.6 a range of colours can be produced by mixing the three primary colours R, G, and B. In a similar way a range of colours can be produced on a television display screen by varying the magnitude of the three electrical signals that energize the red, green, and blue phosphorous. For example, if the magnitude of the three signals are in the proportion

$$0.299R + 0.587G + 0.114B$$

Then the colour white is produced on the display screen. Hence, since the luminance of a source is only a function of the amount of white light it contains, for any colour source its luminance can be determined by summing together the three primary components that make up the colour in the proportion. That is;

$$Y_S = 0.299R_S + 0.587G_S + 0.114B_S$$

Where Y_S is the amplitude of the luminance signal and R_S , G_S , and B_S are the magnitude of the three colour component signals that make up the source. Thus, since the luminance signal is a measure of the amount of white light it contains, it is the same as the signal by a monochrome television. Two other signals, the blue chrominance (C_b) and the red chrominance (C_r), are then used to represent the colouration hue and saturation of the source. These are obtained from the two colour difference signals:

$$C_b = B_s - Y_s \quad \text{and} \quad C_r = R_s - Y_s$$

Which, since the Y signal has been subtracted in both cases, contains no brightness information. Also, since Y is the function of all three colours, then G can be readily computed from these two signals. In this way, the combination of the three signals Y , C_b and C_r contains all the information that is needed to describe a colour signal while at the same time being compatible with monochrome television which use the luminance signal only.

Chrominance Component s:

In practice, although all colour television systems use this same basic principles to represent the colouration of a source, there are some small differences between the two systems in terms of the magnitude used for the chrominance signals.



3.7 Colour models

A colour model is a method for specifying colours in some standard way. It generally consists of a three dimensional coordinate system and a subspace of that system in which each colour is represented by a single point. We shall investigate three systems.

a) RGB

In this model, each colour is represented as three values R, G, and B, indicating the amounts of red, green and blue which make up the colour. This model is used for displays on computer screens; a monitor has three independent electron guns for the red, green and blue component of each colour. Some colours require negative values of R, G, or B. These colours are not realizable on a computer monitor or TV set, on which only positive values are possible. The colours corresponding to positive values form the RGB gamut; in general a colour gamut consists of all the colours realizable with a particular colour model.

The RGB colour model is an additive colour model in which red, green, and blue light are added together in various ways to reproduce a broad array of colours. The name of the model comes from the initials of the three additive primary colours, red, green, and blue. The main purpose of the RGB colour model is for the sensing, representation, and display of images in electronic systems, such as televisions and computers, though it has also been used in conventional photography. Before the electronic age, the RGB colour model already had a solid theory behind it, based in human perception of colours. RGB is device-dependent colour space: different devices detect or reproduce a given RGB value differently, since the colour elements (such as phosphors or dyes) and their response to the individual R, G, and B levels vary from manufacturer to manufacturer, or even in the same device over time. Thus an RGB value does not define the same colour across devices without some kind of colour management. Typical RGB input devices are colour TV and video cameras, image scanners, and digital cameras. Typical RGB output devices are TV sets of various technologies (CRT, LCD, plasma, etc.), computer and mobile phone displays, video projectors, multicolour LED displays, and large screens as JumboTron, etc. Colour printers, on the other hand, are not RGB devices, but subtractive colour devices.

b. HSV

HSV stands for Hue, Saturation, Value. These terms have the following meanings:

Hue: The true colour attribute (red, green, blue, orange, yellow, and so on).

Saturation: The amount by which the colour has been diluted with white. The more white in the colour, the lower the saturation. So a deep red has high saturation, and a light red (a pinkish colour) has low saturation.

Value: The degree of brightness: a well lit colour has high intensity; a dark colour has low intensity.



This is a more intuitive method of describing colours, and as the intensity is independent of the colour information, this is a very useful model for image processing.

Note that a conversion can be made between RGB and HSV.

c.) YIQ

This colour space is used for TV/video in America and other countries where NTSC the video standard (Australia uses PAL). In this scheme Y is the luminance (this corresponds roughly with intensity), and I and Q carry the colour information. The conversion between RGB is straightforward

$$\begin{aligned}
 Y &= 0.299R + 0.587G + 0.114B \\
 I &= 0.596R - 0.274G - 0.322B \\
 Q &= 0.211R - 0.523G + 0.312B
 \end{aligned}$$

and

$$\begin{aligned}
 R &= 1.000Y + 0.956I + 0.621Q \\
 G &= 1.000Y - 0.272I - 0.647Q \\
 B &= 1.000Y - 1.106I + 1.703Q
 \end{aligned}$$

The two conversion matrices are of course inverse of each other. Observe the difference between Y and V

$$\begin{aligned}
 Y &= 0.299R + 0.587G + 0.114B \\
 V &= \max\{R, G, B\}
 \end{aligned}$$

This reflects the fact that human visual system assigns more intensity to the green component of an image than the red and blue components.

Self Assessment Test

- 1) In your own words, explain the following terms
 - a) Luminance
 - b) Chrominance
- 2) Explain the Nyquist's Theorem

4.0 CONCLUSION.

Multimedia technology has in the last few years lead to the development of wide range or applications by combining a variety of information sources such as text, images, graphics, colors, animation, images audio, and full motion video. In order to effectively manipulate the varied data element, digitization process is employed. This entails the conversion from other forms of storage, which are analog, to represent media as a set of binary data which can be manipulated by the machine. There is one level at which text



has been digital since the advent of computers, although text with print-like features did not become available until much later. Audio came next with the advent of digital studio, then still images began to move to digital formats, and the same process is now being used for video and animated graphics. The possibilities offered by digitization have no doubt allowed a number of new applications and services to be introduced. Examples of such services include; Video on demand, high definition television, videoconferencing, medical imaging, surveillance, flight simulation etc.

5.0 SUMMARY

All types of multimedia information are stored and processed within the computer in digital form. Textual information captured via the keyboard made up of characters is represented by a unique combination of a fixed number of bits known as codeword. Images such as a line and arc are represented by pixel indicating the start and end coordinates of the line relative to the complete image. Audio and video data captured by microphones and digital camera respectively produce electrical signals whose amplitude vary continuously with time known as analogue signals. This is usually converted to digital signal for ease of processing by the computer system and reconverted to analogue signals for display on desired devices / display. We also covered color principles in this unit.

6.0 TUTOR MARKED ASSIGNMENT

- 1) Explain the meaning of the following terms:
 - a.) Codeword
 - b) analog signal
 - c) signal encoder
 - d) signal decoder
- 2) Explain the three popular models for color specification

7.0 REFERENCES / FURTHER READINGS

F. Halsall (2001), Multimedia communications, Applications, Networks, Protocols and Standards, Pearson Education

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

S. Heath (1996) Multimedia and Communications Technology, An imprint of Butterworth-Heinemann, Jordan Hill, Oxford

http://upload.wikimedia.org/wikipedia/commons/6/6e/Sampled_signal.png



MODULE 2

UNIT 1: OVERVIEW OF CURRENT TECHNIQUES IN IMAGE/VIDEO COMPRESSION

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Foundation for Multimedia Compression
 - 3.2 A Data Compression Model
 - 3.3 Principles of Still Image Compression
 - 3.4 Principles of Video Compression
 - 3.5 Classification of Compression Algorithms
 - 3.5.1. Run length encoding
 - 3.5.2 Huffman coding
 - 3.5.3 Predictive Coding
 - 3.5.4 Transform coding
 - 3.5.5 Vector Quantization
 - 3.5.6 Fractal compression
 - 3.5.7 Wavelet coding
 - 3.6 Advantages of Data Compression
 - 3.7 Disadvantages of Data Compression
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assessment
- 7.0 References / Further Readings

1.0 INTRODUCTION

Compression is the process of transforming information from one representation to another, smaller representation from which the original, or a close approximation to it, can be recovered. Compression and decompression processes are often referred to as encoding and decoding. Data compression has important applications in the areas of data storage and data transmission. The data compression process is said to be lossless if the recovered data are certain to be like to the source; otherwise the compression process is said to be lossy. Lossless compression techniques are essential for applications involving textual data. Other applications, such as those involving voice and image data, may be sufficiently flexible to allow controlled degradation in the data.

2. OBJECTIVES

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

- Discuss multimedia data compression
- Discuss some compression techniques



- Explain how a compression systems work
- Provide some advantages and disadvantages of data compression

3.0 MAIN CONTENT

3.1 Foundation for Multimedia Compression

In laying the foundation for multimedia compression, we would first examine two reasons that necessitate multimedia compression.

1. Perceptual Redundancy

2. Demand on Computing Resources

1.) Perceptual Redundancy

In digital representation of multimedia (still image, video, voice, animation) data, significant amount of unnecessary or redundant information are used as far as the human perceptual system is concerned. By human perceptual system, we are referring to our eyes and ears. It is of interest to note that when observing a natural image, the human perceptual system does not easily notice the variation in values of the neighboring pixels in the smooth region of the object and at times sees them as very similar. Likewise, successive frames in a motionless or slowly changing scene in a video are very similar and redundant to the eyes of a viewer. In the same light, of discussion, there are some audio data that are beyond the human audible frequency range and therefore considered to be irrelevant for all practical purposes. These are all indications that not all data should be allowed to go through transmission channels or stored on memory devices at all since they are not perceived by human beings. Thus, there are data in audio-visual signals that cannot be perceived by the human perceptual system. This is what is meant by perceptual redundancy.

Therefore, in the context of data compression, the aim is usually to reduce the redundancies in data representations in order to decrease data storage requirements and hence communications cost. Any efforts to reduce the storage requirements is equivalent to increasing the capacity of the storage medium and hence communication bandwidth. This is one of the reasons why the development of efficient compression will continue to be a design challenge for future communication systems and advanced multimedia applications.

2.) Demand on Computing Resources

Data compression is best achieved by first converting analog signals in which multimedia data may exist to digital signals. Though, there are quite a lot of advantages in converting analog signals to their digital equivalents, the need for large bits for storage, high bandwidth and time for data transmission are issues to be tackled. For example:

- a high-quality audio signal requires approximately 1.5 megabits per second for digital representation and storage.

- a television-quality low-resolution colour video of 30 frames per second with each frame containing 640 x 480 pixels (24 bits per color pixel) would need more than 210 megabits per second of storage.
- a digitized one-hour color movie would require approximately 95 gigabyte of storage.
- The storage requirement for high-definition television (HDTV) of resolution 1280 x 720 at 60 frames per second would certainly be far greater.
- A digitized one-hour colour movie of HDTV-quality video would require about 560 gigabyte of storage
- A small document collection in electronic form in a digital library system may easily require to store several billion characters
- The total amount of information spread over the internet is mind boggling
- etc

Even when analog signals are successfully converted to digital signals, the transmission of these digital signals through limited bandwidth communication channel poses greater challenge. Though, the cost of storage has decreased drastically over the past few decades as a result of advances in microelectronics and storage technology the requirement of data storage and transmission for many multimedia applications has grown so explosively in recent times to outpace this achievement.

The values in table 3.1 shows a comparative demand on resources (disk space, transmission bandwidth, and transmission time needed to store and transmit) some multimedia data. The prefix kilo- denotes a factor of 1000 rather than 1024.

Multimedia Data	Size/Duration	Bits/ Pixel or Bits/Sample	Uncompressed Size (B for bytes)	Transmission Bandwidth	Transmission Time (Using a 28.8K Modem)
A page of	11" x 8.5"	Varying resolution	4-8 KB	32-64 Kb/page	2.1 sec text
Telephone quality speech	10 sec	8 bps	80 KB	64 Kb/sec	22.2 sec
Grayscale Image	512 x 512	8 bpp	262 KB	2.1 Mb/image	73 sec
Color Image	512 x 512	24 bpp	786 KB	6.29 Mb/image	219 Sec
Full-motion Video	640 x 480, 1 min (30 frames/sec)	24 bpp	1.65 GB	221 Mb/sec	5 days 8 Hrs

Table 3.1: Comparative Resource Demand of Multimedia Data

It is obvious from the examples above that adequate computing resource must be available for the storage and transmission of multi-media data. This leaves us with no

other option than to compress multimedia data before storing or transmitting it, and decompressing it at the receiver for playback. For example, with a compression ratio of 64:1, the space, bandwidth, and transmission time requirements can be reduced by a factor of 64, with acceptable quality.

3.2 A Data Compression Model

Figure 3.2 represent a model for typical data compression system. The diagram models a three step process for data compression. These steps are;

- 1.) removal or reduction in data redundancy
- 2) reduction in entropy, and
- (iii) entropy encoding.

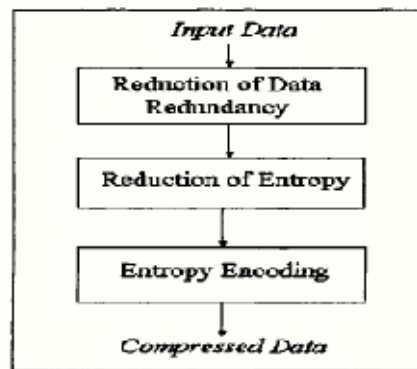


Figure 3.2: A compression Model

The redundancy in data may appear when the neighbouring pixels in a typical image are very much spatially correlated to each other. By correlation we mean that the pixel values are very identical in the non-edge smooth regions of the image. This correlation of the neighbouring pixels is termed as spatial correlation. In case of video or animation the consecutive frames could be almost similar, with or without minor displacement, if the motion is slow. The composition of words or sentences in a natural text follows some context model, based on the grammar being used. Similarly, the records in a typical numeric database may have some sort of relationship amongst the atomic entities which comprise each record in the database. For natural audio or speech data, there are usually rhythms and pauses in regular intervals. All these redundancies in data representation can be reduced in order to realize compression.

Removal or reduction in data redundancy is usually achieved by a transformation process that converts the source data from one form of representation to another, in order to decorrelate the spatial information redundancies present in the data. Amongst the popular techniques used for spatial redundancy reduction is prediction of data samples using some model, transformation of the original data from spatial to frequency domain using methods such as Discrete Cosine Transform (DCT), decomposition of the original dataset into different subbands as in Discrete Wavelet Transformation (DWT), etc. The fact remains that, this spatial redundancy reduction potentially yields more compact



representation of the information in the original dataset, in terms of fewer transformed coefficients or equivalent, and hence makes it amenable to represent the data with less number of bits in order to achieve compression.

The next major stage in a lossy data compression system is "quantization." This technique is applied on the decorrelated data, in order to further reduce the number of symbols or coefficients, by masking irrelevant parts and retaining only the significant details in the data. The outcome of this, is a reduction in entropy of the data, and makes it further open to compression by allocating less number of bits for data transmission or storage. The reduction in entropy is realized by dropping irrelevant details in the transformed data and preserving fewer significant symbols only. Taking a look at the human visual system for example, when an image is transformed in frequency domain, the high-frequency transformed coefficients can be actually be dropped because the human vision system is not sensitive to these. By retaining a smaller number of transformed coefficients in the useful low-frequency range, we can ascertain the fidelity of the reconstructed image. In actual fact, the quality of the reconstructed data is majorly determined by the nature and amount of quantization. The quantized coefficients are then losslessly encoded, using some entropy encoding scheme to compactly represent the quantized data for storage or transmission. Since the entropy of the quantized data is less than that of the source, it can be represented by a fewer number of bits relative to the source data set and hence we realize compression. The decompression system is just an inverse process to reconstruct the data.

In the next two sections we shall describe the principles of compression of still images and video.

3.3 Principles of Still Image Compression

The general model of still image compression framework can be represented by the block diagram depicted in figure 3.1. The statistical analysis of a typical image indicates that a strong correlation usually exist among the neighboring pixels. This leads to redundancy of information in the digital representation of the image. The redundancy can be significantly removed by transforming the image with some sort of preprocessing in order to achieve the desired compression. On a general note, still image compression techniques rely on two fundamental redundancy reduction principles. These are:

- Spatial redundancy
- Statistical redundancy reduction.

By spatial redundancy, we mean the similarity of neighboring pixels in an image. It can be reduced by applying decorrelation or transformation techniques such as predictive coding, transform coding, subband coding, etc.

The statistical redundancy reduction is also known as entropy encoding. The essence is to further reduce, the redundancy in the decorrelated data by using variable-length coding techniques such as Huffman Coding, Arithmetic Coding, etc.

These entropy encoding technique, some of which are discussed in later sections of this study material allocate the bits in the codeword in such a way that more probably appearing symbols are represented with a smaller number of bits compared to the less probably appearing pixels, which helps to achieve the desired compression.

Taking a look at the figure once more, the decorrelation or preprocessing block is the step required to reduce the spatial redundancy of the image pixels due to strong correlation among the neighboring pixels. In lossless coding mode, this decorrelated image is directly processed by the entropy encoder to encode the decorrelated pixel using a variable-length coding technique.

On the other hand, if the lossy compression mode is what you desire, the decorrelated image is subjected to further preprocessing in order to mask or remove irrelevant details depending on the nature of the application of the image and its reconstructed quality requirements. This process of masking is known as quantization process.

Finally the decorrelated and quantized image pixel is subjected through the entropy encoding process to further compact its representation using variable-length codes to produce the compressed image.

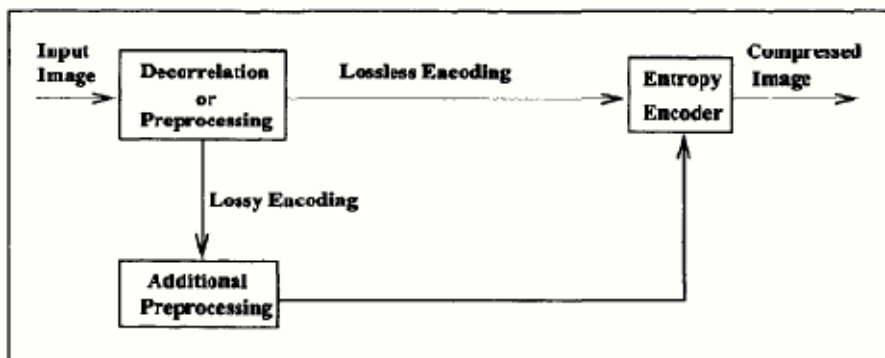


Figure 3.3: Still Image Compression Model

3.4 Principles 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.

To exploit the high correlation between successive frames, we adopt a technique which predicts the content of many of the frames. As we shall describe, this is based on a combination of a preceding and in some instances succeeding frames. Instead of sending the original video as a set of individually compressed frames, just a selection is sent in the form and for the remaining frames, only the difference between the actual frame content and the predicted frame content is sent.

How well any movement between successive frames is estimated will go a long way in determining the accuracy of the prediction operation. The operation is known as motion estimation, and since the estimation process is not exact (just an estimation), therefore, more information must also be sent to indicate any small differences between the predicted and actual positions of the moving segment involved. The latter is known as motion compensation. We shall further discuss this in Module four of the course.

3.5 Classification of Compression Algorithms

Data compression can be accomplished by applying one or more algorithms to multimedia source data. In this section we shall briefly describe the most common ones.

3.5.1. Run length encoding

This is a very simple form of data compression in which runs of data (that is, sequences in which the same data value occurs in many consecutive data elements) are stored as a single data value and count, rather than as the original run. The idea behind Run length encoding (RLE) is to encode strings of zeros and ones by the number of repetitions in each string. RLE has become a standard in facsimile transmission. For a binary image there are many different implementations of RLE; one method is to encode each line separately, starting with the number of 0's. So the following binary image:

```
0 1 1 0 0 0
0 0 1 1 1 0
1 1 1 0 0 1
0 1 1 1 1 0
0 0 0 1 1 1
1 0 0 0 1 1
```

would be encoded as (123) (231)(0321)(141)(33)(132)



Another method is to encode each row as a list of pairs of numbers; the first number is the starting position of a run of 1s and the second number its length. So the above binary image would have the encoding

(22) (33) (1361)(24)(43)(1152)

Furthermore, another implementation would be to encode a sequence or run of consecutive pixels of the same color (such as black or white) as a single codeword.

Example, the sequence of pixels

88 88 88 88 88 88 88 88

could be coded as

8 88 (for eight 88's)

Run length encoding can work well for bi-level images (e.g. black and white text or graphics) and for 8 bit images. Run length encoding does not work well for 24 bit natural images in general. Runs of the same color are not that common. Run-length encoding performs lossless data compression and is well suited to palette-based iconic images. It does not work well at all on continuous-tone images such as photographs, although JPEG uses it quite effectively on the coefficients that remain after transforming and quantizing image blocks. Run-length encoding is used in fax machines (combined with other techniques into Modified Huffman coding. It is relatively efficient because most faxed documents are mostly white space, with occasional interruptions of black.

3.5.2 Huffman coding

Run length coding and Huffman coding are referred to as source coding. From the information theoretic perspective, source coding can mean both lossless and lossy compression. However, researchers often use it to indicate lossless coding only. In the signal processing community, source coding is used to mean source model based coding.

The Huffman coding was invented in 1952 by D. A. Huffman as a technique to produce the shortest possible average code length, given the source symbol set and the associated probability of occurrence of the symbols. The Huffman coding technique is based on the following two observations regarding optimum prefix codes.

- 1.) The more frequently occurring symbols can be allocated with shorter codewords than the less frequently occurring symbols.
- 2.) The two least frequently occurring symbols will have codewords of the same length, and they differ only in the least significant bit.

The mean of the length of these codes is closed to the entropy of the source. For example, if we have m source symbols $\{S_1, S_2, \dots, S_m\}$ with associated probabilities of occurrence $\{P_1, P_2, \dots, P_m\}$. Using these probability values, we can generate a set of Huffman codes of the source symbols. The Huffman codes can be mapped into a binary tree, popularly known as the Huffman tree depicted in figure 3.5.2a

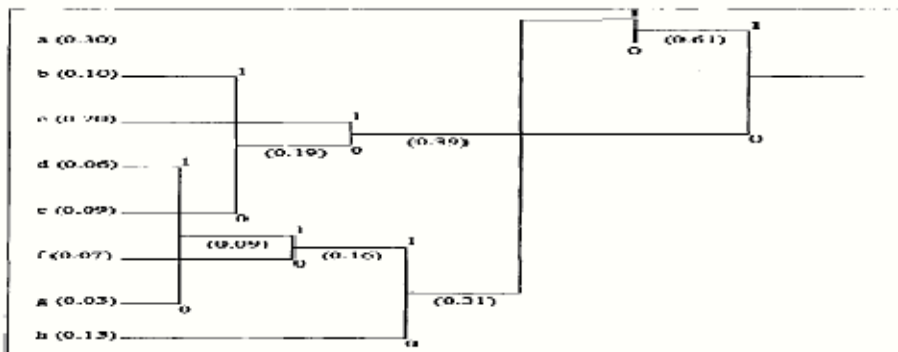


Figure 3.5.2: An example of Huffman Tree form by Applying Huffman algorithm
Source [S. Mitra and Tinkuachar ya, 2003]

3.5.3 Predictive Coding

This coding technique was originally proposed by Cutler in 1952. It should be clear to you from our preceding discussions, that the adjacent pixels in a typical image are highly correlated. This makes it possible to extract a great deal of information about a pixel from its neighboring pixel values. In predictive coding, a pixel value is predicted by a set of previously encoded neighboring pixels. The difference between the pixel and its prediction forms the signal to be coded. It is obvious that, the better the prediction, the smaller the error signal and the more efficient the coding system. The difference between the pixel and its prediction forms the signal to be coded. Invariably, the better the prediction, the smaller the error signal and the more efficient the coding system. We represent this coding scheme with a block diagram in Figure 3.5.3

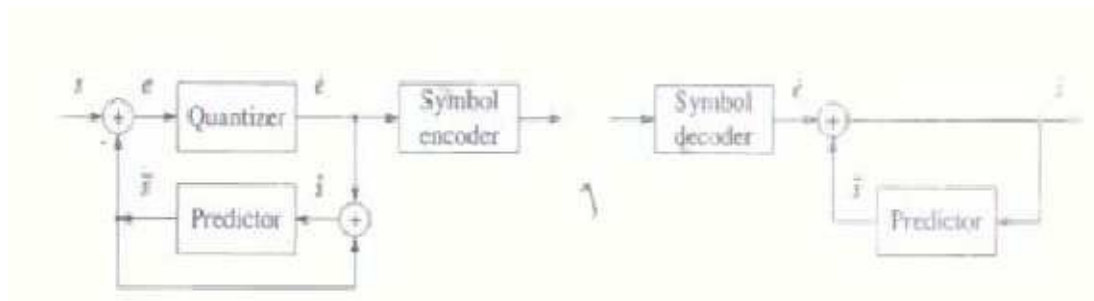
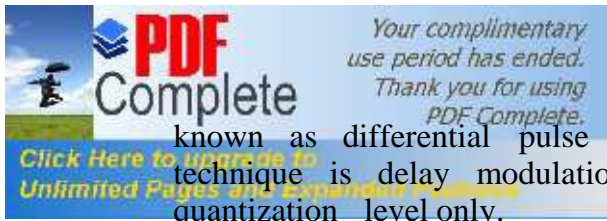


Figure 3.5.3: A predictive coding system

At the decoder, the same prediction is created using previously decoded pixels and the received error signal is added to reconstruct the current pixels. Predictive coding is also



known as differential pulse code modulation (DPCM). A specialized case of this technique is delay modulation (DM), which quantizes the error signal using two quantization level only.

Predictive coding can be realized in different ways depending on the design of the predictor and the quantizer blocks. For example, the predictor can use a linear or nonlinear function of the previously decoded pel, it can be 1-dimensional, that is, using pixels from the same line or 2-dimensional using pixels from the same line and from previous lines, it can be fixed or adaptive. For the quantizer, it can be made to be uniform nonuniform, and it can be fixed or adaptive.

The predicting coding technique requires a small amount of storage and minimum processing to achieve the desired compression on multimedia data. These were partly responsible for the early popularity of this method, when storage and processing devices were scarce and expensive resources. This approach provides only a modest amount of compression. In addition, its performance relies on the statistics of the source data, and it is very sensitive to errors as feedback through the prediction loop can cause error propagation. With the advancement in computing and memory technology that has led to the availability of cheap storage and faster processor, more complex, more efficient methods, like transform coding have become more popular. Despite this, there are instances where predictive coding is still used in video coding.

3.5.4 Transform coding

In predictive coding, the coding process takes place pixel by pixel. Transform coding is an effective way of coding a group of spatially correlated pixels. This technique exploits the fact that the energy of most natural images is largely intense in the low-frequency regions. An appropriate transformation technique produces fewer number of correlated transformed coefficients as compared to the source image, and a large amount of image information is concentrated in these fewer correlated transformed coefficients. As a result, we can get rid of or mask the insignificant transformed coefficients, mainly consisting of the high-frequency components, using a suitable quantization technique without affecting

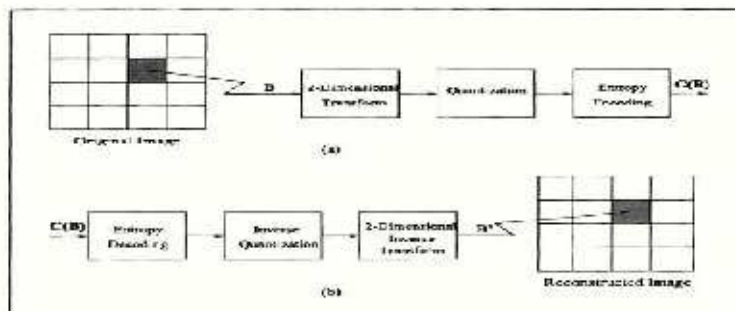
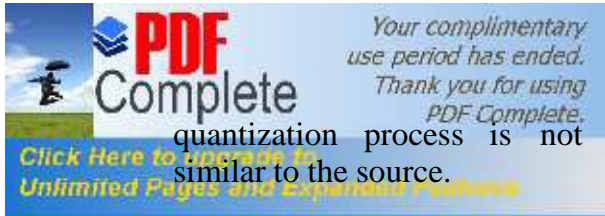


Figure 3.5.4: Transform Coding

reconstructed image quality. This is possible because the human visual system has perceptual masking effects. As a result, the high-frequency components are not as perceptive to reconstruction errors as compared to their low-frequency counterparts. If the

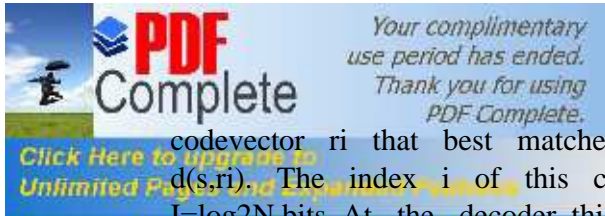


quantization process is not too coarse, then the reconstructed image can be perceptually similar to the source.

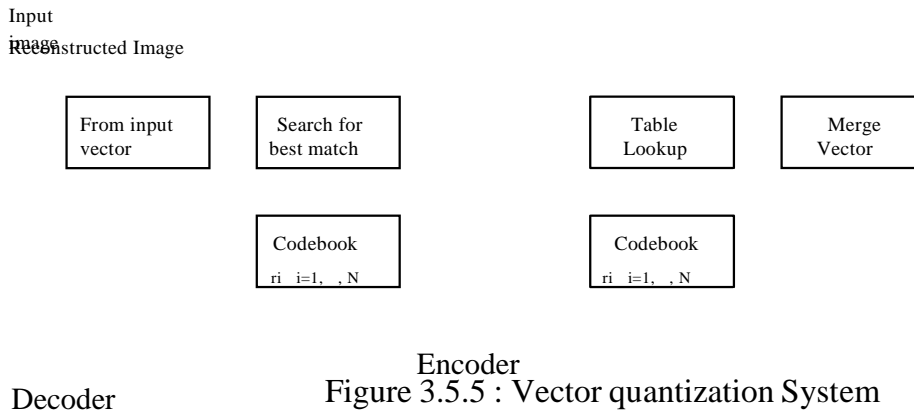
The general framework for transform coding-based image compression systems is captured by Figure. 3.5.4. A close observation shows that the input image is first divided into a number of smaller rectangular blocks B . Each of these blocks is then independently transformed, using a choice linear transformation method. The transformed coefficients are quantized and entropy encoded into bit-stream $c(B)$ in order to achieve compression. In fact, the process of decompression entails that the compressed bit stream $c(B)$ is first entropy-decoded to generate the quantized coefficients. (See figure 3.5.4 above). This is followed by inverse quantization in order to generate an approximation of the transformed coefficients. The inverse transformation is applied on these coefficients to reconstruct the image block B' . The composition of the reconstructed image blocks forms the reconstructed image, as shown in the diagram. The choice of a transformation technique is a major decision in this method. The desire is to transform from the spatial domain to another domain (usually frequency domain) is to represent the data in a more compact form in the transformed domain. Also, using the right transformation technique helps to minimize the mean squared error of the reconstructed image. The Karhunen-Loeve Transform (KLT) is a ready choice since it has been proven to be optimal in terms of the compaction efficiency, by being able to represent images using few principal components containing a significant portion of the image information. However, there are no fast algorithms for practical implementation of KLT, because of its dependency on the input source signal. As a result, other less efficient transform such as the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Sine Transform (DST), and Discrete Hadamard Transform (DHT), etc, are readily used in digital image compression. Amongst these examples, the DCT is the most popular block-based transform, because its performance is very close to that of KLT and a number of algorithms exist for DCT. Discrete cosine transform is a lossy compression algorithm that samples an image at regular intervals, analyzes the frequency components present in the sample, and discards those frequencies which do not affect the image as the human perceives it. Finally, DCT is the basis for most of the image and video compression algorithms, especially the still image compression standard JPEG in lossy mode and the video compression standards MPEG-1, MPEG-2, MPEG-4, H.263, etc.

3.5.5 Vector Quantization

Vector quantization (VQ) is a blocked-based spatial domain method that has become very accepted ever since the early 1980s. The way it works is that the input image data is first decomposed into k -dimensional input vectors. The generation of the input vector can be done in a number of ways; they could be the pel values themselves or to some transformation of them. For example a $k = M \times M$ block of pels can be ordered to form a k -dimensional input vector $s = [s_1, \dots, s_k]^T$. Then the k -dimensional space R^k is decomposed into N region of cells, R_i . Any input vector that falls into the cell R_i then represented by a representative codevector $r_i = [r_{i1}, r_{ik}]^T$. The set of codevectors $= \{r_1, \dots, r_N\}$ is called the codebook. Therefore, the aim of the encoder is to locate the



codevector r_i that best matches the input vector s according to some distortion measure $d(s, r_i)$. The index i of this code vector is then transmitted to the decoder using at most $I = \log_2 N$ bits. At the decoder, this index is used to lookup the codevector from a matching codebook. See figure 3.5.5 below.



Encoder
 Figure 3.5.5 : Vector quantization System
 Source [D. Jankerson et al, 2003]

VQ realizes compression by using a code with relatively few codevectors compared to the number of possible input vector. The resulting bit rate of a VQ is denoted by I/k bits/pel. In theory, as k tends to infinity, the performance of VQ tends towards the rate-distortion bound. Values of K used in typical systems are $k=4 \times 4$ and $N=1024$. When k becomes large, it becomes difficult to store and search the codebook.

A very important problem in VQ is the design of the codebook. The Linde-Buzo-Gray (LBG) algorithm helps to address this. The LBG algorithm computes a codebook with a locally minimum average distortion for a given training set and given codebook size. There are many variants of VQ. It has a performance that rivals that of transform coding discussed earlier in section 3.5.4 of this unit. Although the decoder complexity is insignificant in VQ, the high complexity of the encoder and the high storage requirements of the method limit its use in practice. Like transform coding, VQ suffers from blocking artifacts at very low bit rates.

3.5.6 Fractal compression

Benoit Mandelbrot was the first to use the word fractal to describe a fractured structure which is made up of many similar looking structures and forms. This often occurs in nature in snowflakes crystals trees and river deltas. These kinds of images can be created by taking a simple structure and using it as a building block. By this same principle a basic shape can be reused to create a new image using a set of coefficient and equations to control the processing of the basic image. This process takes the basic image, manipulate it and then lay it onto the image to gradually build up a desired picture.

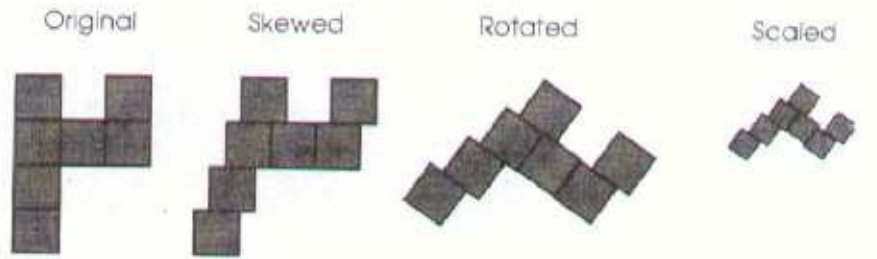


Figure 3.5.6: Fractal transformation [Source S.Heath, 1996]

Using figure 3.5.6 as our example, the manipulations that can be performed are skewing, scaling and translation. These are used to create new derivatives of the original image which can be used to build the final image. This gradual build up is referred to as collaging and the image manipulation are called affine transformation

Many images that appear in nature can be created through collaging using affine transformation. The use of fractals as a compression technology is principally based on the idea that if fractals can be used to create images that are extremely lifelike and natural, then these images can be compressed by discovering the fractal transformations that can be used to replicate these. Invariably, if a few equations can be defined and create a fern leaf, then a picture of a fern leaf could be compressed to a few equations. This idea which is behind fractal compression is further enhanced by the fact that many real life images have tremendous amount of redundancy within them.

The most popularly used application of fractal compression is in the compression of images of Microsoft Encarta multimedia encyclopedia programs. One of the disadvantages of fractal technology is that it is asymmetric. The encoding process is far more process intensive compared to the decoding process, although this is fast becoming a non issue as a result of emergence of high speed processing system.

3.5.7. Wavelet coding

Signal representation using Fourier series in terms of the sinusoids has been well known for more than a century as an effective means of processing stationary as well as non stationary signals. Image compression techniques using Discrete Wavelet Transform (DWT) have received wide attention in recent years. Wavelet coding is a transform coding technique that is not limited to the block-based implementation only. Usually the wavelet transform is performed on the whole image. Wavelet transform decomposes the input signal into low-frequency and high-frequency subbands. Using a separable two dimensional filtering function, two-dimensional DWT can be computed by applying one-dimensional DWT row-wise and column-wise independently. The technique for wavelet coding is depicted in figure 3.5.7 below

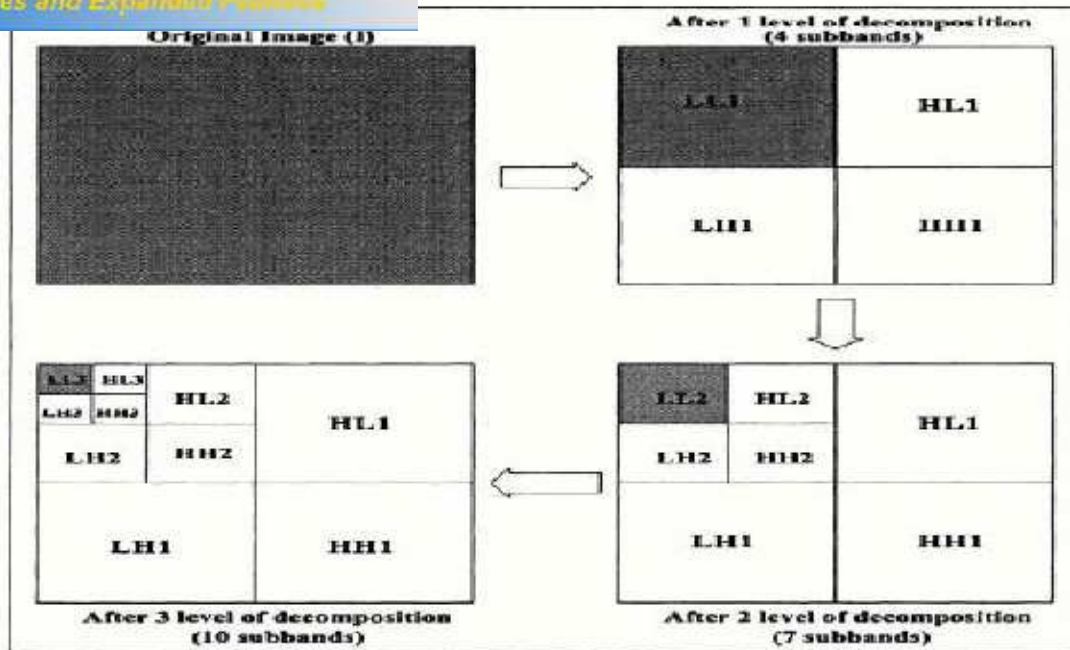
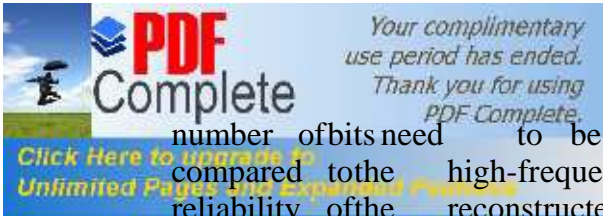


Figure 3.5. 7: Wavelet coding

The diagram shows a hierarchical wavelet decomposition of an image into ten subbands which is achieved after three levels of decomposition. As depicted in figure 3.5.7, after the first level of decomposition, the original image is decomposed into four subbands LL_1 , HL_1 , LH_1 , and HH_1 . The LL_1 subband is the low-frequency subband which can be considered as a 2:1 subsampled (hor izontal and vertically) version of the original image, and its statistical characteristic is similar to the original image which is shown by the shaded regions in the diagram. In this case, HL_1 , LH_1 , and HH_1 are called the high-frequency subbands, where HL_1 and LH_1 correspond to the horizontal and vertical high frequencies, respectively. HH_1 constitutes the high frequencies that are not in either horizontal or vertical orientations. Each of these spatially oriented (hor izontal, vertical, or diagonal) subbands mostly contain in for mation of local discontinuities in the image, and the massive energy in each of the high-frequency subbands are more in the neighborhood of areas which correspond to edge activity in the source image. Since the low-frequency subband LL_1 has comparable spatial and statistical characteristics as the source image, it can be further decomposed into four subbands LL_2 , HL_2 , LH_2 , and HH_2 . By repeating the same process for decomposition in LL_2 , the original image is decomposed into 10 subbands LL_3 , HL_3 , LH_3 , HH_3 , HL_2 , LH_2 , HH_2 , HL_1 , LH_1 , and HH_1 after three levels of pyramidal multiresolution subband decomposition, as shown in the diagram. The same approach is taken to decompose LL_3 into higher levels. Once the right wavelet filters and quantization strategy is chosen for subband we can guarantee that good compression performance will be achieved. Each decomposed subband may be encoded separately using a suitable coding scheme. We can allocate different bit-rates to different subbands. Because of the hierarchical nature of the subbands in wavelet decomposition, a smaller



number of bits need to be allocated to the high-frequency subbands in a lower level compared to the high-frequency subbands in upper levels. This helps to ascertain a proper reliability of the reconstructed image and thereby achieves good compression. Experimental results show that we can even allocate zero bits to the HHI subband and still guarantee quality of reconstructed image from natural images.

3.6 Advantages of Data Compression

The main advantage of compression is that it reduces the data storage requirements. It also offers an attractive approach to reduce the communication cost in transmitting high volume of data over long distances. The reduction in data rates can make the quality of multimedia presentation through limited-bandwidth communication channels to be increased by compression. Also because of the reduced data rates offered by compression techniques, computer network and Internet usage is becoming more and more image and graphics friendly, rather than being just data and text-centric phenomena. In fact, many of the creative applications such as digital library, digital archiving, video teleconferencing, telemedicine, and digital entertainment we see today are as a result of high-performance compression. Other secondary advantages of compression are:

Data compression may enhance the database performance because more compressed records can be stored in memory at any time. This potentially increases the probability that a record being searched will be found in the main memory.

Data security is another area, in which compression is useful. The encryption parameters can be compressed before transmitting them separately from the compressed database files to restrict access of proprietary information. An extra level of security can thus be accomplished by making the compression and decompression processes totally transparent to unauthorized users.

The rate of input-output operations in a computing device can be greatly increased due to shorter representation of data.

Data compression no doubt can reduce the cost of backup and recovery of data in computer systems by storing the enabling large database files in compressed form to be stored easily.

Self Assessment Test

- 1) Discuss the Run Length Encoding algorithm
- 2) Explain the term lossy compression?



3.7 Disadvantages of Data Compression

Data compression offers quite a good number of advantages and has opened diverse areas of opportunities for multimedia industries. It however, also offers some disadvantages which are as follows:

Data compression usually leads to reduced reliability of the data records. For example a single bit error in compressed code will cause the decoder to miss-interpret all subsequent bits producing incorrect data. You may also want to consider the transmission of very sensitive compressed data (e.g. medical information) through a noisy communication channel (such as wireless media). This could become risky because the burst error introduced by the noisy channel can destroy the transmitted data.

Similar to above problems associated with compression is the disruption of data properties, since the compressed data is different from the original data. For example, sorting and searching schemes into the compressed data may be inapplicable as the lexical ordering of the original data is no longer preserved in the compressed data.

Furthermore, the extra overhead incurred by encoding and decoding processes is one of the most serious limitations of data compression, which discourages its usage in some areas (e.g., in many large database applications). The extra overhead is usually required in order to uniquely identify or interpret the compressed data.

In many hardware and systems implementations, the extra complexity added by data compression can increase the system's cost and reduce the system's efficiency, especially in the areas of applications that require very low-power VLSI implementations.

4.0 CONCLUSION

The data compression process is said to be lossless if the recovered data are assured to be identical to the source; otherwise the compression process is said to be lossy. The multimedia data type determines the kind of technique to be used. For example compressing an image is significantly different than compressing raw binary data. Of course, general purpose compression programs can be used to compress images, but the result is less than optimal. This is because images have certain statistical properties which can be exploited by encoders specifically designed for them. Also, some of the finer details in the image can be sacrificed for the sake of saving a little more bandwidth or storage space. This also means that lossy compression techniques can be used in this area.

Lossless compression techniques are recommended when compressing data which, when decompressed, are expected to be an exact replica of the original data. This is the case when binary data such as executables, documents etc. are compressed. They need to be exactly reproduced when decompressed. On the other hand, images (and music too) need not be reproduced 'exactly'. An approximation of the original image is enough for most



purposes, as long as the error between the original and the compressed image is acceptable.

Further more, in lossless compression schemes, the reconstructed image, after compression, is numerically similar to the original image. However lossless compression can only be achieved at modest amount of compression. An image reconstructed following lossy compression contains degradation compared to the original. Often this is because the compression scheme completely discards redundant information. However, lossy schemes are capable of achieving much higher compression. Under normal viewing conditions, no visible loss is perceived (visually lossless).

5.0 SUMMARY

A common characteristic of most images is that the neighboring pixels are interrelated and therefore contain irrelevant information. The foremost task in compression is usually to find the less correlated representation of the image. Two fundamental components of compression are redundancy and irrelevancy reduction. Redundancy reduction aims at removing duplication from the signal source (image / video). Irrelevancy reduction aims at omitting parts of the signal that will not be noticed by the signal receiver say the Human Visual System (HVS). In general, three types of redundancy can easily be identified:

- Spatial Redundancy or correlation between neighboring pixel values.
- Temporal Redundancy or correlation between adjacent frames in a sequence of images (in video applications).

The aim of image compression research is to reduce the number of bits needed to represent an image by removing the spatial and spectral redundancies as much as possible. In this unit we examined different algorithms for achieving image and video compression.

6.0 TUTOR MARKED ASSIGNMENT

1. What are the major difference between Vector Quantization and transform coding ?
2. With the aid of a well labeled diagram describe three (3) compression algorithms
3. Differentiate between lossy and lossless compression?
4. What are the advantages and disadvantages of compression?

7.0 REFERENCES / FURTHER READINGS

A. Gersho and R. Gray (1992), Vector Quantization and Signal Compression, Boston, MA



D. Jankerson, G. A. Harris and P. D. Johnson, Jr (2003), Introduction to Information Theory and Data Compression, Second Edition, Chapman and Hall / CRC , Florida, USA

S. Mitra and Tinkuacharya (2003), Data Mining Multimedia, Soft Computing, and Bioinformatics, John Wiley & Sons, Inc, Hoboken, New Jersey, Canada.

S. Heath (1996) Multimedia and Communications Technology, An imprint of Butterworth-Heinemann, Jordan Hill, Oxford

<http://encyclopedia.jrank.org/articles/pages/6922/Video-Coding-Techniques-and-Standards.html>

S. Image Compression - from DCT to Wavelets : A Review,
<http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html>



UNIT 2: IMAGE PROCESSING AND HUMAN VISUAL SYSTEM

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 What is image processing?
 - 3.1.1 Subdivision of image processing
 - 3.1.2 An image processing task
 - 3.1.3 Types of Digital Images
 - 3.2 Human Visual System
 - 3.3 The Human Visual Systems Verse Computer Screen
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

A digital image is a representation of a two-dimensional image using ones and zeros i.e. binary digits. Depending on whether or not the image resolution is fixed, it may be of vector or raster type. Whenever the term "digital image" is used, it usually refers to raster images or bitmap images. These contain a fixed number of rows and columns of pixels. Pixels are the smallest individual element in an image.

Digital images can be created by a variety of input devices and techniques, such as digital cameras, scanners, coordinate-measuring machines, seismographic profiling, air borne radar, etc. They can also be synthesized from arbitrary non-image data, such as mathematical functions or three-dimensional geometric models. Lots of researches are going on that aim at providing algorithms for their transformation. Some major topics within the field of image processing include image restoration, image enhancement, and image segmentation.

2.0 OBJECTIVES

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

- Explain the meaning of image processing
- Identify the different types of images
- Explain the human visual system

3.1 What is image processing?

Image processing entails analyzing and manipulating an image in order to either improve its pictorial information for human interpretation or render it more suitable for an independent machine perception. In this unit, we shall focus on digital image processing, which involves using a computer to change the nature of a digital image. Humans like their images to be sharp, clear and detailed while machines prefer their images to be simple and uncluttered. The process of improving the pictorial information of images for human interpretation requires one or more of the following processes:

- Enhancing the edges of an image to make it appear sharper
- Removing noise from an image. A simple way to look at noise is to consider it as random errors in the image. Figure 3.1a shows an image with noise while figure 3.1b shows an image without noise



Fig. 3.1a Image with Noise



Fig 3.1b: Image with Noise removed

- Removing motion blur from an image.

To render images suitable for an independent machine perception may entail:

- Obtaining the edges of an image. See figure 3.1c and figure 3.1d



Figure 3.1c: Original Image



Figure 3.1d: The Edged Image



Source [A.

McAndrew, 2004]

- Removing details from an image.

3.1.1 Subdivision of image processing

Image processing is a broad field and generally entails the following broad subclasses which include image restoration, image enhancement, and image segmentation to mention a few. In the following section we shall briefly explain the meaning of each of these subclasses with a view of revisiting some of them in subsequent modules.

Image enhancement:

This is the process of improving the quality of a digitally stored image by manipulating the image with software or some techniques. Advanced image enhancement software may support many filters for altering images in various ways in making them more suitable for a particular application. Image enhancement tasks include for example;

- making an image lighter or darker,
- sharpening or de-blurring an out-of-focus image,
- highlighting edges,
- improving image contrast, or brightening an image
- removing noise

Image restoration

This is the reversing of the damage done to an image by a known cause, for example:

- removing of blur caused by linear motion,
- removal of optical distortions,
- removal of periodic interference.

Image segmentation

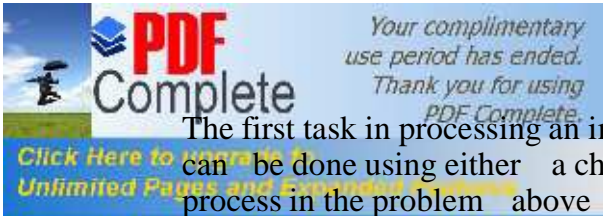
This involves subdividing an image into constituent parts, or isolating certain aspects of an image. For example:

- finding lines, circles, or particular shapes in an image,
- in an aerial photograph, one may be interested in identifying cars, trees, rivers, human beings, or roads networks.

3.1.2 An image processing task

The tasks involved in image processing include image acquisition, image preprocessing, image segmentation, image description and representation and image recognition and interpretation. Let us consider a typical real world problem where we are required to obtain the postcodes from envelopes. The image processing will involve the following:

Acquire the image.



The first task in processing an image is to acquire it from source via an input device. This can be done using either a charge-coupled device (CCD) camera or a scanner. So the first process in the problem above would be to scan the envelopes.

Preprocess the Image

This is the step taken before the major image processing task. The problem here is to perform some basic tasks in order to render the resulting image more suitable for the job to follow. This may entail enhancing the contrast, removing noise, or identifying regions likely to contain the postcode from the scanned document.

Segment the Image

Here is where we actually get the postcode; in other words we extract from the image that part of it which contains just what we want. For example, the postcode from the scanned document.

Represent and describe the Image

These terms refer to extracting the particular features which allow us to differentiate between objects that make up the image. In the case of a post code, we will be looking for curves, holes and corners which allow us to distinguish the different digits which constitute a postcode.

Recognize and interpret the Image

This means assigning labels to objects based on their descriptors (from the previous step), and assigning meanings to those labels. Still with the example of post code under consideration, we identify particular digits, and we interpret a string of four digits of the address as the postcode.

3.1.3 Types of Digital Images

When images are captured in digital forms, they are usually stored in different ways. We consider the following types of digital images

Binary

In this kind of images, each pixel is just black or white. Since there are only two possible values for each pixel, we only need one bit per pixel. Such images can therefore be very efficient in terms of storage. Images for which a binary representation may be suitable include text (printed or handwriting), fingerprints or architectural plans. Binary images are also called bi-level or two-level. This means that each pixel is stored as a single bit (0 or 1). The names black-and-white, B&W, monochrome or monochromatic are often used for this concept, but may also designate any images that have only one sample per pixel, such as grayscale images. Binary images often arise in digital image processing as masks or as the result of certain operations such as segmentation, thresholding, and dithering.

Some input / output devices, such as laser printers, fax machines, and bilevel computer displays, can only handle bilevel images.

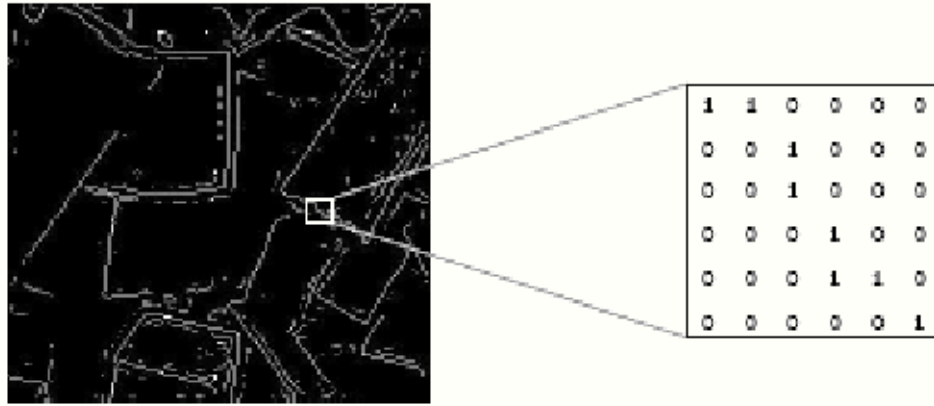


Figure 3.1.3: Binary Image

Gray scale

In photography and computing, a grayscale digital image is an image in which the value of each pixel is a single sample, that is, it carries only intensity information.

Images of this sort, also known as black-and-white, are composed exclusively of shades of gray, varying from black at the weakest intensity to white at the strongest. Grayscale images are distinct from one-bit black-and-white images, which in the context of computer imaging are images with only the two colors, black, and white (also called bilevel or binary images). Grayscale images have many shades of gray in between. Grayscale images are also called monochromatic, denoting the absence of any chromatic variation. Grayscale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. infrared, visible light, ultraviolet, etc.), and in such cases they are monochromatic proper when only a given frequency is captured. But also they can be synthesized from a full color image.

True colour, or RGB.

There are in fact a number of different methods for describing colours, but for image display and storage the standard model is RGB. RGB is the standard for the display of colours: on computer monitors; on TV sets. But it is not a very good way of describing colours. Here each pixel has a particular colour; that colour being described by the amount of red, green and blue in it. If each of these components has a range of 0 to 255, this

gives a total of $256^3 = 16,777,216$ different possible colours in the image. This is enough colour for any image. Since the total number of bits required for each pixel is 24, images are also called 24-bit colour image.



For most digital colours, image pixel is just a RGB data value (Red, Green, Blue). Each pixel's color sample has three numerical RGB components (Red, Green, Blue) to

represent the color. These three RGB components are three 8-bit numbers for each pixel. Three 8-bit bytes (one byte for each of RGB) is called 24 bit color. Each 8-bit RGB component can have 256 possible values, ranging from 0 to 255. For example, three values like (250, 165, 0), meaning (Red=250, Green=165, Blue=0) to denote one Orange pixel.

The composite of the three RGB values creates the final color for that one pixel area. In the RGB system, we know Red and Green make Yellow. So, (255, 255, 0) means Red, Green, each fully saturated (255 is as bright as 8 bits can be), with no Blue (with) the resulting color being Yellow.

Black is a RGB value of (0, 0, 0) and White is (255, 255, 255). Gray is interesting in that it has the property of having equal RGB values. So (220, 220, 220) is a light gray (near white), and (40,40,40) is a dark gray (near black). Gray has no unbalanced color.

Since gray has equal values in RGB, Black & White grayscale images only use one byte of 8 bit data per pixel instead of three. The byte still holds values 0 to 255, representing 256 shades of gray.

Self Assessment Test

- 1) What is image processing ?
- 2) List the steps in image processing task

3.2 Human Visual System

The visual system is the part of the central nervous system which enables organisms see. It interprets the information from visible light to build a representation of the world surrounding the body. The visual system accomplishes a number of complex tasks, including the reception of light, and the formation of monocular representations; the construction of a binocular perception from a pair of two dimensional projections; the identification and categorization of visual objects; assessing distances to and between objects; and guiding body movements to visual objects. The psychological manifestation of visual information is known as visual perception.

The eye is a complex biological device. The functioning of a camera often with the workings of the eye, mostly since both focus light from external objects in the visual field onto a light-sensitive medium. In the case of the camera, this medium is film or an electronic sensor; in the case of the eye, it is an array of visual receptors. With this simple geometrical similarity, based on the laws of optics, the eye functions as a transducer, as does a CCD camera.



Figure 3.2: Human visual system

Light entering the eye is refracted as it passes through the cornea. It then passes through the pupil (controlled by the iris) and is further refracted by the lens. The cornea and lens act together as a compound lens to project an inverted image onto the retina. The retina contains two types of photo sensor cells: rods and cones. There are 75 to 150 million rod cells in the retina. The rods contain a blue-green absorbing pigment called rhodopsin. Rods are used primarily for night vision (also called the scotopic range) and typically have no role in color vision. Cones are used for daylight vision (called the photopic range). The tristimulus theory of color perception is based upon the existence of three types of cones: red, green and blue. The pigment in the cones is unknown. We do know the phenomenon called adaptation (a process that permits eyes to alter their sensitivity) occurs because of a change in the pigments in the cones. The retina cells may also inhibit each another from creating a high-pass filter for image sharpening. This phenomenon is known as lateral inhibition. Finally, the eye-brain interface enables an integration between the sensors polar coordinate scans, focus, iris adjustments and the interpretation engine. These interactions are not typical of most artificial image processing systems.

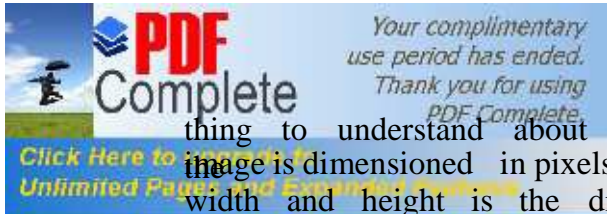
3.3 The Human Visual Systems Verse Computer Screen

One big advantage that the human eye has over a computer screen or electronic camera is its spectral responses to colour. Unlike the PC screen, which typically uses the same number of bits for red, green and blue (RGB) images, the eye's sensitivity is such that the weighting is different, and as a result, it becomes difficult to differentiate between colours that the PC is capable of displaying. If the eye cannot discriminate between the colours, data can be saved by artificially restricting the data that the PC displays. In other words, if the eye cannot discriminate between 16 or 24 bits per pixel, why waste bandwidth transmitting all the data? This data reduction to match the eye's response is normally done by converting RGB data into other formats.

4.0 CONCLUSION

Any image from a scanner, or from a digital camera, or in a computer is represented in digital form for processing. Digitizing is the process of capturing and converting an

analog signal in digital form. Computers process only digitized images. The fundamental



One thing to understand about digital images is that they consist of pixels. The size of an image is dimensioned in pixels, X columns wide and Y rows tall. The number of pixels in width and height is the digital image's spatial resolution. The ratio of width to height in inches is known as image's aspect ratio. The bit depth of an image is the number of bits used to store the value in each pixel. Since the images are to be examined and acted upon by people, the understanding of how the human visual system operates is necessary. The major topics within the field of image processing include image restoration, image enhancement, and image segmentation.

5.0 SUMMARY

In this unit we covered image processing and its subfields, examined the different types of image representation within the computer and the human visual system. We also made a comparison between the human visual system and the computer screen.

6.0 TUTOR MARKED ASSIGNMENTS

1. With the aid of a labeled diagram explain how the human visual system works
2. Explain the different types of image representation within the computer

7.0 REFERENCES/FURTHER READINGS

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

M. S. Onka, V. Hlavac, and R. Boyle (1999), Image Processing, Analysis and Machine Vision. PWS Publishing, second edition

S. E. Umbaugh (1998) Computer Vision and Image Processing: A Practical Approach Using CVIPTools. Prentice-Hall.

<http://en.wikipedia.org/wiki/YIQ>



PDF Complete

Your complimentary use period has ended. Thank you for using PDF Complete.

Click Here to upgrade to Unlimited Pages and Expanded Features

UNIT 3: 2D DATA TRANSFORM WITH DTFT, DFT, DCT

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Background
 - 3.2 The Fast Fourier Transform
 - 3.3 The two-dimensional DFT
 - 3.4 Some properties of the two dimensional Fourier transform
 - 3.5 DTFT
 - 3.6 Discrete Cosine Transform (DCT)
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

From the knowledge of engineering mathematics and signal processing we know that periodic function can be viewed in either the time domain or the frequency domain. It is very important to realize that both viewpoints are valid and either can be used for signal. A transform is a mathematical tool used to change one representation into another

and to make calculation easier. Specifically, the Fourier transform's utility lies in its ability to analyze a signal in the time domain for its frequency content. The transform works by first translating a function in the time domain into a function in the frequency domain. The signal can then be analyzed for its frequency content because the Fourier coefficients of the transformed function represent the contribution of each sine and cosine function at each frequency. An inverse Fourier transform does just the reverse, that is, it transforms data from the frequency domain into the time domain.

2.0 OBJECTIVES

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

- Explain how signals are transformed from one domain to the other
- Explain the different classes of signals
- Explain the meaning and the different types of transforms
- Discuss applications of Fourier transforms in multimedia processing

3.0 MAIN CONTENT

3.1 Background

The Discrete Fourier Transform (DFT) is a specific form of Fourier analysis to convert 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

$$f = [f_0 \ f_1 \ f_2 \ \dots \ f_{N-1}]$$

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

$$F = [F_0 \ F_1 \ F_2 \ \dots \ F_{N-1}]$$

Where

$$F_u = \sum_{x=0}^{N-1} \exp^{-2i \frac{xu}{N}} f_x \quad 3.2$$

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

$$f_x = \sum_{u=0}^{N-1} \exp 2i \frac{xu}{N} F_u \quad 3.3$$

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 algorithms 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 efficient when the vector length is a power of 2.

$$F(u) = \sum_{x=0}^{N/2-1} f(x) \exp^{-2i \frac{xu}{N}} \quad 3.4$$

Your complimentary use period has ended. Thank you for using PDF Complete.

Click Here to upgrade to Unlimited Pages and Expanded Features

$$f(x) = \int_M^1 F(u) \exp \frac{xu}{2p} 3.5$$

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 multiplications required for each method. For a vector of length $2n$, the direct method takes $(2n)^2$ multiplications; while the FFT takes only n^2 . Here the saving with respect to time is of an order of $2n/n$. 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	64	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 The two-dimensional DFT

In two dimensions, the DFT takes a matrix as input, and returns another matrix, 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 of f and write

$$F = F(f).$$

Then the original matrix f is the inverse Fourier transform 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

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

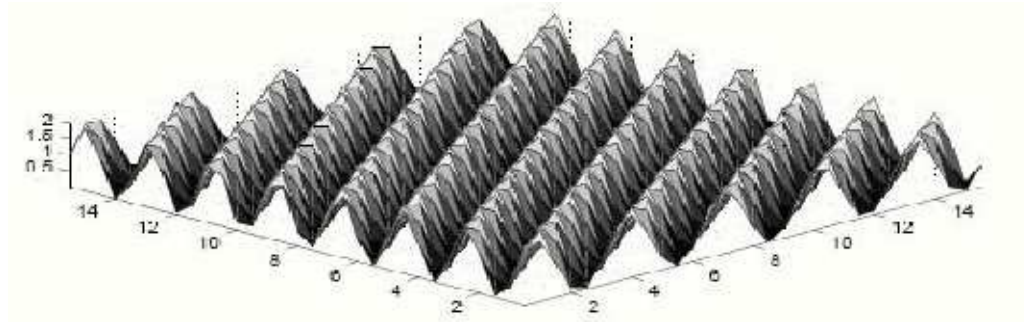


Figure 3.2: Corrugate Function

A sample of such function is depicted in figure 3. 2. And this is in fact exactly what the two dimensional Fourier transform 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 $M \times N$ 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:

$$F(u,v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x,y) \exp - 2i \frac{xu}{M} - \frac{yv}{N} \quad 3.6$$

$$F(x,y) = \frac{1}{MN} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) \exp - 2i \frac{xu}{M} - \frac{yv}{N} \quad 3.7$$

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

Self Assessment Test

- 1) Study Table 3.1 and see if you can reproduce the values in columns 2, 3, and 4 based on the theory provided

3.4 Some properties of the 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 transform reveals similarity except for the scale factor $1/MN$ 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 DTFT

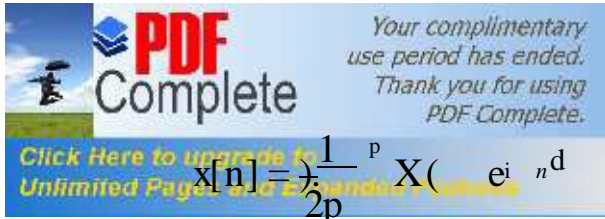
The Discrete-time Fourier transform (DTFT) is one of the specific forms of Fourier analysis. As such, it transforms one function into another, which is called the frequency representation, or simply the "DTFT", of the original function (which is often a function in the time-domain). But the DTFT requires 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. It is Pontryagin dual to the Fourier series which transforms from a periodic domain to a discrete domain.

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

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

The following inverse transform recovers the discrete-time sequence



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

$$= \int_{-\frac{1}{2T}}^{\frac{1}{2T}} X(f) e^{j2\pi f n T} df$$

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)

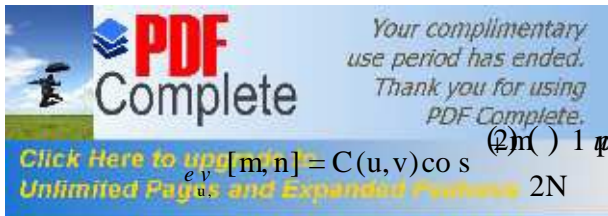
A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. In particular, a DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers. DCTs are equivalent 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 $N \times N$ image for an example is as given by:

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

for $N > 0 = u, v$

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

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



$$f[m, n] = C(u, v) \cos \frac{(2m-1)\pi u}{2N} \cos \frac{(2n-1)\pi v}{2N}$$

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 (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 $2N \times 2N$.

Some sample basis functions are shown in Figure 3.5, for a value of $N=8$. It can be shown that this basis is orthonormal.

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

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

for N $0 \leq m, n < N$

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

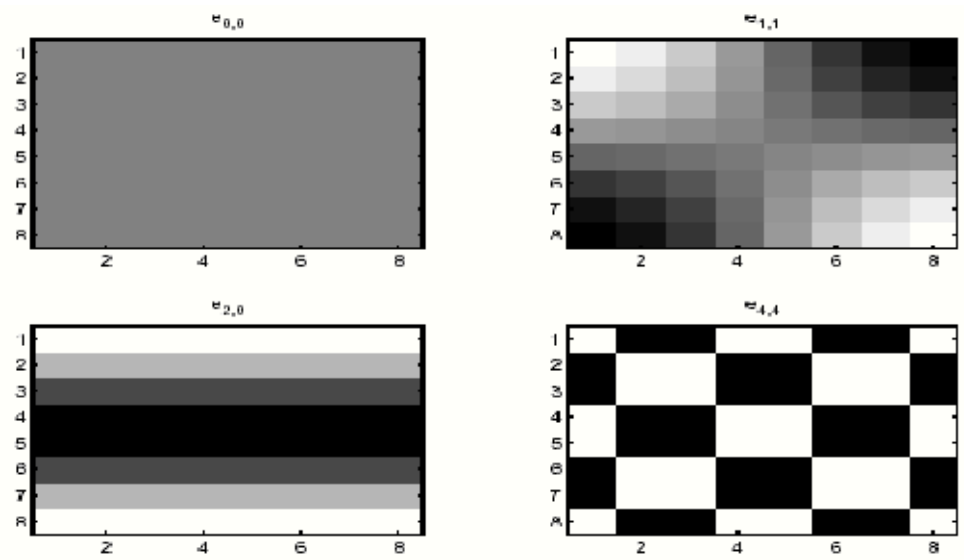


Figure 3.5: Sample basis functions for an 8x8 block of pixels.


 Your complimentary use period has ended. Thank you for using PDF Complete.

[Click Here to upgrade to Unlimited Pages and Expanded Features](#)

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 $32 \times 32 = 1024$ pixels, we only transmitted $10 \times 10 = 100$ coefficients, which is a compression ratio of 10.24 to 1.

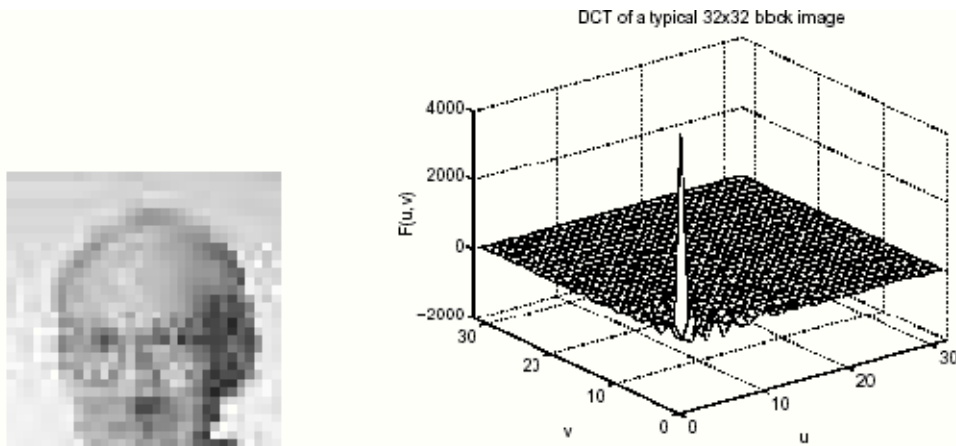


Figure 3.6: (a) 32 x 32 pixel version of our standard test image. (b) The DCT of this image.

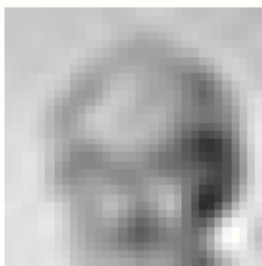


Figure 3.7: 32 x 32 pixel image reconstructed from 10 x 10 subset of DCT coefficients. Overall information has been retained, but some detail has been lost.



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 (Hotelling 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

The 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 TUTOR MARKED ASSIGNMENTS

- 1) The DCT is the basis of many widespread image-coding standards . Explain why this is possible**
- 2) Explain two (2) properties of the two dimensional Fourier transform**

7.0 REFERENCES / FURTHER READINGS

A. 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

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

S. Image Compression - from DCT to Wavelets : A Review,
<http://www.acm.org/crossroads/xrds6-3/sahaimgcoding.html>



MODULE 3

UNIT 1: IMAGE PERCEPTION, IMAGE ENHANCEMENT WITH HISTOGRAM ANALYSIS

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Meaning of Histogram
 - 3.2 Image Enhancement
 - 3.2.1 Histogram stretching (Contrast stretching)
 - 3.2.2 Histogram equalization
 - 3.2.3 Edge Sharpening
 - 3.4 Filters and Noise Removal in Images
 - 3.5 Using MATLAB
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

Image processing is a very important aspect of digital signal processing (DSP) application area. In this unit, we shall examine briefly the application of DSP techniques in the enhancement of images. To do this we shall first provide some mathematical foundations for histogram analysis and then apply them for the enhancement of images. Also in this unit, we will introduce the concept of filters and attempt to cover how they are used for image enhancement.

2.0 OBJECTIVES

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

- Explain the meaning of histogram
- Describe how histograms are used for image enhancement
- Identify sources of noise
- Describe how filters are used for the removal of noise

3.0 MAIN CONTENT

3.1 Meaning of Histogram

Given a grayscale image, its histogram consists of the histogram of its grey levels; that is, a graph indicating the number of times each grey level occurs in the image. Quite a good

number of inferences can be made from the appearance of an image from its histogram. The following statements are results from experiments with histograms.

- In a dark image, the grey levels (and hence the histogram) would be clustered at the lower end:
- In a uniformly bright image, the grey levels would be clustered at the upper end:
- In a well contrasted image, the grey levels would be well spread out over much of the range:

Figure 3.1 shows a poorly contrasted image and its histogram. The histogram was obtained by using MATLAB software. If you are familiar with MATLAB you can use it to perform a good number of image processing functions. For example, you can view the histogram of an image in MATLAB by using the `imhist` function. Many of the topics or diagrams discussed in this section can easily be obtained by using MATLAB provided you have your data or image to work with. Additional information is provided on MATLAB at the end of this unit.

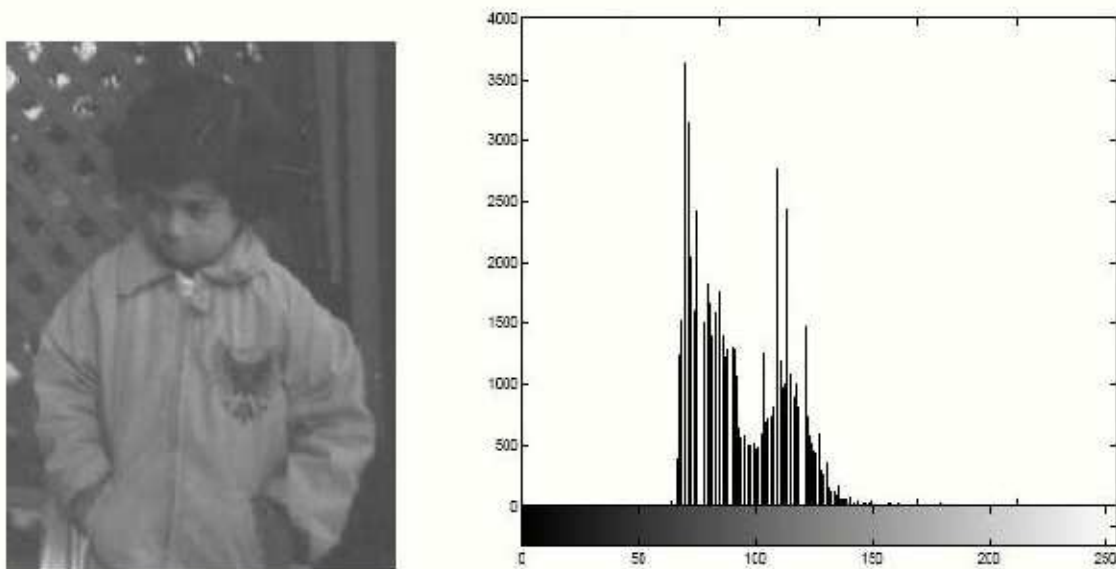


Figure3.1: An image with its histogram: Source [A. McAndrew, 2004]

3.2 Image Enhancement

Image enhancement is the processing of images to improve their appearance. There are a variety of methods, which are suitable for different objectives. Some objectives are to improve the image quality and visual appearance to human viewer. Other ones include the sharpening of an image to make the processed image better in some sense than the processed one.

In the subsequent sections, we shall discuss some of the approaches to image enhancement. For simplicity, we only use gray-scale images.

3.2.1 Histogram stretching (Contrast stretching)

Whenever you have a poorly contrasted image such as shown in figure 3.1 above, you may wish to enhance its contrast by spreading out its histogram. One approach to use is histogram stretching. Suppose we have an image with the histogram shown in figure 3.1.1, associated with a table of the number of gray values:

Greylevel	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14							
I	15	0	0	0	70	10	45	70	35	0	0	0	0	5								

Table 3.1.1 a

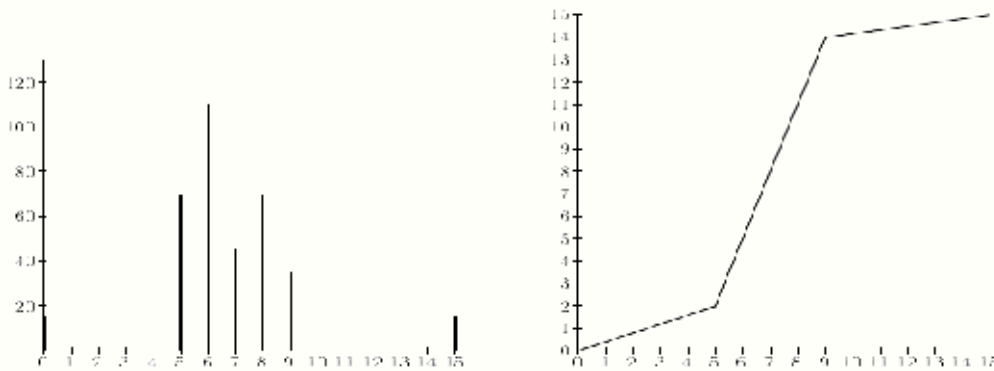


Figure 3.1.1a: A histogram of a poorly contrasted image, and a stretching function [A. McAndrew, 2004]

Assuming $n=360$, the grey levels in the centre of the range can be stretched out by using a piecewise linear function. Please, note that you can derive your piecewise linear function. The MATLAB software could assist you to do this. For example the function given by the following equation has the effect of stretching the grey levels 5-9 to grey levels 2-14 according to the equation

$$j = \frac{14 + 2}{9 - 5} (i - 5) + 2$$

where i is the original grey level and j its result after the function has been applied. Grey levels outside this range are either left alone as in the case under discussion or transformed according to the linear function derivable from the right side diagram in Figure 3.1.1. This yields the corresponding grey values in table 3.1.1b and histogram in figure 3.1.1b which indicate an image with a greater contrast than the original.

PDF Complete
 Your complimentary use period has ended.
 Thank you for using PDF Complete.
 Click Here to upgrade to Unlimited Pages and Features

i	5	6	7	8	9				
j	2	5	8	11	14				

Table 3.1.1 b

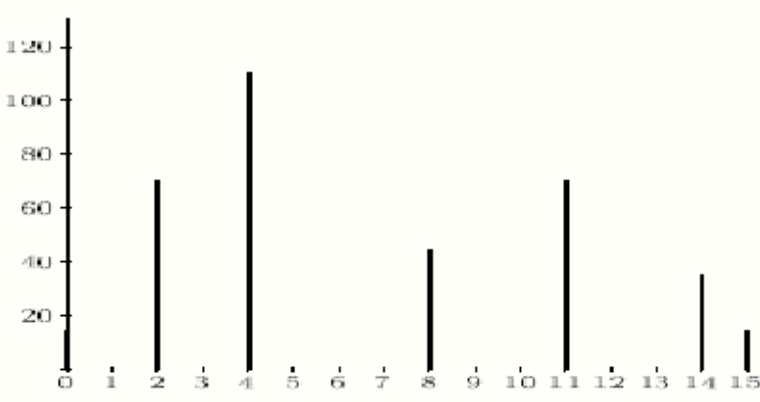


Figure 3.1. 1b: Histogram of I mage with better Contrast

3.2.2 Histogram equalization

The major challenge with histogram stretching is that they require user input. Often times, a better approach to image enhancement is provided by histogram equalization, which is an entirely automatic procedure. The principle here is to change the histogram to one which is uniform; that is one that every bar on the histogram is of the same height meaning that, each grey level in the image occurs with the same frequency. In the sense, this looks impracticable, but the fact remains that, the result of histogram equalization provides very good results for enhancing the quality of an image.

Suppose we have an image with L different grey levels 0, 1, 2, ..., L-1 and that the grey level i occurs n_i times in the image. Assuming also that the total number of pixels in the image is n such that $n_0 + n_1 + n_2 + \dots + n_{L-1} = n$. To transform the grey level to obtain a better contrasted image we change grey level i to

$$\frac{n_i + 1}{n} (L - 1)$$

with the values obtained rounded to the nearest integer.

Let us now consider the following example in order to have a better understanding of the concept discussed above:

Suppose a 4-bit greyscale image has the histogram shown in figure 3.1.1 associated with a table of the number n_i of grey values shown in table 3.1.2

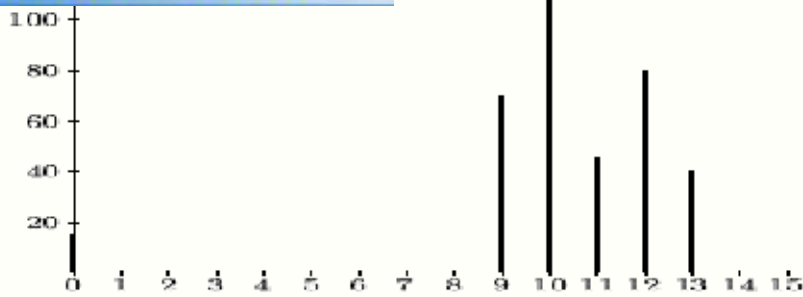


Figure 3.1. 1a: Histogram of grey scale values before equalization

Greylevel	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14				
15	15	0	0	0	0	0	0	0	70	110	45	80	40	0					

Table 3.1.2: Grey scale values of an image

We would expect this image to be uniformly bright, with a few dark dots on it. To equalize this histogram, we form running totals of the n_i , and multiply each by $15/360 = 1/24$

Grey Level i	n_i	n_i	$(1/24) n_i$	Rounded	Value
0	15	15	0.63	1	
1	0	15	0.63	1	
2	0	15	0.63	1	
3	0	15	0.63	1	
4	0	15	0.63	1	
5	0	15	0.63	1	
6	0	15	0.63	1	
7	0	15	0.63	1	
8	0	15	0.63	1	
9	70	85	3.63	4	
10	110	195	8.13	8	
11	45	240	10	10	
12	80	320	13.33	13	
13	40	360	15	15	
14	0	360	15	15	
15	0	360	15	15	

Table 3.1.3: Equalized values

This will give the following transformation of grey values obtained by reading off the first and last column in the above table:

Original grey level i	0	1	2	3	4	5	7	8	9	10	11	12	13	14				
Final grey level j	1	1	1	1	1	1	1	4	8	10	13	15	15					

Table 3.1.4: Original and Final grey values

The resulting histogram from the table 3.1.4 is depicted in Figure 3.1. 1b:

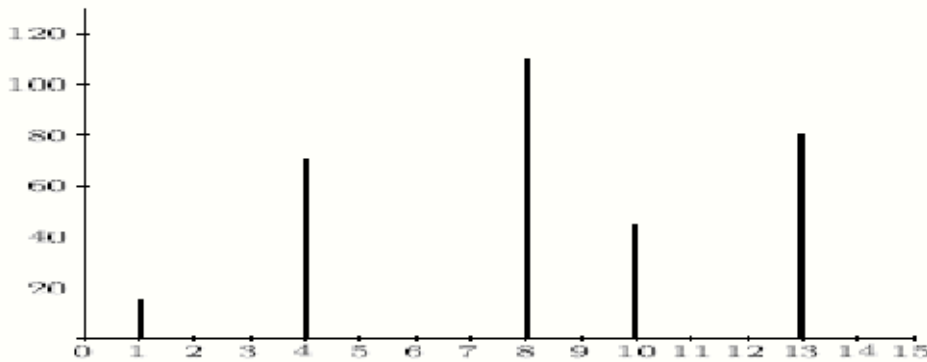


Figure 3.1.1b: Histogram of grey scale values after equalization

3.2.3 Edge Sharpening

Another approach to making edges in an image slightly sharper and crisper, which generally results in an image more pleasing to the human eye is by performing **filtering**. The operation is also known as edge enhancement, edge crispening, or unsharp masking. This last term comes from the printing, industry. Unsharp masking is well known to photographers and astronomers who used the method as darkroom technique to enhance faint details in photographic prints. As designed by photographers, blurred, reverse-contrast negative (or unsharp mask) is made of the original negative. These two negatives are sandwiched together in perfect registration in the enlarger and a print is made. To perform this operation on a computer, an unsharp mask is produced by blurring and reducing the amplitude of the original image; the unsharp mask is then subtracted from the original to produce a sharpened image. The idea of unsharp masking is to subtract a scaled unsharp version of the image from the original. In practice we can achieve this effect by subtracting a scaled blurred image from the original.

The schema for unsharp masking is shown in figure 3.1.3a

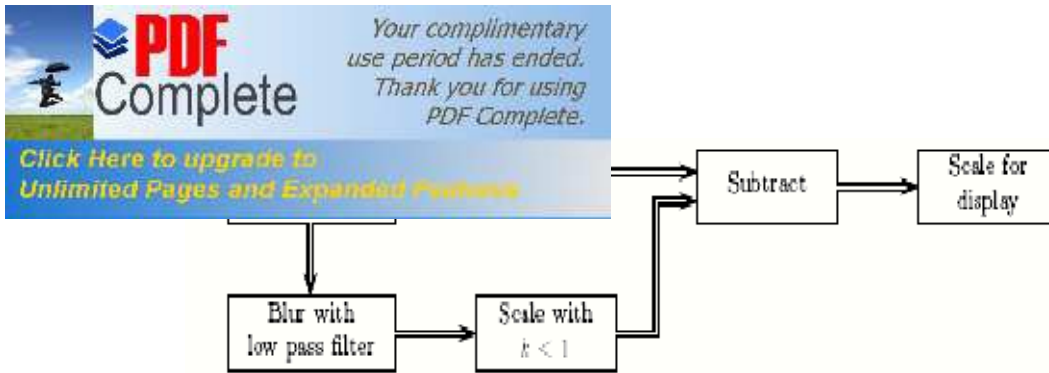


Figure 3.1. 3a: Schema for Unsharp masking

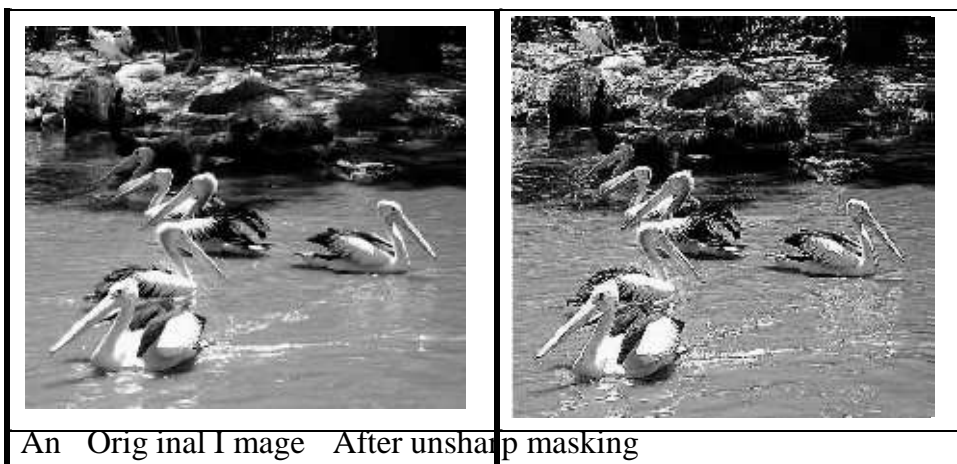


Figure 3.1. 3b: Schema for Unsharp masking

3.4 Filters and Noise Removal in Images

Noise is often introduced during the analog-to-digital conversion process as a side-effect of the physical conversion of patterns of light energy into electrical patterns. Filters usually would have some effect on image processing tasks. To be able to choose the most appropriate filter for image processing, we need to understand the notion of frequency. Roughly speaking, the frequencies of an image are a measure of the amount by which grey values change with distance. High frequency components are characterized by large changes in grey values over small distances; examples of high frequency components are edges and noise. Low frequency components, on the other hand, are parts of the image characterized by little change in the grey values. These may include backgrounds, skin textures. We then say that a filter is a:

high pass filter if it passes over the high frequency components, and reduces or eliminates low frequency components, and a

low pass filter if it passes over the low frequency components, and reduces or eliminates high frequency components.



Filtering operation selectively reduces or enhances low or high spatial frequency in the object image. This means, that a Low pass filtering, otherwise known as "smoothing" can be employed to remove high spatial frequency noise from a digital image. Visit the topic on noise removal in unit 3 of this module.

3.5 Using MATLAB

MATLAB stands for MATrix LABoratory. It is a programming language for technical computing from The MathWorks, Natick, MA. It is used for a wide variety of scientific and engineering calculations, especially for automatic control and signal processing. MATLAB runs on Windows, Mac and a variety of Unix-based systems. Developed by Cleve Moler in the late 1970s and based on the original LINPACK and EISPACK FORTRAN libraries, it was initially used for factoring matrices and solving linear equations. MATLAB has wide area of application today. Most of the image processing tasks can be handled by it with a little programming skill. You can get a copy from the Internet or other sources for your PC and explore its features.

Self Assessment Test

- 1) What is the meaning of Edge Sharpening?
- 2) Access the Internet for more information on MATLAB

4.0 CONCLUSION

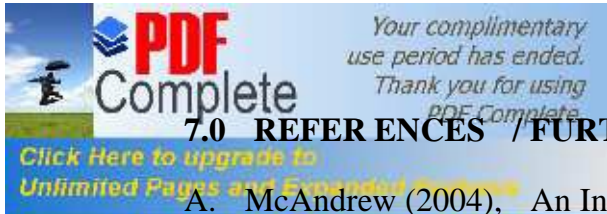
The unsharp filter is a simple sharpening operator which derives its name from the fact that it enhances edges (and other high frequency components in an image) via a procedure which subtracts an unsharp, or smoothed, version of an image from the original image. The unsharp filtering technique is commonly used in the photographic and printing industries for crispening edges.

5.0 SUMMARY

In this unit, we covered histogram and its application in image enhancement. We also took a look at filters and introduce the features of MATLAB which is one software that is currently being used for diverse engineering, science and business applications.

6.0 TUTOR MARKED ASSIGNMENTS

- 1) What is a histogram?
- 2) Describe the processes of Histogram equalization in image enhancement
- 3) Explain the term Edge Sharpening



7.0 REFERENCES / FURTHER READINGS

A. McAndrew (2004), An Introduction to Digital Image Processing with Matlab, Notes for SCM2511 Image, School of Computer Science and Mathematics, Victoria University of Technology

S. Mitra and Tinkuacharya (2003), Data Mining Multimedia, Soft Computing, and Bioinformatics, John Wiley & Sons, Inc, Hoboken, New Jersey, Canada

A. N. Netravali and B. Haskell (1988), Digital Pictures. New York: Plenum Press

A. K. Jain (1989.) Fundamentals of Image Processing. Englewood Cliffs, NJ:, Prentice-Hall,

W. B. Pennebaker and J. L. Mitchell,(1993) JPEG: Still Image Data Compression, Standard. New York: Chapman & Hall



UNIT 2: MORPHOLOGICAL OPERATORS

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Morphological operators
 - 3.1.1 Translation and Reflection
 - 3.1.1.1 Translation
 - 3.1.1.2 Reflection
 - 3.1.2 Dilation and Erosion
 - 3.1.2.1 Dilation
 - 3.1.2.2 Erosion
 - 3.2 Relations between erosion and dilation
 - 3.3 Opening and closing
 - 3.4 An application: noise removal
 - 3.5 Relationship between opening and closing
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignments
- 7.0 References / Further Readings

1.0 INTRODUCTION

Morphology is a branch of image processing which is particularly useful for analyzing shapes in images. We shall examine basic morphological operators that are applied to binary and greyscale images.

2.0 OBJECTIVES

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

- Explain the meaning of morphological operators
- Explain the applications of morphological operations in image processing

3.0 MAIN CONTENT

3.1 Morphological operators

The theory of mathematical morphology can be developed in many different ways. We shall adopt one of the standard methods which use operations on sets of points. Some morphological operations are discussed as follows:

3.1.1 Translation and Reflection

3.1.1.1 Translation

Given that A is a set of pixel in a binary image and $w = (x, y)$ is a particular coordinate point. Then A_w is the set A translated in the direction (x, y) . This means that: $A_w = \{(a, b) + (x, y) : (a, b) \in A\}$

Let us consider a practical example by using figure 3.1.1a

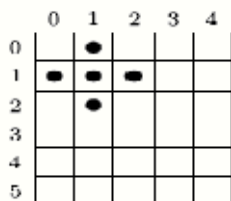


Figure 3.1.1a: Binary Image

Image

The figure depicts a shaped set and let $w = (2, 2)$. The set A has been shifted in the x and y directions by the values given in w . We observe that here we are using matrix coordinates, rather than Cartesian coordinate, so that the origin is at the top left, x goes down and y goes across. The result of the translation operation is depicted in figure 3.1.1b

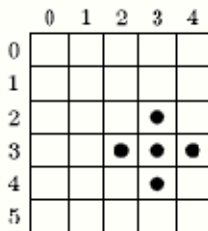


Figure 3.1.1b: Result of Translation Operation

3.1.1.2 Reflection

If A is set of pixel, then its reflection, denoted by A' is obtained by reflecting A in the origin

$$A' = \{(x, -y) : (x, y) \in A\}$$

For example, in Figure 3.1.1c the open and close circles form sets which are reflections of each other.

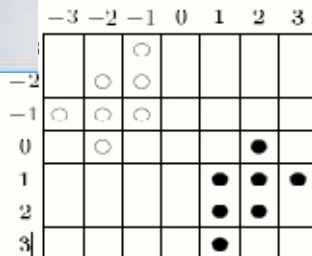


Figure 3.1. 1c: Result of Reflection Operation

3.1.2 Dilation and Erosion

These are the two basic operations that constitute morphology. All other operations are derived from these two operations.

3.1.2.1 Dilation

Assuming that A and B are sets of pixels, the dilation of A by B , denoted by $A \oplus B$ is defined as $A \oplus B = \bigcup_x (A \oplus B)_x$

This implies that for every point $x \in B$ we translate A by those coordinates. Then we take the union of all these translations. This can also be written as

$$A \oplus B = \{(x,y) + (u,v) : (x,y) \in A, (u,v) \in B\}$$

From this last definition, dilation is shown to be commutative; that is

$$A \oplus B = B \oplus A$$

An example of dilation is given in Figure 3.1.2a. In the translation diagram; the grey squares show the original position of the object. Note that $A(0,0)$ is of course just A itself.

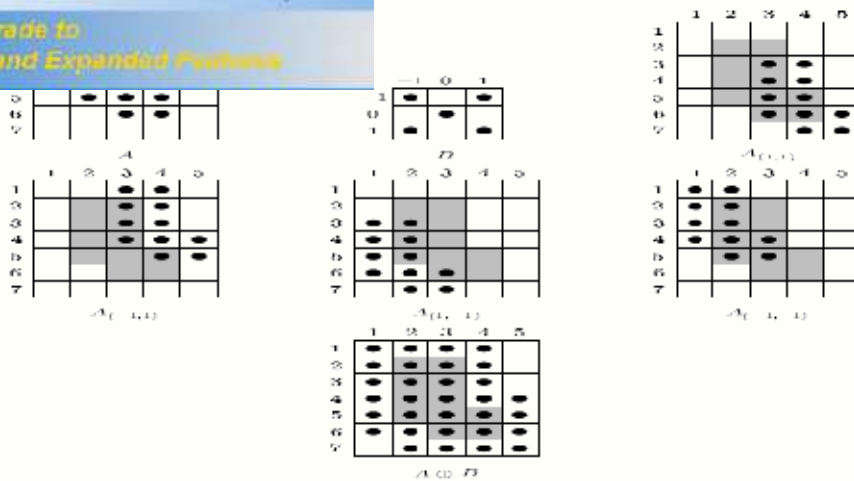


Figure 3.1.2a: Dilation [Source A. McAndrew 2004]

In this example, we have

$B = \{(0,0), (1,1), (-1, 1)(1,-1), (-1,-1)\}$ and these are the coordinates by which we translate A.

In general $A \oplus B$ can be obtained by replacing every point (x,y) in A with a copy of B, placing the $(0,0)$ point of B at (x,y) . Equivalently, we can replace every point (u,v) of B with a copy of A.

3.1.2.2 Erosion

Given sets A and B, the erosion of A by B, written $A \ominus B$ is defined as

$$A \ominus B = \{w: B \oplus w \subseteq A\}$$

In other words, the erosion of A by B consist of all points $w = (x,y)$ for $B \oplus w$ is in A. To perform an erosion, we can move B over A, and find all the places it will fit, and for each place mark down the corresponding $(0,0)$ point of B. The set of all such points will form the erosion. An example of erosion is given in Figure 3.1.2

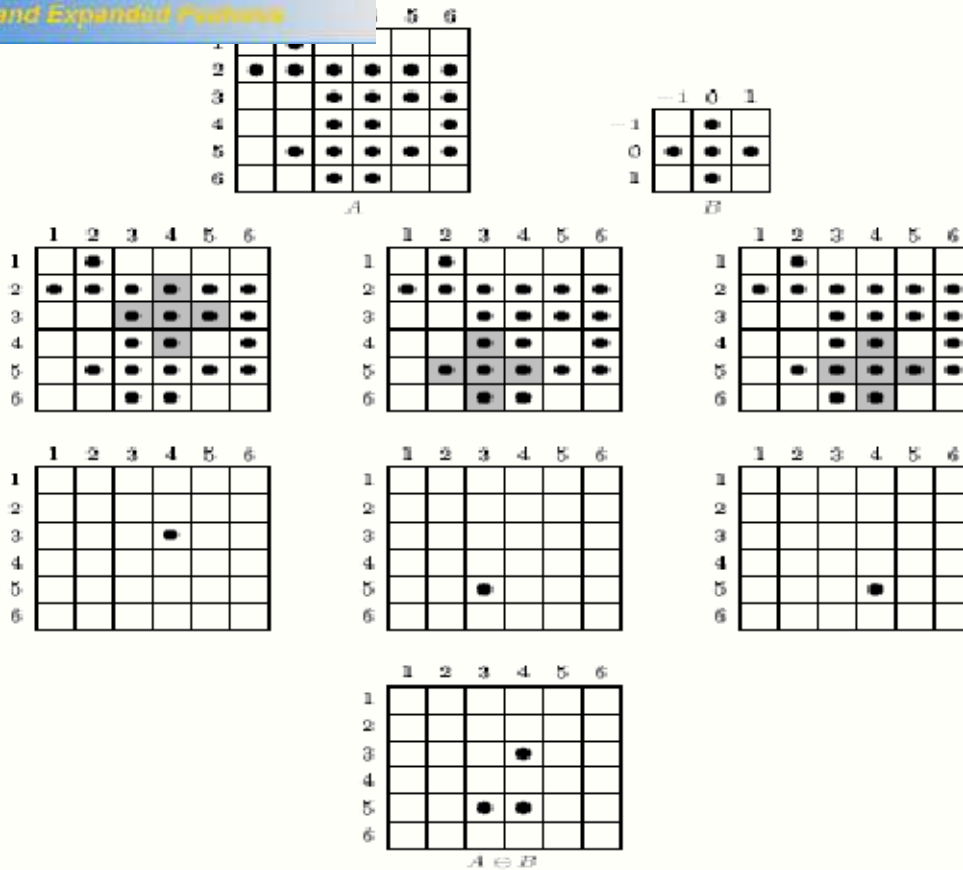


Figure 3.1. 2b: Erosion [Source A. McAndrew 200 4]

3.2 Relations between erosion and dilation

It can be shown that erosion and dilation are inverses of each other; more precisely, the complement of erosion is equal to the dilation of the complement. Thus

$$\overline{A \ominus B} = A \ominus \overline{B}$$

3.3 Opening and Closing

These operations may be considered as second level operations; in that they build on the basic operations of dilation and erosion. They are also, as we shall see, better behaved mathematically.

Opening

Given A and a structuring element B, the opening of A by B, denoted by $A \ominus B$ is defined as $A \ominus B = (A \ominus B) \oplus B$

So an opening consists of an erosion followed by a dilation. An equivalent definition is

That is $A \ominus B$ is the union of all translations of B which fit inside A . Note the difference: the erosion consist only of the $(0, 0)$ point of B for those translations which fit inside A ; the opening consist of all of B . An example of opening is given in Figure

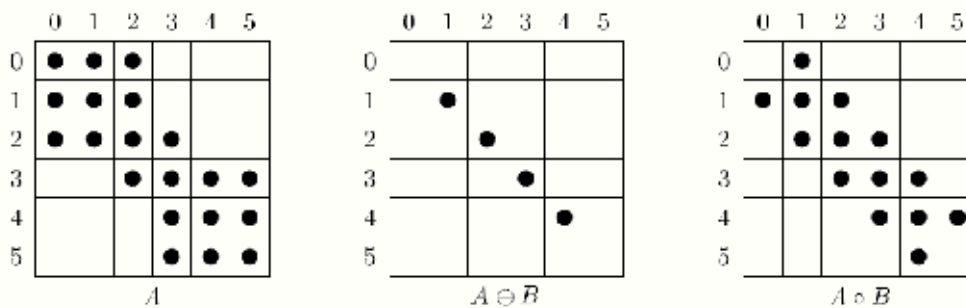


Figure 3.3a: Opening [Source A. McAndrew 2004]

The opening operation satisfies the following properties:

- 1 $(A \odot B) \subseteq A$. Note that this is not the case with erosion; as we have seen, an erosion may not necessarily be a subset
- 2 $(A \odot B) \odot B = A \odot B$. That is, an opening can never be done more than once. This property is called idempotence. Again, this is not the case with erosion; you can keep on applying a sequence of erosion to an image until nothing is left.
- 3 $(A \subseteq C) \implies (A \odot B) \subseteq (C \odot B)$.
- 4 Opening tends to smooth an image, to break narrow joins, and to remove thin protrusions.

Closing

Related to opening we can define closing, which is considered as a dilation followed by an erosion. It is denoted by $A \bullet B$;

$$A \bullet B = (A \ominus B) \odot B$$

Another definition of closing is that $x \in A \bullet B$ if all translations B_w which contains x have non-empty intersections with A . An example of closing is given in figure 3.3b. This closing operation satisfies

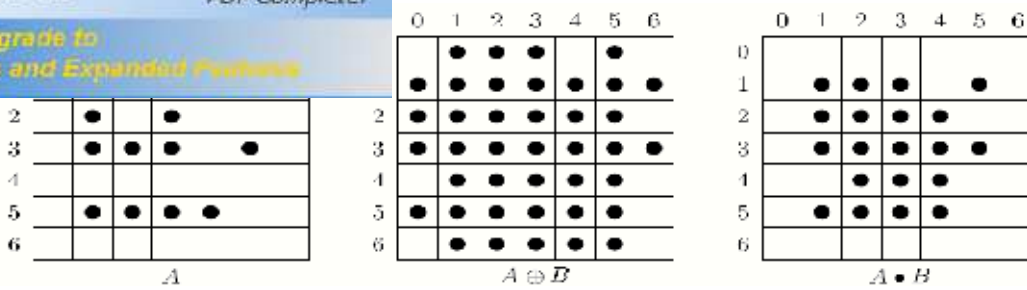


Figure 3.3b: Closing

1. $A \cdot (A \cdot B)$
2. $(A \cdot B) \cdot B = A \cdot B$; that is, closing, like opening, is idempotent
3. If $A \subseteq C$, then $(A \cdot B) \subseteq (C \cdot B)$
4. Closing tends also to smooth an image, but it fuses narrow breaks and thin gulfs and eliminates small holes.

3.4 An application: noise removal

Suppose A is a binary image corrupted by impulse noise—some of the black pixels are white and some of the white pixels are black. An example is given in Figure 3.4. Then $A \cdot B$ will remove the single black pixels, but will enlarge the holes. We can fill the holes by dilating twice:

$$((A \cdot B) \cdot B) \cdot B$$

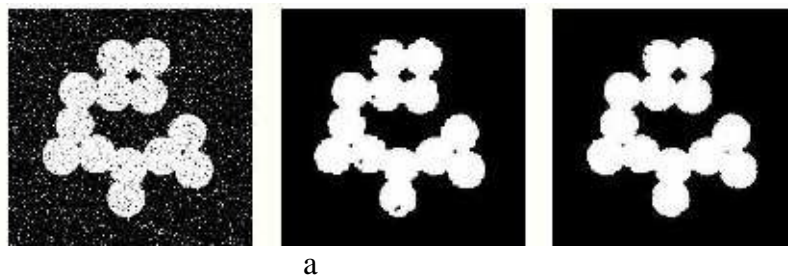
The first dilation returns the holes to their original size; the second dilation removes them. But this will enlarge the object in the image. To reduce them to their correct size, perform a final erosion;

$$(((A \cdot B) \cdot B) \cdot B) \cdot B$$

The inner two operations constitute an opening; the outer two operations a closing. Thus this noise removal method is in fact an opening followed by a closing.

$$((A \cdot B) \cdot B)$$

This is called morphological filtering



b Figure 3.4: Application of morphological operations in noise removal [Source A. McAndrew 2004]



Your complimentary use period has ended. Thank you for using PDF Complete.

Click Here to Upgrade to Unlimited Pages and Expanded Features

3.5 Relationship between opening and closing

Opening and closing share a relationship very similar to that of erosion and dilation; the complement of an opening is equal to the closing of a complement, and complement of a closing is equal to the opening of a complement. Specifically,

$$\overline{A \bullet B} = \overline{A} \bullet \overline{B} \quad \text{and} \quad \overline{A \bullet B} = \overline{A} \bullet \overline{B}$$

4.0 CONCLUSION

Morphology is a very important area of mathematics for image processing. This unit has helped to provide a mathematical foundation to validate some concepts discussed in this study material

5.0 SUMMARY

In this unit we have covered some important morphological operators, their relationships and areas of applications in multimedia systems.

6.0 TUTOR MARKED ASSIGNMENTS

- 1.) Using a relevant example, define the following terms;
 - a) Dilation
 - b) Erosion
- 2.) Define an opening operation and state the properties that are usually satisfied by this operation

7.0 REFERENCES / FURTHER READINGS

- S. E. Umbaugh (1998) Computer Vision and Image Processing: A Practical Approach Using CVIPTools. Prentice-Hall.
- A. McAndrew (2004), An Introduction to Digital Image Processing with Matlab, School of Computer Science and Mathematics, Victoria University of Technology
- J. P. Serra (1982) Image analysis and mathematical morphology. Academic Press



UNIT 3: IMAGE RESTORATION, FEATURE DETECTION AND PATTERN MATCHING

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 A model for Image Degradation
 - 3.2 Noise
 - 3.2.1 Salt and pepper noise
 - 3.2.2 Gaussian noise
 - 3.2.3 Speckle noise
 - 3.2.4 Periodic noise
 - 3.3 Noise Reduction
 - 3.4 Feature Detection and Recognition (Object Detection Basics)
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

Image restoration is considered one of the major areas of image processing. 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. In this unit our emphasis shall be on the techniques for dealing with restoration, rather than with the degradations themselves, or the features of digital systems which give rise to image degradation. We shall also consider feature detection and pattern matching

2.0 OBJECTIVES

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

- Describe **image degradation model**
- Explain the concept of noises in image and their removal
- Explain the principles of object detection
- Describe the technique of object detection

3.0 MAIN CONTENT

3.1 A model for Image Degradation



Suppose we have an image $f(x, y)$ and a spatial filter $h(x, y)$ for which convolution with the image results in some form of degradation. Further, let us assume that $h(x, y)$ has a single line of ones, the result of the convolution will be a motion blur in the direction of the line. We represent this by the equation:

$$g(x, y) = f(x, y) * h(x, y)$$

for the resulting degraded image $g(x, y)$ where the symbol $*$ is used to represent a spatial filtering. Furthermore, we must consider noise, which can be modeled as an additive function to the convolution. Thus if we use $n(x, y)$ to represent the random error which may occur, we will thus have the degraded image expressed as:

$$g(x, y) = f(x, y) * h(x, y) + n(x, y)$$

We can perform the same operations in the frequency domain. To do this we replace the convolution by multiplication, while addition remains as addition, because of the linearity characteristics of the Fourier transform. This resulting expression becomes:

$$G(i, j) = F(i, j)H(i, j) + N(i, j)$$

This expression denotes general image degradation, where of course F , H , and N are the Fourier transformation of f , h , and n respectively.

Once we know the values of H and N we can always recover F by rewriting the above equation as

$$F(i, j) = (G(i, j) - N(i, j)) / H(i, j)$$

In real life, this approach may not be as practical as it appears in the mathematical expression. This is because, though, we may have some statistical information about the noise we may not know the value of $n(x, y)$ or $N(i, j)$ for all, or even any value. When we have the values of $H(i, j)$ which are close to, or equal to zero, we could have difficulties as implied in the formula.

3.2 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.

We shall examine some of the standard noise forms, and provide some details on the different approaches to eliminating or reducing their effects on the image.

3.2.1 Salt and pepper noise

This is also referred to as impulse noise, short noise, or binary noise. The degradation is caused by sharp, sudden disturbances in the image signal; it appears in an image as a randomly scattered white or black (or both) pixels over the image.



Figure 3.2a: An original image
Source [A. McAndrew, 2004]



Figure 3.2a: Effect of Salt and Pepper noise

3.2.2 Gaussian noise

Gaussian noise is an ideal case of white noise. It is caused by random fluctuation in the image signal. A very good example of this is by watching a television which is slightly off to a particular channel. Gaussian noise is white noise which is normally distributed. If the image is represented as I , and the Gaussian noise by N , then we can model a noisy image by simply adding the two represented by $I + N$.

3.2.3 Speckle noise

As can be seen in the mathematical representation for Gaussian noise, we modeled it by adding random values to an image. On the other hand, speckle noise is modeled by random values multiplied by pixel values; hence it is also called multiplicative noise. This is common with applications that involve radar devices. Figure 3.2c and Figure 3.2d depict the effects of Gaussian and Speckle noise respectively on an original image shown in Figure 3.2a.



Figure 3.2c: Effect of Gaussian Noise



Figure 3.2d: Effect of Speckle Noise

Source [A. McAndrew, 2004]

PDF Complete
 Your complimentary use period has ended.
 Thank you for using PDF Complete.
 Click Here to upgrade to Unlimited Pages and Expanded Features

3.2.4 Periodic noise

This type of noise is used to describe the effect of periodic disturbances on an image signal rather than a random disturbance. The effect appears as bars over an image. This is depicted in figure 3.2 e

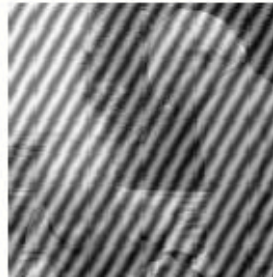


Figure 3.2e: Effect of Periodic noise

Source [A. McAndrew

,2004]

3.3 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 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 filters 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.

Median Filtering

The operations of median filtering make it most suitable for the removal of salt and pepper noise. From the knowledge of simple statistics, you recall that the median of a set is the middle value when they are sorted. If there are even numbers of values, the median is the mean of the middle two. A median filter is an example of a non-linear spatial filter. Using a 3 x 3 mask as an example, the output value is the median of the values in the mask. Let us examine table 3.3

50	55	52	
63	55	58	
61	60	57	



Table 3.3: 3 x 3

50	52	57	58	60
----	----	----	----	----

61 63 65 255 Here 60 is the median value

The operation of obtaining the median means that very large or very small values i.e. noisy value will end up at the top or bottom of the sorted list. Thus the median will in generally replace a noisy value with one closer to its surroundings

3.4 Feature Detection and Recognition (Object Detection Basics)

Object detection is an interesting topic in image processing. It is concerned with locating an object in a scene. To locate an object from a group of other object several questions to be answered. Amongst others, the user need to define what meant by an object? And specify an approach for the object's detection. The ultimate goal of computer vision is to design a system that would be capable analyzing a scene and determining which items in a scene were relevant objects. We shall now examine Pattern matching by using correlation technique to detect object in a scene.

3.4.1 Pattern Matching Using Correlation

The goal of the pattern matching technique is to find every instance of a specific object in the scene by applying a special template. A template in this context is an image of the object of interest. For example, figure 3.4a is the image of a spherical gas tank which we can use as a template.



Figure 3.4a: Spherical gas

As we can see, this template is a grouping of pixel values that correlate with the object of interest. If our task is to locate all the gas tanks in a factory depicted in figure 3.4b. To accomplish this location task, the gas tank mask is applied to the image in such a way that groupings of pixels that correlate with the template will be close to white while groups of pixels that do not correlate with the template will be close to black

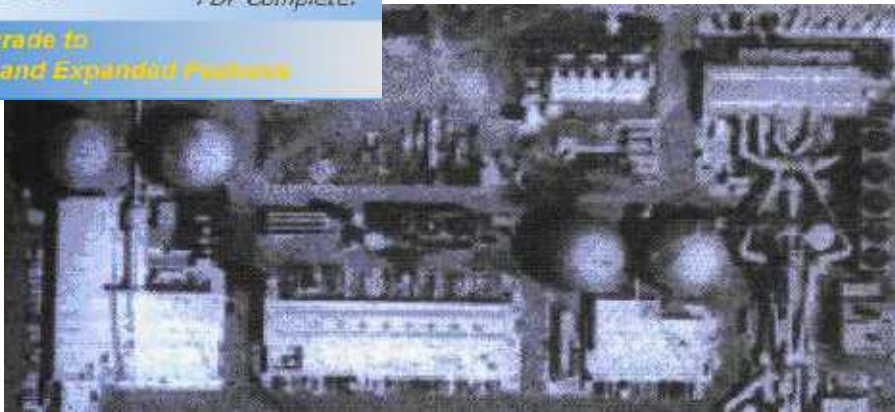
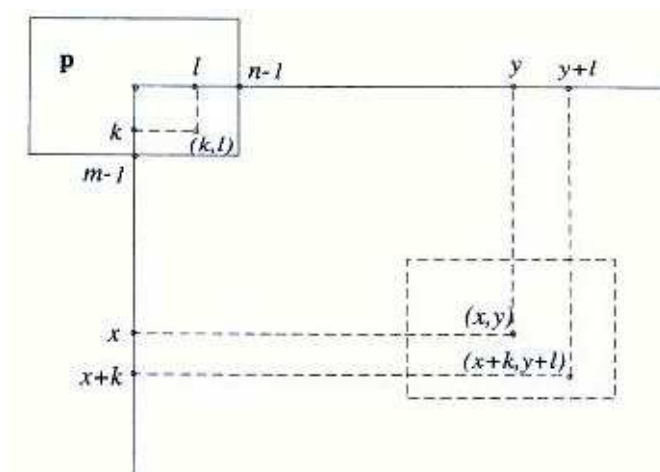


Figure 3.4b: Spherical gas

The figure below shows how the template is applied to the image.




The image algebra to accomplish this template application is given by the expression

$$c := a \cdot t$$

In this equation, c is the output image, a is the source image, and t is the template represented by pixel values p . t can be further defined as:

$$t_{(x,y)}(u,v) = \begin{cases} p(u-x, v-y) & \text{if } -(m-1) \leq u-x \leq m-1 \\ & \text{and } -(n-1) \leq v-y \leq n-1 \\ 0 & \text{otherwise.} \end{cases}$$

t


 Your complimentary use period has ended.
 Thank you for using PDF Complete.
 Click Here to [Purchase PDF Complete](#)
 Unlimited Pages and Expanded Features

By applying the above expressions, we obtain a factory picture which is depicted in Figure 3.4c

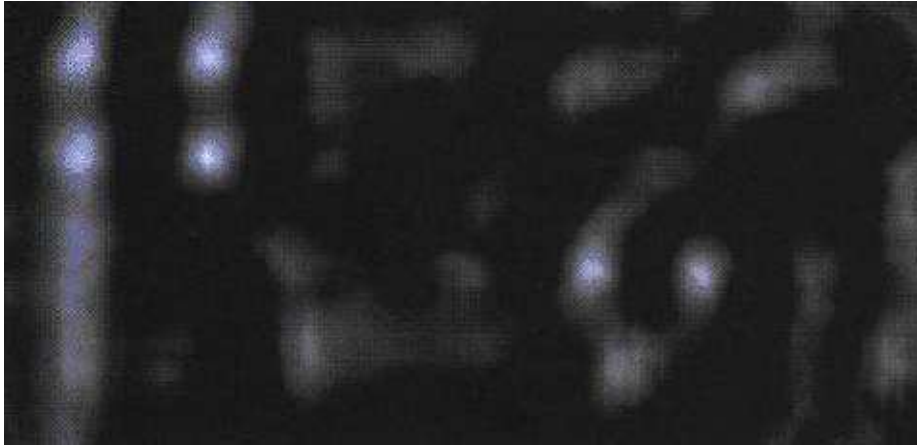


Figure 3.4c: Location of tanks

As you may observe, in the figure, the locations of the six tanks are in white, which goes to validate the method used here for object detection. However, there are other objects in the scene that are almost detected as gas tanks. While these objects do not have as strong a match as the tanks themselves (as can be seen by the relative whiteness of the pixels at those locations), they could be mistakenly identified as tanks. To make the tank locations clearer, we can apply thresholding on values of the pixel. This type of threshold turns all pixels that are not white enough to black. The result of this operation is shown below.



Figure 3.4d: Object after thresholding

The locations of the six tanks are now very clearly seen. This same type of process can be performed in the frequency domain.



3.4.2 Some limitations of Pattern Matching

- It requires an accurate image of the desired object as it is likely to appear the image.
- Changes in the object's orientation and size will adversely affect performance.
- If the template image was taken in different lighting than the source image, then the pixel values are less likely to line up, even for the object of interest.

4.0 CONCLUSION

Image restoration focuses on the removal or reduction of degradations which happened during the acquisition or transmission of image data. In digital image processing systems, the term noise refers to the degradation in the image signal, caused by external disturbance. Noise can be removed by designing an appropriate filter.

5.0 SUMMARY

In this unit, we covered image degradation model, identified sources of noise in digital signals and some kinds of noise. We also covered how noise can be removed or reduced by using appropriate filters. Finally we covered pattern matching using correlation.

6.0 TUTOR MARKED ASSIGNMENTS

- 1.0 Explain the term noise
- 2.0 Briefly explain the terms
 - a) Salt and pepper noise
 - b) Gaussian noise
 - c) Periodic noise
 - d) Speckle noise

7.0 REFERENCES / FURTHER READINGS

H. Benoit, (1996), Digital Television MPEG-1, MPEG-2 and Principles of the DVB systems, Focal Press, Linacre House, Jordan Hill, Oxford

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

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



MODULE 4

UNIT 1: IMAGE CODING WITH DCT AND WAVELET TECHNOLOGIES

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Image Coding with DCT
 - 3.2 Application of DCT
 - 3.3 Wavelets Compression
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Further Readings

1.0 INTRODUCTION

The discrete cosine transform (DCT) is widely applied in image compression. Alternative transforms such as Walsh, Hadamard, Karhunen-Loeve and wavelets have also been introduced. With wavelets the basic functions are not restricted to sine waves and can be quite different in form and thus characteristics. By using wavelets we get better image quality at low data rates and higher compression when compared to the DCT-based compression.

2.0 OBJECTIVES

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

- Explain image coding with DCT
- Discuss the application of wavelet technology in image coding

3.0 MAIN CONTENT

3.1 Image Coding with DCT

One of the most common data transformations used in video compression is the discrete cosine transformation (DCT). It is in CCITT H.261, JPEG, H.263 group of standards for video conferencing, and in both the MPEG1 and MPEG2 standards. The discrete cosine transform is a special case of the Fourier transform applied to discrete (sample) signals which decomposes a periodic signal into a series of sine and cosine harmonics functions. The signal can then be represented by a series of coefficients of each of these functions.

We provided some mathematical foundation in Module two. Under certain conditions, the DCT separates the signal into only one series of harmonic cosine functions in ~~with~~ the signal, which reduces by half the number of coefficients necessary to describe the signal compared to a Fourier transform.

In the case of picture, the original signal is a sampled bidimensional DCT (horizontal and vertical directions), which will transform the luminance (or chrominance) discrete value of a block $N \times N$ pixels into another block (or matrix) of $N \times N$ coefficient representing the amplitude of each of the cosine harmonic functions.

The transformed block consist value of horizontal and vertical frequencies. In the block coefficients of the horizontal axis represent increasing horizontal frequencies from left to right, and on the vertical axis they represent increasing vertical frequencies from top to bottom. The very first coefficient in the top left corner that is coordinates (0, 0) denotes horizontal and vertical frequencies, and is therefore called the DC coefficient, and the bottom right coefficient represent the highest spatial frequency component in the two direction.

For implementation purpose, a block size of 8×8 pixels (see figure 3.1) which the DCT transforms into a matrix of 8×8 (see figure 3.2) is generally used.

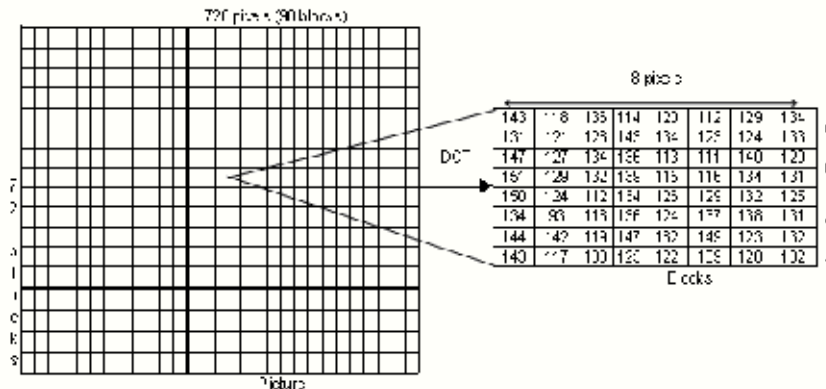


Figure 3.1 : Cutting of blocks of 8×8 pixels

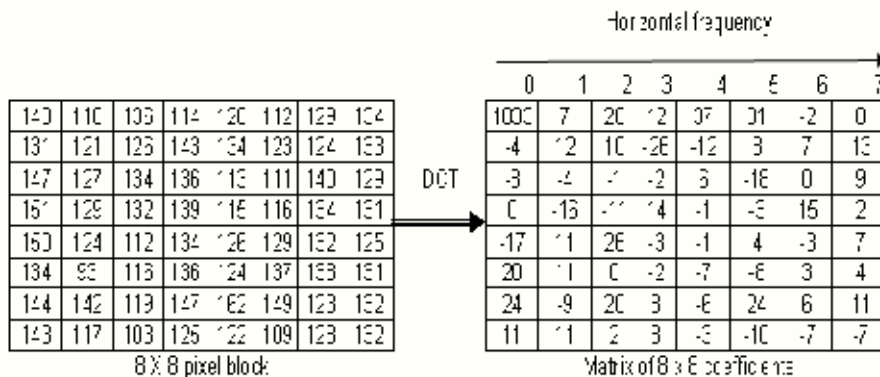


Figure 3.2: Transformation of a block of 8×8 pixels into a matrix of 8×8 coeff. Using DCT

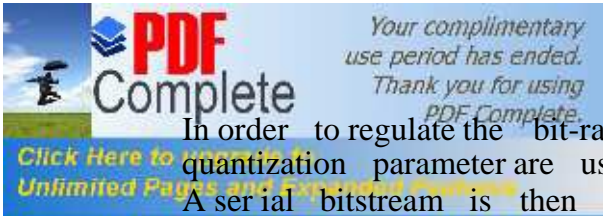


Figure 3.3: Result of thresholding and quantization

Figure 3.4: Zigzag reading of the coefficient of the matrix

The DCT has notable property of concentrating the energy of the block on a comparatively low number of coefficient situated in the top left corner of the matrix with coefficient that are decorated from each other. This property is taken advantage of in subsequent compression processes. The DCT transform process is reversible. Nevertheless, due to the nature of human vision (reduced sensitivity to high spatial frequencies), it is possible, without perceptible degradation of the picture quality, to eliminate the values below a certain threshold function of the frequency. The laminated values are replaced by 0 (an operation known as thresholding); this part of the process is obviously not reversible, as some data are thrown away.

The remaining coefficients are then quantized with an accuracy decreasing with the increasing spatial frequencies, this further reduces the quantity of information required to encode a block. This process is also not reversible, but it has little effect on the perceived picture quality. The thresholding/ quantization process is depicted in figure 3.3



In order to regulate the bit-rate required to transmit moving pictures, the thresholding and quantization parameter are used to dynamically regulate the bit-rate for moving pictures. A serial bitstream is then obtained by zig-zag reading of the coefficients, as shown in Fig 3.4. to allow a relatively long series of null coefficient to be obtained as quickly possible, in order to increase the efficiency of the compression algorithm (such as, Run length, Huffman) used.

3.2 Application of DCT

DCT is used for most image/ video standards. Two popular examples of such standards are JPEG (Joint photographic expert group) and MPEG (Motion picture expert group)

- a) The joint photographic expert group (JPEG) is a standard for compression of still pictures. The colour signal, red, green and blue are sampled and each color component is transformed by DCT in 8 x 8 pixel blocks. The DCT coefficients are quantized and encoded in a way that the more important lower components are represented by more bits than the higher-frequency coefficients. The coefficients are reordered by reading the DCT coefficient matrix in a zigzag fashion, and the data stream is Huffman coded.

- b) The motion picture expert group (MPEG) standard MPEG-1, for compression of full-motion pictures on digital storage media such as CD-ROM and digital versatile disc (DVD), with a bit transfer rate of about 1.5 Mbits/s is similar to JPEG. With this standard compression is achieved by allowing sampled frames to be split into blocks to be transformed using DCT in the same way as JPEG. The coefficients are then coded with either forward or backward prediction or a combination of both. The output from the predictive coding is then quantized using matrix of quantization steps. Since MPEG is more complicated than JPEG, it requires even more computing power. We shall discuss the MPEG standard in more details in the unit 3 of this module.

3.3 Wavelets Compression

Wavelets first appeared in the 1980s in geophysical work but had long spread to other areas such as mathematics, computer science and engineering. The basic idea behind its application is that many signals can be represented by combining many simpler waveforms called basic functions by using weighing factors known as coefficients. Similar to this is the Fourier transform which will decompose a signal into a set of sine waves of specific frequency. With wavelets the basic functions are not restricted to sine waves and can be quite different in form and thus characteristics.

Representation of a signal using sinusoids is very effective for stationary signals, which are statistically predictable and are time-invariant in nature. Wavelet representation is found to be very effective for nonstationary signals, which are not statistically predictable and time-varying in nature. Variation of intensity to form edges is a very important visual characteristic of an image. From signal theoretic perspective, discontinuities of intensities

occur at the edges in any image and hence it can be prominently visualized by the human eye. The time and frequency localization property of wavelets makes it attractive for analysis of images because of discontinuities at the edges.

Wavelets are functions defined over a finite interval and having an average value of zero. The basic idea of the wavelet transform is to represent any arbitrary function $f(t)$ as superposition of a set of such wavelets or basis functions. These basis functions also referred to as baby wavelets, are obtained from a single prototype wavelet called the mother wavelet, by dilations or contractions (scaling) and translations (shifts).

If the mother wavelet is denoted by $\psi(t)$ then other wavelets $\psi_a(t-b)$ for $a > 0$ and a real number b can be represented as:

$$\psi_a(t-b) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right)$$

where a and b represent the parameters for dilations and translations in the time domain. The parameter a causes contraction in time domain when $a < 1$ and expansion when $a > 1$. Figure 3.5 is used to illustrate a mother wavelet and its contraction and dilation.

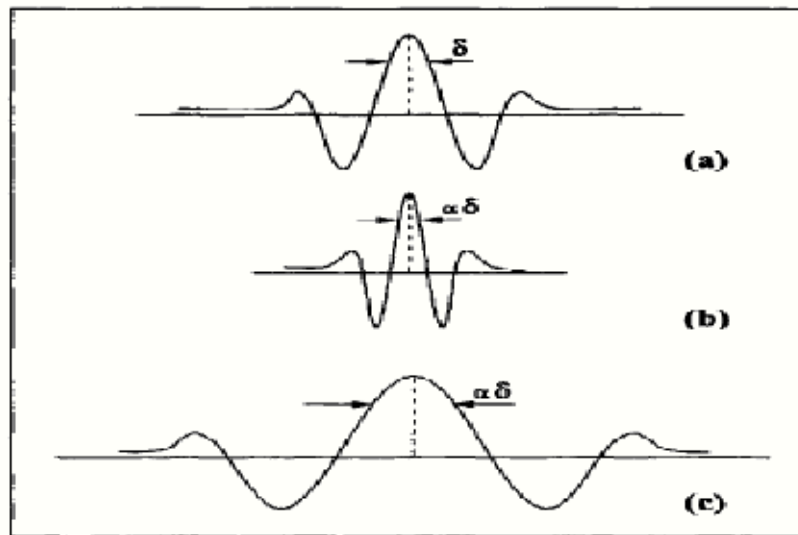


Figure 3.5: (a) Mother wavelet $\psi(t)$ (b) $\psi_a(t-b)$ for $a < 1$ (c) $\psi_a(t-b)$ for $a > 1$

By using wavelets we get better image quality at low data rates and higher compression when compared to the DCT-based compression.

Wavelets compression is in some aspects very similar to the DCT compression techniques that are used with MPEG and H.320 video conferencing. It transforms the



data to make it easier to decide which data can be lost for the minimum impact on video quality and then uses coding techniques to compress the data. The big difference though is that the DCT algorithm is not used and is replaced with wavelet transform. Using wavelets gives better image or video quality at low data rates and higher compression when compared to the DCT- based compression.

Compression schemes based on DCT requires that the input image needs to be "block ed", thus correlation across the block boundaries is not eliminated. This results in noticeable and annoying "blocking artifacts" particularly at low bit rates as shown in figure 3.6(a). Blocking artifacts also called macroblocking are distortion that appears in compressed image / video material as abnormally large pixel blocks. It occurs when the encoder cannot keep up with the allocated bandwidth. It is especially visible with fast motion sequences or quick scene changes.

Video uses lossy compression, and the higher the compression rate, the more content is removed. At decompression, the output of certain decoded blocks makes surrounding pixels appear averaged together and look like larger blocks. As TVs get larger, blocking and other artifacts become more noticeable.

Since there is no need to block the input image and its basis functions have variable length, wavelet coding schemes at higher compression avoid blocking artifacts. Wavelet-based coding is more robust under transmission and decoding errors, and also facilitates progressive transmission of images. In addition, they are better matched to the HVS characteristics. This is because of their inherent multiresolution nature. Wavelet coding has earlier been discussed in module two. Wavelet coding schemes are especially suitable for applications where scalability and tolerable degradation are important such as in Subband Coding.

The fundamental concept behind Subband Coding (SBC) is to split up the frequency band of a signal and then to code each subband using a coder and bit rate accurately matched to the statistics of the band. SBC has been used extensively first in speech coding and in image coding because of its inherent advantages namely variable bit assignment among the subbands as well as coding error confinement within the subbands.



Figure 3.6(a)



Figure 3.6(b)

Figure 3.6(a) Original Lena Image, and (b) Reconstructed Lena with DC component only, to show blocking artifacts.

Self Assessment Test

- 1) What are blocking artifacts?
- 2) What are wavelets?

4.0 CONCLUSION

The basic idea of transform compression is to extract appropriate statistical properties, for instance Fourier coefficients, of an image and let the most significant of these properties represent the image. The image is then reconstructed (decomposed) using an inverse transform. Often it is convenient to express the transform coefficient as a matrix. Two popular transforms that can be used are discrete cosine transform and discrete wavelet transform.

While the DCT-based image coders perform very well at moderate bit rates, at higher compression ratios, image quality degrades because of the artifacts resulting from the



block-based DCT scheme. Wavelet-based coding on the other hand provides substantial improvement in picture quality at low bit rates because of overlapping basis functions and better energy compaction property of wavelet transform.

5.0 SUMMARY

Image and video compression have become an integrated part of today's digital multimedia applications. In this unit, we covered some of the more sophisticated techniques that take advantage of the statistics of the wavelet coefficients. Wavelet theory is also a form of mathematical transformation, similar to the Fourier transform in that it takes a signal in time domain, and represents it in frequency domain. Wavelet functions are distinguished from other transformations in that they not only dissect signals into their component frequencies, they also vary the scale at which the component frequencies are analyzed. Therefore wavelets, as component pieces used to analyze a signal, are limited in space. The ability to vary the scale of the function addresses different frequencies also makes wavelets better suited to signals with spikes or discontinuities than traditional transformations such as the Fourier transform. Applications of wavelet theory include JPEG2000 which is based on wavelet compression algorithms.

6.0 TUTOR MARKED ASSIGNMENTS

- 1.0 What are the advantages of wavelets over DCT?
- 2.0 Briefly describe to areas of application of DCT
- 3.0 What are wavelets?

7.0 REFERENCES / FURTHER READINGS

Herve Benoit, (1996), Digital Television MPEG-1, MPEG-2 and Principles of the DVB systems, Focal Press, Linacre House, Jordan Hill, Oxford

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

Dag Stranded by, William Walker(2004), Digital Signal Processing and Applications, Newnes, An imprint of Elsevier, Jordan Hill, Oxford



UNIT 2: VIDEO CODING WITH MOTION ESTIMATION

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 What is a Digital Video?
 - 3.2 Video Coding of MPEG-1
 - 3.2.1 The different types of MPEG Frame
 - 3.2.2 Decomposition of an MPEG video sequence in layers
 - 3.2.3 Prediction, Motion estimation and compensation
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References and Future Reading

1.0 INTRODUCTION

Digital video has become mainstream and is being used in a wide range of applications. DVD, digital TV, HDTV, video telephony, and teleconferencing are some of the applications. These digital

video applications are feasible because of the advances in computing and communication technologies as well as efficient video compression algorithms. Most of the video compression standards are based on a set of principles that reduce the redundancy in digital video. In this unit we explained motion estimation by discussing video coding techniques with MPEG-1.

2.0 OBJECTIVES

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

- Explain the meaning of motion estimation and compensation
- Explain the different types of frame
- Describe the principles behind MPEG-1 Video Coding

3.0 MAIN CONTENT

3.1 What is a Digital Video ?

We introduced the principles of video compression in module two. Digital video is essentially a sequence of pictures displayed overtime. Each picture of a digital video sequence is a 2D projection of the 3D world. Digital video thus is captured as a series of digital pictures or sampled in space and time from an analog video signal. A frame of digital video or a picture can be seen as a 2D array of pixels. We shall proceed to discuss video compression with respect to MPEG.



3.2 Video Coding of MPEG-1

The main objective of MPEG-1 was to reach a medium quality video with a constant total bit-rate of 1.5 Mbps for storing video and audio on CD-ROM. The video part uses 1.1 Mbps, while the remaining 350 kb/s is used by audio and other details required by the system. Nevertheless the MPEG-1 specification is such that it allows different parameters to be chosen depending on the compromise between encoder complexity, compression rate and quality.

The MPEG-1 standard allows the user to set a whole range of parameter which control the image size, target bitstream rate, etc. However, in order to assure interoperability, a constrained parameter set has been defined and most MPEG-1 decoders conform to this. The implication of this is that, for parameters outside this range, there is less likelihood of compatibility between different encoders and decoders.

- Horizontal resolution less or equal 760 pixels
- Vertical resolution less or equal 576 pixels
- Macroblock per picture less or equal 396
- Macroblocks to be processed per second less or equal 99000
- Frames per second less or equal 30
- Bitstream bandwidth less or equal 1.86 Mbps
- Decoder buffer size less or equal 376832 pixels

These techniques, referred to as prediction with movement compensation, consist of deducing most of the pictures of a sequence from preceding and even subsequent pictures, with minimum of additional information representing the differences between pictures. This usually requires an additional task of movement estimator in the MPEG encoder. This happens to be the greatest task and goes a long way in determining the encoder's performance. Fortunately, this function is not required in the decoder.

When dealing with video, we are actually talking about moving pictures in which decoding has to be accomplished in real time i.e. it has to be done in an acceptable and constant processing delay. This can only be accomplished, for the time being at least through, some specialized hardware. The coding, is usually a more complex process and can be done in more than one pass for applications where real time is not required but where quality is paramount (such as in engraving of disk s); real time will, however, be required for many applications, such as live video transmissions.

In the practical realization of the encoder, a trade-off among speed, compression rate, complexity and picture quality is necessary. In addition, synchronization time and random access time to a sequence have to be maintained within an acceptable limit (not exceeding 0.5 s), which restricts the maximum number of pictures that can be dependent on the first picture to between 10 and 12 for a system operating at 25 fps (frames per second)

3.2.1 The different types of MPEG Frames

MPEG defines three types of pictures which are arranged as shown in Figure 3.2.1a

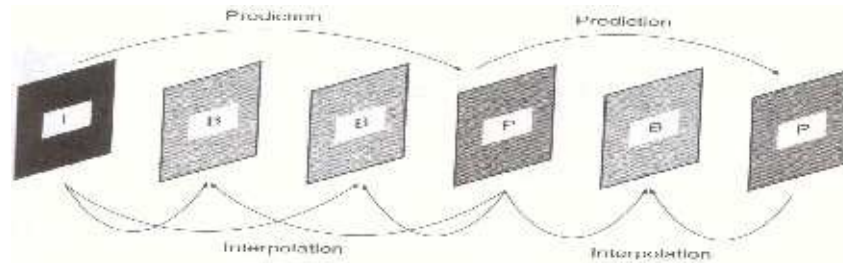


Figure 3.2.1a: Arrangement of frames in MPEG

- Intra (I) pictures are coded without reference to other pictures, in a very similar manner to JPEG, which means that they contain all the information necessary for their reconstruction by the decoder; for this reason, they are the essential entry point for access to a video sequence. The compression rate of I pictures is relatively low, and is comparable to a JPEG coded picture of a similar resolution.
- Predicted (P) pictures are coded from the preceding I or P picture, using the technique of motion compensated prediction. P pictures can be used as the basis for next predicted pictures, but due to inherent imperfection of the motion compensation technique, the number of P pictures between consecutive I pictures must of a necessity be limited. It follows logically that the compression rate of P pictures is significantly higher than that of I picture.
- Bidirectional or bidirectionally predicted (B) pictures are coded by bidirectional interpolation between I and P pictures preceding and following them. As they are not used for coding subsequent pictures, B pictures do not propagate coding errors. B pictures offer the highest compression rate.

Depending on the complexity of the encoder used, it is possible to encode I only, I and P, or I, P and B pictures, with very different results with regards to compression rate and random access resolution, and also with regard to encoding time and perceived quality. Two parameters; M and N, describe the succession of I, P and B frame as depicted in figure 3.2.1b

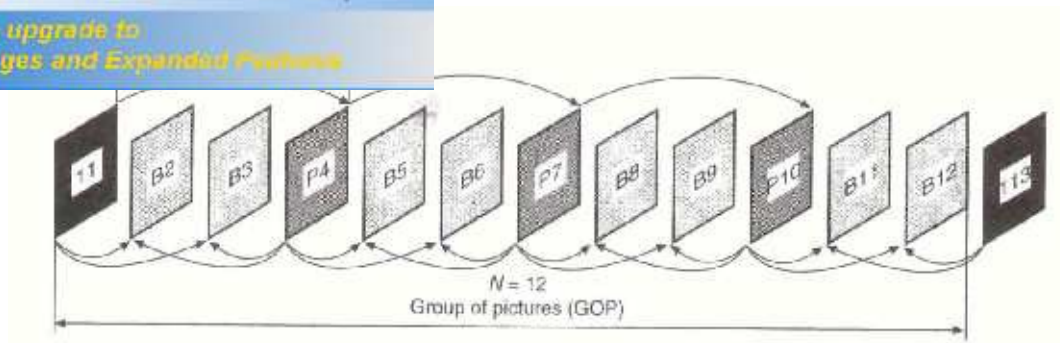


Figure 3.2.1b Example of an MPEG group of pictures for $M=3$ and $N=12$

- M is the distance (in number of pictures) between two successive P pictures;
- N is the distance between two successive I frames, defining a group of pictures (GOP).

The parameters generally used are $M=3$, and $N=12$, in order to obtain a satisfactory video quality with an acceptable random access time (less than 0.5s) within a bit-rate of 1.15 Mbps. With these parameters, a video sequence is made up as follows: 1/12 (8.33%) of its pictures are I pictures, 1/4 (25%) are P pictures, and 2/3 (66.67%) are B pictures. The resultant compression rate is maximized by the fact that the most frequent pictures have the highest compression rate.

Re-Ordering of the Picture / frame

It is obvious that the sequence of the picture after decoding has to be in the same order as the original sequence before encoding. With the parameters earlier stated ($M=3$, $N=12$), the difference between the original picture number and its coded type is as follows:

1(I) 2(B) 3(B) 4(P) 5(B) 6(B) 7(P) 8(B) 9(B) 10(P) 11(B) 12(B) 13(I)

However, in order to encode or decode bidirectional picture, both the encoder and the decoder will need the I or P preceding picture and the I or P subsequent picture. This requires re-ordering of the original picture sequence such that the decoder and the encoder have at their disposal the required I and (or) P picture before the B picture is processed. The re-ordering thus gives the following sequence:

1(I), 4(P), 2(B), 3(B), 7(P), 5(B), 6(B), 10(P), 8(B), 9(B), 13(I), 11(B), 12(B) ..

The increase in compression rate introduced by the B picture has to be compensated for by an increase in encoding delay and in the memory size required for both encoding and decoding (one extra picture to store).

3.2.2 Decomposition of an MPEG video sequence in layers

MPEG defines a hierarchy of layers within a video sequence, as shown in Figure 3.2.2. Each of these layers has specific functions(s) in the MPEG process. Starting from the top level, the successive layers are:

- Sequence this is the highest layer defining the context valid for the whole sequence (basic video parameters, etc)
- Group of Pictures (GOP) - this layer determines the random access to the sequence, which always starts with an I picture. In the example in Figure 3.2.1b the GOP is made up of 12 pictures.

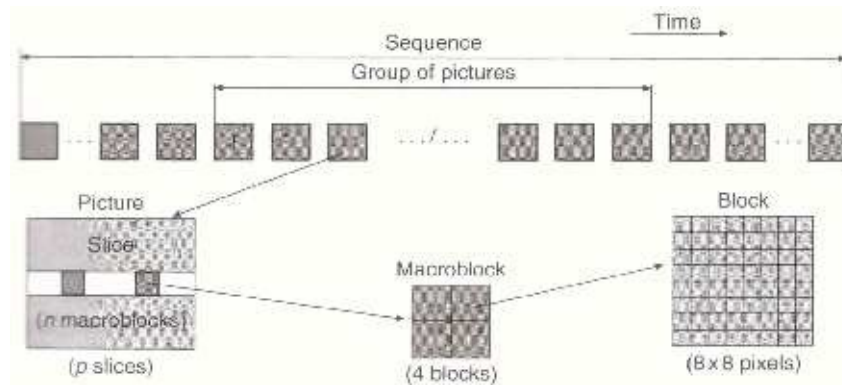


Figure 3.2.2a: Hierarchy of the MPEG VIDEO layer

- Picture this is also referred to as frame. It is the elementary display unit, which can be one of the three types (I, P, or B) discussed earlier.
- Slice this is the layer for intra frame addressing, and (re)synchronization, for instance for error recovery. It consists of a suite of contiguous macroblocks. Theoretically, the size of a slice can range from one macroblock to the whole picture, but in most cases it is made up of a complete row of macroblocks
- Macroblocks this is the layers used for movement estimation / compensation. It consist of a size of 16 x 16 pixels and made up of four blocks of luminance and two blocks of chrominance. See figure 3.2.2b for more details.
- Block Just as in JPEG, a picture is made up of blocks of 8 x 8 pixels. It is the layer where the DCT takes place.

$$\begin{aligned} \text{One macroblock} &= 16 \times 16 \text{ Y samples (4 blocks)} \\ &\quad + 8 \times 8 \text{ Cb samples (1 block)} \\ &\quad + 8 \times 8 \text{ Cr samples (1 block)} \end{aligned}$$

0 =luminance sample, * Chrominance sample

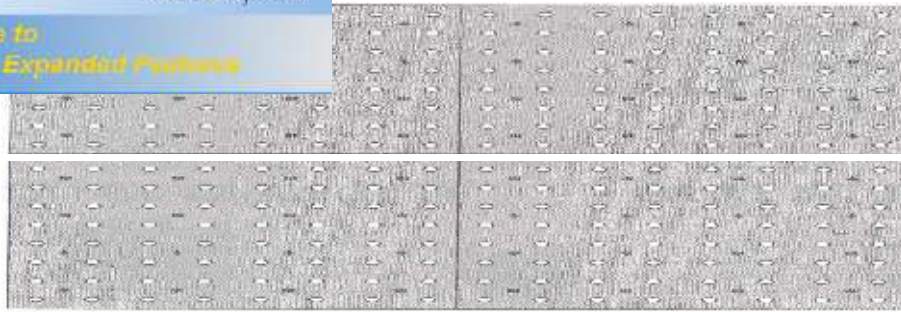
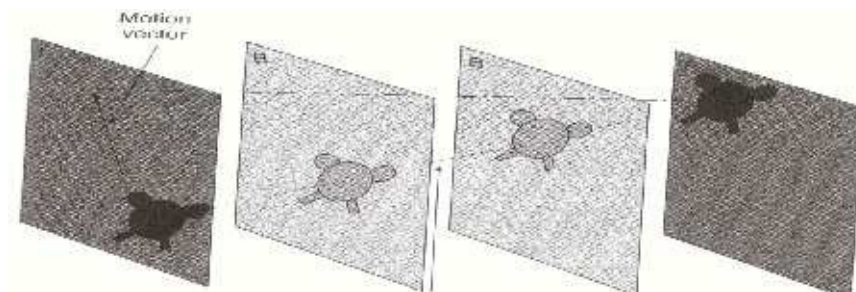


Figure 3.2.2b: Composition of a 4:2:0 macroblock (0=Y sample, * = Cb and Cr Sample)
 [Source H. Benoit, 1997]

3.2.3 Prediction, Motion estimation and compensation

In section 3.2.1, we discussed that P and B pictures were predicted from the preceding and / or subsequent pictures. We shall elaborate more on this to enhance your understanding.

When dealing with videos and animations, the moving objects lead to differences between corresponding zones of consecutive pictures, so that there is no clear correlation between these two zones. Motion estimation analyzes the video frames and calculates where objects are moved to. Instead of transmitting all the data needed to represent the new frame, only the information (i.e the vector or new position) needed to move the object transmitted. This is done at the macroblock level (16x 16 pixels) by allowing macroblock of the current picture within a small search window from the previous picture, and comparing it to all possible macroblocks of the window in order to find the one that is most similar. The difference in position of the two matching macroblocks gives a motion vector which will be applied to all three component of the macroblock (Y, Cb, Cr). See figure 3.2.3 for details.



Interpolation

Figure 3.2.3: A simplified illustration of motion compensation
 [Source H. Benoit, 1997]



Block matching is usually not adequate for making comparison in the temporary distance between a P picture and an I picture or two P picture as the motion vector can be relatively high amplitude. Thus the difference (or prediction error) between the actual block to be encoded and the matching block has to be calculated, and encoded in a similar way to the block of the I pictures (DCT, quantization, RLC/VLC). This process is called motion compensation.

In the case of B frames /pictures, motion vector are computed by temporal interpolation of the vector of the next P picture in three different ways (forward, backward and bidirectional); the smallest prediction error value is retained, and the error is encoded in the same way as for P pictures. Only the macroblocks which are different from the picture(s) used for prediction would need to be encoded. This process no doubt, helps to substantially reduce the amount of information required for coding B and P pictures. As the size of the moving objects is generally bigger than a macroblock, there is correlation between the motion vectors of consecutive blocks, and a differential coding method (DPCM) is used to encode the vectors, thus reducing the number of bits required.

In case, the prediction does not yield a desired result (for example, in the case of moving camera where completely new zones appear in the picture), the corresponding parts of the picture are Intra coded in the same way as for I pictures. Figure 3.2.3a and

Figure 3.2.3b depict the schematic diagrams for an MPEG encoder and decoder respectively. As will be observed, the decoder does not perform motion estimation and so is not as complex as an encoder. This is one of the main objectives of the standard.

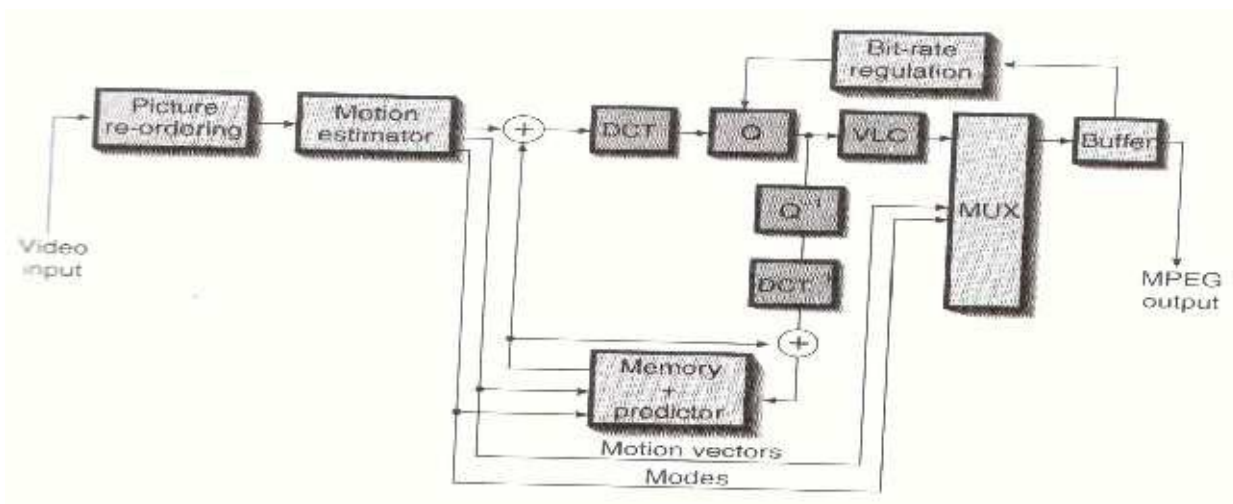


Figure 3.2.3a: Schematic diagram of the MPEG encoder

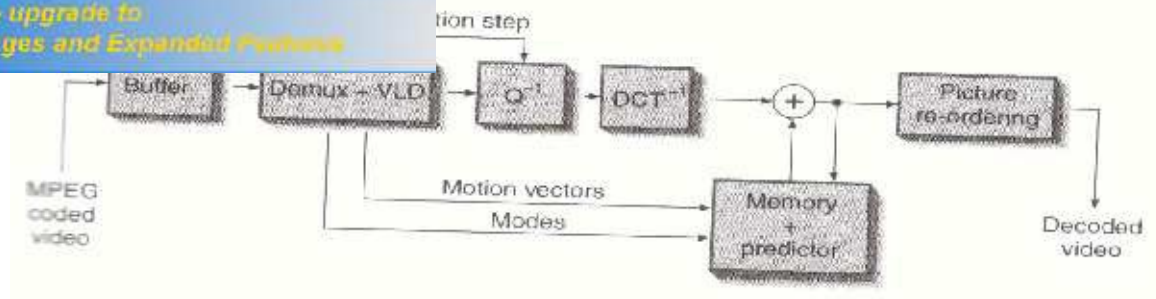


Figure 3.2.3b: Schematic diagram of the MPEG decoder

4.0 CONCLUSION

Video compression typically operates on square-shaped groups of neighboring pixels, often called macroblocks. These pixel groups or blocks of pixels are compared from one frame to the next and the video compression codec sends only the differences within those blocks. This works extremely well if the video has no motion. A still frame of text, for example, can be repeated with very little transmitted data. In areas of video with more motion, more pixels change from one frame to the next. When more pixels change, the video compression scheme must send more data to keep up with the larger number of pixels that are changing.

5.0 SUMMARY

In this unit, we covered video coding with motion estimation. Motion estimation involves comparing small segments of two consecutive frames for differences and, should a difference be detected, a search is carried out to determine to which neighboring segment the original segment has moved. In order to minimize the time for each search, the search region is limited to just a few neighboring segments. The MPEG technology employs this technique for data compression. Also in this unit, we covered motion compensation and prediction. The different types of frame and the MPEG video sequence layers were discussed.

6.0 TUTOR MARKED ASSIGNMENTS

- 1.0 Describe the different types of MPEG frames
- 2.0 Explain the term Motion estimation
- 3.0 With the aid of a label diagram, describe the MPEG the hierarchy of layers within MPEG video sequence

7.0 REFERENCES / FURTHER READINGS

- H. Benoit, (1997), Digital Television MPEG-1, MPEG-2 and Principles of the DVB systems, Focal Press, Linacre House, Jordan Hill, Oxford
- J. D. Gibson (Eds) (2001), Multimedia Communications: Directions and Innovations, Academic Press, San-Diego, USA



UNIT 3: IMAGE / VIDEO COMPRESSION STANDARDS JPEG, MPEG AND H.263.

CONTENTS

- 1.0 Introduction
- 2.0 Objectives
- 3.0 Main Content
 - 3.1 Image / Video Compression Standard
 - 3.2 JPEG/ JPEG 2000
 - 3.2 MPEG (Moving Picture Experts Group)
 - 3.3.1 MPEG-1
 - 3.3.2 MPEG-2
 - 3.3.3 MPEG-4
 - 3.3.4 MPEG-7
 - 3.3.5 MPEG-21
 - 3.3.6 Other MPEG Standards
 - 3.4 H.263
- 4.0 Conclusion
- 5.0 Summary
- 6.0 Tutor Marked Assignment
- 7.0 References / Future Readings

1.0 INTRODUCTION

Image and video compression have become an integrated part of today's digital communications systems such as facsimile, videoconferencing, image archival systems, DVD, movie and video distribution, graphics and film industry, etc. As new applications continue to emerge it is necessary to define standards for common data storage, compressions, retrieval, and transmission in these systems. This is to allow for perfect interoperability of data exchange.

The two main international bodies in the multimedia compression area are the International Organization for Standardization (ISO) and International Telecommunications Union Telecommunications Sector (ITU-T) formerly known as Comité Consultatif International Téléphonique et Télégraphique (CCITT). In the following section we shall discuss the standards these two organizations support.

2.0 OBJECTIVES

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

- Discuss an overview of different image / video standards
- Explain important features of some common standards used in multimedia applications
- Highlight the areas of applications of the standards



3.0 MAIN CONTENT

3.1 Image / Video Compression Standard

The Joint Photographic Experts Group' or JPEG standard established by ISO (International Standards Organization) and IEC (International Electro-Technical Commission) is the standard for still image compression. MPEG (Moving Picture Expert Group) is the standard in ISO for digital compression system to handle pictures (video) and associated audio.

3.2 JPEG/ JPEG 2000

JPEG is the acronym for Joint Photographic Experts Group. It is the first international image compression standard for continuous-tone still images, including both gray scale and color images. The goal of this standard is to support a variety of applications for compression of continuous-tone still images

- (i) of different sizes,
- (ii) in any color space,
- (iii) in order to achieve compression performance at or near the state of the art,
- (iv) with user-adjustable compression ratios, and
- (v) with very good to excellent reconstructed quality.

Another goal of this standard is that it should have manageable computational complexity for widespread practical implementation. JPEG defines the following four modes of operation

1. Sequential Lossless Mode: Compresses images in a single scan, and the decoded image is an exact replica of the original image.
2. Sequential DCT-Based Mode: Compresses images in a single scan using DCT-based lossy compression techniques. This gives a decoded image that is not an exact replica but an approximation of the original image.
3. Progressive DCT-Based Mode: This allows for the compression of images in multiple scans and also decompresses the image in multiple scans, with each successive scan producing a better quality image.
4. Hierarchical Mode: This allows for the compression of images at multiple resolutions for display on different devices.

The three DCT-based modes (2, 3, and 4) in JPEG provide lossy compression, because the precision limitation to digitally compute DCT (and its inverse) and the quantization process introduce distortion in the reconstructed image. For sequential lossless mode of



compression, predictive coding (DPCM) is used instead of the DCT-based transformation and also there is no quantization involved.

The commonly used algorithm for image compression in the sequential DCT-based mode of the standard is called baseline JPEG. Great research efforts have been devoted to image compression since the establishment of the JPEG standard in 1992. The success of JPEG necessitated further research efforts into an enhanced standard called JPEG-2000 for coding of still images. JPEG 2000 is a wavelet-based image compression standard. It was created by the Joint Photographic Experts Group committee in the year 2000 with the intention of superseding their original discrete cosine transform-based JPEG standard (created 1992). The standard include many modern features such as improved low bit-rate compression performance, lossless and lossy compression, continuous-tone and bi-level compression, compression of large images, single decompression architecture, transmission in noisy environments including robustness to bit-errors, progressive transmission by pixel accuracy and resolution, content-based description, and protective image security. The standardized filename extension is .jp2 for ISO/IEC 15444-1 conforming files and .jpx for the extended part-2 specifications, published as ISO/IEC 15444-2. The registered MIME types are defined in RFC 3745. For ISO/IEC 15444-1 it is image/jp2.

In addition, while there is a modest increase in compression performance of JPEG2000 compared to JPEG, another advantages offered by JPEG2000 is the significant flexibility of the codestream. The codestream obtained after compression of an image with JPEG2000 is scalable in nature, meaning that it can be decoded in a number of ways; for instance, by truncating the codestream at any point, one may obtain a representation of the image at a lower resolution. By ordering the codestream in various ways, applications can achieve significant performance increases. However, as a consequence of this flexibility, JPEG2000 requires encoders/decoders that are complex and computationally demanding. Another difference, in comparison with JPEG, is in terms of visual artifacts: JPEG 2000 produces ringing artifacts, manifested as blur and rings near edges in the image, while JPEG produces ringing artifacts and 'blocking' artifacts, due to its 8x8 blocks.

JPEG 2000 has been published as an ISO standard, ISO/IEC 15444. As of 2009, JPEG 2000 is not widely supported in web browsers, and hence is not generally used on the World Wide Web.

3.3 MPEG (Moving Picture Experts Group)

The Moving Picture Experts Group (MPEG) was formed by the ISO to set standards for audio and video compression and transmission. The MPEG standards consist of different Parts. Each part covers a certain aspect of the whole specification. The standards also specify Profiles and Levels. Profiles are intended to define a set of tools that are available, and Levels define the range of appropriate values for the properties associated with them. Some of the approved MPEG standards were revised by later amendments and



or new editions. MPEG has standardized the following compression formats and ancillary standards

3.3.1 MPEG-1(officially known as ISO 11172) is the first generation of digital compression standards for video and two-channel stereo audio to achieve bit-rate of about 1.5 Mbps (Mega bits per seconds) for storage in CD-ROMs. MPEG-1 was standardized in 1994. This standard was based on CD-ROM video applications, and is a popular standard for video on the Internet, transmitted as .mpg files. In addition, level 3 of MPEG-1 is the most popular standard for digital compression of audio-known as MP3. MPEG-1 is the standard of compression for VideoCD, the most popular video distribution format throughout much of Asia.

3.3.2 MPEG -2

This is the standard on which Digital Television set top boxes and DVD compression based. It is based on MPEG-1, but designed for the compression and transmission of digital broadcast television. The most significant enhancement from MPEG-1 is its ability to efficiently compress interlaced video. MPEG-2 scales well to HDTV resolution and bit rates, obviating the need for an MPEG-3. MPEG-2 standard was considerably broader in scope and of wider appeal supporting interlacing and high definition. MPEG-2 is considered important because it has been chosen as the compression scheme for over-the-air digital television ATSC, DVB and ISDB, digital satellite TV services like Dish Network, digital cable television signals, SVCD, and DVD. This is defined in a series of documents which are all subset of ISO Recommendation 13818. It is intended for the recording and transmission of studio-quality audio and video.

3.3.3 MPEG-4

MPEG-4 standard was defined to meet newer challenges of the object-based video coding suitable for multimedia applications. MPEG-4 is based on object-based compression, similar in nature to the Virtual Reality Modeling Language. Individual objects within a scene are tracked separately and compressed together to create an MPEG4 file. It allows developers to control objects independently in a scene, and therefore introduce interactivity. Initially, this standard was concerned with similar range of applications to those of H.263, each running over low bit rate channels ranging from 4.8 to 64 kps. Later its scope was expanded to embrace a wide range of interactive multimedia applications over the Internet and the various types of entertainment networks. The main application domain of the MPEG-4 standard is in relation to the audio and video associated with interactive multimedia applications over the Internet and the various types of entertainment network. The standard contains features to enable a user not only to passively access a video sequence (or complete video) using, for example start/stop/pause command but also to access and manipulate the individual elements that make up each scene within the sequence / video. If the accessed video is a computer-generated cartoon, for example, the user may be given the capability by the creator of the video to reposition, delete, or alter the movement of the individual characters within



a scene. In addition, because of its high coding efficiency with scenes such as those associated with video telephony, the standard is also used for this type of application running over low bit rate networks such as wireless and PSTNs. MPEG-4 uses further coding tools with additional complexity to achieve higher compression factors than MPEG-2. In addition to more efficient coding of video, MPEG-4 moves closer to computer graphics applications. In more complex profiles, the MPEG-4 decoder effectively becomes a rendering processor and the compressed bitstream describes three-dimensional shapes and surface texture. MPEG-4 also provides Intellectual Property Management and Protection (IPMP) which provides the facility to use proprietary technologies to manage and protect content like digital rights management. Several new higher-efficiency video standards (newer than MPEG-2 Video) are included (an alternative to MPEG-2 Video).

3.3.4 MPEG-7

There was a popular misconception that MPEG-7 was going to be another new video compression standard. The fact is that MPEG-7 does not define any new video compression standard. It deals with the file format and metadata description of the compressed video in order to define a standard for description of various types of multimedia coded with the standard codecs. The main objective of MPEG-7 is to serve the need of audiovisual content-based retrieval (or audiovisual object retrieval) in applications such as digital libraries. Nevertheless, it is also applicable to any multimedia applications involving the generation and usage of multimedia data. MPEG-7 became an International Standard in September 2001 with the formal name Multimedia Content Description Interface. MPEG-7 supports a variety of multimedia applications. Its data may include still pictures, graphics, 3D models, audio, speech, video, and composition information (how to combine these elements). These MPEG-7 data elements can be represented in textual format, or binary format, or both.

3.3.5 MPEG-21

The MPEG-21 standard established in 2001, from the Moving Picture Experts Group, aims at defining an open framework for multimedia applications. MPEG-21 is ratified in the standards ISO/IEC 21000 - Multimedia framework (MPEG-21)

Specifically, MPEG-21 defines a "Rights Expression Language" standard as means of sharing digital rights/permissions/restrictions for digital content from content creator to content consumer. As an XML-based standard, MPEG-21 is designed to communicate machine-readable license information and do so in a "ubiquitous, unambiguous and secure" manner.

MPEG-21 is based on two essential concepts: the definition of a fundamental unit of distribution and transaction, which is the Digital Item, and the concept of users interacting with them. Digital Items can be considered the kernel of the Multimedia Framework and the users can be considered as who interact with them inside the Multimedia Framework. At its most basic level, MPEG-21 provides a framework in which one user interacts with another one, and the object of that interaction is Digital Item. Due to that, we could say that the main objective of the MPEG-21 is to define the



technology needed to support users to exchange, access, consume, trade or manipulate Digital Items in an efficient and transparent way.

3.3.6 Other MPEG Standards

Relatively more recently than other standards above, MPEG has started following international standards; each of the standards holds multiple MPEG technologies for a way of application. For example:

- MPEG-A (2007): Multimedia application format (MPEG-A). (ISO/IEC 23000) (e.g. Purpose for multimedia application formats, MPEG music p layer application format, MPEG photo player application format and others)
- MPEG-B (2006): MPEG systems technologies. (ISO/IEC 23001) (e.g. Binary MPEG format for XML, Fragment Request Units, Bitstream Syntax Description Language (BSDL) and others)
- MPEG-C (2006): MPEG video technologies. (ISO/IEC 23002) (e.g. Accuracy requirements for implementation of integer-output 8x8 inverse discrete cosine transform and others)
- MPEG-D (2007): MPEG audio technologies. (ISO/IEC 23003) (e.g. MPEG Surround and two parts under development: SAOC-Spatial Audio Object Coding and USAC-Unified Speech and Audio Coding)
- MPEG-E (2007): Multimedia Middleware. (ISO/IEC 23004) (a.k.a. M3W) (e.g. Architecture, Multimedia application programming interface (API), Component model and others)
- Supplemental media technologies (2008). (ISO/IEC 29116) Part 1: Media streaming application format protocols will be revised in MPEG-M Part 4 - MPEG extensible middleware (MXM) protocols.
- MPEG-V (under development at the time of writing this study material): Media context and control. (ISO/IEC CD 23005) (a.k.a. Information exchange with Virtual Worlds) (e.g. Avatar characteristics, Sensor information, Architecture and others)
- MPEG-M (under development at the time of writing this study material): MPEG eXtensible Middleware (MXM). (ISO/IEC FCD 23006) (e.g. MXM architecture and technologies, API, MPEG extensible middleware (MXM) protocols)
- MPEG-U (under development at the time of writing this study material): Rich media user interfaces. (ISO/IEC CD 23007) (e.g. Widgets)



3.4 H.263

H.263 is a video codec standard originally designed as a low-bit rate compression format for videoconferencing. It was developed by the ITU-T Video Coding Experts Group (VCEG) in a project ending in 1995/1996 as one member of the H.26x family of coding standards in the domain of the ITU-T. H.263 was developed as an evolutionary improvement based on experience from H.261, the previous ITU-T standard for video compression, and the MPEG-1 and MPEG-2 standards.

Briefly, H.261 is an ITU standard designed for two-way communication over ISDN lines (video conferencing) and supports data rates which are multiples of 64Kbit/s. The algorithm is based on DCT and can be implemented in hardware or software and uses intraframe and inter frame compression. H.261 supports CIF and QCIF resolutions.

The first version of H.263 was completed in 1995 and provided a suitable replacement for H.261 at all bitrates. It was further enhanced in projects known as H.263v2 (also known as H.263+ or H.263 1998) and H.263v3 (also known as H.263++ or H.263 2000).

The next enhanced codec developed by ITU-T VCEG (in partnership with MPEG) after H.263 is the H.264 standard, also known as AVC and MPEG-4 part 10. As H.264 provides a significant improvement in capability beyond H.263, the H.263 standard is now considered primarily a legacy design (although this is a recent development).

Most new videoconferencing products now include H.264 as well as H.263 and H.261 capabilities. The H.263 video compression standard has been defined by the ITU-T for use in a range of video applications over wireless and public switch telephone networks and in applications which include video telephony, videoconferencing, security surveillance, interactive game playing, the internet: much Flash Video content (as used on sites such as YouTube, Google Video, MySpace, etc.) used to be encoded in this format, though many sites now use VP6 or H.264 encoding. The original version of the RealVideo codec was based on H.263 until the release of RealVideo 8. H.263 is required video codec in ETSI 3GPP technical specifications for IP Multimedia Subsystem (IMS), Multimedia Messaging Service (MMS) and Transparent end-to-end Packet-switched Streaming Service (PSS). In 3GPP specifications is H.263 video usually used in 3GP container format.

Self Assessment Test

- 1) List five (5) electronic devices that support MPEG data format
- 2) List two other formats apart from JPEG that you can use to store your image files

Answer to Question 2

PNG - Portable Network Graphics
GIF - Graphics Interchange Format



4.0 CONCLUSION

The MPEG video standards are developed by experts in video compression working under the auspice of the International Organization for Standardization (ISO). The standards activity began in 1989 with the goal of developing a standard for a video compression algorithm suitable for use in CD-ROM based applications. The committee has since standardized MPEG-1, MPEG-2, MPEG-4, and MPEG-4, MPEG-7, and MPEG-21. JPEG is the acronym for Joint Photographic Experts Group. It is the first international image compression standard for continuous-tone still images, including both gray scale and color images. The success of JPEG necessitated further research efforts into an improved standard called JPEG-2000 for coding of still images. JPEG 2000 is a wavelet-based image compression standard. H.263 is a video codec standard originally designed as a low-bit rate compressed format for videoconferencing. It was developed by the ITU-T Video Coding Experts Group (VCEG). It is one of the standards in the H.26x family of video coding in the domain of the ITU-T.

5.0 SUMMARY

We covered some data compression standards for multimedia elements images, video, speech, audio, text etc. The standards discussed in this unit will allow for interoperability of multimedia data across various systems or application domain. These standards are; JPEG/JPEG2000, MPEG, and the H.263 standard. While discussing H.263 standard we compared it with H.261, its predecessor H.264, and H.26x family in order to have a better understanding of its features.

6.0 TUTOR MARKED ASSIGNMENTS

- 1) Describe the four operations modes of JPEG
- 2) What are the main extensions in MPEG 4 compared with MPEG 2
- 3) Briefly discuss the features of H.263 standard

7.0 REFERENCES / FURTHER READINGS

S. Mitra and Tinkuacharya (2003), Data Mining Multimedia, Soft Computing, and Bioinformatics, John Wiley & Sons, Inc, Hoboken, New Jersey, Canada.

B. Furht, Stephen W. Smoliar, H. Zhang (1995), Video and Image processing in multimedia systems, Kluwer Academic Publisher

T. Acharya, P. -Sing Tsai (2005), JPEG2000 Standard for Image Compression: Concept, algorithm and VLSI



B. A. Forouzan, S. C. Fegan (2003), Data Communication and Networking, McGraw Hill Higher Education, Singapore

D. Strannedby, W. Walker(2004), Digital Signal Processing and Applications, Newnes, An imprint of Elsevier, Jordan Hill, Oxford

H. Benoit(1997), Digital Television, MPEG-1, MPEG-2 and Principles of the DVB system, Focal Press, An imprint of Elsevier, Jordan Hill, Oxford

<http://www.mpeg.org/>

<http://www.jpeg.org/>