튜 토 리 얼 V

영상회의 시스템을 의한 영상처리기법

▷ 연 사 : **김 승 범** 박사 (나다기연 대표)

▷ 사 회 : **조 형 제** 교수 (동국대)

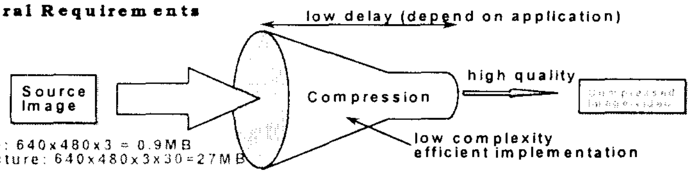
Compression Standard for Videoconferencing and other Standards

(주) 나다기연
서울시 서초구 서초동 1545-7
☎ 525-3941

Table of Contents

I. Introduction

1. General Requirements



Still Image: 640x480x3 = 0.9MB
Motion Picture: 640x480x3x30=27MB

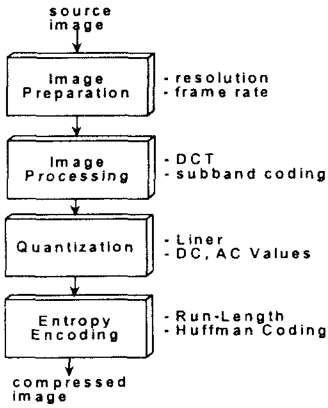
- Amount or degree of compression
  - ⇔ the number of bits per displayed pixel needed in the compressed bitstream
- Image Quality
  - ⇔ lossless compression : perfectly recovering the original representation
  - ⇔ lossy compression : recovering presentation similar to the original one
- Speed of Compression and Decompression
  - ⇔ compression speed is less critical than decompression speed
- by Software or by Hardware
  - ⇔ general-purpose vs. special-purpose acceleration hardware

- **Compression Standards**
  - Why ?
    - ⇔ interoperability of codecs manufactured by different companies
    - ⇔ reduce the prices of codecs
  - Standards
    - ⇔ JPEG : Digital compression and coding of still images (15:1)
    - ⇔ H.261 (px64) : Video coder/decoder for audio-visual services at px64 Kbps (100:1 to 2000:1)
    - ⇔ MPEG : Coding of moving pictures and associated audio (200:1)
  - Applications

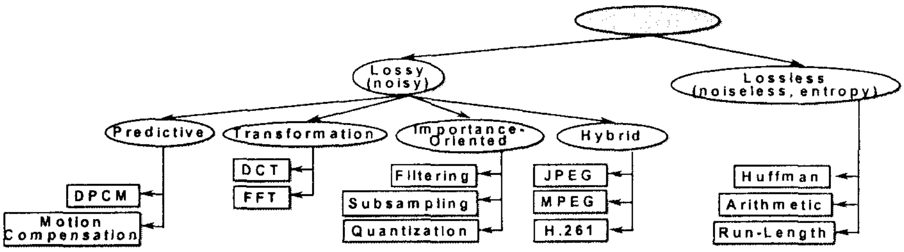
	Bandwidth	Standard	Size (pixels by lines)	Frame rate (frames/sec)
Interactive Multimedia (image stored on CD-ROM)	N/A	JPEG	352 x 288	1
Analog Videophone	5-10 Kpbs	H.263	170x144	2-5
Video Conferencing	64~1000 Kpbs	H.261	352x288	15-30
Interactive Multimedia (Motion Video)	1-2 MB/sec	MPEG-1	352x240	30
Digital NTSC	3-10 MB/s	MPEG-2	720x480	30

II. Basic Compression Techniques

- **Major Steps of Data Compression**
  - Preparation
    - ⇔ the analogue to digital conversion, generating an appropriate digital *representation of the information, dividing* the picture into a block of 8x8 pixels, and represented by a fixed number of bits per pixels
  - Processing
    - ⇔ *a transformation from the time to the* frequency domain using DCT
    - ⇔ interframe coding uses a motion vector for each block of 8x8 blocks
  - Quantization
    - ⇔ the mapping of real numbers into integers (precision reduction)
    - ⇔ the coefficients are distinguished according to their significance
  - Entropy Coding
    - ⇔ compress the sequential digital data stream without loss



2.1 Compression Categories and Techniques



- **Lossy Compression**
  - encoding into a form that takes up a relatively small amount of space, but which can be decoded to yield a representation that humans find similar to the original
  - Methods
    - ⇔ *Predictive*
      - ⇒ predicting subsequent values by observing previous ones, and transmitting only the usually small differences between actual and predicted data
      - ⇒ DPCM, Motion compensation
    - ⇔ *Transformation*
      - ⇒ a transform is a process that converts a bundle of data (a group of pixels, 8x8) into an alternate form which is more convenient for some *particular purpose*
      - ⇒ DCT (Discrete Cosine Transform), FFT
    - ⇔ *Important-based*
      - ⇒ to consider as more important those parts of an image that humans are better attuned to
      - ⇒ Quantization, *Filtering, Subsampling*, CLT *(Color Lookup Table)*
    - ⇔ *Hybrid*
      - ⇒ systems and standards for video compression often apply
        - → *motion compensation* for temporal compression
        - → *transform coding* for spatial compression, and
        - → *Huffman coding* or *arithmetic coding* for statistical compression

2.2 Entropy Coding (Lossless Coding)

- the data stream *to be compressed is considered to be a simple digital* sequence and the semantics of the data are ignored

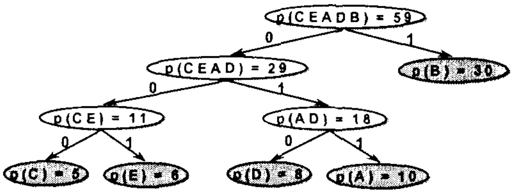
- **Huffman Coding**
  - a statistical data compression technique which codes the more frequently occurring values with words using fewer bits, and the less frequently occurring values will be coded with longer words
  - Assumption
    - ⇔ Some symbols occur more often than other (e.g., characters in an English Text)
  - Principle
    - ⇔ Frequently occurring symbols are coded with shorter strings
  - Needs coding Table (code book)

Symbol	Code
A	001
B	1
C	011
D	000
E	010

A B D C A A E D C
001 1 000 011 001 001 010 000 011

* the code book may be transmitted once for each individual image, or it may even be *transmitted for individual blocks of a single* image

- Algorithm
  1) Rank all symbols in order of probability of occurrence
  2) Successively combine the two symbols of the lowest probability to form a new composite symbol; eventually we will build a binary tree where each node is the probability of all nodes beneath it
  3) Trace a path to each leaf, noticing the direction at each node
- Example
  - ⇔ Characters to be encoded : A, B, C, D, E
  - ⇔ Given probabilities of occurrence:
    - ⇒ p(A) = 10, p(B) = 30, p(C) = 5, p(D) = 8, p(E) = 6



Symbol	Code
A	011
B	1
C	000
D	010
E	001

- **Run-Length Coding**
  - blocks of repeated pixels are replaced with a single value and a count of how many times to repeat that value
  - Assumption
    - ⇔ data should have many subsequently following equal symbols
  - Example

... A B C E E E E E E D A C B ....

↓ compression

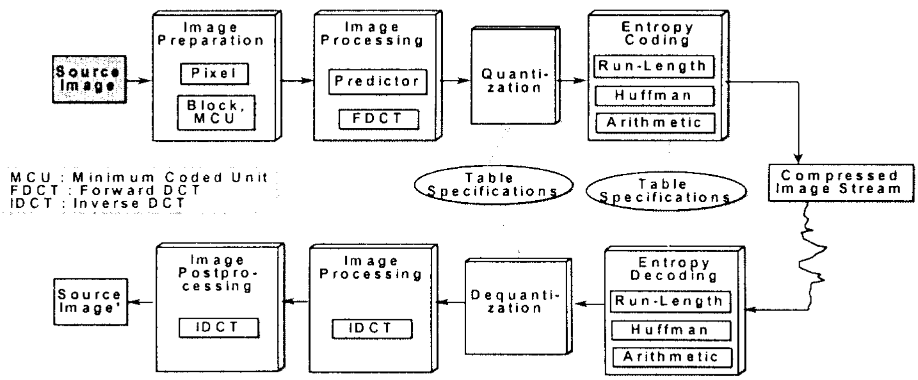
... A B C E 6 D A C B ....

symbol (E) — number of occurrences (6)

2.3 JPEG (Joint Photographic Expert Group)

- **Overview**
  - International Standard
    - ⇔ For digital compression and coding of continuous-tone still images:
      - ⇒ Gray-scale, Color still
  - Very general compression scheme (e.g., 255 planes)
  - Independency of:
    - ⇔ Image resolution
    - ⇔ Image and pixel aspect ratio
    - ⇔ Color representation
    - ⇔ Image complexity and statistical characteristics
  - Well-defined interchange format of encoded data
    - ⇔ *regular format* : the encoded data stream has a fixed interchange format that includes the encoded image data as well as the chosen parameters and the tables of the coding process
    - ⇔ *abbreviated* format : does not guarantee inclusion of necessary tables
  - Implementation in:
    - ⇔ Software only
    - ⇔ Software and Hardware
  - "Motion JPEG" for video compression
    - ⇔ for video editing

- **JPEG Image Compression/Decompression Steps**



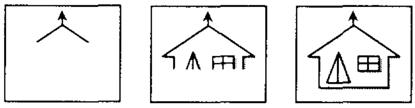
- **4 Modes of Compression**
  - *Lossy Sequential DCT-based Mode (Baseline Mode)*
    - ⇔ each image component is encoded in a single left-to-right, top-to-bottom scan
    - ⇔ must be supported by every JPEG implementation
    - ⇔ "Block, MCU" + "FDCT" + "Run-Length" + "Huffman"
  - *Extended Lossy DCT-based Mode (Progressive Encoding Mode)*
    - ⇔ provides a set of further enhancements to the baseline mode
    - ⇔ the image is encoded in multiple scans for applications in which transmission time is long, and viewer prefers to watch the image build up in multiple coarse-to-clear passes
    - ⇔ an image buffer exists prior to the entropy coding step, so that an image can be stored and then parceled out in multiple scans with successively improving quality
  - *Predictive Lossless Mode*
    - ⇔ has a low compression ratio that allows perfect reconstruction of the original image
  - *Hierarchical Mode*
    - ⇔ comprise images of different resolutions and selects its algorithms from the three modes defined above

(2) Extended Lossy DCT-based Mode (DCT Progressive Mode)

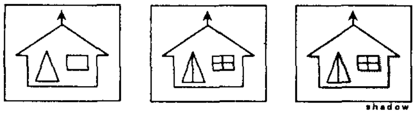
- a sample precision of 12 bits per sample as well as 8 bits per sample can be used

• **Sequential vs. Progressive Image Display**

- Sequential Image Display
  - ⇔ Top -> Bottom
  - ⇔ Good for small image and fast processing



- Progressive Image Display
  - ⇔ Out of focus -> focused
  - ⇔ Good for large and complicated images

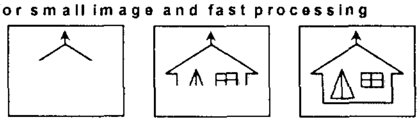


(2) Extended Lossy DCT-based Mode (DCT Progressive Mode)

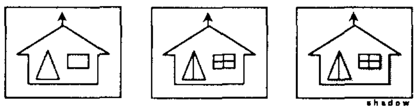
- a sample precision of 12 bits per sample as well as 8 bits per sample can be used

• **Sequential vs. Progressive Image Display**

- Sequential Image Display
  - ⇔ Top -> Bottom
  - ⇔ Good for small image and fast processing

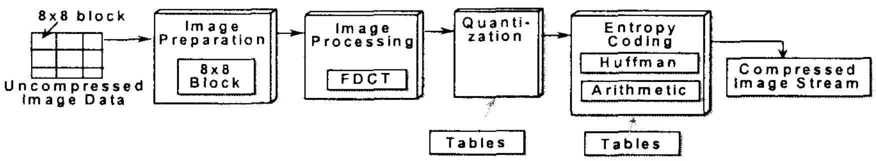


- Progressive Image Display
  - ⇔ Out of focus -> focused
  - ⇔ Good for large and complicated images



(1) Lossy Sequential DCT-Based Mode

- source image samples are grouped into 8x8 blocks, shifted from unsigned integers with range $[0, 2^{P}-1]$ to signed integer with range $[-2^{P-1}, 2^{P-1}-1]$



• **DCT (Discrete Cosine Transformation)**

- DCT is one-to-one mapping for 64-point vectors between the image and the frequency domain
- each 8x8 block of source image samples is effectively a 64-point discrete signal which is function of the two spatial dimension *x* and *y*
- because sample values typically vary slowly from point to point across an image, the FDCT processing step lays the foundation for achieving data compression by concentrating most of the signal in the lower spatial frequencies
  - ⇔ for a typical 8x8 sample block from a typical source image, most of the spatial frequencies have zero or near-zero amplitude and need not be encoded

- **Forward DCT**
  - ⇔ *each of* 8x8 block of source image samples is effectively 1 64-point discrete signal which is a function of the two spatial dimensions *x* and *y*
  - ⇔ the FDCT takes such signal as its input and decomposes it into 64 orthogonal basis signals
  - ⇔ blocks with 8x8 pixel result 64 DCT coefficient
    - ⇒ formula applied 64 times for each 8x8 pixels

$$F(u,v) = \frac{1}{4} C_u \bullet C_v [\sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \bullet \cos\frac{(2x+1)u\pi}{16} \bullet \cos\frac{(2x+1)v\pi}{16}]$$

where $C_u, C_v = \frac{1}{\sqrt{2}}$ for $u,v = 0$; else $C_u, C_v =$ [redacted]

  - ⇔ the output of the FDCT is the set of 64 basis-signal amplitudes or "DCT coefficients" whose values are uniquely determined by the particular 64-point input signal
    - ⇒ F(0,0) : *DC-coefficient*
      - → determines the fundamental color of the data unit of 64 pixels
      - → most important
    - ⇒ Others : *AC-coefficient*
      - → all DCT-coefficient for which the frequency in one or both dimensions is zero
      - → less important

- **Inverse DCT**
  - ⇔ for reconstructing image, the decoder uses the IDCT

$$f(x,y) = \frac{1}{4} [\sum_{u=0}^{7} \sum_{v=0}^{7} C_u \bullet C_v \bullet F(u,v) \bullet \cos\frac{(2x+1)u\pi}{16} \bullet \cos\frac{(2x+1)v\pi}{16}]$$

where $C_u, C_v = \frac{1}{\sqrt{2}}$ for $u,v = 0$; else $C_u, C_v = 0$

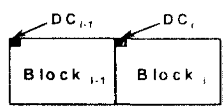
  - ⇔ in principle, the DCT introduce no loss to the source image samples; it merely transforms them to a domain which they can be more effectively encoded

- **Quantization**
  - after output from the FDCT, each of 64 DCT coefficient *is uniformly quantized* in conjunction with a 64-element Quantization Table, which must be specified by the application (or user) as an input to the encoder
  - the purpose of quantization is to achieve further compression by representing DCT coefficients with no greater precision than is necessary to achieve a desired image quality
  - defined as division of each DCT coefficient by its corresponding quantizer step size, followed by rounding to the nearest integer

$$F^{Q}(u,v) = Integer\_Round\left(\frac{F(u,v)}{QT(u,v)}\right)$$

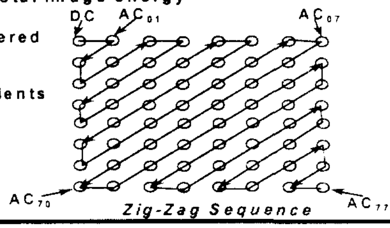
$$F^{Q'}(u,v) = F^{Q}(u,v) \bullet QU(u,v)$$

- **DC Coding and Zig-Zag Sequence**
  - ⇔ after quantization, the DC coefficient is treated separately from 63 AC coefficients
  - DC coefficient
    - ⇔ a measure of the average value of 64 image samples
    - ⇔ because there is usually strong correlation between the DC coefficient of adjacent 8x8 blocks, the quantized DC coefficient is *encoded as the difference from the DC term of the previous block* in the encoder order
      - ⇒ this special treatment is worthwhile, as DC coefficients frequently contain a significant fraction of the total image energy



DIFF = $DC_i - DC_{i-1}$

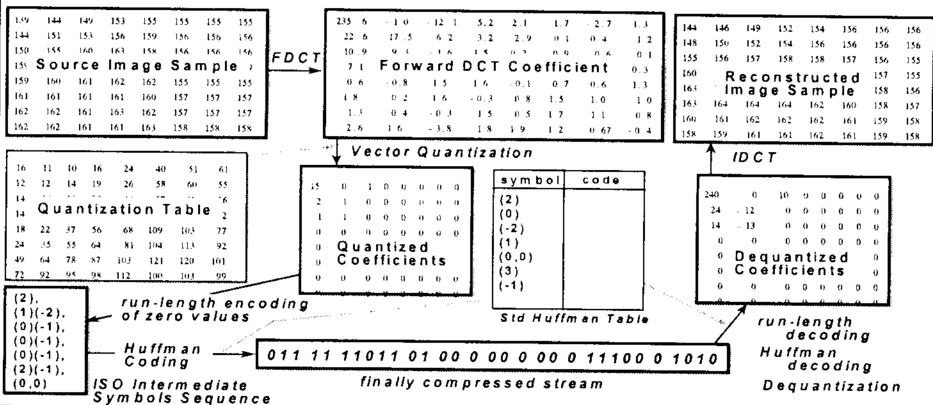
  - AC coefficient
    - ⇔ all of the quantized coefficients are ordered into "Zig-Zag" sequence
    - ⇔ this ordering helps to facilitate entropy coding by placing low-frequency coefficients (which are more likely to be nonzero) before high frequency coefficients



Zig-Zag Sequence

- **Entropy Coding**
  - this final step achieves additional compression losslessly by encoding the quantized DCT coefficients more compactly based on their statistical characteristics
  - Baseline sequential codec uses Huffman coding
  - a 2-step process
    - 1) converts the Zig-Zag sequence of quantized coefficient into an intermediate sequence of symbols
    - 2) converts the symbols to a data stream in which the symbols no longer have externally identifiable boundaries
  - Huffman coding requires that one or more sets of Huffman code tables be specified by the application
    - ⇔ Huffman tables may be predefined and used within an application as defaults, or computed specifically for a given image in an initial statistics-gathering pass prior to compression
- **Compression and Image Quality**
  - for color images with moderately complex scenes, all DCT-based modes of operation typically produce the following levels of picture quality for the indicated range of compression
    - ⇔ 0.25 ~ 05 bits/pixel : moderate to good quality, sufficient for some apps
    - ⇔ 0.5 ~ 0.75 bits/pixel : good to very good quality, sufficient for many apps
    - ⇔ 0.75 ~ 1.5 bits/pixel : excellent quality, sufficient for most apps
    - ⇔ 1.5 ~ 2.0 bits/pixel : usually indistinguishable from the original

- **Baseline Encoding Example**



2.4 H.261 (px64)

- Video codec for audiovisual services at px64 Kbit/s (p=1~30) :CCITT standard
- achieves very high compression ratios for full-color, real-time motion video transmission
- the algorithm combines intraframe and interframe coding to provide fast processing for on-the-fly video compression and decompression, optimized for applications such as video-based telecommunications
- because its applications usually are not motion-intensive, the algorithm uses limited motion search and estimation strategies to achieve higher compression ratios of 100:1 to more than 2000:1

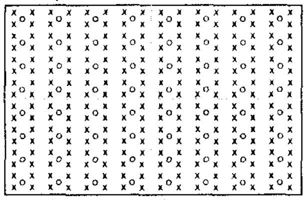
- **Image Preparation**
  - the image refresh frequency at the input must be 29.97 frames/sec, but during encoding it is possible to generate a compressed image sequence with a lower frame rate of e.g. 10 or 15 still image per second
  - the image is encoded as luminance signal (Y) and chrominance difference signals (C_b, C_r) according to the CCIR 601 subsampling scheme (2:1:1)
  - two resolution formats each with an aspect of 4:3 are specified
    - ⇔ Common Intermediate Format (CIF)
      - ⇒ optional, Y:352x288 pixel, at 29.97 frames/s = 36.25 Mbits/sex
    - ⇔ Quarter-CIF (QCIF)
      - ⇒ mandatory, Y:176x144 pixel, at 29.97 frames/s = 9.115 Mbits/sec

- Video Format

	CIF	QCIF
Luminance (Y)	352x288	176x144
Chrominance (C_b)	176x144	88 x 72
Chrominance (Cr)	176x144	88 x 72

- Layered Structure
  - ⇔ Block of 8x8 pixels
  - ⇔ Macro block of
    - ⇒ 4 Y blocks
    - ⇒ 1 C_r block
    - ⇒ 1 C_b block
  - ⇔ Group of blocks (GOBs)
    - ⇒ 3x11 macroblocks
  - ⇔ QCIF picture : 3 GOBs
  - ⇔ CIF picture : 12 GOBs

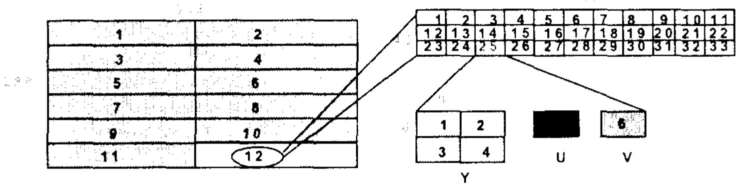
Composition of Macro Block



X : Luminance Sampling
O : Chrominance Sampling

25

- **Picture Structure**
  - 1 CIF picture = 12 GOBs = 12 x (33 macro blocks) = 12 x 33 x (6 blocks)
  - 1 macro block = 4 Y blocks + 1U block + 1V block
  - 1 block = 8 x 8 pixels

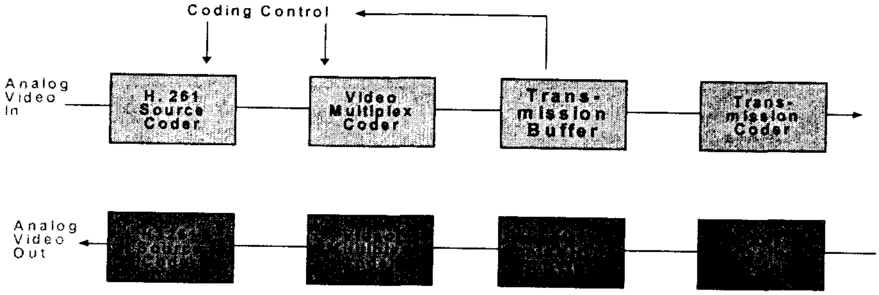


26

- **Image Compression**
  - ⇔ it is a hybrid of DCT and DPCM schemes with motion estimation
  - Intraframe coding
    - ⇔ DCT like JPEG baseline mode : every 8x8 block in a picture frame is transformed into DCT coefficients, linearly quantized, and applying entropy encoding to the AC- and DC-coefficients
  - Interframe Coding, Motion Estimation
    - ⇔ based on a prediction for each macro block of an image
      - ⇒ search of similar macroblock in previous image
      - ⇒ position of this macro block defines motion vector
      - ⇒ *search range is up to the implementation*
        - → i.e. motion vector may be always zero in simple implementation
    - ⇔ results
      - ⇒ *difference between similar macro blocks* and *motion vector*
    - ⇔ difference of previous macro block MVD (coded by DPCM)
      - ⇒ DCT if value higher than a specific threshold
      - ⇒ no further processing if values less than this threshold
    - ⇔ motion vector : components are entropy coded
    - ⇔ quantization
      - ⇒ linear
      - ⇒ adaptation of step size -> enforce a constant data rate at the output of coder

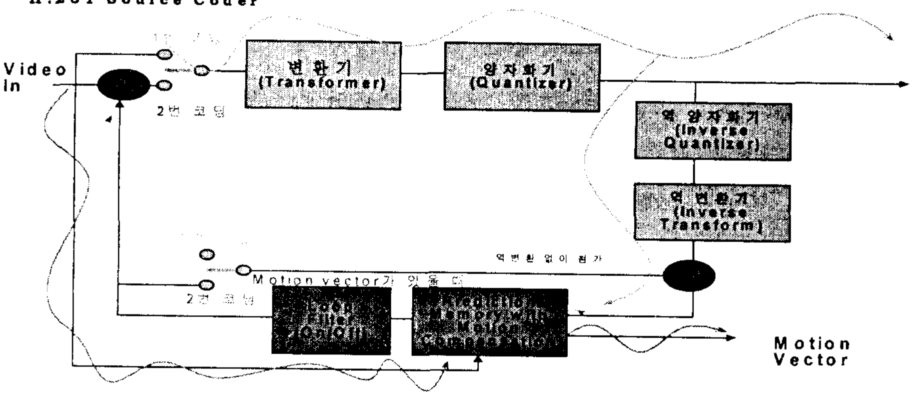
27

- **H.261 Codec**



28

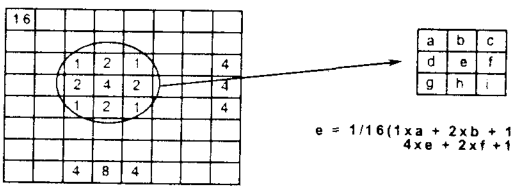
- **H.261 Source Coder**



29

루프 필터

- 역할
  - 양자화에 의한 잡음(거칠게 색조가 나타나는 현상)을 완화
  - 움직임 추정 효율 증가
- 방법
  - block 단위로 공간 저역 필터 (space low pass filter) 사용



$$e = 1/16(1xa + 2xb + 1xc + 2xd + 4xe + 2xf + 1xg + 2xh + 1xi)$$

30

2.4 MPEG

- **Overview**
  - Motion Picture Expert Group (MPEG)
    - ⇔ ISO/IEC JTC/SC29/WG11
    - ⇔ ISO IS 11172 since 3/93
  - it covers motion video as well as audio coding
  - MPEG is striving for a compression of the data stream of about 1.2Mbit/s (150KBytes/sec) which is today's typical CD-ROM data transfer rate
  - consists of 3 parts
    - ⇔ MPEG-Video
      - ⇒ full-motion video compression of television resolution 352x240 down to 1.2Mbps (PAL 형식 352x288@25)
      - ⇒ uses interframe compression, achieving compression ratio of 200:1 by *storing only the differences between successive frames*
    - ⇔ MPEG-Audio
      - ⇒ the compressed bit rate will be two channels of 128 Kbps or possibly 64Kbps
      - ⇒ input sampling rate of 32, 44.1 and 48KHz with 16bits per sample
    - ⇔ MPEG-System
      - ⇒ *audio and video multiplexing and synchronization*

31

- **Features of Video Compression Algorithm**
  - random accessibility, *fast forward/reverse search, reverse play, audio-visual* synchronization, robustness to errors, coding/decoding delay, editability, format flexibility, cost trade-off
- **Applications of Compressed Video**
  - Asymmetric Applications : Electronic Publishing(Education and Training, Travel Guidance, VideoText), Games, Entertainment (movies)
  - Symmetric Applications : Electronic Publishing, Video Mail, Videotelephone, Video Conferencing
- **MPEG Standards**

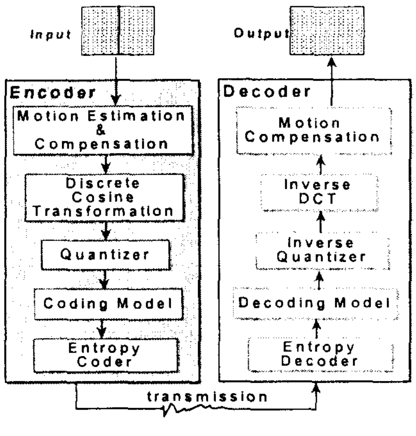
MPEG	Format	Video Parameters	Compressed bit rate
MPEG-1	SIF	352x240@30Hz	1-2 Mb/sec
MPEG-2	CCIR 601 EDTV HDTV	720x480@30Hz 960x480@30Hz 1920x1080@30Hz	3-10 Mb/sec 7-15 Mb/sec 20-40 Mb/sec
MPEG-4	Fill-motion video for small frames and slow refreshments		9-40 Kb/sec

32

(1) MPEG-Video

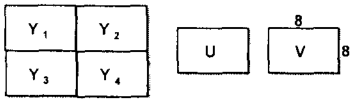
- source video
  - ⇔ a sequence of numbered frames
  - ⇔ each frame is still image
  - ⇔ digitized in a standard RGB format
  - ⇔ 24 bits per pixel
- 352x240 at 30 frames per second
- compressed to 1.5 Mbps

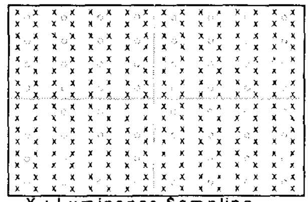
- **Image Preparation**
  - well-defined image format
  - resolution
    - ⇔ should be 최대 768x576 pixel
    - ⇔ 8bit/pixel in each components



33

- color subsampling (2:1:1)
  - ⇔ each image consists of three components (Y, C_b, C_r); the luminance components has twice as many samples in the horizontal and vertical axis as the other two components(가로, 세로 두배 샘플)
    - ⇒ 8 bit luminance per pixel, 8 bits U chrominance per 4 pixels, 8 bits V chrominance per 4 pixels (2:1 compression)
    - ⇒ the number of bits to represent 4 pixels
      - → RGB : 8 bits * 3 (R,G,B) * 4 pixels = 96 bits
      - → YUV : 8 bits *4 (Y) + 8 bits (U) + 8 bits (V) = 48 bits
    - ⇒ this subsampling (lossy step) does drastically affect the quality because the eye is more sensitive to luminance than to chrominance
- each image is divided into areas called '*macro blocks*'
  - ⇔ consists of 6 blocks of 8x8 pixels each
    - ⇒ four luminance (Y) blocks
    - ⇒ two chrominance(U,V) blocks
  - ⇔ the unit of motion compensation





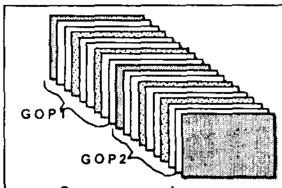
X : Luminance Sampling
O : Chrominance Sampling

- **H.261과의 공통점**
  - 부호화 단위를 MB(Macro Block)으로 한다
  - MB를 Block 단위로 나누어 DCT coding 한다
  - 엔트로피 코딩을 이용하며/움직임 보상 기법을 사용한다
  - DCT 계수의 양자화 스텝을 조정하여 전체 부호량을 조절한다
- **MPEG-1의 새로운 개념**
  - GOP를 도입 random access를 가능케 한다
  - 엔코딩(부호화)시 지연 시간 허용/디코딩(복호)시 실시간
  - 입력 화면 크기 선택이 가능

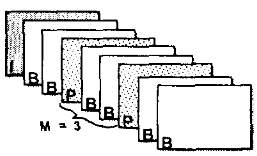
	H 261	MPEG-1
주요대상	통신미디어	저장미디어
비트율(bps)	px64kbps (p=1~30)	약 1.5Mbps까지
복호개시	통신개시	GOP와 시퀀스 헤더
영상 타입	기본적으로 한 종류	주요한 것은 세종류
영상의 드롭	있슴	없슴
예측구조	순방향예측 (MC)	쌍방향예측 (MC)
화면내 구조	GOB	슬라이스
매크로 블록	MBA	MBA (확장)
MB 타입	MB 타입	MB 타입 (영상타입별)
양자화	MQUANT	MQUANT와 양자화 매트릭스
움직임 벡터	정수화소단위 (0~1개)	반화소단위 (0-2개)
루프 필터	있음	없음
CBP	있음	있음
계수의 부호화	이차원 VLC	이차원 VLC (확장)
block coding 방식	DCT	DCT

CBP: Coded Block Pattern(부호화 블록 패턴)

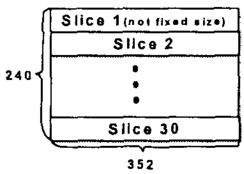
- Video Stream Example (N=9, M=3, 352x240, 4:2:0 subsampling)



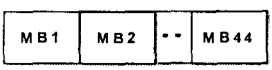
Sequence Layer



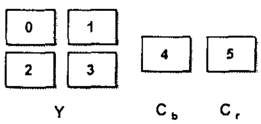
GOP Layer



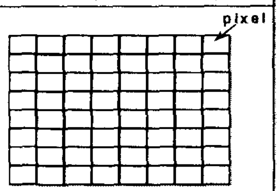
Frame(Picture) Layer



Slice Layer

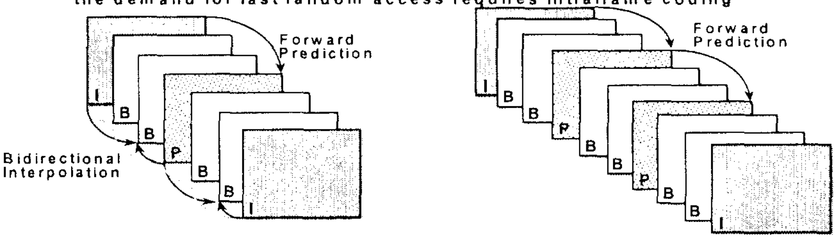


Macroblock Layer



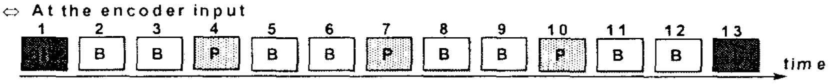
Block Layer

- **Overview of MPEG-Video Compression**
  - *Temporal* Redundancy Reduction (Inter-Frame Coding)
    - ⇔ Block-based Motion Compensation : Prediction, Interpolation
  - Spatial Redundancy Reduction (IntraFrame Coding)
    - ⇔ DCT, Quantization, Entropy Coding
- **4 Types of Coding**
  - ⇔ why ? : contradictory demands for and
    - ⇒ to achieve a high compression ratio, temporal redundancies of subsequent pictures have to be exploited (interframe coding), whereas the demand for fast random access requires intraframe coding

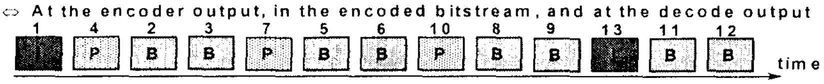


- *I-frames* ('intra coded image)
  - ⇔ self contained : coded without any reference to other images (like J-PEG)
  - ⇔ treated as a still image
  - ⇔ real-time decoding demands
  - ⇔ points for random access in MPEG streams
  - ⇔ the compression ratio of I-frame is the lowest within MPEG
- *P-frame* ('predictive coded frames)
  - ⇔ require information of the previous I- and P-frames for encoding and decoding; the data of the last I-frame as well as from all P-frames that were in between
  - ⇔ a P-frame can be accessed after having decoded the previous I-frame and all other P-frames between the previous I-frame and the current P-frame to be accessed (순차적 디코딩)
  - ⇔ motion vector: MPEG does not define how to determine the motion vector
- *B-frame* ('bi-directional predictive coded frame)
  - ⇔ require for encoding and decoding information of the previous and following I-frame and/or P-frames
  - ⇔ the highest compression ratio is attainable
  - ⇔ is defined as the difference of a prediction of the past image and the following P- or I-frame
- *D-Frame* ('DC-coded frame)
  - ⇔ intracoded frame : only DC-coefficient are DCT coded
  - ⇔ for fast forward and rewind mode

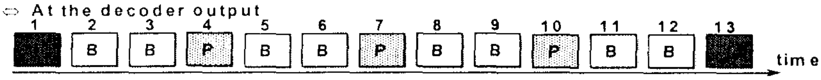
- the regularity of a sequence of I-, P-, and B- frames is determined by the MPEG-applications
  - ⇔ for fast random access, the best resolution would be achieved by coding the whole data stream as I-frames
  - ⇔ the highest degree of compression is attainable by using as many as B-frames as possible
  - ⇔ for practical applications, the following sequence has proved to be useful
    - . IBBPBBPBB IBBPBBPBBI ....의 경우
      - → random access would have resolution of nine still image
      - → still provides a very good compression ratio
- *Frame Reordering*
  - ⇔ At the encoder input



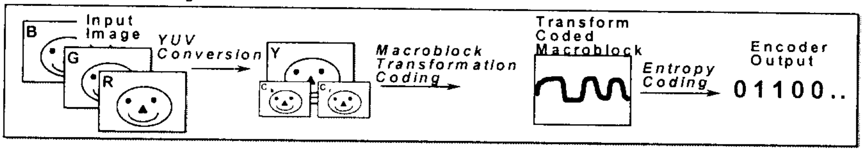
  - ⇔ At the encoder output, in the encoded bitstream, and at the decode output



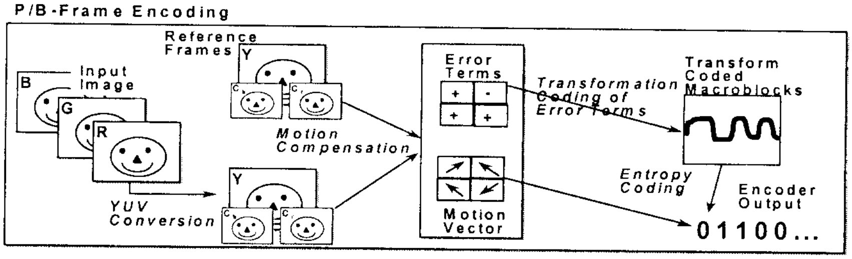
  - ⇔ At the decoder output



- **MPEG Encoding Procedure**
  - I-Frame Encoding

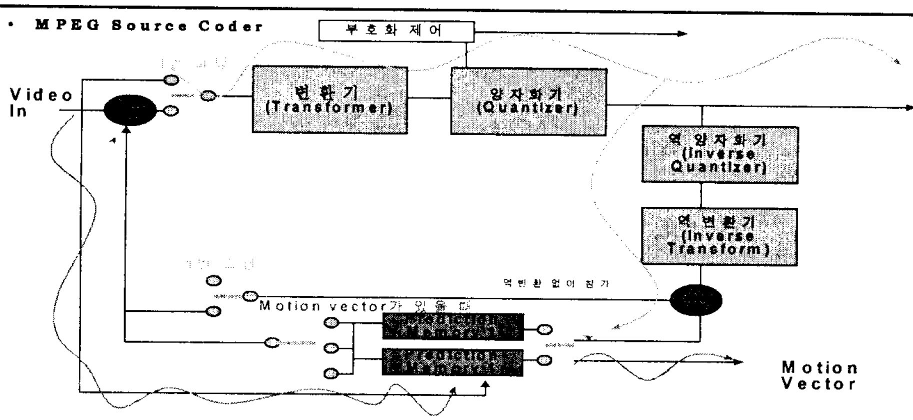


  - P/B-Frame Encoding



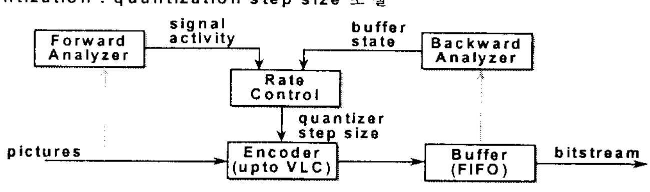
40

- **MPEG Source Coder**



41

- **Rate Control** (비트율 제어)
  - 인코더에서 출력되는 데이터량은 불규칙
    - ⇔ I-Frame >> P-Frame >> B-Frame
    - ⇔ 적절히 제어하지 않을 경우, 움직임이 클수록 많은 비트수가 발생함
  - 고화질을 유지하기 위해서는 적절한 비트배분을 필요
  - Rate control의 목적
    - ⇔ 일정한 전송 속도를 갖는 전송로를 통하여 전송
    - ⇔ buffer overflow나 underflow를 방지
  - Rate control 알고리즘
    - ⇔ 표준화 대상은 아님. (인코더 제작자의 know-how)
    - ⇔ example : MPEG-2 표준화 연구 단계에서 사용된 방법
      - ⇒ Macroblock 단위로 버퍼의 점유도 및 macroblock의 복잡도를 참조하여 quantization : quantization step size 조절



42

3.4 MPEG-2

- 기본 3가지 개념
  - 상호 운영성 (interopeerability)
    - ⇔ 여러가지의 저장 전달 미디어(통신, 방송, 저장 장치 등), 서로 다른 platform (다른 기종의 컴퓨터)등과 정보 교환이 가능도록 설계
  - Scalability
    - ⇔ 비트열의 일부를 취하여 디코딩하여 여러 품질의 비디오로 재생 가능하도록 설계
  - Extensibility
    - ⇔ 어떤 비트율로 전송되는 압축 비트열에 대해 부가분의 정보량을 추가하여 용이하게 비트율을 올릴 수 있다. 예를 들어 10Mbps 비트열에 5Mbps를 더하여 15Mbps 품질의 비트열을 만들 수 있도록 설계
- **MPEG-2에 대한 요구 조건**
  - 컴퓨터
    - ⇔ 디지털 저장 매체를 이용한 고화질 디지털 영상의 제공 (비디오케, VOD등)
  - 방송
    - ⇔ 전파, 케이블을 이용한 고품질 디지털 영상의 제공 (TV, 영화)
  - 통신
    - ⇔ 통신 회선을 통한 고품질 디지털 영상 제공 (영상회의, 전화, 원격 감시 등)

43

- **MPEG-2**
  - From MPEG-1 to MPEG-2 (저급에서 최고급 화질까지)
    - ⇔ improvement in quality : from VCR to TV to HDTV
    - ⇔ No CD-ROM based constraints
  - Data Rates (2 ==> 100 Mbps)
    - ⇔ MPEG-1 : about 1.5 Mbit/s vs. MPEG-2 : 2-100 Mbit/s
- **MPEG-2 Video**
  - inclusion of interlaced video format (일반 camera 입력을 자연스럽게 처리)
  - increase resolution : more than CCIR601(720x480) ==> 1920x1152 (확장된 입력 형식)
  - 5가지 *profiles* (코딩 방식 차이)과 4 levels (해상도의 종류)로 11가지 모드 제공
  - 여러 형태의 scaling :(서로 다른 통신 및 단말 조건들을 수용하여 여러 응용에 사용 가능하도록 하며 상호 호환성 유지)
    - ⇔ *spatial scaling* : decompression of image sequences with dissimilar horizontal and vertical resolution (서로 다른 해상도를 갖는 이미지열의 디코딩)
    - ⇔ *data rate scaling* : a playback with a lower frame rate or for a fast forward mode keeping the frame rate constant (초당 재생 화면 수를 응용에 따라 변경 가능하다)
    - ⇔ *amplitude scaling* : (화질이 다른 두 개의 영상을 동시에 전송 및 재생 가능하다)
      - ⇒ layered coding
        - → can be used for data partitioning in order to be able to transmit more important data with better error correction than less important data
      - ⇒ progressive image presentation (점진적인 재생 가능)

44

- **MPEG-2 Profiles and Levels**

High-Level (1920x1152)		MP@HL (US ATV) < 80Mbit/s			HP@HL < 100Mbit/s
High-1440a Leve (1440x1152)		MP@H-14 < 60Mbit/s		SSP@H-14 (EU HDTV) < 60Mbit/s	HP@H-14 < 80Mbit/s
Main-Level (720x576)	SP@ML (digital CATV) < 15Mbit/s	MP@ML (video disk) < 15Mbit/s	SNP@ML < 15Mbit/s		HP@ML < 20Mbit/s
Low-Level (352x288)		MP@LL < 4Mbit/s	SNP@LL < 4Mbit/s		
Layers and Profiles	Simple Profile	Main Profile	SNR Scalable Profile	Spatial Scalable Profile	High Profile
	No B-frames	B-frames	B-frames	B-frames	B-frames
	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0 or 4:2:2
	Not Scalable	Not Scalable	SNR Scalable	SNRScalable or Spatial Scalable	SNRScalable or Spatial Scalable

- 모두 11가지 동작 모드가 정의되어 있음
- 각 동작모드의 명칭 : (예) MP@ML을 mail profile at mail level이라 부름
- (왼편) ->(오른편), (아래) -> 위, (왼편 아래) -> (오른편 위) 방향으로 순방향 호환성 성립
  (예) MP@HL용 decoder는 MP@ML용 encoder 출력을 decoding할 수 있음

45

Summary

	Official Name	Standards Group	Compression Ratio (Remarks)
JPEG	Digital Compression and Coding of continuous-tone still images	Joint Photographic Expert Group (JTC1/SC2/WG10)	15:1 (full-color still-frame applications)
H.261 (px64)	Video coder/decoder for audio-visual services at px64 Kbps	Specialist Group on Coding for Visual Technology (ITU SG15?)	1000:1 to 200:1 (video-based tele-communications)
MPEG-1	Coding of moving pictures and associated audio	Moving Picture Expert Group (ISO-IEC JTC1/SC29/WG11)	200:1 (motion intensive applications)
MPEG-2	Coding of moving pictures and associated audio	Moving Picture Expert Group (JTC1/SC29/WG11)	40:1~ (motion intensive applications)
MPEG-4	Coding of moving pictures and associated audio	Moving Picture Expert Group(JTC1/ SC29/WG11)	2000:1 ~ (motion intensive applications)
MHEG	Coded representation of multimedia and hypermedia information	Multimedia and Hypermedia Information Coding Expert Group (JTC1/SC2/WG12)	specify multimedia and hypermedia applications any platform

49