영상 압축 : Digital Image Compression

김 경 섭
삼성종합기술원

1. 개 요

- Digital 문자, 음성, 정지 영상, 동영상 등이 결합된 멀티미디어 정보통신의 혁명이 PC나 TV를 통해 일반 가정까지 도달하게 되어 막대한 멀티미디어 Digital 데이터량을 처리, 용량이 제한된 H/W 장치에 저장, 전송하는 문제가 발생하게 됨.

- 영상정보는 데이터량이 방대하여 여러가지 미디어 중 영상 데이터의 처리가 관건이며 멀티미디어 데이터량을 비교하면 [표 1]과 같음.

- 근래에 Digital 의학 영상장비 발달과 더불어 종래의 Analog 의학 영상 Data 가 다음과 같은 이유로 Digital Data로 저장, 전송 처리가 됨.
  - 의학장비의 발달(Helical CT, Open MRI, PET, SPECT)
  - Computers increase in speed, less expensive.
  - Object Oriented Programming Environment.
  - Digital Data Transmission(PACS).
  - Data Storage in Digital Form, Digital Display Media.

표 1. 멀티미디어의 데이타량

멀티미디어 구분	Digital 저장/전송 Data「1byte=8bits」
Digital Texts	80(행)×25(열)×8bits=16kbits=2kbytes
전화 가입자 음성	인간 음성 대역폭 : 3.4KHz, 전화 교환기 : 8KHz Sampling 전화 가입자 음성 전송 Data=8KHz×8bits=64kbps
CD Stereo Music	인간 가청 주파수 : 20KHz → 44.1KHz Sampling 16bits Stereo=44.1KHz×16bits×2=1.5Mbps
정지 흑백 영상	Basic VGA Mode : 640×480×8bits=307.2kbits 일반 Image Display Mode : 512×512×8bits=2Mbits
동 흑백 영상	RGB Color Mode : 512×512×3×8×bits=6Mbits
동 Color 영상	512×512×512×3×8×30frames=180Mbits
NTSC Color TV	YIQ Mode : 14.3MHz×8bits=114.5Mbps
HDTV	1920×1080×8bits×3×30frames=1.5Gbps

표 2. 의학영상의 데이타량

Modality	Pixel Array	Bit Depth	Size, Bits
Nuclear Medicine	128×128	8	131,072
MR	256×256	10	524,288
Ultrasound	512×512	8	2.09Mbits
Ultrasound Color	512×512	24	6.29Mbits
CT	512×512	12	3.14Mbits
CT	1024×1024	12	12.56Mbits
Digital Fluoroscopy	1024×1024	10	11.04Mbits
CR	2048×2560	10	52.4Mbits
CR	4096×5120	10	209Mbits

-Digital Image Processing 활용.

[표 2]는 Digital 의학 영상의 데이터량을 보여줌.

[표 1]과 [표 2]를 비교하면 의학영상의 데이터량이 많은 것을 알 수 있는 데 이것은 의학 영상의 Pixel을 나타내는 Bit Depth가 8-16 bits이기 때문임. 특히 Bit Depth가 10bits 이상일 때 Data 처리를 16bits로 하여야 하기때문에 더욱 더 많은 데이터량을 요구함. 한가지 예로 CT의 경우 실제로 Computer에 저장되는 데이터량은 1024×1024×16bits=16.7Mbits이며, 한 Study당 평균적으로 50 Images가 포함될 때 16.7×50=835M bits의 방대한 데이터량을 저장하여야 함. 따라서 이러한 Images를 그대로 저장, 전송시 저장장치의 제한된

용량, 전송 Network의 Channel Capacity의 제약으로 문제가 야기 되므로 Images의 Data량을 인위적으로 줄여야함.

Image Compression은 Image Pixels간의 중복성과 Image Transforms을 이용하여 원래의 Data의 용량을 줄여 저장하거나 전송하며, 필요시 원래의 정보와 같거나 근접하게 복원(Image Decompression)하는 데 있음.

2. 영상 압축의 표준안

멀티미디어 정보(문자, 음성,(동) 영상) 가 결합이 되어 PC, 비디오, CD, 디지털 비디오디스크(DVD), 주문형 비디오(VOD), 고선명 TV(HDTV), 디지털 방송(지상, 위성, 케이블) 등에 저장되거나 전송되면서 이에 필요한 멀티미디어 정보 데이터량을 압축하는 기술에 대한 표준을 국제표준화기구(ISO : International Organization for Standardization)나 국제전기통신연합(ITU : International Telecommunication Union) 통하여 제정함. 영상 압축의 국제 표준화의 예를 들면,

2.1. JPEG(Joint Photographic Expert's Group) :

정지영상(Gray, Color)의 표준안이며 DCT(Discrete Cosine Transform)를 기반으로 하는 손실 압축(Lossy Compression)과 추정기(Estimator)와 DPCM(Differential Pulse Code Modulation)을 사용하는 비손실 압축(Lossless Compression) 기법을 정의함. ISO와 CCIT(The International Telegraph and Telephone Consultative Committee)에 의해 결성이 되어 1998, 1990년에 DCT를 기반으로 하는 정지 영상 압축에 대한 표준을 제정함.

손실 압축법은 압축율이 클수록 영상 품질이 떨어지나 넓은 응용 분야에 적용되며 압축율이 낮으나 영상이 완전히 복원되는 비손실 압축법은 진료 시스템 및 군사용 시스템에 이용됨.

2.2. JBIG(Joint Bi-Level Image Expert's Group) :

주로 Digital FAX용 Bi-Level 영상의 Lossless 압축 기법 표준안임.

2.3. MPEG I(Moving Picture Expert's Group I) :

동영상 압축전송의 표준안이며 H2.61 의 기법을 기초로 하여 화상 전화용을 주목적으로 개발된 1.5 Mbps의 통신로를 지원하는 표준안이며 CD-ROM, Video CD 같은 디지털 저장 매체 방식에 쓰임. VideoPhone, Video Conference, Digital TV 방송 등에 활용됨. 1992년에 표준안 완성.

2.4. MPEG II(Moving Picture Expert's Group II) :

고품질 동영상 압축전송의 표준안이며 HDTV와 같이 선명한 영상을 2-20Mbps 통신로를 이용하여 전송하는 체계를 대상으로 채택된 표준기법임. Digital TV, HDTV, VOD, DVD, 위성 방송등에 활용됨. 94년에 완성

2.5. MPEG IV(Moving Picture Expert's Group IV) :

객체 지향 멀티미디어 통신을 위한 차세대 압축 방식으로서, MPEG Ⅰ, Ⅱ보다 낮은 전송율인 64Kbps 이하의 통

신로(전화선)에 동영상을 전송하는 동영상 부호화 표준안이며 모든 가능한 압축/coding 기법을(Fractal, Wavelet coding) 활용한 높은 압축율 기법을 사용함. 일명 Very Low Bitrate Audiovisual Coding으로 명명되어 Pesonal Multimedia, Video Telephone, Mobile Communications 등에 활용되며 97년에 표준안이 완성 예정임.

2.6. H 261

1993년 ITU-T(CCIT)에서 제정한 영상 부호화 표준안으로 주로 영상 회의 시스템 및 영상전화의 실현화를 위함. 64-1920Kbps의 통신로에 사용됨.

2.7. H. 263

전화선을 이용한 동영상 고압축 부호화 표준안이며 일명 TMNx로 명명됨.

이 외에 음성 압축의 표준으로 MPEG Audio, G.721/G.722/G.7128이 있으며 정보 압축 시

- 멀티미디어 통신
- 값싼 미디어로 멀티미디어 정보 기록
- 저장 매체의 소형화
- 고품질 영상 전송
- 다채널의 Digital 방송 등이 가능해짐.

3. Lossless Image Compression(무손실 정지영상 압축전송 방식)

무손실 정지영상 압축전송 방식은 PC에서 사용되고 있는 파일 압축 방법과 유사하며 통계학자 Huffman이 개발한 Huffman Coding과 IBM에서 개발된 Arithmetic Coding 기법이 널리 쓰임. Lossless 압축 Coding 기법은 기본적으로 영상 데이터가 보이는 통계학적 특성을 이용하여 자주 발생하는 데이터에는 적은 수의 bits를 할당하고 적게 발생하는 데이터에는 상대적으로 많은 수의 bits를 할당하여 결과적으로 원영상의 데이터량보다 적은 수의 bits로 원영상을 표시함.

3.1. Lossless Bi-Level Image Coding 의 한 예:

그림 1.을 이용하여 10×15Bi-Level Image의 Lossless 압축예를 살펴보면,

0	0	0	0	0	1	0	0	1	0	0	1	0	0	0
0	0	0	0	1	0	0	0	1	0	0	1	0	0	0
0	0	0	0	1	0	0	0	0	0	1	0	0	0	0
0	0	0	0	1	0	0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	1	0	0	1	0	0	0	0	0
0	0	0	1	0	0	1	0	0	0	1	0	0	0	0
0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
0	0	0	1	0	0	0	1	0	0	0	1	0	0	0
0	0	0	1	0	0	1	0	0	0	1	0	0	0	0

그림 1. 10×15 Bi-Level Image의 예

그림 1.에서 0과 1을 각각 1bit로 나타내면 Image를 coding 하는데 총 필요한 bits 수는 10×15×1bit=150 bits d이나 3pixels씩 묶으면 000발생 확률이 다른 데이터 발생율보다 높으므로 0로 표시하고 001을 1001, 010을 1010, 100을 1100으로 표현하면 다음과 같이 총 bits를 계산할 수 있음.

pixels	code	빈도수	bits
000	0	24	24
001	1001	7	28
010	1010	9	36
100	1100	10	40

총 bits : 24+28+36+40=128bits

따라서 그림 1.에서 총 150bits로 표현되었던 영상이 128bits로 압축되어 있음을 알 수 있음(압축비율 : 150/128=1.17)

3.2. Huffman Lossless Image Coding

Huffaman coding 기법은 가장 널리 쓰이는 Lossless Coding 기법이며 영상 데이터의 통계 확률에 분포도에 따라 확률이 높게 발생되는 데이터는 적은bits 로 나타내고 확률이 적게 발생되는 데이터는 보다 더 많은 bits 로 나타내는 가변적 길이 부호화 기법(VLC : variable Length Code)임. VLC 기법에는 Huffman Coding dl외에 Run Length Coding, Arithmetic Coding, Lempel-Zip Coding 기법이 있음. 그림 2를 이용하여 간단한 Huffman Coding의 예를 들어보면

Graylevels	n_i	h_i
0	3441	0.21
1	4423	0.27
2	3932	0.24
3	1802	0.11
4	1311	0.08
5	819	0.05
6	492	0.03
7	164	0.01
n_T	16384	

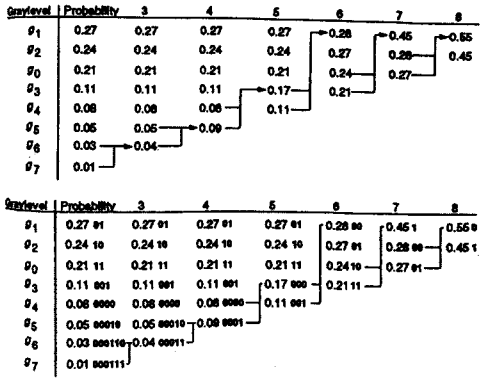


그림 2. Huffman Coding 예

그림 2을 참조하여 Huffman Coding 기법을 다음과 같이 설명할 수 있음.

1) First Phase :

① 영상의 Graylevel 확률 분포도를 구함. 그림 2는 8 Graylevels을 보여줌.

② 확률이 큰 Graylevel 부터 적은 Graylevel로 정리함.

③ 확률이 제일 적은 것 2개를 합하여 새로운 확률로 구한 뒤 크기 순서대로 정렬함.

④ 2개의 확률 분포도가 될 때까지 ③을 반복함.

2) Second Phase :

① 정리된 2개의 확률 분포에 0, 1bit를 assign함. 즉 그림 2에서 0.55 확률에 0, 0.45 확률에 1을 assign 함.

② 2개의 확률 분포를 구성하는 하위 확률들을 찾아 역 방향으로 bits를 assign함.

그림 2에서 최종적으로 얻어진 Huffman Codes는…

Graylevel	Probability	Binary Code	Huffman Code
g1	0.27	001	01
g2	0.24	010	10
g0	0.21	000	11
g3	0.11	001	001
g4	0.08	100	0000
g5	0.05	101	00010
g6	0.03	110	000110
g7	0.01	111	000111

따라서 Binary Code로 Graylevel이 46,384×3bits=49,152bits로 표현되고, Huffman Code로는 (3441+4423+3932)×2+1802×3+1131×4+819×5+(492+164)×6=42,273bits로 되어 19452/42273=1.16의 압축율을 가짐. Lossless Coding은 원영상과 압축된 영상이 복원 시 완전히 일치하는 기법이나 압축율이 낮아

(3 : 1이하) 주로, 인공위성의 사진 영상, 의료용 사진 영상, 영상의 데이터의 가치가 매우 높거나 다시 같은 영상을 획득하려면 많은 비용이 드는 제한된 용도에 쓰임.

4. Lossy Image Compression(손실 정지영상 압축)

●Lossy Coding은 압축되어 복원된 영상의 질을 판단하는 것이 인간의 눈인 것을 감안하여 비록 복원된 영상이 숫자적으로 원래의 영상과 일치하지 않더라도 인간의 눈으로 보기에 원래의 영상과 압축/복원된 영상을 구분하기 어려운 범위 내에서 압축을 하는 기법임.

●일반적으로 압축율과 복원된 영상 품질은 상호 반비례하는 성격을 보여주어 압축율을 높일 경우 압축/복원된 영상의 화질이 점점 떨어짐. 저장 용량이 부족하거나 전송로의 용량이 적을 경우 매우 높은 압축율이 요구되는 데 이때는 화질 저하를 인간의 눈으로 감지할 수 있는 범위까지 압축이 이루어짐.

●Lossy Coding은 Lossless Coding과는 달리 원래의 영상을 완벽하게 복원하는 데 목적을 두지 않으며 원영상의 정보를 일부 잃더라도 인간의 눈이 허용하는 범위 내에서 데이터를 높은 비율로 압축을 하는 데 중점을 둠.

●Lossy Coding 기법은 기본적으로 그림 3.에서 보는 바와 같이 1) Image Mapping(영상 재표현) 2) Image Quantization(양자화) 3) Lossless VLC Encoding(일반적으로 Huffman Coding 을 Modify 하여 씀) 4) Image 저장/전송 System 5) Lossless Decoding 6) Image DeQuantization(역 양자화) 7) Inverse Image Mapping(영상 역방향 재표현) 등으로 구성되어 있음.

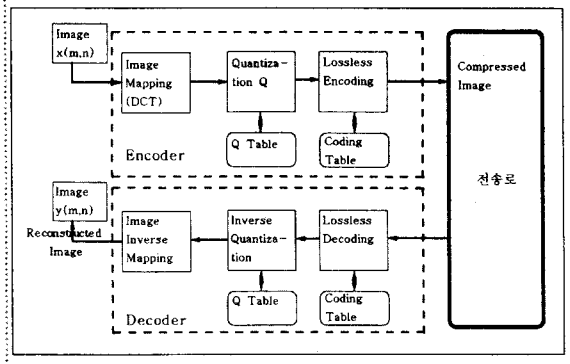


그림 3. Lossy Image Compression/Decompression

●그림 3에서 원래의 영상을 x(m, n), 압축되어 복원된 영상을 y(m, n) 이라 할 때

{x(m, n)−y(m, n)} ^ 2/총 pixel 수=distortion
{y(m, n)을 나타내는 총 bits 수}/{x(m, n)을 나타내는 총 bits 수}=rate

라고 정의함. 즉 distortion은 압축 후 복원된 영상과 원래의 영상의 차이를 각각 pixel에서 구한 후 제곱을 하여 더한 뒤 평균한 값으로 정의되며 이러한 수치적 측정치를 MSE(Mean Squared Error)라고 함. MSE 수치는 인간의 시각적 특성을 전혀 고려하지 않으므로 압축 후 복원된 영상의 MSE가 적다고 인간의 눈에 원래의 영상에 반드시 가깝게 느껴지지는 않음. Rate가 적다는 것은 y(m, n)을 나타내는 총 bits 수가 원래의 영상 x(m, n)을 나타내는 총 bits 보다 적으므로 압축율이 높다는 이야기임.

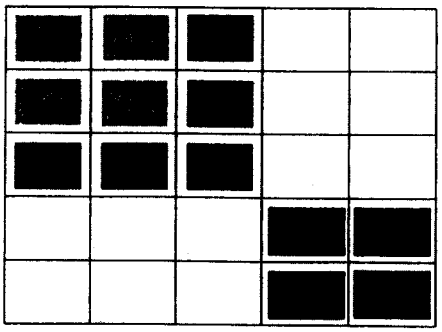
●Lossless Coding 기법은 distortion이 0일 때 rate를

줄이는 기법이라 정의되며 Lossy Coding은 1) distortion이 주어졌을 때 rate를 줄이는 기법 2) rate가 주어졌을 때 distortion을 줄이는 기법임. 일반적으로 rate 값이 커지면(압축율이 낮으면) distortion이 감소하고 distortion이 커지면 rate 값이 적어짐.(압축율이 높아짐) 따라서 Lossy Coding 기법은 영상을 주어진 rate나 distortion에 따라 Image Transform Mapping, Quantization, VLC bits 할당 등을 적절히 구사하여 영상을 압축/복원하는 기법임.

그림 3의 구성 요소들을 순서대로 설명하면 다음과 같음.

5. Image Mapping

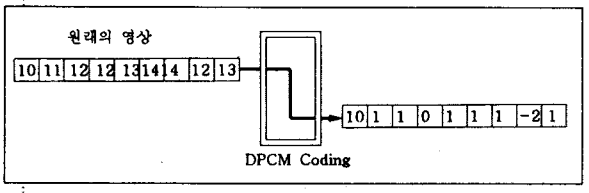
Image Mapping은 영상을 압축하기 편리하게 원래의 영상을 다시 표현하는 것으로 Lossy Compression의 첫 단계임. 일반적으로 영상 데이터는 그 데이터의 통계학적 변화가 인접한 Pixels 에서는 적음.(인접한 Pixel 에서 통계학적 변화가 클 때 보통 Noise로 간주됨) 이러한 것을 영상의 데이터가 "Highly Correlated"되어 있다고 함.

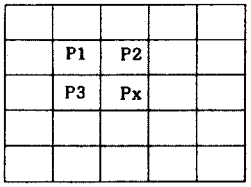


영상 데이터를 압축하기에 편리한 형태로 재표현하는 기법으로는 DPCM Coding과 직교(Orthogonal) Image Transforms이 있음.

5.1. DPCM

첫번째 Pixel Data를 그대로 표현한 후 다음에 이어지는 데이터는 순차적으로 앞 데이터의 차이 값으로 표현하는 등차의 방법으로 영상을 재표현함. 이런 경우 원래의 영상이 갖는 최대와 최소의 차이값 보다 DPCM Coding된 데이터는 훨씬 더 적은 최대 최소의 값으로 영상을 재표현할 수 있음. 또한 단순히 앞 데이터와 현 데이터의 차이로만 표시하지 않고 여러 개의 앞 데이터들로부터 현 데이터를 예측한 후 예측한 값과 현 데이터 값의 차이로 재표현하는 Predictive DPCM Coding 기법이 있음.





예측기의 종류	예측기
1	Px = P1
2	Px = P2
3	Px = P3
4	Px = P1 + P2 −P3
5	Px = P1 + (P2−P3) / 2
6	Px = P2 + (P1 − P3) / 2
7	Px = (P1 + P2) / 2

Predictor

5.2. 직교(Orthogonal) Image Transform

주어진 영상의 데이터를 불규칙하게 퍼져있는 각 주파수의 성분 값, 또는 화소 값(Graylevel)을 저주파 쪽으로 집중시키는 Image Transform을 말함. 이러한 Image Transform을 적용시키면 원래의 영상의 고주파 성분들이 저주파 쪽으로 이동하여 고주파 성분이 매우 작게 되어, 고주파 성분을 생략하여 데이터를 압축하는 것임. 이렇게 고주파 성분을 저주파쪽으로 이동시키어 고주파 성분이 적게 되는 것을 "Energy Packing" 현상이라 부르며 Energy Packing이 크면 클수록(고주파 성분이 적어지면 적을수록) 영상의 압축이 용이해짐. 따라서 Energy Packing 현상을 개선하기 위해 다음과 같은 Image Transforms에 대한 연구가 이루어짐.

- DCT(Discrete Cosine Transform) : 이산여현 변환
- DST(Discrete Sine Transform)
- Hadamard Transform
- Harr Transform
- KLT(Karhunen Loeve Transform)
- Slant Transform

이중에서 KLT는 가장 좋은 Energy Packing 현상을 보이나 계산 속도가 느려 활용되지 않고 KLT를 제외한 Transform 중 가장 좋은 Energy Packing 현상을 보이는 DCT가 활용됨. 특히 DCT에 대한 고속 계산법이 개발되어 JPEG, MPEG, H.261 등의 국제 표준규격 Image Transform으로 채택됨.

5.3. DCT

DCT는 1960년 후반부터 70년대에 걸쳐 K. R. Rao 등이 개발한 Image Transform으로 다음과 같이 정의됨.

$$C(u,v) \quad \alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)cos[(2x+1)u\pi/2N]cos[(2y+1)v\pi/2N]$$

for u, v = 0, 1, 2, ..., N −1, and

$$\alpha(u)=\sqrt{1/N} \text{ for } u = 0$$

$$=\sqrt{2/N} \text{ for } u = 1, 2, \ldots, N-1.$$

α(v)도 α(u)와 동일하게 정의됨.

DCT에 대응되는 IDCT(Inverse Discrete Cosine Transform)는

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)C(u,v)cos[(2x+1)u\pi/2N]cos[(2y+1)v\pi/2N]$$

for x, y = 0, 1, 2, . . . , N − 1

5.4. DCT의 예

f(0,0)	f(0,1)	f(0,2)
f(1,0)	f(1,1)	f(1,2)
f(2,0)	f(2,1)	f(2,2)

영상 데이타 → DCT →

C(0,0)	C(0,1)	C(0,2)
C(1,0)	C(1,1)	C(1,2)
C(2,0)	C(2,1)	C(2,2)

DCT Coefficients C(u, v), (N=3)

$$C(0,0) = \sqrt{1/3} * \sqrt{1/3}\ (f(0,0) + f(0,1) + f(0,2) + f(1,0) + f(1,1) + f(1,2) + f(2,0) + f(2,1) + f(2,2))$$

$$C(0,1) = \sqrt{1/3} * \sqrt{2/3}\ \sum_{x=0}^{2} \sum_{y=0}^{2} f(x,y)cos[0/6]cos[(2y+1)\pi/6]$$

$$= \sqrt{1/3} * \sqrt{2/3}\ (f(0,0)\cos(\pi/6) + f(0,1)\cos(3\pi/6) + f(0,2)\cos(5\pi/6) + f(1,0)\cos(\pi/6) + f(1,1)\cos(3\pi/6) + f(1,2)\cos(5\pi/6) + f(2,0)\cos(\pi/6) + f(2,1)\cos(3\pi/6) + f(2,2)\cos(5\pi/6))$$

$$C(1,1) = \sqrt{2/3} * \sqrt{2/3}\ \sum_{x=0}^{2} \sum_{y=0}^{2} f(x,y)cos[(2x+1)\pi/6]cos[(2y+1)\pi/6]$$

$$C(2,2) = \sqrt{2/3} * \sqrt{2/3}\ \sum_{x=0}^{2} \sum_{y=0}^{2} f(x,y)cos[(2x+1)2\pi/6]cos[(2y+1)2\pi/6]$$

그림 4는 영상과 이를 DCT로 취하여 크기를 표시한 것인데 DCT를 취하면 저주파에 화소 값이 몰리고 고주파의 화소 값이 매우 적은 것을 알 수 있으며 그림 3에서 설명된 것처럼 영상압축은 원래의 영상을 DCT를 취한 뒤 이를 양자화하여 인위적으로 데이터 수를 줄인 후, Lossless VLC Coding을 하여 영상을 저장하거나 전송함. 영상복원은 전송된 Codes를 역으로 Decoding 한 후 이를 역양자화하여 데이터 수를 원래대로 회복한

뒤 이를 IDCT를 취하여 영상을 복원함. 영상을 Lossless Coding/Decoding, 양자화/역양자화 시 미리 정해진 VLC Coding Table과 양자화 Table을 씀. 따라서 영상을 복원 시 수신자는 송신자가 사용하였던 VLC Coding Table과 양자화 Table을 정확히 알아야 함.

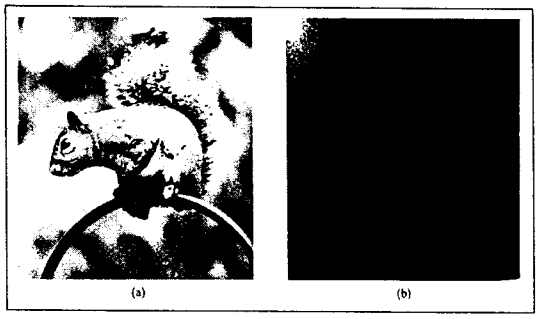


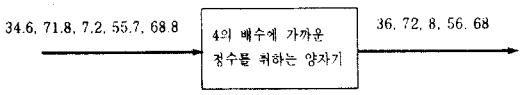
그림 4. a) 영상, b) DCT를 취한 결과

Gray level 정지영상을 JPEG 표준으로 압축 시 기본적으로 DCT를 사용하나 효과적인 계산을 위하여 전체 영상에 대한 DCT를 계산하지 않고 영상을 Non-Overlapping 8×8blocks으로 나누어 각각 block에 대해 DCT를 계산 양자화를 한 후 VLC Lossless Coding하여 압축 전송함. 수신자는 수신된 8×8block에 대한 VLC Coding을 DeCoding, 역양자화를 취한 후 8×8block에 대해 IDCT를 취하여 영상을 8×9block 단위로 복원함.

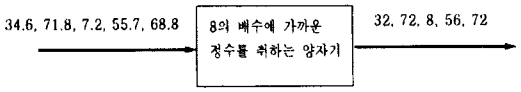
5.5. 양자화(Quantization)

양자화는 인위적으로 데이터의 양을 줄이는 과정이며 Lossy Coding에서 대부분 압축은 이 과정에서 이루어짐. 양자화 과정은 입력 데이터의 값을 미리 정해진 구역별의 값으로 변환 시키며 양자화 스텝이 작으면(양자화 구역이 많으면) 양자화된 영상은 원래의 영상과 가깝게 복원되지만 압축율이 작아짐. 따라서 양자화는 연속적인 입력신호를 이산적(Discrete) 출력 신호로 대응시키는 비선형 함수이며 양자화를 하는 방법에 따라 1) Uniform 2) Nonuniform 3) Vector 4) Adaptive Quantizer 등이 있으며 1960년 Max가 제시한 Nonuniform Lloyd-Max Quantizer가 널리 쓰이고 있음. Lloyd-Max Quantizer는 입력 영상의 확률 분포를 구하여 확률이 높은 입력 데이터들은 작은 양자화 스텝을, 확률이 낮은 입력 데이터는 큰 양자화 스텝을 적용하여 양자화된 데이터와 원래의 데이터 간의 MSE(Mean Square Error)를 최소하 시킴. 양자화기의 간단한 예를 들면

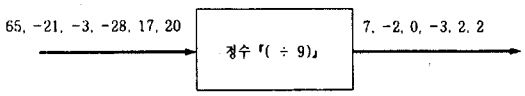
① 양자화기의 한 예



② 양자화기의 한 예



③ 양자화기의 한 예



5.6. Lossless VLC Coding

양자화기를 거쳐 인위적으로 양이 줄여진 데이터는 효과적으로 데이터를 저장하거나 전송하기 위해 VLC Coding을 거치게 되며 보통 앞에서 소개된 Huffman Coding이나 변형된 Huffman Coding을 통해 빈도수가 높은 데이터에는 적은 수의 bits를 할당하고 빈도수가 적은 데이터에는 상대적으로 많은 수의 bits를 할당하게 됨. Huffman Coding은 Lossless Coding 기법이어서

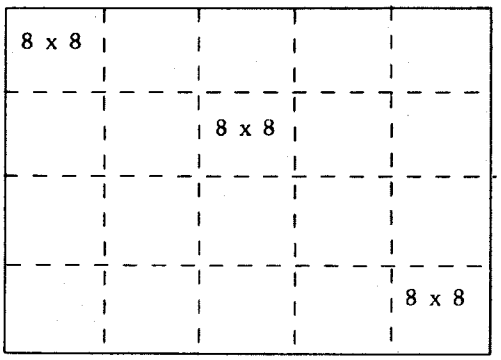
Coding시 사용하였던 Coding Table을 이용하면 정확하게 Decoding 할 수 있으며 일반적으로 JPEG 같은 표준 규격 단체에서 미리 정해진 Huffman Coding Table을 사용함. 다음은 JPEG 에서 규정한 Huffman Coding Table의 한 종류임.

Zero Run	Category	Code length	Codeword
0	1	2	00
0	2	2	01
0	3	3	100
0	4	4	1011
0	5	5	11010
0	6	6	111000
0	7	7	1111000
.	.	.	.
.	.	.	.
1	1	4	1100
1	2	6	111001
1	3	7	1111001
1	4	9	111110110
.	.	.	.
.	.	.	.
2	1	5	11011
2	2	8	11111000
.	.	.	.
.	.	.	.
3	1	6	111010
3	2	9	111110111
.	.	.	.
.	.	.	.
4	1	6	111011
5	1	7	1111010
6	1	7	1111011
7	1	8	11111001
8	1	8	11111010
9	1	9	111111000
10	1	9	111111001
11	1	9	111111010
.	.	.	.
.	.	.	.
End of Block (EOB)		4	1010

그림 5. JPEG Huffman VLC Codes Table

6. JPEG 기준 정지영상 압축의 기본 표준안 :

- 원래의 영상은 8×8 Non-Overlapping blocks으로 분할되어 각각 block마다 독립적으로 8×8 DCT를 취하여 영상을 재편집함.



- 8×8 DCT를 취하여 얻어진 C(u, v), u=0, … 7, v=0, … 7의 계수 값을 사용자가 정의한 값으로 Normalized 시키며 Normalized 값은 모든 blocks에 똑같이 적용시킴. 단, 송신자는 수신자에게 Normalized Array(8×8) 값을 Head 정보로 보내야 함.

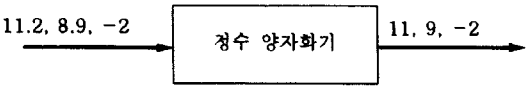
JPEG 표준안에서 쓰이는 대표적인 8×8 Normalized Array N(u, v)는

N(u,v) =

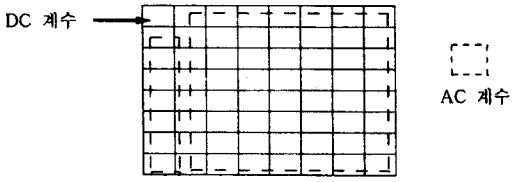
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

그림 6. JPEG 양자호를 위한 8×8 Normalization Array Table

- Normalized 된 8×9 DCT계수 값은 양자화기에 의해 Normalized 계수값(보통 Float Number)의 가까운 정수 값으로 양자화 됨. 예를 들면 Normalized 된 계수 값이 11.2, 8.9, −2이면 양자화기를 거친 값은 11, 9, −2가 됨.



● Normalized, 양자화가 된 8 x 8 DCT 계수값 중에서(0, 0) 에 해당되는 계수 값은 DC, 나머지를 AC 계수 값이라 함.



일반적으로 DC 계수 값은 AC계수 값에 비해 매우 크며, AC 계수 값은 많은 0 값을 포함함.

● DC 계수 Coding : Huffman Codes 로 바꾸기 위해 DC계수와 AC계수를 각각 다른 방법으로 bits 를 할당 함. DC 계수는 값이 크기 때문에 이전 block 의 DC계수 값의 차이를 구하여 다음과 같이 SSS 값과 이에 해당되는 bits 를 assign 함.

SSS	block 과 block 간의 계수값의 차이
0	0
1	-1, 1
2	-3, -2, 2, 3
3	-7..-4, 4..7
4	-15.. -8, 8..15
5	-31..-16, 16..31
6	-63..-32, 32..63
7	-127..-64, 64..127
8	-255..-128, 128..255
9	-511..-256, 256..511
10	-1023..-512, 512..1023
11	-2047..-1024, 1024..2047

그림 7. DC 계수 Codes

예를 들어 이전 block의 DC 값=34, 현재 block의 DC값=45인 경우

DC 차이 값=45−34=11→SSS=4(100)→1001011로 Coding함.

● AC 계수 bits 할당

AC 계수 값은 고주파 쪽으로 갈수록 0을 가지는 특성을 살리기 위해 8×8 DCT 계수 값을 아래와 같이 Zigzag(갈짓자) Scan 순서대로 1-D vector로 정렬함.

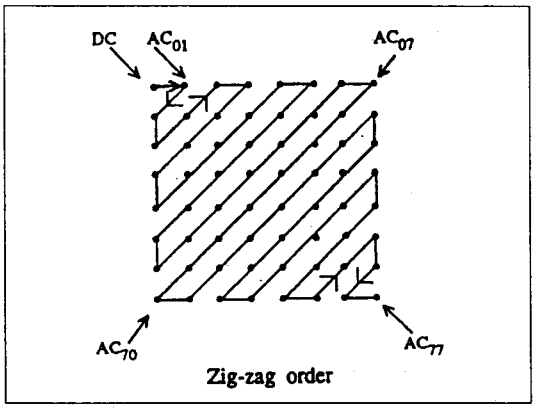


그림 8. Zigzag (갈짓자) AC 계수 Scanning

Zigzag Scanning의 큰 장점은 많은 0을 포함한 AC 계수 값이 Zigzag Scan Order로 정렬되면 0의 순열(Zero Run Length)이 최대화되어 VLC Coding을 효율 있게 수행할 수 있게 함.

AC 계수의 VLC Codes 할당은 먼저 그림 9에서 AC 계수 값에 의해 Grouping Category를 구한 다음 이에 해당되는 Zero Run 숫자를 구하여 그림 5에서 정의한 VLC Huffman Table을 이용하여 Codes를 할당함. 이때 AC 계수 값이 음수이면 이에 해당되는 Sign bit(0: 음수, 양수는 없음)과 실제의 계수 값을 그림 9에서 정해진 Category 안에서 AC계수의 크기를 나타내는 bits 를 추가함.

Category	AC Coefficient Range
1	−1, 1
2	−3, −2, 2, 3
3	−7, ···, −4, 4, ···, 7
4	−15, ···, −8, 8, ···, 15
5	−31, ···, −16, 16, ···, 31
6	−63, ···, −32, 32, ···, 63
7	−127, ···, −64, 64, ···, 127
8	−255, ···, −128, 128, ···, 255
9	−511, ···, −256, 256, ···, 511
10	−1023, ···, −512, 512, ···, 1023

그림 9. AC 계수 Category Grouping

●JPEG 규격 정지영상 압축 예제 :

① 8×8 image f(m, n), m, n=0, 1, ···7가 다음과 같이 주어짐.

$$\begin{bmatrix} 139 & 144 & 149 & 153 & 155 & 155 & 155 & 155 \\ 144 & 151 & 153 & 156 & 159 & 156 & 156 & 156 \\ 150 & 155 & 160 & 163 & 158 & 156 & 156 & 156 \\ 159 & 161 & 162 & 160 & 160 & 159 & 159 & 159 \\ 159 & 160 & 161 & 162 & 162 & 155 & 155 & 155 \\ 161 & 161 & 161 & 161 & 160 & 157 & 157 & 157 \\ 162 & 162 & 161 & 163 & 162 & 157 & 157 & 157 \\ 162 & 162 & 161 & 161 & 163 & 158 & 158 & 158 \end{bmatrix}$$

② 8×9DCT를 취하면 DCT계수 값 C(u, v)를 구하면

$$\begin{bmatrix} 1260 & -1 & -12 & -5 & 2 & -2 & -3 & 1 \\ -23 & -17 & -6 & -3 & -3 & 0 & 0 & -1 \\ -11 & -9 & -2 & 2 & 0 & -1 & -1 & 0 \\ -7 & -2 & 0 & 1 & 1 & 0 & 0 & 0 \\ -1 & -1 & 1 & 2 & 0 & -1 & 1 & 1 \\ 2 & 0 & 2 & 0 & -1 & 1 & 1 & -1 \\ -1 & 0 & 0 & -1 & 0 & 2 & 1 & -1 \\ -3 & 2 & -4 & -2 & 2 & 1 & -1 & 0 \end{bmatrix}$$

③ C(u, v)를 그림 6에서 정의 된 Normalized Array 값으로 나누어 정수에 가까운 값으로 양자화하면 다음과 같이 주어짐.

$$\begin{bmatrix} 79 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

여기서 DC의 값은 79, AC 계수 값은 많은 0을 포함한 것을 알 수 있음.(실제적인 영상 압축이 여기서 일어남).

④ DC 값 79를 이전 8×8block의 DC 값과 비교하여 차이를 구함.(Differential Coding) 만약에 이전 block의 DC값이 70이면 차이는 9이므로 그림 7에서 Category=4, SSS=100, 9=100s로 Coding됨. 따라서 DC계수 값의 Code는

100+1001 = 1001001

⑤ AC 계수 값을 Zigzag Scan Order 로 정렬하면

0 −2 −1 −1 −1 0 0 −1 EOB

(이후 값은 모두 0 이라는 뜻임)

(AC 계수값, Zeroes Run Length)로 표시하면

→(−2, 1)(−1, 0)(−1, 0)(−1, 0)(−1, 2) EOB

⑥ (−2, 1) : 계수값=−2, zero run length=1

그림 9에서 AC Category=2

그림 5에서 Category=2, Zero Run=1→Huffman Code=111011

−2는 음수이므로 추가로 0bit를 assign함.

그림 9에서 Category 2일때 −3, −2, 2, 3이므로 3은 0, 2는 1bit를 assign함.

따라서(−2, 1)에 대한 Huffman Code는

111011+0(sign bit)+1(2를 나타냄)=11101101

⑦ (−1, 0) : 계수값 = -1, zero run length=0

그림 9에서 AC Category=1

그림 5에서 Category=1, Zero Run=0→Huffman Code=00

−1은 음수이므로 추가로 0bit를 assign함.

그림 9에서 Category1 일때 −1이므로 추가로 bits를 assign 할 필요가 없음.

따라서(−1,0)에 대한 Huffman Code는

00+0(sign bit)=000

⑧ (−1, 2) : 계수값=−1, zero run length=2

그림 9에서 AC Category=1

그림 5에서 Category=1, Zero Run=2 → Huffman Code=11011

−1은 음수이므로 추가 0bit를 assign함.

그림 9에서 Category 1 일때 −1, 1이므로 추가로 bits를 assign 할 필요가 없음.

따라서(−1, 2)에 대한 Huffman Code는

11011+0(sign bit)=110110

⑨ EOB → Huffman Code=1010

따라서 8×8 image에 대한 최종 VLC Huffman Code는

1001001/11100101/000/000/000/110110/1010

→ 총 Huffman Code bits 수+34bits

원래의 영상을 나타내는 총 bits 수=8×8×8bits=512bits

압축율=512/34=15 : 1

압축후 Pixel당 bit rate=34/64=0.55 bits/pixel.

압축전 Pixel 당 bit rate=8bits/pixel

⑩ 수신자는 송신자가 사용하였던 양자화 Array와 Huffman Code, IDCT를 이용하여 원래의 영상으로 복원함.

압축 시 : C(u, v)(DCT계수) → C(u, v)/Q(u, v) → VLC Coding

복원 시 : VLC Decoding → VLC Decoding * Q(u, v) → IDCT.

7. Summary

- 영상 압축은 영상의 통계학적 분포, 반복성을 이용하여 빈도가 높은 데이터는 적은 수의 bits를, 빈도가 낮은 데이터에는 보다 많은 수의 bits를 할당하여 전체 영상을 나타내는 bits 수를 줄이는 것임.
- 영상 압축은 크게 Lossy Coding, Lossless Coding으로 나뉘며, Lossy coding은 DCT, 양자화기, VLC Codes를 쓰며 압축 율은 높으나 원래의 영상을 정확히 복원하지 못함.
- 영상 압축에 대한 국제 규격 협회는 JPEG, MPEG Ⅰ, MPEG Ⅱ, MPEG Ⅳ, H.261, H.263 등이 있으나 본 seminar 에서는 JPEG 규격만 논함.
- 의학 영상은 Resolution이 크고 study 단위로 관리되기 때문에 영상 데이터량이 많으나 진단의 목적으로 쓰이기 때문에 주로 lossless 압축을 쓰게 되나 압축율이 낮음.(3 : 1 이하). 최근에는 Fractal, Wavelet Coding을 통한 압축율을 증가 시키는 Image Compression Algorithms이 활용됨.
- MPEG은 동영상의 압축 표준안이며, 동영상은 한 frame 당 25개 이상의 정지 화상으로 이루어지기 때문에 JPEG 규격에서 사용되었던 기법이 그대로 활용되며 영상과 영상간, 또는 frame과 frame 간의 여상의 변화, 움직임을 Vector로 coding하는 interframe Coding 기법을 활용하나 설명하기에는 광범위한 topic이므로 본 seminar에서는 생략함.

References

1. Digital Imaging Compression Techniques, Majid Rabbani, Paul W. Jones, SPIE Tutorial Texts Volume TT7, 1991.
2. 그림으로 보는 최신 MPEG, 정제창 번역, 후지와라 히로시, 교보문고.
3. Fundamentals of Electronic Image Processing, Arthur R, Weeks, Jr. SPIE Press.
4. Video Compression Makes Big Gains, Peng H. Ang, IEEE Spectrum 1991, pp 16-19.
5. Comparison of International Standards for Lossless Still Image Compression, Proceedings of IEEE, vol. 82, no. 6, June 1994.
6. Evaluating Quality of Compressed Medical Images : SNR, Subjective Rating, and Diagnostic Accuracy, Proceedings of IEEE, vole. 82, no. 6, June 1994.