

MVP(Multimedia Video Processor)를 이용한 DCT 알고리즘 구현에 관한 연구

A Study on Application of DCT Algorithm with MVP(Multimedia Video Processor)

°김 상기*, 정 진현*

*광운대학교 제어계측공학과(Tel : 02-940-5156; E-mail: violet@ral.kwangwoon.ac.kr)

Abstract - Discrete cosine transform (DCT) is the most popular block transform coding in lossy mode. DCT is close to statistically optimal transform - the Karhunen Loeve transform. In this paper, a module for DCT encoder is made with TMS320C80 based on JPEG and MPEG, which are international standards for image compression. The DCT encoder consists of three parts - a transformer, a vector quantizer and an entropy encoder.

Keywords - MVP, DCT, JPEG, Quantization

1. 서 론

영상에서 디지털 신호로 처리하면 아날로그 신호에 비해서 잡음 처리에 유리하고 대역폭을 줄일 수 있다는 장점이 있지만 처리해야 할 정보량이 많아진다는 단점도 있다. 일반적으로 영상 압축은 프레임내의 여유분(redundancy)을 줄이는 방법과 프레임간의 여유분을 줄이는 방법으로 나누어진다. 본 논문에서는 정영상의 경우에 프레임내 압축에 대해 다루고 있다.

일반적으로 영상 압축 알고리즘은 transform, quantization, encoding 의 세가지 단계로 나눌수 있다. 첫 번째 transform 단계는 영상을 basis function 에 투영시킴으로 영상을 재구성하여 처리가 용이하도록 하는 것을 말한다. 영상은 주변의 화소값들 사이의 상관 관계가 높기 때문에 이미지를 주파수 성분 에 따라 직교변환(Discrete Cosine Transform)을 하게 되면 전 화면에 불규칙하게 퍼져있는 화소값들이 저주파향 쪽으로 집중되는 경향이 있다. 두 번째 quantization 은 이전의 단계에서 재구성된 영상 정보를 미리 정해놓은 기호에 대응시킨다. 이때 영상의 압축률이 결정되어 진다. 압축률과 영상의 왜곡정도는 역함수관계를 가지게 되므로 어느 한 쪽에 편중되지 않도록 적절한 값을 취하는 것이 중요하다. 영상은 특성상 직교 변환을 하면 저주파향에 집중되므로 저주파향에 큰 비중을 두어 처리하고 고주파향에 작은 비중을 두어 이미지를 처리함으로써 데이터를 압축한다. 마지막 encoding 과정은 제한된 수의 기호로 대응된 영상 정보를 무손실 압축하는 과정이다. 기호의 발생 빈도를 이용한 entropy coding 을 주로 사용한다.

이러한 방법은 정지 영상의 국제 표준인 JPEG에서 사용하는 방법이다. 본 논문에서는 JPEG규정에 의거한 정영상의 압축을 TMS320C80을 이용하여 구현하고자 한다. TMS320C80은 32bit의 RISC구조를 가진 DSP(Discrete Signal Processor)로서 한 개의 MP(Master Processor)와 4개의 PP(Parallel Processor)로 구성되어 있다. TMS320C80은 멀티미디어 비디오 처리를 위해 개발된 DSP로 이를 이용하여 영상을 입력 받고 DCT Coding하여 영상을 압축하도록 시스템을 구성하였다.

2. 본 론

2.1 인코딩 알고리즘

그림[1]에서 실제 영상 압축의 과정을 간단히 나타내었다. 영상 처리에 있어서 특히 DCT와 같은 경우에 영상의 임의의 n×n의 크기로 분할하게 되는데 이는 계산상에 있어서 실제 영상의 크기를 그대로 DCT하기는 현실적으로 불가능하기 때문이다. 이렇게 영상을 분할함으로써 인해서 양자화 step size가 클 경우 복원된 영상에서 블록화 현상(Block artifact)이 보이기도 한다.

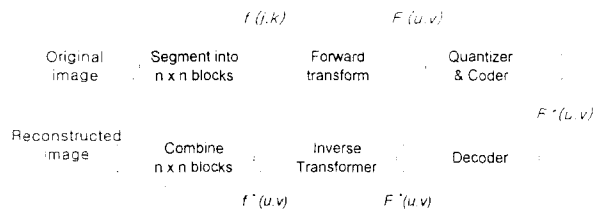


그림 1. 변환 코딩의 구성도
Fig 1. Transform coding block diagram

Transform 은 영상을 기저함수(basis function)에 투영시키는 과정인데 본 논문에서는 주파수 변환의 일종인 discrete cosine transform(DCT)를 이용하였다. 영상을 주파수 공간으로 재구성하면 전 화면에 불규칙하게 퍼져있는 화소값들이 저주파향 쪽으로 집중되는 경향이 있다. 이는 영상은 특성상 주변의 화소값들 사이의 상관 관계가 높기 때문이다. 양자화(quantization)는 제한된 수의 기호로 이전의 단계에서 재구성된 영상 정보를 many-to-one mapping 하는 과정이다. 이 과정에서 영상의 왜곡 비트율(R-D)관계가 결정되어 진다. 압축률과 영상의 왜곡정도는 역함수관계를 가지게 되므로 어느 한 쪽에 편중되지 않도록 적절한 값을 취하는 것이 중요하다. 영상은 DCT를 하게 되면 저주파향에 집중하므로 양자화에 있어서 저주파향의 손실이 적도록 양자화 step size를 작게 하고 고주파향은 step size를 크게 한다. 마지막 encoding 과정은 제한된 수의 기호로 대응된 영상 정보를 무손실 압축하는 과정이다. 기호의 발생 빈도를 이용한 entropy coding 을 사용한다.

2.2 Discrete Cosine Transform

DCT는 블록 단위의 transform 코딩 중에서 가장 널리 사용되는 코딩 방법이다. DCT는 변환하였을 때 성능이 이상적인 변환인 KLT(Karhunen Loeve transform)에 가장 가깝다. 이차원에서 이

산함수 $f(j, k)$, $j, k = 0, 1, \dots, N-1$ 의 DCT는 식(1)으로 정의되고, 역 DCT는 식(2)과 같다.

$$F(u, v) = \frac{4C(u)C(v)}{N^2} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} f(j, k) \cos\left[\frac{(2j+1)u\pi}{2N}\right] \cos\left[\frac{(2k+1)v\pi}{2N}\right] \quad (1)$$

$(u, v = 0, 1, \dots, N-1)$

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

$(j, k = 0, 1, \dots, N-1)$

여기서

$$C(w) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } w=0 \\ 1 & \text{for } w=1, 2, \dots, N-1 \end{cases}$$

이다. 일반적으로 $N=8$ 의 8×8 의 블록을 사용하여 계산하는데 이는 행렬의 곱을 계산할 때 효율이 가장 좋기 때문이다.

DCT는 discrete fourier transform(DFT)에서 허수 스펙트럼 성분이 발생하지 않기 때문에 좀더 높은 압축효과를 얻을 수 있다. DFT는 유한 주기 수열의 이산 Fourier 급수 표현이고 그림 [2]에서 보여 주듯이 절대적인 주기성(implicit periodicity)이 있다. 이러한 주기성은 주파수 공간에서 sampling의 결과이다. 이러한 경우에 반복되는 원래 수열은 각각의 주기 사이에 불연속성을 발생하기도 한다. 이러한 불연속성은 변환의 효율을 적지 않게 감소시킬 수 있는 허수의 고주파 성분들의 결과이다. 비록 이 허수 성분들이 원 수열에 실제 부분이 아니라 할 지라도 이 성분들은 주기적인 수열에서 선명한 경계를 복원시키는데 필요하다. 이러한 경계의 불연속성을 제거하기 위해서 원 수열을 수직축에 대해서 대칭을 이루도록 하여 2배의 길이(원 수열의 길이를 n 으로 할 경우 $2n$)의 수열을 만들 수 있는데 이렇게 확장된 수열은 경계의 불연속성도 없고, DFT를 하였을 때 허수 스펙트럼 성분도 없다(그림 [3] 참고). 이렇게 n 개의 수열을 $2n$ 으로 확장한 수열의 DFT는 원래의 n 개 수열을 DCT한 것과 동일하다. DCT를 사용할 때의 이점을 보면 크게 두 가지로 볼 수 있는데 첫 째는 허수 스펙트럼 성분의 발생이 없어서 코딩효과를 높일 수 있고, 블록화 현상을 많이 줄일 수 있다는 것이고 두 번째는 실수 계산만 필요하다는 것이다. 이러한 이유에서 DCT가 영상 처리에서 가장 많이 사용된다.

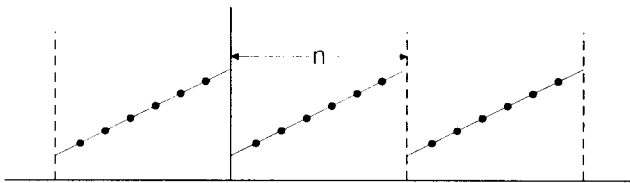


그림 2. DFT의 주기성
Fig 2. DFT periodicity

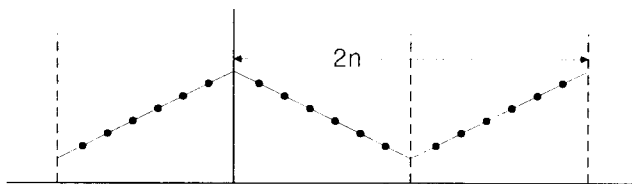


그림 3. DCT의 주기성
Fig 3. DCT periodicity

2.3 Quantizer and encoder

식 (5)를 이용하여 DCT를 통해 얻어진 데이터를 양자화하면 실제 많은 에너지를 가지고 있는 DC성분과 AC의 저주파 성분을 제외한 부분의 값들은 0이 될 수 있다. 그림 [4]는 JPEG recommendation [165]으로부터 얻은 quantization table의 예이다. 그림에서 낮은 주파수 값은 작은 수로 나누어 줌으로써 실제 값과 거의 비슷한 값으로 mapping시키고 높은 주파수는 큰 수로 나누어 주는 것을 알 수 있다. DCT된 신호를 $F(u, v)$ 라하고 그림 [4]의 값들을 $Q(u, v)$ 라고 하면 식(3)을 통하여 quantizer를 거친 신호 $F^*(u, v)$ 를 얻을 수 있다.

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 4. 양자화 표의 예
Fig 4. Sample quantization table

$$F^*(u, v) = \text{Round} \left(\frac{F(u, v)}{Q(u, v)} \right) \approx \left\lceil \frac{F(u, v)}{Q(u, v)} + 0.5 \right\rceil \quad (3)$$

이러한 과정을 거친 신호들은 다시 기호의 발생 빈도를 이용한 entropy coding을 하게 된다. 고주파항이 대부분이 0이 되므로 저주파에서 고주파순으로 그림[5]과 같은 zigzag scan이나 alternate scan을 하여 나열하면 더 나은 코딩의 효과를 얻을 수 있다. 그림의 화살표 방향으로 2차원 배열의 영상 데이터를 1차원 배열로 바꾸어 준다. 이때 0이 아닌 값이 나올 때까지 데이터를 읽어 들이고 어떤 값부터 0이고 그 이후 모든 값이 0이면 마지막 0이 아닌 값까지만을 데이터로 이용하고 나머지는 사용하지 않는다. 복원할 때는 블록의 크기를 알고 있기 때문에 데이터의 나머지 항은 0으로 채워 $n \times n$ 길이의 데이터로 바꾼 뒤 역 scan을 한다.

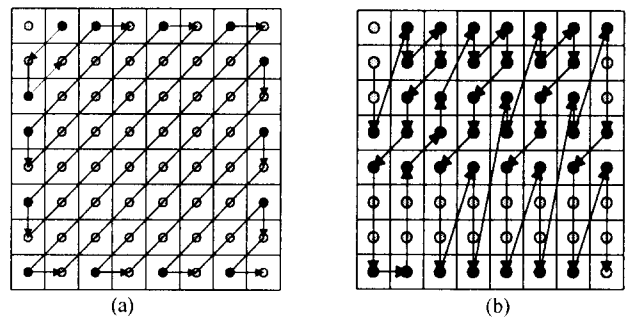


그림 5. (a) 지그재그 스캔 (b) 대체 스캔
Fig 5. (a) Zigzag scan (b) Alternate scan

2.4 시스템의 구성

비디오용 RAMDAC을 이용하여 CCTV카메라로부터 얻은 영상을 TMS320C80을 이용하여 DCT 인코딩을 하고, 코딩된 데이

터는 역 DCT코딩을 하여 RAMDAC으로 다시 보낸다. 그러면 RAMDAC을 통하여 모니터에 영상이 보여진다. 이때, 카메라로부터 얻어지는 영상은 640×480크기의 흑백 영상이고 NTSC 비디오 형식이다.

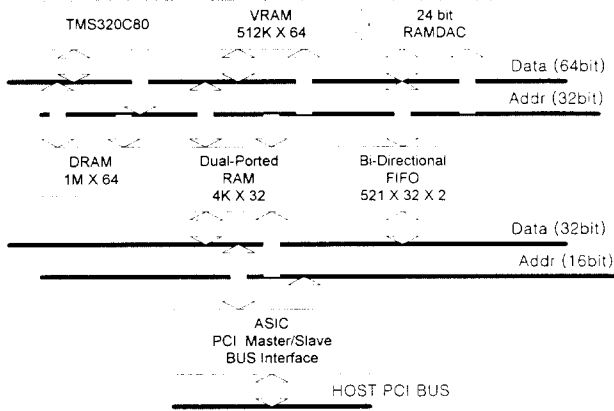


그림 6. DCT인코더 구성도
Fig 6. DCT encoder block diagram

TMS320C80 multimedia video processor(MVP)는 32bit의 RISC구조를 가지고 있는 master processor (MP)와 32bit의 ADSP(advanced DSP)구조를 가진 4개의 parallel processor (PP), transfer controller (TC), video controller (VC), 50K byte의 SRAM으로 구성되어 있는 DSP 칩이다. 4G-byte의 메모리 영역이 있고, memory-mapped I/O로 구성되어 있다.

그림 [6]은 인코더의 구성도이다. NTSC 영상을 받기 위한 RAMDAC과 4M byte의 비디오 RAM, 주 프로세서인 TMS320C80, 640×480크기의 이미지를 처리하기 위한 8M byte의 메모리, 코딩된 데이터 전송을 위한 양 방향 FIFO 그리고 PCI 버스와의 통신을 위한 PCI Bus interface 등으로 구성되어 있다.

2.5 영상의 평가

영상을 복원한 후에 평가하는 방법은 크게 2가지가 있다. 주관적인 평가 방법과 객관적인 평가 방법으로 나누어 지는데 주관적인 평가라 하는 것은 복원된 영상을 여러 사람에게 보여주고 사람들 각각의 주관적인 평가를 토대로 영상을 비교하는 것이고 객관적 평가라 하는 것은 원 영상과 복원된 영상과의 차이를 수식적으로 계산함으로써 비교하고 PSNR을 이용한다.

$$PSNR = 10 \log_{10} \left[\frac{255^2}{\frac{1}{NM} \sum_{m=1}^M \sum_{n=1}^N [(U(m,n) - U_f(m,n))^2]} \right] \quad (4)$$

여기서 M,N은 영상의 크기이고, $U(m,n)$ 은 원 영상이며 $U_f(m,n)$ 은 복원된 영상이다. PSNR의 단위는 dB이다. 이 두 가지 평가 방법은 어느 한 쪽에 치우쳐서는 안 된다. 예를 들어 영상의 어느 작은 한 부분만 손실되었을 경우 사람의 눈에는 그 손실된 부분으로 인하여 전체적인 영상을 나쁘게 볼 수도 있지만 PSNR을 구해보면 큰 값을 가질 수도 있기 때문이다. 반대로 DCT에서처럼 저주파 성분에 민감한 사람의 눈에는 영상이 좋게 보일 지도 모르지만 PSNR을 구해보면 낮은 값을 가지는 경우도 있다.

하지만 일반적인 영상에서는 PSNR이 높을 경우 사람의 눈에도 좋게 보인다. 다음의 그림 [7]은 bit rate를 고정시키고 각 bit rate에서의 PSNR값을 구한 결과이다. Rate-distortion 곡선에서처럼 bit-rate가 높으면(압축율이 낮으면) PSNR값도 높게 나오고 bit-rate가 낮으면(압축율이 높으면) PSNR도 낮은 값을 가진다.

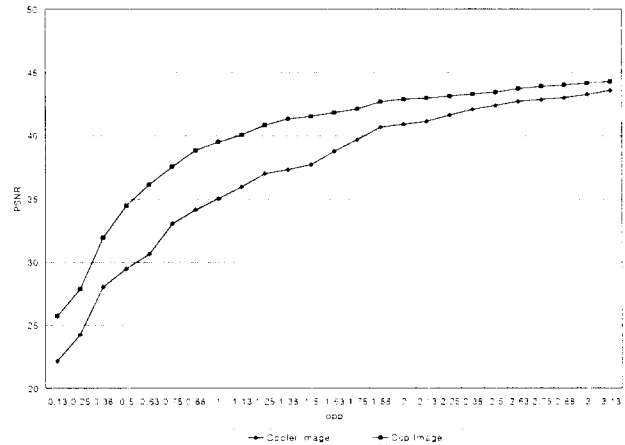


그림 7. PSNR과 bpp
Fig 7. PSNR and bpp

3. 결 론

3.1 실험 결과

앞에 제시한 DCT알고리즘을 이용한 실험은 window95 환경에서 visual C++로 구현하였다. 그림 [8]은 DCT coding전의 원 영상이고 그림 [9]는 DCT coding과 양자화(quantization)를 한 후 다시 복원한 영상이다. Quantization은 앞의 그림 [4]에서 제시한 quantization 테이블을 이용하여 코딩한 후 복원한 영상이다. Cup 영상의 경우 PSNR값은 약 38.77dB이고 0.65bpp로 압축이 되었다. Cooler 영상은 36.49dB이고 1.07bpp로 압축된다. 그림 [10]은 DCT코딩을 하기 전과 후의 영상의 차이 값인데 영상을 보면 주로 고주파 성분인 경계의 정보를 가지고 있음을 알 수 있다.

표 [1]은 그림 [4]의 quantization table을 사용하지 않고 bit rate를 미리 정하고 양자화하여 얻은 결과이다. 전체적으로 cup 영상이 cooler 영상보다 PSNR값이 높게 나왔다. 그 이유는 영상내의 경계가 cup 영상보다는 cooler 영상이 더 확실히 보이고 영상내 물체의 경계가 복잡하기 때문이다.

표 1. DCT 결과
Table 1. Result of DCT

Bit rate (bits/pixel)	Cool PSNR (dB)	Cup PSNR (dB)
0.75	33.08	37.59
1.00	34.80	39.44
1.50	37.88	41.83

앞에서 제시한 알고리즘을 TMS320C80을 이용하여 구현하였다. RAMDAC을 통하여 영상을 받고 TMS320C80을 이용하여 DCT 인코딩과 디코딩을 한 후 비디오 메모리에 영상을 전송하여 영상이 RAMDAC을 통하여 모니터에 나타나도록 하였다. TMS320C80에서 계산할 때 모든 값을 정수로 계산하면 영상을 복원할 때는 약간의 블록화가 보이게 된다. 물론 VQ를 하지 않

으면 블록화 현상은 육안으로 구별이 거의 불가능하다. 화면 (640×480)에 원 이미지와 코딩 후 이미지를 모두 보이기 위해서 이미지의 크기는 320×240으로 하였고, 이때 실제 구현에 걸리는 시간은 인코딩과 디코딩 과정을 합해서 약18초 정도이다.

3.2 추후 과제

영상을 코딩하는데 있어서 많은 메모리를 요구하게 되고 여기서 메모리 할당이 잘못되면 프로그램은 폭주하게 된다. 실제 알고리즘을 적용시켰을 때 가끔 이러한 현상을 보인다. 또 다른 문제점은 인코딩과 디코딩의 계산량이 많다는 점이다. 지금은 프로그램이 MP로만 동작을 하고 있고 약 18초 정도의 계산 시간을 보이는데 알고리즘을 분할하여 PP를 사용한다면 좀 빠른 계산이 가능 하리라 예상된다.

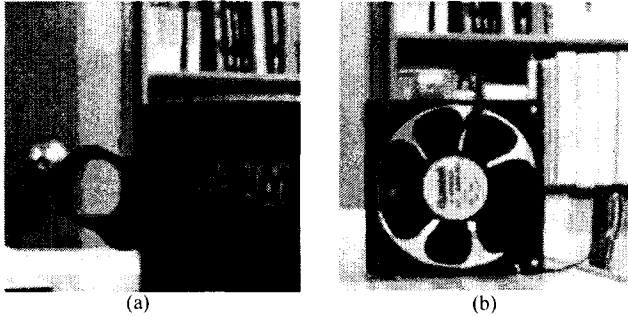


그림 8. 원 영상 (a) Cup (b) Cooler
Fig 8. Original Image (a) Cup (b) Cooler

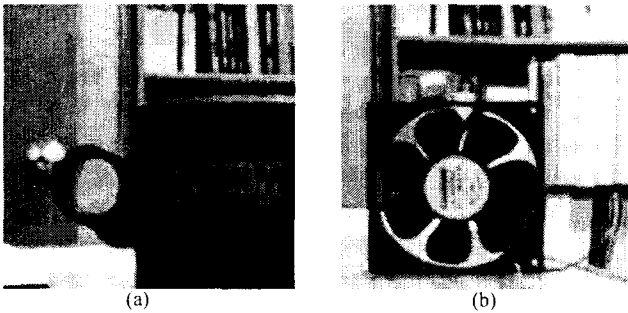


그림 9. 복원된 영상 (a) Cup (b) Cooler
Fig 9. Reconstructed Image (a) Cup (b) Cooler

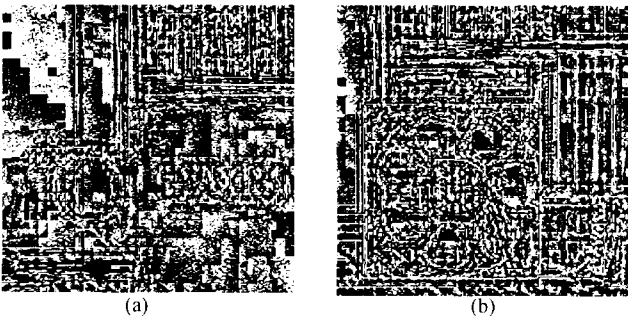


그림 10. 차분 영상 (a) Cup (b) Cooler
Fig 10. Difference Image (a) Cup (b) Cooler

[참 고 문 헌]

[1] K. R. Rao, P. Yip, "Discrete Cosine Transform Algorithms. Advantages. Applications", Academic Press, 1990.
[2] A. Murat Tekalp, "Digital Video Processing", Prentice Hall, 1995.

[3] Majid Rabbani, Paul W. Jones, "Digital Image Compression Techniques", SPIE Optical Engineering Press, 1991.
[4] Khalid Sayood, "Introduction to Data Compression", Morgan Kaufmann Publishers, 1996.
[5] K. R. Rao, J. J. Hwang, "Techniques and Standards for Image Video and Audio Coding", Prentice Hall, 1996.
[6] Phillip E. Mattison, "Practical Digital Video with Programming Examples in C", John Wiley & Sons, 1994.
[7] Prabhat K. Andleigh, Kiran Thakrar, "Multimedia System Design", Prentice Hall, 1996.
[8] "TMS320C80(MVP) User's Guide", Texas Instruments, 1995.
[9] "User's Guide for the GRIFFIN 100 (TMS320C80 PCI-Bus Board)", Ariel Corporation, 1996.