

DV에서 MPEG-2의 인트라 변환 부호화 방식의 연구 및
색차포맷 변환부의 H/W구현

이선행, 김도년

연세대학교 전기전자공학과

e-mail : sunhange@hanmail.net

Intra Transcoding from DV to MPEG-2 and
chrominance format conversion H/W implementation

Sunhang Lee, Donyeon Kim

Dept of Electrical and Electronic Engineering, Yonsei university

요약

디지털 캠코더에서 이용하는 영상 압축 방식인 DV 부호화방식은 DCT와 가변장 부호화 방식을 이용한다. DV 방식은 하드웨어 복잡도가 낮은 반면 압축된 비트율이 약 26Mbps로 높은 편이다. 따라서 스튜디오에서 낮은 복잡도로 영상을 부호화 한 후 VOD 시스템에서 사용하기 위하여 MPEG-2로 변환부호화 할 필요가 있다. 이때의 두 압축방식이 DCT를 이용하므로, DCT영역에서 변환부호화 하면 중간과정을 줄일 수 있어서 계산상의 복잡도를 줄일 수 있다. 본 논문에서는 DV방식에서 MPEG-2의 인트라로 변환부호화시, DV방식의 4:1:1 색차포맷을 MPEG-2의 4:2:2 색차 포맷으로 변환할 때 변환영역에 있는 데이터에 미리 계산된 행렬을 곱하여 병렬처리가 가능하게 설계하였다. 또한 MPEG-2 유효어는 중요한 서브 블록의 분산을 완전히 DCT영역에서 계산하여 하드웨어 복잡도를 줄였다. 색차포맷변환부 하드웨어 구현을 위하여 VHDL로 코딩한 후 FPGA-EXPRESS(synopsys), ALTERA MAX-PLUS II를 사용하여 모의실험을 하였다. 각 모듈별로 기능을 검증한 후, FPGA EXPRESS(synopsys)를 사용하여 합성 및 검증을 하였다.

코딩 방식에서 중복된 과정을 생략하여 구현 시 하드웨어 복잡도를 줄인다.

I. 서론

디지털 캠코더에 이용되고 있는 DV 압축방식[1]은 비트율이 26Mbps로 높아, 전송하거나 매체에 저장하기에 데이터량이 많다. 그러므로 기존의 디지털 캠코더로 촬영한 후 DV 방식으로 압축된 데이터를 MPEG-2[2]로 변환한 다음 비디오 서버에 저장할 필요성이 대두되었다. 그러는 과정에서 DV 인코딩, DV 디코딩, MPEG-2 인코딩, MPEG-2 디코딩 하는 복잡한 과정을 줄여서, DCT영역에서 변환하면 좀 더 빠르고, 좀 더 간단한 하드웨어 구조에서 더 많은 데이터량을 처리할 수 있다. 본 논문은 DV에서 MPEG-2로 변환 코딩할 때 DV 데이터를 부호화 한 후 다시 MPEG-2로 부호화하는 과정을 줄여서 소프트웨어로 변환할 경우 고속 워크스테이션을 이용하여 1분 영상을 변환하는 데 3시간에서 5시간 걸리는 것을 대폭 줄이는 방법을 찾는다. 또한 하드웨어로 구현 시에 계산량 및 복잡도를 줄이는 방법으로, 색차포맷변환부 하드웨어 설계에 대하여 서술할 것이다. II.에서는 DV 압축방식 분석과 MPEG-2 인트라 부호화 방식으로 변환하는 것에 대하여 서술하고, III.에서는 색차포맷변환부 하드웨어 설계에 대하여 서술한다.

1. DV 압축 방식 분석

DV 방식 부호화기의 입력영상은 ITU-R 601 표준의 4:2:2 포맷이다. 압축시 영상의 색차신호 CR, CB를 가로로만 각각 2:1 다운 샘플링하여 4:1:1 영상포맷으로 변환한다. 각 화소는 A/D 변환기에서 0~255 사이 크기로 출력되지만, 이 값에서 128을 빼어서 -128~127 사이의 크기로 바꾼다. 영상신호를 압축하기 위하여 DCT, Run-Length 부호화, 가변장 부호화 과정을 수행한다.

2. 블록 재구성

두 코덱의 색차신호 포맷이 서로 다르며, 따라서 MB 구성이 서로 다르므로 두 코덱 사이에 MB 데이터를 변환시켜야 한다. 색차신호 성분을 1:2 업 샘플링한다.

3. 4:1:1 색차신호 포맷에서 4:2:2 색차신호 포맷으로 변환

MPEG-2에서 DV 방식에서 쓰는 색차신호 포맷인 4:1:1을 지원하지 않으므로 4:2:2 혹은 4:2:0로 변환하여야 한다. 본 연구에서는 4:2:2로 변환한다.

색차 성분인 C_R , C_B 로 구성된 서브 블록들 각각에 대하여 시간영역에서 가로방향으로 1:2 업 샘플링 하여야 한다. 즉 2

II. MPEG-2 인트라 부호화 방식으로 변환

DV 방식으로 압축된 영상신호를 MPEG-2의 인트라 (Intra: I) 픽처로 변환한다. 이때 가능한 DCT 영역 상에서 변환함으로써 두

차원 서브블록에 대하여 1차원으로만 업 샘플링 처리가 필요하다.

이렇게 하는 기본적인 방법은 고속 IDCT 및 DCT 알고리즘을 이용하는 것이다. 즉, 4:1:1 포맷의 DCT 데이터를 업 샘플링 처리하고, 다시 DCT 처리하는 것이다. 지금까지 알려진 가장 우수한 고속 DCT 알고리즘은 [3]의 방법이다. [4]에서는 DCT 영역에서 신호를 필터링하는 것에 대하여 소개한 바 있고, [5]에서는 이웃하는 두 N 포인트 DCT 데이터를 이용하여 N/2 포인트 이동된 DCT 데이터를 DCT 영역에서 얻는 연구를 소개하였다. 본 연구에서도 DCT 영역에서 포맷 변환하는 방법을 모색한다. 일차원 신호에 대한 1:2 업 샘플링 연산은 입력 데이터로부터 만든 (1×4) 크기의 열 행렬에 식 (1)과 같은 (4×8) 크기의 행렬 h를 곱하면 된다. 이미 DCT 처리된 입력 데이터로 만든 열 행렬 $X_{8 \times 1}$ 에 대하여 식 (3)의 행렬 H(1), H(2)를 곱하여, 1:2 업 샘플링한 데이터에 대한 DCT 처리한 결과를 얻는다.

$$Z_{16 \times 1} = \begin{bmatrix} S_{1,1} & 0 & 1,1 \\ 0 & S_{1,1} & 0 \end{bmatrix} h_{16 \times 4} S_{1,1}^{-1} X_{8 \times 1} = \begin{bmatrix} H_{1,1}(1) \\ H_{1,1}(2) \end{bmatrix} X_{8 \times 1}$$

(1)

$$h = \begin{bmatrix} 1 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

(2)

$$H(1) = \begin{bmatrix} 1 & 0.8022 & -0.1633 & -0.3426 & 0 & 0.09 & -0.0676 & -0.1108 \\ 0 & 0.4584 & 0.9435 & 0.4843 & -0.2452 & -0.317 & 0 & 0.1136 \\ 0 & -0.0721 & 0.0884 & 0.6238 & 0.7886 & 0.2779 & -0.2134 & -0.2296 \\ 0 & 0.0382 & 0 & -0.0649 & 0.2079 & 0.6144 & 0.5748 & 0.2158 \\ 0 & -0.013 & 0.028 & 0.0819 & 0 & 0.0365 & 0.3943 & 0.5225 \\ 8 & & & & & & & \\ 0 & 0.0128 & 0 & 0.0109 & 0.1389 & 0.1381 & -0.1715 & -0.3368 \\ 0 & -0.0013 & 0.0366 & 0.062 & -0.056 & -0.1788 & -0.0884 & 0.0485 \\ 0 & 0.0101 & -0.0074 & -0.058 & -0.0488 & 0.0057 & 0 & -0.0316 \end{bmatrix}$$

$$H(2) = \begin{bmatrix} 1 & -0.9756 & 0.1633 & 0.1956 & 0 & -0.1882 & 0.0676 & 0.0763 \\ 0 & 0.3739 & -0.9435 & 0.8722 & -0.2452 & -0.1442 & 0 & 0.17 \\ 0 & 0.0907 & -0.0884 & -0.314 & 0.7886 & -0.7416 & 0.2134 & 0.1358 \\ 0 & 0.0287 & 0 & -0.1628 & 0.2079 & 0.1222 & -0.5748 & 0.5757 \\ 0 & 0.0207 & -0.028 & -0.0163 & 0 & 0.1835 & -0.3943 & 0.3491 \\ 0 & 0.0064 & 0 & -0.0546 & 0.1389 & -0.1908 & 0.1715 & -0.0963 \\ 0 & 0.009 & -0.0366 & 0.0663 & -0.056 & -0.0132 & 0.0884 & -0.0874 \\ 0 & -0.0067 & 0.0074 & 0.0192 & -0.0488 & 0.0401 & 0 & -0.0204 \end{bmatrix}$$

(3)

여기서 $X_{8 \times 1}$ 은 DCT 변환된 4:1:1 데이터 이고, $S_{8 \times 1}$ 은 DCT 계수 행렬이고, $S_{8 \times 1}^{-1}$ 은 IDCT 계수 행렬이고, $Z_{16 \times 1}$ 은 DCT 변환된 4:2:2 데이터이다.

III. 색차포맷변환부 하드웨어 설계

4:1:1의 색차포맷을 4:2:2 색차포맷으로 변환하는 하드웨어를 설계하였다. 사용 프로그램은 Synopsys와 ALTERA MAX -PLUS II를 이용하여 VHDL 코딩하였다. 곱셈기와 메모리의 성능이 특히 중요하다. 이 논문에서 나는 여러개의 곱셈기를 사용하는 것을 배제하고, 하나의 곱셈기에 축적기와 가산기를 달아서, 결과값이 SISO(serial input serial output)하게 출력되게 설계하는 것을 제안한다. 그러면,수십만 게이트를 줄일수 있고,복잡도를 줄일수 있다. 출력 메모리는 4:2:2로 변화되어 계산된 행렬의 열 값, $Z_{16 \times 1}$, $Z_{16 \times 2}$, $Z_{16 \times 3}$

$Z_{16 \times 16}$ 이 한번에 출력되게 40 bit 16워드 RAM 사용을 제안한다.

1. 전체 블록도

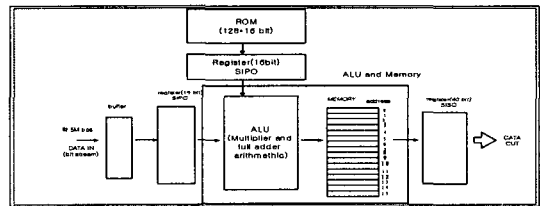


그림 1. 전체 블록도

Fig 1. Total block diagram

식 (3)과 같이 정확히 계산된 행렬 H(1), H(2)값은 표 1과 같이 이진수로 바뀌어져서 ROM(128*16bit)에 저장하게 된다. 그리고 입력값은 DCT로 변환된 4:1:1 데이터 값이다. 입력값은 빠른속도로 들어오기 때문에 버퍼에서 속도를 조절해야 한다. 입력값과 ROM (128*16bit)에 저장되어 있던 행렬 값이 ALU에서 계산되어서, DCT로 변환된 4:2:2 데이터 값이 출력되어 나온다. 그 값은 $Z_{16 \times 1}$, $Z_{16 \times 2}$, $Z_{16 \times 3}$ $Z_{16 \times 16}$ 등 과 같이 업 샘플링된 16×16 행렬에서 하나의 열값이 RAM(40bit*16)에서 출력하게 된다.

2. 모듈 블락 다이어그램

1) ROM(128×16 bit)

계산된 행렬 $H_{8 \times 8}(1)$ $H_{8 \times 8}(2)$ 는 표 1에 나타난 바와 같이 이진수로 고쳐져서 ROM에 저장되게 된다.

표 1. ROM(128×16bit)에 저장된 값들

Table 1. Binary indexes in the

ROM(128 - 16 bit)

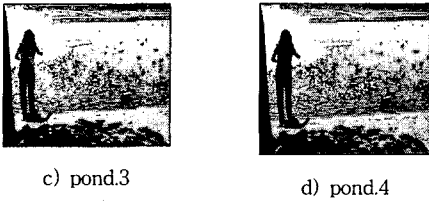


그림 7. DV코딩에서 복원된 영상비교
Fig 7. decoded image from DV coding



그림8. MPEG-2 display에서 보여지고 있는 4:2:2 로 색차포맷 변환된, 연속된 영상파일
Fig 8. Consecutive files converted into a 4:2:2 chroma format conversion are displayed

	POND.1	POND.2	POND.3	POND.4
DV 코딩	43.901400	43.857885	43.608341	43.872511
트랜스 코딩	32.639553	31.384643	31.227313	31.676599

표 2. 연속하는 시퀀스에서 비디오 코딩방식에 따른 Y(dB)의 PSNR 비교

	POND.1	POND.2	POND.3	POND.4
DV 코딩	39.163673	41.284883	40.324500	41.113355
트랜스 코딩	40.805340	40.378231	40.090254	40.112340

표 3. 연속하는 시퀀스에서 비디오 코딩방식에 따른 CB(dB)와 PSNR 비교

	POND.1	POND.2	POND.3	POND.4
DV 코딩	39.733856	41.441169	41.435690	41.323356
트랜스 코딩	29.592200	29.698000	30.676700	29.667217

표 4. 연속하는 시퀀스에서 비디오 코딩방식에 따른 CR(dB)의 PSNR 비교

트율이 26 Mbps인 DV 시퀀스를 10 Mbps인 MPEG-2 하이 프로파일 인트라 코딩의 시퀀스로 변환하였다. 휘도 신호와 색차 신호에 대한 PSNR(Peak Signal-to-Noise Ratio) 비교는 표 2.3.4 에 나타내었다. 표 2.3.4.에서 휘도신호(Y)의 PSNR에서 큰 차이가 나는것은 두 경우 모두 비슷한 압축 방식을 씀에도 불구하고, DV 코딩 경우는 26 Mbps로 압축하고 트랜스코딩 경우 MPEG-2 인트라 코딩방식으로 10 Mbps로 압축하였기 때문이다..

DV 방식 부호화기에서 MPEG-2로 변환하는 방안에 대하여 서술 하였다. 디지털 캠코더에서 DCT된 영상 데이터를 DCT 영역에서 MPEG-2의 인트라(Intra) 프레임으로 변환 및 압축하였으며, PSNR을 비교하였을 때 화질에 영향을 주지 않았다 DV 방식의 4:1:1 색차 포맷을 MPEG-2의 4:2:2 색차 포맷으로 변환할 때 고속 DCT 알고리즘의 Butterfly 구조를 사용하지 않고 병렬처리가 가능하고 보다 정확한 계산을 수행하는 행렬 이용방법을 제안하였다. DCT 영역에서 DCT의 에너지 정리와 DCT의 DC 계수를 이용하여, 시간영역에서 계산한 값과 동일한 m_quant를 계산하는 방법을 제안하였다. 시간영역에서 m_quant를 계산한 값과 DCT 영역에서 제안한 방법으로 계산한 값을 실험을 통하여 비교하였을 때 결과가 동일함을 확인하였다. DCT 압축방식을 이용하는 DV 방식을 MPEG-2로 변환 코딩할 때 하드웨어 혹은 소프트웨어 복잡도를 줄이기 위하여 DCT 계수를 이용한 고속 움직임 추정방법 들[3]을 이용하였다.

VI.참고문헌

- [1] Specifications of Consumer-Use Digital VCRs Using 8mm Magnetic Tape, HD Digital Conference, Dec. 1994.
- [2] ITU-T Recommendation H.262, July 1995.
- [3] W. Kou and T. Fjallbrant, "A Direct Computation of DCT Coefficients for a Signal Block Taken from Two Adjacent Blocks", IEEE Trans. of Signal Processing, Vol. 39, pp. 1692-1695, No. 3, July 1991.
- [4] J. B. Lee and B. G. Lee, "Transform Domain Filtering Based on Pipelining Structure", IEEE Trans. of Signal Processing, Vol. 40, pp. 2061-2064, No. 8, Aug. 1992.
- [5] U. V. Koc and K. J. Liu, "Adaptive Overlapping Approach for DCT-based Motion Estimation", Proc. of IEEE Inter. Conf. on Image Processing, Vol. I, Washington DC, pp. 223-226, Oct. 1995.
- [6] U. V. Koc and K. J. Liu, "Subpixel Motion Estimation in DCT Domain"
- [7] U. V. Koc and K. J. Liu, "DCT-based Subpixel Motion Estimation", Proc. of IEEE Int. Conf. Acoustics, Speech, Signal Processing, Atlanta, GA, Vol. IV, pp. 1931-1934, 1996.
- [8] U. V. Koc and K. J. Liu, "DCT-based Subpixel Motion Estimation and Fully DCT-based Video Coder", _
- [9] U. V. Koc and K. J. Liu, "DCT-based Motion Estimation", IEEE Trans. on Image Processing, Vol. 7, No. 7, pp. 948-965, July 1998.
- [10] Test Model Editing Committee, "Test Model 5", ISO/IEC JTC1/SC29/WG11/N0400, April 1993.
- [11] Jae S. Lim, "Two-dimensional Signal and Image Processing", Prentice Hall, p. 158, 1990.
- [12] P. Yip and K. R. Rao, "On the Shift Property of DCT' s and DST' s", IEEE Trans. of Acoust., Speech,