

MC-DCT 영역에서의 효율적인 비디오 축소 방법

이승철⁰, 이화순, 낭중호

삼성전자 {통신연구소⁰, 네트워크사업부}, 서강대학교 컴퓨터학과

{schlee⁰, trvhslee}@samsung.com, jhnang@ccs.sogang.ac.kr

An Efficient Video Scale-Down Method on MC-DCT Domain

Seung-Cheol Lee⁰, Hwa-Soon Lee, Jong-Ho Nang

{Telecommunication R&D Center⁰, Telecommunication system Division} Samsung Electronics, Department of Computer Science, Sogang Univ.

요 약

본 논문에서는 H.263 과 같은 MC-DCT 방법을 사용하는 코덱에서 IDCT 와 MC(Motion Compensation, 움직임보상) 방법을 부분적으로 수정하여 제한된 비율의 축소영상을 디코딩 과정에서 직접 얻을 수 있는 방법을 제안한다. 제안한 방법에서는 8×8 DCT 블록에 대하여 약간의 연산을 가하여 $k \times k$ IDCT 를 수행하고, 시간적 중복성을 제거하는 과정에서 근사적 움직임벡터를 이용한 $2k \times 2k$ MC 를 수행하였다. 기존의 코덱을 사용하고 후처리로 축소를 하는 경우와 비교하였을 때에 추가적인 오버헤드가 매우 경미하고 디코딩 과정이 많이 축소되어 계산효율에 있어서 대략 32%의 개선된 실험결과를 얻었고, 화질저하 현상이 개선됨을 실험적으로 보였다.

1. 서 론

휴대용 통신/디지털 기기의 소형화 및 고성능화가 진행되면서 CPU 및 이를 내장하는 SOC(System on chip) 내부의 system bus, memory, DCT / IDCT, ME / MC 등의 비디오 가속 하드웨어 블록의 성능이 크게 향상되면서 이를 활용하는 멀티미디어 어플리케이션의 성능이 비약적으로 발전하였다[1,2]. 한편, 물리적 크기의 소형화로 저해상도 디스플레이 장치를 통한 각종 모바일 콘텐츠의 고품질화의 진행은 휴대용 디지털 기기 내의 고용량 저장장치, 고속 다운로드 및 효과적인 축소 디스플레이 관련 기술을 절실히 요구하고 있다.

현재 상용화가 되었거나, 진행되고 있는 휴대전화 / PDA 의 경우 LCD 의 해상도가 대체로 128×96 에서 320×240 사이인데, 이를 이용하여 고해상도 비디오를 캡처 처럼 찍어서 저장, 확인하고, 복수의 콘텐츠(고해상도 JPEG 포함)를 thumbnail 형태로 축소하여 검색하기도 한다[1]. 그러나, 이러한 기능을 모바일용 시스템 자원을 이용하여 구현하는 데에는 여러 가지 제약조건이 뒤따르게 된다. 처리하고자 하는 데이터의 용량이 매우 크고 화질 저하를 수반하지 않고 영상을 축소하기 위해서는 요구되는 계산량이 상당히 크다. 따라서, 디스플레이를 위해서 별도의 영상축소 하드웨어 가속블록을 프로세서 내부에 구현해 넣게 되는 데, 이는 비용이 증가하고 제품의 소형화, 저발열, 배터리 지속시간 등에 걸림돌이 된다. 그렇다고 S/W 만으로 처리하게 될 경우, 영상신호의 불연속성이 지속적으로 나타나는 등의 화질 저하의 문제점을 노출하게 된다. 이러한 문제점을 해결하기 위

한 시도가 지속되어 왔지만, 공간영역-주파수영역간의 변환에 따른 오차에 의한 화질열화를 피하기 위하여 주파수 영역에서 영상처리를 하는 등의 기존의 연구는 디코딩 과정에 추가적인 복잡한 처리과정을 삽입하여 계산효율에 있어서 얻는 효과가 적었으며, 공간영역에서의 행렬계산을 DCT 영역으로 옮겨 놓았기 때문에 화질 왜곡의 근본적인 문제는 해결되지 않았다[4, 5].

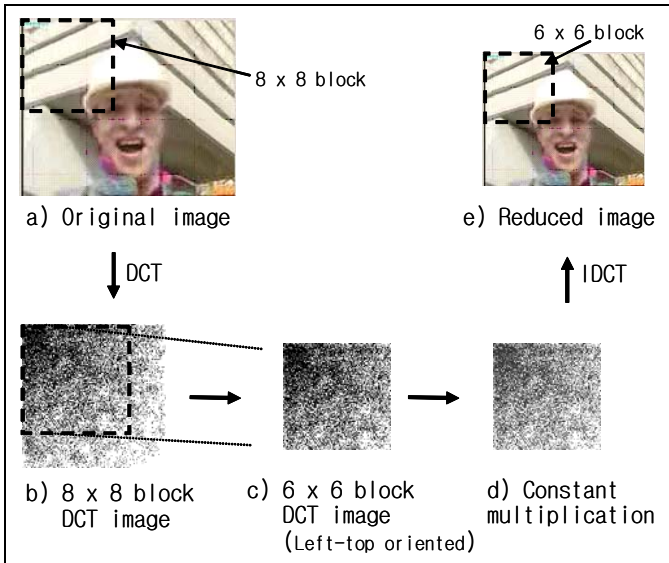
본 논문에서는 MC-DCT 를 사용하는 비디오 및 정지 영상 디코더 내부를 미리 지정된 8 가지 축소비율에 맞게 최적화하여 화질저하를 미연에 방지하면서 처리시간에서도 큰 이득을 얻을 수 있는 방법을 제안한다. 이를 위하여 IDCT 로 입력되기 전의 데이터를 적절하게 가공하고, IDCT 를 수정하였으며, 축소비율에 맞는 움직임 보상방법을 제안하였다. 그리고, 이 제안한 방법 중 IDCT 관련 부분을 구현하여 실제로 테스트해 보고, 기존방법과 화질과 성능 측면에서 큰 이득이 있음을 확인하였다.

2. MC-DCT 영역에서의 비디오 축소방법

일반적으로 인코딩된 비디오의 영상크기와 디코딩 후 재생시의 영상크기가 일치하지 않는 경우 디코딩된 raw image 를 확대 또는 축소하게 된다. 이 중 축소의 경우에 대하여 본 논문에서는 MC-DCT 가 적용되는 비디오 중 화상통화에서 사용하는 H.263[3]에 대하여 디코딩 과정에서 효과적으로 일정 비율($k/8$)로 축소하여 재생하는 방법을 제안하고자 한다. 이를 위하여 8×8 IDCT 및 16×16 MC 를 $k/8$ 의 비율형태로 적절하게 가공하여 디코딩 과정에 적용하고자 한다.

2.1 주파수 영역에서의 축소방법

화상통화에서 비디오 데이터의 DCT 영역은 8×8 크기의 2 차원 배열의 형태이다. 이를 DCT 영역에서 $k/8$ 의 크기(k 은 $1 \leq k \leq 8$ 인 정수)의 제한된 크기로의 축소효과를 얻기 위해서 $k \times k$ IDCT 방법을 8×8 IDCT 대신 적용하게 된다. 즉, 각 블록의 크기를 작게 만들어서 전체의 크기를 일정 비율로 작게 만드는 것이다. <그림 1>은 $k=6$ 인 경우에 대한 예이며 제안하는 방법의 처리 순서를 나타낸다. 이는 $k \times k$ IDCT에 적용하기 위한 전처리 과정이라 할 수 있다. 먼저 a)는 공간영역 상의 영상이며, 이를 DCT 한 결과가 b)이다. 이 b)를 가시적으로 나타낸 것은 좌측 상단의 DC를 중심으로 한 energy compaction 현상을 보기 위한 것으로, DC 값을 기준으로 $k \times k$ 영역을 새롭게 잘라서 c)를 얻게 된다. 축소된 영상에서는 $k \times k$ 영역 밖의 고주파 성분은 표현할 수가 없기 때문에 이를 제거하는 것이다. 이는 256 레벨의 각 공간영역의 영상 데이터에 대하여 DCT를 할 때에 가로 및 세로 방향의 1 차원 DCT를 하는 과정에서 8개의 화소에 대한 누적합이 이루어진 반면, 새롭게 적용할 6×6 IDCT에서는 6개의 화소에 대한 누적합에 대한 역함수가 적용될 것이기 때문이다. d)는 c)에서 $k/8$ 을 각 $k \times k$ 개의 DCT 계수들에 곱하여 전체적인 위상을 축소비율 만큼 감소시켰다. 그림상에서 c)에 비해 d)가 흐려진 것은 DC를 중심으로 각 계수들의 절대값의 크기가 작아진 것을 나타낸 것이다.



<그림 1> $k \times k$ IDCT 적용 전처리 과정 ($k=6$)

$$f(x, y) = \frac{2}{k} \sum_{u=0}^{k-1} \sum_{v=0}^{k-1} C(u)C(v)F(u, v) \cos \frac{(2x+1)u\pi}{2k} \cos \frac{(2y+1)v\pi}{2k}$$

$$C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & u, v = 0 \\ 1 & \text{otherwise} \end{cases} \quad \text{<수식 1>}$$

<수식 1>은 $k \times k$ IDCT의 일반적인 형태이다. 동일 비율로 재생하는 경우에는 $k=8$ 이 적용되고, 축소하는 경

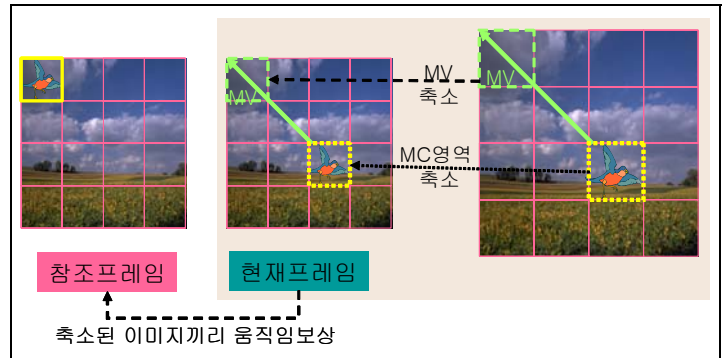
우에는 8 이하의 정수가 적용된다. 공간상의 좌표 x, y 및 이를 인자로 하는 함수 $f(x, y)$ 는 공간영역에서의 영상을 나타내고, 주파수 공간상의 좌표 u, v 를 인자로 하는 함수 $F(u, v)$ 는 주파수 영역에서의 영상을 나타낸다. 이 식에서 $k=6$ 의 경우의 예를 기준으로 계산횟수를 생각해 보면, $8 \times 8=64$ 회에서 $6 \times 6=36$ 회로 감소하게 되어 대략 56%로 처리횟수가 줄어든다. 이는 부가적으로 메모리 접근횟수 및 사용의 감소도 가져오게 되며, 이를 일반화 하면 $\left(\frac{k}{8}\right)^2$ 만큼의 계산 복잡도 감소를 가져오게 된다. 따

라서 제안하는 방법은 비록 전처리 과정에서 각 DCT 계수들에 대하여 일정한 상수의 곱셈을 해야 하는 오버헤드가 있지만, 충분히 이를 상쇄하고도 효율적임을 알 수 있다.

한편, 화질에 있어서 본 방법은 기존의 보간법을 비롯한 각종 공간영역에서의 영상 축소방법과 비교해 보았을 때에 매우 화질저하가 적다. 기존 방법은 공간영역 상에서 왜곡이 일어나는 부분이 평이한 곳이든, 복잡한 곳이든 상관없이 일정한 간격에 따라 처리를 했다. 반면 주파수 영역에서 전체적인 위상을 $k/8$ 크기로 줄인 상태에서 고주파성분들의 상당 부분은 0 이거나 매우 작은 값을 가질 가능성이 많다. 이 값들을 축소 비율에 맞게 제거하고 다시 공간영역으로 복원하였을 때의 영상은 전자에 비하여 화질 왜곡의 가능성이 매우 줄어든다. 최악의 경우 모든 주파수 계수가 값을 가지는 매우 한정된 영상에 대해서만 기존 방법과 비슷한 결과를 얻을 것이다.

2.2 축소된 비율에 대한 움직임보상 방법

지금까지 비디오에서 intra-coded 형식의 프레임에 대한 처리 방법을 설명하였다. 이 처리방법은 MJPEG 과 같은 동영상이나, JPEG 과 같은 정지영상에 대하여 완벽하게 처리할 수 있다. 한편 비디오에서 inter-coded 형식의 프레임은 상기 방법만으로 얻을 수가 없기 때문에 인코딩된 DCT 블록에 대하여 축소된 IDCT 방법으로 raw image를 얻은 후, 전 단계에서 제안한 방법에 의해 얻어진 참조 영상을 축소된 움직임 벡터를 이용하여 참조하여 움직임 보상을 하게 된다.



<그림 2> 축소된 비율에 따른 MC 방법

<그림 2>는 $k=6$ 인 경우에 대한 움직임 보상 방법을 나타내고 있다. 참조프레임이 축소되어 있으므로, 오른

쪽의 현재프레임을 구성하는 residual image decoding 및 움직임 보상방법을 각각 축소 비율에 맞게 새로운 움직임 벡터(MV')와 움직임 보상영역(MC')을 다음과 같이 적용하여야 한다.

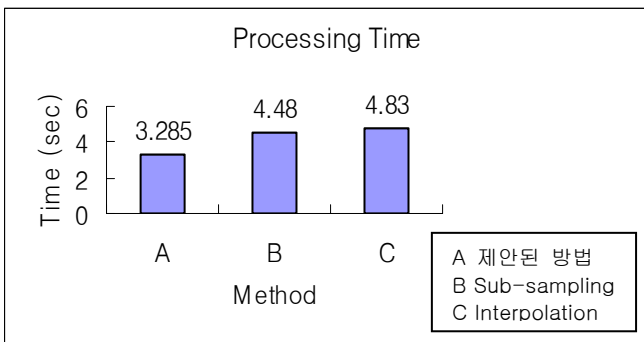
$$MV' = k/8 \times MV \text{ (MV'는 0.5 단위의 근사값)}$$

$$MC' = 2k \times 2k$$

여기서 새로운 움직임 벡터가 근사값이므로, 오차에 따른 화질열화가 발생할 수 있으나, 이는 반화소 단위의 움직임 보상시 참조 프레임의 영상을 보간하는 과정을 거치면서 최소화 된다. 반면 메모리 접근 횟수에 있어서 $8 \times 8 - k \times k$ 만큼의 비율로 이득을 얻게 되어 매우 효율적이다.

3. 실험을 통한 성능비교

본 논문에서 제안한 방법을 이용해서 실제 화상통신이 실행 가능한 휴대전화에 내장된 H.263 디코더를 수정하여 실험을 하였다. 구현의 복잡성 때문에 제안한 모든 기술을 평가하지는 못하고, intra-coded 형식의 프레임에 대하여 제한적인 성능을 측정할 수 있었다. IDCT는 <수식 1>을 구현하였으며, H.263 프레임 디코딩 평균 소요시간을 측정하였다. 임의의 프레임을 골라 원본크기 및 제안된 방법으로 축소된 raw image 를 추출하고, 원본크기 영상을 다양한 방법으로 축소하여 두 축소 영상끼리 PSNR 을 구하였다. 테스트 환경은 CPU ARM920t 135Mhz, System 33.6Mhz 으로 세팅하였으며 운영체제는 pSOS v. 2.5 를 이용하여 구축하였다.

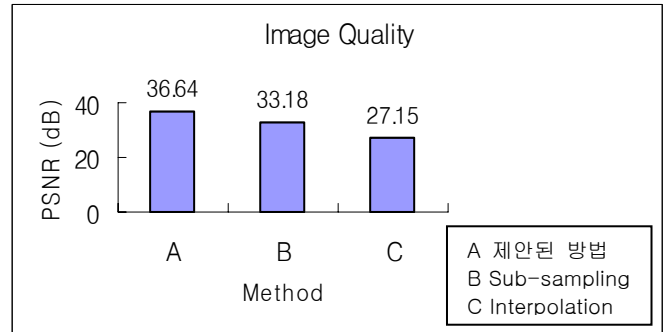


<그림 3> Intra 프레임 평균 생성 시간

샘플로 QCIF Foreman 을 사용하였으며, intra-coding 은 매 10 번째 마다 이루어지도록 했으며, 총 100 프레임을 생성하여 이를 디코딩하였다. <그림 3>에서 방법 A 는 제안하는 방법을 구현한 것이고, 나머지는 일반적인 방법으로 디코딩 후 출력부분을 sub-sampling(방법 B), interpolation(방법 C) 방법으로 75%의 비율(k=6)로 축소하였으며 intra-coded 형식 10 프레임만의 재생시간을 비교하였다. IDCT 가 전체 디코딩 과정의 상당한 비중을 차지하였고, 방법 1 은 후처리 과정이 없기 때문에, 가장 느린 방법 3 의 68%에 불과한 수행시간을 얻었다.

화질에 있어서는 QCIF 출력 10 프레임을 상용 유틸리티(Paint Shop Pro 5.0)[6]를 이용하여 bilinear resampling 방법으로 기준이 되는 축소 영상을 만들고, 각각의 방법에 의해 생성되는 프레임과 비교하였다. <그림 4>는 각

축소방법에 의해 생성한 출력영상의 평균 PSNR 값을 나타내었다. 이 결과가 원본에 가장 가까운 축소영상이라고 단정하기는 어렵지만, 실시간 특성을 지니는 무선화상통신의 출력물이 PC 상에서 생성한 축소영상과의 비교에서 36dB 이상의 화질을 유지했다는 점에서 긍정적인 결과라 할 수 있으며, 비교대상의 다른 방법보다 화질에서 큰 차이를 보이고 있다.



<그림 4> 화질 비교 결과

4. 결론

휴대용 멀티미디어 기기에서 저해상도 디스플레이 장치를 통해 고품질의 멀티미디어 콘텐츠를 재생하기 위해서 효과적인 영상 축소 방법이 요구된다. 제한된 시스템 자원을 효율적으로 사용하기 위하여 MC-DCT 방법을 이용하는 코덱의 디코딩 과정을 수정하여 보다 적은 계산량으로도 더욱 좋은 화질의 영상을 얻을 수 있는 방법을 제안하였다. 이의 구체적인 설명과 검증을 위하여 현재 상용화 된 화상통화가 가능한 휴대전화의 H.263 코덱을 테스트 모델로 하였으며 제한적이거나 제안한 기술에 대하여 실험결과를 얻었고, 현재 일반적으로 구현되고 있는 축소방법에 대하여 최대 32%의 효율성이 있음을 확인하였다. 또한 화질에 있어서도 PC 에서 상용 유틸리티를 이용한 축소영상과 매우 근접한 PSNR 값을 얻을 수 있었다.

향후에는 근사된 움직임 벡터를 이용한 MC 에 관한 실험을 보강하여 제안한 기술에 대한 검증을 보다 구체적으로 실행할 필요가 있고, 이와 관련한 새로운 MC 방법의 개선과 제안이 이루어져야 하겠다.

참고문헌

- [1] Samsung Electronics Co. Ltd. <http://www.sec.co.kr>.
- [2] ARM Ltd. <http://arm.com/products/CPUs/families.html>.
- [3] ITU-T Recommendation H.263, Video Coding for Low Bit Rate Communication.
- [4] S.-F. Chang, W.-L. Chen and D.G. Messerschmitt, " Video Compositing in the DCT domain, " Proc. of IEEE workshop on Visual Signal Processing and Comm., Raleigh, NC, pp. 138-143, Sep. 1992.
- [5] S.-F. Chang and D.G. Messerschmitt, " A New Approach to Decoding and Compositing Motion compensated DCT-Based Images, " Proc. of IEEE ICASSP, Minneapolis, Minnesota, April. 1993.
- [6] Jasc Software, Inc <http://www.jasc.com> .