

핸드폰에서의 소프트웨어 동영상 인코더 구현

한상범, 정진환, 유혁
고려대학교 컴퓨터학과
e-mail : sbhan@os.korea.ac.kr

Cases of Software Video Encoding on Cellular Phone

SangBeom Han, JinHwan Jeong, Hyuck Yoo
Dept of Computer Science and Engineering, Korea University

요 약

셀룰러 네트워크의 발전과 카메라 부착이 가능한 핸드폰 증가와 같은 발전에 따라서 핸드폰에서 비디오 인코딩을 가능하게 하는 핸드폰에 대한 수요가 증가하고 있다. 그러나 현재의 시스템은 별도의 하드웨어 부착 없이는 핸드폰에서 비디오 인코딩을 하는 것이 불가능하다. 이 논문에서는 별도 하드웨어 부착 없이 가능한 소프트웨어 인코더를 설계/구현하고, 그에 따른 실험 결과와 문제점을 보여 준다.

1. 서론

현재 셀룰러 네트워크는 급속한 변화를 겪고 있다. 2G 에서 2.5G 로 변했으며, 현재는 다시 3G 로 변화를 겪고 있고, 앞으로도 계속 변화가 계속될 것이다[1]. 이런 변화 속에서 사용자가 사용 가능한 대역폭도 현저히 증가하고 있다. 이런 늘어난 대역폭은 망 운영자들에게 새로운 서비스를 가능하게 만들고 있다[2]. 셀룰러 네트워크뿐만 아니라 핸드폰에서도 발전은 계속되고 있다. 디스플레이는 과거의 단순한 흑백에서 컬러 디스플레이가 점점 더 확대되고 있다[3][4]. 컬러 디스플레이의 증가는 멀티미디어 영상에 대한 사용자의 욕구를 급격하게 증가시키게 되었다. 네트워크와 핸드폰에서의 급격한 발전으로 인한 망 운영자들의 새로운 서비스 제공 욕구와 사용자의 멀티미디어 요구 증가는 Multimedia Messaging Service 나 무선 멀티미디어 스트리밍과 같은 새로운 서비스의 출현을 가능케 하였다[5].

이런 추세에 더불어 핸드셋에 카메라를 부착하는 것 역시 일반화되고 있다[6]. 앞에서 대역폭의 증가와 컬러 디스플레이의 일반화가 멀티미디어 재생을 위한 소프트웨어나 하드웨어를 요구하였다면, 카메라의 일반화는 멀티미디어 인코딩을 위한 소프트웨어나 하드

웨어를 요구한다.

이 논문에서는 현재 시스템에서 카메라를 통해서 영상을 받아들여 동영상으로 전송한 사례 연구를 통해서 어떤 문제점이 발생하는지와 이에 대한 해결책을 제시한다. 2 장에서는 현재 동영상 전송 기술의 표준으로 사용되고 있는 MPEG-4 를 살펴보고, 현재의 무선 환경에 적용했을 때의 문제점을 보이고, 3 장에서는 이런 문제점을 해결하기 위한 Light Weight Encoder 개념을 제시한다. 4 장에서는 제시된 인코더를 구현한 실제 환경과 목표 플랫폼을 보여주고, 5 장에서는 실험 결과를 보여준다. 마지막으로 제시된 시스템에서의 문제점과 향후 해결 방안을 다룬다.

2. MPEG-4

MPEG-4 Visual 이라고 알려진 MPEG-4 비디오 압축 표준은 현재 모바일 환경에서 동영상을 주고 받는 사실상의 표준이다[7]. MPEG-4 는 scalability, error resilience, object-based coding 와 같은 주요한 기능을 제공한다. 이 중 scalability 는 서버에서 클라이언트까지의 대역폭이 계속적으로 변화하는 것을 허용한다. Error resilience 기능은 전송 중 네트워크에서 에러가 발생하더라도 에러를 극복할 수 있는 수단을 제공하는

다. MPEG-4 의 이 두 가지 특징은 계속적으로 대역폭이 변화하고 에러가 발생할 가능성이 많은 2.5G 나 3G 와 같은 무선 네트워크를 통해 비디오를 전송할 때 MPEG-4 를 적합하게 한다. 이런 특징들로 인해서 현존하는 시스템들은 대부분 MPEG-4 를 기반으로 한 시스템이 되고 있으며[8], 앞으로도 최종적인 형태는 MPEG-4 를 기반으로 하거나, 그 뒤를 이어받은 JVT 를 기반으로 한 시스템이 결국에는 채용될 것이다[9].

MPEG-4 표준은 셀룰러 네트워크에 알맞은 이런 특징을 가지고 있지만, 그에 맞지 않는 특징 역시 가지고 있다. 즉, MPEG-4 표준의 복잡도가 지나치게 커서 셀룰러 폰 안의 CPU 로는 처리하기가 불가능하다[10]. 그래서 대부분의 업체에서는 MPEG-4 자료만을 처리하기 위한 전용 칩을 별도로 두고 있다[11][12][17]. 이런 해결책을 채용함으로써 인해서 문제점이 발생하였다.

즉, MPEG-4 표준을 채용한 핸드셋간만 상호 동영상 전송이 가능하다. 다시 말하면 동영상은 인코딩하여 보내더라도 상대방 핸드폰에 MPEG-4 하드웨어가 장착되어 있지 않다면 재생이 불가능하다. 또한 별도의 하드웨어를 부착함으로써 인해서 모바일 디바이스에서 중요한 전력을 더 많이 소비하게 된다. 마지막으로 현재 대두되고 있는 라이선스에 관한 문제가 있다.

이 문제점을 해결하는 가장 좋은 방법은 MPEG-4 표준을 어떤 CPU 에서도 동작할 수 있도록 소프트웨어화하는 것이지만 이는 현실적으로 불가능하다.

3. Light Weight Encoder

모든 동영상 압축 방식은 압축률과 복잡도라는 두 가지 인자 사이의 끊임 없는 스펙트럼 사이의 어딘가에 존재한다. 그림 1 에서와 같이 MPEG-4 와 같은 경우 인코딩된 데이터의 크기는 최소한으로 하고, 사용된 CPU 는 최대한으로 하였다. 즉, 압축 효율을 증가시키기 위해서 입력된 비디오 데이터에 최대한의 가공을 하였다. 그러나 반대 쪽 끝에는 입력된 비디오 데이터에 전혀 손대지 않을 수도 있다. 즉, 압축률과 사용할 CPU 사이클 간에 균형을 잡는 것이다.

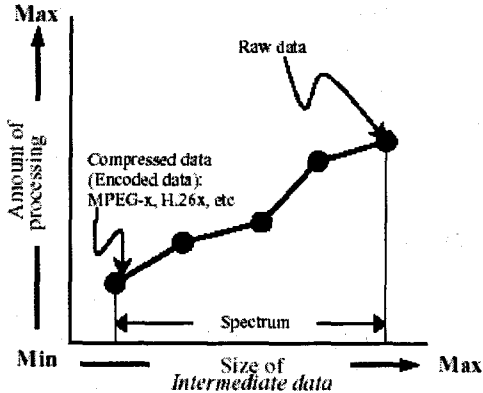


그림 1 인코딩된 데이터의 스펙트럼

MPEG-4 는 이 스펙트럼에서 CPU 사이클을 최대한

사용하여 최대한의 압축 효율을 얻었다. 그러면 셀룰러 네트워크의 대역폭과 CPU 속도의 발전 추이를 고려해보자. 그림 2 에서처럼 셀룰러 네트워크가 진화하면서 네트워크의 대역폭은 빠른 속도로 증가하고 있다. 그러나 핸드폰의 CPU 속도는 그리 빠른 속도로 증가하지는 않는다. 그림에서 보는 바와 같이 점점 네트워크 대역폭과 CPU 속도 차이의 폭은 커진다. 즉, 그림 1 에서 동영상 인코딩에서 복잡도를 조금 줄이고, 압축 효율을 낮추는 쪽이 나올 수 있다[13].

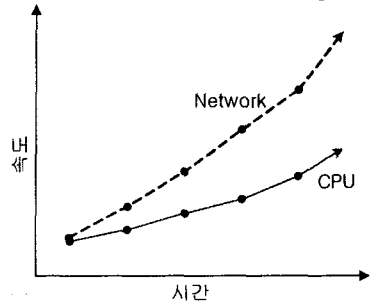


그림 2 네트워크와 CPU 의 속도 증가 추세

이 논문에서 설계/구현한 Light Weight Encoder 는 이런 접근 방식을 선택해서 MPEG-4 에서 복잡도가 높은 부분들을 제거하고, 압축 효율은 타당할 정도로 낮추는 방식을 선택하였다. 즉, MPEG-4 와 같은 Hybrid Transform 방식을 택하되 복잡도가 높은 부분은 복잡도가 낮은 다른 방식으로 교체하거나 제거하였다.

Motion estimation 과 같이 가장 CPU 부담이 큰 부분은 대폭 축소하여, Simple motion compensation 만을 하도록 하였다. 그렇게 해서 압축 효율은 떨어지겠지만, 그에 해당하는 CPU 사이클을 줄임으로서 인코더가 핸드폰에서도 동작할 수 있게 되었다. 또한 반복적으로 호출되는 DCT 부분은 연산량이 적은 Wavelet 함수로 교체하였다. 이런 방식으로 비교적 많은 CPU 사이클을 필요로 하는 루틴들을 모두 압축률은 낮지만 CPU 사이클이 적게 필요로 하는 루틴으로 교체하였다[14][15][16].

4. 구현 환경

구현 환경은 현재 서비스되고 있는 멀티미디어 서비스와 네트워크 상황을 고려하였다. 목표로 정한 네트워크 상황과 인코딩된 데이터를 받아서 디코딩할 핸드폰의 사양은 그림 3 과 같다.

망 종류	CDMA 2000 1X
망의 속도	144kbps
초당 표시할 프레임 수	3 fps
프레임 해상도	112*112
컬러 수	65000 Color

그림 3 목표로 할 핸드폰과 네트워크 상황

구현할 대상은 두 개 회사의 핸드폰을 사용하였다.

두 제품의 사양은 그림 4 와 같다. 두 개 제품 모두 시장에 출시하기 위한 모델의 프로토타입이다. 그림에서 보는 것과 같이 두 제품 모두 별도의 하드웨어 인코더 칩을 장착하지 않은 범용 핸드폰이다.

	A	B
CPU	MSM 5000	MSM 5500
LCD 크기	160*120	160*120
LCD 색수	4096	65000
LCD 종류	STN	TFT
카메라 해상도	352*288	640*480
카메라 색수	4096	4096
최고 지원 네트워크 속도	153.6bps/76.8kbps	2.5Mbps/307kbps

그림 4 인코더가 구현된 핸드폰의 사양

두 가지 핸드폰의 구조 차이는 핸드폰 A 의 경우 ASIC 에서 메모리로 직접 전송되는 방식을 취하고 있으며, 핸드폰 B 의 경우 ASIC 이 항상 LCD Controller 로 동영상을 전송하고 있으며 필요할 때 LCD Controller 를 멈추어서 해당 영상을 메모리로 전송하는 구조를 지니고 있다.

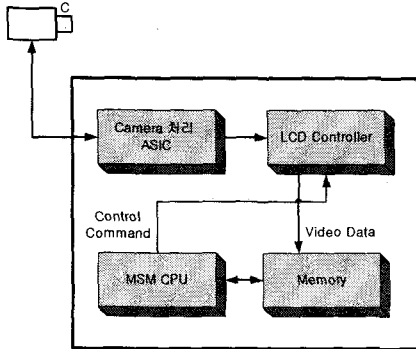


그림 5 핸드폰 A 의 구조

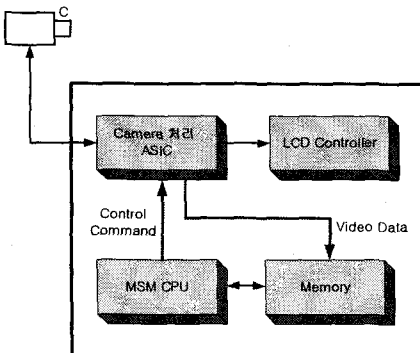


그림 6 핸드폰 B 의 구조

5. 성능 평가

실험에 사용된 핸드폰들은 핸드폰 본래의 기능인 통신 대기 상태를 그대로 유지한 상태에서 실험이 이루어졌다. 원래 실험 목적인 통신 대기 상태에서 동영상 실시시간으로 인코딩하여 저장 후 전송하거나 스트리밍할 상황을 가정한 것이다. 즉, 입력된 동영상은 계속 인코딩하는 상태에서도 백그라운드로 통신 대기 프로세스가 계속 작동하는 중이다. 그리고 인코딩 상태에서 전화 호출 메시지가 도착하면 인코딩 프로세스는 중지 또는 무효화되어 통화 상태로 전환된다.

그림 7 은 핸드폰에서 부착된 카메라를 통해서 핸드폰에서 인코딩된 이미지를 PC 로 전송하여, PC 에서 보여준 것이다. 이미지의 질을 비교하기 위해서 PSNR 과 같은 객관적인 수치가 필요하지만, 원본을 핸드폰을 통해서 저장하는 것이 사실상 불가능하므로 주관적인 비교를 하는 것만이 가능하다. 샘플 이미지를 봤을 때 화면의 일그러짐이나 깨짐 없이 비교적 고르게 인코딩되는 것을 볼 수 있다. 이미지의 색상은 PC 카메라에 비해서 그다지 좋지 않지만 원래 입력이 4096 색인 것을 고려하면 그다지 나쁘지 않으며, 또한 PC 가 아닌 화면의 질이 그리 좋지 않은 핸드폰의 LCD 디스플레이를 통해서 보게 되면 이미지의 질이 더 나아 보이게 된다.

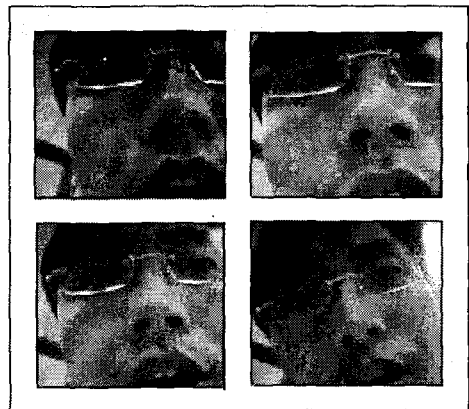


그림 7 인코딩된 샘플 이미지

5.1 사용된 CPU 시간

제대로 된 CPU 활용률을 알려면 CPU 점유율을 측정해야 하겠지만 핸드폰에서의 CPU 점유율을 측정하는 어려움이 많다. 그래서 실제로는 인코딩 루틴의 시작 부분에서 시간을 측정하고 끝 부분에서 시간을 측정하여 인코딩 타임의 시작과 끝을 측정하였다.

시간의 측정은 두 가지로 나누어 하였다. 그림 8 에서 보는 바와 같이 각 핸드폰 별로 Copy Only 경우와 Encoding 의 두 가지 경우로 나누어진다. Copy Only 는 카메라로부터 입력 데이터를 얻어온 후 MSM 칩이 인코딩을 위해 메모리로 옮겨오기만 할 때이다. Encoding 의 경우는 메모리로 옮겨온 후 실제 인코딩을 한 것이다.

두 핸드폰은 실제 한 프레임을 인코딩했을 때 평균 460ms 와 330ms 가 소비되었다. 초당 프레임 수로 환산하였을 때는 각각 2.1fps, 3fps 의 속도를 유지하였다.

각각의 프레임 인코딩에 걸린 시간을 분석해 보면 Capture 에 소비된 시간이 230ms, 180ms 이고, 실제 인코딩에 소비된 시간은 230ms, 150ms 이다. 즉, 캡처에 소비된 시간이 인코딩 시간과 거의 1:1 의 비율을 보인다.

또한 인코딩 시간에 대한 분석을 해보면 Copy Only 때의 시간 분석을 해 보면 인코딩을 전혀 하지 않고 Copy Only 에서 입력 데이터를 MSM 칩에서 얻어와 저장만 했을 때 걸린 시간이 각각 130ms, 50ms 를 차지한다. 즉, 인코딩 부분에서도 입력 자료의 복사가 50%, 30%의 시간을 차지함을 볼 수 있다. 다시 말하면 실제 인코딩에 소비된 시간은 25%, 30%에 불과하다.

실제 인코딩에 소비된 시간보다는 카메라에서 데이터를 캡처하는 시간이나 그것을 다시 MSM 칩에서 처리하기 위해 옮겨 오는 시간이 너무 과다함을 알 수 있다.

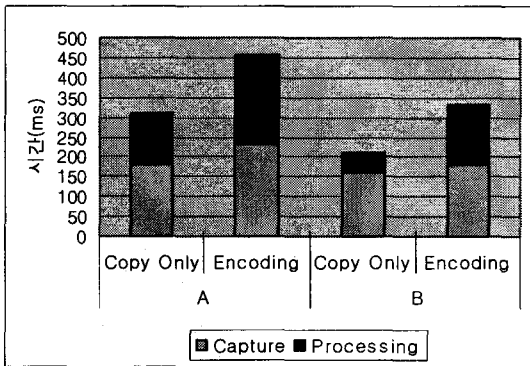


그림 8 한 프레임 인코딩에 걸린 평균 시간

5.2 데이터의 압축률

평균 데이터의 압축률은 3.5:1 이다. MPEG4 의 압축률에 비해서 형편 없이 낮은 수치이다. 그렇다면 목표로 한 네트워크 대역폭인 CDMA 2000 1X 의 대역폭과 비교해보자. 초당 3 프레임을 인코딩한다고 보면 3 프레임이 차지하는 대역폭은 102kbps 이다. CDMA 2000 1X 의 대역폭은 144kbps 이므로 초당 3 프레임의 동영상 자료를 전송하는 데는 아무 문제가 없음을 볼 수 있다.

6. 결론 및 향후 과제

사용된 CPU 시간을 분석한 결과 세 모델의 핸드폰 모두 카메라에서 핸드폰 안의 MSM 칩이 데이터를 처리할 메모리로 옮기는 데 상당한 CPU Cycle 을 사용하고 있다. 즉, 인코더 안으로 데이터로 옮기는 데 이미 상당한 CPU 사이클이 소비된 결과, 실제로 인코딩에 사용할 시간이 너무 짧다는 것을 알 수 있다. 즉,

우리가 제안한 구조에서는 CPU 의 성능보다는 카메라 입력 처리 ASIC 에서 메모리로 전송하는 버스의 대역폭이 문제가 됨을 보여준다. 그러므로 카메라와 동영상 캡처를 담당하는 ASIC 에서 카메라로부터 온 입력 자료를 어느 정도 미리 처리해주어서, 인코더로 입력할 입력 자료의 양을 줄여야 성능의 향상을 얻을 수 있다.

추후에는 카메라 입력 처리 ASIC 에서 카메라로부터의 입력 자료를 어느 정도 처리해서 메모리로 전송되는 자료의 양을 줄일 수 있는 아키텍처를 사용했을 때의 인코딩 효율이 어떤지 확인하는 연구를 할 계획이다. 덧붙여 앞에서 언급했듯이 네트워크 대역폭은 점점 더 늘어날 것인데 예상한 것처럼 대역폭이 늘어난 환경에서도 이런 해결책이 여전히 유효한지 연구할 계획이다.

참고문헌

- [1] Danis M, Fernandes J. Provision of sufficient transmission capacity for broadband mobile multimedia: a step toward 4G. IEEE Communications Magazine 2001; 39(8): 46-54.
- [2] Jukka Yrjanainen, Yrjo Newvo. Wireless meets multimedia. Wireless communications and Mobile Computing 2002; 2:553-562.
- [3] Kimmel J, Hautanen J, Levola T. Display technologies for portable communication devices. Proceedings of the IEEE 2002; 90(4): 581-590.
- [4] [Online]. Available: http://neasia.nikkeibp.com/nea/200105/peri_129439.html
- [5] [Online]. MMS Conformance Document. Available: <http://www.forumnokia.com>.
- [6] ISO/IEC 14496-2:1999, Information Technology - Coding of Audio-Visual Objects-Part 2: Visual, December 1999.
- [7] Samsung Electronics Co., Ltd. Website : <http://www.samsungelectronics.com>
- [8] Packet Video Website : <http://www.pv.com>
- [9] SK Telecom Web Site : <http://www.sktelecom.com>
- [10] Qualcomm Corp. Web Site : <http://www.qualcomm.com>
- [11] Joint Video Team(JVT) of ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, Joint Committee Draft, Advanced Video Coding, May 2002.
- [12] MPEG-4 의 복잡도를 측정 한 논문이 있어야
- [13] Jin Hwan Jeong and Chuck Yoo, A server-centric streaming model. Proceedings of 10th international workshop on Network and Operating System Support for Digital Audio and Video, Jun. 2000.
- [14] Amara Graps. An Introduction to Wavelets. IEEE Computational Science and Engineering. 1995
- [15] Jerome M. Shapiro. Embedded Image Coding Using Zerotrees of Wavelet Coefficients. IEEE transactions on Signal Processing. 1993
- [16] Colm Mulcahy. Image compression using the haar wavelet transform. Spelman Science and Mathematics Journal, 1(1):22-31, 1997.
- [17] [Online]. PacketVideo Technology White Paper. Available: <http://www.pv.com/prodtech/tech.asp>