

수신된 움직임 벡터를 이용한 적응적 블록 양자화 기반 분산 비디오 코딩 방법

준회원 민 경 연*, 정회원 박 시 내*, 남 정 학*, 종신회원 심 동 규*, 김 상 호***

Distributed Video Coding based on Adaptive Block Quantization Using Received Motion Vectors

Kyung-Yeon Min* Associate Member, Sea-Nae Park, Jung-Hak Nam* Regular Members, Dong-Gyu Sim**, Sang-Hyo Kim*** Lifelong Members,

요 약

본 논문은 분산 비디오 코딩을 위한 적응적 블록 양자화 기법을 제안한다. 제안하는 방법에서는 분산 비디오 부호기에서 보조정보 프레임을 생성하면서 예측된 움직임 벡터를 부호기에 보내줌으로써, 부호기는 큰 복잡도의 증가 없이 보조정보 프레임을 완벽하게 복원한다. 또한, 이렇게 복원된 보조정보 프레임과 원본 프레임의 차이를 적응적으로 블록별 양자화를 수행한다. 제안한 방법은 오류 발생 비율을 이용하여, 교차 확률에 따라 적응적으로 부호화함으로써 부호화 비트를 감소시킬 수 있는 특징이 있다. 제안한 방법은 부호기에서 교차 확률 및 교차된 비트의 위치를 알 수 있기 때문에, 채널 복호기의 오류 수정 능력에 맞추어 패리티 비트를 전송하여 낭비되는 비트의 양을 감소시킬 수 있다. 컴퓨터 시뮬레이션을 통하여 제안한 방법이 기존의 방법 대비 66% 비트율 감소를 얻었으며, 기존의 DVC 피드백 채널에 따른 지연을 대폭 감소시켰다.

Key Words : Distributed video coding, adaptive quantization, side information, crossover, feedback channel

ABSTRACT

In this paper, we propose an adaptive block quantization method. The propose method perfect reconstructs side information without high complexity in the encoder side, as transmitting motion vectors from a decoder to an encoder side. Also, at the encoder side, residual signals between reconstructed side information and original frame are adaptively quantized to minimize parity bits to be transmitted to the decoder. The proposed method can effectively allocate bits based on bit error rate of side information. Also, we can achieved bit-saving by transmission of parity bits based on the error correction ability of the LDPC channel decoder, because we can know bit error rate and positions of error bit in encoder side. Experimental results show that the proposed algorithm achieves bit-saving by around 66% and delay of feedback channel, compared with the convntional algorithm

1. 서 론

최근 프로세서, 저장장치 및 전송기술이 급격하

게 발달함에 따라, 멀티미디어 처리 기능을 가진 휴대 기기의 생산 및 보급이 활발하게 이루어졌다. 사

※ 본 연구는 한국산업기술평가위원회의 IT산업원천기술개발사업의 일환으로 수행하였음. [2009-F-032-01], 차세대 모바일 영상서비스를 위한 초경량 비디오 부호화 원천기술개발]

* 광운대학교 컴퓨터공학부(kymin@kw.ac.kr, psea1118@kw.ac.kr, qejixfyza@kw.ac.kr, dgsim@kw.ac.kr)

** 성균관대학교 정보통신공학부(iamshkim@skku.edu)

논문번호: KICS2009-12-602, 접수일자: 2009년 12월 3일, 최종논문접수일자: 2010년 1월 27일

용자는 휴대기기를 이용하여 멀티미디어 데이터를 취득하고, 콘텐츠를 제작 및 전송, 보급할 수 있게 된 것이다. 따라서, 제한된 연산 처리 능력 및 배터리 성능을 가진 환경에서의 사용을 위해, 짧은 부호화 시간과 낮은 전원소비가 가능한 부호화가 필요하게 되었다. 하지만, 기존의 표준 부호화 방법인 ISO MPEG-x 나 ITU-T H.26x 방법은 방송 서비스를 위해 연구, 개발된 기술로 하나의 서버에서 부호화한 데이터를 다수의 사용자에게 송신하는 스트리밍 방식에 최적화되어 설계되었다. 이에 따라, 부호화의 복잡도는 낮고 부호기의 복잡도가 높아, 부호화를 수행하는 동안에는 낮은 배터리 소비와 짧은 부호화 시간이 소요되지만, 부호화를 수행하는 동안 높은 배터리 소비와 긴 부호화 시간이 소요된다. 이러한 부호화 방법은 개개인의 콘텐츠 제작에서 이용되는 휴대 기기 장치에는 부적합하다¹¹.

분산 비디오 부호화는 부호기의 복잡도를 부호기로 분산시켜 부호기의 복잡도를 낮추는 방법으로써, 낮은 배터리 소비와 짧은 부호화 시간이 소요되어, 이동식 비디오 장치에 적합하다. 또한, 분산 비디오 부호화 기술은 낮은 부호화 복잡도가 소요된다는 장점을 이용하여, 높은 부호화 복잡도가 발생하는 다시점 비디오 부호화 방법에 응용되기도 하며, 부호화된 비디오를 서버에 전송하고, 이를 다양한 수신환경에서 부호화 할 수 있는 부호화 방식으로 변환하여, 수신 단말로 전송하여 소비하는 기술로 사용되기도 한다².

분산 비디오 부호화 방법은 입력 영상을 키 프레임과 WZ (Wyner-Ziv) 프레임으로 나눈 후, 키 프레임은 H.264/AVC 등 기존의 인트라 부호화 방식으로, WZ 프레임은 Slepian-Wolf 방식으로 부호화한다. Slepian-Wolf 부호화는 입력영상을 DCT (Discrete Cosine Transform) 변환과 양자화 한 후, Turbo 코드나 LDPC (Low-Density Parity-Check Accumulate) 코드 등으로 채널 부호화를 수행한다^{3,4,8}. 채널 부호화 및 복호화의 성능을 향상시키기 위해서는 오류를 정정하기 위한 패리티 비트의 양과 정확한 교차 확률을 알아야 한다. 하지만, 분산 비디오 부호화 방법에서 부호기에서는 보조정보 영상의 접근이 안 되고, 복호기에서 원본 영상의 접근이 불가능하기 때문에, 필요한 패리티 비트의 양과 정확한 교차 확률을 측정하는 것은 쉽지 않다. 따라서, 채널 부호화 및 복호화의 성능향상을 위하여, 복호기에서 원본 영상을 예측하거나, 부호기에서 보조정보를 예측하기 위한 연구들이 진행되고 있다. 하지

만, 기존의 복호기에서 원본 영상을 예측하는 방법은, 정확한 원본 영상의 예측이 어렵고, 필요한 패리티 비트의 양을 부호기에서 알 수 없다는 단점을 가지고 있고, 기존의 부호기에서 보조정보를 측정하는 방법은 복잡도 증가량과 보조정보 프레임의 예측 정도가 상반 관계를 이루고 있다는 단점이 있다^{4,10}.

제안하는 방법은 복호기에서 움직임 벡터를 부호기에 전송해 줌으로써, 큰 복잡도의 증가 없이, 복호기에서 생성된 보조정보와 똑같은 예측 프레임을 생성할 수 있다. 이를 이용하여 채널 복호기의 성능에 맞도록 필요한 패리티 비트의 양과 오류의 양을 조절해서 전송함으로써, 부호화가 성공할 확률을 높이고, 비트를 효율적으로 사용할 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 기존 연구에 대하여 살펴보고, III장에서는 제안하는 알고리즘에 대하여 자세히 소개한다. IV장에서는 제안하는 알고리즘에 대한 실험 방법과 실험 결과 및 그 결과에 대한 분석을 기술하고, 마지막으로 V장에서는 이 논문의 결론 및 향후 연구계획에 대해 논한다.

II. 기존 방법

기존 분산 비디오 부호화 방법에서 이용하는 LDPC나 Turbo 코딩 방법은 생성된 패리티를 한번에 모두 보내는 것이 아니라, 일부분씩 전송하도록 되어있다. Varodayan 등에 의해서 제안된 LDPCA 코드는 66회로 분리해서 일부분씩 전송하도록 되어 있으며, 복호기에서 복호화가 이루어질 때까지 피드백 채널을 이용하여 부호기에 패리티 비트 전송을 요청한다¹¹. 이때, 채널 복호기는 교차 확률에 따라 보조정보 프레임의 정정하는 오류의 양을 결정한다. 결과적으로 피드백 채널의 지연을 줄이고 정확하게 오류 정정을 위해서는, 원본 영상과 보조정보 영상의 차이를 알아야 한다. 이러한 이유로, 교차확률을 측정하는 방법이 활발히 연구되고 있으며, 교차확률을 측정하는 위치에 따라 크게 두 가지로 나눌 수 있다.

첫 번째로, 교차 확률을 복호기에서 측정하는 방법과 부호기에서 측정하는 방법이 있다. 복호기에서 측정하는 방법은 키 프레임과 보조정보 프레임을 이용한 방법으로써, 부호기의 복잡도의 증가가 전혀 없다는 것이 장점이다. 차의 형태가 라플라시안 분포를 따를 것이라는 가정에 의하여 수행된다. 하지만 복호기에서는 원본 프레임이 존재하지 않기 때문에, $t-1$ 키 프레임과 $t+1$ 키 프레임의 차를 이용

하여 라플라시안 분포의 분산을 측정하고 이를 이용하여 교차 확률을 측정한다^[4].

복호기에서 교차확률의 측정 효율을 높이기 위하여, 라플라시안 분포 대신 최대우도함수를 이용하는 방법도 제안되었다. 이 방법 역시 원본 영상대신 $t-1$ 키 프레임과 $t+1$ 키 프레임의 형태를 이용하여 사용한다^[5]. 이 방법들은 부호기의 복잡도의 증가가 전혀 없다는 장점을 가지고 있지만, 복호기에서 측정된 교차 확률임으로 부호기에 피드백으로 복호화가 수행이 완료될 때까지 패리티 비트를 요구해야 하기 때문에, 피드백 채널에 의한 지연이 이전 분산 비디오 코딩 방법과 똑같이 일어나고, 객체의 움직임이 빠르거나, 카메라 패닝 등이 있는 영상에서는 연속되는 프레임들 사이에도 많은 차이가 생겨 측정된 값이 정확하지 않다는 단점을 가지고 있다.

두 번째로, 부호기에서 측정하는 방법은 원본 프레임과 키 프레임들의 차를 측정하여 필요한 만큼의 패리티 비트를 전송하여 피드백 채널 없이 동작하도록 하는 장점을 가진다. 부호기에 존재하지 않은 보조정보 프레임을 $t-1$ 키 프레임과 $t+1$ 키 프레임의 선형 보간법을 이용하여 복원하고 원본 프레임과 차를 측정하는 것이다. 보조정보 프레임의 예측의 성능을 향상시키기 위하여, 약간의 복잡도의 향상을 감소하고 보조정보 프레임을 복호기에서 생성한 보조정보 프레임과 더욱 유사하게 생성하기 위하여 간단한 움직임 추정을 수행한다^[5-6].

이러한 방법은 피드백 채널이 필요 없기 때문에 피드백 채널에서 발생하는 지연을 막을 수 있다는 장점을 가지고 있으나, 객체의 움직임이 빠르거나, 카메라의 패닝 등이 나타나는 영상에서는 복호기의 보조정보와 많은 차이가 발생하게 되어, 비트의 낭비가 나타날 수 있으며, 교차 확률이 많은 차이를

나타내게 되어 복호화를 실패할 확률이 높아진다는 단점이 있다.

하지만 제안하는 방법은 부호기에서 예측하기 때문에, 부호기에서 모든 정보를 알 수 있으며, 큰 복잡도 증가 없이 복호기의 보조정보와 동일한 보조정보를 예측할 수 있다. 따라서, 측정된 교차확률이 정확하고, 한번에 부호화되는 양 안에 많은 오류가 분포되지 않게 조정할 수 있으므로 낭비되는 비트를 감소시키고 지연을 줄일 수 있다.

III. 제안하는 방법

기존 분산 비디오 부호화 방법은 부호기는 복호기의 보조정보 프레임에 대한 정보가 전혀 없고, 복호기는 원본 영상에 대한 정보가 전혀 없기 때문에, 효율적으로 비트를 사용하지 못한다는 단점이 있었다. 이러한 문제점을 해결하기 위해, 제안하는 방법은 복호기에서 보조정보를 생성하면서 측정된 움직임 벡터를 부호기로 전송하고, 이 움직임 벡터를 이용하여 부호기에서는 복호기에서 사용된 보조정보와 완벽하게 같은 보조정보를 예측하여 복원한다.

그림 1은 제안하는 방법의 블록도이다. 제안하는 방법은 크게, 블록별 적응적 양자화를 이용한 부호기와 움직임 벡터를 측정하는 복호기로 구성된다. 우선, 부호기에서 $t-1$ 키 프레임과 $t+1$ 키 프레임을 이용하여 보조정보 프레임을 생성하고, 움직임 벡터를 측정하여 피드백 채널을 통하여 부호기로 전송한다. 부호기는 수신된 움직임 벡터와 $t-1$ 키 프레임과 $t+1$ 키 프레임을 이용하여 보조정보 프레임을 복원한다. 이렇게 복원된 보조정보 프레임과 원본 프레임을 DCT 변환을 수행한 후 오류를 측정하고, 측정된 오류를 바탕으로, 교차 확률에 따라 적응적

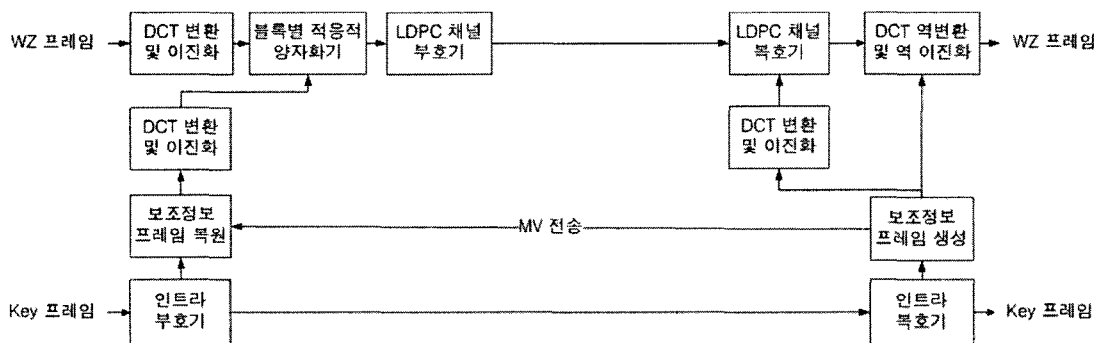


그림 1. 제안하는 방법의 블록도
 Fig 1. The block diagram of the proposed method

양자화를 수행한다.

일반적으로 비디오 방송 표준에서의 영상 간격은 1/60 (progressive: 1/30) 초로, 연속되는 프레임들은 평장히 적은 시간의 차를 가지고 있다고 가정하고 있다. 따라서 $t-1$ 키 프레임 안에 존재하고, $t+1$ 키 프레임에도 존재하는 객체 및 배경은 t 프레임에도 존재할 확률이 높다. 이러한 가정을 바탕으로, 연속적인 프레임들은 유사할 확률이 높기 때문에, 복호기에서는 $t-1$ 키 프레임, $t+1$ 키 프레임을 이용하여 보조정보를 생성한다.

제안하는 방법에서는 여러 방법을 이용하여 보조정보 프레임을 최대한 원본 영상과 유사하게 생성한 후, $t-1$ 시간의 키 프레임과 $t+1$ 시간의 키 프레임을 참조영상으로 이용해서 t 시간의 보조정보 프레임을 기준으로, 재 움직임 추정을 수행한다. 재 움직임 추정에 따른 복잡도가 증가하지만, 영상의 특성에 따라 원본 영상과 유사한 보조정보 프레임을 생성하는 방법이 다양할 수 있기 때문에, 보조정보 프레임을 생성하는 방법과는 별개로 시스템이 동작하도록 하기 위하여, 제안하는 방법은 재 움직임 추정을 수행한다.

그림 2는 재 움직임 추정을 수행하는 방법을 표현하고 있다. 보조정보 프레임을 블록 별로 나눈 후, $t-1$ 키 프레임에서 가장 유사한 블록을 찾고, $t+1$ 키 프레임에서도 가장 유사한 블록을 찾는다. 각 프레임에서 가장 유사한 블록이 찾아지면, 그 중 더 유사한 블록을 선택하여 하나의 블록 당 하나의 움직임 벡터를 가지도록 한다. 이렇게 추정된 움직임 벡터와 어느 프레임을 참조했는지를 가르쳐 주는 비트를 피드백 채널을 통하여 부호기에 전달한다.

3.1 복호기에서 움직임 벡터 추정

일반적으로 비디오 방송 표준에서의 영상 간격은 1/60 (progressive: 1/30) 초로, 연속되는 프레임들은 평장히 적은 시간의 차를 가지고 가정하고 있다. 따라서 $t-1$ 키 프레임 안에 존재하고, $t+1$ 키 프레임에도 존재하는 객체 및 배경은 t 프레임에도 존재할

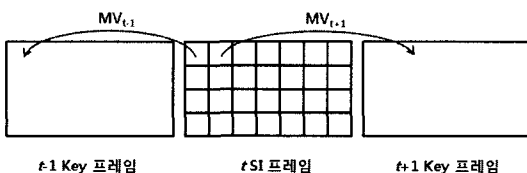


그림 2. 움직임 벡터를 측정하기 위한 재 움직임 추정 방법
Fig. 2. Re-motion estimation for measurement MV

확률이 높다. 이러한 가정을 바탕으로, 연속적인 프레임들은 유사할 확률이 높기 때문에, 복호기에서는 $t-1$ 키 프레임, $t+1$ 키 프레임을 이용하여 보조정보를 생성한다.

제안하는 방법에서는 여러 방법을 이용하여 보조정보 프레임을 최대한 원본 영상과 유사하게 생성한 후, $t-1$ 시간의 키 프레임과 $t+1$ 시간의 키 프레임을 참조영상으로 이용해서 t 시간의 보조정보 프레임을 기준으로, 재 움직임 추정을 수행한다. 재 움직임 추정에 따른 복잡도가 증가하지만, 영상의 특성에 따라 원본 영상과 유사한 보조정보 프레임을 생성하는 방법이 다양할 수 있기 때문에, 보조정보 프레임을 생성하는 방법과는 별개로 시스템이 동작하도록 하기 위하여, 제안하는 방법은 재 움직임 추정을 수행한다.

그림 2는 재 움직임 추정을 수행하는 방법을 표현하고 있다. 보조정보 프레임을 블록 별로 나눈 후, $t-1$ 키 프레임에서 가장 유사한 블록을 찾고, $t+1$ 키 프레임에서도 가장 유사한 블록을 찾는다. 각 프레임에서 가장 유사한 블록이 찾아지면, 그 중 더 유사한 블록을 선택하여 하나의 블록 당 하나의 움직임 벡터를 가지도록 한다. 이렇게 추정된 움직임 벡터와 어느 프레임을 참조했는지를 가르쳐 주는 비트를 피드백 채널을 통하여 부호기에 전달한다.

3.2 부호기에서의 보조정보 생성

분산 비디오 코딩은 원본 프레임과 보조정보 프레임의 차를 채널 오류로 가정하고, 오류를 채널 복호기를 이용하여 수정한다. 이때, 오류가 수정되는데 필요한 패리티 비트의 양과 교차 확률을 정확히 알아야, 오류를 정확히 정정할 수 있다. 하지만, 기존 분산 비디오 코딩에서는 부호화에서 보조정보 프레임의 접근이 불가능하고, 복호기에서 원본 프레임의 접근이 불가능하다. 따라서, 정확한 교차 확률을 측정할 수 없고, 오류정정을 위해 필요한 패리티 비트의 양과 교차 확률의 측정이 정확히 이루어지지 않아 성능저하가 발생한다.

이러한 문제점을 해결하기 위하여 기존 연구에서는, 복호기에서 원본 영상을 예측하거나 부호기에서 보조정보를 예측하는 과정을 수행한다. 하지만, 기존 방법은 복호기에서 예측하기 때문에 부호기에서 예측된 정보를 이용할 수 없거나, 복잡도 증가량과 보조정보 프레임의 예측 정도는 상반 관계로 이루어져 있기 때문에, 복잡도에 대한 요구와 예측에 대한 요구를 모두 수용하지는 못한다. 하지만 제안하

는 방법은 부호기에서 예측한 정보를 이용할 수 있을 뿐만 아니라 상반 관계에 있는 두 가지 요구를 모두 최적으로 이룰 수 있다.

복호기에서 $t-1$ 키 프레임과 $t+1$ 키 프레임을 참조 영상으로 이용하여 움직임 벡터를 추정하였다. 따라서 수신된 움직임 벡터와 부호기에 존재하는 $t-1$ 키 프레임과 $t+1$ 키 프레임을 이용하여, 복호기에서 생성한 보조정보 프레임과 완벽하게 동일한 보조정보 프레임을 많은 복잡도 없이 복원할 수 있다.

3.3 DCT 변환

기존 분산 비디오 부호화 방법에서는 DCT 변환을 이용하여 에너지를 모으고, 사람의 눈의 특성을 고려하여 고주파 성분을 제거함으로써 비디오를 압축하였다. 기존 분산 비디오 부호화 방법의 DCT 변환은 대부분 4×4 블록 크기로 이루어졌다. 하지만, 제안하는 방법에서는 8×8 DCT 변환을 수행한다. DCT 변환은 블록의 크기가 커질수록 복잡도는 높아지나, 에너지의 압축 효율 역시 더 높아진다. 제안하는 방법은 블록마다 적응적으로 양자화를 수행하기 때문에, 복잡도와 효율성을 고려하여, 블록의 크기를 8×8 DCT 변환을 수행하여 효율성을 높였다.

3.4 블록 기반 양자화

기존 비디오 부호화 방법에서는 블록마다 양자화 정도를 변화시켜도, 주변 블록과의 양자화 정도가 유사해야만 급격한 화질 변화가 발생하지 않았다. 하지만, 분산 비디오 부호화 방법은 기존 부호화 방법과는 달리, 블록마다 필요한 비트의 양의 차이가 크다. 따라서 급격한 양자화 정도를 변환하는 것이, 급격하게 화질 변화가 일어나지 않게 하는 방법이다.

그림 3은 원본 프레임과 보조정보 프레임과의 차가 8이상일 경우, 픽셀 값을 하얀색으로, 그 외는 픽셀 값을 검은색으로 표시한 그림이다. 배경부분에는 거의 오류가 발생하지 않으나, 객체의 가장 자리는 오류가 밀집되어 발생하고 있다는 것을 알 수 있다. 픽셀 값의 변화가 거의 없는 배경이나 움직임이 똑같은 방향으로 이동 혹은 정지되는 객체의 영역은 보조정보 프레임에서 오류가 발생할 확률이 낮다. 하지만 $t-1$ 키 프레임에는 존재하지 않다가, $t+1$ 키 프레임에는 객체가 존재하거나, 존재하던 객체가 사라지거나 혹은 객체의 회전 등이 발생하여, $t-1$ 키 프레임과 $t+1$ 키 프레임에서 서로 대응되는 부분이 없을 영역 (hole) 에서는 보조정보 프레임에 오류가

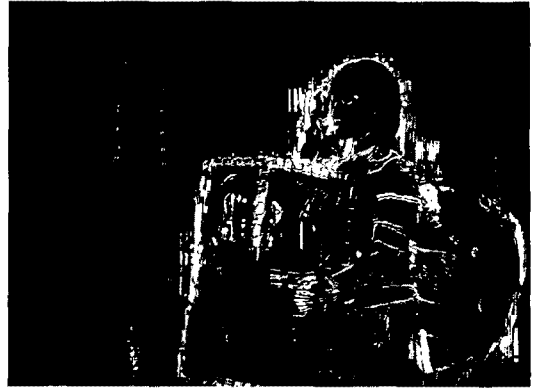


그림 3. 원본 영상과 보조정보 영상과의 차
Fig. 3. The difference between original frame and side information frame

발생할 확률이 높다. 또한, 이렇게 대응되지 않는 부분은 밀집되어 있기 때문에 오류 역시 밀집된다.

표 1은 두 프레임 사이에서 대응되는 부분이 없는 영역 (hole) 이 전체 프레임에 끼치는 영향을 측정한 표이다. 표에서 (A) 부분은 대응되는 블록이 없어도 다양한 방법을 이용하여 보간한 후, 원본 프레임과 PSNR을 측정한 것이고, (B) 부분은 서로 대응되는 블록이 존재할 경우에만 보간하고, 이렇게 보간된 부분에서만 원본 프레임과 PSNR을 측정한 것이다. 대응되는 블록이 없는 경우, 0.73dB 영향을 끼치는 것을 확인 할 수 있었다.

제안하는 방법은 수신된 움직임 벡터를 바탕으로, 블록마다 적응적으로 양자화를 수행한다. 부호기에서는 복호기에서의 보조정보 프레임과 동일한 보조정보 프레임과 원본 프레임을 이용하여, 오류의 양, 교차 확률 및 오류가 발생한 위치를 정확히 측정한다. 측정된 결과를 이용하여 블록기반 양자화를 수행한다.

그림 4는 제안하는 양자화 방법의 흐름도를 나타낸다. 제안하는 방법은 우선 오류의 양 ($Count_{coeff,block}$), 오류의 위치 (M_{error}) 를 측정하여, 한 번에 채널 부호화를 수행할 범위 ($Range_{Div}$) 를 측정하고, 각각

표 1. 원본 영상과 보조정보의 PSNR
Table 1. PSNR of original frame and side information

영상	Hole된 영역을 포함 (A)	Hole된 영역을 제외 (B)	A-B
Exit	37.23	37.47	0.24
Akko	32.63	33.85	1.22
평균			0.73

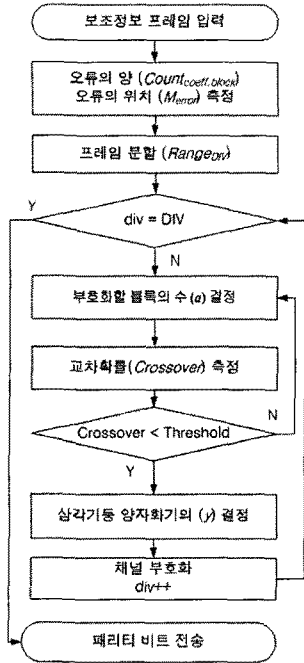


그림 4. 제안하는 양자화 방법의 흐름도
Fig. 4. Flowchart of the proposed quantization method

의 범위 안에서 교차 확률 (Crossover) 을 측정한다. 측정된 교차 확률에 따라 부호화할 블록의 수 (a) 를 결정하고, 그 수에 따라 양자화 너비 (y) 를 결정한다.

우선, 보조정보 프레임의 오류의 양 (Count_{coeff.block}), 교차확률과 위치를 측정하고, 이를 이용하여 각각의 블록들을 부호화를 수행할지, 생략할지를 결정한다. 다음으로 부호화를 수행하도록 결정된 블록을 어느 정도로 양자화를 수행할지 결정하게 된다. 다음의 식은 오류가 발생한 블록의 위치(M_{error})를 측정하는 식이다.

$$M_{error}[i] = \begin{cases} 1, & \sum_{j=0}^J \sum_{k=0}^K |ORI_{i,j,k} - SI_{i,j,k}| > 0 \quad i < I \quad (1) \\ 0, & otherwise \end{cases}$$

ORI는 원본 프레임이고, SI는 보조정보 프레임을 뜻한다. i는 블록의 인덱스이며 I는 프레임 내의 블록의 개수를 의미한다. j는 블록 내에서의 계수의 인덱스이며, J는 블록 내의 부호화를 수행할 계수의 개수를 의미한다. k는 비트플레인의 인덱스이며, k가 0일 때, 최상위 비트를 의미하고, k의 값이 증가할수록 하위 비트를 의미한다. K는 계수 내의 비트층의 부호화를 수행할 개수를 의미한다. K와 J는

삼각기둥의 높이 (x) 와 너비 (y) 의 초기값에 해당한다.

다음의 식을 이용하여, 제안한 방법은 보조정보 프레임 내의 오류의 양 (Count_{coeff.block}) 을 측정한다.

$$Count_{coeff.block}[I, J, K] = \sum_{i=0}^I \sum_{j=0}^J \sum_{k=0}^K (M_{error}[i] \times |ORI_{i,j,k} - SI_{i,j,k}|) \quad (2)$$

LDPCA 코드 길이보다 원본 영상이 크기 때문에 한 번에 프레임 전체를 모두 채널 부호화를 수행하는 것이 아니라, 나눠서 수행을 한다. 한 프레임 당 분할되는 횟수를 DIV라고 했을 때, 제안하는 방법은 오류의 양을 균등하게 분포하게 하기 위하여, 아래의 식으로 한 번에 부호화 될 오류의 양을 결정하고, 이를 이용하여 프레임을 분할한다. 이때, DIV는 한 번에 부호화되는 범위 안에, 너무 많은 오류가 존재하지 않도록 해야 하기 때문에, 오류의 양이 많을수록 DIV는 큰 값을 갖게 된다.

$$Range_{DIV}[div] = \sum_{div=0}^{DIV} \sum_{i=0}^I i$$

$$if (div \times NB = Count_{coeff.block}[i, J, K]) \quad (3)$$

$$NB = \frac{Count_{coeff.block}[I, J, K]}{DIV}$$

div는 프레임의 분할에 관한 인덱스이며, length_{LDPCA}는 LDPCA 코드 길이를 의미한다.

다음으로, 초기값으로 정한 부호화할 블록의 수 (a) 를 이용하여 분할된 프레임에서 각각의 교차 확률 (Crossover) 을 아래와 같이 측정한다. 또한, 교차 확률이 충분히 작지 않으면 부호화할 블록의 수 (a) 값을 수정하여 반복해서 구하게 된다.

$$Crossover[div] = \sum_{div=0}^{DIV} \frac{Count_{coeff.block}[Range_{DIV}[div], J, K]}{\alpha} \times 100 \quad (4)$$

분할된 프레임 내에서 교차확률에 맞도록 채널 부호화하기 위하여, 오류가 없는 블록을 부호화할지를 결정하여 수행한다. 측정된 교차 확률이 충분히 낮도록 부호화한다. 제안한 방법은 블록의 오류의 유무를 표시한 M_{error}를 이용하여, 오류가 없는 블록으로 결정된 블록은 부호화를 적응적으로 수행한다.

즉, 오류가 있는 블록에 오류가 밀집되어 있다면, 교차확률을 낮추기 위하여 부호화를 수행하고, 오류가 있는 블록만 부호화해도 교차확률이 충분히 낮을 경우, 오류가 없는 블록은 부호화를 수행하지 않는다. 이는, 오류가 많은 블록에 많은 비트를 할당하며, LDPCA의 성능 향상을 위해 교차확률에 맞게 비트를 할당함으로써, 비트를 효율적으로 사용할 수 있도록 하였다.

제안한 방법은 부호기에서 복호기의 채널 복호화 능력을 알고 있다는 가정 하에, 부호기에서 오류가 없는 비트와 오류가 있는 비트의 비율을 복호화가 가능한 만큼의 교차 확률에 맞추어 전송하게 된다. 따라서 영상에 따라 전송해야 하는 비트의 양은 증가할 수 있지만, 복호화가 실패할 확률을 최대한 낮추어, 복호기의 안전성을 높이고, 비트를 효과적으로 사용할 수 있다.

기존의 분산 비디오 부호화 방법에서 사용되는 양자화 매트릭스는 그림 5와 같이, 4×4 DCT 변화에 맞추어 설계되었으며, 이 양자화 매트릭스는 사면체의 형태를 띠고 있다. 이를 이용하여, 사람의 눈이 잘 인식하지 못하는 고주파 성분을 제거함으로써 비디오를 압축한다. 기존의 분산 비디오 부호화방법은 각 DCT 계수 별로, 각 계수의 비트 층 별로 부호화를 수행하도록 설계되었다. 하지만, 이렇게 양자화를 수행하여 비트 층 별로 부호화를 수행할 경우, 비트 층이 높은 최상위 비트 쪽은 교차 확률이 낮아 복호화가 성공할 수 있지만, 최하위 쪽으로 갈수록 오류가 많이 발생하여 교차 확률이 높아져서 복호화를 실패할 확률이 높게 된다.

제안하는 방법은 그림 6처럼 이등변삼각기둥 모양의 양자화를 블록별로 수행한다. 오류가 적은 최상위 비트 쪽의 비트들과 오류가 많은 최하위 쪽의 비트들을 이용하여 교차 확률을 낮추도록 하였다. 그림 6의 이등변 삼각형에서 길이가 같은 두 변 중 하나를 양자화 너비 (y) 라고 하며, 삼각기둥의 크기를 양자화 높이 (x) 라고 한다.

7	6	5	4
6	5	4	3
5	4	3	2
4	3	2	0

그림 5. 양자화 매트릭스 (M^6)
Fig. 5 Quantization Matrix (M^6)

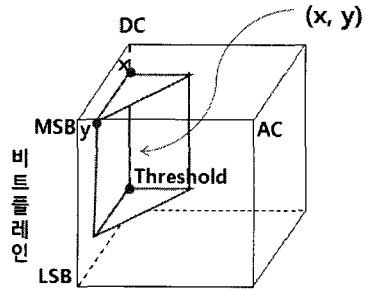


그림 6. 이등변삼각기둥 모양의 양자화
Fig. 6. Quantization of the isosceles triangular prism

영상의 크기가 LDPCA 코드의 길이보다 크기 때문에, 나눠서 전송하게 된다. 부호기에서 채널 부호화를 수행할 때, 제안하는 방법은 채널 부호화의 입력마다 그림 6의 양자화 너비 (y) 를 결정하여 양자화를 수행하고, 복호기에 전송하여 복호화 할 수 있도록 한다. 양자화 높이 (x) 는 고정으로 수행하여, 양자화 너비 (y) 를 측정하기 위한 수식은 다음과 같다. 블록마다 하나의 삼각기둥이 생성되며, 이 삼각기둥 안에 포함된 비트들의 양은 다음과 같다.

$$\frac{y^2}{2} \times x \tag{5}$$

하나의 삼각기둥에 포함된 비트들의 양이 블록의 개수만큼 증가되기 때문에, 하나의 LDPCA 코드로 부호화 되는 블록의 개수를 α 라고 했을 때 아래와 같은 식이 만족하게 된다.

$$length_{LDPCA} = \frac{y^2}{2} \times x \times \alpha \tag{6}$$

양자화 너비 (y) 를 구하는 것이 목적이기 때문에, 양자화 너비 (y) 에 관한 식으로 변형시키면 아래와 같다.

$$y = \sqrt{\frac{2length_{LDPCA}}{\alpha \times x}} \tag{7}$$

IV. 실험 결과

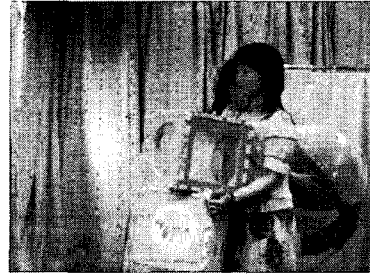
제안한 방법의 성능을 평가하기 위해 기존 방법과 비트의 사용율과 PSNR을 측정하였다. 실험은 640×480 크기의 VGA급 영상, “Exit”와 “Akko”를 이용하였으며, 각각 3 개씩의 QP 값을 이용하였다.

또한, 보조정보 프레임 생성하는 방법은 참조 논문 [6] 방법을 이용하였다. 기존 방법은 그림 5의 양자화기를 이용하여 고정 양자화를 이용하였다. LDPCA는 66단계로 나뉘어 복호기로 전송되도록 설계되어 있다. 하지만 66단계를 다 전송하는 것은 압축이 전혀 되지 않기 때문에, 압축율을 위하여 45 단계까지만 전송을 허용하여 그 이하에서만 비트를 전송할 수 있도록 설계하였다. 공정한 비교를 위하여 제안하는 방법은 목표 비트량 이하에서만 비트를 전송할 수 있도록 설계하였다. 제안하는 방법에서 고정적으로 사용한 양자화 높이 (x) 는 7로, 7개의 비트 층을 사용했다.

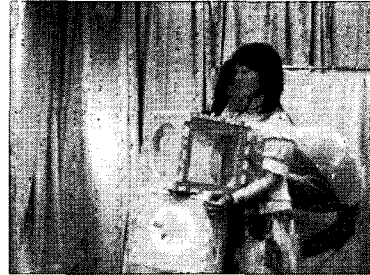
제안하는 방법에서는 부호화한 방법을 복호기에 지시하는 비트가 필요하다. 블록이 부호화 되었는지 혹은 생략되었는지를 가르쳐주는 비트를 블록마다 1개의 비트씩 할당하였다. 또한, 제안하는 방법은 6336 코드 길이의 LDPCA를 사용하였고, 양자화 너비 (y) 를 가르쳐주기 위한 비트를 LDPCA 코드 길이 6336 마다 양자화 정도를 알려주도록 하였다. 피드백 채널에 전송되는 사용된 참조영상과 움직임 벡터는 엔트로피 코딩 방법을 이용하여 압축하여 전송했다.

그림 7은 복호기의 보조 정보 영상, 움직임 벡터를 바탕으로 예측한 보조정보와 선형 보간법으로 예측한 보조정보 영상으로 두 가지의 방법으로 예측한 보조정보 영상이다. 선형보간법을 이용하여 예측된 보조정보 영상은, 복호기의 보조정보 영상과 비교하여 픽셀값의 변화가 없는 배경 부분을 제외한 나머지 영역에서 많은 오류가 있는 것을 확인할 수 있다. 하지만 움직임 벡터를 이용하여 예측된 보조정보 프레임은 움직임이 있는 부분에서도 복호기의 보조정보와 일치 하는 것을 확인할 수 있었다.

움직임 벡터를 바탕으로 복원한 보조정보 영상과 선형 보간법을 이용한 보조정보 영상의 성능을 측정하였다. 측정된 결과는 표 2와 같다. 선형 보간을 이용하여 예측된 보조정보 영상의 평균 PSNR은 18.31dB이며, 제안한 방법으로 보조정보를 예측된 영상의 평균 PSNR은 무한대인 것을 확인할 수 있다. 부호기에서 예측된 보조정보를 바탕으로 교차확률 및 필요한 패리티 비트의 양을 예측하게 되는데, 부호기에서 예측한 보조정보와 복호기에서 생성한 보조정보 프레임의 유사도가 18.31dB 정도 밖에 안되기 때문에, 예측한 교차확률 및 필요한 패리티 비트의 양이 실제 교차확률과 필요한 패리티 비트와 상당히 차이가 날 수 있다. 하지만 제안하는 방법은



(a)



(b)



(c)

그림 7. (a) 복호기에서 생성한 보조정보 프레임 (b) 부호기에서 움직임 벡터를 이용하여 복원한 보조정보 프레임 (c) 부호기에서 선형보간을 이용하여 복원한 보조정보 프레임
Fig. 7. (a) Side information interpolated in decoder side (b) Reconstructed side information by transmitted motion vector in encoder side (c) Reconstructed side information by transmitted motion vector in encoder side

예측한 교차확률 및 필요한 패리티 비트의 양이 실제 교차확률과 필요한 패리티 비트와 같아, 부호화의 효율 및 비트 사용의 효율성을 높일 수 있다.

표 3은 기존 방법과 제안한 방법의 비트율과 PSNR을 나타낸 것이다. 제안하는 방법의 비트율이 기존 비트율에 비해 66%로 감소했으며 PSNR은 차이가 거의 없는 것을 확인할 수 있었다. 복호기에서 생성한 보조정보 프레임에는 오류가 밀집되어 있지만 부호기는 밀집된 정도를 예측하지 못하기 때문에, 비트 층별로 전송한다. 교차확률이 어느 정도 유사할 경우는, 필요한 비트량만 수신되면 복호화가 잘 이루어지나, 교차확률이 다를 경우, 실제 필요한 패리티 비트의 양보다 훨씬 더 많은 비트를 요구하

표 2. 선형보간법에 의한 보조정보 프레임 예측 성능
Table 2. The performance of estimated side information by linear interpolation

시퀀스	선형보간을 이용하여 예측된 보조정보 프레임의 PSNR	움직임 벡터를 이용하여 예측된 보조정보 프레임의 PSNR
Exit	18.71	∞
Akko	19.56	∞
평균	18.31	∞

표 3. 기존 방법 및 제안한 방법의 비트율 및 PSNR
Table 3. Bit rate and PSNR of conventional method and the proposed method

영상		기존 방법		제안한 방법	
		Rate	PSNR	Rate	PSNR
Exit	26	651700	35.50	224313	35.70
	29	651700	35.12	228078	35.33
	31	595840	35.09	249988	35.20
Akko	24	707560	32.99	211259	33.30
	29	726180	31.57	225584	31.93
	36	744800	30.59	231652	31.02

게 되고, 이렇게 많은 패리티 비트를 전송받아도 복호화를 실패할 확률이 높아지기 때문에 성능 저하가 보조정보의 성능에 따라 크게 변화한다. 또한, 복호기에서 복호화 성능 이상으로 오류가 있을 경우, 교차된 비트가 아닌 비트도 잘못 고쳐지는 경우가 많기 때문에 화질저하도 발생한다. 하지만, 제안하는 방법은 교차확률을 정확히 부호기에서 측정 가능하므로, 복호기가 복호화 할 수 있는 정도로 교차 확률을 맞추어 전송한다. 따라서 의미가 없이 낭비되는 비트량을 줄일 수 있고 잘못 복호화 하는 경우가 현저히 감소하게 되며, 화질이 향상된다.

V. 결 론

본 논문에서는 수신된 움직임 벡터에 따른 적응적 블록 양자화를 이용한 분산 비디오 부호화 방법을 제안한다. 제안한 알고리즘을 이용하여 부호화를 수행한 결과, 의미 없는 비트의 낭비 없이 비트를 효율적으로 사용했다. 뿐만 아니라 비트량도 현저히 감소된 것을 확인할 수 있었으며, 복호기에서 더 많은 패리티 비트를 요구하기 위한 지연을 감소시킬

수 있었다. 하지만, 움직임 벡터에 대한 비트와, 부호화 방법을 복호기에 알려주기 위한 비트가 주기적으로 들어간다는 단점이 있으며, 부호기에서는 복호기에서 정보를 전송해 주기 전에는 부호화를 수행할 수 없다는 문제점을 가지고 있다. 추후 분산 비디오 부호화 방법에 관련하여, 제안한 방법에서 움직임 벡터를 전송하기 위해 필요한 비트를 감소시킬 수 있는 방법과 부호화 방법을 복호기에 알려주기 위한 비트를 효율적으로 사용할 수 있는 방법, 그리고 피드백 채널을 이용하지 않고 분산 비디오 부호화를 수행하는 방법들을 연구할 예정이다.

참 고 문 헌

- [1] 심혁재, 전병우, “분산 비디오 압축 기술,” 전자공학회지, 제 36권 4호, 91-105쪽, 2009년 4월
- [2] 박시내, 심동규, 전병우, “조명보상 기반 분산 다시점 비디오 코딩,” 전자공학회논문지, 제 45권, SP편, 제 6호, 17-26쪽, 2008년 11월
- [3] W.J. Chien, L.J. Karam, and G.P. Abovslleman, “Rate-distortion based selective decoding for pixel-domain distributed video coding,” Conf. on ICIP 2008, pp.1132-1135, Oct. 2008
- [4] J.L. Martinez, C.Holder, G.E. Fernandez, H. Kalva, and F. Quiles, “DVC using a half-feedback based approach,” Conf. on Multimedia and Expo, pp.1125-1128, April 2008
- [5] Q. Linbo, H. Xiaohai, L. Rui, and D. Xiewei, “Application of punctured turbo codes in distributed video coding,” Conf. on ICIG 2007, pp.241-245, Aug. 2007
- [6] C. Brites, and F. Pereira, “Encoder rate control for transform domain wyner-ziv video coding,” Conf. on ICIP 2007, Vol.2, pp.5-8, Oct. 2007
- [7] K.Y. Min, S.N. Park, and D.G. Sim, “Side information generation using adaptive search range for distributed video coding,” Conf. on RacRim 2009, pp.854-857, Aug. 2009
- [8] D. Varodayan, A. Aaron, and B. Girod, “Rate-adaptive codes for distributed source coding,” EURASIP Signal Processing Journal, Special Section on Distributed Source Coding, Vol.86, No.11, pp.3123-3130, Nov. 2006

민 경 연 (Kyung-Yeon Min)

준회원



2009년 2월 광운대학교 컴퓨터
공학과 학사
2009년 3월~현재 광운대학교
컴퓨터공학과 석사과정
<관심분야> 분산코딩, 다시점
비디오

심 동 규 (Dong-Gyu Sim)

종신회원



1999년 2월 서강대학교 전자공
학과 공학 박사
2000년 (주) 현대 전자
2002년 (주) 바로 비전
2005년 3월~현재 광운대학교
컴퓨터공학과 부교수
<관심분야> 영상신호처리, 영
상압축, 컴퓨터비전

박 시 내 (Sea-Nae Park)

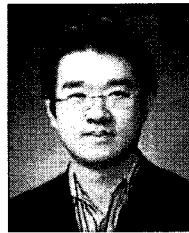
정회원



2004년 2월 광운대학교 컴퓨터
공학과 학사
2006년 2월 광운대학교 컴퓨터
공학과 석사
2008년 3월~현재 광운대학교
컴퓨터공학과 박사과정
<관심분야> 분산코딩, 다시점
비디오

김 상 호 (Jung-Hak Nam)

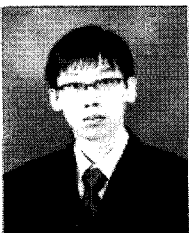
종신회원



2004년 2월 서울대학교 전기
컴퓨터공학부 공학 박사
2004년 삼성전자
2006년 미국 남가주대학교, 통
신과학연구소, 박사후 연구원
2007년 9월~현재 성균관대학
교 정보통신공학부 조교수
<관심분야> 통신분야

남 정 학 (Jung-Hak Nam)

정회원



2006년 2월 광운대학교 컴퓨터
공학과 학사
2008년 2월 광운대학교 컴퓨
터공학과 석사
2009년 9월~현재 광운대학교
컴퓨터공학과 박사과정
<관심분야> 영상신호처리, 영
상압축