

다양한 부가정보 생성 기법을 이용한 분산 동영상 부호화 시스템의 성능 비교

*문학수 **이창우

가톨릭대학교 정보통신전자공학과

*haksoo84@catholic.ac.kr **changwoo@catholic.ac.kr

Performance Comparison of Distributed Video Coding System Using Various Side Information Generation Methods

*Hak-Soo Moon **Chang-Woo Lee

Dept. of Information, Communications and Electronics Eng., The Catholic University of Korea

요약

분산 동영상 부호화 시스템에서는 복호기에서 움직임 보상 보간 기법을 이용하여 부가정보를 생성한다. 생성된 부가정보의 정확성이 Wyner-Ziv 프레임의 복원하기 위한 parity 정보량에 큰 영향을 미치기 때문에 부가정보의 정확한 생성이 분산 동영상 부호화 시스템의 성능에 큰 영향을 미친다. 따라서 본 논문에서는 다양한 부가정보 생성 기법의 성능을 분석하고 효율적인 부가정보 생성 기법을 제안하였다. 또한 생성된 부가정보가 분산 동영상 부호화 시스템의 성능에 미치는 영향을 비교분석 하였다.

1. 서론

기존의 동영상 부호화 시스템, 특히 H.264 부호화 표준은 우수한 압축 효율을 보인다. 하지만 움직임 추정으로 인하여 부호화의 복잡도가 매우 크다[1]. 부호화의 복잡도를 감소시키기 위해서 분산 동영상 부호화(distributed video coding : DVC) 시스템에서는 부호기에서 움직임 추정을 하지 않고 프레임간의 상관성을 복호기에서 이용한다[2]. 특히 복호기에서 Wyner-Ziv 프레임의 복원하기 위해 부가정보(side information)를 생성하게 되는데 이때 부가정보와 Wyner-Ziv 프레임간의 유사도에 따라서 분산 동영상 부호화 시스템의 성능이 크게 좌우된다. 최근에 효율적으로 부가정보를 생성하는 기법이 활발히 연구되고 있는데 순방향 움직임 추정 및 양방향 예측을 이용하여 움직임 벡터를 추정하고 가중 벡터 중간 값 필터 등을 이용하여 움직임 벡터를 보정하는 방법이 많이 사용된다[4-9].

본 논문에서는 다양한 부가정보 생성 기법을 이용하여 생성한 부가정보가 분산 동영상 부호화 시스템의 성능에 미치는 영향을 비교 분석하기 위해 다양한 방법을 통해 생성된 부가정보와 원 영상간의 PSNR결과를 분석하고 분산 동영상 부호화 시스템에 적용시켰을 때의 성능에 대한 실험을 하였다. 2장에서는 분산 동영상 부호화 시스템에 대하여 설명한다. 3장에서는 부가정보 생성 방법을 위한 다양한 기법들을 설명하고 4장에서는 다양한 방법을 통해 생성된 부가정보와 원 영상간의 PSNR 결과와 분산 동영상 부호화 시스템에 적용시켰을 때 성능의 모의실험결과를 제시하고 5장에서는 결론을 맺는다.

2. 분산 동영상 부호화 시스템의 개요

분산 동영상 부호화 시스템은 그림 1에 도시한 것과 같이 크게 key 프레임을 부호화하는 부분과 Wyner-Ziv 프레임을 부호화 하는 부분으로 나눌 수 있는데 key 프레임은 H.264와 같은 기존 부호화 시스템의 intraframe 부호화 기법으로 부호화된다[3]. 또한 Wyner-Ziv 프레임은 부호기에서 4x4 블록 크기로 나누어 DCT 영역으로 변환하고 양자화를 하게 된다. 양자화를 거친 계수들은 각 계수별로 bitplane을 만들게 되는데 이때 양자화 테이블에 따라서 bitplane 수가 결정되며 이렇게 결정된 각 bitplane을 터보부호기의 입력으로 하여 오류 정정 부호 위한 parity 비트를 생성한다. 이 parity 비트를 Wyner-Ziv 비트라 하고 이 Wyner-Ziv 비트들을 복호기로 전송 한다. 복호기에서는 key 프레임과 이전 복호화 된 Wyner-Ziv 영상 등을 이용하여 부가정보를 생성하게 되는데 이렇게 생성된 부가정보와 부호기에서 전송한 Wyner-Ziv 비트들을 이용하여 Wyner-Ziv 프레임을 복원한다.

3. 부가정보 생성 기법

분산 동영상 부호화 시스템에서 부가정보는 key 프레임들을 이용하여 생성한다. 일반적으로 움직임 보상 보간 방식을 이용하여 부가정보를 생성하는데 그 중에서 대표적인 방법을 소개한다. 먼저 순방향 움직임 추정 방법에서는 이후 key 프레임 블록에서 이전 key 프레임으로 움직임을 추정을 하면 현재 삽입 프레임에 해당하는 블록은 움직임 추정으로 생성된 벡터의 중간에 위치하게 된다고 가정한다[2].

※이 논문은 2009년도 정부의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No.2009-0088064)

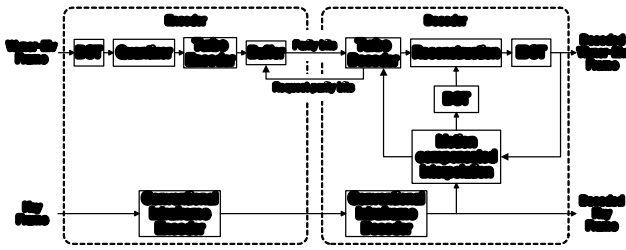


그림 1. 분산 동영상 부호화 시스템의 구조

그림 2는 순방향 움직임 추정방법을 도시한 것이다. 이때 현재 삽입 프레임에서는 블록의 가운데를 지나가지 않을 경우에 블록 간 겹치거나(overlapped) 빈 공간(hole)이 생길 수 있다. 이런 문제를 해결하기 위해 순방향 움직임 추정을 통해 생성된 벡터 중 삽입 프레임의 블록과 가장 가까이 지나가는 벡터를 블록의 가운데로 그대로 가져와 현재 삽입 프레임의 블록을 채우게 된다.

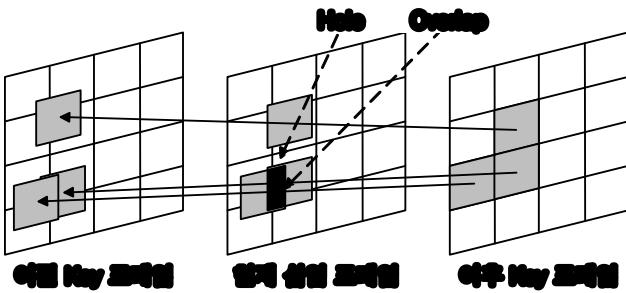


그림 2. 순방향 움직임 추정 방법

양방향 움직임 추정 방법은 중간 삽입 프레임의 블록을 가운데에 두고 이전 key 프레임과 이후 key 프레임에서 대칭적으로 움직임 추정 방법을 사용한다[8]. 움직임이 큰 영상에서는 순방향 움직임 추정 방식이 좀 더 좋은 결과를 보여주지만 움직임이 작은 영상에서는 양방향 움직임 추정 방법이 더 좋은 결과를 보인다.

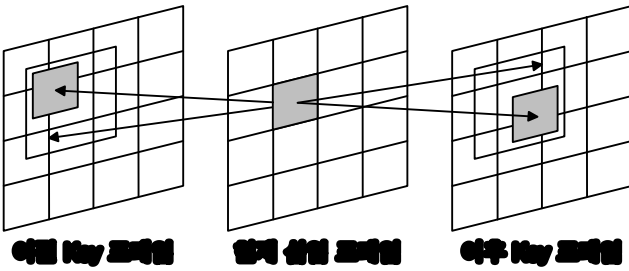


그림 3. 양방향 움직임 추정 방법

순방향 움직임 추정 방법으로 벡터를 생성하고 양방향 움직임 추정 방법을 통하여 좀 더 세밀하게 움직임 추정을 하게 되면 전체 성능을 높일 수 있게 된다[4-6]. 이렇게 찾은 벡터를 가중치 벡터 중간 값 필터를 이용하여 보정하게 되는데 가중치 벡터 중간 값 필터는 현재 블록과 주위 블록의 벡터들을 사용하여 현재의 블록의 벡터를 보정하는 기법인데 벡터간의 거리와 가중치를 이용하여 가중치 벡터 중간 값으로 현재 블록을 보정한다. 가중치 벡터 중간 값 필터는 식 (1)과 같다 [7].

여기서 $\|v_j - v_i\|$ 는 v_j 와 v_i 간의 거리를 나타낸다. 또한 n 는 가중치인데 이때 가중치는 식 (2)과 같이 계산된다.

$$\sum_{i=1}^N w_i \|v_m - v_i\| \leq \sum_{i=1}^N w_i \|v_j - v_i\| \quad j = 1, 2, \dots, N \quad (1)$$

$$w_i = \frac{MSE(v_i)}{MSE(v_j)} \quad (2)$$

다른 대표적인 부가정보 생성 방법 중 Peixoto등이 제안한 방법은 순방향 움직임 추정방법과 역방향 움직임 추정 방법을 사용하여 중간 삽입 프레임을 각각 생성하여 그 두 개의 중간 삽입 프레임을 합치는 방법이다. 그림 4에 Peixoto등이 제안한 중간 삽입 프레임을 생성하는 방법을 도시하였다[9].

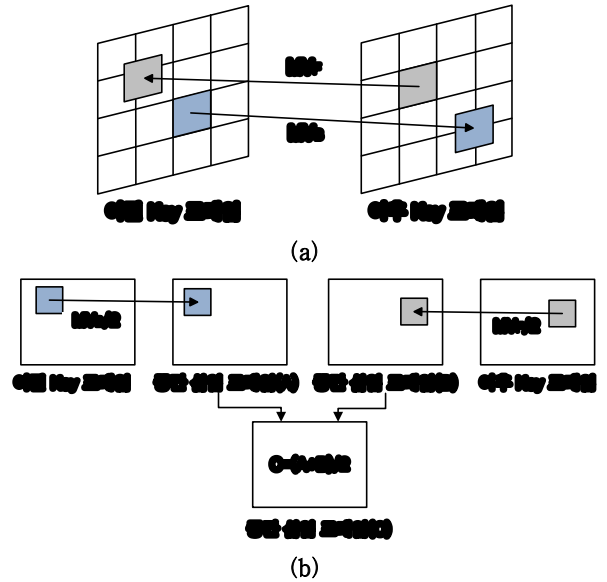


그림 4. Peixoto등이 제안한 중간 삽입 프레임을 생성하는 방법 (a) 순방향 및 역방향 움직임 추정 (b) 중간 삽입 프레임 생성

이때 역시 빈 공간이 생길 수 있는데 중간 삽입 프레임을 합치므로 많은 부분의 빈 공간을 채울 수 있다. 아직 남은 빈 공간은 일정 크기의 마스크를 씌워서 쌍방향 움직임 추정 방법(bidirectional)을 사용하여 채우게 된다.

부가정보를 생성하기 위해 이전 key 프레임과 이후 key 프레임을 사용하는데 이때 잡음을 제거하기 위해 지역 통과 필터(LPF)를 통과 시킨 key 프레임을 사용할 수 있다. 또한 거리에 대한 페널티 방법을 사용할 수 있는데 이 방법은 기준 벡터와 움직임 추정된 벡터 사이에 거리에 대한 페널티를 주는 방법이다. 이렇게 지역통과 필터를 통과 시킨 key 프레임을 사용하거나 거리에 대한 페널티를 하면 좀 더 좋은 결과를 얻을 수 있다[10].

4. 모의실험

부가정보를 생성하는 방법이 분산 동영상 부호화의 성능에 미치는 영향을 확인하기 위해 다음과 같은 분산 동영상 부호화 시스템을 사용하였다. 먼저 Wyner-Ziv 프레임은 부호기에서 4x4 블록 크기로 DCT 영역으로 변환된다. 그 다음에 그림 5에 제시한 8 개의 양자화 테이블 중 하나를 사용하여 양자화를 하게 된다[1,6]. 이를 통해 bitplane를 생성하게 되고 터보 부호기를 이용하여 Wyner-Ziv 비트를 생성하게 된다.

표 1. 다양한 부가정보 생성방법에 따라 생성된 부가정보와 원 영상간의 PSNR 결과

생성 방법 영상(QCIF)	Forward	Bilateral	LPF+ Forward+ Bilateral+ WVMF	Forward+ Bilateral+ WVMF	Forward+ Backward+ Bidirectional 1	Forward+ Backward+ Bidirectional 2
Coastguard	37.04338	35.81805	38.62316	38.61581	37.0217	38.45124
Foreman	34.93393	29.62104	36.57355	36.5593	35.67255	36.50068
Hall monitor	40.20148	39.20212	40.58557	40.5806	40.30746	40.66266
Stefan	28.93317	24.28842	30.32004	30.29803	29.57104	30.45811
프레임당 평균	36.40854	33.44821	37.66652	37.65599	36.75799	37.62647

16	8	0	0	32	8	0	0	32	8	4	0	32	16	8	4
8	0	0	0	8	0	0	0	8	4	0	0	16	8	4	0
0	0	0	0	0	0	0	0	4	0	0	0	8	4	0	0
0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0
32	16	8	4	64	16	8	8	64	32	16	8	128	64	32	16
16	8	4	4	16	8	8	4	32	16	8	4	64	32	16	8
8	4	4	0	8	8	4	4	16	8	4	4	32	16	8	4
4	4	0	0	8	4	4	0	8	4	4	0	16	8	4	0

그림 5. 양자화를 위한 8가지 테이블

복호기에서 생성된 부가정보 역시 4×4 블록크기의 DCT 영역으로 변환되고 이렇게 변환된 부가정보와 부호기에서 전송한 Wyner-Ziv 비트를 이용해 Wyner-Ziv 프레임의 복원하게 된다. 모의실험을 위한 부가정보 생성 방법으로는 순방향 움직임 예측 방법, 양방향 움직임 예측 방법, DISCOVER 시스템에서 사용한 순방향 움직임 예측 방법으로 백터를 생성한 다음 양방향 움직임 예측방법을 통해 세밀하게 찾고 가중치 벡터 중간 값 필터를 사용한 방법과 Peixoto 등이 제안한 순방향과 역방향 움직임 예측을 통해 각각 만들어진 중간 프레임을 합친 다음 빈 공간을 마스크를 이용해 채우는 방법 등을 사용하였다[5]. 순방향과 역방향 움직임 예측을 위한 블록 크기는 16×16으로 하였으며 움직임 탐색 영역은 48×48로 하였다. 양방향 및 가중치 벡터 중간 값 필터를 위한 블록 크기는 8×8로 하였다. 또한 선택적으로 저역 통과 필터를 사용하였다. 실험 결과의 신뢰성을 높이기 위해서 QCIF 규격 Coastguard 300 프레임, Foreman 400 프레임, Hall monitor 300 프레임, Stefan 100 프레임을 각각 사용하였으며 Coastguard 영상과 Stefan 영상은 25Hz 이고 Foreman 영상과 Hall monitor 영상은 30Hz 이다. 또한 각 부가정보 생성 방법들로 만들어진 부가정보와 원 영상간의 PSNR은 식 (3)을 이용하여 구할 수 있다. 이 때 MSE(mean square error)는 식 (4)와 같이 계산되는데 $u(m,n)$ 는 원 영상의 화소 값, $u'(m,n)$ 는 부가정보의 화소 값을 나타낸다.

$$PSNR = 10 * \log_{10} (255^2 / MSE) \text{ dB} \quad (3)$$

$$MSE = \frac{1}{MN} \sum_{m=0}^M \sum_{n=0}^N |u(m,n) - u'(m,n)|^2 \quad (4)$$

표 1은 각 영상에 대해서 여러 방법으로 생성된 부가정보와 원 영상간의 PSNR 결과이다. 표 1에서 순방향, 역방향, 쌍방향 움직임 예측 방법 2는 기존에 소개한 방법 1에서 블록을 나누어 좀 더 세밀하게 찾은 방법이다. 이렇게 생성된 부가정보가 분산 동영상 부호화 시스템의

성능에 미치는 영향을 분석하기 위해서 각 영상에 대해서 다양한 방법으로 만들어진 부가정보와 부호기에서 전송된 Wyner-Ziv 비트들을 가지고 복원된 Wyner-Ziv 프레임과 원 영상간의 PSNR을 비교하였다.

그림 6은 Coastguard 영상을 분산 동영상 부호화 시스템에 적용시켰을 때의 전송된 Wyner-Ziv 비트율에 따른 Wyner-Ziv 영상과 원 영상간의 PSNR 결과이다. 부가정보와 원 영상간의 PSNR이 높으면 만들어진 Wyner-Ziv 영상을 복원 할 때 적은 Wyner-Ziv 비트로도 높은 PSNR 결과를 보이는 것을 알 수 있다. 그림 7은 Foreman 영상을 분산 동영상 부호화 시스템에 적용시켰을 때의 전송된 Wyner-Ziv 비트율에 따른 Wyner-Ziv 영상과 원 영상간의 PSNR 결과이다.

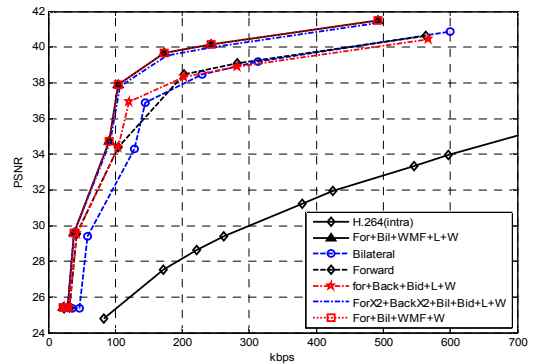


그림 6. 다양한 방법으로 생성된 부가정보를 이용해 만들어진 Wyner-Ziv 영상과 원 영상간의 PSNR 결과 (Coastguard)

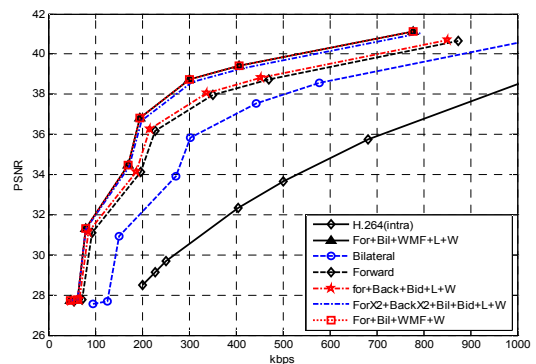


그림 7. 다양한 방법으로 생성된 부가정보를 이용해 만들어진 Wyner-Ziv 영상과 원 영상간의 PSNR 결과 (Foreman)

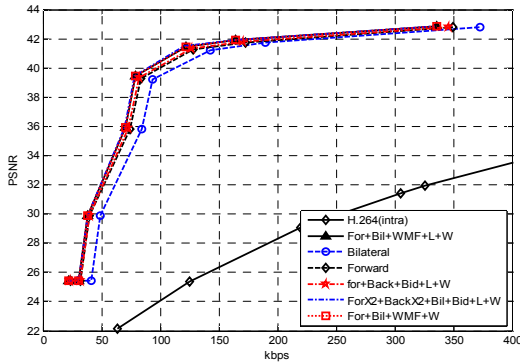


그림 8. 다양한 방법으로 생성된 부가정보를 이용해 만들어진 Wyner-Ziv 영상과 원 영상간의 PSNR 결과 (Hall monitor)

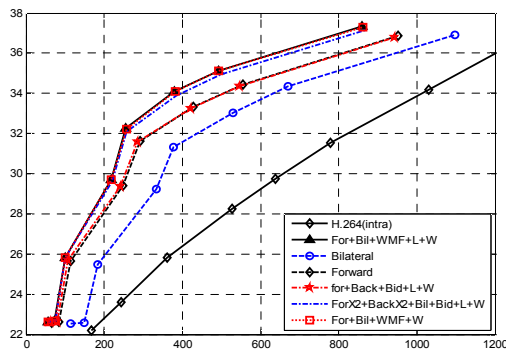


그림 9. 다양한 방법으로 생성된 부가정보를 이용해 만들어진 Wyner-Ziv 영상과 원 영상간의 PSNR 결과 (Stefan)

역시 생성된 부가정보와 원 영상간의 PSNR이 높으면 적은 Wyner-Ziv 비트만으로도 좋은 Wyner-Ziv 영상을 복원 할 수 있다는 것을 알 수 있다.

그림 8은 Hall monitor 영상에 대한 결과이다. Hall monitor 영상 역시 앞의 영상들과 같이 높은 PSNR의 결과를 보인 부가정보를 사용하였을 때 적은 Wyner-Ziv 비트만으로 좋은 Wyner-Ziv 영상을 복원 할 수 있다는 것을 볼 수 있다. 움직임이 큰 영상인 Stefan의 경우도 마찬가지이다. 이는 Wyner-Ziv 영상을 복원하기 위해 생성한 부가정보가 잘 생성되어 원 영상과 비슷하게 되면 적은 Wyner-Ziv 비트만으로도 원 영상을 복원 할 수 있음을 의미한다. 또한 H.264 intraframe 부호화 방법과 비교하였을 때에도 Wyner-Ziv 영상을 부호화 하는 방법이 더 좋은 결과를 얻는 것을 확인 할 수 있다.

5. 결론

본 논문에서는 분산 동영상 부호화 시스템에서 다양한 부가정보 생성 방법을 통해 생성된 부가정보가 분산 동영상 부호화 시스템의 성능에 미치는 영향을 비교 분석 하였다. 모의실험을 통해 다양한 부가정보 생성 방법을 통해 생성된 부가정보와 원 영상간의 PSNR 결과가 높은 방법이 분산 동영상 부호화 시스템에 적용하였을 때 더 적은 Wyner-Ziv 비트로도 더 높은 PSNR의 Wyner-Ziv 영상을 복원할 수 있는 것을 입증하였다.

- [1] T. Wiegand, G. Sullivan, G. Bjontegaard and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. on Circuits and Syst. for Video Technol.*, vol. 13, pp. 560-576, July 2003.
- [2] B. Girod, A. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed Video Coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71-83, Jan. 2005.
- [3] A. Aaron, S. Rane, E. Setton and B. Girod, "Transform-Domain Wyner-Ziv Codec for Video," *Proc. SPIE Visual Communications and Image Processing*, San Jose, CA, pp. 520-528, Jan. 2004.
- [4] J. Ascenso, C. Brites and F. Pereira, "Improving Frame Interpolation with Spatial Motion Smoothing for Pixel Domain Distributed Video Coding," *5th EURASIP*, Slovak Republic, pp. 176-181, July 2005.
- [5] X. Artigas, J. ascenso, M. Dalai, D. Kubasov, and M. quaret, "The DISCOVER codec: architecture, techniques and evaluation," *Picture Coding Symposium*, pp. 1103-1120, Nov. 2007.
- [6] C. Brites, J. Ascenso and F. Pereira, "Improving transform domain Wyner-Ziv video coding performance," *IEEE ICASSP*, Toulouse, France, pp. II-525-II-528, May, 2006.
- [7] L. Alparone, M. Barni, F. Bartolini, V. Cappellini, "Adaptively Weighted Vector-Median Filters for Motion Fields Smoothing," *IEEE ICASSP*, Georgia, USA, vol. 4, pp. 2267-2270, May 1996.
- [8] B. D. Choi, J. W. Han, C. S. Kim, S. J. Ko, "Motion-Compensated Frame Interpolation Using Bilateral Motion Estimation and Adaptive Overlapped Black Motion Compensation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 17, pp. 407-416, April 2007.
- [9] E. Peixoto, R. L. Queiroz and D. Mukherjee, "On side information generation for Wyner-Ziv video coding," in *Proc. of Simposio Brasileiro de Tel.*, Rio de Janeiro, Brazil, Sept. 2008.
- [10] J. Ascenso, C. Brites and F. Pereira "Content Adaptive Wyner-Ziv Video Coding Driven by Motion Activity," *IEEE ICIP*, Atlanta, USA, pp. 605-608, Oct. 2006.