

효과적인 분산 비디오 부호화를 위한 비대칭성 움직임 벡터 기반 보조정보 생성 방법

*나태영 **김문철

한국과학기술원

*tyna@kaist.ac.kr **mkim@ee.kaist.ac.kr

Asymmetric Motion Vector-Based Side Information Generation for Efficient Distributed Video Coding

*Taeyoung Na **Munchurl Kim

Korea Advanced Institute of Science and Technology

요약

분산 비디오 부호화(distributed video coding)는 분산 소스 부호화의 대표적인 응용분야로서 부호화 복잡도가 부호화에 서 복호화기로 이동되어 저전력 부호화 환경에 매우 적합하다. 본 논문에서는 분산 비디오 부호화의 성능 향상에 있어 가장 중요한 보조 정보의 효과적인 생성 방법을 제안한다. 우선 보조 정보 생성을 위한 키 프레임들 간의 블록 움직임 추정에 있어 기존 방법들이 대체적으로 가정하고 있는 선형적인 움직임 이동에 따른 잘못된 예측을 해결하기 위해 두 장 이상의 키 프레임을 사용하여 블록 움직임을 추정 후, 선형 회귀(linear regression)를 이용하여 보조 정보 상의 블록 움직임 궤적을 추정한다. 이때 움직임 추정을 위한 키 프레임 번호를 증가하며 선입선출(FIFO)형 버퍼에 저장 및 삭제하여 동일한 보조정보에 해당하는 여러 움직임 벡터 필드와, 기존의 선형적인 움직임이 가정된 움직임 벡터 필드를 동시에 생성한다. 다음으로 보간(interpolation)하려는 보조 정보 프레임 내의 임의의 블록에 가장 가깝게 통과하는 움직임 벡터 필드를 선택하여 해당하는 블록의 최종 움직임 벡터로 선택한다. 실험결과 제안하는 보조 정보 생성 방법은 기존의 방법과 비교했을 때 비대칭성 움직임 벡터 사용만으로 평균 PSNR이 0.216dB 만큼 증가하는 것을 확인할 수 있었다.

1. 서론

분산 비디오 부호화는 기본적으로 부호화기가 아닌 복호화기에서 부호화를 위한 움직임 예측 및 보상, DCT 변환 등을 수행하여, 부호화기에서 전송되지 않는 프레임을 복원하는 것이 기존 부호화 방식과의 큰 차이점이라 할 수 있다. 반면 부호화기는 전송하지 않는 프레임들을 채널 부호화하여 생성된 패리티 비트들과 전송되는 프레임들을 기존의 방법으로 화면 내 예측(intra prediction) 부호화 후 전송하는 간단한 역할만을 담당한다.

이론적으로 D. Slepian, J. Wolf, A. Wyner와 J. Jiv의 연구에 기초를 두고서 초기의 분산 비디오 부호화 연구는 B. Girod와 K. Ramchandran에 의해 전체적인 부호화기 및 복호화기의 구조가 최초로 제안되었으며, 그 후 유럽의 6개국이 연합하여 결성한 DICSCOVER(DISTRIBUTED CODING FOR VIDEO SERVICES) 그룹에서 부/복호화기 구조와 부호화 및 복호화 성능 향상에 대한 연구 및 실용화 검토가 본격적으로 시작되었으며 그 후 많은 연구자들에 의해 부호화 성능을 향상시키기 위한 노력이 전개되어 현재에 이르고 있다.[1][2][3][4][5]

분산 비디오 부호화의 성능에 가장 큰 영향을 미치는 부분은 효과적인 보조 정보의 생성에 있는데 보조정보는 차후의 채널 복호화를 통해 최종적으로 복원되어 전송되지 않은 프레임으로 간주되기 때문이

다. 본 논문에서는 보조 정보 생성을 위한 키 프레임들 간의 블록 움직임 추정 후, 기존 방법들이 대부분 가정하고 있는 선형적인 움직임 이동에 따른 잘못된 예측을 해결하기 위해 두 장 이상의 키 프레임들 상에서의 움직임을 추정한 후, 선형 회귀(linear regression)를 이용하여 보조 정보 상의 블록들의 움직임 궤적을 보다 정밀하게 추정한다. 본 논문의 구성은 다음과 같다. 우선 2장에서는 보조 정보 생성과 관련된 기존의 방법들에 대해 간단히 언급하고 3장에서는 제안하는 선형 회귀를 통한 블록 궤적 추정에 따른 보조 정보 생성 방법에 대해 구체적으로 언급한다. 4장에서는 제안하는 방법과 기존 방법과의 성능을 율-왜곡(rate-distortion) 측면에서 비교하며 5장에서 결론을 맺는다.

2. 보조 정보 생성

분산 비디오 부호화의 원리는 기본적으로 부호화기에서 전송되지 않은 프레임 정보를 복호화기에서 최대한 유사하게 생성한 후, 부호화기에서 생성된 패리티 비트들만을 활용하여 채널 복호화 기법을 통해 원 영상을 복원하는 데 있다. 여기서 부호화기에서 전송되지 않은 프레임 정보와 최대한 유사하게 복호화기에서 생성된 프레임을 보조 정보라 하며 분산 비디오 부호화의 성능에 가장 큰 영향을 미친다. 대표적인 보조 정보의 생성 방법은 MCFI(Motion Compensated Frame Interpolation) 방식에 기반을 둔 J. Ascenso가 제안한 방법이 아직도

많은 연구들에 참조되고 있는데, 이는 생성하려는 보조 정보의 앞과 뒤에 위치한 복호가 끝난 키 프레임들 간 움직임 예측(motion estimation)된 움직임 벡터들이 해당 보조 정보에서는 정확히 절반의 크기를 갖는다는 가정 하에 움직임 보상(motion compensation)을 실시하게 된다.[6] 또한 핀-홀 현상을 방지하기 위해 보조 정보를 임의의 블록 크기로 나눈 다음, 가장 가까운 지점을 통과하는 움직임 벡터를 선택하여 해당하는 키 프레임 상의 화소 값들로 정한 후, 블록 단위의 움직임 보상을 실시한다. 마지막으로 잘못된 보상을 방지하기 위해 주위 블록들의 움직임 벡터들과의 적응적 중간 여파(adaptive median filtering)를 수행한다.

3. 제안하는 선형 회귀를 통한 움직임 벡터 도출 방법

앞서 언급한 J. Ascenso의 방법뿐 아니라 MCFI에 기반을 둔 대부분의 보조 정보 생성 방법들은 움직임 예측에 활용되는 블록들의 움직임이 선형적이라는 가정을 두고 있다. 하지만 이러한 방법들은 프레임 내에서의 실제 물체나 촬영 카메라의 움직임이 선형적이지 아닌 경우에 대해선 부호화 성능을 떨어뜨릴 수 있다. 제안하는 방법은 움직임 예측을 위한 키 프레임 수를 [6]과 같이 두 장으로 고정하지 않고, 여러 장을 한 번에 버퍼에 저장한 후, 첫 번째 키 프레임을 일정한 블록 크기로 나눈 후, 모든 블록들의 이동 경로들을 MAD(Mean Absolute Distortion) 기법을 사용하여 탐색한다. n장의 키 프레임들이 버퍼에 저장되었기 때문에 최초 시작하는 키 프레임상의 임의의 블록에 대해 x, y축에 대해 각각 n개의 좌표가 도출된다. n개의 좌표들을 바탕으로 변이 값에 대해 선형 회귀를 실시하여 키-프레임 사이에 위치하는 n-1개의 보조 정보 프레임들 상의 모든 블록을 움직임 벡터들을 결정하게 된다. 그림 1에 n을 4로 하였을 때 키 프레임들 간 움직임 예측을 통해 움직임 궤적이 비선형으로 추정되는 경우 보조 정보 프레임들 상에 예상되는 블록 위치를 도시화한 결과를 나타내고 있다.

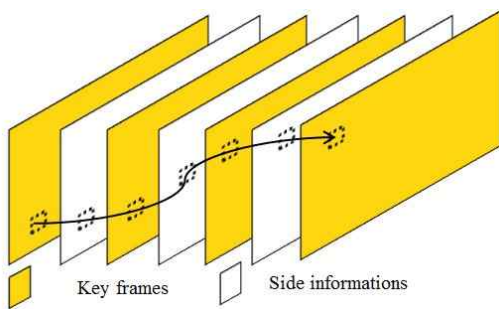


그림 1 비선형적 궤적을 갖는 블록 움직임의 예 (n=4)

키 프레임 상에서 FME(Forward Motion Estimation)를 수행하는 기본 블록 크기를 8x8로 한다면 QCIF영상의 경우에 총 396개의 블록들이 존재하게 되며 이들 각각에 대해 움직임 궤적 추적을 위한 선형 회귀 모델이 필요하게 된다. 선형 회귀에는 여러 가지 방법이 있으나, 본 논문에서는 블록 궤적을 추적하는 본래의 목적에 맞게 곡선 집합(curve fitting)을 위한 선형 회귀법을 사용하도록 하며 이는 [7]에 잘 소개되어 있다. 간략히 소개하면 우리가 선형 회귀를 통해 구하고자 하는 값은 보조 정보 프레임 상의 블록 위치이며 만약 n이 4인 경우, 임

의 블록에 대해 키 프레임 FME를 수행한 후 결정된 블록 위치를 학습(training)을 위한 목적 값(target value, t)으로 둔다면 다음 모델을 생각할 수 있다.

$$y(x, \bar{w}) = w_0 + w_1\Phi_1(x) + \dots + w_3\Phi_3(x) \quad (1)$$

$$= w_0 + \sum_{j=1}^3 w_j\Phi_j(x)$$

여기서 $\Phi_j(x)$ 는 기저 함수이고, (w_0, w_1, w_2, w_3) 는 모델의 특성을 결정하는 매개 변수(parameter)이다. 이제 최적의 모델 매개 변수를 도출하는 문제가 남아 있는데, 이는 주어진 학습 세트(training set)인 \bar{x} 와 이에 대응하는 목적 값인 \bar{t} 값을 최소 자승법(least square) 측면에서 매개 변수 벡터인 \bar{w} 를 구하는 것이 바람직하다. 학습이 완료되어 새로운 입력 x 가 주어졌을 때, 이에 상응하는 목적 값 t 는 가우시안 분포를 따르는 함수 $y(x, \bar{w})$ 의 평균이라고 가정하면 다음의 확률 분포를 생각할 수 있다.

$$p(t|x, \bar{w}, \beta) = N(t|y(x, \bar{w}), \beta^{-1}) \quad (2)$$

우리는 학습을 위한 데이터인 $\{\bar{x}, \bar{t}\}$ 를 이용하여, 미지수인 \bar{w} 와 β 를 구하는 것이 목적이고, 각각의 \bar{x} 가 서로 독립적인 사건이라면 다음과 같은 최대 우도(maximum likelihood) 식을 생각할 수 있다.

$$p(\bar{t}|\bar{x}, \bar{w}, \beta) = \prod_{n=1}^N N(t_n|y(x_n, \bar{w}), \beta^{-1}) \quad (3)$$

β 는 가우시안 분포의 폭을 결정하는 분산을 의미한다. 일련의 과정을 거쳐 우리는 최종적으로 다음의 최적 선형 회귀 파라미터 \bar{w}_{opt} 를 구할 수 있다.

$$\bar{w}_{opt} = \{(\bar{\Phi})^T(\bar{\Phi})\}^{-1}(\bar{\Phi})^T\bar{t} \quad (4)$$

여기서 $\bar{\Phi}$ 는 설계 행렬(design matrix)로서 다음을 의미한다.

$$\bar{\Phi} = \begin{pmatrix} \Phi_0(x_0) & \dots & \Phi_3(x_0) \\ \vdots & \ddots & \vdots \\ \Phi_0(x_3) & \dots & \Phi_3(x_3) \end{pmatrix} \quad (5)$$

입력 값에 해당하는 키 프레임 번호가 n이 4일 경우 각각 (0,2,4,6)이므로 \bar{w}_{opt} 도출하여 $y(x, \bar{w})$ 의 x 에 보조 정보 프레임 번호(1, 3, 5)를 각각 입력하면 선형 회귀에 의한 블록 위치 변이를 얻을 수 있다. 또한 좀 더 정확한 움직임 벡터 필드를 도출하기 위해 그림 2와 같은 구조를 고안하였다.

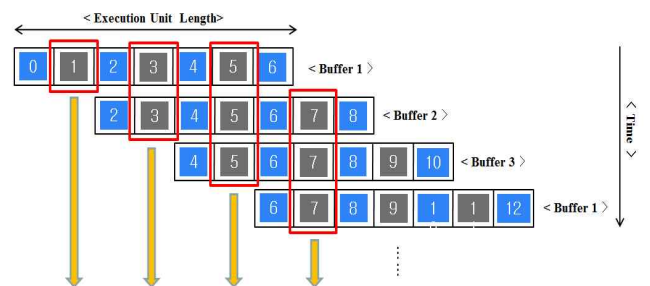


그림 2 움직임 벡터 추출을 위한 실행 단위(execution unit)

그림 2의 구조는 네 장의 키 프레임들 간의 FME를 통해 도출되는

움직임 벡터들이 비교적 적은 표본으로의 잘못 학습되어 보조 정보가 잘못 생성되는 것을 방지하고, 여러 움직임 벡터 후보들을 고려하여 가장 최적의 움직임 벡터를 선택할 수 있는 여건을 제공한다. 그림 2에 나타난 바와 1, 3번 보조 정보 프레임에 제외하고는 한 장의 보조 정보 생성을 위해 세 가지의 움직임 벡터들을 고려하며 여기에 영상의 움직임이 비교적 선형적인 경우를 고려하기 위해 [6]의 방법으로 도출된 움직임 벡터들도 포함하여 총 네 가지의 움직임 벡터들 중에서 가장 최적값을 선택하게 된다. [6]에서의 방법과는 다르게 복수의 움직임 후보군 중에서 최적의 경우를 선택해야 하므로 선택을 위한 판별식이 필요하게 된다. 판별식은 다음과 같이 정의한다.

$$J_e = \alpha \cdot D_{distortion} + (1 - \alpha) \cdot D_{distance} \quad (6)$$

식 (6)의 첫 번째 항은 보조 정보 상의 블록에 할당된 후보 움직임 벡터들로 양방향 움직임 보상을 실시하였을 때의 오차이고, 두 번째 항은 각 블록에 움직임 벡터를 할당할 때 블록의 원점과 움직임 벡터가 통과하는 지점과의 거리를 의미한다. 다양한 실험 영상에 대해 실험한 결과 α 의 값은 0.3일 때 가장 최상의 성능을 보여주었다.

4. 실험 결과

제안하는 방법을 검증하기 위해 자체적으로 구현된 분산 비디오 부호화기를 사용하였다. 전체적으로 [6]에서 제안하는 방법과 유사하게 구현하였으며, 키 프레임 부호화를 위해서는 H.264/MPEG-4 Part 10 AVC 참조 소프트웨어인 JM 13.2를 사용하였다.[8] 실험 영상으로는 QCIF 크기의 *Foreman*, *Carphone*, *Soccer* 영상을 각각 100 프레임씩 사용하였으며, 키 프레임간의 FME를 위한 탐색 영역은 ± 16 으로 하였고 한 버퍼에 저장되는 키 프레임 수는 4장으로 하였다. 모든 키 프레임은 베이스라인 프로파일로 QP 27의 조건으로 I-프레임 부호화 하였다. 또한 앞장에서 언급한 선형 회귀 모델을 위한 기저 함수는 $\Phi_j(x) = x^j$ 으로 하였다.

표 1. 제안된 방법과 [6]의 비교 실험 결과 (WZ 프레임만)

Sequences		QP 27		
		PSNR(dB)		
		FME	BME	Δ (dB)
<i>Foreman</i>	Ascenso's	31.41	32.44	0
	Proposed	31.41	32.50	+0.06
<i>Carphone</i>	Ascenso's	30.28	30.71	0
	Proposed	30.73	31.05	+0.34
<i>Soccer</i>	Ascenso's	27.65	28.11	
	Proposed	27.75	28.36	+0.25
평균	Ascenso's	-	-	0
	Proposed	-	-	+0.216

표 1은 기존의 [6]의 방법에 비해 울-왜곡 성능 측면에서 제안하는 방법이 더 우수함을 잘 드러내고 있다. *Foreman* 영상의 경우 전체적으로 볼 때 비선형적인 움직임이 크지 않기 때문에 제안하는 방법에서의 선택되는 비율이 낮아 성능 향상이 두드러지지 않지만, *Carphone* 영상의 경우 차량 바깥 풍경이 빠르게 변하고 화자의 움직임도 불규칙적으로 빠른 편이기 때문에 제안하는 방법의 부호화 성능이 더 뛰어난 것을 확인할 수 있다.

5. 결론

본 논문에서는 기존의 방법들이 대부분 키 프레임간의 블록 움직임이 선형적이지 않은 경우에 보조 정보의 생성이 잘못되는 것을 방지하기 위해 여러 장의 키 프레임을 저장한 후, 움직임을 예측한 후, 선형 회귀를 통해 실제 궤적을 추론한 후, 해당 보조 정보 프레임에 해당하는 블록 위치를 보간하는 방법을 제안하였다. 활용하였다. 제안하는 방법을 다양한 실험 영상을 통해 실험한 결과, 울-왜곡 측면에서 기존의 방법과 비교하여 PSNR이 최대 0.34dB 향상됨을 확인하였다.

참고문헌

- [1] D. Slepian and J.K. Wolf, "Noiseless coding of correlated information sources," *IEEE Transactions in Information Theory*, Vol. 19, pp.471-480, July. 1973.
- [2] A. Wyner and J. Ziv, "The rate-distortion function for sources coding with side information at the decoder," *IEEE Transactions in Information Theory*, Vol. 22, pp. 1-10, Jan. 1976.
- [3] B. Girod, A. Aaron, S. Rane and D. R. Monedero, "Distributed Video Coding," *Proceedings of the IEEE*, Vol.93, No.1, pp. 7-83, Jan. 2005.
- [4] R. Puri and K. Ramchandran, "PRISM: A New Robust Video Coding Architecture Based on Distributed Compression Principles," *40th Allerton Conference on Communication, Control and Computing*, Allerton, USA, Oct. 2002.
- [5] DISCOVER: <http://www.img.lx.it.pt/~discover/home.html>
- [6] J. Ascenso, C. Brites and F. Pereira, "Improving Frame Interpolation with Spatial Motion Smoothing for Pixel Domain Distributed Video Coding," *5th Eurasic Conference on Speech, Image Processing, Multimedia Communications and Services, Smolenice, Slovak Republic*, July 2005.
- [7] C. Bishop, "Pattern Recognition and Machine Learning," Springer, New York, 2006.
- [8] http://iphome.hhi.de/suehring/tml/download/old_jm/