

모션 벡터들의 분산값을 이용한 프레임률 상향 변환

*양순모 **김규헌

경희대학교

*nuclear944@naver.com **kyuheonkim@khu.ac.kr

Frame Rate Up Conversion by the Variance of Motion Vectors

*Yang, Soon Mo **Kim, kyuheon

Kyunghee University

요 약

본 논문에서는 계산의 복잡성을 줄이고 피크 신호 대 잡음 비율(PSNR) 성능을 개선하기 위한 새로운 프레임 상향 변환 (Frame Rate Up Conversion) 알고리즘을 제안한다. 제안된 알고리즘을 사용하기 위한 모션 추정 과정(Motion Estimation)은 이전 프레임과 현재 프레임에서 매크로블록(Macroblock) 값의 최소 차이값(Sum of absolute differences)을 이용하여 보간된 프레임(Interpolated Frame)의 매크로블록이 가지게 되는 모션 벡터 값을 추출한다. 이 때 반복된 배경 패턴 및 여러 움직임들 때문에 모션 추정 과정에서 출력되는 벡터값이 비정상적으로 출력되는 경우가 있다. 여기서 제안된 알고리즘을 통해 모션 벡터값들의 특이치(Outlier)를 검출하고 이를 교정하기 위한 분산값(Variance)을 이용하여 모션 벡터 평활화 작업(Motion Vector Smoothing)을 거친다. 이와 같이 제안된 알고리즘을 이용하여 실험한 결과값으로 프레임률 상향 변환 과정을 통해 상대적으로 계산의 복잡성은 낮으면서 양호한 PSNR 값이 출력됨을 확인할 수 있다.

1. 서론

최근에 많이 사용 되고 있는 LCD(Liquid Crystal Display)에서 동영상을 재생할 때 액정의 응답속도가 느려 급격한 움직임이 포함된 출력에 움직임 흐림 현상(Motion Blur)가 발생하는 경우가 있다.[1] 특히 LCD를 많이 사용하는 디지털 TV 디스플레이 기기와 관련하여 위와 같은 문제점을 해결하기 위해 FRUC(Frame Rate Up Conversion)과 같은 방법이 주목을 받았다. 여기서 FRUC는 프레임 수를 증가시키는 하나의 방법으로 원래 프레임 사이에 새로운 프레임을 보간하여 영상의 연속성을 증가 시키는 것으로 60Frame/s에서 120Frame/2 그리고 240Frame/2로 높임으로써 움직임 흐림 현상을 해결하는데 활용된다.[1],[2].

일반적으로 프레임을 상향 변환기법(FRUC)은 새로운 프레임을 보간하기 위해 다음과 같은 프로세스를 추정한다. 먼저 움직임 추정 과정(Motion Estimation)을 거친다.[3],[8] ME 과정은 두 개 이상의 연속 프레임 사이에서 움직이는 물체의 변위를 계산하여 모션 벡터 탐색한다. 그 다음으로 ME 과정에서 얻은 모션 벡터로 보간된 프레임을 만드는 과정을 거친다. 이 때 FRUC에서 영상 화질은 모션 벡터의 정확도에 따라 달라지기 때문에 FRUC에서 모션 벡터의 실제 운동 궤적을 찾는 것이 매우 중요시 여겨진다.[3][4]

지금까지 여러 FRUC에 대해 Frame Repetition 방법, Frame Averaging 방법 등 초기에 다양한 방법들이 고안되었으나 이와 같은 방법들은 물체의 움직임이 매우

동적이면 물체가 흐릿하게 보이는 문제점을 보여준다. 따라서 이와 같은 문제점들을 보완하기 위해 움직임 보상 보간(Motion Compensated Interpolated) 방법이 고안이 되었다.[3-8] 대부분의 MCI에서 모션을 추정하기 위해 매크로블록들의 움직임을 추정하는 블록 정합 기법(Block Matching Algorithm)을 이용한다. 이러한 BMA에는 단방향에서 모션을 추정하는 단방향 움직임 추정(Unilateral Motion Estimation) 방법이 있다.[7],[8] UME는 현재 프레임의 블록을 중점으로 중간 프레임을 보간하는 방법이다. 하지만 UME를 이용하면 Fig. 1.1과 같이 중간 프레임에서 겹치는 부분(Overlap)과 보간이 안된 부분인 (Hole)이 발생하게 된다.[7-10]

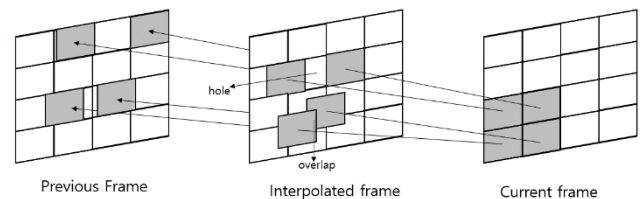


Fig 1.1 Unilateral Motion Estimation(UME)

이러한 중첩된 영역과 홀의 문제점을 해결하기 위해 중간값을 이용하거나 인접한 움직임들의 정보를 이용한 방법들이 있다. 하지만 이 방법들은 복잡한 연산을 요구하고 보간 영상에 원하지 않는 화질 저하를 불러오는 문제점을 갖는다.[4],[8] 따라서 이 문제점들을 해결하기 위해 UME가 아닌 양방향

움직임 추정(Bilateral Motion Estimation) 방법이 고안되었다 [8-10] Fig 1.2 에서 보이는 바와 같이 BME 는 양방향 움직임을 예측하여 보간하게 되므로 중첩 부분과 홀의 부분이 더 이상 생기지 않고 UME 와 비교하여 더 적은 연산량으로 벡터들의 움직임을 찾을 수 있다.[7],[8],[11-13]

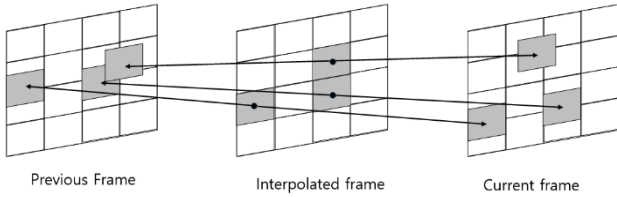


Fig 1.2 Bilateral Motion Estimation(BME)

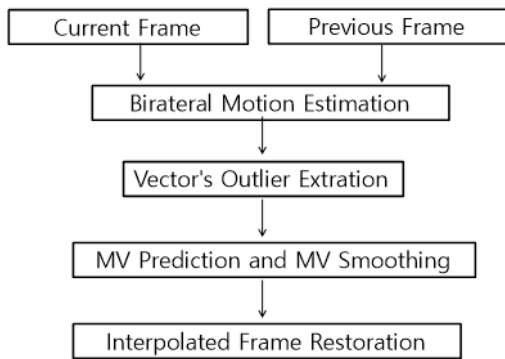


Fig 1.3 Flowchart of the proposed algorithm .

UME 또는 BME 를 이용하여 모션 벡터를 추출하면 모션 벡터를 갖는 각 매크로블록들의 인접 블록과 관련되어 특이치(Outlier) 가 생성된다. 따라서 이를 검출하고 수정하기 위해 ME 과정을 거치고, 모션 벡터 평활화 작업(Motion Vector Smoothing)에 들어간다. 기존 MVS 작업은 주변 MV 의 단순 평균을 계산하는 방식을 이용하여 그 값이 부정확하거나 정확성은 뛰어나나 연산과정이 매우 복잡한 단점이 있다. 따라서 본 논문에서는 기존 MVS 의 장,단점을 보완하기 위해 분산값을 이용하여 보다 더 정확하면서 연산과정도 복잡하지 않는 다음과 같은 알고리즘을 제시한다.

2. 제안된 알고리즘

2.1 양방향 움직임 추정(BME)

양방향 움직임 추정기법(BME) 는 중간 프레임의 한 블록에 대응하는 블록을 이전 프레임과 현재 프레임에서 크기는 같고 방향은 반대인 모션 벡터를 이용하여 구한다. 이 때 블록들의 모션벡터는 다음 식 (2.1)의 Sum of bilateral Absolute Differences(SBAD) 값이 가장 작은 값을 사용한다.[7][8]

$$SBAD(dx, dy) = \sum_x \sum_y |f_p(x - dx, y - dy) - f_c(x - dx, y - dy)| \quad (2.1)$$

$$v = \operatorname{argmin}\{SBAD(dx, dy)\}$$

여기서 (dx,dy)는 각 벡터의 변화량을 나타내며 f_p 는 이전 프레임, f_c 는 현재 프레임을 가리킨다. 이와 같은 방법을 거쳐 각 블록마다 모션 벡터들을 추출하여 Fig 2.1 과 같이 모션 벡터 분포를 도식화 시킬 수 있다.

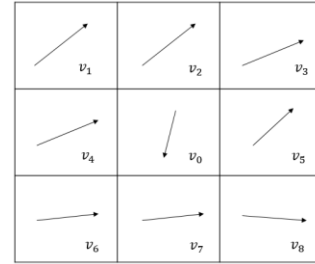


Fig 2.1 Predicted Motion Vectors

2.2 분산값을 이용한 모션 벡터 평활화

모션 벡터 평활화(MVS) 의 핵심은 주변 벡터에 비해 급격하게 다른 방향으로 향하고 있는 잘못된 모션 벡터 특이값(Outlier) 를 추출하고 이를 수정하는 것이다. 여기서 특이값은 블록화 현상을 발생시키는 주된 요인으로 주관적인 화질에 좋지 않은 영향을 끼친다. 따라서 특이값을 추출하기 위해 다음의 수식을 이용하여 판별한다.

$$v_m = \frac{1}{9} \sum_{i=0}^8 v_i \quad (2.2)$$

$$D_c = |v_m - v_0| \quad (2.3)$$

$$D_n = \frac{1}{8} \sum_{i=1}^9 |v_m - v_i| \quad (2.4)$$

위 식을 계산하여 만약 $D_c > D_n$ 이라면 v_0 는 특이값(Outlier) 로 판정된다. 이와 같은 방법을 이용하여 모든 블록에 걸쳐 특이값들을 검출 한 후 다음 제안된 알고리즘을 이용하여 모션 벡터 평활화 작업을 거친다.[8]

식(2.5)는 벡터 평활화 작업을 위해 먼저 각 벡터 값들이 평균값으로부터 얼마나 신뢰할 수 있는지 분산값을 계산하기 위해 편차의 제곱을 구하는 과정을 나타낸다.

$$w_i = (v_i - v_m)^2 \quad (2.5)$$

v_m 을 기준으로 그 값의 변화량이 클수록 연관성을 줄이기 위해 역수를 취한 후 식(2.6)과 같이 가중치를 정의한다.

$$k_i = \frac{1}{w_i} \quad (2.6)$$

앞서 특이값 검출 과정이 끝난 후, 특이값이 검출 되면 다음과

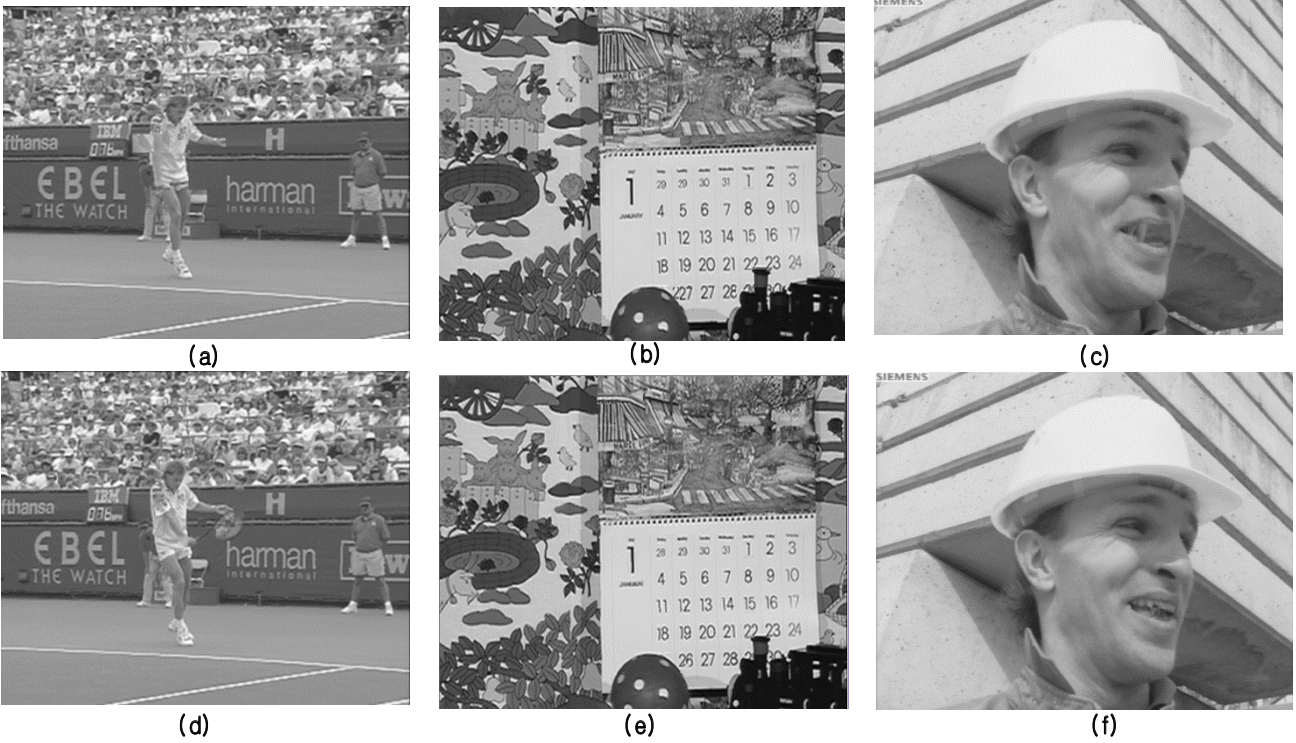


Fig.3 Interpolated frame by bilateral motion estimation (a)Stefan (b)Mobile (c)Foreman
Bilateral motion estimation with proposed algorithm (e)Stefan (f)Mobile (g)Foreman

같이 블록을 4 등분으로 분해를 한 후 각 블록에 해당되는 벡터를 식(2.7)과 같은 과정을 거쳐 벡터 평활화 과정을 진행시킨다. 이 때 나뉜진 4 칸을 기준으로 x_1 의 값을 설정할 때 이와 인접한 벡터인 v_1, v_2, v_4 값에 (2.5)에 구한 가중치를 이용하여 값을 구한다. 나머지 벡터 x_i 값들도 같은 방법으로 그 값을 구한다.

| | | |
|-------|-------------|-------|
| v_1 | v_2 | v_3 |
| v_4 | x_1 x_2 | v_5 |
| | x_3 x_4 | |
| v_6 | v_7 | v_8 |

Fig. 2.2 Motion Vector Smoothing

$$x_i = \frac{k_i v_i + k_j v_j + k_k v_k}{k_i + k_j + k_k} \quad (2.5)$$

이와 같은 과정을 통해 상대적으로 복잡성은 낮으면서 모션 벡터의 정확성을 높이는 결과를 도출해 낼 수 있다. 마지막으로 이렇게 구한 모션 벡터를 디코딩 과정을 거쳐 보간된 프레임을 복원한다.

3. 실험결과

본 실험에서는 블록의 크기를 16 x 16 로 설정하고 검색 범위를 10 픽셀로 설정하였다. 그리고 30Hz 프레임률을 가진 CIF : 352 x 288 의 영상을 가지고 실험을 하였다. 실험방법은 FRUC 알고리즘을 이용하여 30Hz 의 프레임률을 가진 원본영상의 짝수 프레임들을 제거하고 제안된 알고리즘을 이용하여 새로운 프레임을 보간하여 실험을 진행하였다. 그리고 제안된 알고리즘의 성능을 객관적으로 비교하기 위해 벡터 평활화 작업이 이루어지지 않은 보간된 프레임과 평활화 작업이 이루어진 보간된 프레임의 피크 신호 대 잡음 비율을 비교함으로써 보간 프레임의 객관적 영상 품질을 평가했다.

Table1
PSNR 값 비교(dB)

| | BME | [8] | [7] | Proposed Algorithm |
|---------|-------|-------|-------|--------------------|
| Stefan | 23.31 | 27.33 | 24.36 | 27.16 |
| Mobile | 24.74 | 26.91 | 24.91 | 26.64 |
| Foreman | 26.23 | 31.13 | 27.92 | 32.37 |
| News | 30.76 | 35.99 | 31.98 | 36.32 |

Fig 3.(a)와 (e), (b)와 (f) 그리고 (c)와 (g)는 주관적인 성능을 비교 할 수 있도록 나타내었다. Fig 3.(a)에서 벡터 평활화 작업을 적용하지 않았을 때, 동적인 영역인 라켓을

휘두르는 부분이 크게 손상되었다. 하지만 평활화 작업을 거친 Fig 3.(e)를 보면 완벽하지는 않지만 상대적으로 양호하게 동적인 라켓부분이 개선된 것을 확인할 수 있다. Fig 3.(b)를 보면 공이 굴러가면서 순간적으로 26 일이 나타나는 부분에 벡터가 잘못 추출되어 27 일로 출력되었음을 확인할 수 있다. 또한 12월 28 일을 나타내는 숫자도 29 일로 출력되었음을 보여준다. 벡터 평활화 작업을 거친 Fig 3.(f)을 보면 벡터의 특이값을 검출하여 상대적으로 정상적인 영상으로 복원되었음을 보여준다. Fig 3.(c)를 보면 남성의 입안의 영역에서 움직임이 상대적으로 동적이어서 흐릿하게 영상이 출력되었음을 확인할 수 있다. 벡터 평활화 작업을 거친 Fig 3.(f)를 보면 동적인 입안 영역에서 흐릿한 영역이 대부분 제거된 상태로 주관적으로 매우 양호하게 출력되었음을 보여준다. 그리고 객관적인 값을 비교하기 위해 PSNR 값을 비교한 Table 1을 보면 제안된 알고리즘이 상대적으로 양호한 객관적인 성능을 보여준다. 제안된 알고리즘의 평균 PSNR은 평활화 작업을 적용하지 않은 영상들을 비교했을 때, PSNR 값이 Stefan의 경우에는 약 4dB, Mobile은 2dB, Foreman은 6dB, News는 6dB 정도 향상되었다. 그리고 연산량이 높은 [8]과 비교했을 때, 동적인 움직임이 큰 Stefan과 Mobile은 [8]의 PSNR 값이 조금 더 양호하게 출력되었으나 그 차이가 매우 미세하다. 그리고 정적인 부분에서는 제안된 알고리즘이 양호하게 출력되었다. 결론적으로 제안된 알고리즘은 복잡성이 상대적으로 높은 알고리즘과 비교했을 때, 연산량은 낮으면서 성능은 객관적으로 양호하게 출력된다는 것을 확인할 수 있다.

4. 결론

본 논문에서는 분산값을 이용한 벡터 평활화 방법에 대한 알고리즘을 제안했다. 제안된 알고리즘은 모션 벡터의 특이값을 검출하여 이를 분산값을 이용한 파라미터와 연관시켜 그 값을 수정하고 영상을 복원한다. 실험 결과에서 확인할 수 있는 것처럼 상대적으로 양호하게 주관적인 영상 품질이 개선되었다는 점을 확인할 수 있고, 객관적인 영상 품질을 알 수 있는 PSNR 값도 확인할 수 있다. 향후 연구에서는 경계 영역에서의 벡터들의 성능을 개선시킬 수 있는 방법에 대한 고찰, 분산값이 아닌 다른 파라미터를 적용하여 신뢰성을 향상시키는 방법, 그리고 벡터값의 신뢰도를 상승하여 이를 추정하는 방법의 정확성을 상승시키는 방법 등에 대해 고찰할 필요가 있다.

5. 참고문헌

- [1] J. Someya, N. Okuda and H. Sugiura, "The Suppression of Noise on a Dithering Image in LCD Overdrive," *IEEE Trans. Consumer Electron.*, vol. 52, no. 4, pp. 1325–1332, 2006
- [2] K. Hilman, H. W. Park, and Y. Kim, "Using motion-compensated frame-rate conversion for the correction of 3:2 pulldown artifacts in video sequences," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 10, pp. 869–877, 2000.
- [3] S.-J. Kang, D.-G. Yoo, S.-K. Lee, and Y. H. Kim, "Multiframe-based bilateral motion estimation with emphasis on stationary caption processing for frame rate up-conversion," *Consumer Electronics, IEEE Transactions on*, vol. 54, pp. 1830–1838, 2008.
- [4] S. Liu, C.-C. J. Kuo, and J. W. Kim, "Hybrid global-local motion compensated frame interpolation for low bit rate video coding," *J. Visual Commun. Image Represent.*, vol. 14, pp. 61–

- 79, 2003.
- [5] T. Liu, K.-T. Lo, J. Feng, and X. Zhang, "Frame interpolation scheme using inertia motion prediction," *Signal Process.: Image Commun.*, vol. 18, pp. 221–229, 2003.
- [6] J. Benois-Pineau and H. Nicolas, "A new method for region-based depth ordering in a video sequence: application to frame interpolation," *J. Visual Commun. Image Represent.*, vol. 13, pp. 363–385, 2002.
- [7] B.-D. Choi, J.-W. Han, C.-S. Kim, and S.-J. Ko, "Frame rate up-conversion using perspective transform," *Consumer Electronics, IEEE Transactions on*, vol. 52, pp. 975–982, 2006.
- [8] D.-G. Yoo, S.-J. Kang, and Y. H. Kim, "Direction-select motion estimation for motion-compensated frame rate up-conversion," *Display Technology, Journal of*, vol. 9, pp. 840–850, 2013.
- [9] S.-C. Han and J. W. Woods, "Frame-rate up-conversion using transmitted motion and segmentation fields for very low bit-rate video coding," in *Proc. Int. Conf. Image Process.*, 1997, pp. 747–750.
- [10] T.-Y. Kuo and C.-C. J. Kuo, "Motion-compensated interpolation for lowbit-rate video quality enhancement," in *Proc. SPIE Conf. Appl. Digital Image Process.*, 1998, pp. 277–288.
- [11] T. Ha, S. Lee, and J. Kim, "Motion compensated frame interpolation by new block-based motion estimation algorithm," *IEEE Trans. Consum. Electron.*, vol. 50, no. 2, pp. 752–759, May 2004.
- [12] S.-J. Kang, K.-R. Cho, and Y. H. Kim, "Motion compensated frame rate up-conversion using extended bilateral motion estimation," *IEEE Trans. Consum. Electron.*, vol. 53, no. 4, pp. 1759–1767, Nov. 2007.
- [13] S.-J. Kang, D.-G. Yoo, S.-K. Lee, and Y. H. Kim, "Multiframe-based bilateral motion estimation with emphasis on stationary caption processing for frame rate up-conversion," *IEEE Trans. Consum. Electron.*, vol. 54, no. 4, pp. 1830–1838, Nov. 2008