

이웃하는 블록 정보와 양방향 움직임 예측을 이용한 적응적 프레임 보간 기법

준회원 오 형 철*, 정회원 이 주 현*, 준회원 민 창 기*, 정회원 정 제 창*

Adaptive Frame Rate Up-Conversion Algorithm using the Neighbouring Pixel Information and Bilateral Motion Estimation

Hyeongchul Oh* *Associate Member*, Joohyun Lee* *Regular Member*,
Changki Min* *Associate Member*, Jechang Jeong* *Regular Member*

요 약

본 논문에서는 비트율의 증가와 동시에 주관적 화질의 향상을 얻을 수 있는 알고리즘인 Frame Rate Up-Conversion (FRUC)을 제안한다. 제안하는 알고리즘에서는 단방향이 아닌 양방향의 움직임 예측을 통해 신뢰도가 높은 블록을 탐색하여 두 장의 예비 프레임을 생성한다. 예비 프레임 생성과정에서는 설정된 임계값 보다 작은 경우를 제외한 나머지 영역에 대해서는 폐색 구간으로 설정하기 때문에 해당 구간에 대한 추가적인 보간 과정이 수행된다. 폐색 구간을 중심으로 이용 가능한 이웃하는 블록의 수를 고려하여 적응적으로 참조 라인을 설정하여 추가적인 재탐색 과정을 수행한다. 선택된 최적의 블록이 두 장의 예비 프레임에 모두 존재할 경우와 한 장에만 존재할 경우를 고려하여 선택적으로 보간한다. 본 논문에서는 제안하는 알고리즘의 성능을 입증하기 위해 기존에 제안되었던 알고리즘과의 PSNR 및 주관적 화질을 비교한다. 또한 실험 결과를 통해 기존의 알고리즘 보다 제안하는 알고리즘이 우수한 성능을 나타냄을 확인 할 수 있다.

Key Words : Frame up-conversion, Frame interpolation, Bilateral prediction, MCFL, FRUC

ABSTRACT

In this paper, we propose a new Frame Rate Up-Conversion (FRUC) scheme to increase the frame rate from a lower number into a higher one and enhance the decoded video quality at the decoder. The proposed algorithm utilizes the preliminary frames of forward and backward direction using bilateral prediction. In the process of the preliminary frames, an additional interpolation is performed for the occlusion area because if the calculated value of the block with reference frame is larger than the predetermine threshold the block is selected as the occlusion area. In order to interpolate the occlusion area, we perform re-search to obtain the osiomal block considerhe osiomnumber of available ne block consblock. The experimental results show that performance of the proposed algorithm has better PSNR and visual quality than the conventional methods.

I. 서 론

최근 디스플레이 장치의 급속한 발전으로 다양한

크기의 영상 포맷과 방대한 양의 고해상도 비디오의 수요가 증가하고 있다. 그러나 제한된 대역폭에서 고해상도의 데이터를 전송하기 위해서는 비트율을 고려

※ 본 연구는 서울시 산학연 협력사업(10570)으로 구축된 서울 미래형 콘텐츠 컨버전스 클러스터 지원으로 수행되었습니다.

※ 이 논문은 2010년도 두뇌한국21사업에 의하여 지원되었음.

* 한양대학교 전자컴퓨터통신공학과 영상통신및신호처리 연구실(ohc8203@hotmail.com)

논문번호 : KICS2010-04-157, 접수일자 : 2010년 04월 07일, 최종논문접수일자: 2010년 8월 19일

해야 한다. 주관적 화질의 향상을 위한 고해상도의 비디오의 경우, 허용된 대역폭 내의 범위로 비트율을 줄여야 하기 때문에 비디오의 주관적 화질은 떨어진다. 이와 같은 비트율이 감소함으로 인해 발생하는 화질의 저하를 막기 위한 방법으로 Frame Rate Up-Conversion (FRUC)에 대한 다양한 연구가 진행되어 왔다.

전통적인 FRUC 알고리즘으로는 Frame Repetition (FR), Frame Averaging (FA)^[1] 그리고 Linear frame Interpolation (LI)^[2] 등이 있다. 위의 기법들은 움직임 벡터(Motion vector)를 고려하지 않기 때문에 복잡도가 낮고 쉽게 보간이 가능하다. 그러나 영상 포맷이 확장되거나 물체의 움직임이 클 경우에는 영상이 고르지 못하고 영상이 중복되는 고스트 현상(Ghost artifact)이 발생한다. 또한 블록 단위로 보간을 하기 때문에 블록간의 경계부분에서 블록화 현상(Block artifact)이 발생한다^[3].

이러한 문제점을 개선하기 위해 움직임 벡터를 고려한 Motion Compensation FRUC (MC-FRUC) 방법이 연구되었다^{[4][5]}. MC-FRUC의 기본적인 알고리즘인 Motion-Compensated Frame Interpolation (MCFI)는 현재 프레임을 중심으로 이전 프레임에서 획득한 움직임 벡터를 이용하여 보간 프레임을 생성하는 방법이다. MCFI는 움직임이 적은 영상에는 효율적으로 프레임 보간이 가능하지만 움직임이 클 경우에는 정확한 움직임 벡터의 예측이 어렵다. 또한 블록단위로 프레임을 보간 수행하기 때문에 블록과 블록의 경계부분은 고려되지 않는다. 따라서 생성되는 영상에는 블록화 현상이 존재하게 된다. 이러한 문제점을 해결하기 위해 Overlapped Block Motion Compensation (OBMC)가 제안되었다^[6]. OBMC는 MCFI에서 발생하는 블록화 현상을 막기 위해 이웃하는 블록의 경계부분에 따른 움직임 벡터값을 달리함으로써 부드러운 효과를 얻는다. 그러나 경계부분의 과도한 중복으로 인해 화질이 열화되는 문제점이 발생한다.

Thaipanich 등은 양방향 움직임 예측을 이용하여 참조되는 프레임을 달리 적용함으로써 순차적으로 패색 구간을 보간하는 방법인 Robust-FRUC 알고리즘을 제안했다^{[7][8]}. 참조 프레임에 따라 움직임 벡터를 다르게 적용함으로써 보다 정확한 예측이 가능하지만, 필요 이상으로 과도하게 중복되는 구간이 발생하는 문제점이 있다.

본 논문에서는 최적의 벡터 예측을 위해 양방향의 움직임 벡터를 예측하여 선택된 블록을 이용하여 각

각의 방향에 대해 예비 프레임을 생성한다. 생성된 예비 프레임의 중복 여부를 판단하여 보간 프레임을 생성한 후, 존재하는 블록이 가지는 정보를 이용하여 최적의 움직임 벡터를 가지는 블록을 탐색한다. 탐색 방법은 기존의 Sum of Absolute Difference (SAD)를 이용하여 움직임 벡터를 예측하는 방법 보다 정확한 예측을 위해 설정된 bin 내에서의 누적치를 통해 최적의 블록을 선택한다. 또한 이웃한 블록이 가지는 정보를 이용하여 최적의 움직임 벡터를 탐색할 경우 이웃하는 블록이 가지는 정보의 양을 고려하여 선택적으로 참조라인을 적용한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존에 제안된 움직임 벡터를 이용한 프레임 보간 기법을 설명하고, 3장에서는 새로운 알고리즘을 제안한다. 4장에서는 기존의 알고리즘과 제안하는 알고리즘의 성능을 분석한다. 마지막으로 5장에서는 결론을 맺는다.

II. 기존의 FRUC 알고리즘

2.1 MCFI 알고리즘

MCFI는 순방향 움직임 예측을 이용한 기본적인 FRUC 알고리즘으로 그림 1에서 제시된 것과 같이 기준이 되는 현재 프레임 R_{t+1} 과 이전 프레임 R_{t-1} 의 탐색과정을 통해 얻어진 움직임 벡터를 이용하여 프레임 \hat{R}_t 을 보간한다.

$$\hat{R}_t(k, l) = \frac{1}{2}(R_{t-1}(x - v_x, y - v_y) + R_{t+1}(x + v_x, y + v_y)) \quad (1)$$

식(1)과 같이 이전 프레임 R_{t-1} 에서 예측된 벡터를 v 라 할 때 보간 프레임 \hat{R}_t 는 이전 프레임 R_{t-1} 에서 $-v$ 에 위치한 블록과 현재 프레임 R_{t+1} 에서 $+v$ 에 위치

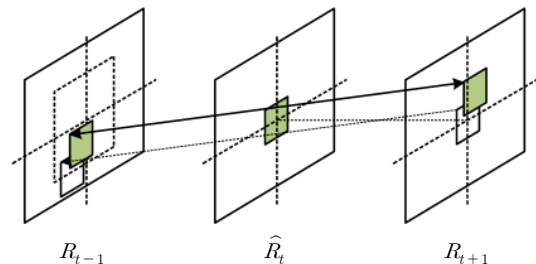


그림 1. MCFI 알고리즘
Fig. 1. Motion Compensation Frame Interpolation (MCFI) algorithm

한 블록의 평균값을 이용하여 블록을 생성한다.

MCFI는 이전 프레임에서 예측된 움직임 벡터를 이용하여 선택된 블록이 현재 프레임에서는 선형적인 움직임을 가정하여 예측하기 때문에 움직임 벡터가 정확하지 않을 경우 현재 프레임에서의 움직임 벡터 또한 잘못된 위치를 선택하게 된다. 따라서 보간되는 프레임은 심각한 블록화 현상이 발생한다.

2.2 OBMC 알고리즘

MCFI에서 발생하는 블록화 현상을 줄이기 위해 블록의 경계부분이 포함하는 이웃하는 모든 움직임 벡터를 고려하여 가중치를 적용하는 OBMC가 제안되었다. 그림 2와 같이 블록의 경계부분 A, B 그리고 C 영역에 대해서 이웃하는 블록이 가지는 움직임 벡터를 고려하여 가중치를 부여한다.

A영역의 경우, 이웃하는 블록이 가지는 움직임 벡터 $v1$, $v2$, $v3$ 그리고 $v4$ 를 모두 고려하여 해당 구간을 생성한다. 즉, A영역에 포함하는 이웃하는 블록이 가지는 모든 움직임 벡터를 고려하여 예측된 블록에 대해 평균값을 취함으로써 블록화 현상을 줄인다. 같은 방법으로 B영역은 움직임 벡터 $v3$ 와 $v4$ 를 적용하고, C영역에 대해서는 움직임 벡터 $v4$ 를 이용하여 해당 블록을 예측하여 C영역에 블록을 생성한다.

$$\hat{f}_t(x, y) = \frac{1}{8} \left(\sum_{l=1}^4 f_{t-1}(x+v_{lx}, y+v_{ly}) + f_{t+1}(x-v_{lx}, y-v_{ly}) \right) \quad (2)$$

$$\hat{f}_t(x, y) = \frac{1}{4} \left(\sum_{l=3}^4 f_{t-1}(x+v_{lx}, y+v_{ly}) + f_{t+1}(x-v_{lx}, y-v_{ly}) \right) \quad (3)$$

$$\hat{f}_t(x, y) = \frac{1}{2} \left(f_{t-1}(x+v_{lx}, y+v_{ly}) + f_{t+1}(x-v_{lx}, y-v_{ly}) \right) \quad (4)$$

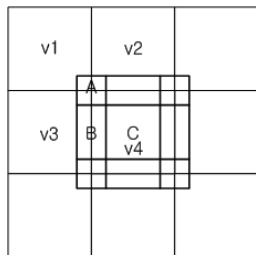


그림 2. OBMC 알고리즘

Fig. 2. Overlapped Block Motion Compensation (OBMC) algorithm

식 (2)-(4)에서 \hat{f}_t 는 보간되는 프레임을 나타내고, f_{t-1} 과 f_{t+1} 는 각각 이전프레임과 현재 프레임을 나타낸다. v_{lx} 와 v_{ly} 는 각 구간이 가지고 있는 (x, y) 에서의 움직임 벡터를 나타낸다.

블록의 경계 부분에 따라 이웃하는 블록이 가지는 움직임 벡터를 고려함으로써 OBMC는 블록화 현상을 감소시킨다. 그러나 이웃하는 블록이 서로 다른 움직임 벡터를 가질 경우 블러링 현상 (Blurring artifact) 또는 과도한 스무딩 현상 (Oversmoothing artifact)이 발생한다.

2.3 Robust-FRUC 알고리즘

Thaipanich 등은 MCFI와 OBMC 알고리즘에서 발생하는 블록화 현상 및 과도한 스무딩 현상의 문제점을 개선하기 위해 양방향 움직임 예측 방법인 Robust-FRUC 알고리즘을 제안했다.

Robust-FRUC는 그림 3과 같이 4단계의 과정을 통해 보간 프레임을 생성한다. 그림 3(a)와 (c)는 현재 프레임에서 예측한 이전프레임에서의 움직임 벡터와 이전프레임에서 예측한 현재프레임에서의 움직임 벡터를 이용하여 설정된 임계값에 만족할 경우 움직임 벡터의 중간에 해당되는 위치에 해당 블록을 생성한다. 그림 3(b)와 (d)는 보다 정확한 예측을 위해 움직임 벡터의 기울기를 추가적으로 고려한다. 움직임 벡

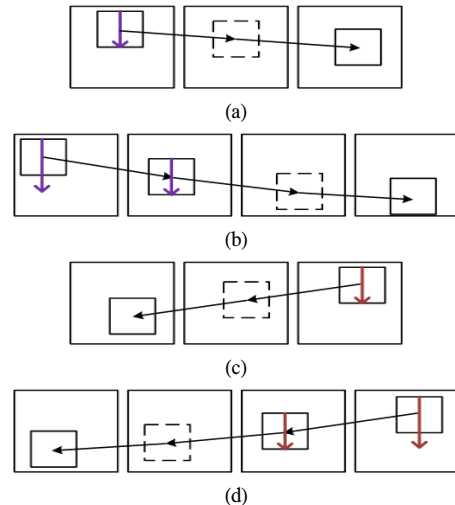


그림 3. Robust-FRUC 알고리즘. (a) 순방향 예측. (b) 순방향 예측(가중치 적용). (c) 역방향 예측. (d) 역방향 예측(가중치 적용)

Fig. 3. Robust Frame Rate Up-Conversion (Robust-FRUC) algorithm using (a) the first-order forward translation, (b) the second-order forward translation, (c) the first-order backward translation, and (d) the second-order backward translation

터의 중간값에 선택된 블록을 생성하기 때문에 위의 과정을 수행하라 경우 블록간의 중복되는 구간이 발생한다. 중복되는 구간에 대해서는 평균값을 취한다. 또한 그림 3(b)와 (d)의 경우에는 시간적으로 예측되는 프레임과의 상관도가 낮기 때문에 높은 임계값의 설정을 통해 정확한 움직임 벡터를 고려한다. Robust-FRUC의 보간을 위한 과정은 식 (5)-(8)과 같다.

$$f_{FP,t+1}^{1st}(P_t(m,n)) = P_{t+1}(m + MVX_t(m,n), n + MVY_t(m,n)) \quad (5)$$

$$f_{FP,t}^{1st}(P_{t+1}(m,n)) = P_t(m + MVX_{t+1}(m,n), n + MVY_{t+1}(m,n)) \quad (6)$$

$$f_{BP,t+1}^{2nd}(P_t(m,n)) = P_{t+1}(m + MVX_t(m,n) - MVX'_t(m,n), n + MVY_t(m,n) - MVY'_t(m,n)) \quad (7)$$

$$f_{BP,t}^{2nd}(P_{t+1}(m,n)) = P_t(m + MVX_{t+1}(m,n) - MVX'_{t+1}(m,n), n + MVY_{t+1}(m,n) - MVY'_{t+1}(m,n)) \quad (8)$$

여기서 $f_{FP,t}^{N^{th}}$ 와 $f_{BP,t}^{N^{th}}$ 는 각각 $N^{th}(N=1,2)$ 에 해당하는 프레임 t 에서 이전 프레임 또는 현재 프레임을 나타낸다. $P_t(m,n)$ 는 프레임 t 에서 (m,n) 위치를 가지는 블록이고, MVX_t , MVY_t 는 움직임 벡터를 나타낸다. 식 (5)-(8)과 같이 보간되는 블록의 위치는 두 영상간의 움직임 벡터의 중간이기 때문에 불규칙하게 폐색 구간(Occlusion regions)이 발생한다. 이러한 폐색 구간에 대해서는 오류 은닉 기법인 Overlapped Boundary Block Matching Algorithm (OBMA)를 이용하여 최종 영상을 생성하게 된다.

양방향 예측을 통해 움직임 벡터 예측의 정확성은 높였지만 그림 3과 같이 모든 경우에서 중복되는 구간이 발생할 경우, 특정 영역에만 블록이 과도하게 집중되는 결과가 발생한다. 또한 OBMA를 이용하여 폐색 구간을 보간 할 경우에는 이웃하는 블록의 움직임 벡터를 모두 고려하여 블록을 재탐색하기 때문에 연산량이 많을 뿐만 아니라 참조되는 정보의 양 또한 적기 때문에 정확한 예측이 어려운 문제점이 있다.

III. 제안하는 알고리즘

본 장에서는 양방향 예측을 이용하여 중간 프레임

의 생성 과정에서 정확한 예측을 위해 MCFFI를 이용하여 신뢰도가 높은 블록을 우선적으로 선택한 후, 생성된 블록이 가지는 정보를 이용하여 최종적으로 블록을 생성하는 방법을 제안한다. 제안하는 방법의 전체 구성은 그림 4와 같다.

제안하는 알고리즘은 크게 두 단계로 나누어진다. 첫 번째 단계에서는 이전 프레임과 현재 프레임이 가지는 벡터를 이용하여 설정된 임계값에 만족하는 블록만을 선택한 예비 프레임을 생성한다. 두 번째 단계에서는 생성된 예비 프레임을 병합한 후, 폐색 구간 보간을 위해 이웃하는 픽셀 정보를 이용하여 재탐색 과정을 통해 프레임을 보간하여 최종 영상을 생성한다.

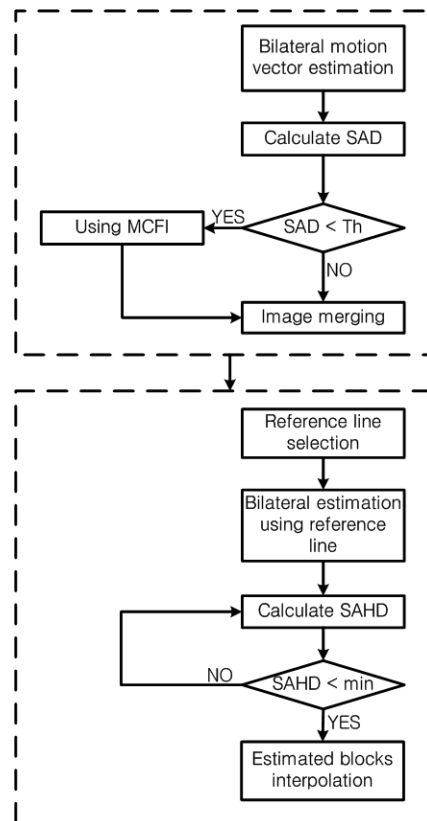


그림 4. 제안하는 적응적 FRUC 기법의 보간 과정
Fig. 4. The block diagram of the proposed adaptive FRUC method

3.1 양방향 움직임 예측을 이용한 예비 프레임 생성

제안하는 알고리즘은 예비 프레임 생성을 위해 양방향 예측 방법을 적용한다. 그림 5와 같이 이전 프레임 및 현재 프레임이 가지는 각각의 움직임 벡터를 고

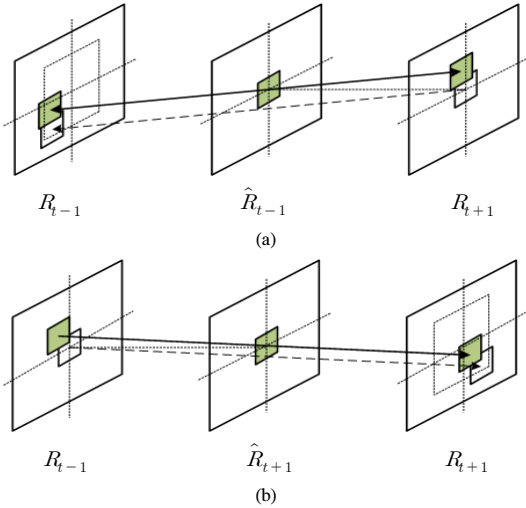


그림 5. 양방향 움직임 예측을 이용한 적응적 MCFFI. (a) 역방향 움직임 벡터 고려. (b) 순방향 움직임 벡터 고려
Fig. 5. Adaptive MCFFI scheme with bilateral motion estimation of (a) the backward and (b) the forward frames

려하여 보다 정확한 움직임을 예측한다.

각각의 프레임에서 가지는 움직임 벡터를 이용하여 \hat{R}_{t-1} 과 \hat{R}_{t+1} 의 예비 프레임을 생성한다. 즉, 식 (9)-(10)과 같이 이전 프레임의 움직임 벡터를 고려할 경우 $mv_{x_{t-1}}$ 과 $mv_{y_{t-1}}$ 를 이용하여 \hat{R}_{t-1} 을 생성하고, 현재 프레임을 이용하여 움직임 벡터를 고려할 경우 $mv_{x_{t+1}}$ 과 $mv_{y_{t+1}}$ 를 이용하여 \hat{R}_{t+1} 을 생성한다.

$$\hat{R}_{t-1}(k, l) = \frac{1}{2} (R_{t-1}(k + mv_{x_{t-1}}, l + mv_{y_{t-1}}) + R_{t+1}(k - mv_{x_{t-1}}, l - mv_{y_{t-1}})) \quad (9)$$

$$\hat{R}_{t+1}(k, l) = \frac{1}{2} (R_{t+1}(k + mv_{x_{t+1}}, l + mv_{y_{t+1}}) + R_{t-1}(k - mv_{x_{t+1}}, l - mv_{y_{t+1}})) \quad (10)$$

양방향 움직임 예측 방법을 이용할 경우에도 에러가 포함된 블록이 선택된다. 따라서 정확히 예측된 블록만을 선택하기 위해 식 (11)과 같이 설정된 임계값에 만족하는 블록만을 선택하는 방법을 제안한다.

$$SAD = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} |R_{t \pm 1}(k, l) - R_{t \pm 1}(k + mv_x, l + mv_y)| \quad (11)$$

$$= \begin{cases} MCFFI & \text{if } SAD \leq Th \\ 0 & \text{else } SAD > Th \end{cases}$$

예비 프레임 생성과정에서 SAD를 이용하여 설정된 임계값(Th)보다 작은 경우 MCFFI를 이용하여 블록을 생성한다. 만약 설정된 Th 값보다 블록이 가지는 최소 SAD값이 큰 경우에는 예측된 움직임 벡터의 위치가 오류라고 판단하여 패색 구간으로 설정한다.

3.2 임계값(Th) 설정

정확히 예측된 블록만을 선택하기 위해 임계값(Th)을 설정하여 예비 프레임을 생성한다. 참조되는 이전 프레임과 현재 프레임은 시간적으로 인접하기 때문에 두 프레임간의 상관도는 높다. 따라서 임계값을 설정하기 위해 상관도를 고려할 수 있는 공분산(Covariance)을 적응적으로 이용한다. 만약 고정된 임계값을 사용할 경우에는 움직임 큰 블록은 최적의 움직임 벡터를 획득하더라도 움직임이 적은 블록 보다 상대적으로 최소 SAD값이 크다. 결국 정확히 예측된 블록까지도 패색 구간으로 선택하는 문제점이 발생한다.

그림 6은 고정된 임계값과 적응적으로 각 프레임에 적용한 임계값의 PSNR 결과를 나타낸다. 고정된 임계값을 사용한 경우, 각 실험 영상에 100부터 6000까지 10단위로 증가시키면서 가장 좋은 결과를 얻는 임계값의 PSNR을 나타내었다. 적응적으로 적용된 임계값은 실험 영상의 다른 최적의 고정된 임계값이 가지는 가장 높은 PSNR을 따르는 것을 확인할 수 있다. 각각의 영상이 갖는 최적의 임계값이 다르기 때문에 고정적인 값을 이용하여 패색구간을 정하기는 어렵다. 반면에 적응적 임계값을 사용할 경우에는 각각의 참조되는 프레임의 상관도를 고려하여 해당 프레임의 최적이 되는 임계값을 설정하기 때문에 적절한 임계값의 설정이 가능하다. 공분산을 이용하여 설정된 임계값은 식 (12)와 같다.

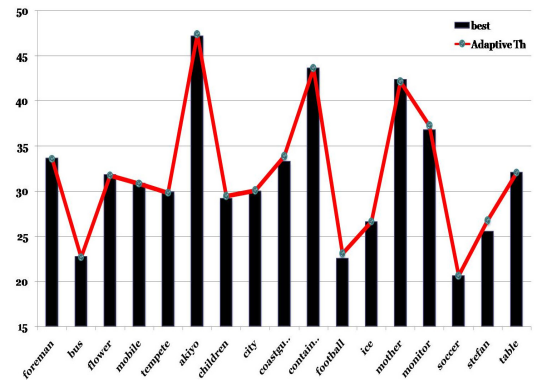


그림 6. 고정적 임계값과 적응적 임계값의 PSNR 성능 비교
Fig. 6. PSNR comparison of the adaptive and steady Th

$$Th_{R_{t \pm 1}, R_{t \pm 1}} = \frac{cov(R_{t \pm 1}, R_{t \pm 1})}{\gamma}, \quad \gamma = 4, \quad N = 8$$

$$cov(R_{t \pm 1}, R_{t \pm 1}) = \frac{1}{N} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} (R_{t \pm 1}(k, l) - \sigma_R)(R_{t \pm 1}(k, l) - \sigma_R) \quad (12)$$

$$\sigma_R = \sqrt{\sigma_R^2}$$

$$\sigma_R^2 = \frac{N \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} R(k, l)^2 - (\sum_{k=0}^{N-1} \sum_{l=0}^{N-1} R(k, l))^2}{N(N-1)}$$

여기서 $Th_{R_{t \pm 1}, R_{t \pm 1}}$ 는 각각 이전 프레임과 현재 프레임에 적용되는 임계값이고, cov 는 공분산을 나타낸다. γ 는 선택되는 블록이 여러를 포함하는 최적의 임계값을 설정하기 위해 실험적으로 얻어진 값이다.

3.3 예비 프레임 병합

양방향 예측을 이용하여 생성된 영상은 그림 7과 같다. 그림 7(a)는 이전 프레임에서의 움직임 벡터를 이용하여 생성된 영상 \hat{R}_{t-1} 이고, (b)는 현재 프레임에서의 움직임 벡터를 고려한 영상 \hat{R}_{t+1} 이다. 그림에서와 같이 폐색 구간을 제외한 생성된 블록은 보다 정확하게 예측되었음을 알 수 있다.

생성된 예비 프레임 \hat{R}_{t-1} 과 \hat{R}_{t+1} 를 이용하여 그림 7(c)와 같이 새로운 프레임 \hat{R}_t 를 생성한다. 각각의 예비 프레임은 자신의 프레임이 참조 프레임이 되기 때문에 시간적으로 인접하는 프레임을 이용하는 것보다 두 영상의 상관도가 높다. 따라서 \hat{R}_t 를 생성하기 위해서 그림 8에서와 같이 동일한 위치의 블록이

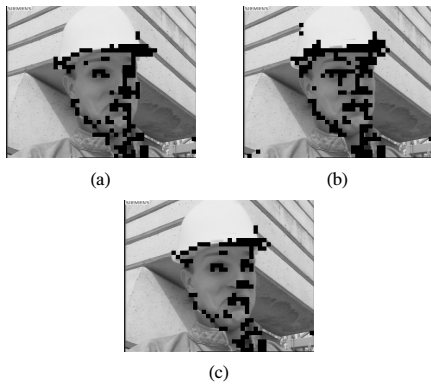


그림 7. 보간된 영상. (a) 예비 프레임 \hat{R}_{t-1} . (b) 예비 프레임 \hat{R}_{t+1} . (c) 병합된 프레임 \hat{R}_t
Fig. 7. Interpolated preliminary frames of (a) \hat{R}_{t-1} and (b) \hat{R}_{t+1} , and (c) merged image \hat{R}_t

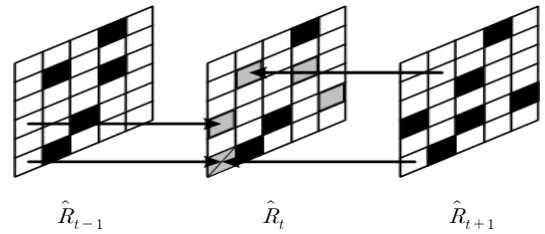


그림 8. 예비 프레임 병합
Fig. 8. Merging of preliminary interpolated frames of \hat{R}_{t-1} and \hat{R}_{t+1}

두 프레임 중 한 곳에만 존재할 경우 해당 블록을 선택한다. 만약 두 프레임 모두 블록이 존재하지 않을 경우에는 해당 구간은 폐색 구간으로 설정한다. 그리고 두 블록이 모두 존재 할 경우 참조되는 블록의 평균값을 취한다.

3.4 이웃하는 픽셀 정보를 이용한 적응적 폐색 구간 보간

그림 7(c)와 같이 각각의 예비 프레임을 이용하여 생성된 \hat{R}_t 에는 두 프레임을 병합하는 과정에서 프레임에 영상이 존재하는 구간은 보간이 가능했지만, 나머지 구간에 대해서는 여전히 폐색 구간이 존재한다. 폐색 구간의 최종적인 보간을 위해 이웃하는 픽셀의 정보를 이용한다. 폐색 구간을 중심으로 이웃하는 픽셀을 참조가 가능한 경우는 그림 9와 같다. 그림 9(a)는 모든 블록이 이용 가능하지만, (b)-(e)의 경우에는 이웃하는 블록에 폐색 구간이 존재하기 때문에 참조 가능한 픽셀만을 고려한다. 이웃하는 블록에 폐색 구간이 많이 포함되어 있을 경우 이용할 수 있는 정보가 적기 때문에 이를 보완하기 위해 폐색 구간 블록이 증가함에 따라 참조라인을 추가한다.

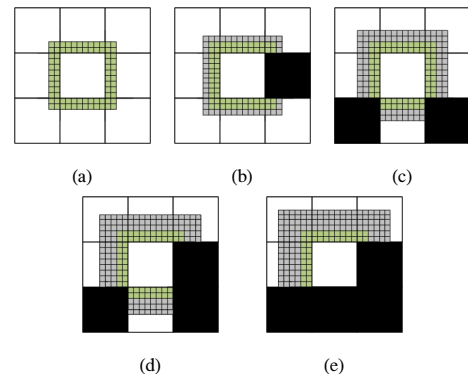


그림 9. 이웃하는 픽셀정보에 따른 적응적 참조라인 선택.
(a) 모든 픽셀 정보 이용 가능. (b)-(e) 폐색 구간 존재
Fig. 9. Applied neighbouring reference lines according to the non-existent (a) or existent (b)-(e) occlusion areas

참조 라인을 기반으로 이전 프레임과 현재 프레임에서 최종적 패색 구간 보간을 위해 재탐색 과정을 수행한다. 정확한 블록 예측을 위해 기존에 탐색 과정에서 사용되었던 SAD를 사용하지 않고, 식 (13)에서와 같이 각 픽셀값에 대해 bin을 설정한 후 해당 구간내의 누적치를 계산하여 최적의 블록을 선정하는 SAHD (Sum of Absolute Histogram Difference)를 이용한다.

$$SAHD = \sum_{i=0}^B |h_{\hat{R}_t}^s[i] - h_{R_{t+1}}^{s+mv}[i]|, \quad s \in B_{k,l} \quad (13)$$

여기서 B 는 설정된 bin을 나타내고, $h_{\hat{R}_t}^s$ 와 $h_{R_{t+1}}^{s+mv}$ 는 각각 이전 프레임과 현재 프레임의 bin 내에 포함되는 누적치를 나타낸다. 탐색 영역 내에서 최소 SAHD를 가지는 블록을 탐색하여 식 (14)와 같이 \hat{R}_t 의 패색 구간을 보간한다.

여기서 $inter_mv_x_{t+1}$ 와 $inter_mv_y_{t+1}$ 는 각각 이전 프레임과 현재 프레임에서의 이웃하는 블록 정보를 이용하여 재탐색 과정을 통해 획득한 움직임 벡터를 나타낸다.

그림 10과 같이 움직임 벡터 $inter_mv_x_{t+1}$ 와 $inter_mv_y_{t+1}$ 는 참조 프레임에 대해 각각 독립적으로 적용하여 선택된 블록에 대해 평균값을 취함으

로써 최종적인 보간 영상을 생성한다.

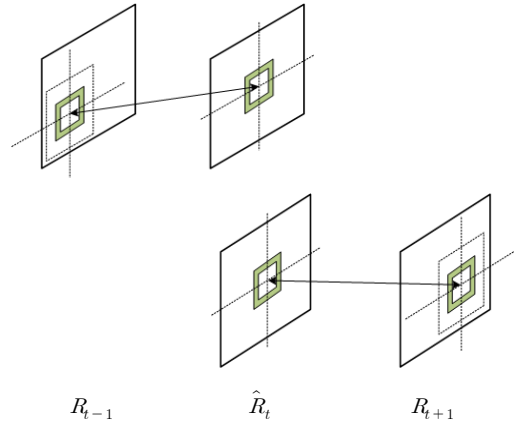


그림 10. 참조라인을 이용한 보간 방법
Fig. 10. Interpolation method using reference lines

IV. 실험 결과 및 고찰

제안된 알고리즘의 타당성을 입증하기 위해 기존에 제안되었던 LI, MCFI, OBMC, 그리고 Robust-FRUC와 성능을 비교 분석하였다. 실험 영상은 모두 CIF(352×288) 크기이며, 예측되는 블록의 크기는 8×8로 고정하였고, 초기 MCFI 선택을 위한 단계에서의 탐색 영역은 16×16으로 1/4픽셀까지 탐색한다. 이웃



그림 11. Foreman (CIF) 영상의 주관적 화질 비교. (a) LI. (b) MCFI. (c) OBMC. (d) Robust-FRUC. (e) 제안한 알고리즘. (f) 원본 영상

Fig. 11. Visual comparison of part of the original frame and the interpolated frames of the 132th frame within Foreman (CIF). (a) LI, (b) MCFI, (c) OBMC, (d) Robust-FRUC, (e) proposed algorithm and (f) original frame

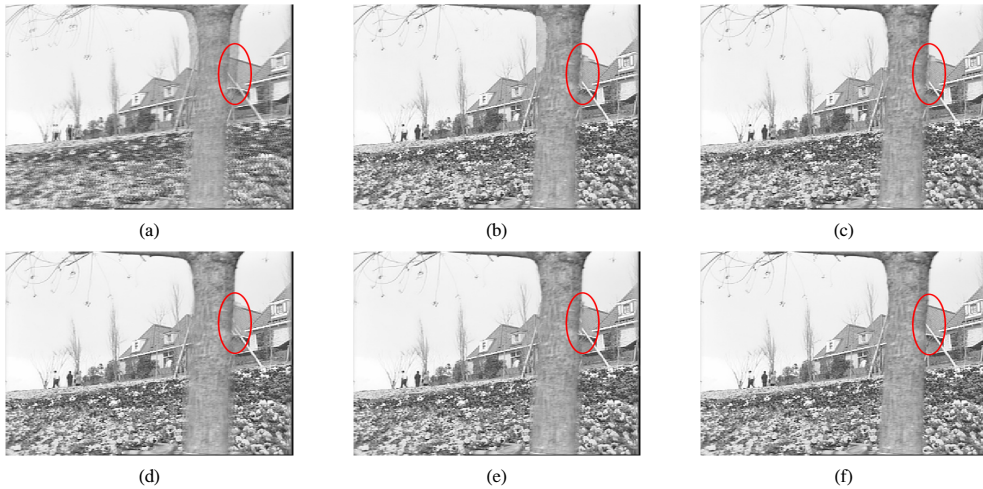


그림 12. *Flower* (CIF) 영상의 주관적 화질 비교 (a) LI. (b) MCFI. (c) OBMC. (d) Robust-FRUC. (e) 제안한 알고리즘. (f) 원본 영상
Fig. 12. Visual comparison of part of the original frame and the interpolated frames of the 204th frame within *Flower* (CIF), (a) LI, (b) MCFI, (c) OBMC, (d) Robust-FRUC, (e) proposed algorithm and (f) original frame

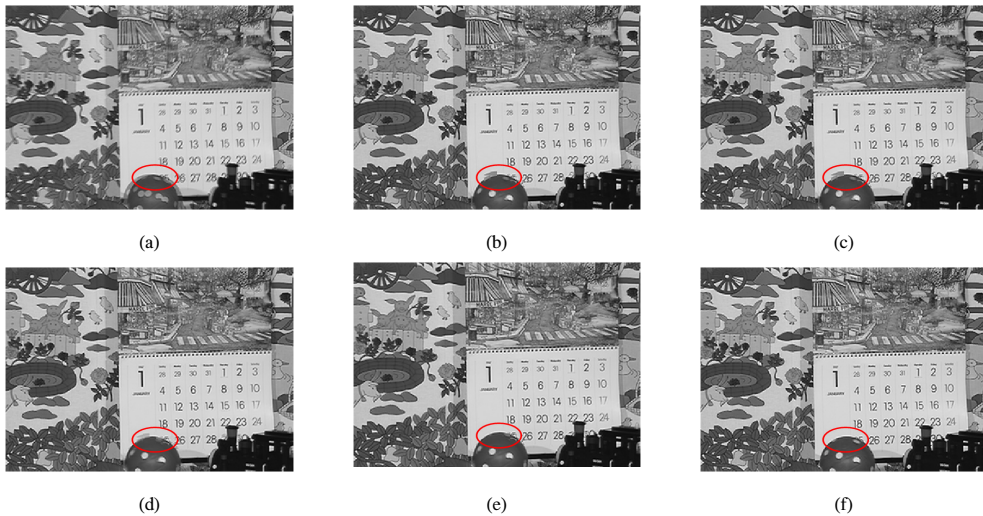


그림 13. *Mobile* (CIF) 영상의 주관적 화질 비교 (a) LI. (b) MCFI. (c) OBMC. (d) Robust-FRUC. (e) 제안한 알고리즘. (f) 원본 영상
Fig. 13. Visual comparison of part of the original frame and the interpolated frames of the 12th frame within *Mobile* (CIF), (a) LI, (b) MCFI, (c) OBMC, (d) Robust-FRUC, (e) proposed algorithm and (f) original frame

하는 블록의 정보를 이용하여 움직임 벡터를 예측하는 단계에서의 탐색 영역은 이웃하는 블록의 움직임 벡터의 중간값을 고려하기 때문에 2×2 로 설정한다. 실험 결과에서는 객관적인 성능 분석을 위해 각 영상의 PSNR 및 주관적 화질 또한 비교하였다.

그림 11-그림 13은 제안하는 알고리즘과 기존 알고리즘의 비교 영상을 나타낸다. 그림 11의 *Foreman* 영상의 경우 제안하는 알고리즘이 모자의 경계영역 부분에서 기존의 알고리즘 보다 정확하게 보간된다. 또

한 배경부분과 사람과의 구분이 명확해 지는 것을 확인할 수 있다. 그림 11(d)의 Robust_FRUC는 그림 11(a)의 MCFI에서 발생하는 예측 오류를 포함하는 블록의 개선된 결과를 나타내지만 여전히 잔여 에러 블록이 존재한다. 그러나 그림 11(e)의 제안하는 알고리즘의 영상은 잔여 에러 블록까지 제거하여 주관적으로 높은 성능을 보임을 알 수 있다. 그림 12의 *Flower* 영상에서도 그림 12(b), (c) 그리고 (d) 영상보다 (e)의 영상에서 배경이 되는 집과 나무 영역이

표 1. PSNR값 (dB)를 통한 성능 비교
Table 1. PSNR performance comparison

Video sequences	PSNR				
	LI	MCFI	OBMC	Robust-FRUC	Proposed
<i>Foreman</i>	29.3092	32.7338	33.1404	33.4895	34.4829
<i>Bus</i>	18.9636	21.6715	21.9122	22.6575	22.8798
<i>Flower</i>	19.8022	31.2147	31.2147	31.8080	32.6242
<i>Mobile</i>	24.6965	29.2996	29.2996	30.8095	31.8043
<i>Children</i>	26.2263	28.9145	28.9650	29.6032	29.8559
<i>City</i>	24.0753	29.5009	29.8009	30.1538	30.7650
<i>Container</i>	41.8994	43.3318	43.3807	43.4022	43.6108
<i>Football</i>	20.5605	22.6757	22.8751	23.4082	23.9764
<i>Monitor</i>	36.3804	36.8180	36.9649	37.3367	37.6899
<i>Soccer</i>	19.5316	20.2045	20.5903	20.9306	21.6223
Average	27.8689	31.2220	31.4616	31.9043	32.2685

열화되지 않고 보간되는 화질의 개선을 보인다. 이는 양방향 예측과 함께 신뢰도 높은 정보를 이용하여 추가적으로 채탐색을 통해 최적의 블록을 예측하기 때문이다. 마찬가지로 그림 13의 *Mobile* 영상에서 그림 13(b), (c) 그리고 (d)에서 제거하지 못한 공의 여러 부분을 (e)의 제안된 알고리즘을 수행한 영상에서는 정확하게 예측하고 있는 것을 확인할 수 있다.

표 1은 제안하는 알고리즘의 객관적 성능을 입증하기 위해 기존에 제안된 알고리즘과의 PSNR 결과를 비교하였다. 제안하는 알고리즘의 PSNR이 기존의 알고리즘 보다 성능이 향상된 것을 표 1을 통해 확인할 수 있다. FA의 경우, 움직임 벡터를 고려하지 않기 때문에 움직임 벡터를 고려하여 영상을 보간하는 다른 알고리즘에 비해 PSNR 성능이 현저히 떨어진다. OBMC는 기존의 MCFI의 블록화 현상을 개선하기 위한 방법으로 PSNR 측면에서는 향상된 결과를 보이지만 위에서 언급했듯이 과도한 블록의 중복으로 영상의 열화 현상이 발생한다. Robust-FRUC 역시 PSNR의 성능은 향상되었지만 양방향의 움직임 정보를 이용하는 연산량의 증가에 비해 크게 향상된 결과를 얻을 수 없다. 이에 반해 제안하는 알고리즘은 기존의 알고리즘보다 향상된 결과를 보여준다. *Foreman* 과 *Mobile* 영상의 경우는 MCFI 보다는 각각 1.4dB, 2.5dB 향상되었고, Robust-FRUC 보다는 0.99dB 높은 결과를 얻었다. 또한 Robust-FRUC와 비교하여 나머지 영상들 역시 평균적으로 0.6dB 증가했음을 알 수 있다.

V. 결 론

본 논문에서는 새로운 프레임을 생성하기 위해 이

전 영상과 현재 영상의 양방향 예측을 이용한 프레임 보간 알고리즘을 제안하였다. 제안된 알고리즘은 단순히 양방향의 움직임 정보에만 의존하지 않고 신뢰도 높은 정보를 기반으로 하여 추가적인 채탐색 과정을 통해 폐색구간을 보간한다. 실험 결과에서 나타났듯이 PSNR 성능이 기존의 알고리즘 보다 제안된 알고리즘이 좋은 성능을 보인다. 또한 기존에 제거하지 못한 여러 부분에 대해서는 제안된 알고리즘에서 원본 영상과 유사하게 보간이 가능했음을 주관적 화질의 비교를 통해 확인하였다. 그러나 제안하는 알고리즘은 초기 예비 프레임 생성과정에서 MCFI에 많은 의존을 하기 때문에 MCFI에서 정확하지 않은 움직임 정보를 가질 경우, 연쇄적인 에러 발생의 문제점이 있다. 따라서 초기 예비 프레임 생성과정에 대해 MCFI 보다 신뢰도가 높은 새로운 방법을 고안할 필요성이 요구된다.

참 고 문 헌

- [1] G. Dane and T. Q. Nguyen, "Optimal temporal interpolation filter for motion-compensated frame rate upconversion," *IEEE Trans. Image Processing*, Vol.15, No.4, pp.978-991, Apr. 2006.
- [2] K. A. Bugwadia, E. D. Petajan, and N. N. Puri, "Progressive-scan rate up-conversion of 24/30 source materials for HDTV," *IEEE Trans. Consumer Electronics*, Vol.42, No.3, pp. 312-321, Aug. 1996.
- [3] Jiefu Z., Keman Y., Jiang L., and Shi-peng L.,

"A low complexity motion compensated frame interpolation method," in *proc. Int. Symposium On Circuits and Systems*, May 2005.

- [4] K. Hilman, H. W. Park, and Y. Kim, "Using motion-compensated frame rate conversion for the correction of 3:2 pull down artifacts in video sequences," *IEEE Trans. Circuits Systems for Video Technology*, Vol.10, No.6, pp.867-877, Sep. 2000.
- [5] R. Castagno, P. Haavisto, and G. Ramponi, "A method for motion adaptive frame rate upconversion," *IEEE Trans. Circuits Systems for Video Technology*, Vol.6, No.5, pp. 436-446, Oct. 1996.
- [6] B. Girod, "Efficiency analysis of multihypothesis motion-compensated prediction for video coding," *IEEE Trans. Image Processing*, Vol.9, No.2, pp.173-183, Feb. 2000.
- [7] T. Thaipanich, P.-H. Wu, and C.-C. J. Kuo, "Robust Video Frame Rate Up-Conversion (FRUC) Techniques," in *proc. Int. Conference on Consumer Electronics*, Jan. 2008.
- [8] T. Thaipanich, P.-H. Wu, and C.-C. J. Kuo, "Low Complexity Algorithm for Robust Video Frame Rate Up-Conversion (FRUC) Technique," *IEEE Trans. Consumer Electronics*, Vol.55, No.1, pp.220-228, Feb. 2009.

오 형 철 (Hyeongchul Oh)

정회원



2009년 2월 강남대학교 전자공학과 졸업
2009년 3월~현재 : 한양대학교 전자컴퓨터통신공학과 석사과정
<관심분야> 영상처리, 영상압축, Frame Interpolation

이 주 현 (Joohyun Lee)

정회원



2004년 2월 한양대학교 전자컴퓨터공학부
2006년 2월 한양대학교 전자통신컴퓨터공학과 석사
2006년 3월~현재 한양대학교 전자통신컴퓨터공학과 박사과정
<관심분야> 영상처리, 영상압축, H.264 Video Coding, Scalable Video Coding, HDR

민 창 기 (Changki Min)

정회원



2009년 2월 세종대학교 전자공학과
2009년 2월~현재 한양대학교 전자컴퓨터통신공학과 석사과정
<관심분야> 영상처리, 영상압축, 에러 은닉

정 제 창 (Jechang Jeong)

정회원



1980년 2월 서울대학교 전자공학과 졸업
1982년 2월 KAIST 전기전자공학과 석사
1990년 미국 미시간대학 전기공학과 공학박사
1980~1986 KBS 기술연구소 연구원(디지털 TV 및 뉴미디어 연구)
1990~1991 미국 미시간대학 전기공학과 연구교수 (영상 및 신호처리 연구)
1991~1995 삼성전자 멀티미디어 연구소 (MPEG, HDTV, 멀티미디어 연구)
1995~현재 한양대학교 전자통신컴퓨터공학과 교수 (영상통신 및 신호처리 연구실)
1998년 11월 27일 과학기술자상 수상
1998년 12월 21일 정보통신부장관상 표창
<관심분야> 영상처리 및 영상압축