

논문 2009-46SP-3-4

계층적 적응적 탐색과 양방향 움직임 예측을 이용한 프레임율 증가 방법

(Frame-rate Up-conversion using Hierarchical Adaptive Search and
Bi-directional Motion Estimation)

민 경 연*, 박 시 내**, 심 동 규**

(Kyungyeon Min, Seanae Park, and Donggyu Sim)

요 약

본 논문은 비디오의 시간적 화질 향상을 위한 새로운 프레임율 증가 방법을 제안한다. 제안하는 방법에서는 계층적 움직임 추정 시에 탐색범위를 적응적으로 변환하는 방법을 이용하며, 움직임 보상 시 보간되지 않은 부분에 한하여 양방향 움직임 추정 및 보상과 선형 보간법을 수행한다. 부정확한 움직임 벡터 추정으로 인한 오류를 방지하기 위하여 신뢰도를 기반으로 탐색 범위를 적응적으로 조절하며, 움직임 추정에 대한 신뢰도를 높이기 위하여 분산이 높은 블록 순으로 움직임 추정을 수행한다. 또한, 보간되지 않은 영역에서 배경과 객체를 분리한 후 배경인 영역에서는 선형보간법을 수행하고, 객체로 추정된 영역에서는 양방향 움직임 추정 방법을 이용하여 보간한다. 알고리즘의 성능을 평가하기 위하여 원본 프레임과 제안한 알고리즘을 이용하여 보간한 프레임 사이의 PSNR을 측정하였다. 그 결과, 화질이 기존 알고리즘보다 약 2dB 정도 개선되었으며, 블록화 현상과 뭉뚱화 현상이 감소한 것을 확인할 수 있었다.

Abstract

In this paper, we propose a frame-rate up-conversion method for temporal quality enhancement. The proposed method adaptively changes search range during hierarchical motion estimation and reconstructs hole regions using the proposed bi-direction prediction and linear interpolation. In order to alleviate errors due to inaccurate motion vector estimation, search range is adaptively changed based on reliability and for more accurate, motion estimation is performed in descending order of block variance. After segmentation of background and object regions, for filling hole regions, the pixel values of background regions are reconstructed using linear interpolation and those of object regions are compensated based on the proposed bi-directional prediction. The proposed algorithm is evaluated in terms of PSNR with original uncompressed sequences. Experimental results show that the proposed algorithm is better than conventional methods by around 2 dB, and blocky artifacts and blur artifacts are significantly diminished.

Keywords : Hierarchical motion estimation, adaptive search range, bi-direction, frame-rate up-conversion

I. 서 론

최근 디스플레이 장치의 생산 기술이 급격히 발전함에 따라 CIF급 디스플레이 장치인 PDA부터 HD급 텔레비

전까지 다양한 크기의 디스플레이가 사용되고 있다. 이 중에서 고해상도 비디오 서비스의 경우, 주관적 화질 및 객관적 화질의 향상을 위하여 높은 비트율을 필요로 하게 된다. 하지만, 부호화 된 비디오의 비트율을 전송 네트워크 채널이 허용하는 범위 내로 적응적으로 맞춰 주어야 하기 때문에, 필요한 만큼의 비트율을 증가시킬 수 없다^[1~4]. 또한, 최근 수요가 급증하고 있는 대형 평면 텔레비전의 주요 패널인 PDP (Plasma display panel)의 경우, 패널을 구성하는 화소들이 기존 방송 표준에서의 프레임

* 학생회원, ** 정회원, 광운대학교 컴퓨터공학과
(Dept. Computer Eng., Kwangwoon University)

※ 이 논문은 2006년 정부(교육인적자원부)의 재원으로
한국학술진흥재단의 지원을 받아 수행된 연구임.
(KRF-2006-331-D00547)

접수일자: 2008년9월25일, 수정완료일: 2009년4월14일

간격인 1/60초 동안 일정량 정도 이상의 에너지를 유지할 수 없다. 따라서 형광등이 깜빡거리는 것과 같은 효과가 발생하여, 텔레비전 시청 시 눈의 피로도가 빠르게 증가하는 문제가 발생한다.

위와 같은 문제점들을 해결하기 위하여 프레임율 증가 기법에 관한 연구가 활발하게 진행되고 있다. 프레임율 증가 알고리즘이란 주어진 입력 영상의 프레임율을 디스플레이에 적당한 비율로 증가시키는 것을 말한다^[2]. 즉, 프레임율 증가는 송신측에서 프레임율을 증가시키기 위한 어떠한 정보도 수신측으로 보내지 않고, 수신측에서 후처리 과정을 통해 낮은 프레임 수를 높은 프레임 수로 변환하는 방법이다. 따라서 이러한 프레임율 변환 기법은 다양한 형식의 고화질 멀티미디어 디스플레이 장치로 화질의 저하 없이 영상을 효과적으로 전송할 수 있도록 한다. 또한 송신측에서 보내는 동일한 영상 정보를 이용하여 수신측에서 프레임 수를 증가시키면 영상의 재생이 자연스러워지므로 사용자가 느끼는 주관적 화질을 향상시켜 눈의 피로를 줄일 수 있다.

프레임율 증가 기법은 크게 화소 기반 알고리즘과 블록 기반 알고리즘으로 나뉘질 수 있다^[3~6]. 우선, 화소 기반으로 하는 알고리즘은 프레임 반복, 선형 보간법으로 나뉘질 수 있다. 프레임 반복, 선형 보간법과 같은 프레임율 변환 알고리즘은 계산량이 적으며 복잡도가 낮다. 특히 선형 보간법은 단순한 영상에서는 효과적인 성능도 기대할 수 있다. 하지만 물체의 경계에서 몽롱화 현상을 발생시키거나 객체의 부자연스러운 움직임과 같은 문제를 일으킨다.

다음으로 블록 기반을 이용한 알고리즘은 움직임 벡터를 이용하며, 성능 향상을 위하여 다양한 기법을 적용한다^[7~14]. 잘못된 움직임 벡터를 수정하기 위한 미디언 필터를 적용하는 방법, 움직임 벡터가 0일 때 가중치를 두는 방법 등이 있으며, 블록화 현상을 방지하기 위한 중첩 블록 움직임 추정 등을 활용한다. 블록 기반을 이용한 알고리즘은 프레임을 분할하여 움직임 추정을 수행하기 때문에 영상 내에서 객체의 움직임을 고려하는 장점을 가진 반면에, 블록 경계에서 블록 현상과 잘못된 움직임 벡터로 인한 오류를 발생시키는 단점이 있다.

본 논문에서는 화소 기반 알고리즘과 블록 기반 알고리즘을 모두 이용하여 중간 프레임율 생성함으로써 프레임율을 증가시킨다. 우선 블록 기반 알고리즘을 이용하여 전체적인 프레임을 보간하고, 보간되지 않은 영역에서 화소 기반 알고리즘과 블록 기반 알고리즘을 이용

하여 보상을 수행한다. 블록 기반의 움직임 추정을 수행할 때, 움직임 추정의 신뢰도에 따라 탐색 영역을 적응적으로 변환하는 기법을 이용한다. 여기서 신뢰도는 움직임 추정 시 실제 움직임을 표현하는 움직임 벡터를 추정할 확률을 의미하며, 분산이 높은 블록일수록 정확한 움직임 추정이 이루어질 확률이 높다는 것을 가정하여 수행한다. 보간되지 않은 영역에서는 화소값 기반으로 배경과 객체를 분리한 후 배경으로 추정된 부분은 선형보간법을 이용하고, 객체로 추정된 영역에서는 블록 기반의 양방향 움직임 추정을 수행한다.

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

II. 적응적 탐색 기법을 이용한 프레임율 증가 방법

기존의 프레임율 증가 알고리즘은 부정확한 움직임 벡터에 의하여 블록의 겹침 현상이나 블록화 현상이 발생하여, 주관적 화질 저하를 발생시킨다. 본 논문은 탐색 영역을 적응적으로 하여 움직임 추정을 수행하여 잘못된 움직임 추정을 방지하고, 보간되지 않은 영역에 한하여 양방향 움직임 추정 방법을 이용하여 프레임율 증가 기법을 제안한다.

그림 1은 제안하는 알고리즘에 대한 전체 시스템 순서도이다. 순서도에서 연속되는 두 개의 프레임은 $t-1$, $t+1$ 프레임을 나타내고, t 프레임은 보간된 프레임을 나타낸다. 제안하는 방법은 우선, 계층적 움직임 추정을 수행하기 위하여 기존 프레임을 복수 개의 계층으로 분리하여 각각의 계층 중 저해상도 계층부터 중첩블록 움직임 추정을 수행한다^[7]. 이때, 잘못된 움직임 벡터의 발생 빈도를 감소시키기 위하여, 분산이 큰 블록 순으로 탐색 영역을 수정하면서 움직임 추정을 수행하는 방법을 적응적으로 수행한다. 측정된 움직임 벡터를 이용하여 원본 프레임의 해상도를 가진 계층에서 움직임 보상을 수행하며, 블록의 SAD가 임계치보다 작을 경우에만 움직임 보상을 수행한다. 움직임 보상을 수행한 후의 t 프레임에서, 보간되지 않는 부분에 관하여 화소값 기반으로 배경과 객체를 분리한다. 배경으로 추정된 영역에 대해서는 선형 보간법을 이용하여 화소 값을 보간하며, 객체로 추정된 영

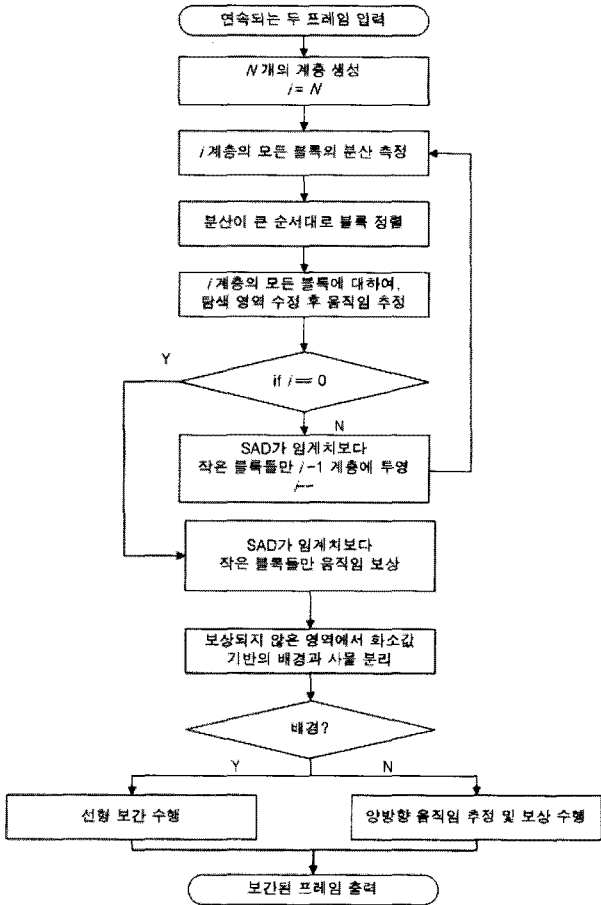


그림 1. 제안한 알고리즘의 흐름도
Fig. 1. Flowchart of proposal algorithm.

역에 한해서는 양방향 움직임 추정을 이용한 움직임 보상을 수행한다.

1. 계층적 움직임 추정을 위한 계층 생성

제안하는 방법에서는 계산량을 줄이고 정확한 움직임 벡터를 구하기 위하여 계층적 움직임 추정을 수행한다. 계층적 움직임 추정을 하기 위하여 원 프레임을 다수의 계층으로 분리한다.

일반적으로 원본 크기의 1/N 크기의 계층을 생성하기 위하여 다운샘플링을 수행한다. 하지만, 단순히 서브샘플링을 이용하여 계층을 만들면 엘리어싱이 발생하게 된다. 따라서 제안하는 알고리즘에서는 네 개의 화소 밝기 값에 저대역 통과 필터를 취한 후 다운샘플링을 수행하여 이 문제를 해결한다.

계층적 움직임 추정을 위해 다수의 계층을 생성하기 위하여 t-1 프레임과 t+1 프레임 각각에서 다음 식을 이용한다.

$$layer_N(j, i) = \frac{1}{4}(layer_{N-1}(2j, 2i) + layer_{N-1}(2j, 2i+1) + layer_{N-1}(2j+1, 2i) + layer_{N-1}(2j+1, 2i+1)) \quad (1)$$

여기서, j, i 는 화소 인덱스이며 N 에 계층을 구분하기 위한 인덱스이다.

2. 블록 단위 분산 측정

제안하는 방법은 계층적 움직임 추정을 위하여 계층을 N개로 분리한다. 이렇게 분리된 계층에서, 각 블록의 신뢰도를 측정하기 위하여 블록 단위로 분산을 측정한다.

분산이 높은 블록은 움직임 추정 시, 실제 움직임을 나타내는 움직임 벡터를 추정할 확률이 높다. 본 논문에서는 정확한 움직임 벡터가 추정될 확률을 블록의 신뢰도라 하며, 움직임 추정이 정확하게 수행될 블록을 신뢰도가 높은 블록으로 표현한다.

본 논문은 t+1 프레임을 기준으로 t-1 프레임에 대해 움직임 추정을 수행함으로, t+1 프레임의 블록의 분산을 측정한다. t+1 프레임의 (j, i) 위치에 있는 블록의 분산은 다음과 같이 측정한다.

$$\sigma^2(t+1, x, y) = \frac{1}{NM} \sum_{j=0}^M \sum_{i=0}^N |S(t+1, x, y) - f(t+1, x+j, y+i)|^2 \quad (2)$$

여기서, x, y 는 블록 인덱스이며, N과 M 는 블록의 수평, 수직 크기를 나타내며 $\sigma^2(x, y)$ 는 (x, y) 위치의 블록의 화소값의 분산, S(x, y) 는 (x, y) 위치의 블록의 평균, f(j, i) 는 (j, i) 위치의 화소값을 나타낸다.

3. 적응적 탐색 범위 조정

본 논문에서는 정확한 움직임 벡터를 추정하기 위하여 탐색 영역을 조정하는 방법을 이용한다. 잘못된 움직임 추정에 의하여 블록의 일부가 겹쳐지게 되면, 정확한 움직임 벡터에 의하여 보간된 블록까지 오류가 나타나게 된다. 따라서 이런 문제점을 해결하기 위하여 본 논문에서는 탐색 범위를 적응적으로 설정함으로써 잘못된 움직임 벡터의 발생으로 인한 겹쳐짐을 방지한다.

그림 2는 움직임 추정 시 탐색영역을 적응적으로 조절하는 방법을 일차원적으로 표현한 것이다. t+1 프레임의 A 블록을 기준으로 t-1 프레임에 대해서 탐색을 수행할 때, t-1 프레임에서 수정하기 전의 탐색 범위를 D라 하면 A 블록은 t 프레임의 B+C 구간 내에서 보간된다. A 블록을 보간하기 이전에 이미 A 블록보다 분산이 높은 F 블

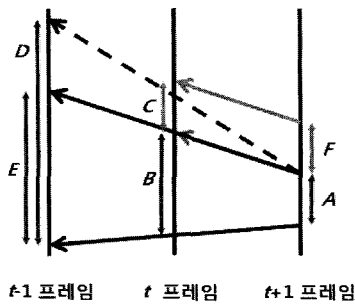


그림 2. 검색 범위 결정
Fig. 2. Update search range.

록이 C 구간에 보간 되어져 있다고 가정하면, A 블록은 보다 분산이 높은 F 블록의 움직임 벡터에 영향을 받아 탐색 구간이 수정되며, 수정된 탐색 영역은 E가 된다. t+1 프레임의 A블록이 t-1 프레임의 E에서 움직임을 추정하였을 경우, t 프레임에서의 보간이 가능한 영역은 B 구간이므로 F블록과 겹쳐지지 않게 된다.

이 방법은 신뢰도가 높은 순서대로 움직임 추정이 수행됨으로, 신뢰도가 낮은 블록들은 신뢰도가 높은 블록들의 영향을 받게 되어 잘못 측정된 움직임 벡터의 발생을 방지할 수 있다.

제안하는 방법은 블록 기반 알고리즘에서 나타나는 문제점인 블록 열화를 감소시키기 위하여 중첩 움직임 추정을 수행한다. 따라서 탐색 영역 수정 시, 중첩 움직임 추정에 의한 블록의 중첩은 탐색 범위에 영향을 주지 않도록 한다.

4. 중첩 블록 움직임 추정 및 보상

블록 단위로 분산을 측정된 후, 측정된 분산을 이용하여 중첩 움직임 추정 및 보상을 수행한다. 분산이 높은 블록일수록 움직임 추정의 신뢰도가 높으므로, 분산이 높은 블록 순으로 중첩 블록 움직임 추정을 수행한다.

중첩 움직임 추정 및 보상이란 블록의 일부분을 겹쳐서 움직임 추정 및 보상을 수행하는 방법이다. 이를 위하여 수정된 탐색 범위에서 최소의 SAD (Sum of Absolute Difference) 를 가지는 블록을 찾은 후 임계값과 비교하여, SAD가 더 작을 경우에만 그 블록으로 보간 프레임을 구성한다. SAD가 임계값보다 큰 경우는 t+1 프레임에는 존재하지만 t-1 프레임에는 존재하지 않는 부분이거나, 그 반대의 경우일 수 있다. 또한 한 블록 안에 다른 방향으로 움직이는 객체가 두 개 이상 있는 경우에도 SAD가 크게 나타날 수 있어, 본 논문에서는 단 방향 움직임 추정을 통하여 블록을 보간 하지 않는다. 여기서, 전역 탐색

을 위한 (x, y) 위치의 블록의 SAD는 다음과 같이 정의된다.

$$SAD(x,y) = \sum_{j=0}^M \sum_{i=0}^N |f(t+1, x+j, y+i) - f(t-1, x+j, y+i)| \tag{3}$$

본 논문에서는 계층적 움직임 추정 및 보간에 바탕을 두고 있다. 최저해상도 프레임에서부터 원본 크기 바로 이전 해상도 프레임까지는 움직임 추정만을 수행하고, 원본 해상도 프레임에서 최종 움직임 추정을 완료한 후, 이를 바탕으로 프레임 보간을 이룬다. 원본 해상도 프레임 레벨에서 추정한 최종 움직임 벡터 (MV_j, MV_i)를 이용하여, 다음의 식에 따라 보간 프레임을 얻는다.

$$f(t, k, l) = \frac{(f(t-1, j+MV_j, i+MV_i) + f(t+1, j, i))}{2}, \tag{4}$$

$$k = j + \frac{MV_j}{2}, l = i + \frac{MV_i}{2}$$

중첩 블록 기반 움직임 보상 알고리즘과 탐색 범위에서 블록의 경계 부분이 겹치는 것은 허용했기 때문에 발생하는 부분은 블록의 경계 부분에서 발생하는 블록화 현상을 완화시키기 위한 것이므로, 겹치는 블록들의 평균으로 보상한다.

5. 저해상도 계층에서 고해상도 계층으로 투영

본 논문은 계층적 움직임 추정으로 구성되어 있어, 계층 간 움직임 투영을 필요로 한다. 고해상도 계층에서는 이전의 저해상도 계층에서 움직임 추정으로 추정한 움직임 벡터를 이용한다. 즉, 저해상도 계층에서 구한 움직임 벡터를 다음 계층에 투영시켜 그 좌표를 기준으로 탐색 범위를 정함으로써, 정확도를 향상시키면서 계산량을 감소시키는 것이 저해상도 계층에서 고해상도 계층으로 투영시키는 목적이다.

그림 3은 저해상도 계층에서 고해상도 계층으로 투영시키는 것을 표현하였다. 그림 3에서 볼 수 있듯이, 저해상도 계층에서의 블록 위치를 고해상도 계층에 해당하는

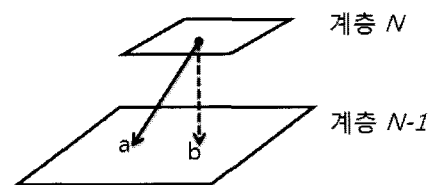


그림 3. 계층 간 투영 방법
Fig. 3. Projection between layers.

위치로 투영시킨 지점이 b 이고, 저해상도 계층에서 구한 움직임 추정 벡터를 이용하여 고해상도 계층으로 투영시킨 지점이 a 라고 할 때 저해상도 계층에서는 a 의 위치를 중심으로 탐색 범위를 적용하여 움직임 추정을 한다. 이때, 고해상도 계층에서의 탐색 영역의 기준 범위를 저해상도 계층에서 사용했던 움직임 추정 범위의 절반의 범위로 수행하여 계산량을 감소시킨다.

6. 화소값 기반의 배경과 객체의 분리

일반적으로 의미하는 객체와 배경을 나누기 위해서는 움직임 벡터를 이용하는 기법을 사용하지만, 잘못된 움직임 벡터의 발생으로 인하여 배경과 객체로 나누는 것은 쉽지 않다^[12~13]. 따라서 제안하는 알고리즘에서는 연속되는 프레임에서 같은 위치의 화소 사이의 차를 구한 후, 그 차이가 임계값 보다 작으면 화소값 기반의 배경으로, 크면 화소값 기반의 객체로 정의한다. 다음의 식은 연속된 두 프레임의 차이값을 나타낸다.

$$J(t, j, i) = |f(t+1, j, i) - f(t-1, j, i)| \quad (5)$$

여기서 J 는 두 화소의 밝기 값의 차이이다. 그림 4는 식 5에 의해 배경으로 추정된 영역에 대해서는 선형보간법을 이용하여 보간하고, 객체로 추정된 영역에서는 0으로 보간하여 화소값 기반의 배경과 객체로 분리한 프레임을 나타낸다. 움직임이 있는 부분만 0으로 보간된 것을 확인할 수 있다.



그림 4. 화소값 기반으로 배경과 객체 분리를 수행한 보간 프레임 (“Ballroom” 프레임)

Fig. 4. Interpolated frame performed pixel value based background and object segmentation (“Ballroom” sequence).

7. 값이 보간되지 않은 곳에서 선형 보간과 양방향 움직임 추정

화소의 단위가 소수점을 표현할 수 없는 정수형이고,

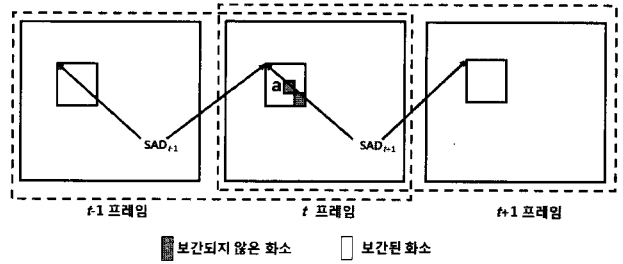


그림 5. 양방향 움직임 추정
Fig. 5. Bi-directional prediction.

움직임 추정과 보상에서의 임계값 때문에, 움직임 보상을 수행한 후에도 중간 프레임에는 값이 채워지지 않은 부분이 발생한다. 따라서 일반적으로 주변의 화소값들을 이용하여 보간 하거나 양방향 움직임 추정을 이용하여 빈 지역을 채우는 과정을 수행한다^[9~14].

그림 5는 양방향 움직임 추정에 대한 설명한다. 그림 5의 회색 부분은 값이 채워지지 않은 부분이고 흰색 부분은 값이 채워져 있는 부분이며 a 는 값을 채우고자 하는 화소를 표현하였다. a 의 값을 채울 때 a 를 중심으로 블록을 만들어 그 블록과 가장 닮은 블록을 $t-1$ 프레임과 $t+1$ 프레임에서 모두 찾으려 한다. 이때 블록 내에 a 외에도 값이 없는 부분이 존재하기 때문에 블록에서 값이 있는 화소만을 이용하여 SAD를 구한다. 이때 블록 내에서 값이 있는 화소의 개수는 변수이므로, 정확한 비교를 위하여 블록 내에서 값이 있는 화소의 개수로 나눠 정규화하여 임계치와 비교한다.

값을 채우기 위해 우선 (5)식으로 구한 J 값을 이용하여 보간되지 않은 화소가 화소값 기반의 배경 혹은 객체로 추정된 화소인지 확인한다. 배경으로 추정된 화소일 경우, 이 부분은 선형 보간법으로 보간한다. 객체로 결정된 화소일 경우, $t-1$ 프레임에서는 존재하지만 $t+1$ 프레임에는 존재하지 않는 객체이거나 그 반대의 경우에 의하여 값이 보간되지 않은 경우를 위하여 양방향 움직임 추정을 통해 값을 보간한다. 이 때, 값이 채워지지 않은 화소를 중심으로 블록을 만든 후, 이 블록을 $t-1$ 프레임과 $t+1$ 프레임에 대해서 움직임 추정을 수행하여 각각 SAD가 가장 작은 블록을 각각 찾는다. 다음의 수식으로 보간되지 않은 부분을 보간한다.

$$SAD_{t-1}(x, y) = \sum_{j=0}^M \sum_{i=0}^N |f(t, x+j, y+i) - f(t-1, x+j, y+i)|,$$

$$SAD_{t+1}(x, y) = \sum_{j=0}^M \sum_{i=0}^N |f(t, x+j, y+i) - f(t+1, x+j, y+i)|, \quad (6)$$

$$Norm.SAD_{t-1} = \frac{SAD_{t-1}}{NP_{j,i}}, \quad Norm.SAD_{t+1} = \frac{SAD_{t+1}}{NP_{j,i}}$$

$$\begin{cases} f(t,j,i) = \frac{f(t-1,j,i)+f(t+1,j,i)}{2}, & J(t,j,i) < \text{threshold} \\ f(t,j,i) = f(t-1,j,i), & \text{NomSAD}_{t-1} < \text{NomSAD}_{t+1} \\ & \text{and } \text{NomSAD}_{t-1} < \text{threshold} \\ f(t,j,i) = f(t+1,j,i), & \text{NomSAD}_{t+1} < \text{NomSAD}_{t-1} \\ & \text{and } \text{NomSAD}_{t+1} < \text{threshold} \\ f(t,j,i) = \frac{f(t-1,j,i)+f(t+1,j,i)}{2}, & \text{otherwise} \end{cases}$$

NormSAD는 보간된 화소들의 개수로 정규화 된 SAD 이며, $NP_{j,i}$ 는 (j, i) 위치의 블록 안에서 보간된 화소의 개수를 의미한다.

한 영상에만 사물이 존재하여 값이 보간 되지 않은 경우에는 한쪽의 영상의 SAD만 작게 나타나고, 다른 쪽의 프레임의 SAD는 크게 나타나게 된다. 따라서 두 SAD 중 더 작은 SAD가 임계값보다 작을 경우에만 그 블록으로 보간한다. 두 SAD 모두 임계값보다 클 경우에는 두 프레임 모두에서 찾지 못한 부분이다. 이런 부분에 한 쪽의 프레임의 블록으로 보상하는 것은 블록 현상의 발생 위험이 크므로 선형보간법으로 값을 채워준다.

선형보간법을 이용하면 뭉뚱화 현상이 발생하지만 블록 현상의 발생을 막을 수 있어 주관적 화질 저하를 막을 수 있다.

이 단계를 수행할 때, 블록의 크기가 커서 한 블록 안에 다른 방향으로 이동하는 객체가 두 개 이상 포함되어 있어서 보간 되지 않는 문제를 해결하기 위하여 블록의 크기를 줄여서 수행한다. 보간 되지 않은 부분은 움직임 벡터에 관한 정보가 없기 때문에 원본 크기의 프레임에서의 기준 탐색 영역을 수정 없이 사용한다.

III. 실험

제안한 방법의 성능을 평가하기 위해 원영상과 제안한 방법을 통해 보간된 프레임 사이의 PSNR을 측정하였다. 실험은 640×480 크기의 VGA급 영상 6개와 352×288 크기의 CIF급 영상 4개를 이용하였다. VGA급에서는 움직임 속도가 빠른 "Rena", "Race", "Ballroom" 영상과 중간 단계인 "Akko", "Flamenco2" 영상과 움직임이 느린 "Exit" 영상으로 실험하였고 CIF급에서는 움직임이 느린 "Hall_monitor", "Container" 영상과 움직임이 중간 단계인 "Foreman", "Coastguard" 영상으로 실험하였다.

실험을 위해 0부터 100번째까지의 프레임 중 짝수 번째 프레임을 이용하여, 나머지 50개의 홀수 번째 프레임을 생성하였고 실제 영상의 홀수 번째에 대하여 PSNR을 계산하였다. 세 개 이상의 계층을 이용하는 것은 계산량의 증가 폭에 비하여 성능 향상이 적기 때문에 계산량과

표 1. 계층에 따른 움직임 추정시 계산량 비교표
Table 1. Comparison complexity of each layers.

	단일 계층 움직임 추정	2개의 계층 움직임 추정
블록 개수	$1200 \left(= \frac{640 \times 480}{16 \times 16} \right)$	$300 \left(= \frac{320 \times 240}{16 \times 16} \right) + 1200 \left(= \frac{640 \times 480}{16 \times 16} \right)$
탐색 영역	$4096 (= 64 \times 64)$	$1024 (= 32 \times 32) + 256 (= 16 \times 16)$
계산 량	$4915200 (= 1200 \times 4096)$	$614400 (= 300 \times 1024 + 1200 \times 256)$
	단일 계층에서 움직임 추정에 대한 계산량이 2개의 계층에서 움직임 추정에 대한 계산량의 8배	

성능 향상과의 관계를 고려하여 두 개의 계층을 이용하였다. 블록 크기는 두 개의 계층 모두 16×16 매크로 블록 크기로 수행했으며, 탐색 영역은 계층 1에서는 ± 32 , 계층 0에서는 절반인 크기인 ± 16 으로 하였고, 전체 영역 탐색 방법으로 실험하였다.

제안하는 방법에서는 임계값을 많이 이용했는데 그 임계값들은 실험에 의해 결정된 값이다. 화소값 기반의 배경과 객체를 구분한 임계값은 17로 설정하였고, 움직임 추정의 임계값은 6000으로 하였다. 또한 양방향 탐색에서의 한 화소 당의 임계값은 10으로 수행하였고, 양방향 탐색 시 보간되지 않은 화소를 중심으로 하기 위하여 블록의 크기는 7×7 로 하였다.

표 1은 계층적 움직임 추정과 단일 계층에서의 움직임 추정에 대한 복잡도를 비교하기 위하여, 단일 계층과 2개의 계층에서 움직임 추정에 대한 계산량을 나타낸 것이다. 두 경우에서 VGA급 영상에서 16×16 매크로블록 단위로 ± 64 탐색 영역에서 전영역 탐색을 수행한 효과를 내도록 설계한 후, 수식을 통해 계산량을 측정하였다. 계산 결과, 단일 계층에서 움직임을 수행한 것보다 두 개의 계층으로 나누어 움직임을 추정하는 방법의 계산량이 8배 적은 것을 확인할 수 있었다.

그림 6과 7은 "Ballroom" 비디오의 36번째 프레임을 나타낸다. 그림 6은 블록의 분산의 우선순위에 따른 움직임 추정 및 보상 단계 전까지 수행한 결과이다. 프레임 대부분이 보간된 것을 확인할 수 있으며, 일반적으로 블록 현상이 많이 발생하는 객체의 경계 부분에서도 블록 현상이 발생하지 않고 자연스럽게 보간된 것을 확인할 수 있었다. 그림 7은 알고리즘의 모든 과정을 다 수행한 후의 결과이다. 뭉뚱화 현상이나 블록 현상이 거의 발생



그림 6. 움직임 보상까지 수행한 보간 프레임
Fig. 6. Interpolated frame performed motion compensation step.



그림 7. 최종 보간된 프레임
Fig. 7. Interpolated frame performed all steps.

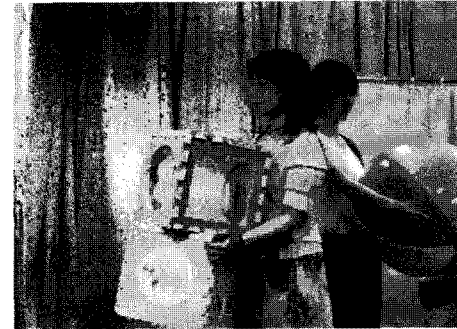
하지 않았고, 중간 프레임의 모든 화소가 보간되어진 것을 확인 할 수 있다.

그림 8의 프레임은 “Akko” 영상이며 23번째 프레임을 나타낸다. 제한한 알고리즘으로 보간한 “Akko” 영상의 PSNR 평균은 34.08이고, (b) 프레임의 PSNR은 34.07로 “Akko” 비디오 중 평균의 PSNR을 가진 프레임이다. 그림 8의 (a) 프레임은 원본 프레임이고 (b) 프레임은 제한한 알고리즘으로 보간한 프레임이다. (c) 프레임은 가우시안 피라미드와 라플라시안 피라미드를 이용하여 다수의 계층을 생성한 후, 각 계층에서 움직임 추정 및 보상을 수행하는 Lee 알고리즘을 이용하여 보간한 프레임이며 (d) 프레임은 움직임 벡터가 0인 블록에 가중치를 주는 방법과 미디언 필터를 사용하여 다양한 블록 크기로 움직임 추정을 수행하는 Fujiwara 알고리즘을 이용하여 보간한 프레임이다^[10~11].

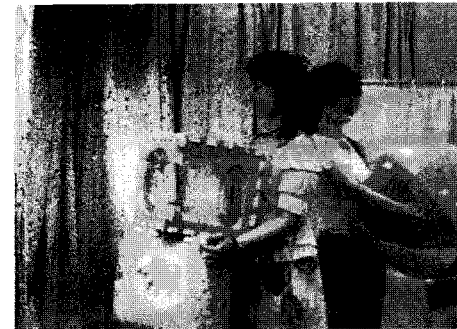
(c) 프레임은 배경 부분은 대부분 보상이 잘 이루어졌지만 잘못된 움직임 벡터에 의하여 사람의 얼굴 부분, 치마 부분과 상자 부분에서 오류가 발생한 것을 확인 할 수 있다. 즉, 객체 경계면에서 블록 현상과 몽롱화 현상이 나타나는 것을 볼 수 있다. (d) 프레임은 배경 부분은 대부분 보상이 잘 이루어진 것을 확인 할 수 있다. 하지만 객



(a)



(b)



(c)



(d)

그림 8. “Akko” 비디오에 대한 보간 프레임
(a) 원본 프레임 (b) 제안한 알고리즘에 의한 보간 프레임 (c) Lee 알고리즘에 의한 보간 프레임 (d) Fujiwara 알고리즘에 의한 보간 프레임

Fig. 8. “Akko” interpolated sequence.
(a) original sequence (b) interpolated sequence using proposal method (c) interpolated sequence using Lee’s algorithm (d) interpolated sequence using Fujiwara’s algorithm

체의 경계면에서는 다양한 크기의 블록 현상과 몽롱화 현상이 나타나는 것을 볼 수 있다.

제안한 방법으로 보간한 (b) 프레임은 배경 부분이 잘 보간되었고, 분산이 큰 블록을 우선적으로 움직임 추정을 수행하고 분산이 작은 블록들은 주변 블록의 움직임 벡터에 영향을 받기 때문에 (b) 프레임은 객체의 경계 부분이 블록 현상과 몽롱화 현상이 나타나지 않고 선명하게 나타나는 것을 볼 수 있다.

대부분의 비디오에서 객체의 움직임과 배경의 움직임은 다르게 나타난다. 기존의 알고리즘은 움직임 벡터의 정확도를 높이기 위하여 주변 움직임 벡터에 영향을 받도록 한다. 그렇기 때문에, 객체의 경계와 배경의 경계가 만나는 부분에서 움직임 벡터도 배경의 움직임 벡터와 객체의 움직임 벡터에 영향을 받게 된다. 실제, 분산이 낮은 배경 부분에서는 주변의 많은 화소들이 유사한 값을 가지고 있기 때문에 실제 움직임을 나타내는 움직임 벡터를 추정하기가 어렵다. 주변의 움직임 벡터가 정확하게 측정되었다 하여도, 경계 부분에서는 객체의 움직임 벡터와 배경의 움직임 벡터에 의하여 잘못된 움직임 벡터가 추정될 확률이 높아지게 된다.

하지만 제안한 방법은 분산이 높은 블록부터 움직임 추정을 수행한다. 분산이 높은 블록은 분산이 낮은 블록보다 실제 움직임을 나타내는 움직임 벡터를 추정할 확률이 높다. 정확한 움직임 추정에 대한 신뢰도가 높은 블록의 움직임 벡터의 영향을 받아 주변의 블록들이 움직임 추정을 수행함으로써 더 정확한 움직임 벡터를 추정할 수 있다. 따라서 경계 부분이 선명하게 나타나기 때문에

주관적 화질 개선 및 객관적 PSNR이 올라가는 것을 확인할 수 있다.

표 2는 VGA급 영상 6개와 CIF급 영상 4개의 PSNR을 비교한 표로, VGA급 프레임에서 약 3 dB 정도 향상된 것을 확인할 수 있었으며, CIF급 프레임에서는 약 1 dB 정도 향상된 것을 알 수 있다. 특히 움직임이 많은 프레임에서 높은 PSNR을 확인할 수 있다.

IV. 결 론

본 논문에서는 신뢰도에 따른 적응적 탐색 기법과 양방향 움직임 추정 및 선형보간법을 이용한 프레임율 증가 알고리즘을 제안하였다. 제안한 알고리즘을 이용하여 보상을 수행한 결과, 잘못된 움직임 벡터의 발생이 적어 주관적, 객관적으로 화질이 향상된 것을 확인할 수 있었다. VGA급 프레임에 대하여 기존의 방법 대비 PSNR이 2dB가 향상된 것을 확인할 수 있었다. 하지만, 제안하는 알고리즘은 계산량이 많고 분산이 높은 블록의 잘못 움직임 추정이 잘못될 경우, 주변 블록이 영향을 받게 된다는 단점이 있다. 추후 프레임율 증가 알고리즘과 관련하여, 제안한 방법의 계산량을 줄여 실시간 처리 가능 알고리즘으로 고안할 필요가 있다.

참 고 문 헌

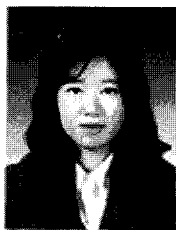
- [1] R. Castagno, P. Haavisto, and G. Ramponi, "A method for motion adaptive frame rate upconversion," *IEEE Transactions on Circuits Systems for Video Technology*, vol. 6, no. 5, pp. 436-446, Oct. 1996.
- [2] H. Sasai, S. Kondo, and S. Kadono, "Frame-rate up-conversion using reliable analysis of transmitted motion information," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Osaka, Japan, vol 5, pp. 257-260, May 2004.
- [3] Y.-K. Chen, A. Vetro, H. Sun, and S. Y. Kung, "Frame-rate up-conversion using transmitted true motion vectors," in *Proc. 2nd IEEE International Workshop on Multimedia Signal Processing*, CA, pp. 622-627, Dec. 1998.
- [4] Y.-T. Yang, Y.-S. Tung, and J.-L. Wu, "Quality enhancement of frame rate up-converted video by adaptive frame skip and reliable motion extraction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol 17, no. 12,

표 2. 알고리즘 별 성능 비교
Table 2. Comparison accuracy of each algorithms.

Size	Sequences	Fujiwara	Lee	Proposed
VGA (640× 480)	Exit	35.97	34.18	37.19
	Flamenco2	28.70	27.77	31.38
	Akko	31.29	24.55	34.13
	Race	25.72	23.99	28.94
	Rena	33.80	32.14	35.65
	Ballroom	27.54	26.01	29.22
Average		30.50	28.11	32.68
CIF (352× 288)	Container	43.36	39.29	42.86
	Hall_monitor	35.44	33.05	36.19
	Foreman	31.65	29.51	32.49
	Coastguard	28.77	27.09	31.00
Average		34.81	32.24	35.63

- pp. 1700-1713, Dec. 2007.
- [5] C. Cafforio, F. Rocca, and S. Tubaro, "Motion compensated image interpolation," *IEEE Transactions Communication*, vol. 35, no. 2, pp. 215-222, Feb. 1990.
- [6] G.-I. Lee, B.-W. Jeon, R.-H. Park, and S.-H. Lee, "Frame rate up-conversion using the wavelet transform," *Intv Conference on Consumer Electronics 2000*, Los Angeles, CA, pp. 172-173, June 2000.
- [7] B.-D. Choi, S.-J. Ko, J.-W. Han, and C.-S. Kim, "Frame rate up-conversion using perspective trans," *IEEE Transaction on Consumer Electronics*, vol. 52, no. 3, pp. 975-982, Aug. 2006.
- [8] J. Zhai, K. Yu, J. Li, and S. Li, "A low complexity motion compensated frame interpolation method," *IEEE Intactions Symposium on Circuits and Systems*, Kobe, Japan, pp. 4927-4930, May 2005.
- [8] J. Zhai, K. Yu, J. Li, and S. Li, "A low complexity motion compensated frame interpolation method," *IEEE Intactions Symposium on Circuits and Systems*, Kobe, Japan, pp. 4927-4930, May 2005.
- [9] D. Alfonso, D. Bagni, and D. Moglia, "Bi-directionally motion-copensated frame-rate up-conversion for H.264/AVC decoder," *ELMAR Symposium*, Zadar, Croatia, pp. 41-44, June 2005.
- [10] G.-I. Lee, B.-W. Jeon, R.-H. Park, and S.-H. Lee, "Hierarchical motion compensated frame rate up-conversion based on the Gaussian/Laplacian pyramid," *IEEE International Conference on Consumer Electronicspp*, Los Angeles, CA, pp. 350-351, June 2003.
- [11] S. Fujiwara and A. Taguchi, "Motion-compensated frame rate up-conversion based on block matching algorithm with multi-size blocks," *Proceedings of 2005 International Symposium on Intelligent Signal Processing and Communication Systems*, Tokyo, Japan, pp. 353-356, Dec. 2005.
- [12] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888-905, Aug. 2000.
- [13] S.-H. Lee, S. Yang, Y.-Y. Jung, and R.-H. Park, "Adaptive motion-compensated interpolation for frame rate up-conversion," *Proc 2002, IEEE International Conference on Consumer Electronicsin*, Los Angeles, CA, pp. 68-69, June 2002.
- [14] T. Chen, "Adaptive temporal interpolation using bidirectional motion estimation and compensation," *IEEE International Conference of Image Processing*, Princeton, NJ, pp313-316, Sept. 2002.

 저 자 소 개



민 경 연(학생회원)
 2009년 광운대학교 컴퓨터공학과
 학사.
 2009년 광운대학교 컴퓨터공학과
 석사 과정.
 <주관심분야 : 분산코딩, 컴퓨터
 비전>



박 시 내(정회원)
 2004년 광운대학교 컴퓨터공학과
 학사.
 2006년 광운대학교 컴퓨터공학과
 석사.
 2008년 광운대학교 컴퓨터공학과
 박사 수료.
 <주관심분야 : 분산코딩, 다시점 비디오 코딩>



심 동 규(정회원)
 1999년 서강대학교 전자공학과
 공학박사.
 1999년~2000년 (주) 현대 전자.
 2000년~2002년 (주) 바로 비전.
 2002년~2005년 Univ. of
 Washington.
 2005년~현재 광운대학교 컴퓨터공학과(부교수)
 <주관심분야 : 영상신호처리, 영상압축, 컴퓨터
 비전>