

특집논문 (Special Paper)

방송공학회논문지 제21권 제4호, 2016년 7월 (JBE Vol. 21, No. 4, July 2016)

<http://dx.doi.org/10.5909/JBE.2016.21.4.472>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

프레임율 변환을 위한 주파수 영역에서의 블로킹 현상 검출

김 남 옥^{a)}, 전 동 산^{b)}, 이 진 호^{b)}, 이 영 렬^{a)†}

Blocking Artifacts Detection in Frequency Domain for Frame Rate Up-conversion

Nam-Uk Kim^{a)}, Dongsan Jun^{b)}, Jinho Lee^{b)}, and Yung-Lyul Lee^{a)†}

요 약

본 논문에서는 MC-FRUC(Motion Compensated Frame Rate Up-Conversion) 에서 사용될 수 있는 주파수 영역에서의 블로킹 현상(blocking artifacts) 검출 알고리즘을 제안한다. MC-FRUC 알고리즘은 블록 단위로 움직임 벡터를 계산하여 움직임 보상을 수행함으로써 블록 경계선 주변에서 블로킹 현상이 발생할 수 있다. 본 논문에서는 효율적인 블로킹 현상 검출을 위해 보간된 영상(interpolated frame)의 주파수 성분을 분석하여 블록경계에서 블로킹 현상을 감소시킨다. 실험결과에서 제안된 알고리즘이 기존의 FRUC 방법보다 더 나은 주관적 화질을 보인다. 제안한 방법은 또한 기존의 BDMC(Bi-Directional Motion Compensation) 대비 평균 약 0.45 dB PSNR(Peak Signal to Noise Ratio)이 향상되었다.

Abstract

This paper proposes a blocking artifacts detection algorithm in frequency domain for MC-FRUC (Motion Compensated Frame Rate Up-Conversion). Conventional MC-FRUC algorithms occur blocking artifacts near interpolated block boundaries since motion compensation is performed from block-based motion vector. For efficiently decreasing blocking artifacts, this paper analyses frequency characteristics of the interpolated frame and reduces blocking artifacts on block boundaries. In experimental results the proposed method shows better subjective quality than some conventional FRUC method and also increases the PSNR(Peak Signal to Noise Ratio) value on average 0.45 dB compared with BDMC(Bi-Directional Motion Compensation).

Keyword : frame rate up-conversion, bi-directional motion compensation, median filter, occlusion detection, blocking artifacts

a) 세종대학교 컴퓨터공학과(Sejong University, Dept. of Computer Eng.)

b) 한국전자통신연구원(Electronics and Telecommunications Research Institute)

† Corresponding Author : 이영렬(Yung-Lyul Lee)

E-mail: yllee@sejong.ac.kr

Tel: +82-2-3408-3753

ORCID: <http://orcid.org/0000-0003-2709-8282>

※ 이 논문의 일부는 2016년도 정부(미래창조과학부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임 (B0101-16-0295, 초고품질 콘텐츠 지원 UHD 실감방송/디지털 시네마/사이니지 융합 서비스 기술 개발)

※ 이 논문의 연구결과 중 일부는 “IWAIT 2016”에서 발표한 바 있음.

· Manuscript received March 7, 2016; Revised May 9, 2016; Accepted May 17, 2016.

1. 서론

최근 UHD 디스플레이의 보급화와 고화질 동영상 스트리밍을 사용하는 모바일 사용자의 증가로 네트워크상의 비디오 트래픽의 비중이 급속도로 증가하고 있다. 그러나 이러한 소비자들의 수요에 맞추어 고수준의 UHD 방송이나 영상을 제공하기 위해 필요한 네트워크 대역폭은 현재로서는 확보가 불가능하다. LCD (Liquid Crystal Display)에서 UHD 영상을 잔상 없이 재생하기 위해서는 120Hz 이상의 프레임 율을 가져야 하지만 현재의 네트워크 기술로는 30Hz 정도가 한계이다. 이러한 낮은 프레임 율을 갖는 영상을 UHD 해상도의 대형 화면에서 재생할 경우 플리커 현상이나 모션블러 현상이 발생하여 영상 품질에 심각한 영향을 줄 수 있다. 그러므로 낮은 프레임 율을 갖는 영상에서 높은 프레임 율로 변환하는 FRUC (Frame Rate Up-Conversion) 기술을 사용하여 이러한 문제점들을 보완해야 한다. FRUC는 인접한 두 개의 프레임 사이에 1개 이상의 보간된 프레임을 삽입하여 영상의 프레임율을 증가시킴으로써 영상의 연속성을 증가시키는 기술이다^{[1][2]}. 최근까지의 FRUC 알고리즘은 크게 2가지 범주로 구분된다. 첫 번째 범주는 선형적인 수식을 사용하는 단순한 방법으로 인접한 프레임으로부터 픽셀들을 그대로 복사하거나 평균을 계산하여 보간된 프레임을 생성하는 기술이다^[3]. 이 방법은 계산속도가 빠르고 구현이 간단하다. 하지만 영상에서 물체의 움직임이나 변화가 빠를수록 고스트 현상(ghost arti-

facts)이 발생하거나 영상의 연속성이 개선되지 않는 단점이 있다. FRUC 알고리즘의 두 번째 범주로는 MC-FRUC (Motion Compensated Frame Rate Up-Conversion)^{[4][9]}가 있다. MC-FRUC는 움직임 추정(Motion Estimation)과 움직임 보상 보간(Motion Compensated Interpolation) 과정을 통해 중간 프레임을 보간하는 방법이다. 움직임 추정을 통해 물체의 움직임을 계산하여 보상해줌으로써 고스트현상을 줄여주고 영상의 연속성을 크게 증가시켜준다. 하지만 첫 번째 범주의 알고리즘에 비해 계산복잡도가 높은 단점이 있다. 최근에는 기술 발전으로 인하여 작은 크기의 처리기로도 충분히 처리 가능한 MC-FRUC 알고리즘이 주로 쓰이고 있다.

MC-FRUC 알고리즘은 단방향 움직임 벡터를 사용하는 방법^{[4][5]}과 양방향 움직임 벡터를 사용하는 방법^{[6][7]}으로 나뉜다. 단방향 움직임 벡터를 사용하는 방법은 그림 1과 같이 인접한 프레임 F_{n-1} 과 F_{n+1} 에 대하여 하나의 움직임 벡터만을 사용하는 방법이다. 보간되는 위치가 움직임 벡터의 값에 따라 가변적으로 변하기 때문에 보간되는 영역들 사이에 구멍(hole)이나 겹침(overlaps)과 같은 문제가 발생할 수 있다. 이러한 문제를 해소하기 위하여 양방향 움직임 벡터를 사용하는 BDMC(Bi-Directional Motion Compensation) 방식이 제안되었다^{[6][7]}. 이 방법은 움직임 벡터 두 개를 사용하여 보간하는 방법으로 단방향 움직임 벡터를 사용하는 방법과 다르게 보간되는 영역이 움직임 벡터의 값과 상관없이 항상 일정하기 때문에 구멍이나 겹침과 같은 문제가 발생하지 않는다. 하지만 이 방법은 움직임 벡

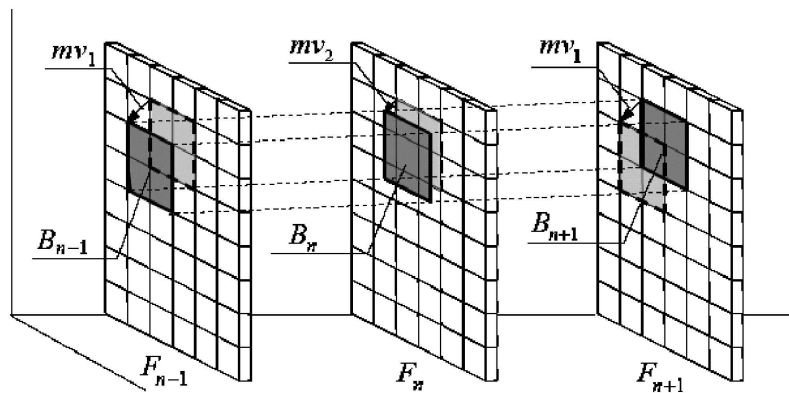


그림 1. 단방향 움직임 보상 과정
 Fig. 1. Unidirectional motion compensation process

터들이 폐색된 영역에서 블로킹 현상(blocking artifact)이나 움직임 벡터의 오류로 인한 심각한 영상 열화가 발생하는 단점이 있다. 이러한 단점을 줄이기 위해 최근까지 많은 연구가 진행되고 있다. EBME(Extended Bilateral Motion Estimation)^[10]는 움직임 추정 과정에서 블록을 가로와 세로 방향으로 블록크기의 1/2만큼 이동하여 추가적인 움직임 추정과정을 수행하는 방법이다. 움직임 보상 과정에서 보간 블록은 4개의 서브블록으로 분할되어 블록과 동일 위치에서 움직임 추정으로 얻어진 움직임 벡터와 1/2 이동되어 추가적으로 움직임 추정을 하여 얻는 움직임 벡터 중 최적의 움직임 벡터를 선택하여 움직임 벡터의 오류를 최소화한다. 이 방법은 움직임 추정의 계산복잡도가 2배로 증가하지만 복잡도 대비 성능향상이 적어 실제로 사용하기는 힘든 기술이다. SMD(Side match distortion)^[11]는 움직임 추정 시 SAD(Sum of Absolute Difference)를 포함하여 여러 개의 비용 함수를 사용하여 움직임 벡터의 오류를 최소화하는 방법이다. 이 기술은 재귀적인 알고리즘이 사용되어 현실적으로 사용하기 힘든 계산복잡도를 갖기 때문에 실제로 적용하기는 힘든 기술이다. 본 논문에서는 계산복잡도의 큰 증가 없이 움직임 벡터의 오류를 줄이기 위해 폐색영역 탐지와 중간값 필터링을 사용하는 방법을 소개하며 블로킹 현상 문제를 줄이기 위해 BDMC를 수행한 보간 블록에 대하여 주파수 성분을 분석하여 블로킹 현상이 발생했는지

판별하고 주파수 성분의 에너지 분포에 따라 디블로킹 필터를 적응적으로 사용하는 방법에 대하여 제안한다.

이후 2장 “FRUC 알고리즘”에서는 BDMC에 대한 자세한 설명과 폐색영역에서 움직임 벡터의 오류와 화질 저하를 최소화하기 위한 방법에 대해 설명하고 3장 “제안하는 방법”에서는 주파수 영역에서의 블로킹 현상 검출에 대해 설명한다. 4장 “실험 결과 및 분석”에서는 다양한 시퀀스들을 기존의 FRUC 알고리즘들과 제안되는 알고리즘과의 화질 차이에 대하여 비교 및 분석하였다.

II. FRUC 알고리즘

제안하는 방법을 소개하기 전에 FRUC 알고리즘 중 하나인 BDMC 과정과, 폐색영역 검출 그리고 중간값 필터링에 대하여 자세히 설명한다. 제안하는 주파수 영역에서의 블로킹 현상 검출 알고리즘은 BDMC 과정을 수행하여 보간 프레임을 생성한 후 수행된다. 폐색영역은 주변과의 움직임 정보의 연속성이 단절되는 영역으로 블로킹 현상이나 다른 영상 열화가 발생할 가능성이 높은 영역이다. 그러므로 이러한 영역을 검출하는 알고리즘에 대해 알아보고 화질 개선을 위해 중간값 필터링을 사용한다. 그리고 블로킹 현상을 감소시키기 위하여 주파수 영역에서 블로킹 현상을

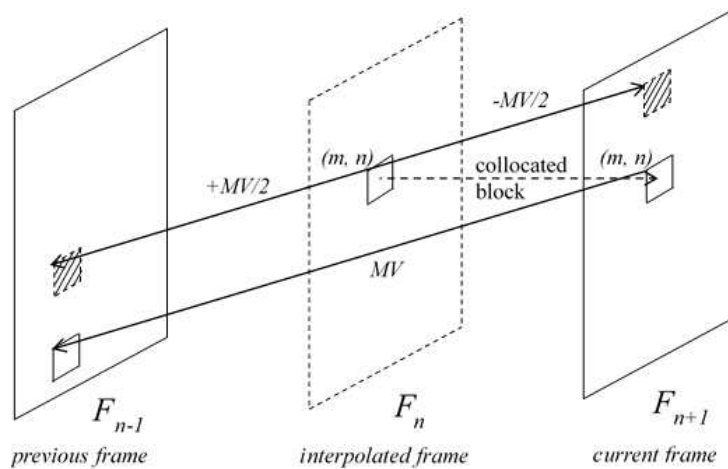


그림 2. BDMC(Bi-Directional Motion Compensation) 과정
 Fig. 2. BDMC(Bi-Directional Motion Compensation) process

검출하고 주파수 분포에 따라 다른 디블로킹 필터를 적용하여 화질을 개선하는 방법에 대해 제안한다.

1. BDMC

BDMC는 2개의 움직임 벡터를 사용하여 움직임 보상을 하는 방법이다. 그림 2는 블록기반 BDMC 수행 과정을 보여준다. 보간 프레임 F_n 을 블록으로 나누어 각각의 블록마다 BDMC 과정이 독립적으로 수행된다. BDMC 과정은 움직임 추정과 움직임 보상의 단계로 나뉜다. 움직임 추정에서는 최적의 움직임 벡터를 계산하기 위해 BMA(Block Matching Algorithm)을 이용하여 보간 블록과 동일한 위치 (m, n) 에 있는 F_{n+1} 의 블록으로부터 F_{n-1} 로의 움직임 벡터 MV 를 계산한다. 계산된 MV 를 이용하여 이전 프레임 F_{n-1} 으로의 움직임 벡터를 $MV/2$ 로 이후 프레임 F_{n+1} 으로의 움직임 벡터를 $-MV/2$ 로 할당한다. 이후 움직임 보상 과정

에서 식 (1)과 같이 보간 프레임 F_n 을 생성한다. MV_{hor} 는 MV 의 x 성분, MV_{ver} 는 MV 의 y 성분이다.

2. 폐색영역탐지 알고리즘

BDMC 과정에서 생성되는 움직임 벡터들은 항상 유효한 값을 얻어내진 않는다. 만일 계산된 움직임 벡터의 값이 실제 움직임과 전혀 다른 값일 경우 블로킹 현상을 포함하여 심각한 영상열화를 초래할 수 있다. 그렇기 때문에 유효하지 않은 움직임 벡터가 존재하는 폐색영역을 검출하는 알고리즘^[12]이 필요하다. 폐색영역이란 움직임의 연속성이 단절되는 영역으로 이러한 영역내의 움직임 벡터들은 대부분 잘못된 움직임을 나타낸다. 만일 폐색영역 탐지 알고리즘으로 폐색영역으로 결정된다면 중간값 필터링을 적용하여 열화를 최소화시킨다. 폐색영역이 아니라고 결정된다면 기존의 BDMC를 적용하여 보간화소를 생성한다. 폐색영역

$$F_n(x,y) = BDMC(x,y) = \frac{F_{n-1}(x + MV_{hor}/2, y + MV_{ver}/2) + F_{n+1}(x - MV_{hor}/2, y - MV_{ver}/2)}{2} \quad (1)$$

$$\begin{aligned} T_x &= |AboveMV_{hor} + LeftMV_{hor}| \\ T_y &= |AboveMV_{ver} + LeftMV_{ver}| \\ T &= |T_x - T_y| \end{aligned} \quad (2)$$

$$occlusion = (|MV_{hor} - LeftMV_{hor}| > T \text{ or } |MV_{ver} - AboveMV_{ver}| > T)$$

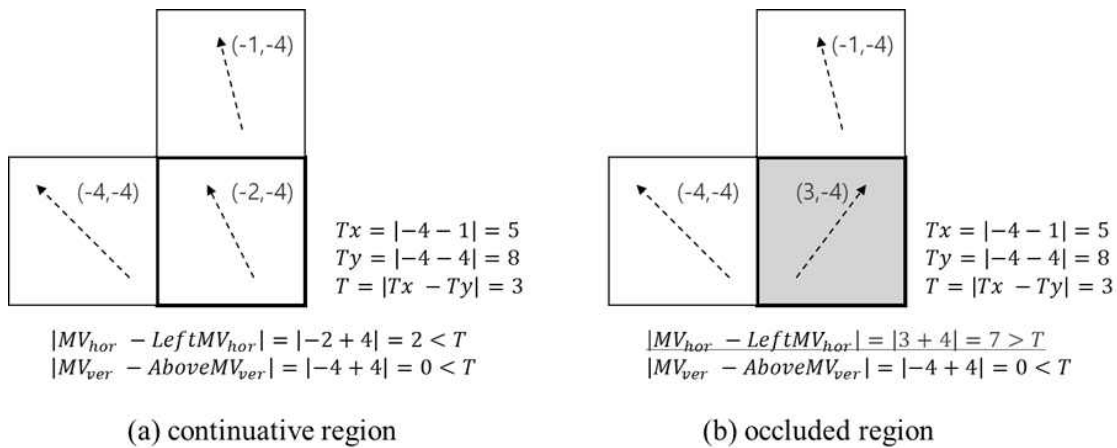


그림 3. 폐색영역 탐지의 예
 Fig. 3. An example of occluded region detection

탐지의 자세한 알고리즘은 식 (2)와 같다.

식 (2)에서 $|\cdot|$ 연산은 절대값을 취하는 연산이다. LeftMV 와 AboveMV 는 MV 와 공간적으로 인접한 움직임 벡터를 의미하며 각각 왼쪽과 위쪽에 인접한 움직임 벡터를 나타낸다. T 는 주변 움직임 벡터와의 차의 임계값 (Threshold)을 의미한다. 임계값은 주변 움직임 벡터와 차이가 클수록 임계치가 줄어들어 폐색영역이 될 확률이 증가하도록 하였다. 실제 식 (2)를 바탕으로 폐색영역 탐지 알고리즘을 적용한 예시를 그림 3에 도시하였다. 왼쪽 (a)는 폐색영역이 아닌 경우이고 오른쪽 (b)는 폐색영역으로 탐지된 경우이다. 폐색영역이 아닌 경우에는 주위의 움직임 벡터와 비슷한 움직임 벡터를 가지고 있지만 폐색영역이 탐지된 경우에는 주변의 움직임 벡터와 상이한 값을 갖는다.

3. 중간값 필터

폐색 영역 탐지로 검출된 영역에서 계산된 움직임 벡터를 그대로 사용하여 BDMC를 적용할 경우 부적절한 보간 영상을 생성하게 된다. 이러한 문제를 줄이기 위하여 중간값 필터를 사용한다. 중간값 필터링은 폐색영역에서의 화질 감소를 최소화해준다. 중간값 필터링은 움직임 보상된 픽셀과 움직임 보상 없이 보간된 픽셀 값들 중 중간 값을 택한다. 움직임 벡터의 정확성이 크게 떨어지는 부분에서 중간값 필터링을 적용한다면 블로킹 현상을 크게 줄일 수 있다. 중간값 필터링에는 정적 중간값 필터링(Static Median Filtering)과 동적 중간값 필터링(Dynamic Median Filtering)이 있다. 정적 중간값 필터링은 아래의 식(3)과 같이 F_{n-1} 과 F_{n+1} 프레임에서 움직임 보상이 되지 않은 픽셀값 2개를 후보로 선택하고 BDMC값을 후보픽셀로 선택한 총 3가지 후보 중 중간 값을 선택한다.

$$\begin{aligned} v_0 &= F_{n-1}(x,y) \\ v_1 &= F_{n+1}(x,y) \\ v_2 &= BDMC(x,y) \\ F_n(x,y) &= median(v_0, v_1, v_2) \end{aligned} \tag{3}$$

median은 3개의 값들 중 중간값을 택하는 함수이다. 정적

중간값 필터링은 사람의 표정 변화나 환경광의 변화같이 물체의 경계가 복잡하게 변할 경우 효과적이지만 고정된 형태의 물체가 이동하는 영상에는 성능이 매우 떨어졌다. 두 번째로 동적 중간값 필터링(Dynamic median filtering)은 식(4)와 같이 F_{n-1} 과 F_{n+1} 으로부터 움직임 보상이 된 두 픽셀을 후보로 선택하고 F_{n-1} 과 F_{n+1} 에서 평균값을 구해후보로 선택한 총 3가지 후보 중 중간 값을 선택한다.

$$\begin{aligned} v_0 &= F_{n-1}(x + MV_{hor}/2, y + MV_{ver}/2) \\ v_1 &= F_{n+1}(x - MV_{hor}/2, y - MV_{ver}/2) \\ v_2 &= \frac{F_{n-1}(x,y) + F_{n+1}(x,y)}{2} \\ F_n(x,y) &= median(v_0, v_1, v_2) \end{aligned} \tag{4}$$

동적 중간값 필터링은 정적 중간값 필터링과 반대로 물체의 움직임이 비교적 클 경우 효과적이었고 작은 물체의 경계선 변화에는 비교적 낮은 성능을 보였다. BDMC에서 사용하는 움직임 추정만은 물체의 움직임만을 모델링 하였으므로 물체의 경계의 변화에 효과적인 정적 중간값 필터링을 사용하는 것 보다 물체의 이동에 비교적 높은 성능을 보이는 중간값 필터링을 적용하는 것이 효율적이다. 실제 영상으로 실험을 하였을 경우 정적 중간값 필터링 보다 동적 중간값 필터링을 사용할 경우 객관적 화질이 더 높았고 링잉 노이즈(Ringing artifacts)도 적게 발생하였으며 물체의 경계선이나 무늬같이 뚜렷한 경계가 뭉개지는 현상도 비교적 적게 나타났다.

III. 제안하는 방법

폐색영역 탐지 알고리즘을 통해 검출된 블록에서는 주변 블록과 상이한 움직임 벡터를 사용하여 움직임 보상을 수행하였기 때문에 블로킹 현상이 발생하였다. 동적 중간값 필터링을 사용하여 움직임 보상의 오차를 최소화하였지만 여전히 블록의 경계선에서 블로킹 현상이 발생한다. 이러한 열화를 줄이기 위하여 블록의 경계 픽셀들의 주파수 특성에 적응적인 디블로킹 필터를 수행하는 알고리즘을 제안

한다. 기존의 블로킹 현상 검출 알고리즘^[13]에서는 블로킹 현상을 감소시키기 위해 참조 프레임간의 움직임 추정 오차인 SAD(Sum of Absoulte Difference) 값의 크기를 일정 임계치와 비교하여 오차가 임계치보다 큰 블록에 대하여 디블로킹 필터를 적용하였다. 디블로킹 필터는 좌측 또는 상단 경계에 인접한 화소들에 대해 고정된 필터 탭인 [1, 2, 1]을 적용하여 수행된다. 이 방법은 블로킹 현상의 강도에 상관없이 항상 일정한 필터링을 수행한다. 하지만 영상의 종류 혹은 영상의 영역마다 블로킹 현상의 강도가 다르기 때문에 블로킹 현상의 강도에 적합한 디블로킹 필터를 적용한다면 보다 좋은 결과를 얻을 수 있을 것이다. 본 절에서는 블로킹 현상의 강도를 주파수 성분의 에너지로 계산하여 디블로킹 필터의 강도를 조절하는 알고리즘을 제안한다.

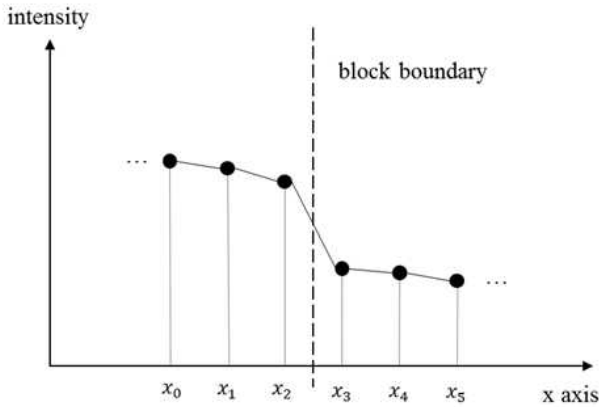


그림 4. 블록 경계에서 블로킹 현상이 발생할 경우의 예시
 Fig. 4. An example of blocking artifacts occurrence on block boundary

그림 4는 폐색영역의 경계선에서 발생하는 블로킹 현상의 일반적인 패턴을 보여준다. 가로축은 영상의 x축이고 세로축은 픽셀의 명도를 나타낸다. 그림과 같이 블로킹 현상이 발생할 경우 블록의 경계선에서 픽셀의 명도의 급격한

변화가 발견된다. 이러한 특성을 이용하여 BDMC로 보간된 블록의 경계선 근처의 픽셀들의 주파수 특성을 이용하여 블로킹 현상을 검출하는 방법을 제안한다.

경계선 근처의 픽셀들의 도메인을 공간 영역에서 주파수 영역으로 변환하기 위해 DCT-II(Discrete Cosine Transform type-II)^[14]가 사용된다. 식 (5)은 DCT-II를 나타낸다.

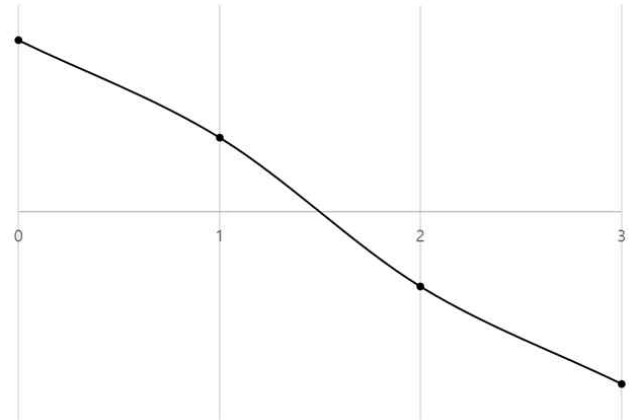


그림 5. N=4 일때의 DCT-II의 X_1 성분
 Fig. 5. X_1 component of DCT-II when N=4

N은 입력되는 신호(화소) 수(number of pixels)이고 x_i 는 입력되는 신호 그리고 X_k 는 주파수 성분을 의미한다. N=4 일 경우 주파수 성분은 X_0, X_1, X_2, X_3 로 구분된다. 이중 X_1 성분은 그림 5와 같다. 이 그래프의 모양은 그림 4의 블록 경계선 근처의 점 4개의 모양과 유사한 모양을 가지고 있다. 주파수 변환의 특성상 특정 주파수 성분의 모양과 비슷한 신호일수록 더 많은 에너지가 그 주파수 성분에 모이게 된다. 그러므로 X_1 성분에 많은 에너지가 모였을 경우 블로킹 현상이 발생할 확률이 높다고 판단할 수 있다.

블로킹 현상 검출은 움직임 추정과 움직임 보상을 통해 보간 프레임이 완성된 후 수행된다. 움직임 추정을 수행하는 블록 크기가 8x8일 경우 제안하는 블로킹 현상 검출 알

$$X_k = \alpha_k \sum_{i=0}^{N-1} x_i \cos\left[\frac{\pi}{N}(i+0.5)k\right] \quad \alpha_0 = 1/\sqrt{N} \quad k=0$$

$$\alpha_k = 2/\sqrt{N} \quad k=1, \dots, N-1$$
(5)

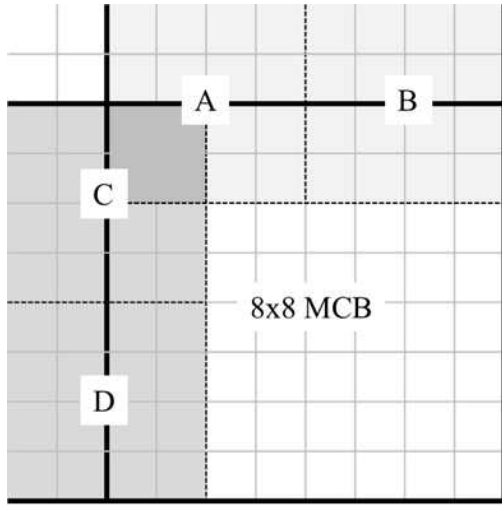


그림 6. 8x8 MCB(Motion Compensated Block)에서의 블로킹 현상 검출 영역
 Fig. 6. Blocking artifacts detection region on MCB(Motion Compensated Block)

고리즘은 그림 6과 같이 8x8블록의 상단과 좌측 경계선 근처의 4x4 영역 단위로 수행된다. A와 B는 상단 경계선 근처 영역으로 세로 방향 블로킹 현상 검출을 수행한다. C와 D는 좌측 경계선 근처 영역으로 가로 방향 블로킹 현상 검출을 수행한다. 8x8 블록 간에 독립적으로 수행할 수 있도록 프레임 내의 모든 8x8 블록에 대하여 가로 방향 블로킹 현상을 먼저 검출한 후 세로 방향 블로킹 현상을 검출한다.

4x4 영역에서 세로 방향 블로킹 현상 검출 과정은 그림 7과 같이 4x4 영역을 참조하여 수행된다. 블로킹 현상 검출에는 $a_{0,-2}$ $a_{0,-1}$ $a_{0,0}$ $a_{0,1}$ 와 $a_{3,-2}$ $a_{3,-1}$ $a_{3,0}$ $a_{3,1}$ 두 개의 라인의 1차원 DCT-II 변환계수가 사용된다. 두 개의 라인 모두 X_1 성분의 에너지가 일정 임계치보다 클 경우 블로킹 현상이 크게 발생했다고 판단하여 4x4 블록에 대하여 강한 디블로킹 필터링을 적용한다. 그렇지 않다면 폐색영역일 경우에만 약한 디블로킹 필터링을 적용한다. 폐색영역이 아닐 경우에는 디블로킹 필터링을 수행하지 않는다. 주파수 성분의 에너지는 식 (6)과 같이 계산된다.

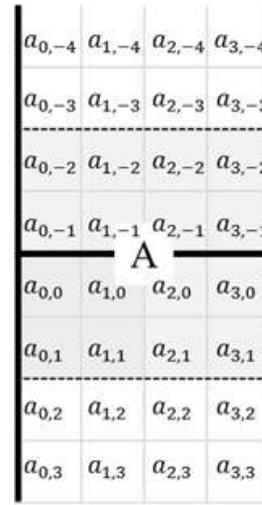


그림 7. 그림 6의 A영역 내 화소들
 Fig. 7. Pixels in A region of Fig. 6

$$E_k = \frac{|X_k|}{|X_0|+|X_1|+|X_2|+|X_3|} \times 100(\%) \quad (6)$$

주파수 성분의 에너지는 4개의 주파수 성분의 절대총합 중 특정 주파수 성분의 절대값의 비율로 계산된다. DCT 변환의 특성상 X_0 (DC)성분에 절대적으로 많은 에너지가 모인다. 그렇지만 에너지를 측정해 DC 성분을 포함한 이유는 블로킹 현상이 발생했을 경우의 X_1 에너지의 분포의 분산을 줄이기 위해서이다.

블로킹 현상이 발생했을 경우 아래 식 (7)과 같은 강한 디블로킹 필터링을 4x4 블록에 대하여 적용한다. 만일 블로킹 현상이 발생하지 않았더라도 폐색영역이라면 식 (8)과 같은 약한 디블로킹 필터링을 적용한다. 아래의 식 (7)과 (8)에서 $a'_{x,y}$ 는 수정된 화소 값을 의미한다. 식 (7), (8) 모두 $x=0, 1, 2, 3$ 에 대하여 적용하여 4x4 블록 전체에 대하여 필터링을 수행한다.

$$a'_{x,-2} = \frac{a_{x,-4} + 2a_{x,-3} + 2a_{x,-2} + 2a_{x,-1} + a_{x,0}}{8} \quad (7)$$

$$a'_{x,-1} = \frac{a_{x,-4} + a_{x,-3} + a_{x,-2} + 2a_{x,-1} + a_{x,0} + a_{x,1} + a_{x,2}}{8}$$

$$\begin{aligned}
 a'_{x,0} &= \frac{a_{x,-3} + a_{x,-2} + a_{x,-1} + 2a_{x,0} + a_{x,1} + a_{x,2} + a_{x,3}}{8} \\
 a'_{x,1} &= \frac{a_{x,-1} + 2a_{x,0} + 2a_{x,1} + 2a_{x,2} + a_{x,3}}{8} \\
 a'_{x,-1} &= \frac{a_{x,-3} + a_{x,-2} + 4a_{x,-1} + a_{x,0} + a_{x,1}}{8} \\
 a'_{x,0} &= \frac{a_{x,-2} + a_{x,-1} + 4a_{x,0} + a_{x,1} + a_{x,2}}{8}
 \end{aligned} \tag{8}$$

C와 B영역에 적용되는 가로 방향 블로킹 검출과정은 방향만 세로에서 가로방향으로 변경되어 세로 방향 블로킹 검출과정과 동일하게 적용된다.

IV. 실험 결과 및 분석

1. 실험 조건

실험에는 HEVC 표준 테스트 시퀀스로 사용되는 일부 영상이 사용되었다. 각각 영상에 대해 101프레임을 사용하였으며 영상의 해상도는 2560x1600와 1920x1080 으로 구성되어있다. 영상들의 프레임율은 24~60Hz로 구성되어 있으며 컬러포맷은 YUV420이고 비트심도는 8비트와 10비트 두 종류가 있지만 10비트인 경우 8비트로 다운스케일링하여 사용하였다. FRUC를 실험하기 위하여 101 프레임중 홀수 번째 프레임을 생략하고 짝수 번째 프레임들에 FRUC 알고리즘을 사용하여 홀수 번째 프레임 위치에 보간영상을

표 1. 실험 조건

Table 1. Experimental conditions

video resolution	block size	search range
2560x1600(WQXGA)	16x16 pixel	±32 pixel
1920x1080(FHD)		

생성하였다. 생성된 보간영상은 원래의 영상과 오차를 비교하여 성능을 측정하였다. 성능 측정에는 PSNR(Peak Signal to Noise Ratio)를 사용하였고 실험 조건에 대한 요약은 표 1과 같다.

2. 실험 결과

표 2 는 기존의 FRUC 알고리즘과 제안하는 방법과의 평균 PSNR 값의 비교를 나타낸다. 알고리즘들 중 가장 높은 평균 PSNR을 굵은 글씨로 표기하였다. 제안하는 알고리즘의 계산 복잡도는 기존 BDMC 알고리즘과 거의 동일하여 따로 표기하지 않았다. BDMC의 움직임 추정과정의 계산 복잡도가 높아 블로킹 현상 검출 과정의 계산량이 전체 연산시간에 영향을 크게 주지 않았다.

표 2에서 보는 바와 같이 대부분의 시퀀스에서 제안하는 알고리즘의 객관적 화질이 향상되었다. 가장 높은 향상을 가졌던 시퀀스는 “PeopleOnStreet”로 횡단보도에 많은 사람들이 길을 건너는 모습을 고정된 위치의 카메라로 촬영한 영상이었다. 시퀀스 전체적으로 많은 움직임들이 있어 기존의 FRUC 알고리즘에서 발생하던 블로킹 현상을 제안

표 2. PSNR 비교 결과

Table 2. Comparison of PSNR results

Sequences	resolution	frame rates	DMF[2]	BDMC[6]	OD[12]	Proposed
Traffic	WQXGA	30	34.69	35.66	35.87	35.72
PeopleOnStreet	WQXGA	30	25.23	24.78	25.42	25.62
Nebuta	WQXGA	60	24.85	25.49	25.78	25.80
SteamLoco	WQXGA	60	33.43	34.10	34.46	34.52
Kimono	FHD	24	31.28	31.00	31.57	31.75
ParkScene	FHD	24	32.35	32.84	33.09	33.08
Cactus	FHD	50	30.72	30.42	31.01	31.02
BasketballDrive	FHD	50	27.23	27.19	27.46	27.53
BQTerrace	FHD	60	28.79	29.28	29.60	29.75
Average PSNR			29.84	30.08	30.47	30.53

하는 알고리즘을 통해 효과적으로 감소되었다. “Traffic” 시퀀스는 오히려 블로킹 현상 검출 알고리즘을 적용하고도 화질이 떨어졌다. “Traffic”은 열차가 철도를 지나는 장면을 촬영한 시퀀스이다. 열차의 굴뚝에서 연기가 크게 퍼져나가는 부분에서 폐색영역이 넓은 범위로 발생하여 보간 화소의 정확성이 크게 떨어졌다. 정확성이 크게 떨어지는 보간 화소들에 대하여 더블로킹 필터링을 수행하였을 경우 오히려 객관적 화질이 떨어짐을 확인할 수 있었다. 평균 PSNR은 제안하는 알고리즘이 가장 좋았으며 BDMC에 비해 약 0.45dB의 성능 향상이 있었다. 더블로킹 필터링만의 성능향상은 약 0.06dB로 높은 객관적 화질 향상은 아니지

만 실제 보간된 시퀀스를 확인한 결과 날카로운 블록 경계가 보다 덜 보여 주관적 화질 향상이 있음을 확인할 수 있었다.

그림 8은 PeopleOnStreet 테스트 시퀀스의 75번째 장면 중 일부분이다. 그림 8(b) BDMC를 사용하여 보간된 영상인 경우 블로킹 현상이 크게 두드러지는 것을 확인할 수 있었다. (c) 동적 중간값 필터링을 사용한 영상인 경우 BDMC를 사용하였을 때 보다 블로킹 현상이 다소 감소하였다. 하지만 약간의 링잉 현상이 발견되었다. (d) 폐색영역 검출 알고리즘을 적용한 영상은 (b) 와 (c)에 나타났던 블로킹 현상과 링잉 현상이 약간씩 감소하였다. (e) 제안하는



(a) 원본 영상
(a) Original image



(b) BDMC로 보간된 영상
(b) Image interpolated by BDMC



(c) 동적 중간값 필터링으로 보간된 영상
(c) Image interpolated by dynamic median filtering



(d) 폐색영역 검출 알고리즘을 사용하여 보간한 영상
(d) Image interpolated by occlusion detection algorithm



(e) 폐색영역 검출 알고리즘과 제안된 알고리즘을 사용하여 보간한 영상
(e) Image interpolated by both occlusion detection algorithm and the proposed deblocking filter

그림. 8. PeopleOnStreet 비디오 시퀀스의 75번째 영상(a~e)
그림. 8. 75th image (a-e) of PeopleOnStreet video sequence

알고리즘을 적용한 영상은 (d)에서 발생하는 블로킹 현상을 좀 더 부드러운 경계선을 만들어 줌으로써 주관적 화질이 향상되었고 실험결과 객관적 화질도 향상되었다. 하지만 기존 블로킹 현상 검출 알고리즘[13]과의 차이는 생각보다 크지 않았다. 객관적 화질비교에서는 거의 비슷한 정도였고 주관적 화질비교에서는 미묘한 차이만 발생하였다. 일부 폐색영역의 경계에서는 제안하는 방법이 블록의 경계를 적정수준으로 뭉개어 좋은 성능을 보이기도 하였다. 하지만 전체적으로 차이가 발생하는 부분이 적기 때문에 보다 좋은 성능을 나타내기 위해서는 알고리즘의 개선이 필요할 것이다.

V. 결 론

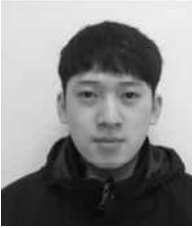
주파수영역 분석을 통한 블로킹 현상 검출 알고리즘을 기존의 FRUC 알고리즘에 적용하였을 경우 블로킹 현상의 강도에 따라 적응적으로 디블로킹을 수행함으로써 주관적 화질 향상과 객관적 화질 향상이 있음을 실험결과를 통해 보였다. 제안되는 방법은 이전에 연구하였던 폐색영역 검출로 BDMC와 DMF를 선택적으로 적용하는 알고리즘[10]에 발생하였던 블로킹 현상 문제를 해결 혹은 감소시키기 위하여 연구되었다. 실험을 통해 약간의 화질 향상이 있다는 것을 확인하였지만 실험 영상을 확인한 결과 블로킹 현상의 경계선은 부드럽게 변하였지만 블로킹 현상은 사라지지 않았음을 확인할 수 있었다. 블로킹 현상의 근본적인 원인인 부정확한 움직임 벡터를 수정하지 않는다면 블로킹 현상의 해결은 어려울 것으로 보인다. 이러한 문제를 해결하기 위해 폐색영역에 대하여 물체의 변형을 고려한 다양한 움직임 모델을 적용하여 최적의 움직임 벡터를 계산하는 연구가 다음의 연구 주제로 수행될 예정이다.

참 고 문 헌 (References)

- [1] C. Cafforio, F. Rocca, and S. Tubaro, "Motion Compensated Image Interpolation," IEEE Trans. Communications, vol. 38, no. 2, pp. 215-222, Feb. 1990.
- [2] O. A. Ojo and G. de Haan, "Robust motion-compensated video up conversion," IEEE Trans. Consum. Electron., vol. 43, no. 3, pp. 1045-1056, Nov. 1997.
- [3] K. Hilman, H.-W. Park, Y.-M. Kim, "Using motion-compensated frame rate up conversion for the correction of 3:2 pull down artifacts in video sequences," IEEE Trans. Circuits Syst. Video Technol., vol. 10, no. 6, pp. 869-877, Sept. 2000.
- [4] A.-M. Huang, and T.Q. Nguyen, "A multistage motion vector processing method for motion-compensated frame interpolation," IEEE Trans. Image Processing, vol. 17, no. 5, pp. 694-708, May. 2008.
- [5] Y. Ling, J. Wang, Y. Liu, and W. Zhang, "A novel spatial and temporal correlation integrated based motion-compensated interpolation for frame rate up-conversion," IEEE Trans. Consum. Electron., vol. 54, no. 2, pp. 863-869, May. 2008.
- [6] W. Demin, L. Zhang, and A. Vincent, "Motion-compensated frame rate up-conversion part II: New algorithms for frame interpolation," IEEE Trans. Broadcast., vol. 56, no. 2, pp. 142-149, June. 2010.
- [7] G. Dan, and T. Nguyen, "Motion vector processing for frame rate up conversion," in Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, 2004, pp. 309-312.
- [8] H. Lui, R. Xiong, D. Zhao, S. Ma, and W. Gao, "Multiple hypotheses Bayesian frame rate up-conversion by adaptive fusion of motion-compensated interpolations," IEEE Trans. Circuits Syst. Video Technol., vol. 22, no. 8, pp. 1188-1198, Aug. 2012.
- [9] S. -G. Jeong, C. Lee, and C.-S. Kim, "Motion-compensated frame interpolation based on multihypothesis motion estimation and texture optimization," IEEE Trans. Image Process., vol. 22, no. 11, pp. 4497-4509, Nov. 2013.
- [10] S.-J. Kang, S.-K. Lee, and Y. H. Kim "Motion Compensated Frame Rate Up-Conversion Using Extended Bilateral Motion Estimation," IEEE Tran. Consumer Electronics, vol. 53, no. 4, pp. 1759-1767, Nov. 2007.
- [11] B.-D. Choi, J.-W. Han, C.-S Kim, and S.-J Ko, "Motion-compensated frame interpolation using bilateral motion estimation and adaptive overlapped block motion compensation," IEEE Trans. Circuits and Systems for Video Technology, vol. 17, pp.407-416, Apr. 2007.
- [12] Dang Ngoc Hai Nguyen, Nam-Uk Kim, Yung-Lyul Lee, "Frame Rate Up-Conversion Using Median Filter and Motion Estimation with Occlusion Detection", IEICE Trans. Fundamentals, Vol. E98-A, No. 1, pp. 455-458, Jan 2015.
- [13] Lee, Yen-Lin, "Method and architecture design for motion compensated frame interpolation in high-definition video processing," IEEE Internation Symposium on Circuits and Systems, pp 1633-1636, May 2009
- [14] Ahmed, N.; Natarajan, T.; Rao, K. R. (January 1974), "Discrete Cosine Transform", IEEE Transactions on Computers

[1] C. Cafforio, F. Rocca, and S. Tubaro, "Motion Compensated Image

저 자 소 개



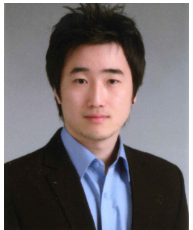
김 남 욱

- 2015년 2월 : 세종대학교 컴퓨터공학과 (학사)
- 2015년 3월 ~ 현재 : 세종대학교 컴퓨터공학과 (석사과정)
- ORCID : <http://orcid.org/0000-0002-2005-2647>
- 주관심분야 : Future Video Coding, HEVC(High Efficiency Video Coding), Image compression, Image processing



전 동 산

- 2002년 2월 : 부산대학교 전자공학과 (학사)
- 2004년 2월 : 한국과학기술원 전기·전자공학과 (석사)
- 2011년 2월 : 한국과학기술원 전기·전자공학과 (박사)
- 2011년 9월 ~ 현재 : 과학기술연합대학원대학교(UST) 이동통신 및 디지털방송공학과 겸임교원 (부교수)
- 2004년 5월 ~ 현재 : 한국전자통신연구원(ETRI) 실감방송미디어연구부 책임연구원
- ORCID : <http://orcid.org/0000-0002-1865-8666>
- 주관심분야 : Future Video Coding Algorithm, HEVC(High Efficiency Video Coding), H.264/AVC, Image processing



이 진 호

- 2007년 2월 : 고려대학교 전자 및 정보공학부 (학사)
- 2009년 2월 : 과학기술연합대학원대학교(UST) 이동통신 및 디지털방송공학과 (석사)
- 2009년 3월 ~ 현재 : 한국전자통신연구원(ETRI) 실감방송미디어연구부 선임연구원
- ORCID : <http://orcid.org/0000-0002-7558-5731>
- 주관심분야 : Future Video Coding, HEVC, H.264/AVC, HDR, WCG



이 영 렬

- 1985년 2월 : 서강대학교 전자공학과 (학사)
- 1987년 2월 : 서강대학교 전자공학과 (석사)
- 1999년 2월 : 한국 과학기술원 전기·전자공학과 (박사)
- 1987년 1월 ~ 2001년 8월 : 삼성전자 R&D 센터 Digital Media Lab. 수석연구원
- 2001년 9월 ~ 현재 : 세종대학교 전자정보대학 컴퓨터공학부 컴퓨터공학과 교수
- ORCID : <http://orcid.org/0000-0003-2709-8282>
- 주관심분야 : Future Video Coding Algorithm, HEVC(High Efficiency Video Coding), Multi-view video coding(MVC), Multimedia systems, H.264/AVC, Image processing, Frame rate up-conversion, Transcoding, Transform design