

선형 움직임 기반 프레임률 향상 기법

김동형

한양여자대학교 컴퓨터정보과

Motion Linearity-based Frame Rate Up Conversion Method

Donghyung Kim

Dept. of Computer Science & Information Systems, Hanyang Women's Univ.,

요약 낮은 프레임률을 가지는 동영상은 높은 프레임률 디스플레이 장치를 통해 재생하는 경우 프레임률 향상기법이 필요하다. 프레임 향상 기법은 연속된 원본 프레임 사이에 프레임을 보간함으로써 생성하며 이러한 프레임 보간방법은 크게 프레임 복사 기법과 움직임벡터 기반의 보간기법으로 나뉜다. 프레임 복사 기법은 단순히 보간하고자 하는 프레임의 이전 또는 이후 프레임을 복사하는 기법으로 매우 낮은 복잡도를 가지지만 저키현상이 발생하는 단점이 있다. 움직임 벡터 추정 및 보상을 통한 프레임 보간기법은 다시 화소단위의 방법과 블록단위의 움직임 추정 방법으로 나눌 수 있다. 화소기반의 경우 영역별 다른 보간 방법을 사용하여 보간을 수행하며 높은 복잡도를 가지는 단점이 있다. 블록기반 방법은 상대적으로 낮은 복잡도를 가지지만 블록킹현상이 발생한다는 단점을 가진다. 제안하는 방법은 블록단위의 움직임 추정 기반 프레임률 향상 기법으로 프레임내 움직임의 선형성을 이용하여 움직임을 추정한다. 이 과정에서 보간하고자 하는 프레임 이전 두 프레임과 이후 한 프레임을 사용하여 움직임 추정 및 보상을 수행함으로써 프레임을 보간한다.

실험결과 제안한 알고리즘은 객관적 화질 측면에서 이전의 다른 방법과 비교하여 높은 성능을 보이고 특히 해상도가 높은 경우 더욱 우수한 성능을 가지는 것으로 나타났으며 주관적 화질 또한 상대적으로 우수한 것을 볼 수 있었다.

Abstract A frame rate up-conversion scheme is needed when moving pictures with a low frame rate is played on appliances with a high frame rate. Frame rate up-conversion methods interpolate the frame with two consecutive frames of the original source. This can be divided into the frame repetition method and motion estimation-based the frame interpolation one. Frame repetition has very low complexity, but it can yield jerky artifacts. The interpolation method based on a motion estimation and compensation can be divided into pixel or block interpolation methods. In the case of pixel interpolation, the interpolated frame was classified into four areas, which were interpolated using different methods. The block interpolation method has relatively low complexity, but it can yield blocking artifacts. The proposed method is the frame rate up-conversion method based on a block motion estimation and compensation using the linearity of motion. This method uses two previous frames and one next frame for motion estimation and compensation.

The simulation results show that the proposed algorithm effectively enhances the objective quality, particularly in a high resolution image. In addition, the proposed method has similar or higher subjective quality than other conventional approaches.

Keywords : Frame rate up conversion (FRUC), Bi-directional motion estimation, Frame Interpolation, Overlapping Boundary Matching Algorithm (OBMC), Motion Linearity-based Frame Rate Up Conversion Method (MLFRUC)

본 논문은 2017년도 1학기 한양여자대학교 교내연구비에 의하여 연구됨

*Corresponding Author : Donghyung Kim(Hanyang Women's Univ.)

Tel: +82-2-2290-2210 email: kimdh@hywoman.ac.kr

Received May 12, 2017

Revised June 1, 2017

Accepted July 7, 2017

Published July 31, 2017

1. 서론

최근 디스플레이 하드웨어 장치의 발달로 프레임률 (frame rate)이 높은 콘텐츠 디스플레이 장치 보급이 확산되고 있다. 반면 아날로그에서 디지털로 넘어오던 시기에 제작된 수많은 콘텐츠는 여전히 이전의 낮은 프레임률 (24Hz, 30Hz)을 가진다. 이러한 콘텐츠를 높은 프레임률을 가지는 디스플레이 장치에서 효과적으로 사용할 수 있는 다양한 프레임률 향상기법(FRUC, frame rate up conversion)이 연구되어 왔다.

가장 쉽게 적용할 수 있는 FRUC 방법은 프레임 보간(interpolation)을 수행하고자 하는 이전 위치와 이후 위치의 프레임들을 그대로 복사하는 방법이다. 하지만 이 경우 시간차이가 존재하는 프레임들을 그대로 복사하여 사용함으로써 저기현상(jerky artifact)이 발생하는 문제점이 있다[1-2]. 또한 유사한 방법으로 이전 또는 이후 프레임의 평균을 이용하여 프레임을 보간하는 방법도 제시되었지만 이 역시 블러링현상(blurring artifact) 또는 고스트 현상(ghost artifact)이 나타나는 단점을 가진다.

이러한 문제를 해결하기 위한 방안으로 움직임 추정에 기반한 FRUC 방법이 다양하게 연구되어 왔다. Moulin 등은 실제 프레임내 객체의 움직임 벡터 추정을 통해 프레임의 보간을 수행하는 방법을 제안하였다[3]. 하지만 이 경우 보간 프레임내에 움직임 벡터의 위치가 중복되어 보간 수행후 다수의 움직임 보상이 이루어지는 중첩(overlapped)영역과 보간이 수행되지 않은 비워(hole)있는 영역이 발생한다는 단점이 있다. Kuo 등은 이렇게 중첩된 화소 영역에서는 미디언 필터를 적용하여 중첩문제를 해결하고 비워있는 영역은 공간영역 내에서 보간을 수행하였다[4]. 하지만 보간 수행 후 비워있는 영역이 다수 포함되어 있는 경우 공간영역에서의 보간에는 한계가 있다는 단점이 있다.

Liu([5])와 Chen([6])은 보간하고자 하는 프레임을 네 개의 영역으로 나누고 각 영역별 다른 보간 방법을 사용하여 화소단위로 프레임을 보간한다. 하지만 영역의 구분과 화소단위의 보간으로 인해 다른 방법들과 비교하여 높은 복잡도를 가진다. 움직임벡터추정에 기반한 방법으로 일정 임계값에 따라 움직임벡터를 추정하는 기법이 Zhai 등([7])에 의해 제안되었으며, Choi 등은 입력영상을 다운샘플 후 낮은 해상도에서 움직임벡터를 추정함으로써 복잡도를 감소시켰다[8]. 특히 Choi는 움직임 보상 과정에서 중첩영역과 비워져있는 영역의 문제를 해결하

기 위한 방법으로 양방향 움직임 추정기법을 제안하였다. 이는 보간 프레임을 기준으로 양방향 움직임 추정 영역을 제한함으로써 보간 프레임에서의 중첩영역과 비워져있는 영역이 발생하지 않는다. 또한 블록 경계면에서의 블록킹현상(blocking artifact)을 줄이기 위해 블록의 외곽화소를 움직임 보상에 사용하는 OBMC(overlapping boundary mapping algorithm)를 적용하였다. 또한 Guo 등은 후처리 기법을 통해 낮은 복잡도를 가지는 움직임 추정기법의 방법을 제안하였다[9].

본 논문은 연속 프레임간에서 발생하는 움직임의 선형성을 활용한 프레임률 향상기법(MLFRUC, Motion Linearity-based Frame Rate Up Conversion)을 제안한다. 이전방법들과는 상이하게 움직임의 선형성을 활용하기 위해 보간하고자 하는 프레임의 이전 두 개의 프레임과 이후 하나의 프레임을 움직임벡터 추정과정에 사용하며 움직임벡터 추정시 비용(cost)값으로 사용되는 SAD(sum of absolute difference)를 재정의한다. 이후 OBMC기반으로 움직임 보상을 통해 프레임을 보간한다.

논문의 구성은 2장에서 기존의 대표적인 프레임률 향상기법들을 소개하고 3장에서 제안한 프레임률 향상기법인 MLFRUC의 요소기술들을 단계별로 기술한다. 4장에서 제안한 알고리즘의 평가를 수행하며 마지막 절에서 결론을 맺는다.

2. 기존의 방법

2.1 프레임 복사 및 평균 기법

프레임률 향상을 위한 가장 간단한 방법은 식 (1)에 나타나 있는 것처럼 보간을 수행하고자 하는 프레임 위치의 이전(forward) 또는 다음(backward)에 위치하는 원본영상의 프레임을 그대로 복사하는 기법이다.

$$\begin{aligned} f_{FFR}(t_i, x, y) &= f(t_p, x, y) \\ f_{BFR}(t_i, x, y) &= f(t_n, x, y) \end{aligned} \quad (1)$$

여기서 $f(t, x, y)$ 는 시간 t 에서 (x, y) 위치의 화소값을 의미하고, t_i, t_p, t_n 은 각각 보간프레임, 이전 원본프레임 및 이후 원본프레임의 시간을 의미하며 이들 사이에는 $t_p < t_i < t_n$ 의 관계가 성립한다. 또한 $f_{FFR}(\cdot)$ 및 $f_{BFR}(\cdot)$ 은 각각 전방향프레임 복사기법(FFR, forward frame repetition) 및 역방향프레임 복사기법(BFR, backward

frame repetition)으로 보간된 프레임은 말한다.

FFR 및 BFR 기법은 단순히 이전 또는 이후의 프레임을 복사함으로써 프레임률을 높이기 때문에 매우 낮은 복잡도를 가지지만 화면내 객체의 움직임이 크거나 카메라 패닝 등의 화면 움직임이 있는 경우 저키 현상 등이 발생한다.

한편 FFR, BFR의 사용시 발생하는 저키 현상을 줄이기 위한 방법으로 이전프레임과 이후프레임의 평균값으로 프레임을 보간하는 양방향프레임평균기법(BA, bi-frame average)은 식 (2)와 같다.

$$f_{BA}(t_i, x, y) = (w_p \times f(t_p, x, y) + w_n \times f(t_n, x, y)) / 2 \quad (2)$$

여기서 w_p 와 w_n 은 두 프레임간 가중평균을 구하기 위한 가중치로 $w_p = (t_n - t_i) / (t_n - t_p)$ 로 표현할 수 있으며 $w_n = (t_i - t_p) / (t_n - t_p)$ 이다.

BA기법 역시 단순히 이전과 이후의 동일위치에서의 화소값들에 대한 가중 평균으로 현재의 프레임을 보간함으로써 저키현상은 다소 줄 수 있으나 블러링 현상(blurring artifact)가 발생하는 단점이 있다.

2.2 움직임 보상기반의 프레임 보간 기법

앞서 기술한 방법들이 가진 저키 현상 또는 블러링 현상의 문제점을 줄이기 위한 대표적인 방법이 프레임간 움직임을 고려하여 보간을 수행하는 움직임 보상프레임 보간 기법(MCFI, motion compensated frame interpolation) 이 있다. MCFI는 프레임을 보간하기 위해서 현재프레임 위치에서 움직임벡터를 추정하고 이를 보간에 사용함으로써 프레임 내에 포함된 다양한 움직임을 반영한다.

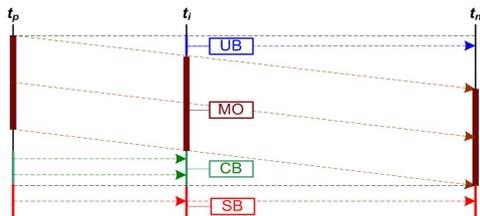


Fig. 1. Pixel-unit based frame interpolation

MCFI는 크게 화소기반으로 움직임보상을 수행하는 방법과 블록기반으로 움직임보상을 수행하는 방법으로 나뉜다. 먼저 화소기반의 움직임 보상 기반의 프레임 보

간기법은 프레임내의 각각의 영역을 MO(moving object), SB(static background), CB(covered background), 그리고 UB(uncovered background)의 영역으로 구분하며 이는 그림 1과 같다.

여기서 MO는 프레임 내에서 움직이고 있는 객체를 의미하고 SB는 프레임마다 동일형태로 존재하는 같은 위치에 존재하는 배경을 의미한다. CB는 MO에 의해 이전프레임에는 존재하지만 현재 위치에서는 가려진 영역을 의미하고 UB는 이전에는 존재하지 않지만 MO의 이동에 따라 나타난 영역을 의미한다. 각 영역에 따라 보간 방법을 달리하는데 영역에 따른 보간에 사용되는 수식은 식 (3)과 같다.

$$f_{MCFI}(t_i, x, y) = \begin{aligned} &MO: w_p \times f(t_p, x - w_p(mv_x), y - w_p(mv_y)) \\ &\quad + w_n \times f(t_n, x + w_p(mv_x), y + w_p(mv_y)) \quad (3) \\ &CB: f_{FFR}(t_i, x, y) \\ &UB: f_{BFR}(t_i, x, y) \\ &SB: f_{BA}(t_i, x, y) \end{aligned}$$

화소기반의 MCFI 기법은 영역별로 최적의 기법을 적용할 수 있어 이상적이지만 영역을 구분하기 위한 높은 복잡도를 가지기 때문에 실시간 적용이 매우 어렵다는 단점이 있다.

화소기반 MCFI의 복잡도 문제를 해결하기 위한 방법이 블록기반의 MCFI. 기법은 8×8 또는 16×16 크기의 블록단위로 움직임을 추정한다. 블록단위 추정에서 나타날 수 있는 가장 큰 문제점은 블록킹현상(blocking artifact)을 발생할 수 있는데 이를 해결하기 위해 움직임 벡터를 추정하는 과정에서 외곽 화소를 함께 사용하는 방법 등이 있다[8].

3. 제안하는 알고리즘

제안하는 MLFRUC(Motion Linearity-based Frame Rate Up Conversion) 기법은 그림 2와 같이 크게 3단계

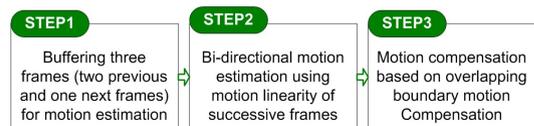


Fig. 2. Block diagram of the proposed algorithm

로 구성된다. 첫 번째 단계는 프레임의 보간에 사용될 참조프레임에 대한 버퍼링과정이다. 이때 보간하고자 하는 프레임위치의 이전 두 프레임 및 이후 한 장의 프레임을 사용한다. 이후 짧은 시간 내 객체는 선형적 움직임을 가질 확률이 높다는 성질을 이용하여 움직임을 추정한다. 즉, 이전 및 이후 프레임 위치에서의 차이값 뿐만 아니라 이전 두 프레임 사이의 차이값 또한 움직임 추정시 사용한다. 이 과정에서 보간하고자 하는 프레임내에 홀(보간되지 않는 부분)이 발생하는 문제를 방지하기 위해 보간 프레임의 블록을 기준으로 하는 양방향 움직임을 사용한다. 마지막 단계에서는 이전 단계에서 추정된 움직임 벡터를 사용하여 보간을 수행하며 이때 블록현상을 최소화하기 위해 외곽화소까지 함께 고려하여 움직임을 보간하는 OBMC(overlapped boundary motion compensation)를 적용한다.

3.1 참조 프레임 버퍼링

제안하는 방법에서는 움직임 추정을 위해 그림 3과 같이 두 장의 이전 프레임(t_{p-1}, t_p)과 한 장의 이후 프레임(t_n)을 사용한다. 여기서 t_{p-1}, t_p, t_n 시간에서의 프레임은 원본 동영상의 프레임을 의미하며 t_i 시간에서의 프레임이 프레임률 향상을 위해 보간을 수행하는 프레임 위치를 말한다.

이때 보간을 수행하기 전 원본 동영상의 프레임률은 식 (4)와 같이 계산할 수 있다.

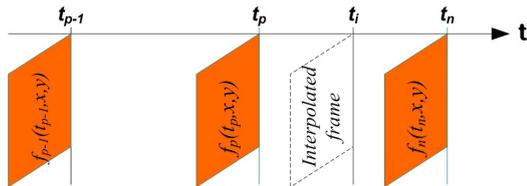


Fig. 3. Buffering structure of three reference frames

$$frame\ rate_{origin} = 1/(t_n - t_p) \quad (4)$$

3.2 선형 움직임 기반 움직임 추정 (Motion Estimation)

프레임간 움직임 추정을 통해 얻어진 움직임벡터를 $\vec{mv}(t_1, t_2) = [mv_x(t_1, t_2), mv_y(t_1, t_2)]$ 라 하자. 이때 $\vec{mv}(t_1, t_2)$ 는 t_1 시간에서의 프레임을 기준으로 t_2 시간에서의 프레임을 참조하여 움직임 벡터를 추정한 것을 의미하며 그림 4는 이를 표현한다.

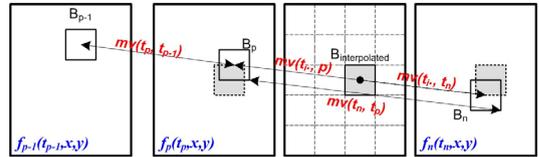


Fig. 4. Bi-directional motion estimation using three reference frames

이 경우 각 참조 프레임 사이에서의 움직임벡터들 간에는 다음의 관계식이 성립하도록 검색영역을 설정한다.

$$\begin{aligned} \vec{mv}(t_i, t_n) &= (w_p/w_n) \times \vec{mv}(t_p, t_i) \\ &= -(w_p/w_n) \times \vec{mv}(t_i, t_p) \\ \vec{mv}(t_p, t_n) &= w_p \times \vec{mv}(t_p, t_i) = \vec{mv}(t_p, t_{p-1}) \end{aligned} \quad (5)$$

식 (5)는 움직임의 선형성을 이용한 것으로 연속적인 프레임간의 움직임벡터 사이에는 많은 경우 선형성을 가지고 있다는 특징을 이용하여 한정된 움직임벡터 검색구간을 설정함으로써 움직임벡터의 잘못된 추정 및 벡터간의 분산을 최소화하는 기능을 수행한다.

움직임벡터의 추정은 기존의 움직임벡터 추정기법에서 사용되는 SAD(sum of absolute difference) 기반의 비용값을 사용하지만 세 개의 프레임을 통해 선형움직임을 예측하는 제안하는 방법에서는 특징 (mv_x, mv_y) 위치에서의 SAD값을 식 (6)을 사용하여 계산한다.

이후 검색영역 내 포함된 모든 (mv_x, mv_y) 에 대한 SAD가 계산되면 최종적인 움직임벡터는 식 (7)을 이용

$$\begin{aligned} SAD(mv_x, mv_y) &= \sum_{x,y}^{block\ size} |f(t_n, x + mv_x(t_i, t_n), y + mv_y(t_i, t_n)) - f(t_p, x + mv_x(t_i, t_p), y + mv_y(t_i, t_p))| \\ &+ \sum_{x,y}^{block\ size} |f(t_p, x + mv_x(t_i, t_p), y + mv_y(t_i, t_p)) \\ &\quad - f(t_{p-1}, x + mv_x(t_i, ref_2) + mv_x(t_p, t_{p-1}), y + mv_y(t_i, t_p) + mv_y(t_p, t_{p-1}))| \end{aligned} \quad (6)$$

하여 도출한다.

$$\begin{aligned}
 \mathbf{mv} &= [mv_x, mv_y], mv_x, mv_y \in [search\ range] \\
 &= \underset{mv_x, mv_y}{\operatorname{minarg}} (SAD(mv_x, mv_y))
 \end{aligned}
 \tag{7}$$

3.3 OBMC 기반의 움직임 보상

마지막 단계에서는 이전단계에서 추정된 움직임벡터를 이용하여 블록단위의 움직임보상(motion compensation)을 수행한다. 하지만 단순히 블록단위의 움직임 보상을 수행하는 경우 블록 경계면에서의 블록킹현상이 심하게 발생하여 현저히 낮은 주관적 화질을 보일 수 있다. 이를 해결하기 위한 방법으로 제안하는 방법에서는 블록의 외곽 화소를 함께 보상하는 OBMC를 사용한다. 블록단위로 움직임 보상이 이루어지고 있기 때문에 중첩된 블록 경계면에서는 두 개 또는 네 개의 움직임 보상이 중첩해서 나타나며 각 영역은 그림 5에 나타나있다. 제안하는 방법에서는 해당 중복영역에 대해서 평균을 수행한다.

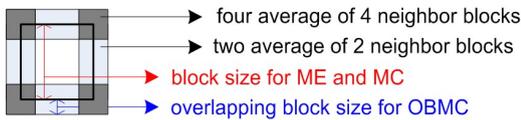


Fig. 5. Overlapped motion compensation areas

4. 실험결과

제안하는 알고리즘의 타당성 검증을 위해서 네 개의 서로 다른 해상도(QCIF (176×144), CIF (352×288), 4CIF (704×576), 720p (1280×720)와 다른 움직임 특성을 가지는 12개의 영상을 대상으로 실험을 수행하였다. 각 테스트영상의 최초 100프레임 중 짝수프레임을 원본 영상으로 하여 알고리즘을 적용하였으며 홀수프레임을 기준으로 보간된 프레임의 객관적 및 주관적 화질을 비교하였다.

Table 1은 각 테스트영상에서의 객관적 화질을 나타내는 PSNR을 비교하고 있다. 첫 번째 방법(frame average)은 보간프레임의 이전 및 이후 프레임의 평균값을 의미하고 biLiteral(forward/ backward/ bi-directional)은 각각 정방향, 역방향, 양방향 움직임 추정을 통해 움직임벡터를 찾고 이를 활용하여 보간을 수행한 방법이다. 결과에서 볼 수 있는 바와 같이 대부분에 실험에서 제안하는 방법이 우수한 PSNR을 가짐을 알 수 있으며 영상의 해상도가 커질수록 향상 정도는 더 커짐을 알 수 있다.

그림 6은 주관적 화질을 비교하고 있으며 주관적 화질 역시 이전의 여러 방법들과 비교하여 유사하거나 우수한 것으로 나타났다. 특히 mobile_cif 영상의 달력부분과 soccer_4cif 영상의 기둥부분에서 그 차이점을 더욱 두드러지게 확인할 수 있다.

Table 1. The comparison of objective video qualities of frame average, biliteral(forward), biliteral(backward), biliteral(bi-directional), and proposed method.

	test sequences	frame Average	biLiteral (forward)	biLiteral (backward)	biLiteral (bi-directional)	proposed method
QCIF	foreman_qcif	34.44	32.70	32.62	37.05	37.21
	grandma_qcif	47.23	43.95	43.74	46.86	47.11
	news_qcif	40.22	38.10	38.11	40.06	40.19
CIF	bus_cif	18.49	20.29	20.37	21.72	23.01
	children_cif	20.74	21.17	21.11	22.10	23.22
	mobile_cif	24.93	22.75	22.76	25.77	27.89
4CIF	city_4cif	22.03	25.00	24.99	25.76	27.38
	harbour_4cif	23.67	24.31	24.30	25.52	24.72
	soccer_4cif	21.70	21.74	21.77	23.17	23.41
720p	bigships_720p	40.02	36.39	36.40	39.92	40.27
	city_corr_720p	27.06	28.68	28.62	29.89	34.67
	night_720p	27.16	26.00	26.00	27.66	27.84



Fig. 6. The comparison of subjective video qualities of frame average, bilateral(forward), bilateral(backward), bilateral(bi-directional), and proposed method to left-to-right

5. 결론

디스플레이 기기 성능의 향상으로 동영상 프레임 해상도는 크게 향상되고 있다. 반면 기존의 다양한 동영상 콘텐츠는 상대적으로 낮은 프레임 율을 가지며 이들 콘텐츠의 프레임해상도를 향상시키기 위해 제안하는 방법은 움직임의 선형성을 이용하여 보간을 수행하는 이전 두 프레임과 이후 한 프레임을 이용하며 움직임 벡터 추정 시 이들 참조프레임을 모두 고려하는 SAD값을 정의

하고 이를 통해 프레임 보간을 수행하였다. 실험결과 기존의 여러 방법들과 비교하여 유사하거나 높은 객관적 화질을 보이는 것을 알 수 있으며 특히 영상의 해상도가 큰 경우 객관적화질 향상은 더욱 두드러졌다. 주관적 화질 역시 이전의 방법들과 비교하여 높은 것으로 나타났다. 향후에는 프레임내의 객체움직임이 선형적인경우와 비선형적인 영역을 분리하여 보다 적응적인 프레임률 향상을 시키는 방안에 대한 연구를 진행할 예정이다.

References

- [1] C. K. Wong, O. C. Au, "Modified motion compensated temporal frame interpolation for very low bitrate video," IEEE Conf. Acoust. Speech, Signal Processing, vol. 4, pp. 2327-2330, May 1996.
DOI: <https://doi.org/10.1109/ICASSP.1996.547748>
- [2] K. Hilman, H.-W. Park, Y.-M. Kim, "Using motion compensated frame-rate conversion for the correction of 3:2 pulldown artifacts in video sequences," IEEE Trans. on CSVT, vol. 10, no. 1, pp. 869-877, Sep. 2000.
DOI: <https://doi.org/10.1109/76.867925>
- [3] P. Moulin, R. Krishnamurthy, J. W. Woods, "Multiscale modeling and estimation of motion fields for video coding," IEEE Trans. Image Processing, vol. 6, no. 12, pp. 1606-1620, Dec. 1997.
DOI: <https://doi.org/10.1109/83.650115>
- [4] T. Y. Kuo, C. -C. I. Kuo, "Motion- compensated interpolation for low-bit-rate video quality enhancement," SPIE Proc. Visual Communications and Image Processing, vol. 3460, pp. 277-288, July 1998.
DOI: <https://doi.org/10.1117/12.323181>
- [5] T. Liu, K. T. Lo, J. Feng, X. Zhang, "Frame interpolation scheme using inertia motion prediction," Signal Processing: Image Communication, vol. 18, no. 3, pp. 221-229, Mar. 2003.
DOI: [https://doi.org/10.1016/S0923-5965\(02\)00141-8](https://doi.org/10.1016/S0923-5965(02)00141-8)
- [6] Y.-K. Chen, A. Vetro, H. Sun, S. Y. Kung, "Frame-rate up-conversion using transmitted true motion vectors," IEEE Multimedia Signal Processing, Second Workshop, pp. 622-627, Dec. 1998.
DOI: <https://doi.org/10.1109/MMSP.1998.739050>
- [7] J. Zhai, K. Yu, J. Li, S. Li, "A Low Complexity Motion Compensated Frame Interpolation Method," IEEE Proc. International Symposium on Circuits and Systems, pp. 4927-4930, May 2005.
DOI: <https://doi.org/10.1109/ISCAS.2005.1465738>
- [8] B.-T. Choi, S. H. Lee, S. J. Ko, "New frame rate up-conversion using bi-directional motion estimation," IEEE Trans. Consumer Electronics, vol. 46, no. 3, Aug. 2000.
DOI: <https://doi.org/10.1109/30.883418>
- [9] Y. Guo, L. Chen, Z. Gao, X. Zhang, "Frame Rate Up-Conversion Method for Video Processing Applications," IEEE Trans. on Broadcasting, vol. 60, no. 4, pp. 659-669, Dec. 2014.
DOI: <https://doi.org/10.1109/TBC.2014.2354951>

김 동 형(Donghyung Kim)

[정회원]



- 1999년 2월 : 충북대학교 전자공학과(공학사)
- 2001년 8월 : 충북대학교 전자공학과(공학석사)
- 2007년 2월 : 한양대학교 전자통신전파공학과(공학박사)
- 2007년 6월 ~ 2008년 8월 : 한국전자통신연구원(ETRI) 선임연구원
- 2008년 9월 ~ 2011년 2월 : 한라대학교 정보통신방송공학부 전임강사
- 2011년 3월 ~ 현재 : 한양여자대학교 컴퓨터정보과 조교수

<관심분야>

영상처리, 멀티미디어통신, 영상압축