

Greedy 기법을 이용한 계층적 객체 기반 움직임 보상

(An Object-Oriented Hierarchical Motion Compensation Technique Using the Greedy Method)

李 濬 瑞 * , 金 麟 澈 ** , 李 商 郁 *

(Jun-Seo Lee, Rin-Chul Kim, and Sang-Uk Lee)

요 약

본 논문에서는, hierarchical grid interpolation(HGI) 기법을 이용한 객체 단위 움직임 보상 기법에 대하여 논한다. HGI 기법에서 콰드트리 분할에 내재된 R-D 관계를 효과적으로 이용하지 못한 점을 참조하여, greedy 기법을 이용한 객체단위 HGI 기법을 제안한다. 제안한 기법에서는 greedy 기법을 사용하여 R-D 측면에서 분할 이득이 최대가 되도록 콰드트리 분할을 수행하고, 분할된 콰드트리 구조하에서 HGI 기법을 적용한다. 모의 실험을 통하여 제안된 기법은 기존의 객체 단위 움직임 보상 기법에 비해 우수한 성능을 보임을 확인한다.

Abstract

In this paper, we describe the object-oriented motion compensation technique using the hierarchical grid interpolation(HGI). By noting that the HGI does not exploit effectively the rate-distortion (R-D) trade-off inherent in the quadtree decomposition, we propose an object-oriented HGI technique employing the greedy method. In the proposed technique, input image is decomposed in a quadtree basis using the greedy method, yielding maximum split gain in the R-D sense. Then, the motion compensation is performed using the HGI technique. The performance of the proposed technique is examined in simulation, and it will be shown that the proposed technique provides better performance than the conventional object-oriented motion compensation techniques.

I. 서 론

움직임 보상은 프레임간의 중복성을 효과적으로 제거하는 기법으로 동영상 부호화에서 핵심적인 기법중의 하나이다. 대표적인 움직임 보상 기법인 블록 정합 기법(block matching algorithm: BMA)^[1]은

현재 프레임을 같은 크기의 블록들로 나누고, 이전 프레임에서 가장 유사한 위치를 탐색하는 기법이다. 그러나 BMA는 블록내 모든 화소가 동일한 병진 운동을 한다고 가정하므로, 확대·축소·회전과 같은 복잡한 움직임과, 블록 내부에 2 이상의 객체가 존재할 때, 움직임을 적절히 기술하지 못하는 단점을 가진다.

최근에는, 복잡한 움직임을 표현하는 한 방법으로, 영상내에 grid point라는 몇개의 점들에서 움직임을 추정한 다음, grid point에서의 움직임을 보간(interpolation)하여 전체 영상의 움직임을 보상하는 CGI(control grid interpolation) 기법^[2, 3]들이 소개되고 있다. CGI 기법에서는 영상 내의 grid point들의 배열 및 보간 방법 등에 따라 다양한 방법들이

* 正會員, 서울大學校 電氣工學部

(School of Electrical Eng., Seoul Nat'l Univ.)

** 正會員, 漢城大學校 情報電算學部

(School of Infor. and Computer Eng., Hansung Univ.)

※ 본 연구는 '97 과학재단 특정기초연구(97-0100-0201-3)의 지원으로 수행되었습니다.

接受日字:1997年11月5日, 수정완료일:1998年6月12日

알려져 있다. 이 중에서 Huang과 Hsu가 제안한 HGI (hierarchical grid interpolation) 기법^[4]은 영상의 국부적인 특성에 따라 grid point들을 적응적으로 설정한 다음, grid point들을 근간으로한 움직임 추정 및 보상을 수행하기 때문에, 움직임 추정 보상 성능이 우수한 것으로 알려져 있다. HGI 기법에서는 콰드트리(quadtree) 구조로 영상을 균일한 움직임 특성을 보이는 블록으로 분할한다. 그런 다음, 각 콰드트리 블록의 꼭지점을 grid point로 하는 CGI 기법을 적용함으로써 움직임을 보상한다. 한편, 콰드트리 분할은 전송률-왜곡치(rate-distortion; R-D) 측면에서 움직임 보상 성능과 밀접한 관계에 있다^[5, 6]. 즉, 콰드트리 분할이 계속되면, 소요 비트량(bit rate)은 증가하지만, 보다 충실한 움직임 보상을 할 수 있다. 만약, 콰드트리 분할을 하지 않으면, 그 반대의 결과를 얻는다. 그러나, HGI 기법은 콰드트리 분할과 움직임 추정이 독립적으로 수행되기 때문에, R-D 측면에서 최적의 움직임 보상을 하지 못한다는 단점이 있다.

본 논문에서는 객체 기반 부호화(object oriented coding)에 HGI 기법을 적용하였을 때 성능을 개선할 수 있는 방법들을 제안한다. 객체 기반 부호화^[7, 8]는 기존의 프레임 단위 부호화와는 달리, 영상을 몇 개의 객체(object)로 나누고, 각 객체를 개별적으로 부호화하는 기법이다.^[4]에서는, 그들이 제안한 HGI 기법을 기본적으로 영상 적응적인 움직임 보상 기법으로 보고, 객체 기반 움직임 보상 기법을 간단히 구현한 것으로 볼 수 있다고 하였다. 그러나, 본 논문에서는 실제 객체 영역에 대해서만 콰드트리 분할에 근거한 움직임 보상을 수행한다. 객체 단위 부호화에서 움직임 보상은 객체 영역에 대해서 BMA를 적용하는 polygon matching^[7]을 통하여 이루어진다. 그리고,^[8]에서는 삼각형 메쉬를 이용한 객체 기반 움직임 보상 기법을 제안하여, 복잡한 움직임도 기술할 수 있도록 하고 있다. 이러한 기법들은 영상의 국부적 움직임 특성을 적절히 반영하지 못하며, R-D 측면에서 효율적인 움직임 보상을 수행하지 못한다는 단점이 있다.

본 논문에서 제안한 기법은 객체 영역에 대하여 콰드트리 분할에 근거한 HGI 기법^[4]을 적용하는 것이다. 먼저, 기존의 HGI 기법에서 왜곡치에 근거하여 움직임 추정을 수행한 것과는 달리, 전송률과 왜곡치를 함께 고려하여 움직임을 추정하는 기법을 제안한다. 그리고, 영상을 균일한 움직임을 가지는 영역으로 적

절히 분할하기 위하여 콰드트리 분할과 움직임 추정을 동시에 수행한다. 즉, 기존의 방법에서 VFD (variance of frame difference)에 근거하여 분할한 것에 반해, 제안하는 기법에서는 콰드트리 분할에 따른 움직임 보상의 R-D 성능을 비교함으로써, 분할된 블록들이 보다 균일한 움직임 특성을 갖도록 한다. R-D 관계를 고려한 콰드트리 분할은 BFOS 기법^[9, 10] 혹은 greedy 기법^[11] 등을 통하여 얻을 수 있다. BFOS 기법은 bottom-up 접근 기법으로, 완전히 분할된 콰드트리로부터 시작하여, R-D 이득이 가장 작은 가지(branch)를 순차적으로 제거함으로써 원하는 분할을 얻는 방식이다. 이에 반해, greedy 기법은 top-down 접근 기법으로, 각 블록을 한번 더 분할함에 따른 R-D 이득을 비교하여, 그 이득이 최대가 되는 블록들을 순차적으로 분할하는 방식이다. BFOS 기법은 R-D 관계가 선형성 및 단조성을 만족할 때 R-D 최적의 분할을 얻을 수 있다^[9]. 그러나, 실제 영상을 콰드트리 분할할 때, 일반적으로 선형성 및 단조성을 만족한다고 보기 힘들므로, 본 논문에서는 계산량이 적은 greedy 기법을 이용하여 콰드트리 분할한다. 다음에는 제안한 기법을 구체적으로 서술하고, 제안한 기법이 기존의 객체 기반 움직임 보상 기법들보다 우수한 성능을 보임을 모의실험으로 입증하겠다.

II. 객체 기반 HGI

Huang과 Hsu^[4]는 콰드트리 구조에 근거한 HGI 기법을 제안하였다. HGI 기법에서는 먼저 콰드트리 구조로 전체 영상을 균일한 움직임 특성을 갖는 가변 크기 블록으로 나눈다. 그런 다음, control point(CP)라고 부르는 블록의 특정 꼭지점들에 대하여 BMA로 최적의 움직임 벡터를 구한다. 각 화소에서의 움직임 벡터는 CP에 대한 움직임 벡터들을 보간하여 얻고, 이때, 각 화소에서의 움직임 보상 오차가 최소가 되도록 CP에 대한 움직임 벡터들을 갱신한다. 본 절에서는 이러한 HGI 기법을 객체 기반 부호화에 적용할 수 있도록 확장시킨 객체 기반 HGI 기법의 움직임 모델과 움직임 추정/보상 기법에 대하여 구체적으로 설명하겠다.

1. 움직임 모델

객체 기반 부호화에 적용하기 위한 객체 기반 HGI 기법은 MPEG-4 검증 모델^[7]의 polygon mat-

ching과 동일하게 객체 영역에 대해서만 움직임을 추정하고 보상한다. 먼저, 그림 1에 보인 바와 같이 객체 영역을 모두 포함하는 일정한 크기의 정방형 블록들을 형성한다. 본 논문에서 이 블록들을 루트 블록이라고 정의한다. 루트 블록의 크기는 미리 정해 놓으며, 그 위치는 객체의 윤곽선 정보로부터 알 수 있다. 각각의 루트 블록은 영상의 특성에 따라 콰드트리 형태로 분할된다.

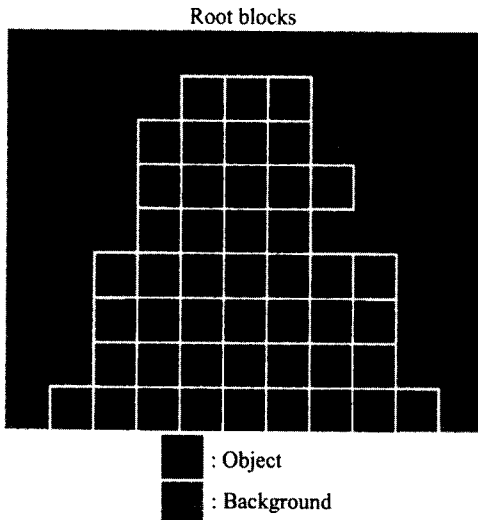


그림 1. 객체와 루트 블록의 예
Fig. 1. Example of the object enclosed by the root block.

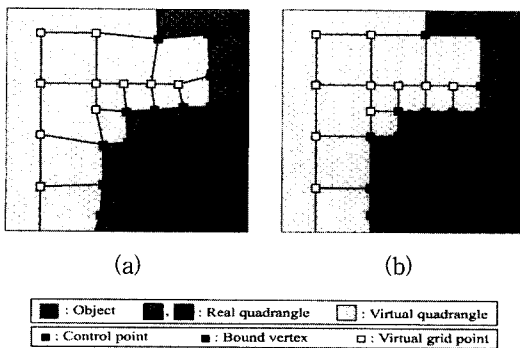


그림 2. Grid point와 사각형
Fig. 2. Examples of grid points and quadrangles.

콰드트리^[5, 6]는 영상을 균일한 특성을 가지는 영역으로 분할하는 방법 중에서 비교적 효율적이면서 간단한 방법으로 알려져 있다. 또한, 콰드트리 정보는 중복성이 거의 없이 가변 길이 부호화할 수 있다는 장점이 있다. 객체 기반 HGI 기법에서는^[4]와 동일하게,

각 블록이 미리 정한 문턱값 이하의 VFD를 갖도록 콰드트리 분할한다. 만약, 분할된 블록이 미리 정한 크기에 도달하면, 분할을 멈춘다.

다음으로, 콰드트리 블록들의 꼭지점, 즉 grid point에 대한 움직임 벡터를 계산한다. 여기서, 움직임 벡터는 현재 프레임의 grid point로부터 과거 프레임에서 가장 유사한 위치를 찾는 후방 예측 방식으로 구한다. 이때, grid point들은 객체와의 관계 및 움직임 벡터 전송 여부에 따라 다음과 같이, virtual grid point (VGP), control point(CP), 그리고 bound vertex (BV) 등 3 종류로 나뉘어진다(그림 2에 예를 나타내었다).

• Virtual grid point(VGP)

VGP는 모든 이웃 사각형들이 객체를 포함하지 않는 grid point를 말한다. 이것은 콰드트리 구조를 표현하기 위해서만 존재하는 것이며, 전송해야 할 움직임 벡터를 할당받지 않는다.

• Control point(CP)

CP는 다른 네개의 grid point와 연결되고 그중 적어도 하나의 grid point가 VGP가 아니거나, 객체 경계를 포함하는 사각형의 바깥 꼭지점인 grid point를 말한다. 3 종류의 grid point 중에서 CP만이 독립적인 움직임 벡터를 가지며, 따라서 이 움직임 벡터를 전송해야 한다. 객체 경계를 포함하는 사각형의 바깥 꼭지점에 움직임 벡터를 할당하는 이유는, 객체 경계의 움직임을 잘 나타내기 위한 것이다.

• Bound vertex(BV)

BV는 VGP가 아닌 세계의 다른 grid point와 T 형태로 연결되어있는 grid point를 말한다. 이러한 grid point는 다른 두개의 grid point와 일직선을 이루도록 그 움직임이 제한된다. 이 grid point의 움직임 벡터는 일직선을 이루는 다른 두개의 grid point의 움직임 벡터를 선형 보간하여 얻는다. 따라서 이 grid point는 전송해야 할 움직임 벡터를 가지지 않는다.

객체 기반 HGI에서는^[4]와는 달리 객체 단위로 움직임을 추정하므로, 분할된 콰드트리 블록 중에서 객체를 포함하지 않는 블록이 발생할 수 있다. 이러한 블록의 꼭지점을 정의하기 위해 VGP를 새로이 정의하였으며, 다른 꼭지점의 분류는^[4]와 동일하다. 여기서, BV는 콰드트리 블록내의 움직임 벡터를 bilinear 보간을 통하여 얻을 수 있도록 하기 위한 구

속 요건이다. 콰드트리 분할된 블록들은 다음과 같이 2 부류로 나누어진다(그림 2에 예를 나타내었다).

• Virtual quadrangle(VQ)

Virtual quadrangle은 객체를 포함하지 않는 블록을 말한다. 즉, 적어도 1개 이상의 grid point가 VGP인 블록이다.

• Real quadrangle(RQ)

Real quadrangle은 객체를 포함하는 블록을 말한다. 즉, 4개의 grid point가 CP 혹은 BV로 정의되는 블록으로, 움직임 추정과 보상을 수행하는 블록이다.

그리고, 주어진 CP의 움직임 벡터의 변화로 인하여 변형되는 주위 사각형들을 DQ (deformable quadrangle)라고 정의한다.

2. 움직임 추정 및 보상

객체 기반 부호화에서 움직임 추정은 CP들에 대하여 수행하고, 이에 따라 RQ의 객체 영역에 대하여 움직임 보상이 이루어진다. RQ의 객체 영역내 각 화소들에 대한 움직임 벡터는 RQ의 grid point에서의 움직임 벡터들을 bilinear 보간함으로써 얻고, 이를 이용하여 움직임 보상을 수행한다.

객체 기반 HGI를 위한 움직임 추정은 [4]의 방법을 객체 영역에 대해서 적용함으로써 수행되는데, 다음과 같은 두 단계로 이루어진다. 첫번째, 콰드트리 분할된 영상에서 CP에 대한 움직임 벡터의 초기 추정치를 CP 근방에서 polygon matching 기법을 적용하여 얻는다. 두번째, 영상내의 모든 CP들에 대하여 CP에 대응되는 DQ들의 움직임 보상 오차를 최소화시킬 수 있는 움직임 벡터를 반복적으로 탐색한다. 두번째 단계를 보다 구체적으로 설명하면 다음과 같다.

먼저, 주어진 CP의 움직임 벡터를 초기 추정치 근방에서 변화시키면서, 이에 의해 영향을 받는 DQ들에 대한 움직임 보상 오차를 구한다. 이때, 움직임 추정은 객체 영역에서만 수행되므로, 객체 영역을 O 라고 정의하면, CP의 움직임 벡터가 DV 일 때 움직임 보상 오차 $OMCE$ 는 다음과 같이 정의된다.

$$OMCE(DV) = \sum_{x_i \in DQ \cap O} N(I_k(X_i) - I_{k-1}(X_i + DV)) \quad (1)$$

여기서, $I_k(X_i)$ 는 k 번째 프레임의 공간 위치 X_i 에서 영상의 밝기를 나타내고, $N(\cdot)$ 은 거리 함수이다.

그리고 식 (1)의 움직임 보상 오차를 최소로 하는 DV 를 현재 CP에 대한 국부 최적 움직임 벡터로 갱신한다. 이와 같은 과정을 모든 CP에 대하여 수행한다. 그러나, RQ들은 1개 이상의 CP 움직임 벡터에 영향을 받게 되므로, 객체 영역의 전체 움직임 보상 오차가 수렴할 때까지 모든 CP들에 대한 움직임을 반복적으로 추정하여야 한다.

III. R-D 관계를 고려한 객체 기반 HGI

전 절에서 설명한 객체 기반 HGI에서는 먼저, VFD의 크기를 참조하여 객체를 콰드트리 분할한 다음, 콰드트리 블록의 꼭지점들 중에서 독립적인 움직임 벡터를 가지는 CP들에 대하여 움직임을 추정한다. 그러나, 콰드트리 분할은 앞에서 언급한 바와 같이 R-D 절충 문제가 발생하지만, 객체 기반 HGI에서는 이를 효과적으로 해결할 수 없다. 본 절에서는, 콰드트리 분할시 발생하는 R-D 절충 문제를 효과적으로 해결하는 한 방법으로서 greedy 기법을 이용하여 R-D 관계를 고려한 HGI 기법을 제안한다. 분할된 콰드트리를 Q 라고 표기하면, R-D 최적의 콰드트리 분할을 통한 움직임 추정은 다음과 같은 조건을 만족하는 콰드트리를 찾는 문제로 귀착된다.

$$\min D(Q) \text{ subject to } R(Q) \leq \hat{R}, \quad (2)$$

여기서 $D(Q)$ 와 $R(Q)$ 는 주어진 콰드트리 Q 에 대한 움직임 보상 오차와 전송률을 각각 나타내고, \hat{R} 은 목표 전송률을 나타낸다. 식 (2)를 만족하는 콰드트리를 얻기 위해, 본 논문에서는 먼저, 루트 블록의 객체 영역에 대하여 CGI 기법 [2]을 적용하여 루트 블록의 모든 grid point들을 CP로 하는 움직임 보상을 수행한다. 그런 다음, 각 블록의 추가적인 콰드트리 분할시 얻어지는 R-D 기율기를 참조하여, 그 기율기가 최대가 되는 블록들을 반복적으로 분할한다. 제안하는 방법을 자세히 설명하면 다음과 같다.

1. 객체 기반 CGI

제안하는 기법은 top-down 방식으로 움직임 추정 및 보상을 수행하게 된다. 먼저, 초기 움직임 추정치를 얻기 위하여, CGI 기법을 객체 영역에 대해서만 적용하는 객체 기반 CGI 기법으로 루트 블록들에 대하여 움직임을 보상한다. 보다 구체적으로 설명하면, 동일한

크기의 정방형 블록인 루트 블록들을 둘러싸는 grid point들은 전 절의 정의에 의하면 모두 CP가 된다. 따라서, 이들 CP들에 대하여 2.2절에서 설명한 기법으로 움직임을 추정한다. 이때 얻어지는 각 grid point에 대한 움직임 벡터를 초기 추정치로 사용한다. 기존의 CGI 기법과 객체 기반 CGI 기법의 다른 점은, 객체 기반 CGI 기법은 객체 영역에 대한 움직임 보상 오차를 최소화시킨다는 것이다. 또한, 2.2절에서 설명한, hexagonal matching^[3]을 확장시킨 기법으로 움직임을 추정하기 때문에 국부적으로 최적인 움직임 벡터를 얻을 수 있다.

2. 콰드트리 분할

제안하는 기법에서는 전 절에서 구한 초기 추정치를 참조하여, greedy 알고리즘을 이용하여 R-D 절충점을 찾는다. 즉, 객체 영역내의 모든 RQ에 대하여 추가적인 콰드트리 분할시 얻어지는 전송률과 왜곡치의 변화량 즉, R-D 곡선의 기울기를 구한다. 그런 다음, 원하는 전송률을 만족할 때까지 최대의 기울기를 가지는 블록들을 반복적으로 분할함으로써 R-D 관계를 고려한 콰드트리 분할을 수행할 수 있다. 제안하는 기법에서의 콰드트리 분할은 기울기 계산, 분할, 기울기 갱신 등의 과정을 원하는 전송률에 이를 때까지 반복하는 것으로 이루어진다. 제안하는 방법을 좀더 자세히 설명하면 다음과 같다.

먼저, 객체 영역을 포함하는 모든 RQ에 대하여 개별적으로 추가적인 분할에 의한 R-D 기울기를 계산한다. 현재 콰드트리 $Q^{(i-1)}$ 의 RQ 들을 한단계 분할함으로써 생기는 콰드트리에 대하여, 분할 이득 $\lambda(i-1, Q)$ 를 다음 식과 같이 구한다.

$$\lambda(i-1, Q) = \frac{\Delta d}{\Delta r} \tag{3}$$

$$= \frac{-(D(Q^{(i-1)})) - D(Q)}{R(Q^{(i-1)}) - R(Q)}, Q \in Q^{(i-1)}$$

여기서 $R(\cdot)$ 과 $D(\cdot)$ 는 각각, 콰드트리에 대한 전송률과 왜곡치를 나타낸다. 그리고 $Q^{(i-1)}$ 은 콰드트리 $Q^{(i-1)}$ 의 각 단말 노드를 한단계 분할함으로써 생기는 콰드트리들의 집합을 나타낸다. 즉, 식 (3)의 기울기는 단말 노드의 한단계 분할에 의한 R-D 측면의 이득을 나타낸다.

콰드트리 분할은 주어진 블록을 같은 크기의 4개의 부분블록(subblock)으로 나누는 것으로, 분할 후 블록의

중심에 새로운 CP가 추가된다. 따라서, 각 RQ에 대하여 분할에 따른 기울기를 계산하기 위해서는 새로운 CP에 대한 움직임을 추정하여야 한다. 또한, 새로운 CP의 움직임 벡터는 인접한 CP들의 움직임 벡터에도 영향을 주게 되므로, 이들에 대한 움직임 벡터들을 갱신하여야 한다. 그러나, 분할되는 블록에서 멀리 떨어져 있는 CP들은 분할에 의한 영향이 적게 된다. 본문에서는 움직임 추정 성능과 계산량의 절충으로, 그림 3과 같이, 분할되는 블록에 인접한 CP들의 움직임 벡터만을 갱신하여 R-D 기울기를 구하였다. 이러한 간략화는 움직임 추정 성능을 약간 저하시키나, 계산량을 대폭적으로 감소시킬 수 있는데, 이러한 영향은 실험 결과에서 검토하겠다. 추가적인 콰드트리 분할을 수행하면, DQ 내의 움직임을 보다 충실히 묘사할 수 있으므로, 왜곡치는 분할 이전에 비해 감소하게 된다. 이때의 왜곡치를 측정함으로써, 추가적인 분할에 따른 왜곡치 변화량 Δd 를 구할 수 있다. 또한, 추가적인 분할에 의한 전송률 증가량 Δr 은 다음의 요소들을 고려함으로써 구할 수 있다.

- 새로 생기는 4개의 부분블록에 대한 콰드트리 정보량
- 블록 중심의 새로운 CP에 할당되는 움직임 벡터의 정보량
- 기존의 grid point 들에 할당된 움직임 벡터들에 의한 정보량 변화

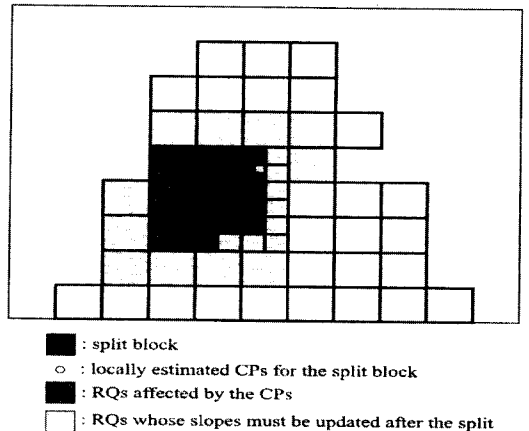


그림 3. 분할 이득 갱신을 위한 국부 영역의 예
Fig. 3. Local area for updating split gain.

이와 같은 방법으로, 각 RQ에서 추가적인 콰드트리 분할시 발생하는 왜곡치 감소량 Δd 와 전송률 증가량

Δr 을 계산한다. 그런 다음, 원하는 전송률을 초과하지 않으면서, R-D 곡선의 기울기 $\lambda(i-1, Q)$ 가 가장 큰 RQ를 선택하여 이를 분할한다. 즉, 현재 콰드트리 $Q^{(i-1)}$ 로부터 다음 콰드트리 $Q^{(i)}$ 를 다음과 같이 순차적으로 얻는다.

$$Q^{(i)} = \arg \max_{Q \in Q(i-1)} \lambda(i-1, Q). \quad (4)$$

선택된 RQ를 분할하면, 이로 인해 주위에 있는 CP의 움직임 벡터가 변하므로, 프레임 내의 모든 RQ의 기울기를 새로 계산해야 한다. 그러나 앞서 설명한 간략화 방법을 사용할 경우 갱신해야 하는 RQ들은, 그림 3과 같이, 움직임 갱신된 CP들이 영향을 미치는 주위 RQ들로 국한됨을 알 수 있다. 이상의 과정을 원하는 전송률을 만족할 때까지 반복함으로써 전송률 및 왜곡치를 모두 고려한 움직임 추정을 수행할 수 있다.

IV. 모의실험

본 절에서는 “Claire”(CL), “Foreman”(FM), “Mother&Daughter”(MD) 등 3개의 MPEG-4 실험 영상을 이용하여, 제안한 기법의 성능을 평가하였다. 실험에 사용한 영상들은 8.33 Hz의 프레임율(즉, 두 프레임씩 건너 뛴)과 176×144 의 크기이고, 객체 영역은 주어진 배경/전경 영상^[13]을 이용하였다. 모든 움직임은 ± 15.5 화소 범위에서 반화소 단위로 추정하였고, 움직임 벡터는 MPEG-4 VM^[7]의 VLC (variable length codeword) 표를 이용하여 가변 길이 부호화하였다.

1. 콰드트리 블록 크기에 따른 성능 분석

우선, 제안하는 기법에 대해 다음과 같은 3가지 종류의 블록 크기에 따른 성능을 평가하였다.

- “32-4” : 32×32 , 16×16 , 8×8 , 4×4
- “32-8” : 32×32 , 16×16 , 8×8
- “16-4” : 16×16 , 8×8 , 4×4

예를 들면, “32-4”의 경우는, 루트 블록의 크기가 32×32 이고, 콰드트리 분할이 진행됨에 따라 16×16 , 8×8 , 4×4 크기의 블록으로 분할되도록 하는 것이다.

그림 4에서는 전형적인 예로서, Claire 및 Foreman 각 4번째 프레임에 대하여 제안한 기법의 움직임 추정 시 R-D 성능을 제시하였다. 그래프 상의 각

점들은 초기 콰드트리부터 시작하여 제안한 방법을 적용하면서 각 분할 단계에서 콰드트리 및 움직임 벡터 정보량과 PSNR 측면에서의 움직임 보상 오차를 나타낸다. 그림 4를 참조하면, 일반적으로 알려진 바와 같이, 정보량이 적은 경우에는 루트 블록의 크기가 클수록 움직임 추정 성능이 우수하며, 정보량이 많은 경우에는 최소 허용 블록의 크기가 작을수록 우수한 성능을 보인다는 것을 알 수 있다. 일반적으로 MPEG-4 표준안에 따라 실험할 경우, 움직임 벡터에 의한 정보량이 프레임당 약 500 ~ 1000 bits 인 것을 참조한다면, 실험 결과 “32-4”의 경우가 가장 적절함을 알 수 있다. 따라서, 이후의 실험에서는 콰드트리 블록 크기는 “32-4”의 경우를 사용하였다.

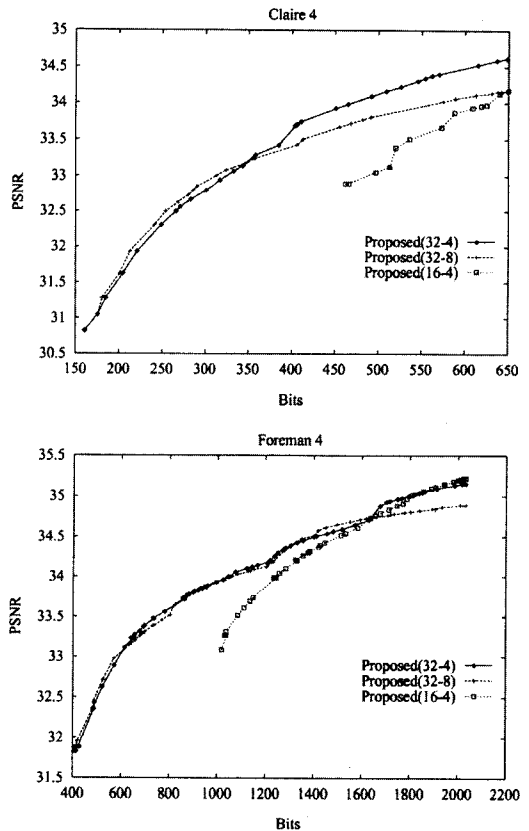


그림 4. 블록 크기에 따른 성능 비교(괄호 안은 블록 크기)

Fig. 4. Performance comparison according to the quadtree block size.

2. 국부 영역을 이용한 분할 이득 갱신 방법의 성능 제안한 기법에서는 계산량과 성능의 절충으로, 그림 3에 제시한 바와 같이 분할되는 RQ 근방의 grid

points 들로 제한되는 국부 영역에 대하여 움직임을 추정하여 쿼드트리 분할 이득을 측정하였다.

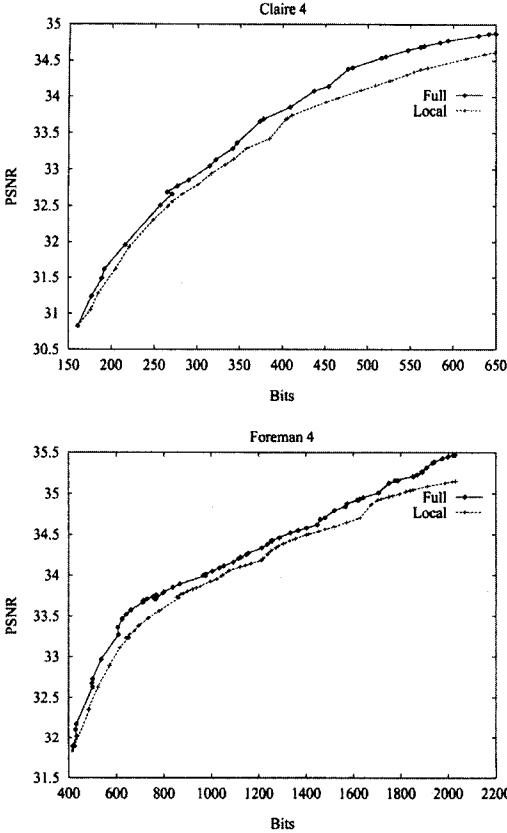


그림 5. 국부 영역을 사용한 분할 이득 갱신에 의한 성능
 Fig. 5. Performance of the proposed technique employing the local area for updating split gain.

그림 5에서는, 전 절에서 사용한 영상들에 대하여, 국부 영역을 이용한 분할 이득 갱신 방법과 전체 영상을 이용한 방법의 성능을 비교하였다. 그리고, 표 1에서는 두 방법들을 HP-735 워크스테이션에서 수행하였을 때 소요 시간을 분 단위로 측정한 결과를 제시하였다. 그림 5를 참조하면, 국부 영역을 이용하여 분할 이득을 갱신하였을 때, 약 0.2~0.4 dB 정도 성능이 저하됨을 알 수 있다. 그러나, 표 1에 제시된 소요 시간을 보면, 국부 영역을 이용한 방법은 계산 시간을 약 16배 이상 단축시킬 수 있음을 알 수 있다. 전체 영상을 이용하여 분할 이득을 갱신하는 방법은 매 단계에서 R-D 측면에서 최적의 움직임 벡터를 추정할 수 있다는 장점이 있다. 그러나, 이러한 방법은 계산

시간이 지나치게 많이 소요될 뿐만 아니라, 국부 최적인 움직임 추정을 수행하게 된다^[12]. 이러한 점을 참조하여, 본 논문에서는 계산량과 움직임 추정 성능의 절충으로 국부 영역만을 이용하여 분할 이득을 갱신하였다.

표 1. 계산 복잡도

Table 1. Computational complexity.

동영상	평균 계산 시간 (단위: 시간:분)			
	Poly	OHGI	제한한 기법	
			국부 움직임	전체 움직임
CL	0:01	1:13	4:12	66:53
FM	0:01	1:18	10:17	132:53
MD	0:01	3:26	13:32	
평균	0:01	1:48	8:49	

3. 동영상 모의실험

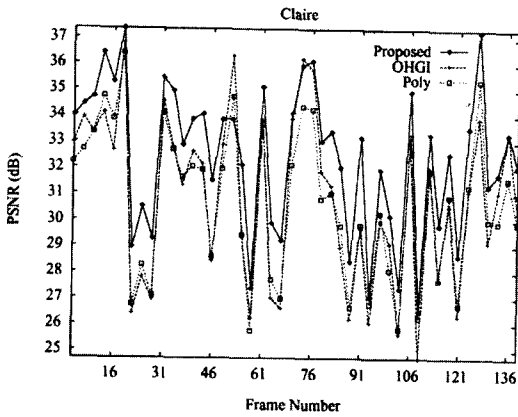
그림 6에서는 3개의 실험 영상(CL, FM, MD)에 대하여 프레임별 예측 오차의 PSNR을 제시하였다. 그리고 각 영상들에 대한 평균 PSNR과 전송률을 표 2에 나타내었다. 비교 대상으로는 MPEG-4 VM^[7]에서 채택하고 있는 기법, 즉 16×16 / 8×8 예측 모드 결정 방법을 polygon matching에 적용한 기법 (Poly)이며, 또 하나는 2장에서 설명한 객체 기반 HGI (OHGI) 기법 등 2 방법을 선택하였다. 각 방법들의 공정한 비교를 위하여, 제안하는 기법과 OHGI에서의 목표 전송률로는, polygon matching시 발생되는 움직임 벡터의 정보량으로 정하였다.

표 2. 평균 성능

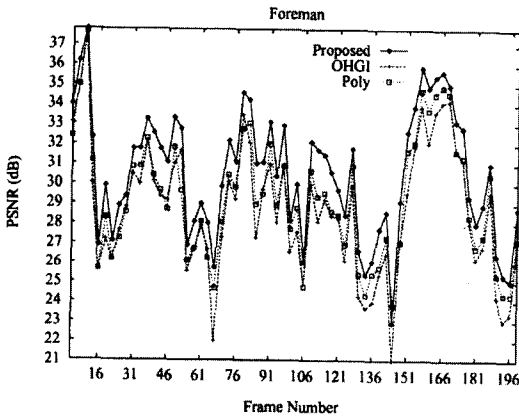
Table 2. Average Performance of 3 different techniques.

동영상	평균 PSNR (dB)		
	Poly	OHGI	제한한 기법
CL	30.66	30.8	32.44
FM	29.25	28.69	30.56
MD	32.11	31.99	33.36

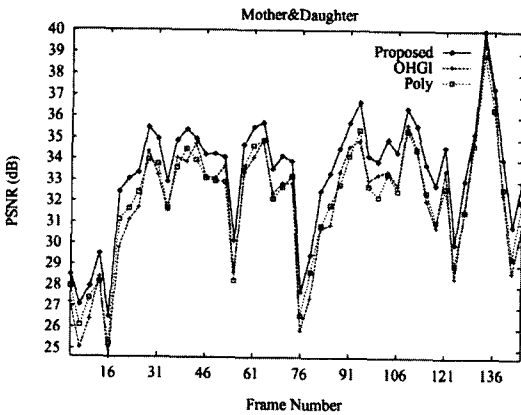
동영상	평균 전송률 (bits/frame)		
	Poly	OHGI	제한한 기법
CL	567	568	548
FM	1061	1071	1043
MD	968	989	954



(a)



(b)

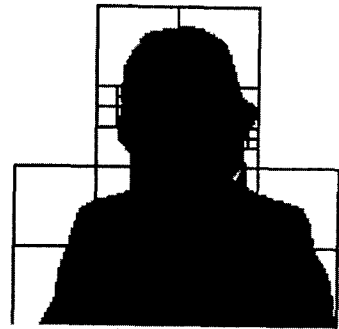


(c)

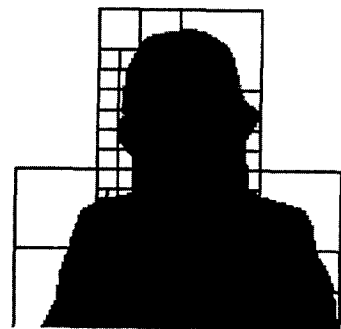
그림 6. 3가지 동영상에 대한 성능
Fig. 6. Performance on the 3 different moving sequences.

여기서 전송률은 움직임 벡터와 쿼드트리 정보를 나타내기 위한 전송률이며, 객체 형태에 대한 정보는 모든 기법들에서 공통으로 전제되어 있으므로 이에 대한

부호화 전송률은 포함하지 않았다. 이 그림과 표를 참조하면, 제안하는 기법이 다른 기법들보다 같은 전송률에서 더 우수한 움직임 보상 성능을 나타내는 것을 알 수 있다. 구체적으로는, "Claire" 영상의 경우에는, 제안하는 기법이 polygon matching 기법보다 1.8 dB, OHGI 기법보다 1.6 dB 우수한 성능을 보이고 있다. 그리고 "Foreman"과 "Mother&Daughter" 영상들에 대해서도 제안한 기법은 기존의 방법들에 비해 약 1.3 ~ 1.8 dB 의 이득이 있음을 알 수 있다. 한편, 어느 동영상이든, OHGI가 polygon matching 기법보다 오히려 낮은 성능을 나타내는 경우가 발생함을 볼 수 있는데, 이것은 OHGI가 쿼드트리 분할 과정에서 R-D 절충 문제를 다루지 않으므로, 쿼드트리 분할이 비효율적으로 이루어짐을 입증하는 것이다.



(a)



(b)

그림 7. "Claire" 4번째 프레임에서의 쿼드트리 분할 움직임 보상 오차 영상
Fig. 7. The quadtree split and motion compensated error image on the 4-th frame of the "Claire".

그림 7에는 분할된 쿼드트리와 움직임 보상 오차의 예를 나타내었다. 이 그림에서, 객체 기반 HGI와 제안한 기법 사이에 많은 차이가 있음을 볼 수 있다. 전자

에서는 VFD의 크기만을 이용하여 분할하는데 반해, 후자에서는 R-D 절충 관계를 고려하여 분할하기 때 문으로 판단된다.

V. 결 론

본 논문에서는 새로운 R-D 관계를 이용한 객체 기 반 HGI 기법을 제안하였다. 제안하는 기법은, greedy 기법을 이용하여 객체 단위로 움직임 보상의 R-D 성 능을 참조하여 콰드트리 분할을 수행하고, 이 때 분할 된 콰드트리 구조하에서 HGI 기법을 적용한다. 모의 실험을 통하여, 제안된 기법은 기존의 객체 단위 움 직 임 보상 기법에 비해 우수한 성능을 보임을 입증하였 다. 본 논문은 움직임 벡터에 의한 정보량 및 움직임 추정 오차로 그 성능을 평가하였다. 앞으로 MPEG-4 등 실제 움직임 보상 부호화기에 적용하였을 때의 성 능 검증이 뒤따른다면, 본 논문에서 제안된 움직임 추 정 기법은 기존의 객체 단위 부호화 기법의 성능을 향 상시킬 수 있는 대안이 될 것으로 판단된다.

참 고 문 헌

- [1] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Commun.*, vol. 29, no. 12, pp. 1799-1808, Dec. 1981.
- [2] G. J. Sullivan and R. L. Baker, "Motion compensation for video compression using control grid interpolation," *Int'l Conf. on ASSP'91*, pp. 2713-2716, May 1991.
- [3] Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformations," *IEEE Trans. CAS for Video Tech.*, vol. 4, no. 3, pp. 339-356, June 1994.
- [4] C.-L. Huang and C.-Y. Hsu, "A new motion compensation method for image sequence coding using hierarchical grid interpolation," *IEEE Trans. CAS for Video Tech.*, vol. 4, no. 1, pp. 42-52, Feb. 1994.
- [5] F. Dufaux and F. Moscheni, "Motion estimation techniques for digital TV: a review and a new contribution," *IEEE Proceeding*, vol. 83, no. 6, pp. 858-876, June 1995.
- [6] J. Lee, "Optimal quadtree variable block size motion estimation," *Int'l Conf. on Image Proces.'95*, pp. 480-483, 1995.
- [7] ISO/IEC JTC1/SC29/WG11 MPEG97/N1796, "MPEG4 Video Verification Model Version 8.0", July 1997.
- [8] P.J.L. van Beek and A.M. Tekalp, "Object-based video coding using forward tracking 2-D mesh layers," *SPIE: Visual Commun. and Image Proces.'97*, vol. 3024, pp. 699-710, San Jose, CA, Feb. 1997.
- [9] P.A. Chou, T. Lookabaugh, and R. M. Gray, "Optimal pruning with applications to tree-structured source coding and modeling," *IEEE Trans. Inform. Theory*, vol. 35, pp. 299-315, March 1989.
- [10] H.B. and W.-Y. Chan, "Rate-constrained hierarchical motion estimation using BFOS tree pruning," *Int'l Conf. on ASSP'96*, pp. 2315-2318, 1996.
- [11] E.A. Riskin and R.M. Gray, "A greedy tree growing algorithm for the design of variable rate vector quantizers," *IEEE Trans. Signal Process.*, November 1991.
- [12] M. Balakrishnan, W.A. Pearlman, and L. Lu, "Variable-rate tree-structured vector quantizers," *IEEE Trans. Inform. Theory*, vol. 41, no. 4, pp. 917-930, July 1995.
- [13] ISO/IEC JTC1/SC29/WG11 MPEG95/50, "HHI Test Sequences for MPEG-4 Tests : Distribution of Segmented Sequences FOREMAN, MOTHER&DAUGHTER, and CLAIRE," 1995.

저 자 소 개



李 濬 瑞(正會員)

1969년 8월 20일생. 1992년 2월 서울대학교 제어계측공학과 학사. 1994년 2월 서울대학교 대학원 제어계측공학과 석사. 1994년 3월 ~ 현재 서울대학교 대학원 제어계측공학과 박사과정 재학중. 주관심분야는 영상

데이터 전송, 디지털 신호 처리



金 鱗 澈(正會員)

1963년 2월 27일생. 1985년 2월 서울대학교 제어계측공학과 학사. 1987년 2월 서울대학교 대학원 제어계측공학과 석사. 1992년 2월 서울대학교 대학원 제어계측공학과 박사. 1992년 ~ 1994년 대우전자 영상연구소 신

임연구원. 1994년 ~ 현재 한성대학교 정보전산학부 조교수. 주관심분야는 영상 처리, 추적기, 영상 데이터 압축, VLSI 신호 처리.

李 商 郁(正會員)

第 35卷 S編 第 2號 參照

현재 서울대학교 전기공학부 교수