
블록 움직임벡터 기반의 움직임 객체 추출

김동욱* · 김호준*

Moving Object Extraction Based on Block Motion Vectors

Dong-wook Kim* · Ho-joon Kim*

이 논문은 정부(교육인적자원부)의 재원으로 학술진흥재단의 지원을 받아 수행된 연구임(D00534)

요 약

움직임 객체의 추출은 비디오 서비스 등에서 주요한 연구목적 중의 하나이다. 본 논문은 블록 움직임 벡터를 이용하여 움직임 객체를 추출하는 새로운 기법을 제시한다. 이를 위하여, 1) 사후 확률 밀도와 Gibbs 랜덤필드의 이용하여 블록 움직임 벡터를 결정하고, 2) 2-D 히스토그램을 바탕으로 전역 움직임을 구하고, 3) 경계 블록 분할 단계를 통해 객체 추출을 달성한다. 제안된 알고리즘은 특히 압축된 비디오 신호의 움직임 객체에 특히 유용하게 이용될 수 있다. 제안된 알고리즘을 여러 가지 영상에 적용한 결과 양호한 결과를 얻을 수 있었다.

ABSTRACT

Moving object extraction is one of key research topics for various video services. In this study, a new moving object extraction algorithm is introduced to extract objects using block motion vectors in video data. To do this, 1) a maximum a posteriori probability and Gibbs random field are used to obtain real block motion vectors, 2) a 2-D histogram technique is used to determine a global motion, 3) additionally, a block segmentation is followed. In the computer simulation results, the proposed technique shows a good performance.

키워드

움직임 객체, 블록벡터, Gibbs 랜덤 필드, 경계블록 분할

I. 서 론

움직이는 객체의 추출 또는 트래킹(tracking)은 컴퓨터 비전(computer vision) 및 영상 인식 분야에서 관심있게 다루어져 왔으며, 최근 들어, 동영상 데이터의 압축 및 검색 등의 분야에서 객체의 추출 및 트래킹 기술이 더욱 필요하게 되었다. 특히, 대부분의 비디오 데이터는 압축된 형태로 전송되거나 저장되며, 비디오 데이터로부터 의미있

는 객체를 추출하고자 할 때, 압축된 정보로부터 원하는 정보를 추출하는 기술이 매우 필요하다. 본 논문에서 다루는 움직임 객체 추출 기술은 비디오 신호로부터 인덱싱 및 검색에 적용하는 것을 목적으로 한다.

오늘날 MPEG-1, 2, 4 등에서 기본적으로 적용하고 있는 기법은 블록기반 움직임 추정 및 예측 기술이다 [1]. 비디오 데이터는 이를 통해 얻어진 블록 움직임 벡터와 예측오차를 부호화하여 전송된다. 따라서, 압축된 비디오

데이터로부터 원하는 객체를 추출하기 위해서는 블록 움직임 벡터로부터 얻는 것이 좋은 방법 중의 하나이다 [2][3][4].

대부분의 객체 분할 알고리즘은 변화영역 검출로부터 출발한다 [5][6]. 비디오 데이터의 임의의 프레임에서 움직이는 객체의 추출은 배경과 움직이는 객체의 분할이며, 대부분의 분할 기법은 인접한 프레임간의 변화된 화소 검출(change detection)에 의존한다. 이러한 기법들은 움직이는 물체들이 보통 연속된 프레임 간에 밝기의 변화를 수반한다는 점을 이용하는데, 밝기의 차를 구하고 이로부터 움직이는 객체 영역을 찾아낸다. 이러한 기법은 화소간의 차 연산에 의존하기 때문에 계산이 간단한 반면, 움직이는 내부는 배경으로 처리되는 등의 단점을 가지고 있다. 또한, 화소 단위로 움직임 필드를 구성하고 이를 바탕으로 움직이는 객체를 구하는 방법은 계산적으로 매우 복잡하며, 지역 최소값 문제에 빠질 위험을 가지고 있다. 객체 추출의 또 다른 방법으로서 영역 분할 기법을 이용한다. 그러나 이러한 영역 분할 기법은 영역 분할 알고리즘의 문턱치 등의 설정에 따라 과분할되거나 잘못 분할되는 결과를 초래하기 쉽다.

본 논문에서는 이러한 문제점들을 개선하기 위하여 블록 움직임 벡터를 이용하여 움직이는 객체를 추출하기 위한 새로운 기법을 제안한다. 움직임 객체의 추출은 배경과 객체의 분리로부터 시작된다. 이를 위해 먼저, 2차원 히스토그램의 분포로부터 통계적으로 전역 움직임이 추정되며, 다음으로 객체의 움직임이 대략적으로 얻어진다. 이를 바탕으로 사후 확률 밀도 최대화를 통해 블록 움직임 벡터를 교정한다. 이러한 결과를 바탕으로 경계 블록의 분할을 이루며, 분할된 영역들에 대해 객체와 배경으로 구분한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 제안된 객체 추출 기법에 대해 설명하고, 제 3장에서 제시된 기법에 대한 모의 실험 결과를 제시하고, 마지막으로 4장에서 결론을 맺는다.

II. 제안된 움직임 객체 추출기법

제안된 전체 알고리즘의 흐름을 그림 1에 나타내었다.

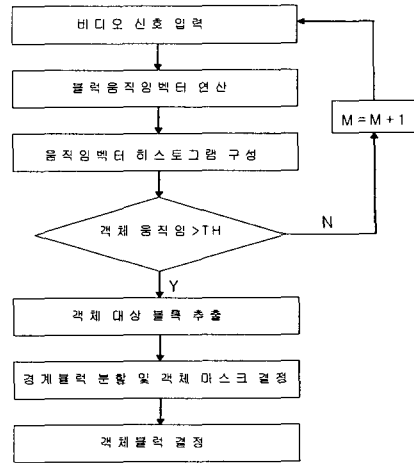


그림 1. 제안된 알고리즘 흐름도
Fig. 1. Block diagram of the proposed algorithm

2.1. 블록 움직임 추정

DFD (displaced frame difference) 기반의 블록기반의 움직임 추정은 영상의 각 블록에 대한 DFD 함수를 최소화 하는 과정이다. 블록 움직임 벡터는 정해진 탐색 범위내에서 블록 정합법(block matching algorithm)에 의해 얻을 수 있다. 블록정합을 달성하는 방법으로서 전역탐색법, 고속블록정합법 등 여러 가지 방법이 있으나, 본 논문에서는 편의상 전역 탐색법에 의해 블록 움직임 벡터를 얻는다. 또한, 블록의 크기는 MPEG-1, 2, 4 등에서 널리 사용되는 16×16 크기를 적용한다. 블록 정합법에 의해 얻어진 벡터는 가현적, 즉 겉보기 움직임 벡터이다. 따라서, 실제 움직임과는 다소 차이를 가지고 있다. 또한, 최소의 DFD 함수를 만족하면 그 블록의 움직임 벡터로 되기 때문에 몇 가지 문제점이 발생한다. 특히, 실제 겉보기 움직임과 매우 다른 움직임 벡터를 얻게 되는 경우이다. 또한 배경과 움직이는 객체의 경계부분에서는 실제와 다른 움직임 벡터를 얻기 쉽다. 따라서, 이러한 문제점을 해결하기 위해서 본 논문의 뒷 부분에서 블록 벡터 교정 단계를 거친다.

2.2. 2-D 히스토그램을 통한 전역 움직임 및 객체 움직임 예측

움직이는 객체 추출을 위해서는 전역 움직임을 추출할 필요가 있다. 전역 움직임은 대부분의 경우 카메라의 이동에 의해 발생하며, 프레임 단위로 변화가 크지 않는 것이 일반적이다.

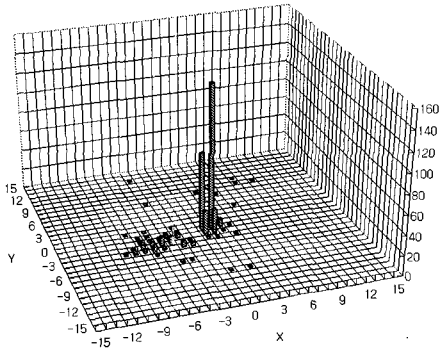


그림 2. fourman 영상에 대한 2-D 히스토그램
Fig. 2. 2-D Histogram(fourman image)

카메라의 움직임은 병진, 회전, 줌(zooming) 등 여러 가지 존재하나 [2], 본 연구에서는 병진이동만을 대상으로 한다. 그림 2는 fourman 비디오 시퀀스 중 한 영상에 대한 블록 움직임 벡터의 2-D 히스토그램을 나타낸 것이다. 그림 2의 x축은 움직임 벡터의 x-방향 성분을, 그리고 y축은 y-방향의 성분을 나타낸 것이며, 막대는 해당 위치의 움직임 벡터의 누적치를 나타낸 것이다. 그림에서 (0,0) 부근에서 가장 높은 블록 움직임 벡터의 누적치를 볼 수 있다. 따라서, 이 값은 배경에 해당되는 부분의 전역 움직임으로 간주할 수 있다. 즉, 전역 움직임이 거의 없음을 의미한다.

이와 같이 본 논문에서는 가장 높은 블록 벡터 누적치를 갖는 벡터를 전역 움직임 성분으로 간주한다. 이 때 얻어진 전역 움직임을 V_g 라고 하자. 만일 $V(x,y)$ 를 블록 좌표가 (x,y) 인 위치의 블록의 움직임 벡터라고 하면, 해당 프레임의 임의의 블록의 움직임으로부터 전역 움직임을 제거하면, 전역 움직임을 제거한 각 블록의 움직임 벡터 $V_o(x,y)$ 을 얻을 수 있다. 즉,

$$V_o(x,y) = V(x,y) - V_g \quad (1)$$

2.3. 객체의 움직임 정도 예측

앞에서 2-D 히스토그램의 분포를 이용하여 전역 움직임을 구했다. 다음으로 히스토그램으로부터 객체의 움직임에 대한 통계적 특성을 얻고자 한다. 이것은 객체 전체의 움직임을 추정하여 이를 바탕으로 정확한 움직임 객체를 추출할 때 사용될 것이다. 객체의 움직임 정도를 추정하는 방법은 다음과 같다.

첫째, 2-D 히스토그램의 분포에서 전역 움직임에 해당되는 벡터군의 값을 영(zero)으로 치환한다.

둘째, 히스토그램에서 전역 움직임 성분을 제거한 후, 남아있는 성분 중 가장 누적 성분이 높은 벡터군을 갖는 벡터 V_p 를 구한다. 즉,

$$V_p = \operatorname{argmax}_{x,y} \sum_{|p,q| < R} (h(x+p,y+q)) \quad (2)$$

여기서 $h(a,b)$ 는 벡터 (a,b) 를 갖는 움직임 벡터의 누적치를 나타내며, R 은 지정된 범위의 크기를 갖는다. 본 연구에서 R 은 3의 범위를 갖는다. 위의 계산을 통해 가장 높은 누적 값을 갖는 벡터를 객체의 움직임 벡터로 간주한다.

2.4. 객체 움직임 적합성

본 단계에서는 전역 움직임과 객체 움직임의 거리를 계산하여 두 벡터간의 거리가 정해진 문턱치보다 클 경우 다음 단계로 이동하고, 그렇지 않을 경우 프레임 간격 M 을 증가시키고 처음으로 돌아가 위의 과정을 반복한다. $V_g = (x_g, y_g)$ 이고, $V_p = (x_p, y_p)$ 라고 할 때,

$$D = \sqrt{(x_g - x_p)^2 + (y_g - y_p)^2} \quad (3)$$

2.5. 사후확률밀도 기반의 벡터 교정

먼저, 본 논문에서는 실제 움직임에 가까운 움직임 벡터를 얻기 위하여 Gibbs 모델링 기법을 적용한다 [7][8].

먼저, 블록벡터를 V 로 놓으면, V 를 추정하기 위하여 사후 확률밀도(posterior probability density) $p(V|I_1, I_2)$ 를 최대화하는 변이 필드 V 를 결정해보자. Bayes 이론에 따르면,

$$p(V|I_1, I_2) = p(I_1, I_2|V) \cdot p(V) \quad (4)$$

이며, 여기서 $p(I_1, I_2|V)$ 는 변이 필드가 주어질 때 영상 I_1 과 I_2 간의 likelihood이며, $p(V)$ 는 변이 필드의 선험률 밀도(prior probability density)이다.

likelihood $p(I_1, I_2|V)$ 는 다음과 같이 주어진다.

$$p(I_1, I_2 | V) \propto \prod_{k=1}^N \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{d^2(k, v(k))}{2\sigma^2}\right\} \quad (5)$$

여기서, $d(k, v(k)) = i_1(k) - i_2(k + v(k))$ 를 의미한다. 여기서, σ^2 는 카메라 잡음 P_C 와 기타 완전하지 않은 변이 보상오차 등에 의해 발생하는 것으로 생각할 수 있다 [7]. 파라미터 σ 는 다음과 같이 주어진다.

$$\sigma^2(V) \approx \text{Max}\left(\frac{1}{K} \sum_{j=1}^K d^2(j, v(j)), P_C\right) = \text{Max}(DFD(V), P_C) \quad (6)$$

식 (5)와 식 (6)을 조합하면,

$$p(I_1, I_2 | V) \propto \frac{1}{\sqrt{2\pi} \cdot \text{Max}(DFD(V), P_C)^K} \quad (7)$$

또한, 벡터간의 상호의존성은 Gibbs/Markov 랜덤필드에 의해 모델링될 수 있다 [8]. 즉,

$$p(V) \propto \exp\left\{-\sum C_i\right\} \quad (8)$$

여기서, C_i 는 클릭 i 에 대한 포텐셜을 의미한다. 여기서 클릭은 주변블록과의 거리에 비례한 값을 가진다. 식 (7)과 식 (8)을 식 (4)에 대입하면,

$$K \cdot \ln(\text{Max}(DFD(V), K \cdot P_C)) + \sum C_i \quad (9)$$

이 최소화될 때 식 (4)의 확률이 최대가 된다. 본 연구에서는, 공간적으로 인접한 클릭뿐만 아니라 시간적으로 인접한 전후 프레임의 클릭을 적용한다. 식 (9)를 바탕으로 임의의 블록에 대한 블록 벡터를 결정한다. 이 때 이용되는 벡터는 공간적으로 8-근방으로 인접한 블록의 벡터 및 시간적으로 바로 인접한 벡터가 이용된다. 이러한 벡터와 DFD를 이용하여 식 (9)를 연산하여 최소의 값을 갖는 벡터를 그 위치의 블록 움직임 벡터로 삼는다.

2.6. 경계 블록 분할과 객체 추출

위 단계에서 얻어진 움직임 객체는 블록기반에 의한 것이다. 따라서, 움직이는 객체와 배경이 함께 포함되어

있는 블록에 대해 후처리가 필요하다. 이를 위해 블록 영역 분할을 행한다. 영역분할 후 인접한 객체 움직임 벡터와 배경 움직임 벡터를 이용하여 최소 오차를 갖는 움직임 벡터를 선택한다. 그 결과에 따라, 다시 배경과 움직이는 객체로 구분을 한다.

본 논문에서 제시하는 알고리즘의 마지막 단계로서 위 단계에서 처리된 결과에 대해 형태학적 필터링(morphological filtering)을 행한다. 필터링의 방법으로서, 개방처리(opening) 후 폐쇄 처리(closing) 연산을 행한다. 개방 처리 연산을 하면 경계면의 거칠게 나온 부분을 제거할 수 있으며, 또한 폐쇄처리 연산을 하면 국부적으로 오목한 부분을 채울 수 있고, 따라서, 전체적으로 경계면을 매끄럽게 처리하게 된다.

III. 모의 실험

본 논문에서 제시된 기법의 검증을 위해 몇 가지 비디오 시퀀스에 대해 실험을 행하였다. 본 실험에서는 블록 움직임 벡터를 얻기 위해 전역 탐색 블록 매칭 알고리즘(full search block matching algorithm)을 이용하여 블록 벡터를 얻었으며, 이 때 사용한 블록의 크기는 16×16 이다. 또한 실험에 사용된 영상은 fourman 영상과 coast guard 영상 2종이다.

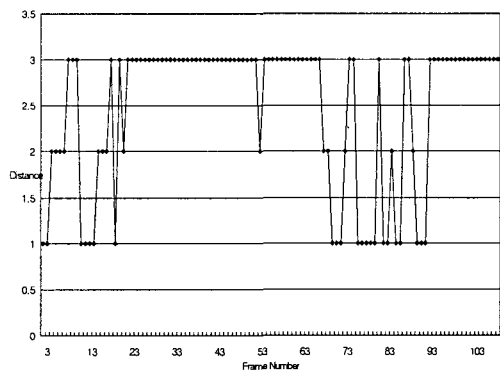
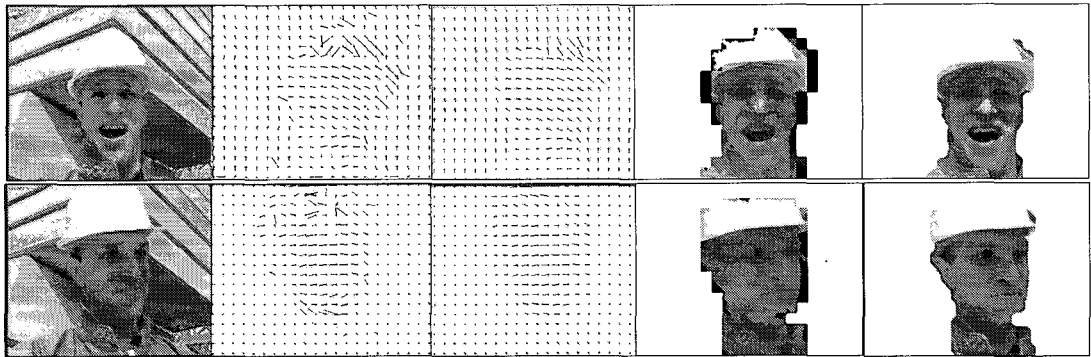


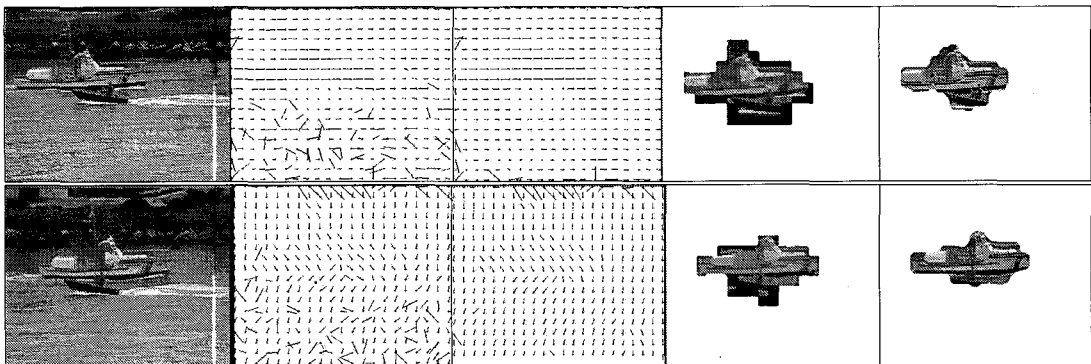
그림 3. 프레임 간격
Fig. 3. Frame intervals

그림 3은 fourman 영상을 대상으로 각 프레임의 현재 프레임과 이전의 기준프레임간 시간간격을 조사하여 나타낸 것이다(그림 1의 M에 해당). 초당 30프레임의 원래



(a)원영상(위-24번째, 아래-120번째래) (b)움직임 벡터(교정전)
 (c) 움직임 벡터(교정 후) (d) 움직임 객체블록 (e)움직임 객체 영역
 (왼쪽부터순서대로)

그림 4. 움직임 객체 영역 추출 1 (fourman)
 Fig. 4 Extraction of moving object regions 1 (fourman)



(a)원영상(위-57번째, 아래-68번째) (b)움직임 벡터(교정전)
 (c) 움직임 벡터(교정 후) (d) 움직임 객체블록 (e)움직임 객체 영역
 (왼쪽부터순서대로)

그림 5. 움직임 객체 영역 추출 2 (coastguard)
 Fig. 5 Extraction of moving object regions 2 (coastguard)

의 영상열에서 프레임간 시간간격이 스킵프레임이 없는 경우가 1이다. 식 (3)의 계산에 의해 배경 움직임과 객체의 움직임 차가 5.0 이상이 되도록 하였는데, 조사대상 108프레임 중 프레임간 간격이 1인 경우가 11회, 2인 경우가 22회, 그리고 나머지는 3인 경우가 대부분이다. 본 영상의 경우 움직임이 크지 않기 때문에 대체적으로 프레임 간격은 3인 경우가 많은 것으로 사료된다.

그림 4와 5는 제안된 알고리즘의 모의실험 결과들을 나타낸 것이다. 각 그림의 (b)는 블록 정합법을 적용하여 얻은 블록 움직임 벡터를 디스플레이한 것이다. 각 그림에서 알 수 있듯이 정합 오류와 복잡한 움직임으로 인하여 움직임 벡터가 여러 방향과 크기를 가지고 있음을 알 수 있다. 각 그림의 (c)는 (b)에 대해 사후 확률 밀도를 최대화하는 움직임 벡터를 얻은 결과이다. (c)의 결과는 (b)에 비해 매우 정돈된 결과를 얻는다.

실험에 사용된 영상들의 전역움직임은 foreman 영상의 경우 각각 24 번째 프레임은 (5,0), 120번째 (0,0), 139번째 프레임은 (-2,-3)이 얻어졌으며, coastguard 영상의 경우 57번째 영상은 (0,-6), 68번째 영상은 (-7,0)이 얻어졌다.

각 그림의 (d)는 배경부분의 전역 움직임을 고려하여 객체를 추출한 결과이다. 물론 이 결과는 블록 단위로 얻어진 것이다. 그림 (d)의 검은색 부분은 추출된 객체 블록에 대해 경계 블록을 분할하고, 분할된 영역 중 배경에 해당되는 부분을 나타낸 것이다. 즉, 이 부분은 움직임 객체 부분에서 제외되는 부분으로서 제거된다. 이러한 결과를 바탕으로 형태학적 필터링을 적용한 후 최종적으로 얻어진 결과가 각 그림의 (e)이다. 각 그림에서 볼 수 있듯이 대부분 잘 검출되었으나, 문제점으로는 움직임으로 인하여 드러난 배경 부분이 제거되지 못하고 남아 있음을 알 수 있다.

IV. 결 론

본 논문에서는 비디오 데이터에서 움직이는 객체를 추출하기 위해 블록 움직임 벡터 기반의 기법을 제시하였다. 사후 확률 밀도를 이용한 블록 벡터 교정기법을 제시하였으며, 특히, 본 알고리즘은 블록 기반으로 동작하기 때문에 압축된 비디오 데이터에 적용하기에 알맞으며, 특히 비디오 검색 분야에서 특히 유용할 것으로 사료된다.

참고문헌

- [1] F. Pereria, T Ebranhimi, *The MPEG-4 Book*, Prentice-Hall PTR, Englewood Cliffs, NJ, 2003.
- [2] L. Favalli, A. Mecocci, and F. Moschetti, "Object tracking for retrieval applications in MPEG-2," *IEEE Trans. Circuits Syst. Video Technology*, vol. 10, pp. 427-432, Apr. 2000.
- [3] D. Zhong and S. F. Chang, "An Integrated approach for content-based video object segmentation and retrieval," *IEEE Trans. Circuits System Video Technology*, vol. 9, pp. 1259-1268, Dec. 1999.
- [4] R. V. Baru, K. R. Ramakrishnan, and S. H. Srinivasan, "Video object segmentation: a compressed domain approach," *IEEE Trans. Circuits System Video Technology*, vol. 14, pp. 462-474, April 2004.
- [5] N. Paragios and G. Tziritas, "Adaptive detection and localization of moving objects in image sequences," *Signal Processing: Image Comm.*, vol. 14, pp. 277-296, Feb. 1999.
- [6] I. Kompatsiaris, G. Mantzaras, and M. G. Strintzis, "Spatiotemporal segmentation and tracking of objects in color image sequence," *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS 2000)*, vol. 5, Geneva, Switzerland, May 2000, pp. 29-32
- [7] C. Stiller, "Motion-estimation for coding of moving video at 8 kbit/s with Gibbs modeled vectorfield smothing," *SPIE vol. 1360 Visual Comm. and Image Proc. '90.*, pp. 468 - 476, 1990.
- [8] Derin, Cole: "Segmentation of textures images using Gibbs Random Fields," *Computer Vision, Graphics and Image Processing 35*, pp. 72-98, 1986.
- [9] P. Salembier and F. Marques, "region-based representations of image and video: segmentaiton tools for multimedia services," *IEEE Trans. Circuits and Video Technol.* vol 9, pp. 1147-1169, Dec. 1999.

저자소개



김 동 욱(Dong-wook Kim)

1987년 2월 성균관대학교 전자공학과(공학사)
1992년 2월 중앙대학교 전자공학과(공학석사)

1996년 8월 중앙대학교 전자공학과(공학박사)
1997년 3월-1998년 2월 청운대학교 전임강사
1998년 3월 - 현재 전주대학교 전기전자정보통신공학부
부교수

※ 관심분야: 영상통신, 신호처리



김 호 준(Ho-joon Kim)

1986년 2월 연세대학교 전기공학과(공학사)
1988년 2월 연세대학교 전기공학과(공학석사)

1998년 8월 연세대학교 전기공학과(공학박사)
1988년 3월-1993년 4월 삼성종합기술원 주임연구원
1999년 2월-2001년 3월 삼성전자 통신연구소 수석연구원
2001년 3월 - 현재 전주대학교 전기전자정보통신공학부
조교수

※ 관심분야: 신호처리, 통신시스템, 이동통신