

움직임 블록간 연결정보를 이용한 움직임 객체의 윤곽선 추출

김진희, 이주호, 정승도, 최병욱
한양대학교 전자통신전파공학과
전화 : 02-2290-0363

Contour Extraction of Moving Object using Connectivity of Motion Block

Kim Jin-hee, Yi Chu-ho, Jeong Seung-do, Choi Byung-uk
Division of Electrical and Computer Engineering, Hanyang University
E-mail : kjh010@ihanyang.ac.kr

Abstract

This paper proposes a new approach to extract contour of moving object from compressed video stream. We segment the area of moving object by using motion vector and extract the motion object block from it. And then we describe the connectivity direction of outline moving block, detect the edge related to connectivity direction in the block and finally obtain the contour by connecting the edges. This can divide the moving object only with motion vector and detect the exact contour on the basis of the edge automatically. Also, we can reduce spending time using motion block and remove the noise with directional edge. The experimental results demonstrate the accurate and effective quality of the proposed method.

1. 서론

오늘날 우리는 다양하고 많은 멀티미디어 매체를 접하고 있다. 또한 인터넷의 확산과 통신망의 고속화로 매체에 대한 접근이 용이해졌고, 실시간으로 방대한 멀티미디어 데이터를 사용할 수 있게 되었다. 그 중

사용자에게 시각적으로 정보를 전달하는 매체의 대표적인 예로 동영상을 들 수 있다. 하지만 동영상내에서 어떤 정보를 얻기 위해선 재생시간동안 그 동영상에 집중을 해야만 한다. 이에 내용기반 동영상 검색 기술이 요구되고 이와 관련된 많은 연구가 진행되고 있다 [1][2][3][4][5].

본 논문은 내용기반 동영상 검색을 목적으로, 움직이는 객체의 윤곽선을 자동으로 추출하는 기법을 제안한다. 본 논문에서는 압축 동영상의 움직임(motion) 벡터만으로 움직이는 객체를 분할하고, 그 객체의 외곽 영역에 포함된 에지를 검출함으로써 움직이는 객체의 구체적인 윤곽선을 추출하였다. 특히 객체에 윤곽선을 포함하는 움직임 블록의 연결 관계를 기술하고 이를 이용하여 블록 연결 방향과 동일한 방향의 에지만을 구하므로 복잡한 배경에 의한 오류를 줄였고 객체의 윤곽선을 효율적으로 추출할 수 있었다.

제안 방법의 전체구조는 압축 동영상에서 움직임 벡터를 추출한 후 보정을 거쳐 움직이는 객체를 분할하는 과정, 객체의 외곽 움직임 블록간의 연결방향을 기술하는 과정, 외곽 움직임 블록의 원영상을 복원한 후 연결 방향과 동일한 방향의 에지를 구하는 과정, 그리고 외곽 움직임 블록의 연결 방향을 고려한 에지 연결 과정으로 이루어진다. 그림 1에서 시스템의 전체구조를 보였다.

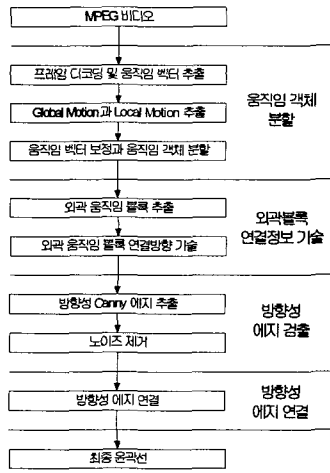


그림 1. 시스템 전체구조

2. 움직이는 객체의 윤곽선 추출

2.1 움직이는 객체 분할

움직이는 객체를 분할하기 위해 본 논문에서는 MPEG-2 압축 동영상의 움직임 벡터를 사용했다. 움직임 벡터는 매크로 블록 단위로 인코딩되며 현재 프레임의 움직임 벡터는 참조 프레임에서 참조해야 할 위치를 나타내준다[6]. 제안 방법에서는 압축 동영상 처리시 가장 문제가 되는 처리시간을 줄이고자 MPEG-2 내의 GOP 단위로 P 프레임에서 움직임 벡터를 추출하였다. 먼저 motion histogram으로부터 global motion을 구하고 그 값을 감산한 local motion에 대해서 임계값 이상의 움직임 값을 유효한 객체의 움직임으로 취했다[7]. 그리고 움직임 벡터의 신뢰성을 높이기 위해 각 블록에 대하여 8-이웃 움직임 블록의 개수를 조사하고, 일정 개수 이상일 경우 그 평균 움직임을 현재 블록의 움직임 값으로 했다. 그 다음 GOP의 첫 번째 P 프레임과 두 번째 P 프레임의 움직임 벡터를 가지고 변형된 Horn-Schunck의 움직임 추정을 사용하여 움직임 벡터를 보정했다. Horn-Schunck의 방법은 연속된 전후 영상의 픽셀 데이터를 사용하지만 본 논문에서는 MPEG-2의 움직임 벡터를 이용하여 움직임을 추정하였다. 그림 2는 움직임 추정을 대상으로 하는 두 개의 P 프레임 움직임 벡터를 나타내는데 현재 프레임을 프레임 1, 이전 프레임을 프레임 2 라고 하면 Horn-Schunck의 편미분 근사화 값을 움직임 벡터에 맞도록 식 1과 같이 변형하여 두 단계에 걸친 근사화 과정으로 보정하고 나머지 과정은 Horn-Schunck의 방법을 따른다[8]. 그림 3은

보정 전과 보정 후의 움직임 벡터를 보여주고 있다.

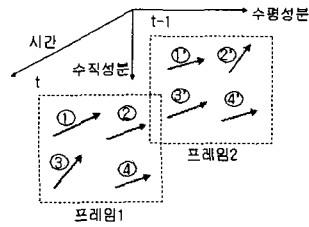


그림 2. 두 프레임의 움직임 벡터

$$\frac{\partial s_c(x, y, t)}{\partial x} \approx \frac{1}{4} (\textcircled{2}_x - \textcircled{1}_x + \textcircled{4}_x - \textcircled{3}_x + \textcircled{2}'_x - \textcircled{1}'_x + \textcircled{4}'_x - \textcircled{3}'_x)$$

$$\frac{\partial s_c(x, y, t)}{\partial y} \approx \frac{1}{4} (\textcircled{2}_y - \textcircled{1}_y + \textcircled{4}_y - \textcircled{3}_y + \textcircled{2}'_y - \textcircled{1}'_y + \textcircled{4}'_y - \textcircled{3}'_y)$$

$$\frac{\partial s_c(x, y, t)}{\partial t} \approx \frac{1}{4} (\textcircled{1}_x - \textcircled{1}'_x + \textcircled{2}_x - \textcircled{2}'_x + \textcircled{1}_y - \textcircled{1}'_y + \textcircled{2}_y - \textcircled{2}'_y)$$

식 1. 변형된 Horn-Schunck의 움직임 추정

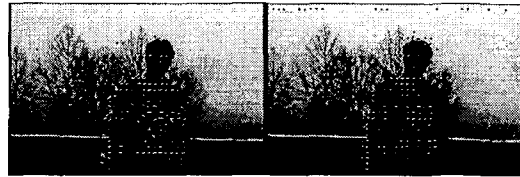


그림 3. 움직임 벡터 보정 결과

이렇게 보정된 움직임 벡터를 가지고 임계값 이상의 움직임 값을 갖는 매크로 블록을 움직이는 객체로 분할한다.

2.2 외곽 움직임 블록 연결정보 기술

외곽 움직임 블록의 연결정보를 기술하기 위해서는 먼저 분할된 객체의 외곽 움직임 블록이 결정되어야 한다. 이 과정을 위해 모폴로지 기법 중 열림(openning)을 사용하여 움직임 블록 오류를 줄이고 침식(erosion)한 움직임 블록과의 차를 구해서 외곽 움직임 블록을 구한다[9]. 그 후 인접한 8-이웃 블록의 연결방향을 조사해서 그림 4와 같이 네 개의 값으로 비트를 설정한다.

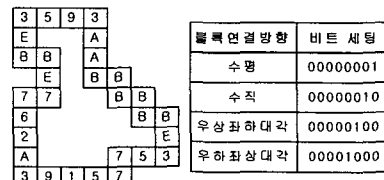


그림 4. 외곽 블록의 연결방향 설정

인접한 블록의 수와 방향이 두 개 이상 일때는 비트 조합을 통하여 복수의 블록 연결 방향 설정이 가능하도록 하였다. 기술된 정보는 해당 블록내의 에지를 구할 때 기술된 방향의 에지만을 검출하는데 사용되고 에지 연결시에도 탐색범위를 제한하기 위해 활용된다. 그림 5는 보정된 움직임 벡터로부터 외곽 움직임 블록을 추출한 결과이다.



그림 5. 외곽 움직임 블록 추출 결과

2.3 외곽 움직임 블록내 방향성 에지 검출

외곽 움직임 블록내 방향성 에지를 구하기 위해서 Canny 에지 검출기를 사용했으며, Nonmaximum suppression 과정에서 외곽 블록 연결 방향을 고려하여 해당 방향에 대해서만 에지를 검출하게 된다.

본 논문에서는 검출한 에지의 기울기가 표 1 과 같은 범위를 가질 때 네 개의 에지의 방향을 설정하고 외곽 블록에 설정된 방향과 일치하는 에지만 검출하였다. 그리고 외곽 블록이 여러 개의 방향성을 가질 때, 설정된 방향과 일치하는 에지는 모두 유효하게 된다.

표 1. 에지의 방향 설정 (θ : Degree)

에지의 기울기 범위	에지의 방향 설정
$-30 \leq \theta < 30$	수평(0x01)
$-90 \leq \theta < -60$ $60 \leq \theta \leq 90$	수직(0x02)
$30 \leq \theta < 60$	우상좌하대각(0x04)
$-60 \leq \theta < -30$	우하좌상대각(0x08)

* \arctan 함수의 각 범위 : $-90 < \theta < 90$

그러나 외곽 블록과 동일 방향성을 가지는 에지가 배경에도 존재하고, 이러한 에지들도 검출되기 때문에 후처리 과정을 통해 배경에서 추출되는 에지 성분을 제거해야 한다. 따라서 해당 블록에서 영역별 프레임 차분치를 구하고 유효한 차분치 결과에 존재하는 방향성 에지만을 구한다[4]. 이 때 영역별 프레임 차분치 내에 에지가 존재하기 위해서는 두 P 프레임 사이의

B 프레임에서 에지를 구하는 것이 P 프레임에서 보다 효율적이다. 잔존하는 노이즈는 8-이웃 픽셀을 조사하여 Connectivity가 2 미만인 픽셀을 삭제함으로써 제거할 수 있다. 그림 6에서 외곽 블록내의 방향성 에지를 추출한 결과와 후처리로 노이즈를 제거한 결과를 보였다.

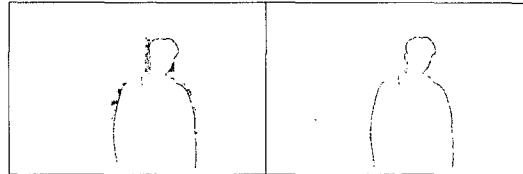


그림 6. 외곽 블록내 방향성 에지 추출과 후처리 결과

2.4 방향성 에지 연결

그림 6의 결과에서 보여지듯이, 외곽 움직임 블록이 움직이는 객체의 외곽선을 포함하고 있지 않는 경우, 또는 에지 미검출 등의 오류에 의해서 방향성 에지가 존재하지 않는 경우에 에지 성분들은 완전한 윤곽선을 이루지 못한다. 따라서 완전한 윤곽선을 추출하기 위해서는 에지 연결 과정이 필요하다. 이 때, 외곽 블록의 연결 방향을 이용하여 탐색 방향을 제한하고 제한된 범위내에서 발견한 에지와 연결한다.

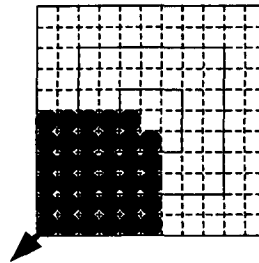


그림 7. 에지 탐색 방향

그림 7은 블록의 방향이 우상좌하대각 방향일 때의 에지를 탐색하는 방향에 대한 예이다. 3x3 크기의 마스크에서 11x11 크기의 마스크에 이르기까지 5단계 (3,5,7,9,11)로 마스크의 크기를 확장하면서 블록의 설정된 방향의 인접 픽셀에서만 에지를 탐색한다. 여러 방향이 설정된 블록에 대해서는 대각성분, 수평, 수직 순으로 우선순위를 두어 각 방향으로 마스크를 확장해서 탐색하는 과정을 에지를 찾을 때까지 반복하고 5단계인 11x11 크기의 마스크에 이르기까지 찾지 못하면 에러로 처리하고 다음 연결 시작점을 찾는다. 연결할 두 에지점을 찾으면 두 점을 지나가는 직선의 방향으로

로 윤곽선을 근사화 한다. 이러한 방법은 에지 방향의 연속성에 기인한 에지 연결 방법으로써 실제 윤곽선과 거의 동일한 결과를 보여준다.

3. 실험 결과

그림 8은 실험에 사용한 MPEG-2 비디오 시퀀스의 원영상과 최종 움직이는 객체의 윤곽선을 추출한 결과이다. 결과에서 보이는 것처럼 제안 방법은 이미 압축된 동영상의 움직임 벡터 정보만으로 객체를 분할하고 방향성 에지 검출을 통해 노이즈를 최소화하면서 움직이는 객체의 윤곽선을 구체적으로 추출할 수 있다. 그러나 제안한 방법의 실험이 MPEG-2에서 기술된 움직임 벡터를 기초로 하므로 그림 9의 그림자의 경우처럼 외곽 움직임 블록이 움직이는 객체의 윤곽을 포함하고 있지 않거나 움직임 벡터 인코딩 자체가 잘못된 경우는 한계가 있음이 나타났다. 특히 빠르게 움직이는 객체의 움직임 벡터의 경우 오류가 심한데 이것은 MPEG-2의 움직임 벡터 인코딩이 정밀하지 않고 경우에 따라 인트라 코딩을 해서 움직임 벡터값이 없기 때문이다[6].

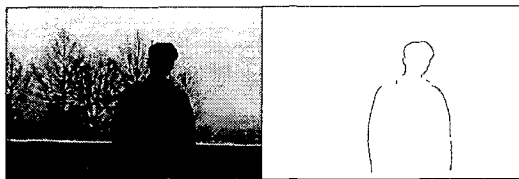


그림 8. 최종 윤곽선 추출 결과

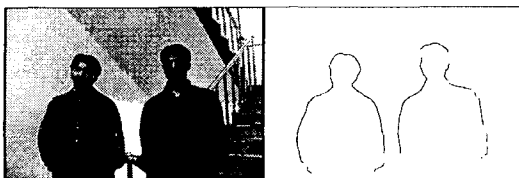
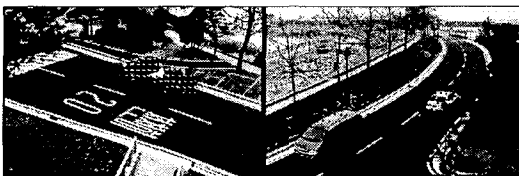


그림 9. 움직임 벡터의 오류로 인해 윤곽선 추출에 실패한 경우



4. 결론

본 논문에서는 압축 비디오 스트림으로부터 움직이는 객체의 윤곽선을 자동으로 추출하는 방법을 제안했다. 제안한 방법은 이미 압축된 동영상의 움직임 벡터만으로 움직이는 객체를 분할하고 객체의 외곽 움직임 블록에서만 처리함으로써 처리시간을 줄였고, 에지 기반으로 윤곽선을 추출함으로써 객체의 구체적인 형태를 추출할 수 있다. 이 과정에서 에지의 기울기 성분 중 블록의 연결 방향에 일치하는 에지만을 사용하여 배경 노이즈의 영향을 최소화 하였다. 추출된 윤곽선은 내용기반 동영상 검색의 중요 정보로 활용할 수 있으리라 예상된다. 실제 내용기반 동영상 검색에서 활용되기 위해서는 추출한 윤곽선의 형태 인식과 윤곽선과 움직임에 대한 샷 단위의 인덱싱 연구가 진행되어야 할 것이다.

참고문헌

- [1] A. Murat Tekalp, "Handbook of Image & Video Processing : Video Segmentation", Academic Press, pp383-400, 1995.
- [2] Datchakorn Tancharoen, "Object Segmentation Based on Multiple Feature for Low Bit Rate Video Coding", Proceedings of ICSP2000, 2000.
- [3] Lili Qiu, "Contour Extraction of Moving Objects", Pattern Recognition vol.2, pp1427-1432, 1998.
- [4] P.K.Rhee, "Boundary Extraction of Moving Objects From Image Sequence", TENCON 99 vol.1, pp621-624, 1999.
- [5] M. P. Dubuisson, A. K. Jain, "Contour Extration of Moving Objects in Complex Outdoor Scenes", International Journal of Computer Vision 14, pp83-104, 1995.
- [6] ISO/IEC JTC1/SC29/WG11 13818-2, Mar 1995
- [7] Janusz Konrad, "Handbook of Image & Video Processing : Motion Detection and Estimation", Academic Press, pp207-227, 2000.
- [8] Henning Bassmann and Philipp W. Besslich, "Ad Oculos Digital Image Processing", International Thomson Publishing, 1995.
- [9] J. R. Parker, "Algorithm for Image Processing and Computer Vision", John Wiley & sons, Inc., 1997.