방송용 스포츠 경기 비디오에서 제스처의 자동 추출

노명철, 이성환
고려대학교 정보통신대학 인공시각연구센터/컴퓨터학과
{mcroh, swlee}@image.korea.ac.kr

Automatic Spotting of Gestures in Broadcast Sports Videos

Myung-Cheol Roh and Seong-Whan Lee
Dept. of Computer Science and Engineering, Korea University

요 약

비디오 데이터 분석은 감시, 검색, 스포츠 경기 자동 요약 등 많은 분야에서 사용되는 기술이다. 그러나 감시 카메라나 스포츠 경기 비디오와 같이 사람의 영역이 저해상도인 환경에서는 포즈 추정, 모델과의 매칭이 어렵기 때문에 제스처 인식 연구는 많이 이루어지고 있지 못하다. 본 논문에서는 카메라가 Pan/Tilt/Zoom 동작을 하고 사람이 빠르게 움직이는 방송용 테니스 비디오에서, 사람을 추출하고, Curvature Scale Space를 기반으로 한 특징을 추출하여 학습된 포즈 모델과 매칭하는 방법과, 차원의 축소를 통해 일련의 포즈들을 학습된 제스처와 매칭하는 방법을 제안한다. 50개의 방송용 테니스 경기 비디오 장면에 대하여 서브 제스처 추출을 수행한 결과, 서브 포즈에 대하여 모델과 매칭이 잘 되고, 매칭이 되지 않는 포즈를 포함하는 시퀀스에 대해서도 강인한

1. 서 론

고속의 디지털 카메라와 비디오 처리 기술의 발달로 인해 사람들의 관심이 비디오 자동 분석의 여러 가지 분야에 모아지고 있다. 대표적인 예로, 감시 비디오 분석, 비디오 검색, 스포츠 경기 자동 분석 등을 들 수 있다. 특히 스포츠 비디오 자동 분석은 스포츠의 경기 개요 자동 생성, 하이라이트 추출, 승리 패턴 분석, 컴퓨터 그래픽을 통한 재생, 광고 삽입 등의 광범위한 응용분야를 가지고 있고, 기존의 연구의 응용 분야로 테니스, 야구, 축구 등에 적용된 사람의 추적, 볼의 추적을 통한 경기 패턴 추출, 임의로 촬영된 테니스 경기 영상에서의 선수의 스트로크 인식 등이 있다[1,2]. 대부분의 기존 연구는 공의 추적을 이용하여 경기 분석을 시도하거나 특수한 환경에서 촬영된 비디오 영상을 이용하여 사람의 제스처 인식을 통한 연구들로, 축구공, 야구공 추적 등을 이용하여 사람의 제스처를 간접적으로 인식하거나 고해상도 환경에서 촬영된 영상을 이용하여 주어진 모델과 유사한 프레임을 선택함으로 제스처를 인식하는 방법을 이용하고 있다. 이와 같은 연구는 방송용 스포츠 비디오와 같이 사람의 영역이 저해상도이고 카메라와 사람이 동시에 움직이는 환경으로의 적용이 어렵다. 효율적으로 객체를 추출하고 제스처를 인식하기 위해서 본 논문에서는 노이즈에 강인한 윤곽선의 특징점 추출 방법과 차원 축소를 통한 제스처의 매칭을 방법을 제안한다.

2. 객체 추출

방송용 테니스 비디오에서 카메라는 동적 카메라로 Pan/Tilt/Zoom을 수행함으로써 일반적으로 많이 사용되는 고정 카메라에서의 객체 추출은 불가능하다. 그러므로 모자이킹(Mosaicing)을 이용하여 카메라의 움직임이 배제된 배경을 생성하고, 생성된 배경을 이용하여 테니스 선수, 볼과 같이 움직이는 객체를 추출한다.

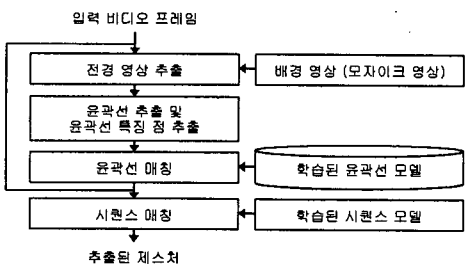
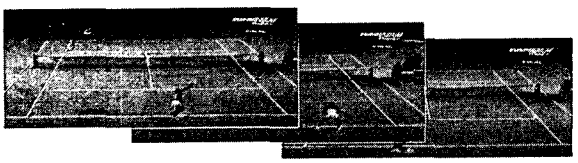


그림 1. 테니스 비디오의 서브 제스처 추출을 위한 흐름도

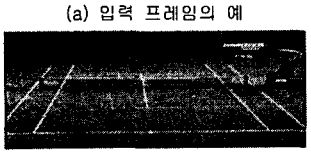
모자이킹 기법은 두개 이상의 영상 간의 호모그래피(Homorgraphy)를 이용하여 하나의 결합된 영상을 만드는 것으로 항공사진 생성, 파노라마 영상 생성, 렌즈 보정, 저해상도 복원 등에서 널리 쓰인다. 모자이킹 기법으로는 인접 프레임간의 대응 관계를 이용하여 정렬하는 지역 정렬(Local Alignment) 방법과 비디오 전체의 프레임들의 대응관계를 동시에 계산하는 전역 정렬(Global Alignment)방법이 있다. 지역 정렬은 인접 프레임간의 호모그래피를 이용하므로 처리해야할 프레임의 수가 늘어나면 이전의 프레임과의 에러가 누적되고, 전역 정렬은 프레임의 수에 따라서 연산 속도가 크게 늘어나는 단점을 가지고 있다. 이를 해결하기 위해서 Mosaic-to-Frame의 방법을 이용하여 배경을 생성하였다.

각 프레임들을 하나의 영상으로 구성할 때, 미디언(Median) 연산을 하여 픽셀의 값을 결정함으로써, 공과 선수와 같이 움직임이 있는 객체들의 픽셀 값을 배제한 배경 영상을 얻을 수 있다. 이렇게 얻어진 배경 영상과 모자이킹 영상에 투영된 각 프레임간의 차이를 이용하여 전경 영상(선수, 볼)을 쉽게 얻을 수 있고, 얻어진 전경

영상에 간단한 영상 처리를 통하여 선수의 윤곽선을 추출하였다. 그림 2은 입력 프레임들의 예와 모자이킹을 이용하여 생성된 배경 영상을 보여준다.



(a) 입력 프레임의 예



(b) 생성된 배경 영상

그림 2. 모자이킹을 이용하여 생성된 배경

3. 윤곽선 매칭

모델의 포즈를 추정하기 위하여 학습된 포즈와 입력된 포즈를 윤곽선 정보를 이용하여 매칭을 수행한다. 윤곽선 매칭을 수행하기 위하여서 본 논문에서는 Curvature Scale Space 영상을 이용한 특징 점 추출 방법을 제안한다[3]. Curvature Scale Space Image는 노이즈에 강인한 윤곽선 표현 방법으로 MPEG-7의 표준화 기술이다.

$$r(u) = (x(u), y(u)) \tag{1}$$

$$\Gamma_\sigma = \{(\chi(u,\sigma), \psi(u,\sigma) \mid u \in [0,1]\} \tag{2}$$

$$\text{where } \chi(u,\sigma) = x(u) \otimes g(u,\sigma)$$

$$\psi(u,\sigma) = y(u) \otimes g(u,\sigma)$$

곡선은 수식(1)과 같이 매개 벡터 수식 (Parametric vector representation)으로 표현되어 질 수 있고, 가우시안 함수에 의해서 전개된 곡선 버전(Evolved version of curve)은 수식(2)로 정의 된다. $g(u,\sigma)$는 가우시안 함수를 나타낸다. 위의 수식(1)과 (2)를 이용하여 곡률(Curvature)을 구하면 수식 (3)과 같다.

$$\kappa(u,\sigma) = \frac{\chi_u(u,\sigma)\psi_{uu}(u,\sigma) - \chi_{uu}(u,\sigma)\psi_u(u,\sigma)}{(\chi_u(u,\sigma)^2 + \psi_u(u,\sigma)^2)^{3/2}} \tag{3}$$

$$\text{where } \chi_u(u,\sigma) = \frac{\partial}{\partial u}(x(u) \otimes g(u,\sigma)) = x(u) \otimes g_u(u,\sigma)$$

$$\chi_{uu}(u,\sigma) = \frac{\partial^2}{\partial u^2}(x(u) \otimes g(u,\sigma)) = x(u) \otimes g_{uu}(u,\sigma)$$

$$\psi_u(u,\sigma) = y(u) \otimes g_u(u,\sigma)$$

$$\psi_{uu}(u,\sigma) = y(u) \otimes g_{uu}(u,\sigma)$$

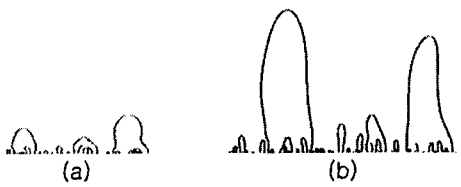
$$\kappa(u,\sigma) = 0 \tag{4}$$

수식(4)를 만족하는 해를 u와 σ 공간에 나타내면 그림 3와 같고 이를 Curvature Scale Space(CSS)표현이라고 한다. CSS를 이용한 매칭은 노이즈에 강인하고, 여러 가지 변환에 강인하지만, 그림 4의 오른쪽 그림의 윤곽선과 같이 잘못된 영역을 포함하게 되면 좋은 매칭을 얻을 수 없게 된다. 그림 3는 그림 4의 오른쪽 그림에서 추출된 윤관선과 해당하는 모델의 윤곽선의 CSS 표현으로 각각 나타낸 영상으로, 상이한 형태의 모양을 보여준다.

$$F = \{(r(u),\sigma) \mid (u,\sigma) \in I_c^t\} \tag{5}$$

$$\text{where } I_c^t = \{(u,\sigma) \mid \kappa(u,\sigma) = 0, u \in [0,1], \sigma = t\}$$

본 논문에서는 식(5)에 의하여 추출된 곡선의 이미지 좌표를 특징 점으로 하는 방법을 제안한다. 선택된 일련의 제스처에 대하여 위의 식을 이용하여 특징을 추출하여 80개의 모델로 이루어진 데이터베이스를 만들었고, 윤곽선의 유사도는 특징 점들과 대응점들의 거리로서 측정하였다. 그림 4는 부정확하게 추출된 윤곽선에 대하여서도 정확하게 매칭된 모델들의 예를 보여준다.



(a) (b)

그림 3. CSS 표현으로 나타내진 두 윤곽선

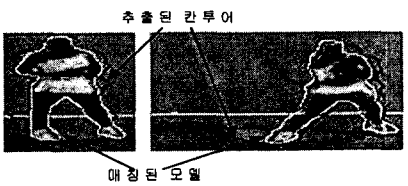


그림 4. 추출된 윤곽선과 매치된 모델의 예

4. 시퀀스 매칭

본 논문에서는 제스처를 인식하기 위한 방법으로 일련의 제스처 데이터를 시간 축에 대하여 곡선 형태를 따르도록 정렬함으로 학습시키고, 입력된 시퀀스를 학습된 공간에 투영하여 학습된 곡선과의 가능도(Likelihood)를 측정함으로 인식하는 방법을 제안한다.

N을 총 모델의 개수, g_k는 시간의 흐름에 따라 정렬된 모델에 대한 인덱스라고 하고, $0 < k \leq n$인 모델이 인식할 제스처에 대한 모델이라고 하자. 여러 개의 비슷한 윤곽선들이 있을 경우에 클러스터링 기법을 이용하여 하나의 인덱스로 지정할 수 있다.

$$D = \{g_1, g_2, \cdots, g_n, g_{n+1}, \cdots, g_N\} \tag{6}$$

$$D' = \{h_1, h_2, \cdots, h_n, (h_n + g_{n+1}), \cdots, (h_n + g_N)\} \tag{7}$$

$$\text{where } h_i = C(g_i)$$

곡선 C에 대하여 새로운 D'는 식 (7)과 같이 주어진다. C를 직선의 방정식, $C(g_k) = ag_k + b$로 사용하면 D'는 선형적으로 정렬되고, 두개의 매개변수, a, b만 학습이 필요하다. $v_s = [s, s+l]$을 길이 l을 가지고 시작 프레임을 s로 가지는 입력 시퀀스의 구간이라고 하자. 이 구간에 대하여 $C'(g_j) = a'g_j + b'$, $g_j \in v_s$를 만족시키는 a', b'를 계산하면 학습된 제스처에 대한 가능도를 식 8과 같

이 계산 할 수 있다.

$$L(v_s) = -\frac{1}{2\pi}\left(\frac{1}{\alpha_\sigma} e^{\frac{(a'-a_\mu)^2}{2a_\sigma^2}} \times \frac{1}{\beta_\sigma} e^{\frac{(b'-b_\mu)^2}{2b_\sigma^2}}\right) \quad (8)$$

구현에 있어서 구간의 길이를 고정하여 실험하였지만, 같은 의미를 가지는 동작의 속도는 인식에 영향을 미칠 정도로 크게 바뀌지 않는다고 가정 할 수 있으므로 a, b의 편차를 범위 내에서 인식 할 수 있다.

$L(v_s) > L_{threshold}$인 $L(v_s)$ 중에서 국소 최대 값을 가지는 구간을 추출하므로 제스처를 추출할 수 있다. 해당하는 제스처에 대한 구간과 그렇지 않은 구간에서의 $L(v_s)$ 값은 실험 결과 10^{11} 이상의 값을 차이를 보이므로 $L_{threshold}$의 값은 성능을 크게 좌우하지 않는다.

5. 실험 결과 및 분석

670x536해상도의 2003년도 윔블던 테니스 여자 단식 경기 결승 비디오에서 50개의 경기 장면을 추출하였고, 제스처 매칭 대상은 카메라 쪽에 가까이 있는 선수로 하여 실험 하였다. 임의의 서브가 있는 장면을 선택하여 서브 제스처를 포함한 80개의 윤곽선을 추출하여 모델의 데이터베이스를 만들었으며, 제스처 매칭을 위한 직선의 매개 변수 학습을 위하여서는 4개의 서브가 있는 장면을 추출하였다. 표 1은 실험 결과를 보여준다. 'A'는 윤곽선 모델 추출에 사용된 선수, 'B'는 윤곽선 모델 추출에 사용되지 않은 선수를 의미하고, '오른쪽', '왼쪽'은 테니스 코트의 어느 쪽에서 서브를 넣느냐를 의미한다. True Positive(TP), True Negative(TN), False Positive(FP), False Negative(FN)는 각각 올바르게 서브를 검출한 경우, 서브가 없는 시퀀스에 대해서는 서브 검출을 안 한 경우, 서브가 없는 시퀀스에서 서브를 오검출 한 경우, 서브가 있는 시퀀스에서 서브를 오검출 한 경우를 의미한다. CDP(Continuous Dynamic Programming) 방법과의 비교를 보여준다[4].

표 1 실험 결과

		Player A		Player B		Total
		Left	Right	Left	Right	
CDP	TP	5/11	8/11	4/8	7/8	62% correct
	TN	7/12				
	FP	4/11	3/11	4/8	3/8	36.8% incorrect
	FN	6/11	3/11	4/8	1/8	
제안한 방법	TP	8/11	11/11	7/8	7/8	90% correct
	TN	12/12				
	FP	0/11	0/11	0/8	0/8	5.2% incorrect
	FN	2/11	0/11	1/8	1/8	

그림 5는 제안한 방법으로 추출된 결과 영상들을 보여준다. 첫 번째 줄은 매칭된 포즈의 결과를 보여주고, 두 번째 줄은 학습된 서브 제스처와의 가능도를 보여주고, 세 번째 줄은 추출된 서브 구간에서 대표적인 영상들을 보여준다. 그림 5(a)는 제안된 제스처 매칭 방법으로, 서브 제스처를 포함하고 있는 시퀀스에 대하여 매칭을 수행한 결과이다. 서브가 있는 부분에서는 가능도의 값이 높고, 그렇지 않은 부분에서는 0에 가까운 값을 보여준다. 그림 5(b)는 한 장면 안에 두개의 서브 제스처가 있는 시퀀스에 대한 결과이다. 0번째 프레임과 574번째 프레임에서 피크를 보여주고 있다.

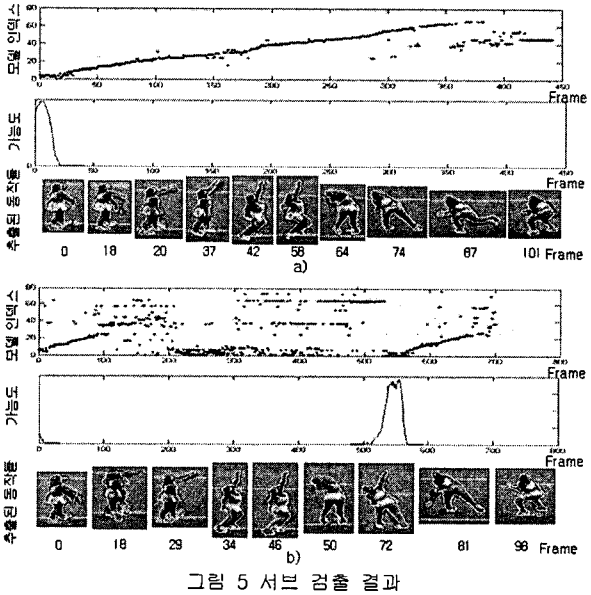


그림 5 서브 검출 결과

실험을 위해 학습시킨 윤곽선 모델은 테니스 코트의 오른쪽에서 서브를 넣는 장면을 사용하였고, 왼쪽에서 서브를 넣는 경우는 약간 다른 포즈를 취하게 되므로 대체적으로 왼쪽에서 서브를 넣는 경우 윤곽선 매칭이 오른쪽에서의 경우보다는 떨어지는 결과를 가져왔다. 그러나 많은 부분의 윤곽선 매칭은 잘 이루어지므로 결과적으로 시퀀스 매칭 시에 크게 영향을 주지 않게 되어서 선수 A가 왼쪽에서 서브를 넣었을 경우 한 장면을 제외하고는 서브를 잘 추출할 수 있었다.

추후 연구로는 좀 더 다양한 제스처에 대하여 인식이 가능하도록 모델 데이터를 학습하는 방법의 확장이 필요하고, 제스처 실시간 추출/인식을 위하여 효율적인 객체의 추출 및 윤곽선 매칭이 필요하다.

참고 문헌

[1] J. R. Wang and N. Parameswaran, "Survey of Sports Video Analysis: Research Issues and Applications," Proc. of Pan-Sydney Area Workshop on Visual Information Processing, Sydney, Australia, Vol. 36, 2004, pp. 87-90.

[2] J. Sullivan and S. Carlsson, "Recognizing and Tracking Human Action," Proc. of European Conference on Computer Vision, Vol. 2350, 2002, pp. 629-644.

[3] F. Mokhtarian, M. Z. Bober, Curvature Scale Space Representation: Theory, Applications, and Mpeg-7 Standardization, Kluwer Academic, 2003.

[4] R. Oka, "Spotting method for classification of real world data," The computer Journal, Vol. 41, No. 8, pp. 559-565, 1998