

3차원 체인코드와 은닉마르코프 모델을 이용한 권투모션 인식

Recognition of Fighting Motion using a 3D-Chain Code and HMM

한 창 호*, 오 춘 석, 최 병 욱
(Changho Han¹, Choonsuk Oh¹, and Byung-Wook Choi²)

¹Sunmoon University

²Korea Institute of Industrial Technology

Abstract: In this paper, a new method to recognize various motions of fighting with an aid of HMM is proposed. There are four kinds of fighting motion such as hook, jab, uppercut, and straight as the fighting motion. The motion graph is generalized to define each motion in motion data and the new 3D-chain code is used to convert motion data to motion graphs. The recognition experiment has been performed with HMM algorithm on motion graphs. The motion data is captured by a motion capture system developed in this study and by five actors. Experimental results are given with relatively high recognition rate of at least 85%.

Keywords: HMM, recognition, motion graph, motion capture, 3D chain code

I. 서론

최근 모션 인식에 대한 연구가 지속적으로 증가하고 있다. 그동안 글씨체 인식, 지문 인식 등으로 활용되다가 손, 발 동작에서 전체 모션을 인식하는 분야로 확대되었다. 그리고 이러한 인식 능력을 로봇에 적용하기에 이르렀다. 실제, 게임이나 애니메이션에도 활용되기도 한다.

본 논문에서는 권투 모션을 연기자로부터 얻어 각 모션을 인식하는 방법을 제시한다. 연기자로부터 모션을 얻기 위해 모션캡처 시스템이 사용되며, 얻어진 모션 데이터는 모션 그래프로 변환되며, 이때 새로 제시한 3차원 체인코드를 사용한다. 그리고 HMM 인식 모델을 설정하고, 모션 인식률을 얻기 위해 각 모션별로 실험을 했다.

II. HMM 알고리즘 적용 방법

1. HMM 적용 프로세스

은닉 마르코프 모델(hidden markov model)이란 마르코프 모델이 감추어졌다는 의미이다. 관찰이 불가능한 숨겨진 확률론적 과정을 관찰이 가능한 기호로 발생시키는 다른 확률론적 과정을 통하여 모형화하는 이중의 확률론적 과정이다. HMM은 통계적 모형의 하나로서 시계열로 표현되는 신호의 패턴을 인식하거나 필기체를 인식하는 문제에 널리 이용되었다. 어떤 객체의 패턴을 표현하는 모형으로서, 그 객체의 패턴을 심볼 스트링 형태로 분석한다. 통계적 특징을 훈련하여 입력 스트링에 대해 관찰할 확률이 얼마인가

하는 최고 확률을 계산하는 방법이다. 본 연구에서는 구해진 최고의 관찰된 확률을 통해 패턴의 스트링을 인식하게 된다.

HMM의 모델 학습과정은 관찰열에 대하여 최대 확률을 가지는 적합한 모델을 선택하고, 이 모델의 내부 상태들 중에서 최적의 상태 순서를 결정한 후, 관찰열 X 에 대한 $P(X|\lambda)$ 를 최대화하는 모델 $\lambda = (\pi, A, B)$ 의 매개 변수를 결정한다. 마르코프 매개변수는 다음 식과 같이 세 가지 매개변수를 갖는다.

① 사전 확률(prior probability): 초기 상태 확률 벡터라고도 하는데, HMM을 처음 시작할 때 어느 상태에서 시작할지를 결정하는데 사용한다.

$$\pi_i = P(q_1 = s_i), \sum_{i=1}^t \pi_i = 1 \quad (1)$$

② 상태 천이 확률(transition probabilities): A는 HMM이 시행중 다음 상태를 결정하는데 사용된다. a_{ij} 는 시간 t-1에 상태 s_i 에서 시간 t에 s_j 로 이동할 확률이다. 조건은 a_{ij} 의 모든 합은 1이어야 한다.

$$A = \{a_{ij}\}, 1 \leq i, j \leq t$$
$$a_{ij} = P(q_t = s_j | q_{t-1} = s_i), \sum_{j=1}^t a_{ij} = 1 \quad (2)$$

③ 관찰 심볼 확률(emission probabilities): B는 HMM이 어느 상태에 도달했을 때 그 상태에서 관찰되는 심볼을 결정하는데 쓰인다. B_{ik} 는 상태 S_i 에서 심볼 V_k 가 관측될 확률이다.

이렇게 구해진 모델은 확률 추정하게 된다. 방법은 3단계로 이루어진다.

① 평가(evaluation): 관측열 $X = \{x_1, x_2, \dots, x_T\}$ 와 모델 $\lambda = (\pi, A, B)$ 에 대하여 주어진 HMM에서 관찰되어진 순서의 확률 $P(X|\lambda)$ 를 계산하는 문제이다. 이 경우는 전방향

* 책임저자(Corresponding Author)

논문접수: 2010. 4. 16., 수정: 2010. 6. 4., 채택확정: 2010. 6. 10.

한창호: 선문대학교 전자공학과(liberman@paran.com)

오춘석: 선문대학교 정보통신공학과(csoh@sunmoon.ac.kr)

최병욱: 한국생산기술연구원(bwchoi@kitech.re.kr)

※ 본 논문은 지식경제부 전략기술개발사업 로봇컴폰넌트기술개발 과제 지원하여 연구하였음(No.10030817).

※ 본 논문은 2010년도 ICROS 학술대회에서 초안이 발표되었습니다.

알고리즘과 후방향 알고리즘을 이용하여 해결이 가능하다.

② 디코딩(decoding): 관찰열 $X = \{x_1, x_2, \dots, x_T\}$ 와 모델 $\lambda = (\pi, A, B)$ 에 대하여 최적의 상태순서 $Q = \{q_1, q_2, q_3, \dots, q_M\}$ 를 생성할 확률이 가장 높은 은닉상태들 간의 순서를 찾는 문제이다. 일반적으로 동적 프로그래밍 기법의 하나인 비터비 알고리즘을 사용한다.

③ 학습: 관찰된 관찰열 $X = \{x_1, x_2, \dots, x_T\}$ 에 대하여 $P(X|\lambda)$ 를 최대로 하는 모델 $\lambda = (\pi, A, B)$ 의 매개변수를 결정하는 문제이다. 매개변수의 추정에는 Baum-Welch 알고리즘을 사용하여 수행할 수 있다.

먼저 HMM을 이용한 전체 프로세스는 다음 그림과 같다. 모션 데이터를 읽어 들여 초기화를 거쳐 평가단계를 거쳐 인식단계로 넘어간다.

그림 1에서 HMM을 초기화 하는 과정에서 전이 행렬(transition matrix)과 관찰 행렬(observation matrix)은 훈련 과정을 통해 확률 값이 결정된다. 마르코프 모델(markov model)에서는 3x3 전이 행렬의 의미는 3가지 상태를 갖고 있으며 상태가 전이될 확률을 갖는 행렬을 의미한다. 즉, 본 논문에서는 상태를 모두 27가지 3차원 체인코드로 설정

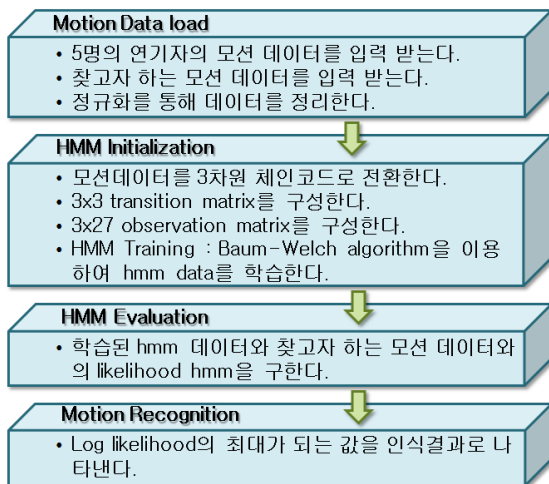


그림 1. HMM 인식 프로세스.
Fig. 1. HMM recognition process.

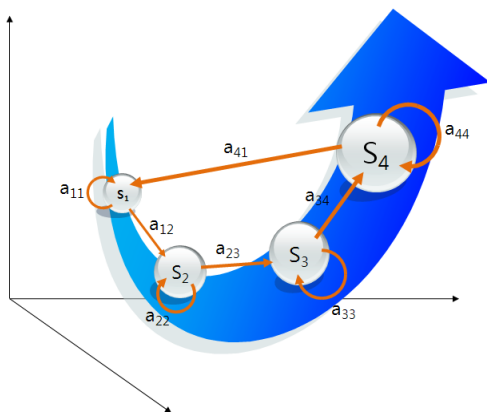


그림 2. 상태 전이 다이어그램 (어퍼컷).
Fig. 2. State transfer diagram (uppercut).

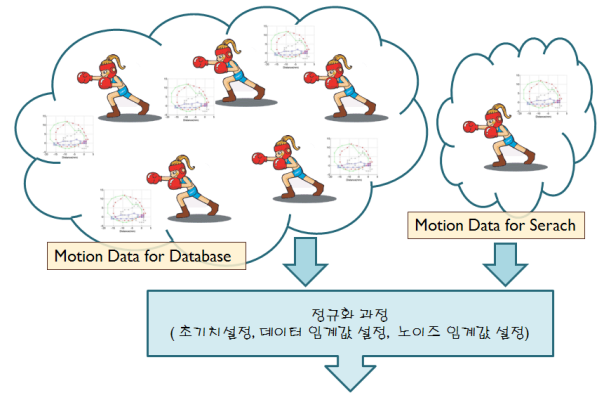


그림 3. 모션 데이터 불러오기.
Fig. 3. Load motion data.

하여 수행을 하므로, 27x27 형태의 행렬이 필요하다. 하지만 은닉 마르코프 모델(HMM)에서는 3x3 전이 행렬과 3x27 관찰 행렬로 구성하여 마치 3x3 행렬에 숨겨진 듯한 모델로 표현을 하고 있다.

그림 2는 어퍼컷의 마르코프 모델을 보여주고 있는데 여기서는 3개가 아닌 4개의 상태를 보여주고 있다. 실제로는 4개로는 어퍼컷을 구성할 수 없으며, 이것을 HMM에서는 s1, s2, s3, s4, (s5, s6, ...)중, 각 상태에서 27개의 상태 중 관찰된 상태가 시간에 따라 진행되는 것을 의미한다. 이 s1, s2, s3, s4, .. 를 묶어서 상태열이라 한다.

상태열은 모션 그래프로부터 3차원 체인 코드를 이용하여 데이터를 변환하여 얻을 수 있다. 이것을 HMM에 적용하여 매개변수를 구하고 전이확률을 구하여 HMM 모델을 생성한 후, 인식할 때는 학습 모델과 비교하여 likelihood 값으로 판단한다. 다음은 HMM 평가 단계이다. 학습된 HMM 데이터와 찾고자 하는 모션 데이터의 HMM-likelihood를 구하며, 그 값 중 가장 큰 값을 모션이 가까운 것으로 인식한다.

2. 연기자 모션 데이터

모션 데이터를 구축하기 위해 다섯 명의 후보자를 선출하였다. 이들은 권투 전문가가 아닌 일반인이다. 다섯 명에게 각자 3가지 형태의 큰 동작들로 구성된다고 인지 시켰으며, 세 가지 연기동작은 기본동작(normal), 자유동작(free), 실전동작(real) 등으로 구분하였다. 그리고 세부 동작에는 훅(hook), 잭(jab), 스트레이트(straight), 어퍼컷(uppercut) 이 있다. 그리고 하나의 세부 동작에는 4개의 마커 데이터로 이루어지다. 이렇게 준비된 연기자를 통해 마커를 들고 실험을 했다. 최종 실험에서 사용되는 모션 그래프 개수는 모두 240개 (연기자 5명 x 3가지 연기동작 x 4가지 세부동작 x 4개 마커)가 된다.

3. 모션 그래프

대량의 모션 데이터로부터 적절한 데이터를 부분적으로 선택하고 이를 새로운 순서로 연결하여 새로운 동작 시퀀스를 생성하는 것이다. 이러한 작업을 효율적으로 수행하기 위해 모션 데이터를 방향성 그래프로 표현하고 저장한다. 모션 데이터가 입력으로 주어지면, 그 안의 모든 프레임들 중에 서로 자세와 속도가 비슷한 프레임 쌍을 찾아내고 이

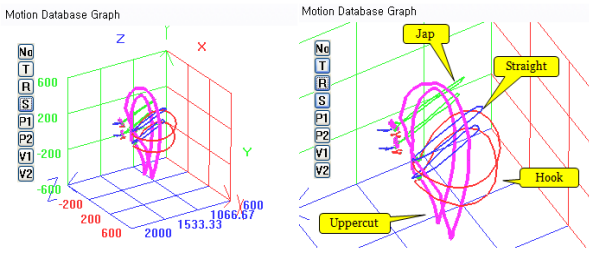


그림 4. 네 가지 모션의 그래프.
Fig. 4. Four kind of motion graph.

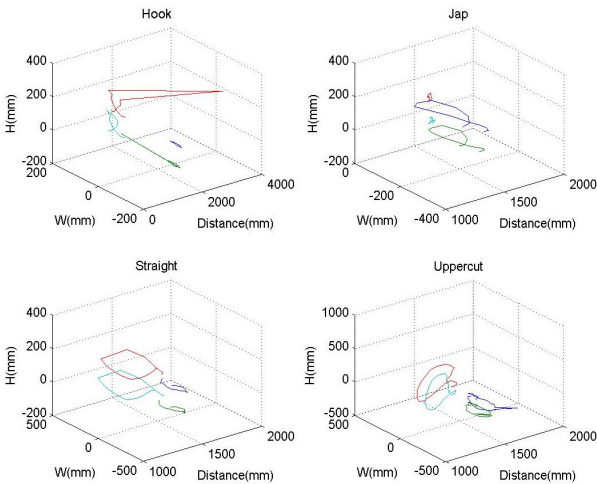


그림 5. 모션 그래프.
Fig. 5. Motion graph.

들 간의 전이 에지를 연결함으로써 모션 그래프를 구성할 수 있다. 마커로부터의 정보를 각각의 모션으로 분류하여 데이터베이스에 저장한다. 저장된 모션들은 각각의 그래프로 구성되어 인식시스템에서 사용된다.

그림 4에서 x, y, z축을 기준으로 그래프가 보여지고 있는데, 이것은 월드 좌표계이며, 모션 캡처 시스템에서 가져온 그림이다. 마커가 4개인데 왼쪽손에 두개와 오른쪽 손에 두개를 사용하고 있으며, 그림에서는 오른쪽 손의 두 개의 마커가 표현되고 있다. 즉, 각 모션들이 두 개씩 보여진다.

그림 5는 다섯 명의 연기가 각자 연기 동작에 따라 세부 동작을 펼치는 연기의 모션 그래프를 보여준다. 한사람의 연기는 세 가지 동작을 하도록 했다. 첫째는 기본 동작이다. 변화가 거의 없이 모션에 충실한 동작이다. 둘째는 자유동작이다. 기본동작과 유사하지만 좀 편안하게 동작하도록 했다. 셋째는 실전동작이다. 실전과 비슷하게 동작하도록 했다. 연기자들은 모두 권투선수가 아니므로 실전동작이 권투모션과는 다를 수 있다. 하나의 동작에는 모두 네 가지 모션을 연기한다. 네 가지 모션은 오른쪽 훅, 왼쪽 잼, 오른쪽 스트레이트, 오른쪽 어퍼컷이다. 연기가 마다 어깨 너비와 팔 길이, 키, 속도 등이 다르므로 모션 데이터는 조금씩 다르게 보인다.

그림 5에서는 좌표계가 약간 달리 표현을 했다. 넓이(width), 높이(height), 그리고 거리(distance)를 사용했는데, 인식 시스템에서 사용하고 있을 뿐, 월드 좌표계와 동일한

표현이다.

4. 3차원 체인 코드

3차원의 데이터에서는 8방향 형태의 기존 체인코드와는 다르게 27방향(자기 자신 포함) 체인코드를 사용한다. 이런 방법 말고 정육면체 형태로 구성된 내부 공간에서의 위치를 상태로 만들 수도 있으나 만약 30개씩 각각 x, y, z축을 구성한다면 총 상태는 30x30x30=27000개가 된다. 너무 많은 상태가 존재하게 되어 HMM 모델로는 적합하지 않기에 다음과 같은 27개 코드로 구성된 체인코드를 관측 심볼로 사용한다. 문자열로 구성된 데이터베이스의 모션 데이터는 체인코드로 이루어진 문자열로 나타난다. 왼손과 오른손의 구별이 있으며, 정규화 과정을 거쳐 나타난 결과이다.

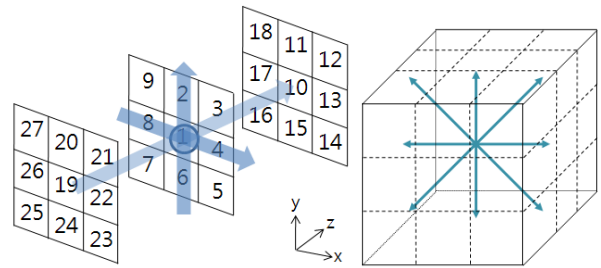


그림 6. 27방향의 3차원 체인코드.

Fig. 6. 27-direction of 3 dimension chain-code.

표 1. 데이터베이스의 모션 데이터와 체인코드 문자열.

Table 1. Motion data and chain code in database.

	Hook [x,y,z,θ ₁ ,θ ₂]	Jab	Straight	Uppercut				
3차원 모션 데이터와 체인코드	0 0 0 3 3	0 0 0 3 3	0 0 0 3 3	0 0 0 3 3				
	0 1 1 0 1	1 1 0 0 0 1	4	0 0 1 1 0 1	11			
	1 0 1 1 0	5	0 1 0 0 1	9	1 0 0 0 0	4	0 1 1 1 0	1
	1 1 1 0 0	2	0 1 0 1 0	1	1 1 0 0 0	2	1 1 1 0 0	4
	2 1 2 1 1	13	0 1 0 0 1	1	1 1 -2 0 0	19	2 0 1 1 0	5
	4 1 1 0 1	22	0 1 0 1 0	1	2 2 -4 0 0	21	4 -1 0 0 1	23
	6 0 0 0 1	23	0 1 -2 1 0	19	1 3 -7 1 0	27	6 -5 -2 0 1	23
	8 1 -1 0 0	21	0 2 -4 1 0	20	1 4 -11 1 0	20	7 -9 -2 1 0	5
	10 1 -3 0 0	22	-1 3 -7 1 0	27	0 4 -14 1 0	26	8 -13 -3 0 1	23
	11 2 -6 0 0	21	0 4 -10 0 0	21	0 4 -16 1 0	19	7 -15 -3 1 1	7
	12 3 -9 0 0	21	0 5 -14 0 0	20	-1 5 -16 1 0	9	6 -16 -4 1 1	25
	10 4 -11 1 0	27	-1 5 -16 1 0	26	-1 5 -15 0 1	10	5 -16 -5 1 0	26
	8 4 -14 1 0	26	0 6 -16 1 1	3	-1 4 -13 0 0	15	4 -14 -7 1 0	27
	6 5 -15 1 0	27	0 5 -15 1 0	15	-1 4 -11 1 0	10	4 -11 -9 1 0	20
	2 6 -15 1 0	9	0 5 -12 1 0	10	-1 3 -8 1 0	15	3 -7 -10 1 0	27
	0 6 -14 0 1	17	0 4 -9 0 0	15	-1 2 -5 1 0	15	3 -4 -11 1 0	20
	-2 6 -12 0 0	17	0 2 -6 0 0	15	0 1 -2 1 0	14	2 1 -11 1 0	9
	-2 5 -9 0 0	15	0 1 -3 0 0	15	0 0 0 1 0	15	2 4 -11 0 1	2
	-2 4 -7 1 0	15	0 1 -2 0 0	10	0 0 1 1 0	10	1 8 -10 0 1	18
	-2 3 -5 1 0	15	-1 1 0 0 0	17	1 0 1 1 0	4	1 11 -9 0 1	11
	-1 2 -3 1 0	14	-1 1 -1 1 1	19	1 0 0 1 1	19	0 13 -7 0 1	18
	0 2 -1 1 0	13	-1 1 -1 0 1	1	0 0 0 0 0	8	-1 14 -6 0 1	18
	1 1 0 1 0	14	-1 1 -1 0 0	1			-2 14 -3 0 1	17
	1 1 1 1 0	10	-1 1 -1 0 0	1			-3 13 -1 0 0	16
	2 1 1 0 0	4	-1 1 -1 0 0	1			-3 11 0 0 0	15
	2 1 1 0 0	1	-1 1 -1 0 1	1			-3 9 1 1 0	15
							-2 8 2 1 0	14
						-2 6 1 0 1	24	
						-1 4 1 1 0	5	
						0 2 1 0 1	5	
						0 1 1 0 0	6	
						0 0 1 1 1	6	
						0 1 0 1 0	20	
						0 1 0 0 0	1	

혹의 경우 구해진 문자열이 [11 5 2 13 22 23 21 22 21 21 27 26 27 9 17 17 15 15 15 14 13 14 10 4 1]로 되는데, 첫 번째 11은 x=0,y=1,z=1의 방향 체인코드를 의미한다. 그 다음은 5->2->... 이렇게 동작의 움직임을 표현하고 있다. 혹의 모션데이터의 행은 각 x, y, z값과 더불어 θ_1, θ_2 로 구성된다. 각도 θ_1, θ_2 는 x-y, y-z간의 각도를 의미한다.

III. 실험 결과

1. 실험장비

모션을 캡처하기 위해 사용된 장비는[7,8]에 나와 있는 자체 개발한 시스템을 사용했다. 캡처는 실시간으로 이루어지고 있으나, 인식은 현재 실험 단계에 있으며, 시스템에 적용하지 않았다.

2. 인식 결과 그래프

은닉 마르코프 모델(HMM) 알고리즘을 실행하면 log likelihood로 값이 구해진다. 그림 7은 첫 번째 연기자의 모션을 HMM 기법을 이용해 인식한 결과를 보여준다. 위쪽의 그래프들은 각각 모션 데이터의 위쪽과 옆에서 보여주는 모션 그래프이며, 아래쪽의 그래프는 마르코프 모델 상태에 대한 변이도와 데이터베이스의 모션과 찾고자하는 모션의 likelihood값의 차이를 보여주고 있다. 절댓값으로 가장 작은 값이 인식한 결과로 한다. 왼쪽 아래 그림에서는 세로축

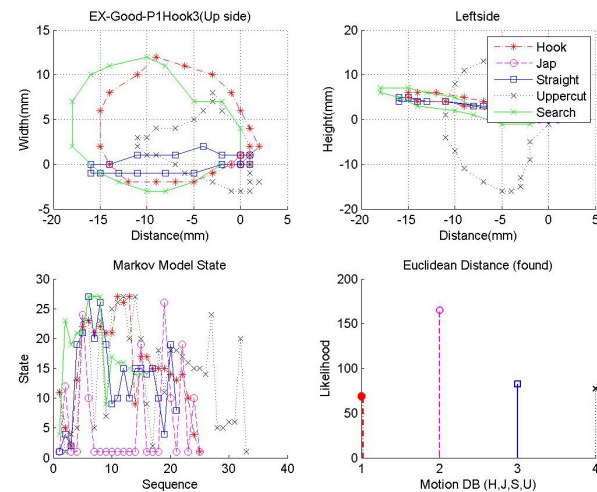


그림 7. 혹 모션 인식.

Fig. 7. Recognition of Hook motion.

표 2. HMM을 이용한 모션 인식 결과.

Table 2. Recognition result for HMM.

모션 연기자	Hook	Jab	Straight	Uppercut
Person 1	○	○	○	○
Person 2	○	○	○	○
Person 3	○	○	○	✗
Person 4	○	○	○	○
Person 5	○	○	○	○
인식률	95 %			

이 state로 되어 있는데, 이것은 시간의 따른 모션의 위치 상태이다. 예를 들어, 혹은 26개의 상태로 나타나고, 어퍼컷은 31개의 상태로 나타난다.

3. 인식을 결과

마르코프 모델을 이용한 인식한 결과 최대 95% 성능을 보였다. 물론 마르코프 모델을 훈련하는 과정에서 확률로 최대치를 추정하므로 실험할 때마다 결과가 항상 같게 나오지 않는다. 보통 85%-95%의 결과로 나타나고 있다. 95%는 마르코프 모델을 이용한 최대의 인식률이라고 할 수 있다. 모션인식률은 100%에 미달하는 주요한 이유는 인식하기 애매한 중간모션의 경우에 간단한 인식의 오차에 의한 경우이다. 그 밖에 모션이 너무 작거나 움직임의 변화가 많은 경우에 발생한다.

IV. 결론 및 향후 연구

본 연구에서는 권투 모션을 인식하기 위해 HMM 알고리즘을 이용했으며 적용 방법을 제시했다. 정규화를 했을 경우도 우수한 비슷한 결과를 얻었다. 인식률이 85%이상의 실험결과를 보여준다. 앞으로의 과제는 연구된 이론을 실용화하는 문제와 실시간으로 모션을 인식해야 하는 문제가 남아 있다. 실제 응용 프로그램으로 개발하여 게임이나, 애니메이션, 로봇 등에 적용할 수 있는 방안을 연구해야 한다.

참고문헌

- [1] P. Seo, Y. Ryu, and C. Oh, "Development of Omnidirectional Active Marker for Motion System with a Monocular PSD Camera," *Proc. of CICS'08*, Korea, pp. 379-381, 2008.
- [2] L. Kovar, M. Gleicher, and F. Pighin, "Motion graphs," *Proc. of ACM Transactions on Graphics (SIGGRAPH 2002)*, Los Angeles, USA, vol. 21, no. 3, pp. 473-482, 2002.
- [3] R. Bowden, "Learning statistical models of human motion," *Proc. of the IEEE Workshop on Human Modelling, Analysis and Synthesis*, pp. 10-17, 2000.
- [4] Chai, J., Hodgins, and J. K., "Performance animation from low-dimensional control signals," *ACM Transactions on Graphics (SIGGRAPH 2005)*, pp. 686-696, 2005.
- [5] D. DiFranco, T. Cham, and J. Rehg, "Reconstruction of 3-D figure motion from 2-D correspondences," *Proc. of Computer Vision and Pattern Recognition*, Hawaii, USA, vol. 1, pp. 307-314, 2001.
- [6] G. Xu and Z. Zhang, *Epipolar Geometry in Stereo, Motion, and Object Recognition: A Unified Approach*, Springer Netherlands, 2009.
- [7] C. Han, P. Seo, S. Kim, Y. Ryu, and C. Oh, "The study of improving the accuracy in the 3D data acquisition of Motion Capture System," *Proc. of Korea Computer Congress 2008*, Seoul, Korea, pp. 503-506, 2008.
- [8] C. Han, C. Oh, E. Shin, and B. Choi, "Comparing two

methods for acquiring 3D data of motion capture system by using PSD camera and CCD camera," *Proc. of International Conference on Convergence and Hybrid Information Technology*, Daejeon, Korea, pp. 558-563, 2009.



한 창 호

1993년 청주대 전자공학과(학사). 1995년 동 대학원 전자공학과(석사). 2009년 선문대학교 전자공학과(박사). 관심분야는 영상처리, 모션캡처, 자동검사, 3차원 영상, 모바일 로봇.

- [9] C. Han, C. Oh, and B. Choi, "The study on recognition of fighting motion by using HMM," *Proc. of ICROS Annual Conference 2010*, Chuncheon, Korea, pp. 78-79, May 2010



오 춘 석

1980년 서강대학교 전자공학과(학사). 1986년 Marquette University 컴퓨터공과(석사). 1992년 University of Arizona 컴퓨터공과(박사). 1980년~1993년 전자통신연구원(ETRI) 선임연구원. 2001년 미국 Optical Sciences Center Visiting Scholar. 1993년~현재 선문대학교 전자정보통신공학부 교수. 주 관심분야는 화상처리, 자동검사, 머신비전, 광학시스템설계.



최 병 욱

1978년 서울대학교 전기공학과(학사). 1990년 영국 Imperial College 전기공학과(석사). 1994년 영국 University of Leicester 제어시스템공학과(박사). 1978년~1989년 현대건설. 1994년 영국 레스터대 Research Associate. 1995년 서울대 제어계측신기술연구센터 선임연구원. 1996년~현재 한국생산기술연구원 수석연구원. 관심분야는 시스템제어, 자동화 및 정보화.