

논문 2006-43CI-1-3

제스처 인식을 위한 은닉 마르코프 모델

(Hidden Markov Model for Gesture Recognition)

박혜선*, 김은이**, 김항준***

(Park Hye Sun, Kim Eun Yi, and Kim Hang Joon)

요약

본 논문에서는 은닉 마르코프 모델 (HMM: hidden Markov model)을 이용한 제스처 인식 방법을 제안하고, 이를 게임 시스템의 인터페이스로 적용한 사례를 소개한다. 제안된 방법은 다음의 두 가지 특징을 가진다. 첫 번째는 사전에 분할된 데이터 열을 입력으로 사용하는 기존의 방법과는 달리, 제안된 방법은 카메라로부터 입력되는 비디오 스트림을 HMM의 입력으로 사용한다는 것이다. 두 번째는 제안된 HMM은 제스처의 분할과 인식을 동시에 수행한다는 것이다. 제안된 방법에서 사용자의 제스처는 13개의 제스처들을 인식하는 13개의 specific-HMM들을 결합하는 하나의 통합된 HMM을 통해 인식된다. 제안된 HMM은 사용자의 머리와 양손의 2D-위치 좌표로 구성된 포즈 심볼들의 열을 입력받는다. 그리고 새로운 포즈가 입력될 때마다, HMM의 상태 확률 값을 갱신한다. 그때, 만약 특정 상태의 확률 값이 미리 정해진 임계치보다 큰 경우, 그 특정 상태를 포함하고 있는 제스처로 인식한다. 제안된 방법의 정당성을 입증하기 위하여, 제안된 방법은 Quake II라는 컴퓨터 게임에 적용되었다. 실험결과는 제안된 방법이 높은 인식 정확률과, 계산 시간을 확연하게 감소시킬 수 있었음을 보여주었다.

Abstract

This paper proposes a novel hidden Markov model (HMM)-based gesture recognition method and applies it to an HCI to control a computer game. The novelty of the proposed method is two-fold: 1) the proposed method uses a continuous streaming of human motion as the input to the HMM, instead of isolated data sequences or pre-segmented sequences of data and 2) the gesture segmentation and recognition are performed simultaneously. The proposed method consists of a single HMM composed of thirteen gesture-specific HMMs that independently recognize certain gestures. It takes a continuous stream of pose symbols as an input, where a pose is composed of coordinates that indicate the face, left hand, and right hand. Whenever a new input pose arrives, the HMM continuously updates its state probabilities, then recognizes a gesture if the probability of a distinctive state exceeds a predefined threshold. To assess the validity of the proposed method, it was applied to a real game, Quake II, and the results demonstrated that the proposed HMM could provide very useful information to enhance the discrimination between different classes and reduce the computational cost.

Keywords: 제스처 인식, 은닉 마르코프 모델, 포즈 분류, 사용자-컴퓨터 인터페이스

I. 서론

최근 다양한 컴퓨터 비전 기술을 이용한 제스처 인식 기반의 인터페이스에 대한 연구가 많은 관심을 받고 있다^{[1][4][7]}. 이에 따라 많은 인식 시스템의 개발이 이루어지고 있으며, 이중 가장 많이 사용되는 방법은 다이내믹 프로그래밍이나 은닉마르코프 모델 (HMM: hidden Markov model)을 이용한 방법이다^{[1]-[7]}. 특히, HMM은 제스처 인식에서의 효율성이 증명되어 가장 널리 사용되고 있다. HMM은 제스처 인식 및 음성 인식 등의 분류문제를 풀기 위해 많이 사용되고 있는데, 그 이유는 HMM이 동적인 신호 및 이벤트를 모델화하는 데 유용

* 학생회원, 경북대학교 컴퓨터공학과
(Dept. of Computer Eng., Kyungpook National Univ.)

** 정회원, 건국대학교 인터넷미디어학부
(Dept. of Internet and Multimedia Engineering, NITRI, Konkuk Univ.)

*** 정회원, 경북대학교 컴퓨터공학과
(Dept. of Computer Eng., Kyungpook National Univ.)

※ 이 논문은 2004년도 한국학술진흥재단의 지원에 의하여 연구되었음. (KRF-2004-041-D00643)

접수일자: 2005년9월21일 수정완료일: 2006년1월3일

하기 때문이다.

하지만, 일반적으로 HMM은 분할된 동영상 스트림을 입력으로 사용하기 때문에, 인식을 위해 사전에 입력 데이터의 분할을 필수로 한다. 그러나 이러한 분할 작업은 쉬운 일이 아니다^[8].

Otsuka와 Ohya는 얼굴 표정 인식을 위해 HMM을 사용하였다^[9]. 이 때 HMM의 입력을 위해서 연속적인 얼굴영역의 변화로부터 휴리스틱 방법을 이용하여 시작과 끝을 추출하였다. 그러나 수많은 얼굴 데이터에 대해 휴리스틱 방법만으로 시작과 끝을 추출하기에는 한계가 있다. Quak와 kendon은 각 제스처는 일정한 속력을 가진 3단계로 구성된다고 가정하고, 단계별 속력 변화를 기반으로 제스처를 분할하여 인식하였다^{[10][11]}. 그러나 다양한 사람에 의해 수행되는 제스처의 동작에서 일정한 속력이 유지되기는 어렵기 때문에 분할 결과의 정확도에는 한계가 있었다. Lee와 Kim은 임계값 모델을 이용하여 제스처를 분할한 후, 인식하였다. 이 모델은 구현하기 쉽고 좋은 인식률을 보이지만, 많은 계산 시간을 필요로 한다는 단점이 있다^[12]. 이처럼 많은 제스처 분할 작업에 의한 제스처 인식은 분할 작업에 필요한 부가적인 계산 비용이 필요하며, 분할결과에 따라 인식 결과율이 크게 좌우되는 단점이 있다.

이러한 문제점을 해결하기 위해, 본 논문에서는 입력 비디오 스트림으로부터 제스처를 자동으로 분할하는 동시에 인식할 수 있는 새로운 HMM을 제안한다. 제안된 모델은 각 제스처를 인식하는 HMM들을 결합하는 통합된 하나의 HMM을 사용하여, 제스처들을 분할하고 인식한다. 제안된 모델은 사용자의 얼굴과 양손의 위치 좌표로 구성된 포즈들의 열(stream)을 입력받는다. 새로운 포즈가 입력될 때마다, 제안된 HMM은 상태 확률을 계속적으로 갱신한다. 이 때, 특정 상태의 확률 값이 미리 정해 놓은 임계값을 초과할 때 제스처로 인식한다. 제안된 모델의 유효성을 증명하기 위해 Quake II 라는 컴퓨터 게임의 인터페이스로 적용하였으며, 그 결과를 기존의 HMM과 비교하였다. 실험결과를 통해 제안된 방법이 인식 결과를 개선함과 동시에 연산시간을 감소할 수 있음이 증명 되었다.

그림 1은 제안된 제스처 인식 시스템을 보여준다. 제안된 시스템은 특징 추출 단계와 포즈 분류 단계, 그리고 제스처 인식 단계로 구성된다. 특징 추출 단계는 각 프레임으로부터 피부색 정보와 움직임 정보를 이용하여 특징 벡터를 검출한다. 특징 벡터는 사용자의 얼굴과 양손의 위치좌표이다. 검출된 각 특징 벡터에 대해서, 포

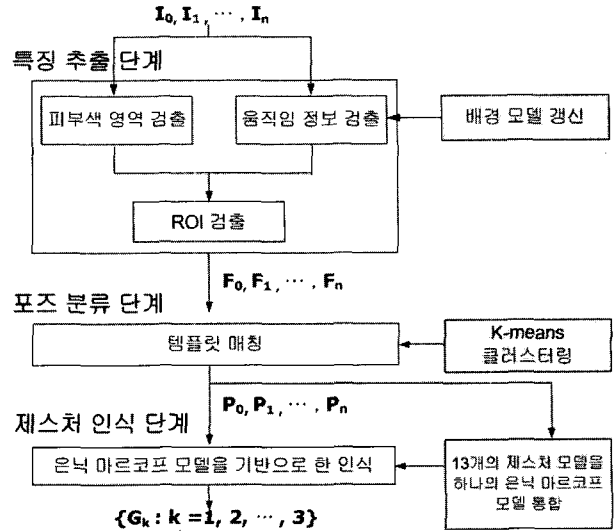


그림 1. 제안된 제스처 인식 개요
Fig. 1. Outline of proposed gesture recognition.

즈 분류단계에서는 K-means 클러스터링 방법을 이용하여 특징 벡터마다 포즈 심볼을 할당한다. 특징 추출 단계와 포즈 분류 단계를 통해서 입력 비디오 열은 포즈 심볼들의 열로 표현되고, 이는 제스처 인식단계의 입력으로 사용된다. 마지막으로 제스처 인식 단계에서는 입력 포즈 열을 정의된 제스처들로 자동으로 분할하는 동시에 인식한다.

제안된 논문은 다음과 같이 구성된다. II장과 III장에서는 각각 특징 추출 방법과 포즈 분류 방법을 설명하고, IV장에서는 제안된 제스처 인식 방법을 자세히 설명한다. V장에서는 실험 결과 및 성능 평가가 보이고, 마지막으로 결론을 이어진다.

II. 특징 추출

제안된 논문에서는 사용자의 얼굴영역과 양손의 움직임을 이용하여 제스처를 표현한다. 복잡한 배경 및 조명하에서 사용자의 얼굴 및 양손을 강건하게 검출하기 위해서, 움직임 정보와 피부색정보를 사용한다.

1. 움직임 정보 추출

이 단계에서는 입력 영상을 전경과 배경 영역으로 구분하기 위해, Chen^[13]이 제안한 방법을 사용한다.

이 방법은 입력 비디오 열의 첫 번째 프레임, I_0 로부터 배경 모델 BG_k 를 생성한 후, 현재 프레임 I_k 와 그 안에 전경 영역인 FG_k 를 사용하여 배경모델을 계속적으로 갱신한다. 이러한 갱신과정은 식 (1)과 같이 이루어진다.

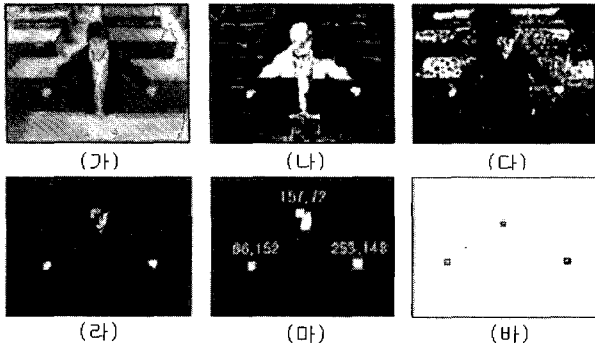


그림 2. 특징 추출 및 포즈 분류 결과: (가)원 영상, (나)움직임 영역, (다)피부색 영역, (라)ROI 영역, (마)특징 벡터 (바) 분류된 포즈
 Fig. 2. Feature extraction and pose classification results: (a) an original image, (b) moving region, (c) skin color regions, (d) ROI results, (e) a feature vector, (f) a classified symbol.

$$BG_{k+1} = (1-w)BG_k + wI_k, \tag{1}$$

이 식에서 BG_{k+1} 은 갱신된 배경모델을 말하고, w 는 가중치 인수로서 실험에서 0.1로 주어진다.

일단 배경 모델이 정해지면, 각 영상에서 전경영역은 다음 식에 의해서 얻어진다.

$$FG_k(x, y) = \begin{cases} 1 & |BG_k(x, y) - I_k(x, y)| \geq \theta \\ 0 & otherwise \end{cases} \tag{2}$$

이 식에서 임계값 θ 는 실험을 통해 30으로 정하였다. 즉, 전경 이미지 FG_k 는 움직임이 있는 부분은 1로, 그렇지 않은 부분은 0으로 표현한 이진 영상이다.

그림 2는 입력 영상으로부터의 특징벡터 추출 과정을 보여준다. 그림 2(가)는 카메라로부터 얻어진 원 영상을 나타내고, 그림 2(나)는 움직임 검출 결과를 보여 준다.

2. 피부색 영역 검출

사용자의 얼굴과 손 영역을 추출하기 위해 피부색 정보를 사용한다. 피부색은 조명 변화에 강건하고, 계산 및 구현이 용이한 SCT 상에서 정의 된다^[14]. 입력 영상은 아래의 식을 사용하여 SCT 색상 공간으로 변환된다.

$$L = \sqrt{R^2 + G^2 + B^2}, \quad \angle A = \cos^{-1} \left[\frac{B}{L} \right], \quad \angle B = \cos^{-1} \left[\frac{R}{L \sin(\angle A)} \right]$$

여기서 L 은 명도이고, $\angle A$ 와 $\angle B$ 는 색조를 나타낸다. 100장의 샘플 영상으로 피부색 영역의 색상 분포도를 조사해 본 결과, 표 1과 같은 분포도를 얻었다.

만약 어떤 화소의 색상 분포도가 표 1에 기술된 피부

표 1. 피부색 분포도

Table 1. Skin color distribution.

파라미터	살색 분포도
각 A	1.001 ~ 1.18
각 B	0.82 ~ 0.97
L	170 ~ 330

색 분포의 범위에 속하면, 그 화소는 피부색 영역으로 간주된다. 피부색 추출 결과가 그림 2(다)에 나타난다.

3. ROI 검출

움직임 영역과 피부색 영역을 결합하여 사용자의 얼굴과 손 영역이 추출 된다. 즉, 피부색 분포도를 가진 전경 영역은 얼굴 및 양손으로 간주된다.

그림 2는 특징벡터 추출과정을 보여준다. 그림 2(나)와 (다)의 움직임 추출 및 피부색 추출결과를 통합하여 얻어진 ROI영역이 그림 2(라)에 나타난다. ROI 영역은 연결성분 분석 방법을 이용한 후처리 과정을 거친 후, 최종적으로 그림 2(마)와 같은 얼굴 및 양손의 위치 좌표로 구성된 특징 벡터로 나타내어진다.

III. 포즈 분류

포즈 분류 단계에서는 추출된 각 특징 벡터에 대해 포즈 심볼을 할당한다. 본 논문에서는 하나의 제스처는 다음의 5개의 포즈로 구성되어있다고 가정 한다: {start pose, intermediate pose, distinctive pose, intermediate pose, end pose}. 또한 모든 제스처는 같은 포즈로 시작하고 끝이 나며, start pose와 end pose는 동일하다고 가정한다. Distinctive pose는 각 제스처를 구별 짓는 포즈로 제스처마다 유일하고, intermediate pose들은 start/end pose와 distinctive pose 사이의 포즈를 의미한다. 본 논문에서는 Quake II의 제어 명령어로 13개의 제스처를 정의하였고, 이러한 제스처를 위해 23개의 포즈를 정의하였다.

일반적으로 같은 포즈라 할지라도, 각 개인에 따라서 그리고 같은 사람일지라도 다르게 표현될 수 있기 때문에, 강건한 포즈 분류기가 필요하다. 이를 위해, 본 논문에서는 K-means clustering을 이용하여 생성된 템플릿과 추출된 특징벡터를 비교하여 포즈 심볼을 할당한다. 템플릿 생성과정은 다음과 같다. 트레이닝 데이터의 수집을 위해 10명의 사람이 13개의 제스처를 각각 10번씩 수행하였다. 트레이닝 데이터로부터 특징 벡터를 추출하고, 추출된 특징벡터 영상들에 대해 클러스터링 알고

리즘을 적용했다. 그 결과, 32개의 템플릿이 얻어졌다. 얻어진 템플릿들은 start/end pose와 distinctive pose, 그리고 intermediate pose의 3그룹으로 나누어진다. 생성된 템플릿을 이용하여 입력 특징 벡터는 포즈 심볼로 분류된다. 입력 특징과 템플릿과의 적합도(matching)를 계산하기 위해서 유클리디언 거리가 사용된다. 그림 2(바)는 추출된 특징 벡터에 대한 포즈 분류 결과를 보여 준다. 그림 2(가)는 'distinctive CV (표3의 포즈 정의 참조)'로 분류되었다.

IV. 제스처 인식

제스처 인식 단계에서는 추출된 포즈 심볼 열로 부터 제스처를 인식한다. 본 논문에서는 제스처 인식을 위해, HMM을 사용한다. 이 때 제스처 분할 문제의 어려움 및 인식의 정확도를 개선하기 위해 새로운 HMM을 제안한다. 그림 3은 제안된 HMM의 구조를 보여준다. 제안된 HMM은 정의된 13개의 제스처들을 각각 인식할 수 있는 독립된 13개의 specific-HMM들을 통합하는 모델이다. 이러한 통합은 13개의 제스처들이 모두 같은 포즈로 시작하고 종료된다는 가정에 의해서 이루어진다. 이러한 가정은 '일반적으로 사람들이 어떤 행동을 취한

후, 무의식적으로 자신의 고유 자세를 취하게 된다' 는 사람들의 행동 방식에서의 특징을 반영한 것이다. 이러한 가정에 의해, 제안된 모델은 그림 3의 (a)와 같은 구조를 가지게 된다. 준비(ready) 상태에서 시작하여, 제안된 HMM은 새로운 포즈 심볼이 입력될 때 마다, 각 상태들의 확률 값을 갱신한다. 이때 어떤 상태의 확률 값이 정해진 임계값보다 높을 경우, 그 상태를 포함하는 제스처로 인식이 되는 동시에 입력 스트림은 자동으로 분할된다. 제스처의 분할과 인식을 동시에 수행하는 제안된 HMM은 기존의 인식 방법들과 비교했을 때, 분할 단계에서 발생하는 에러를 감소할 수 있을 뿐만 아니라, 또한 분할 작업을 수행하는데 필요한 추가적인 계산을 제거할 수 있다는 장점이 있다.

1. Specific-HMM

그림 3의 (나)와 같이 각 제스처를 인식할 수 있는 specific-HMM들은 모두 같은 동일한 시작과 끝 상태를 가지는 left-right HMM이다. Specific-HMM은 파라미터의 집합 $\{\pi, \lambda, A, B\}$ 으로 정의된다. 이때, $\pi = P(q_1 = i | 0 \leq i \leq N - 1)$ 는 초기 상태 확률들을 가리키는 원소 벡터이다. 그리고 $A = \{a_{ij}\}$ 는 상태 전의 확률을 나타내는 $N \times N$ 의 행렬이다. 이 때, a_{ij} 는 상

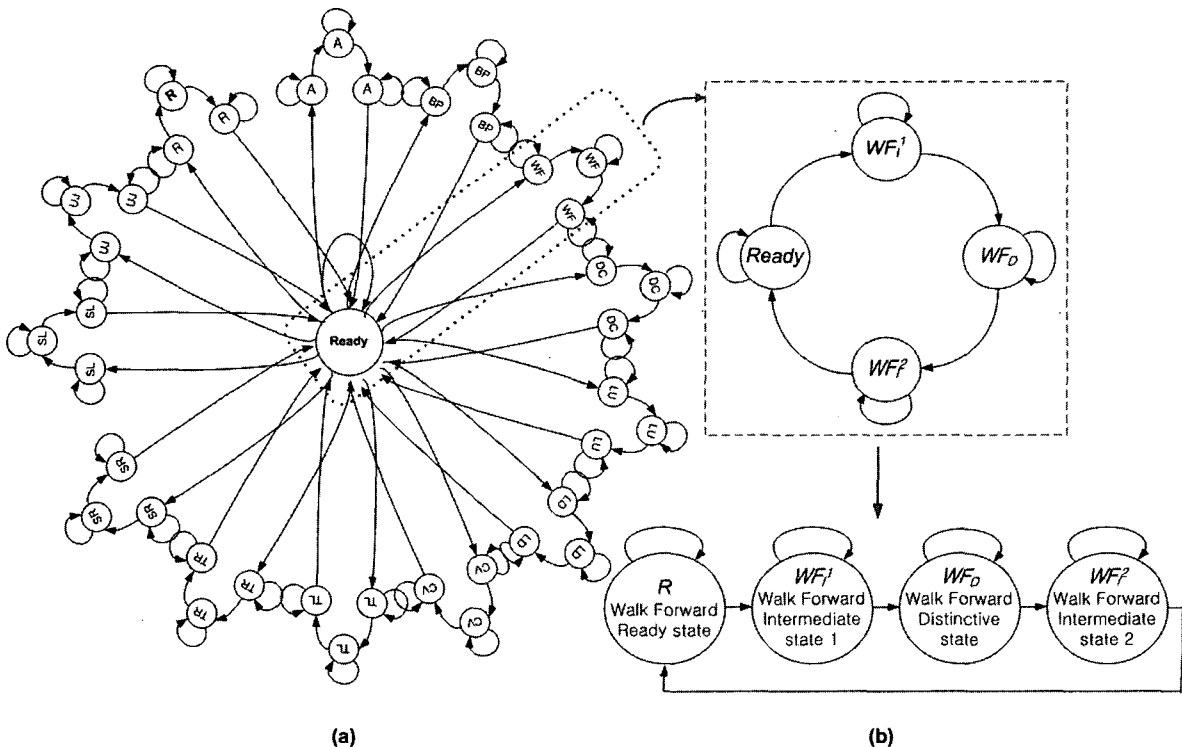


그림 3. 제안된 HMM 구조: (가) Unified-HMM, (나) Specific-HMM
 Fig. 3. Architecture of proposed HMM: (a) an unified HMM and (b) gesture-specific HMM.

태 i 에서 j 까지 전이확률이다.

$B = \{b_{ij} | 0 \leq i \leq N-1, 0 \leq j \leq M-1\}$ 는 상태 확률을 나타내는 $N \times M$ 의 행렬이다. 각 원소는 임의의 시간 t 에서 상태 j 에서 관측 심볼 o_t 가 나타날 확률로써,

조건 $\sum_{o_t=0}^{M-1} b_j(o_t) = 1$ 을 만족한다. 여기서, N 은 모델

에서의 상태의 수를 나타낸다. 제안된 방법에서는 실험을 통해 5로 주어진다. q_t 는 임의의 시간 t 에서 모델 안의 상태이다. T 를 출력 관측 심볼 열의 길이라고 할 때, q_t 와 t 는 각각 $1 \leq q_t \leq N$ 하고 $1 \leq t \leq T$ 하다. M 은 포즈 심볼의 수로써 실험에 의해 23으로 결정되어졌다.

$O_t = \{o_t | 0 \leq o_t \leq M-1\}$ 는 임의의 시간 t 에서의 각 상태에 대해 가능한 모든 관측 심볼이다.

HMM을 실제 문제에 적용하기 위해서는, HMM의 파라미터들이 학습과정을 통해 선택되어야 한다. 주어진 λ 에 대해, 관측 심볼 시퀀스 o 에 대한 최대 유사도인 $P(o | \lambda)$ 를 구하기 위해, λ 을 Baum-Welch 방법을 사용하여 추정하고, 이 후 $P(o | \lambda)$ 은 Forward-Backward algorithm을 사용하여 계산한다.

2. Unified-HMM

각 제스처를 인식하는 specific-HMM의 통합은 'ready state'의 공유를 통해 쉽게 이루어진다. 제안된 Unified-HMM은 미리 입력, 데이터로 잘라놓은 심볼 열을 받아 작업하는 일반적인 HMM과는 달리, 연속적인 심볼 스트림을 입력받아 제스처를 인식한다.

그림 3(a)에서와 같이 Unified-HMM은 1개의 'ready state'와 26개의 'intermediate state'들과 13개의 'distinctive state'들로, 총 40개의 상태(state)로 구성된다. 제안된 HMM에서의 상태 전이 확률과 관측 심볼 확률을 각각 행렬 A^U 와 B^U 로 표기하자. 이러한 확률들은 다음과 같은 과정을 통해 얻어진다. A^U 와 B^U 에서 준비상태의 확률 값의 경우에는 13개의 specific-HMM에서의 시작 상태의 확률 값들의 평균으로 얻어진다. 그렇지 않은 경우는 specific-HMM의 확률 값을 그대로 사용한다.

제안된 HMM에서, 초기 확률 값 π^U 는 수동적으로 결정되어진다. 모든 제스처는 준비 상태에서 시작하기 때문에, 'ready state'로의 전이 확률은 0.9로 하고, 이 'ready state'를 제외한 다른 39개의 상태로 전이확률은 0.1로 초기 확률 값을 정한다.

파라미터 집합, (A^U, B^U, π^U) 이 주어졌을 때, 제안된

시스템은 다음 식을 사용해서 각 상태 확률 값들을 계속해서 갱신한다.

$$S_n^t = \frac{\sum_{k=0}^{k-1} (S_k^{t-1} \times a_{kn}) \times b_{np}}{Z} \quad (3)$$

식 5에서, $S = \{s_k\}$ 는 상태 확률 벡터이다. 이때 s_k 는 k 번째 상태에 대한 상태 확률 값을, a_{kn} 은 s_k 의 상태에서 s_n 으로의 전이확률을 나타내고, b_{np} 는 상태 s_n 에서 포즈 심볼 v_p 가 발생할 확률 값을 나타낸다. Z 는 정규화 인수로, $\sum S_k = 1$ 을 만족한다.

식 (5)을 이용하여 새로운 포즈 심볼이 입력 될 때 마다, 제안된 모델은 각 상태 확률 값들을 계속적으로 갱신하고, 이 때 만약 특정 상태의 확률 값이 미리 정해진 임계값 값을 초과하면, 그 특정 상태를 포함하고 있는 제스처로 인식하고, 입력 스트림을 분할하게 된다.

V. 구현

제안된 방법의 효율성을 증명하기 위해, 제안된 인식 시스템을 Quake II 게임의 인터페이스로 적용하였다. 표 2는 게임명령어로 사용된 13개의 제스처를 보여준다. 각 제스처는 start/end pose와 distinctive pose, 그리고 intermediate pose로 구성되어있다고 가정하였기 때문에, 표 3에서와 같이, 23개의 포즈들로 나타난다.

VI. 실험 결과

제안된 방법은 여러 가지 조명상태를 가진 다양한 환경에서 실험되었다. 실험을 위해 10명의 사용자가 13개의 제스처들을 각각 10번씩 수행함을 통해 1300개의 실험데이터를 수집했다. 이 중 절반의 영상은 HMM의 훈련을 위해, 나머지는 테스트로 사용되었다.

그림 4는 각각 서로 다른 환경에서 얻어진 테스트 영상에 대한 실험결과를 보여준다. 이는 제안된 방법이 복잡한 배경 하에서도 특정벡터들을 정확히 추출하고 있으며, 사용자에게 강건한 포즈 분류 결과를 제공한다.

표 4는 제안된 시스템의 제스처 인식 결과를 보여준다. 제안된 방법은 95.92%의 인식 정확도를 가진다. 표 4에서 보이듯이, 제안된 HMM은 몇 가지 제스처들을 잘못 인식한다. 하지만 그 예러는 매우 작기 때문에 무시할 수 있다. 한편 제안된 방법에서는 많은 'rejection'

표 2. 퀘이크 2 게임에서의 13개의 제스처 명령어
Table 2. Thirteen types of gesture command in Quake II.




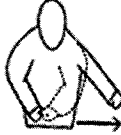
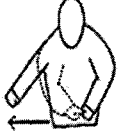


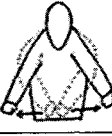


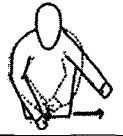


제스처 명령어	제스처 명령어	제스처 명령어	제스처 명령어	제스처 명령어
WF(Walk Forward)	BP(Back Pedal)	R(Run)	TL(Turn Left)	TR(Turn Right)
				
오른손을 앞으로 흔든다.	왼손을 뒤로 흔든다.	양손을 좌우로 흔든다.	왼손을 옆으로 뻗는다.	오른손을 옆으로 뻗는다.
J/U(Jump/Up)	DC(Down/Crouch)	CV(Center View)	LU(Look Down)	LD(Look Up)
				
양손을 위로 든다.	양손을 아래로 내린다.	양손을 옆으로 뻗는다.	고개를 왼쪽으로 숙인다.	고개를 오른쪽으로 숙인다.
SL(Step Left)	SR(Step Right)	AT(Attack)		
				
오른손은 아래로 왼손은 옆으로 뻗는다.	왼손은 아래로 오른손은 옆으로 뻗는다.	왼손을 앞으로 뻗는다.		

표 3. 포즈 정의
Table 3. Pose definition.

포즈 유형	제스처	심볼	포즈 묘사
Start/End pose	준비	R	양손을 가슴과 허리 사이에 둔 자세
Distinctive pose	공격(AT)	DA	왼손을 앞으로 뻗은 자세
	뒤로 걷기(BP)	DB	왼손만 머리 옆으로 올린 자세
	정면 보기(CV)	DC	양손을 수평으로 뻗은 자세
	앞으로 걷기(WF)	DW	오른손을 앞으로 뻗은 자세
	아래로 내려가다/몸을 숙이기(DC)	DB	양손을 아래로 내린 자세
	위로 올라가다/ 점프하기(J/U)	DU	양손을 머리위로 올린 자세
	달리기(R)	DR	양손이 오른쪽으로 휘둘러 올린 자세
	밑으로 보기(LD)	DLD	머리가 왼쪽으로 기우린 자세
	위로 보기(LU)	DLU	머리가 오른쪽으로 기우린 자세
	왼쪽으로 한 발짝 가기(SL)	DSL	왼손은 수평, 오른손은 아래로 뻗은 자세
	오른쪽으로 한 발짝 가기(SR)	DSR	오른손은 수평, 왼손은 아래로 뻗은 자세
	왼쪽으로 방향 틀기(TL)	DTL	왼손을 수평으로 뻗은 자세
	오른쪽으로 방향 틀기(TR)	DTR	오른손을 수평으로 뻗은 자세
Intermediate pose	정면 보기(CV)	IC	양손을 약간 수평으로 뻗은 자세
	뒤로 걷기(BP)	IB	왼손만 머리 뒤로 올린 자세
	달리기(R)	IR	양손이 왼쪽으로 휘둘러 올린 자세
	앞으로 걷기(WF)	IW	오른손을 오른 귀 옆으로 올린 자세
	위로 올라가다/ 점프하기(J/U)	IU	양손이 가슴까지 올린 자세
	왼쪽으로 방향 틀기(TL)	ITL	왼손을 수평으로 약간 뻗은 자세
	오른쪽으로 방향 틀기(TR)	ITR	오른손을 수평으로 약간 뻗은 자세
	왼쪽으로 한 발짝 가기(SL)	ISL	D _{SL} 자세에서 왼손이 약간 수평인 자세
	오른쪽으로 한 발짝 가기(SR)	ISR	D _{SR} 자세에서 오른손이 약간 수평인 자세

이 일어난다. 이러한 'rejection'은 사용자가 제스처를 불분명하게 수행하는 경우로, 사용자가 제스처 명령어에 익숙하도록 사전연습을 거친 후 실험한 경우 급격히 감소된다.

제안된 인식 방법의 효율성을 증명하기 위해 Lee와 Kim방법과 제안된 방법을 비교하였다^[12]. 이 방법은 제스처 분할 및 인식을 위해 임계값 모델이라 불리는 새로운 모델을 이용하였다. 이 모델은 최근 많은 연구자들에 의해 사용 및 참조되고 있다.

표 5는 두 방법의 비교 결과를 보여준다. 표에서 'Insert'와 'Delete'는 각각 비제스처를 제스처로 또는 제스처를 비제스처로 인식한 경우이고 'Substitute'는 어떤

제스처를 다른 제스처로 오인식하는 경우이다. 이 때, 'Detection'은 입력된 제스처와 정확히 인식된 제스처('Correct')의 비율로 정의되고, 'Reliability'는 제스처로 인식된 모든 경우와 'Correct'의 비율로 정의된다.

제안된 방법은 95.92 퍼센트의 추출률과 94.47 퍼센트의 신뢰도를 제공하고 있으며 Lee와 Kim의 방법은 95.31 퍼센트의 추출률과 94.83 퍼센트의 신뢰도를 제공한다. 또한 인식 속도를 비교했을 때, 제안된 방법은 0.073frame/sec의 속도를 가지는 데에 반해, Lee와 Kim의 방법은 0.218frame/sec 속도를 가진다.

결과적으로 제안된 방법은 정확한 인식률을 유지하면서, 계산시간을 크게 단축할 수 있다. 또한 제안된 방법

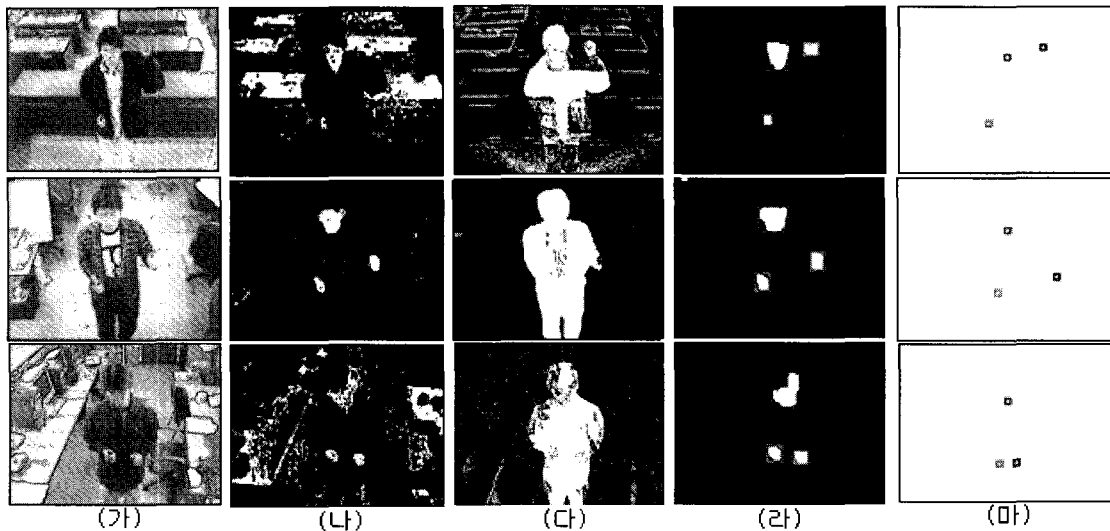


그림 4. 다양한 환경에서의 실험결과: (가) 원 영상, (나) 피부색 영역, (다) 움직임 영역, (라) 특징 벡터, (마) 분류된 포즈

Fig. 4. Experimental results for various environments: (a) an original image, (b) skin color regions, (c) moving regions, (d) a feature vector, and (e) a classified pose symbol.

표 4. 제스처 인식 결과

Table 4. Confusion matrix for the proposed method.

	AT	BP	CV	DC	I/U	LD	LU	R	SL	SR	TL	TR	WF	Reject
AT	94													6
BP		98												2
CV			98											2
DC				91										9
J/U					91									9
LD						92								8
LU							93							7
R								95			2	2		1
SL									98					2
SR										98		1		1
TL											98			2
TR												98		2
WF													98	2

표 5. 인식 결과 비교

Table 5. Comparison of recognition results.

명령어	제스처 수	방 법	결 과					
			Insert	Delete	Substitute	Correct	Detection(%)	Reliability(%)
AT	100	Lee et al ^[12]	2	6	0	92	92	90.20
		제안된 HMM	11	6	0	94	94	84.68
BP	100	Lee et al ^[12]	0	2	0	94	94	94.00
		제안된 HMM	0	2	0	98	98	98.00
CV	100	Lee et al ^[12]	0	2	0	97	97	97.00
		제안된 HMM	0	2	0	98	98	98.00
DC	100	Lee et al ^[12]	2	9	0	91	91	89.22
		제안된 HMM	5	9	0	91	91	86.67
J/U	100	Lee et al ^[12]	3	9	0	92	92	89.32
		제안된 HMM	6	9	0	92	92	86.79
LD	100	Lee et al ^[12]	0	8	0	92	92	92.00
		제안된 HMM	0	8	0	93	93	93.00
LU	100	Lee et al ^[12]	0	7	0	95	95	95.00
		제안된 HMM	0	7	0	93	93	93.00
R	100	Lee et al ^[12]	0	1	4	98	98	98.00
		제안된 HMM	0	1	4	98	98	98.00
SL	100	Lee et al ^[12]	0	2	0	98	98	98.00
		제안된 HMM	0	2	0	98	98	98.00
SR	100	Lee et al ^[12]	0	1	0	98	98	98.00
		제안된 HMM	0	1	1	98	98	98.00
TL	100	Lee et al ^[12]	0	2	0	98	98	98.00
		제안된 HMM	0	2	0	98	98	98.00
TR	100	Lee et al ^[12]	0	2	0	98	98	98.00
		제안된 HMM	0	2	0	98	98	98.00
WF	100	Lee et al ^[12]	0	2	0	96	96	96.00
		제안된 HMM	0	2	0	98	98	98.00

은 구현하기 쉽다는 장점을 가진다. 이러한 장점은 제안된 방법을 실제 게임 인터페이스로 적용하는 실험을 통해 증명되었다. 실험 결과는 제안된 방법이 게임시스템 뿐만 아니라 로봇 제어시스템과 같은 다양한 인터페이스로 사용될 수 있다는 가능성을 보여주고 있다.

VII. 결 론

본 논문에서는 새로운 HMM을 사용한 제스처 인식 방법을 제안하였으며, 이를 이용한 게임 인터페이스 시스템을 개발하였다. 제안된 방법은 사용자의 움직임에 대한 연속적인 스트림을 입력으로 받아서 제스처의 분할과 인식을 동시에 수행할 수 있는 새로운 구조의 HMM이다. 제안된 방법을 총 650개의 제스처 영상에 대해 평가해 본 결과 96.92%의 정확도를 보였다. 또한 다른 방법과의 성능비교를 통해 제안된 모델이 높은 정확도를 유지함과 동시에 속도를 많이 향상할 수 있음을

증명하였다.

그러나 제안된 방법은 몇 가지 성능개선을 필요로 한다. 그 중 하나는 현재시스템에서는 사용자의 움직임만을 통해 제스처를 인식하지만, 복잡한 시스템의 응용을 위해서는 움직임뿐만 아니라 형태정보의 사용을 필요로 한다. 향후 연구에서는 사용자의 움직임과 형태정보를 모두 고려한 인식기를 개발할 것이다.

참 고 문 헌

- [1] T. Frantti, S. Kallio, Expert system for gesture recognition in terminal's user interface, *Expert Systems with Applications*, Vol. 26, 189-202 (2004).
- [2] H. Kang, C.W. Lee, K. Jung, Recognition-based gesture spotting in video games, *Pattern Recognition Letters*, Vol. 25, 1701-1714, (2004).
- [3] W.N. Chan, S. Ranganath, Real-time gesture

- recognition system and application, *Image and Vision Computing*, Vol. 20, 993-1007, (2002).
- [4] P. Marco, Vision-based user interfaces: methods and applications, *International Journal of Human-Computer Studies*, Vol. 57, 27-73, (2002).
- [5] K. Oka, Y. Sato, H. Koike, Real-time tracking of multiple fingertips and gesture recognition for augmented desk interface systems, *Automatic Face and Gesture Recognition*, 411-416, (2002).
- [6] H.S. Yoon, J. Soh, Y.J. Bae, S.Y. Yang, Hand gesture recognition using combined features of location, angle and velocity, *Pattern Recognition*, Vol. 34, 1491-1501, (2001).
- [7] L. Gupta, S. Ma, Gesture-based interaction and communication: automated classification of hand gesture contours, *IEEE Transactions on Systems, Man and Cybernetics*, Part C, 114-120, (2001).
- [8] I. Cohen, N. Sebe, A. Garg, L.S. Chen, S.H. Thomas, Facial expression recognition from video sequences: temporal and static modeling, *Computer Vision and Image Understanding*, Vol. 91, 160-187, (2003).
- [9] T. Otsuka, J. Ohya, An HMM-based approach for off-line unconstrained handwritten word modeling and recognition, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 21, 752-760, (1999).
- [10] F. Quek, Toward a vision-based human gesture interface, *Conf. on Virtual Reality Software Technol.*, 17-31, (1994).
- [11] A. Kendon, Current issues in the study of gesture, *The Biological Foundation of Gestures: Motor and Semiotic Aspects. Lawrence Erlbaum Associate*, 23-47, (1986).
- [12] H.K. Lee, J.H. Kim, An HMM-based threshold model approach for gesture recognition, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 21, 961-973, (1999).

저 자 소 개



박 혜 선 (학생회원)
 2001년 영남대학교 철학과 학사
 2004년 경북대학교 컴퓨터공학과 석사
 현재 경북대학교 컴퓨터공학과 박사 수료생

김 은 이 (정회원)-교신저자
 전자공학회 제 41권 CI편 4호 참조

김 항 준 (정회원)
 전자공학회 제 41권 CI편 4호 참조

<주관심분야 : 인공지능, 패턴 인식, 신경망>