

확률적 정규 문법 추론법에 의한 사람 몸동작 인식

(Human Action Recognition by Inference of Stochastic Regular Grammars)

조 경 은 [†] 조 형 제 ^{††}
(Kyungeun Cho) (Hyungie Cho)

요 약 이 논문은 사람의 비언어적 행동을 자동적으로 분석하는 것을 목적으로 60 가지의 기본적인 사람의 윗몸 동작들을 인식하는 방법을 제안한다. 사람 몸동작을 인식하기 위한 방법으로는 확률적 문법 추론법을 이용하였으며 모든 관절의 움직임 분석으로 임의의 동작을 인식하는 방법을 사용하였다. 시스템의 입력 데이터로 쓰여지는 각 관절의 실세계 3 차원 좌표들을 일정간격으로 양자화한 후, 각각 xy , zy 평면에 투영하고, 이들을 다시 4 방향 코딩하여 확률적 문법 추론법에 적합한 입력형식으로 변환한다. 또한 비언어적 행동 분석을 위한 사람의 동작 인식에는 손과 다른 부위와의 관계인 근접 정보가 동작 구분의 중요한 요소가 됨을 감안하여, 확률 문법 추론 방법을 확장하고, 일반적인 확률 문법 추론 방법과 비교하여 인식률이 향상됨을 실험결과를 통해 확인하였다.

Abstract This paper proposes a human action recognition scheme to recognize nonverbal human communications automatically. Based on the principle that a human body action can be defined as a combination of multiple articulation movements, we use the method of inferencing stochastic grammars to understand each human actions. We measure and quantize each human action in 3D world-coordinate, and make two sets of 4-chain-code for xy and zy projection plane. Based on the fact that the neighboring information among articulations is an essential element to distinguish actions, we designed a new stochastic inference procedure to apply the neighboring information of hands. Our proposed scheme shows better recognition rate than that of other general stochastic inference procedures.

1. 서 론

최근 사람의 몸 동작 인식을 활용할 응용 분야가 급격히 증가하기 시작하였다. 사람의 걸음 동작을 분석하는 의학 분야, 비언어적 행동을 분석하는 사회심리학 분야, 아바타 제어를 필요로 하는 가상현실 분야, 사람-기계 상호작용이 필요한 자동화 분야, 보안용 감시 시스템 개발 분야, 수화 인식이 개발 분야, 발레 무도법이나 체조 동작을 분석해야하는 체육학 분야 등이 여기에 속하며[1][2][3], 일부 분야에서는 이미 사람 몸 동작 인식 시스템들이 구축되어 있다. 그 예가 중국의 체조식 권법

을 분석하는 시스템인 Sensei[4], 수화를 인식하고 번역해주는 ARGo[5] 등이다. 이 연구는 사람의 비언어적 의사소통 분석을 자동화하기 위한 것으로 사람의 몸 동작 인식 방법에 중점을 둔다. 의사소통은 사람들간에 감정, 태도, 사실, 생각을 전달하는 과정으로 상호 대화를 통해 서로에게 영향을 주고 서로를 이해하는 모든 수단을 포함한다. 언어적 행동뿐만 아니라 얼굴표정, 눈맞춤, 몸짓, 움직임, 자세와 공간적 거리를 포함하는 모든 비언어적 행동들이 상징적으로 사용되고 있다. 이러한 연구가 비언어적 행동을 분석하는 사회심리학 분야에서 이루어지고 있으며 이 중에서 의사 전달의 많은 비중을 차지하는 몸짓, 손짓에 관한 자동화 연구도 이루어지고 있는 추세이다[6][7]. 사람의 몸짓, 손짓을 자동으로 인식할 수 있으면 사람이 일일이 수동적으로 분석해야 할 작업들을 자동화시킬 수 있으며, 컴퓨터와 사람간의 편리한 통신 매체 수단으로서도 사용 가능하다. 이처럼 컴

[†] 학생회원 : 동국대학교 컴퓨터공학과
cke@dongguik.edu

^{††} 정 회 원 : 동국대학교 영상정보통신대학원 멀티미디어학과 교수
chobj@dongguik.edu

논문접수 : 2001년 3월 2일

심사완료 : 2001년 4월 3일

퓨터가 사람의 동작을 이해하면 좀 더 편리한 인터페이스와 환경을 제공받을 수 있는 것이다.

실질적으로 영상에서 사람 몸 동작을 인식하기 위해서는 영상에서 사람을 추적하고 사람을 배경에서 분리하는 작업을 수행한 후 각 몸 구성 성분을 추출하고 마지막으로 각 몸 동작 패턴을 정의하고 인식하는 등의 여러 단계가 필요하다. 사람의 몸 동작은 몸을 구성하는 성분(손, 아래팔, 위팔, 어깨, 머리, 몸통 등)들의 움직임으로 이루어진다. 다시 말해서 각 몸 구성 성분의 3차원 위치 정보로 움직인 경로를 유추하고 이들을 조합하여 사람의 몸 동작을 표현하게 된다[7]. 사람 동작 인식에 많이 사용되어진 접근 방법으로는 시간축의 비선형 신축에 의한 정합법 (Dynamic Time Warping : DTW) [8], 구문론적인 방법[9], 템플릿 매칭 방법[10], 퍼지 방법[11], 은닉마르코프 모델 (Hidden Markov Model: HMM)[3] 등이 있다.

구문론적 패턴인식은 패턴의 구조에 주목하는 인식 방법으로, 대상 패턴을 간단한 서브패턴으로 분해하고, 이 서브 패턴 사이에 존재하는 관계를 형식언어의 문법 이론으로 기술하여 인식하는 방법이다. 일반적인 구문론적 패턴 인식방법에서는 고려되는 모든 패턴이 소속되어야 할 문법을 정확하게 결정할 수 있을 때 제대로 인식된다고 볼 수 있다. 즉, 입력되는 모든 패턴은 어느 특정 클래스로 반드시 분류된다는 것을 의미한다. 그러나, 실제로 어떤 패턴의 서브패턴은 다른 서브패턴보다 더 빈번히 나타날 수도 있고 어떤 경우에는 오류 때문에 패턴의 심한 변형이 포함될 수도 있다. 제한된 수의 패턴 클래스가 있고 모든 클래스는 확실하게 구별되는 문법으로 표현되어지나, 입력 스트림에 포함될 수 있는 임의의 오류나 그것으로 생성된 언어들은 하나의 문법 클래스로 수렴할 수는 없게 되는 경우가 이에 해당된다. 이때 어떤 클래스에 더 가깝다는 확률값을 반영할 수만 있다면 확률을 문법에 적용하는 방법은 모호하게 분류되는 것을 극복할 수 있게 해준다. 사람의 각 몸 동작들은 같은 동작이더라도 변형이나 잠음, 관측계의 오차, 특징 추출의 불완전성 등으로 인하여 예기치 않은 변화 패턴들이 많이 포함될 수 있어 유일하게 하나의 문법으로 표현하기는 불가능하다. 이러한 성질을 가장 잘 반영해줄 수 있는 것이 확률적 문법이다[12][13].

기존의 알고리즘에서는 각종 동작을 패턴 해석적으로 패턴화, 문법화하여 직접 인식하는 연구가 수행되었다. 그러나 이러한 시스템은 모든 인식대상 마다 인위적으로 특징적 요소를 추출하고 패턴화하는 복잡하고 방대한 작업이 필수적이고, 특징 추출 방법과 패턴화 방법에

따라 인식기능이 크게 좌우되며, 사람에 따라 같은 동작이더라도 너무 다르게 표현되는 패턴나열 때문에 인식능력이 크게 변화하는 등 어려움이 많다. 이의 보완 기술로 컴퓨터가 표준 패턴을 자동으로 설계하여 지식을 축적하고, 미지의 기호를 축적된 지식에 적용하여 인식하도록 하는 연구도 활발히 진행되고 있으며 크게 구문론적인 해석에 의한 문법적 추론 방법이 이에 해당한다. 이러한 학습에 의한 인식방법은 인위적 패턴 설계 과정 없이 손쉽게 인식 시스템을 만들 수 있고, 학습을 통해 새로운 지식의 축적이 용이하다[12][13].

문법적 추론 방법과 확률적 문법의 두 가지 특징을 모두 만족시켜주는 인식 방법이 확률적 문법 추론 방법이다. 확률적 문법 추론 방법은 이미 정상 또는 비정상 염색체 인식 연구에 시도된 바 있다[12]. 사람 동작 인식에 확률적 문법을 사용한 기존 연구[9]에서는 확률 문법을 적용했다는 점에서 같으나 문법 추론이 불가능하다는 문제점이 제시되어졌다. 이 논문에서는 문법 추론 기능을 추가하여 이 문제점을 개선하였으며 실제 3 차원 좌표를 입력으로 하는 확률적 정규 문법 추론 방법을 이용하여 다양한 종류의 사람 동작들이 제대로 분류됨을 보임으로써 이 인식 방법의 타당성을 확인하였다. 또한 이 방법에 추가적으로 근접 정보를 모델링하여 보다 더 향상된 인식률을 제공하는 새로운 방법인 확률적 정규 문법 추론 방법에 근접 정보를 적용한 인식 방법을 제안한다. 인식 대상은 60 가지의 기본적인 사람 동작으로서 비언어적 행동 분석을 위한 윗몸 동작으로 인식 범위를 제한하였다.

이 논문의 구성은 다음과 같다. 2장에서는 일반적인 확률적 문법 추론 방법, 3장에서는 전체 시스템 구조도와 전처리 단계 그리고 이 논문에서 사람 몸 동작 인식에 적합하게 적용된 확률적 정규 문법 추론 방법에 대하여 기술하고 4장에서는 이 논문에서 제안하는 근접 정보를 이용하여 확장된 확률적 정규 문법 추론 방법을 소개한다.

2. 확률적 문법 추론

주어진 문법과 학습 패턴들을 가지고 각 문법의 생성 규칙에 해당하는 확률값을 구하는 것이 확률적 문법 추론(inference of stochastic grammars) 방식이라 할 수 있다. 주어진 학습 패턴들은 서로 다른 문법을 생성하는 패턴들과 각각의 문법에 해당하는 확률값으로 이루어져 있다. 이 장에서는 확률적 문법 추론을 인식에 이용하기 위해 각각의 생성규칙에 해당하는 확률값을 추정하는 방법을 설명한다.

일반적으로 각 생성규칙의 확률을 학습하는 방법인

확률적 문법 추론 방법[13]은 다음과 같다.

$G_{sk}=(N_k, \Sigma_k, P_k, D_k, S_k)$ 에서 $k=1, 2, \dots, M$ 인 확률적 문법으로 표시되어지는 M 개의 클래스로 분류를 해야하는 문제를 고려할 때 N_k 는 비종단 기호의 집합, Σ_k 는 종단 기호의 집합, P_k 는 확률적 생성 규칙의 집합, D_k 는 추정되어야 할 각 생성규칙의 확률값의 집합 그리고 S_k 는 시작 기호를 각각 의미한다.

여기서 N_k, Σ_k, P_k 그리고 S_k 는 이미 알고 있는 정보라고 가정하고 G_{sk} 는 모호하지 않은 문맥자유(context-free) 또는 정규문법(regular grammar)이다. 이 가정들을 바탕으로 학습 패턴 집합 $X=(x_1, x_2, \dots, x_m)$ 에서 $D_k(k=1, 2, \dots, M)$ 의 생성규칙들의 확률을 추정하는 것이 요구된다. 학습 패턴 집합 X 의 각 패턴들은 종단 기호들의 조합으로 이루어진 것으로 미리 정해진 확률적 문법들을 만족해야 한다. 각 생성규칙의 확률값의 집합인 D_k 를 추정하기 위해서는 G_{sk} 문법에서 각 생성 규칙들이 모든 학습 패턴 집합 X 의 생성에 이용되었을 확률(P_{kij})들을 구해야 한다. G_{sk} 문법에서 $A_i > \beta_j$ 이 모든 X 의 생성에 이용되었을 확률을 P_{kij} 라 하고 다음 수식과 같이 구할 수 있다.

$$\text{estimated } P_{kij} = \hat{P}_{kij} = \frac{n_{kij}}{\sum_{k=1}^M n_{kij}} \quad (1)$$

수식(1)의 n_{kij} 는 모든 학습 패턴에 대해서 G_{sk} 문법에서 하나의 생성규칙인 $A_i > \beta_j$ 가 사용되었던 총 평균 횟수를 의미하며 다음 수식에 의해서 구하게 된다.

$$n_{kij} = \sum_{x_h \in X} n(x_h) p(G_{sk}/x_h) N_{kij}(x_h) \quad (2)$$

수식(2)에서 각 항목이 의미하는 바는 다음과 같다. $n(x_h)$ 는 학습 패턴 집합의 모든 패턴에 대해서 이들이 학습 패턴 집합에서 발생한 빈도수를 의미한다. $N_{kij}(x_h)$ 는 하나의 패턴 x_h 를 파싱할때 G_{sk} 문법의 하나의 생성규칙인 $A_i > \beta_j$ 가 사용되었던 횟수를 나타낸다. 비록 문법들의 각 생성규칙들에 해당하는 확률은 알고 있지 않지만, 사용하는 생성규칙들은 이미 알고 있기 때문에 이 파싱 단계가 가능한 것이다. $p(G_{sk}/x_h)$ 는 하나의 패턴 x_h 가 G_{sk} 문법으로부터 생성되었을 확률을 의미한다. 수식(1)의 $\sum_{k=1}^M n_{kij}$ 는 $A_i > \beta_j$ 의 경우에 대한 총합을 의미하며 G_{sk} 문법에서 같은 left part nonterminal A_i 를 가지는 모든 생성규칙에 대하여 위의 방법으로 n_{kij} 를 계산하여 총합을 구한다.

모든 문법의 생성 규칙에 해당하는 확률을 표현하는 집합 D_k 가 구해지면 확률적 문법의 추론 단계인 학습 단계가 완료되고 테스트 단계를 수행할 수 있다. 입력되는 각각의 테스트 패턴은 모든 문법 클래스에 대해서

각 패턴에 해당하는 확률값을 구하고 이 중에서 가장 높은 확률값을 갖는 문법 클래스로 분류된다[13].

3. 확률적 정규문법 추론을 이용한 사람 몸 동작 인식

이 장에서는 사람의 몸 동작 인식에 확률적 문법 추론 방식을 적용한 방법을 자세히 기술한다. 3.1절에서는 이 논문에서 구현한 전체 시스템의 구성도와 확률적 문법 추론기의 입력으로 사용되어질 입력 데이터들에 대한 전처리 단계를 설명한다. 3.2절에서는 확률적 정규문법 추론기에서 필요로 하는 정규문법 생성 방법과 인식에 사용하게 될 확률 테이블 생성에 관하여 상세히 기술한다.

3.1 전체 시스템 구성도와 전처리 단계

전체 시스템은 학습 단계를 수행하는 확률적 문법 추론기와 테스트 단계를 수행하는 파서로 구성되어있다. 그림 1은 이 논문에서 제시한 사람 몸 동작 패턴 인식기의 전체 시스템 구성도를 나타내며 전체 시스템은 크게 전처리 단계, 파서, 확률 문법 추론기로 이루어져 있다. 그림 2는 전처리 단계와 학습 단계를 나타낸다. 이 논문에서 제안한 인식기는 측정된 실세계 3차원 좌표 자료를 양자화한 후 각각 xy, zy 평면에 투영한 값을 구하고 4방향 코딩하는 전처리 단계를 거친 데이터를 입력으로 한다. 학습 데이터들은 확률적 문법 추론기에 입력되고 미리 정의된 정규문법을 이용하여 16개의 확률 테이블을 결과물로 생성한다. 테스트 데이터들은 파서에 입력되는데 이들은 파싱되면서 사전에 구한 확률 테이블의 정보를 가지고 가장 높은 확률값을 갖는 문법 클래스를 찾는다. 자세한 내용은 다음 각 단계별로 설명한다.

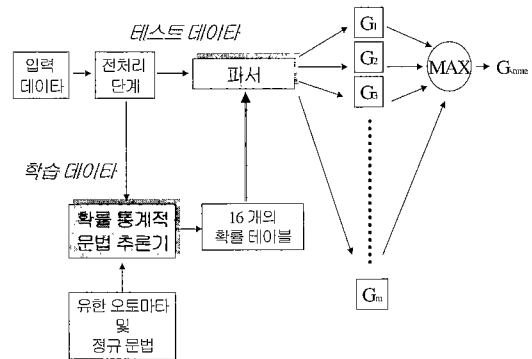


그림 1 전체 시스템 구성도

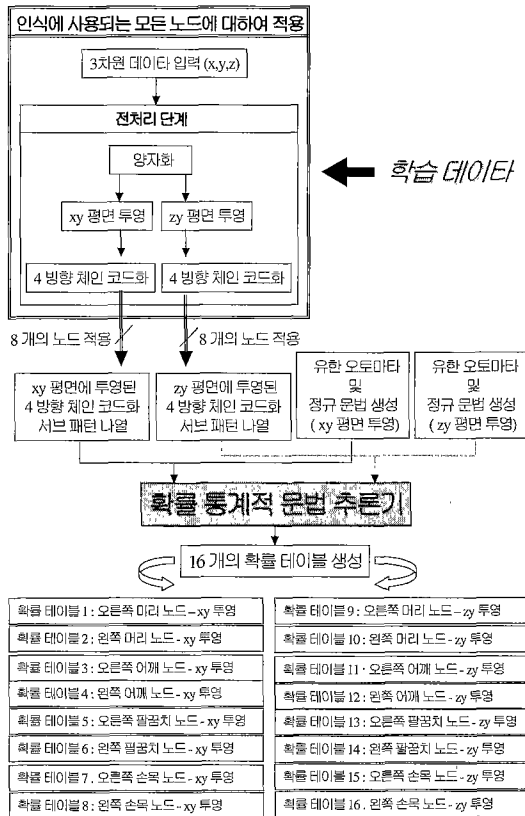


그림 2 전처리 단계와 학습 단계

(1) 입력 데이터와 양자화

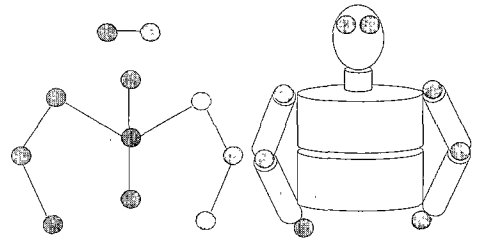
인식기의 입력 데이터를 생성하기 위해서 칼라마크를 부착한 사람을 추적하여 각 관절의 3차원 좌표를 출력해주는 STABIL++ 시스템을 이용한다. 그림 3(a)는 실제 STABIL++ 시스템을 이용하여 동작을 녹화하는 장면이다. STABIL++ 시스템의 출력 결과로서 그림 3(b)의 왼쪽 그림처럼 각 관절과 몸통 그리고 머리 부분에 대해서 칼라마크 총 11개에 대해서 3차원 좌표를 얻을 수 있다. 몸통에 부착된 목, 가슴, 배의 세 마크는 위치 이동만 조금 있을 뿐 각 관절에 붙은 마크처럼 동작의 변화에 큰 영향을 미치지 않는다. 따라서 이 세 마크들을 포함시키지 않아도 60가지의 사람 윗몸 동작들을 구분할 수가 있으므로 실제 패턴인식기의 입력 데이터는 그림 3(b)의 오른쪽 모델에 부착된 8개 마크만을 사용한다. 이 8개의 몸 구성성분의 데이터는 실제계 좌표계에서의 x, y, z 값이다. 각 몸 구성성분의 관절에

대한 x, y, z 좌표들은 3cm 간격으로 양자화(quantization)하여 사용된다. 양자화를 하는 이유는 너무 작은 움직임을 무시하기 위해서이다.



(a)

STABIL++ 시스템의 출력 데이터 패턴 분류기의 입력 데이터

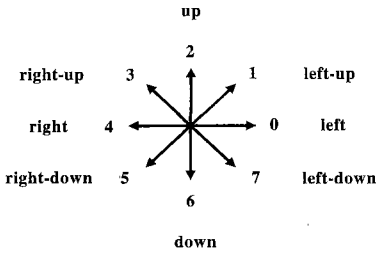


(b)

그림 3 실제 녹화 장면(a)과 입력 데이터 모델(b)

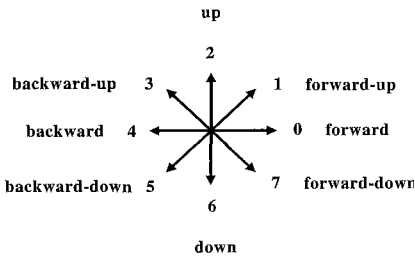
(2) 양자화된 데이터의 평면 투영과 4방향 체인 코드화 3cm 간격으로 양자화되어 코딩된 데이터에 대하여 각각 xy, zy 평면에 투영한 결과를 구한다. 이 단계를 수행하는 이유는 사람이 정면과 측면에서 사람의 동작을 관찰한 것과 같은 결과를 유추하여 인식에 적용하기 위해서이다. 양자화된 데이터를 각각 xy, zy 평면에 투영한 값으로 변형한 후 이 결과를 8방향 체인 코드화한다. 투영한 평면에 따라 각각 의미하는 바는 다르다. 즉, xy 평면에 투영한 데이터 결과에서의 8방향 체인 코드화는 그림 4(a)에서 볼 수 있듯이 차례로 left, left-up, up, right-up, right, right-down, down의 의미가 부여되고 이는 사람의 동작을 정면에서 관찰한 것과 같다. zy 평면에 투영한 데이터에 대한 8 방향 코드는 그림 4(b)에 나타난다. 그림 4(c)는 왼쪽 팔을 앞으로 돌린 동작에서 왼쪽 손의 움직임을 나타낸 결과 그림이다. 그림 4(c)의

xy plane (앞)

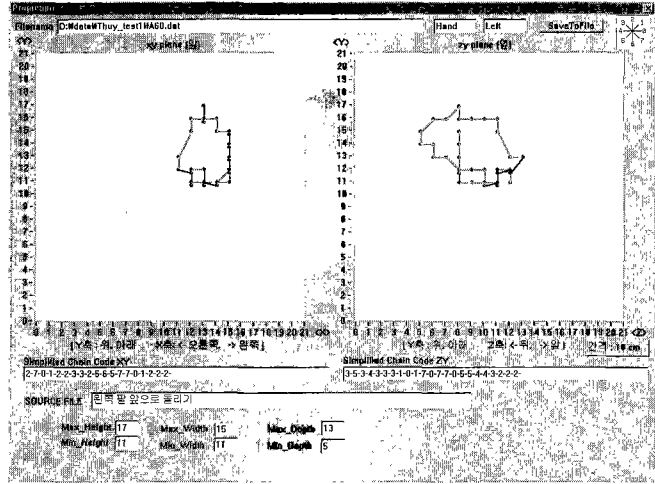


(a) xy 평면 8 방향 코드

zy plane (옆)



(b) zy 평면 8 방향 코드



(c) xy, zy 평면 투영 및 8방향 체인 코드화한 예

그림 4 8방향 코드와 평면 투영 후 8방향 체인 코드화 결과

왼쪽 창이 xy 평면에 투영한 결과를, 그리고 오른쪽 창이 zy 평면에 투영한 결과를 8방향 코드화한 후 화면에 출력한 것이다. 즉, 사람의 동작을 각각 앞, 옆에서 관찰한 결과와 같다.

실제로 인식기에 입력되는 서브 패턴나열들은 8방향 체인 코드화한 결과에서 사전으로 나타나는 모든 코드들을 두 개의 코드로 분리하게 된다. 4방향 코드화를 하는 이유는 문법을 간결화하기 위해서이다. 생성규칙이 너무 많을 경우에는 다음 상태로 전이하게 되는 경우가

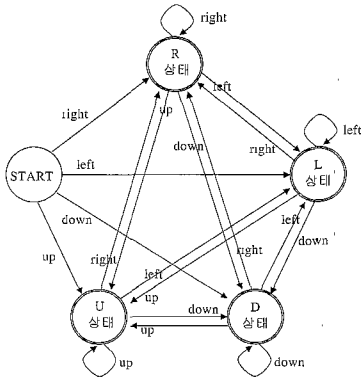
너무 많아져 결과적으로 인식률을 감소시키게 될 위험성이 있기 때문이다. 서브 패턴나열의 예를 표 1에서 살펴볼 수 있다. 이 서브 패턴 나열은 왼쪽 팔을 앞으로 돌린 동작에서 왼쪽 손에 대한 예를 보여주며, 각각 xy 평면과 zy 평면으로 투영한 결과를 나타낸다. 이 서브 패턴 나열은 본 논문에서 사용한 확률적 정규 문법 추론기의 입력 형식이다. 서브 패턴 나열의 구조는 [코드 코드갯수] 와 같이 표시된다. 코드는 각각의 방향코드를 의미하며 코드갯수는 그 코드가 반복해서 나타난 수를 의미한다.

표 1 왼쪽 팔을 돌린 동작에서 왼쪽 손에 대한 서브 패턴 나열-확률적 정규문법 추론기의 입력 형식

3.2 확률적 정규문법 추론을 이용한 사람 몸 동작 인식 방법

node=왼쪽 손 plane=xy	left 12 up 11 right 4 up 5 right 9 down 13 left 2 down 1 left 5 down 2 left 3 up 13 left 1 up 1 #
node=왼쪽 손 plane=zy	backward 23 up 16 forward 23 down 14 backward 6 down 2 backward 8 up 7 backward 2 up 1 backward 6 down 2 backward 15 up 3 backward 2 up 5 backward 2 up 9 forward 2 up 1 forward 2 up 1 forward 15 down 1 forward 2 down 2 forward 4 down 10 backward 24 up 2 backward 2 up 13 forward 3 up 2 forward 22 down 12 forward 2 down 5 #

전처리 단계를 거친 입력 자료들은 이 절에서 설명하게 되는 인식기의 입력이 된다. 이 논문에서는 확률적 정규 문법 추론으로 사람 몸 동작 인식기를 구축하였다. 정규문법을 사용하는 이유는 사람의 동작을 정규문법만으로도 충분히 표현할 수가 있기 때문이다. 실제 x, y, z 입력 데이터를 양자화한 후 xy 평면으로 투영하고 4방향 코드화한 학습 패턴들을 파싱해주는 유한오토마타의 상태전이도(그림 5(a))와 이로부터 파생된 정규문법(그림 5(b))은 다음과 같다.



(a) 유한오토마타 M_{xy} 의 상태 전이도

$G_{xy} = \{S, R, L, U, D\}, \{right, left, up, down\}, P, S$ with productions

S → right R	R → right R	L → right R
S → left L	R → left L	L → left L
S → up U	R → up U	L → up U
S → down D	R → down D	L → down D
	R → right	L → right
	R → left	L → left
	R → up	L → up
	R → down	L → down
	U → right R	D → right R
	U → left L	D → left L
	U → up U	D → up U
	U → down D	D → down D
	U → right	D → right
	U → left	D → left
	U → up	D → up
	U → down	D → down

(b) 파생된 정규문법 G_{xy}

그림 5 유한오토마타 M_{xy} 의 상태 전이도와 파생된 정규문법 G_{xy}

2장에서 설명한 확률적 문법 추론 방법을 이용하여 학습 패턴 집합의 모든 패턴에 대해서 앞에서 정의한 정규문법의 모든 생성규칙에 해당하는 확률값을 추정한다. M개의 사람 몸 동작을 구분하는 M개의 문법을 다음과 같이 표시한다. 이것은 xy 평면에 투영한 학습 패턴 집합에 대한 문법이다.

$$G_{s1} = (N, \Sigma, P, D_1, S)$$

$$G_{s2} = (N, \Sigma, P, D_2, S)$$

⋮

$$G_{sm} = (N, \Sigma, P, D_m, S)$$

$$N = \{S, R, L, U, D\}$$

$$\Sigma = \{right, left, up, down\}$$

P = 유한오토마타 M_{xy} 에 의해서 파생된 정규문법의 생성규칙들

D_k ~ 추정되어야 할 각 생성규칙의 확률값

S = 시작 상태

zy 평면에 투영한 학습 패턴 집합에 대한 문법에서는 다음 항목만 바뀐다.

$$N = \{S, B, F, U, D\}$$

$$\Sigma = \{backward, forward, up, down\}$$

P = 유한오토마타 M_{zy} 에 의해서 파생된 정규문법의 생성규칙들

실질적으로 각 생성규칙에 해당하는 확률은 다음 표 2에서와 같이 구한다. 생성 규칙들은 확률을 표기하는 방법 P_{kij} 을 따르기 위해서 정의한 대치규칙에 의해서 대치된다. G_{sk} 문법에서 각 생성 규칙들이 모든 학습 패턴 집합 X의 생성에 이용되었을 확률을 P_{kij} 로 표기한다.

다. 여기서 첨자 k는 소속 문법 클래스, 첨자 i는 생성 규칙의 좌측향(left-hand side)의 첨자 그리고 첨자 j는 생성 규칙의 우측향(right-hand side)의 첨자를 나타낸다. 각 생성규칙에 해당하는 확률값을 구하는 방법은 2장에서 설명한 방법과 같다. 실제적으로 추정되어진 확률값에 대한 예를 표 2의 D_{44} 열에서 보여준다. D_{44} 는 학습 패턴 집합에서 왼쪽 팔을 앞으로 돌린 동작들을 학습한 후에 추정되어진 생성규칙들의 확률값을 나타낸다. 이 예는 왼쪽 팔을 앞으로 돌린 동작에서 왼쪽 손의 움직임 xy 평면에 투영하여 얻어진 학습 패턴에 대해 추정되어진 확률값을 나타낸다.

앞에서 설명된 확률적 정규문법 추론 단계는 동작 인식에 사용하는 모든 노드에 대해서, 그리고 각각 xy, zy 평면으로 투영한 경우에 대해서 진행된다. 그리하여 확률적 정규 문법 추론의 결과로 생성되는 확률 테이블은 그림 2와 같이 총 16개가 생성된다. 생성된 확률 테이블들은 나중에 테스트 단계의 인식 알고리즘에서 이용된다. 그림 6은 전체 테스트 단계를 진행하는 파서를 설명한 그림이다. 테스트하게 되는 데이터가 입력되면 전처리를 거친 후 각각의 노드와 각 평면에 대하여 서브 패턴 나열을 구하게 된다. 여기서 각 노드가 의미하는 바는 3.1 절에서 설명한 바와 같다. 이 서브 패턴 나열들은 파서로 입력되어지며 이때 유사 움직임 노드 판별 분류기를 이용하여 테스트 패턴의 움직인 노드들을 검사하게 된다. 이는 확률값을 계산하게 되는 범위를 같은 움직임 정도를 지니는 동작거리 제한하기 위해서이다. 왜냐하면 임의의 동작에서 2개의 노드가 움직인 경우와

표 2 각 생성규칙에 해당하는 확률표

	생성규칙	대치된 생성규칙	D ₁	D ₂		D ₄₄		D _m	[대치 규칙]
1	S -> right R	A1 -> β1	P ₁₁₁	P ₂₁₁		0.12217		P _{m11}	A1 = S A2 = R A3 = L A4 = U A5 = D
2	S -> left L	A1 -> β2	P ₁₁₂	P ₂₁₂		0.56045		P _{m12}	
3	S -> up U	A1 -> β3	P ₁₁₃	P ₂₁₃		0.30227		P _{m13}	
4	S -> down D	A1 -> β4	P ₁₁₄	P ₂₁₄		0.01511		P _{m14}	
5	R -> right R	A2 -> β1	P ₁₂₁	P ₂₂₁		0.69052		P _{m21}	
6	R -> left L	A2 -> β2	P ₁₂₂	P ₂₂₂		0.03151		P _{m22}	β1 = right R β2 = left L β3 = up U β4 = down D
7	R -> up U	A2 -> β3	P ₁₂₃	P ₂₂₃		0.16288		P _{m23}	
8	R -> down D	A2 -> β4	P ₁₂₄	P ₂₂₄		0.10734		P _{m24}	
9	R -> right	A2 -> β5	P ₁₂₅	P ₂₂₅		0.00694		P _{m25}	
10	R -> left	A2 -> β6	P ₁₂₆	P ₂₂₆		0.00053		P _{m26}	
11	R -> up	A2 -> β7	P ₁₂₇	P ₂₂₇		0		P _{m27}	
12	R -> down	A2 -> β8	P ₁₂₈	P ₂₂₈		0.00027		P _{m28}	
13	L -> right R	A3 -> β1	P ₁₃₁	P ₂₃₁		0.00829		P _{m31}	
14	L -> left L	A3 -> β2	P ₁₃₂	P ₂₃₂		0.74862		P _{m32}	
15	L -> up U	A3 -> β3	P ₁₃₃	P ₂₃₃		0.12081		P _{m33}	
16	L -> down D	A3 -> β4	P ₁₃₄	P ₂₃₄		0.09496		P _{m34}	
17	L -> right	A3 -> β5	P ₁₃₅	P ₂₃₅	...	0.00065	...	P _{m35}	
18	L -> left	A3 -> β6	P ₁₃₆	P ₂₃₆		0.00618		P _{m36}	
19	L -> up	A3 -> β7	P ₁₃₇	P ₂₃₇		0.0174		P _{m37}	
20	L -> down	A3 -> β8	P ₁₃₈	P ₂₃₈		0.00309		P _{m38}	
21	U -> right R	A4 -> β1	P ₁₄₁	P ₂₄₁		0.07366		P _{m41}	
22	U -> left L	A4 -> β2	P ₁₄₂	P ₂₄₂		0.0362		P _{m42}	
23	U -> up U	A4 -> β3	P ₁₄₃	P ₂₄₃		0.86688		P _{m43}	
24	U -> down D	A4 -> β4	P ₁₄₄	P ₂₄₄		0.02035		P _{m44}	
25	U -> right	A4 -> β5	P ₁₄₅	P ₂₄₅		0		P _{m45}	
26	U -> left	A4 -> β6	P ₁₄₆	P ₂₄₆		0.00047		P _{m46}	
27	U -> up	A4 -> β7	P ₁₄₇	P ₂₄₇		0.00174		P _{m47}	
28	U -> down	A4 -> β8	P ₁₄₈	P ₂₄₈		0.00071		P _{m48}	
29	D -> right R	A5 -> β1	P ₁₅₁	P ₂₅₁		0.00736		P _{m51}	
30	D -> left L	A5 -> β2	P ₁₅₂	P ₂₅₂		0.05008		P _{m52}	
31	D -> up U	A5 -> β3	P ₁₅₃	P ₂₅₃		0.00908		P _{m53}	
32	D -> down D	A5 -> β4	P ₁₅₄	P ₂₅₄		0.87996		P _{m54}	
33	D -> right	A5 -> β5	P ₁₅₅	P ₂₅₅		0.00019		P _{m55}	
34	D -> left	A5 -> β6	P ₁₅₆	P ₂₅₆		0.00038		P _{m56}	
35	D -> up	A5 -> β7	P ₁₅₇	P ₂₅₇		0.00038		P _{m57}	
36	D -> down	A5 -> β8	P ₁₅₈	P ₂₅₈		0.05257		P _{m58}	

또 다른 임의의 동작에서 8개의 노드가 움직인 경우에는 각 동작의 최종 확률값이 비교 대상이 될 수 없기 때문이다. 따라서 모든 동작에 대해서 확률값을 구하여 가장 최대값을 구하는 방법을 사용하지 않고 유사한 움직임 정도를 지니는 동작들의 집합 내에서만 최종적으로 계산된 확률값 중에서 최대값을 구하는 방법을 사용한다. 이를 처리해주는 단계가 유사 움직임 노드 판별

분류기이다. 유사 움직임 노드 판별은 현재 입력되어진 테스트 패턴의 움직인 노드들과 학습 테이블에 학습된 각 클래스의 움직임 노드들의 평균 움직임을 비교하여 그 차이가 어느 임계치 이내인 학습 패턴만을 비교대상 집합으로 둔다. 우리의 실험에서는 그 임계치를 1로 두었다. 즉, 움직임 노드의 종류 차가 한 개인 것까지 비교 대상으로 둔 것이다.

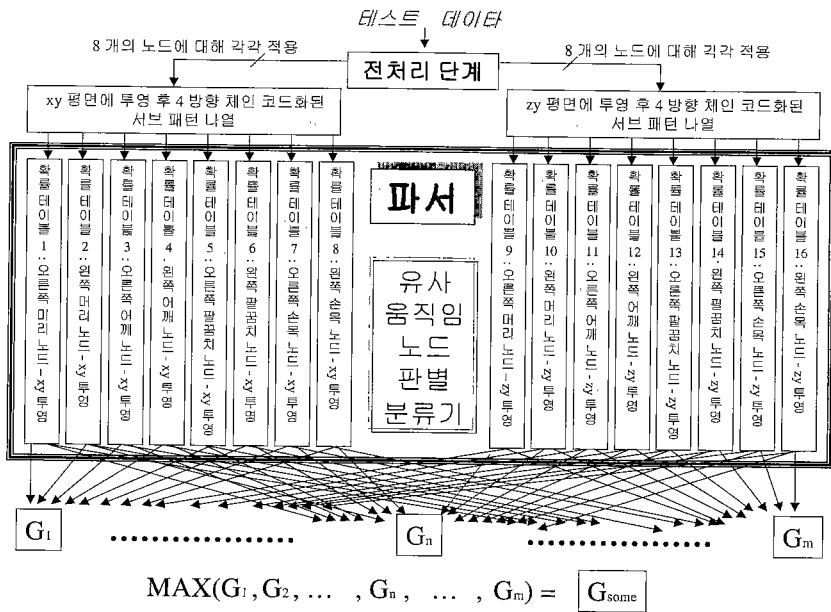


그림 6 파서의 구조.

지금까지 확률적 문법 추론 방법을 사용하여 다양한 사람의 윗몸 동작을 인식하는 방법을 소개하였다. 일반적으로 연구되어져왔던 손 동작 인식의 경우와는 달리 비언어적 행동 분석을 위한 사람의 윗몸 동작 인식에는 손과 다른 부위와의 관계인 근접 정보가 동작 구분의 중요한 요소가 된다[6] [7]. 그리하여 손의 근접정보를 문법에 적용할 수 있는 새로운 방법을 고안하여 확률적 문법 추론에 근접 정보를 적용한 인식 방법을 제안한다. 다음 장에서 확률적 정규문법 추론에 근접 정보를 적용한 사람 몸 동작 인식 방법을 기술한다.

4. 확률적 정규문법 추론에 근접 정보를 이용한 사람 몸 동작 인식 방법

이미 앞에서도 언급했듯이 비언어적 행동 분석에 있어서 많은 비중을 두는 것이 사람의 손이 자신 몸의 어느 부위에 근접했는 지에 관한 정보이다. 그러나 일반적인 확률적 문법 추론 방식의 인식 구조로는 이 특징을 반영해줄 수가 없기 때문에 이를 반영해줄 수 있는 새로운 인식기 구조를 제안한다. 이 장에서는 양손의 근접 정보를 문법에 적용한 방법을 자세히 설명하고자 한다. 3장에서 고려한 확률적 문법 추론기를 확장하여 근접 정보를 표시할 수 있는 구조로 변형하였다. 그림 7은 근접 정보를 구하게 되는 부위를 나타낸다. 머리, 배, 왼쪽

어깨, 왼쪽 위팔, 왼쪽 아래팔, 왼쪽 손, 오른쪽 어깨, 오른쪽 위팔, 오른쪽 아래팔, 오른쪽 손이 근접정보를 구하게 되는 10가지 부위이다. 따라서 이전에 정의된 모든 생성규칙들은 각각 11가지의 문법으로 확장하게 된다. 10개의 경우는 임의의 어느 부위를 근접했을 경우이고 나머지 한 경우는 임의의 어느 부위도 근접하지 않은 경우의 문법이다. 어느 일정 부위에 근접했다고 판정되는 경우는 각각 왼손, 오른손이 어느 부위에서 10cm 이내인 경우로 두었다. 만약 여러 군데가 동시에 10cm 이내인 경우가 발생했을 때에는 그 중에서 가장 거리가

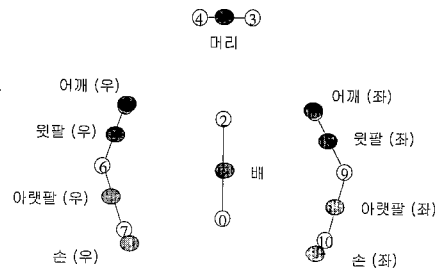


그림 7 근접 정보

최소인 것을 근접한 것으로 간주한다. 최소 거리를 10cm 라고 둔 이유는 근접 정보를 구하게 되는 손의 위치와 거리를 측정할 목적 위치의 오차를 각각 5cm로 두었기 때문이다. 입력 자료를 근접 정보가 포함된 형태로 변형 시키고 이를 양자화하고 평면 투영한 후 4방향 체인 코드화한 서브 패턴나열의 한 예가 다음 표 3에 나타난다. 이것은 오른쪽 손으로 머리를 접촉한 동작을 표현한 것으로 오른쪽 손의 서브 패턴 나열을 보여준 예이다. 서브 패턴 나열의 구조는[코드 근접정보]⁺와 같이 표시된다. 코드는 각각의 방향코드를 의미하며 근접 정보는 0에서 10까지의 값으로 표현된다. 다음 예에서 볼 수 있듯이 근접 정보가 0, 5, 4, 1, 0으로 변화되면서 코딩이 됨을 관찰할 수 있다. 이는 오른쪽 손이 어깨, 머리, 배순으로 근접했음을 알 수 있다.

표 3 오른쪽 손으로 머리 접촉 동작에서 오른쪽 손에 대한 서브 패턴 나열-근접 정보를 이용한 확률적 정규문법 추론기의 입력 형식

node=오른쪽 손 plane=xy	up 0 up 0 up 0 up 0 up 0 up 0 up 0 up 0 left 5 left 5 left 4 left 4 left 4 down 5 down 1 left 1 down 0 right 0 right 0 right 0 right 0 down 0 right 0 down 0 right 0 down 0 down 0 0 *
node=오른쪽 손 plane=zy	up 0 up 0 backward 0 backward 0 up 0 up 0 up 0 up 0 up 0 backward 0 backward 0 backward 5 up 5 up 4 backward 4 backward 4 down 5 down 1 backward 1 backward 1 backward 1 forward 1 forward 1 forward 0 forward 0 forward 0 forward 0 down 0 down 0 down 0 forward 0 forward 0 down 0 down 0 forward 0 forward 0 *

근접 정보를 기존의 문법에 적용한 결과가 다음과 같다. 그림 8은 xy 평면에 투영한 데이터에 대한 유한오토마타를 나타내고 그림 9는 이 유한오토마타에서 파생된 정규문법을 나타낸다. 이 정규 문법은 개수가 너무 많아서 단축된 형태로 표시하였다. 그림 8이나 그림 9에서 (1_11)로 표시된 내용은 하나의 심볼이 11가지 종류임을 나타낸다. 이는 근접 정보를 이용하여 입력심볼이 11가지로 구분되기 때문이다.

```

S_state -> right(1-11) R_state
S_state -> left(1-11) L_state
S_state -> up(1-11) U_state
S_state -> down(1-11) D_state
=> 44 개

R_state -> right(1-11) R_state
R_state -> left(1-11) L_state
R_state -> up(1-11) U_state
R_state -> down(1-11) D_state
=> 88 개

L_state -> right(1-11) R_state
L_state -> left(1-11) L_state
L_state -> up(1-11) U_state
L_state -> down(1-11) D_state
=> 88 개

U_state -> right(1-11) R_state
U_state -> left(1-11) L_state
U_state -> up(1-11) U_state
U_state -> down(1-11) D_state
=> 88 개

D_state -> right(1-11) R_state
D_state -> left(1-11) L_state
D_state -> up(1-11) U_state
D_state -> down(1-11) D_state
=> 88 개
    
```

그림 9 유한오토마타에서 파생된 정규문법

위에서 정의된 정규문법은 2장에서 언급된 확률적 문법 추론 단계와 같은 방법으로 인식기 구축에 적용된다. 각각의 확률테이블은 근접 정보를 포함한 생성규칙으로 표현되므로 396 개의 생성규칙이 포함된 확률 테이블로 확장되어진다.

5. 실험 방법 및 결과

이 논문에서는 상반신의 사람 동작을 STABIL++ 시스템으로 녹화한 후 이로부터 필요한 자료를 받아 실험이 행해졌다. 실험은 세 사람에게 대해서 표 4에 나열한 서로 다른 동작 60가지를 중복되게 한 사람 당 100여 개를 세 번씩 녹화하여 행하였다. 실제 비언어적 행동 분석을 위한 인식 데이터는 머리 동작, 몸통 동작, 팔 동작, 복합동작으로 이루어져 있다. 이 동작들의 종류는 비언어적 행동 분석 연구[6]의 자료를 참고로 하면서 결정하였다. 연구[6]에서는 사람의 비언어적 행동을 위해서 인식되어야 할 동작들을 자세히 언급하였다. 우리는 여기서 윗몸 동작에 대해서만 동작들을 인식 범위에 포함시켰다. 실험 입력 자료는 총 9세트(세 사람이 각각 세 번 실험한 자료)로서 5세트(500여 개의 데이터)는 학습자료, 4세트(400 여 개의 데이터)는 테스트 자료로서 사용하였다.

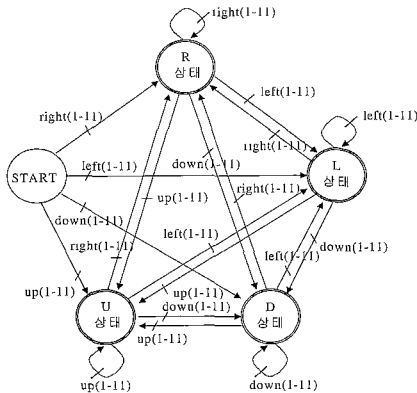


그림 8 근접 정보를 적용한 유한 오토마타의 상태 전이도

표 4 사람 윗몸 동작 인식에 사용한 동작 종류

[머리] 머리를 숙이고 제자리	[오른팔] 오른쪽 팔 뒤로 돌리기
[머리] 머리를 위로 올리고 제자리	[왼팔] 왼손으로 머리 접촉
[머리] 머리를 오른쪽으로 돌리고 제자리	[왼팔] 왼손으로 몸통(배) 접촉
[머리] 머리를 왼쪽으로 돌리고 제자리	[왼팔] 왼손으로 오른쪽 어깨 접촉
[머리] 머리를 오른쪽으로 꺾고 제자리	[왼팔] 왼손으로 오른쪽 위팔 접촉
[머리] 머리를 왼쪽으로 꺾고 제자리	[왼팔] 왼손으로 오른쪽 아래팔 접촉
[머리] 머리를 오른쪽 방향으로 회전하기	[왼팔] 왼손으로 오른손 접촉
[머리] 머리를 왼쪽 방향으로 회전하기	[왼팔] 왼손으로 왼쪽 어깨 접촉
[몸통] 오른쪽 뒤로 어깨와 함께 보고 제자리	[왼팔] 왼쪽 어깨 앞으로 돌림 (팔도 약하게 움직임)
[몸통] 왼쪽 뒤로 어깨와 함께 보고 제자리	[왼팔] 왼쪽 어깨 뒤로 돌림 (팔도 약하게 움직임)
[몸통] 몸통을 오른쪽으로 꺾고 제자리	[왼팔] 왼쪽 팔 앞으로 위로 올리고 내리기
[몸통] 몸통을 왼쪽으로 꺾고 제자리	[왼팔] 왼쪽 팔 옆으로 위로 올리고 내리기
[몸통] 몸통을 오른쪽으로 돌리고 제자리	[왼팔] 왼쪽 팔 뒤로 위로 올리고 내리기
[몸통] 몸통을 왼쪽으로 돌리고 제자리	[왼팔] 왼쪽 팔 앞으로 돌리기
[몸통] 몸통을 앞으로 구부리고 제자리	[왼팔] 왼쪽 팔 뒤로 돌리기
[몸통] 몸통을 뒤로 기대고 제자리	[양팔] 어깨 여러 번 올리고 내림
[몸통] 몸통을 좌우로 흔들기	[양팔] 양쪽 어깨 앞으로 돌림 (팔도 약하게 움직임)
[오른팔] 오른손으로 머리 접촉	[양팔] 양쪽 어깨 뒤로 돌림 (팔도 약하게 움직임)
[오른팔] 오른손으로 몸통(배) 접촉	[양팔] 양손 위로 올리고 내리기 (팔꿈치 움직이지 않기)
[오른팔] 오른손으로 왼쪽 어깨 접촉	[양팔] 손을 책상 위에서 모으고 제자리
[오른팔] 오른손으로 왼쪽 위팔 접촉	[양팔] 손이 턱 근처에서 모아짐. 그리고 제자리
[오른팔] 오른손으로 왼쪽 아래팔 접촉	[양팔] 팔짱끼고 제자리
[오른팔] 오른손으로 왼손 접촉	[양팔] 팔을 교차하고 제자리
[오른팔] 오른손으로 오른쪽 어깨 접촉	[복합] 머리를 좌우로 흔들면서 동시에 팔짱끼는 동작
[오른팔] 오른쪽 어깨 앞으로 돌림 (팔도 약하게 움직임)	[복합] 몸통을 앞으로 기대면서 양손으로 턱을 받침
[오른팔] 오른쪽 어깨 뒤로 돌림 (팔도 약하게 움직임)	[복합] 몸통을 뒤로 기대면서 양손을 머리 뒤로 모으기
[오른팔] 오른쪽 팔 앞으로 위로 올리고 내리기	[복합] 몸통을 뒤로 기대면서 양손을 앞으로 모으기
[오른팔] 오른쪽 팔 옆으로 위로 올리고 내리기	[복합] 머리를 흔들면서 오른손 머리에 대기
[오른팔] 오른쪽 팔 뒤로 위로 올리고 내리기	[복합] 머리를 좌우로 흔들면서 양손을 좌우로 흔드는 동작
[오른팔] 오른쪽 팔 앞으로 돌리기	[복합] 머리를 흔들면서 왼손 머리에 대기

표 5는 확률적 추론 문법을 이용한 인식기와 확률적 추론 문법에 근접 정보를 적용한 인식기를 가지고 실험한 결과를 나타낸다. 실험 1은 테스트 단계에서 유사 움직임 노드 판별 분류기를 사용하기 전의 실험이고 실험 2는 유사 움직임 노드 판별 분류기를 두어 실험을 한 내용이다. 확률적 추론 문법을 이용한 인식기는 움직임 노드 판별 분류기를 사용하기 전과 후에 각각 64.6 %와 84.9 %의 인식률을 얻었다. 근접 정보를 확률적 추론

표 5 확률적 추론 문법을 이용한 인식기와 확률적 추론 문법에 근접 정보를 적용한 인식기에서의 실험 결과 비교

실험	실험 내용	확률적 추론 문법을 이용한 인식기	확률적 추론 문법에 근접 정보를 적용한 인식기
		정인식 / 총개수	정인식 / 총개수
실험 1	움직임 노드 판별 분류기 사용	265/410=64.6 %	291/410=71.0 %
실험 2	움직임 노드 판별 분류기 사용	348/410=84.9 %	380/410=92.7 %

문법에 적용한 인식 방법에서는 각각 71.0 %와 92.7 %의 인식 결과를 얻음으로써 근접 정보를 확률 문법에 적용하고 움직임 노드 판별 분류기를 사용했을 경우 가장 인식률이 높음을 관찰할 수 있다.

다음은 인식률이 가장 좋은 결과에 대해서 오인식을 데이터를 분석한 자료이다. 실험 2의 확률적 추론 문법에 근접 정보를 적용한 인식기의 실험 결과가 92.7 %로 가장 인식률이 높았고 이에 대한 분석은 다음 표에 나타난다. 표 6은 오인식이 발생한 동작들을 동작의 종류에 따라 분석한 결과를 나타낸다. 각 동작별 오인식율을 살펴보면 한팔동작이 11.4 %로 가장 많았으며 복합 및

표 6 동작의 종류에 따른 오인식 분석표

동작 종류	오인식 개수/총개수	오인식율
한팔동작	18 / 158	11.4 %
머리동작	6 / 80	7.5 %
양팔동작	4 / 100	4.0 %
복합+몸통동작	2 / 72	2.8 %
총계	30 / 410	

몸통 동작이 2.8 %로 가장 적게 나온 실험 결과를 보인다. 한팔동작에서 가장 많은 오인식들이 발생한 것은 실제 전체 동작의 종류에서 한팔동작이 전체 동작의 1/3 이상을 차지하기 때문에 비례적으로 에러가 많이 나타난 것으로 해석할 수도 있으며, 또한 한편으로는 본 논문에서 제안한 방법이 복합 및 몸통 동작에서 오히려 더 좋은 효과를 얻을 수 있음을 해석할 수 있다. 즉, 복합 및 몸통 동작은 한팔동작일 때보다 움직임 노드의 개수가 많기 때문에 더 많은 노드의 확률값을 최종 확률값에 반영하므로 작은 에러를 감소시킬 수 있는 장점이 있는 것이다. 반면에 움직임 노드의 개수가 적은 경우에는 하나의 노드만이 에러를 포함한다해도 전체 확률값에는 큰 영향을 미칠 수가 있기 때문에 에러의 확률이 더 커질 수 밖에 없는 것으로 고찰된다.

표 7은 인식 근접도를 나타낸 것으로 유사 움직임 노드 판별 분류기에 의해 구해진 각각의 후보들 중에서 몇 번째로 인식되었는 지에 대한 분석 결과를 나타낸다.

표 7 인식 근접도

후보 중에 찾아진 경우	오인식 개수 /총개수
2 번째 (오인식)	19 / 30
3 번째 (오인식)	7 / 30
4 번째 (오인식)	2 / 30
5 번째 이상 (오인식)	2 / 30

표 8은 후보 중에서 2번째로 찾아진 경우의 실례로 머리동작과 한팔동작에서 발생한 오인식의 예를 보여준다. 패턴 후보 중에서 2번째로 찾아지는 데이터가 오인

식들의 50%를 차지하는 것으로 보아 미세한 확률값의 차이로 오인식이 발생했음을 알 수 있다. 미세한 확률값의 차이로 오인식이 발생한 데이터 분석 결과를 크게 두 부류로 나눌 수가 있다. 이 중 한 부류는 동작 자체의 정확도 때문에 오인식이 발생한 것으로 사람이 인식하기에도 애매한 경우이다.

표 8의 테스트 패턴 6이 이에 해당하며, 실제적으로 개선 불가능한 것으로 보인다. 다른 한 부류에 해당하는 데이터는 각 손에 대해서 움직임 크기를 비교한다면 오인식을 개선할 수 있을 것으로 여겨지며 이를 확률 문법 추론 방법의 후처리 단계로 추가하면 될 것이다. 이에 해당하는 오류 데이터는 표 8의 테스트 패턴 29이다.

6. 결론

이 논문에서는 확률적 문법 추론 방식을 사용하여 사람의 상반신 동작 인식과 같은 복합적인 움직임이 이루어지는 동작 인식에 적합한 새로운 인식 모델을 제시하였다. 추론 기능이 없는 확률 문법을 이용한 기존의 동작 인식 연구 방법을 개선하기 위하여 확률 문법에 추론 기능을 추가한 인식 방법과, 비언어적 행동 분석의 특징을 반영할 수 있는 새로운 모델인 근접정보를 활용한 정규 문법 추론 모델을 제안하였고, 실험을 통하여 인식률이 향상됨을 보였다. 비언어적 행동 분석에 주로 이용되어지는 사람 윗몸 동작 60 가지에 대해서 실험하여 92.7%의 인식률을 획득함으로써 이 논문에서 제안한 방법의 타당성을 보였다. 향후 연구과제로는 실생활에 활용할 수 있는 시스템으로의 확장에 관한 연구가 있다.

표 8 후보 중에서 2번째로 찾아진 경우의 실례

동작종류	테스트 패턴	후보
머리동작	6 : 왼쪽으로 꺾기, 그리고 다시 제자리	4 : 왼쪽으로 돌리다, 그리고 다시 제자리 : 9.926396e+305 6 : 왼쪽으로 꺾기, 그리고 다시 제자리 : 7.489708e+305 7 : 오른쪽 방향으로 회전시키기 : 1.249047e+305 2 : 뒤로 올리다, 그리고 다시 제자리 : 9.928185e+304 1 : 숙이다, 그리고 다시 제자리 : 6.681826e+304 8 : 왼쪽 방향으로 회전시키기 : 3.573108e+304 3 : 오른쪽으로 돌리다, 그리고 다시 제자리 : 1.918542e+304 5 : 오른쪽으로 꺾기, 그리고 다시 제자리 : 7.918658e+303
한팔동작	29 : 오른쪽 팔 위로 올리고 내리기	25 : 오른쪽 어깨 앞으로 돌림 (팔도 약하게 움직임) : 1.035340e+234 29 : 오른쪽 팔 위로 올리고 내리기 : 7.030614e+231 30 : 오른쪽 팔 앞으로 돌리기 : 1.224607e+230 28 : 오른쪽 팔 앞으로 위로 올리고 내리기 : 3.255068e+225 27 : 오른쪽 팔 앞으로 위로 올리고 내리기 : 3.850903e+224 26 : 오른쪽 어깨 뒤로 돌림 (팔도 약하게 움직임) : 4.592945e+223 31 : 오른쪽 팔 뒤로 돌리기 : 9.315756e+222 9 : 오른쪽 뒤로 어깨와 함께 돌리기, 그리고 다시 제자리 : 5.482774e+220 58 : 머리를 흔들면서 오른손 하나 머리에 대기 : 1.580669e+217

참고문헌

[1] L. Campbell and A. Bobick, "Recognition of Human Body Motion Using Phase Space Constraints," MIT Media Lab Perceptual Computing Group TR 309, 1995.

[2] E. Clergue, et al. "Automatic Face and Gestural Recognition for Video Indexing," Int'l Workshop on Automatic Face and Gesture Recognition, pp. 110-115, 1995.

[3] 송효섭 외 3 인, "손의 형상과 움직임 방향 정보를 이용한 수화인식", 정보과학회논문지(B) 제 26권 제 6호, pp. 804-810, 1999.

[4] D.A. Becker, "Sensei: A Real-Time Recognition, Feedback and Training System for T'ai Chi Gestures," MIT Media Lab Perceptual Computing Group TR 426, 1997.

[5] A. Braffort, "ARGo: An Architecture for Sign Language Recognition and Interpretation," Proc. of Gesture Workshop, pp. 17-30, 1996.

[6] P.E. Bull, "Posture and Gesture," Pergamon Press, pp. 163-187, 1990.

[7] S. Frey, et al. "Das Berner System zur Untersuchung nonverbaler Interaktion," Methoden der Analyse von Face-to-Face-Situationen, Stuttgart, Metzler, pp. 203-236, 1981.

[8] D.M. Gavrilu and L.S. Davis, "3-D Model-based recognition of human movement by dynamic time warping," Computer Vision Lab., Univ. of Maryland, 1995.

[9] Y.A. Inanov, "Application of stochastic grammars to understanding action," 1998.

[10] A. Sutherland, "Real-time Video-based Recognition of Sign Language Gestures using Guided Template Matching," Gesture Workshop, pp. 31-38, 1996.

[11] 김종성 외 4 인, "한글 수화용 동적 손 제스처의 실시간 인식 시스템의 구현에 관한 연구", 전자공학회논문지, 제 34권 C편 제 2호, pp. 61-70, 1997.

[12] K.S. Fu, "Syntactic Methods in Pattern Recognition," Academic Press, pp. 124-229, 1974.

[13] R.C. Gonzalez and M.G. Thomason, "Syntactic Pattern Recognition," Addison-Wesley Publishing Company, pp. 177-270, 1978.

[14] O. Munkelt, et al. "A model driven 3D image interpretation system applied to person detection in video images," 14'th ICPR 1998, pp. 70-73, 1998.



조 경 은

1989년 3월 ~ 1993년 2월 동국대학교 전자계산학과(공학사). 1993년 3월 ~ 1995년 2월 동국대학교 컴퓨터공학과 대학원(공학석사). 1995년 3월 ~ 현재 동국대학교 컴퓨터공학과 대학원(박사과정). 관심분야는 멀티미디어 정보처리, 컴퓨터비전, 컴퓨터 그래픽스, 형태인식



조 형 제

1969년 3월 ~ 1973년 2월 부산대학교 전자공학과(공학사). 1973년 9월 ~ 1975년 8월 한국과학기술원 전기&전자 공학과 대학원(공학석사). 1982년 9월 ~ 1986년 2월 한국과학기술원 전기&전자 공학과 대학원(공학박사). 1975년 8월 ~ 1982년 8월 금성통신(주)연구소 실장. 1986년 2월 ~ 현재 동국대학교 영상정보통신대학원 멀티미디어학과 교수. 관심 분야는 멀티미디어 정보처리, 컴퓨터비전, 컴퓨터 그래픽스, 형태인식