

# 인스턴스 기본 학습과 상징적 학습 알고리즘을 이용한 핸드제스처의 인식에 관한 연구

최성균, 이정환, 이명호  
연세대학교 공과대학 전기공학과

## A study on the Hand Gesture Recognition using Instance Based Learning and Symbolic Learning Algorithms

S. K. Choi, J. W. Lee, M. H. Lee,  
Department of Electrical Engineering, Yonsei University

### ABSTRACT

This paper is a study on the hand gesture recognition using Instance-based learning, Symbolic learning algorithms and PowerGlove which supplies information on finger position, hand position and orientation. The data were carefully examined, and a few features of the data that would serve as good discriminants between signs when used with the learning algorithms were extracted. The hand gesture data collected from 5 people were applied to the learning algorithms. In spite of the noise and accuracy constraints of the equipment used, some accuracy rates were achieved.

### 1. 서 론

80년대 초에 컴퓨터의 계산 능력이 증대되면서, 더욱 복잡한 주변장치들이 컴퓨터에 부착되기 시작하였다. 또한, "인간-기계 인터페이스"의 중요성이 이해되기 시작하였고, 컴퓨터를 기본으로 한 패턴인식 테크닉을 향상시키기 위해, 다양한 접근이 이루어졌다. 패턴인식은 인간에게는 쉽지만 컴퓨터에게는 어려운 작업이었다. 초기에는 이러한 연구를 위해 필요한 계산용량이 너무 컸기 때문에 많은 실패가 있었다. 그러나, 신경회로망, Bayesian의 통계적인 방법, 인스턴스 기본 학습(Instance-based learning: IBL), 상징적 학습(Symbolic learning) 알고리즘과 같은 많은 접근들이 이루어졌다. 컴퓨터를 이용한 사람이 만드는 제스처를 인식하는 분야는 일반적으로 "제스처 인식"이라 알려져 있다. 이 실험에서 사용된 PowerGlove는 Mattel사가 1989년에 제조를 시작한 글러브를 기반으로 한 핸드 트래킹 장치이다. PowerGlove는 3차원공간에서 글

러브의 움직임이나 손가락의 위치를 추적할 수 있다. 이것은 4개의 손가락에 센서를 가지고 있고, 다른 glove들과 비교하여 성능이 떨어지는 트래킹 메카니즘을 가지고 있으나(4D-x,y,z,roll) 가격이 저렴하다는 이점을 가지고 있다. 또한 인터페이스 박스를 가지고 있어서, 표준 RS-232 시리얼 포트를 통하여 거의 모든 기계들과 쉽게 연결할 수 있다. 본 연구에서는 간단한 테크닉과 저렴한 장치를 이용하여 이산적인 수화를 어느정도 인식할 수 있는지 그 가능성을 조사하였고, 글러브를 기본으로 한 간단한 핸드 제스처 인식에서 더 나아가, 연속적, 실시간으로 더 많은 핸드 제스처를 인식할 수 있을지에 대해 고찰하였다.

### 2. 학습 알고리즘

제스처를 인식할 수 있는 시스템을 만드는데 사용될 수 있는 여러 가지 학습방법론이 있다. 일반적으로, 이러한 학습 알고리즘은 일종의 분류(classification) 작업이다. -보통 특성(attribute)라고 알려진 일련의 입력이 주어졌을 때, 시스템은 그러한 특성들을 기초로하여 대상(object)의 타입이나 클래스를 결정해야 한다. 그 특성들은 일반적으로 직접적으로 물리적인 것일 필요는 없고, 보통 어떤 형태의 전처리(pre-processing)과정이 실제로 학습 알고리즘에 의해 사용되는 특성들을 제공하기 위해 필요하다.

#### 상징적 학습 알고리즘(Symbolic learning algorithms)

상징적 학습 알고리즘은 특성과 클래스와의 관계를 형성하는 일련의 규칙을 결정하는 알고리즘이라 할 수 있다. 이러한 방식에는 여러 가지가 있으며 이 방법들은 규칙을 구성하는 방법에 차이가 있다. 상징적 학습 알고리즘의 한 예가 C4.5이다. C4.5는 "정보이득"이라는 개념을 사용하여 최종결과를 출

력하는 “결정트리(decision tree)”를 만드는 방법이다. 예를 들어 어느 한 지점에서, 구별을 하기 위한 특정한 속성을 지정해 준다면 이것은 시스템이 클래스를 얼마나 잘 구별하는지에 영향을 미칠 것이다. 트리의 어느 한 특정한 노드에서 구별요소(discriminant)로서 사용하기에 어느 특성이 가장 좋은지를 고려함으로써 결정트리를 구성할 수 있고, 따라서 연속적으로 특성을 검사하면서 트리의 처음에서 끝까지 진행해 나갈 수 있다. 트리의 한 예가 아래에 나와 있다.

Thumb Bend  $\leq 1$  : Is Not a "B"  
 Thumb Bend  $> 1$  :  
 | Ring Finger Bend  $\leq 2$  : Is a "B"  
 | Ring Finger Bend  $> 2$  : Is Not a "B"

이 트리를 그림으로 설명한 것은 다음과 같다.

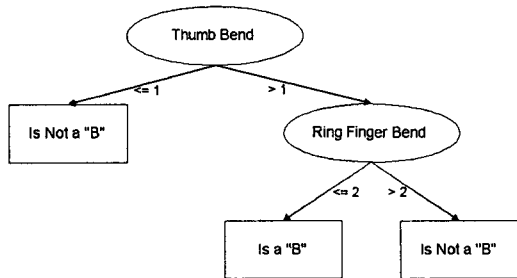


그림 2-1 C4.5에 의해 만들어진 결정 트리  
 Fig. 2-1 A graphical representation of the decision tree produced by C4.5

어떠한 데이터에 관하여 결론을 끌어내기를 원한다고 할 때 그 데이터의 특성들을 고려함으로써 잎노드(leaf-node)에 도달할 수 있고, 이 노드는 출력어 어떠한지 알려준다.

상징적 학습 알고리즘은 공간을 분할한다. 이것은 그림 2-2에 나타나 있다. 일반적으로, 2개 이상의 분할 영역이 있을 수 있고, 더 많은 속성이 있을 수 있다. 여기에서는 8개의 특성이 있지만 관심이 가는 것은 2개다.

C4.5는 한번에 하나의 특성만을 고려할 수 있고, 공간을 단지 “orthogonal”한 부분들로 나눌 수 있을 뿐이다.-공간을 축과 평행하게 나눈다. 예를 들어, 수직으로 혹은 수평으로 평행하지 않고 비스듬한 개념경계(concept boundary)(데이터의 실제 인스턴스가 하나의 클래스에서 다른 클래스로 변하는 경계-경계의 한쪽에는 양의 값이고 다른 한쪽에서는 음의 값을 갖는)를 생각해 보자. 이러한 개념은 그림 2-3에서와 같이 하나의 결정 트리에 의해서만 근사화될 수 있다.

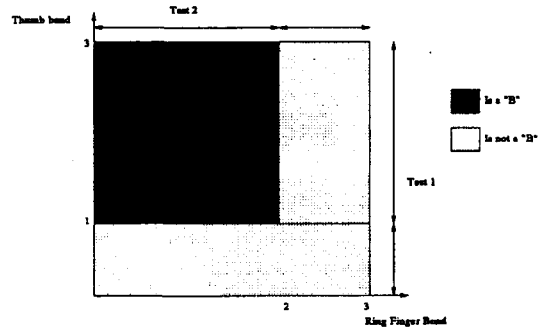


그림 2-2 결정 트리에 의한 공간 분할.  
 Fig. 2-2 The decision tree partitions the space.

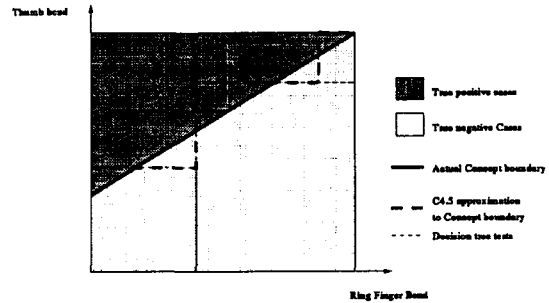


그림 2-3 C4.5의 한계. 개념경계가 orthogonal하지 않을 때의 가정.  
 Fig. 2-3 The limitations of C4.5. If the concept boundary is non-orthogonal C4.5 has to make some approximations.

**인스턴스 기본 학습(Instance-based Learning)**

인스턴스를 기본으로 한 학습 방법은 주로, 각각의 클래스를 위한 전형적인 특성 예를 보유하고 있음으로 해서 동작한다. Aha, Kibler, Albert는 3가지의 특성을 가지고 있는 일련의 인스턴스를 기본으로 한 학습 알고리즘을 정의 하였다.

- 1) 유사도 함수(A similarity function); 어떤 두 개의 인스턴스가 서로 얼마나 가까운지를 알고리즘에 알려준다.
- 2) 전형적인 인스턴스 선택 함수(A typical instance selection function); 인스턴스중에 어느것을 예로서 유지 할것인지를 알고리즘에 알려준다.
- 3) 분류 함수(A classification function); 새로운 사례가 주어졌을 때 그것이 학습된 사례와 어떻게 관계되는지를 결정하는 함수이다.

IBL1: 모든 예의 인스턴스를 저장하고 단순히 가장 가까운 인스턴스를 찾는다.-이 인스턴스의 클래스는 가장 가까운 인스턴스의 클래스이다. 많은수의 인스턴스를 저장하는 경우에는 더욱 큰 공간이 요구된다.

### 3. 특성 추출과 선택

#### 3.1 특성 분석

클러브상태에 대하여 프레임으로서 만들어지는 파 일속에 저장되는 일련의 패키지가 생긴다. 핸드 제 스처를 만들어 내는 매초당 약 25개의 프레임이 있 다. 이렇게 얻은 정보를 직접 어느 학습 알고리즘 속으로 입력시키는것도 가능하겠지만 이러한 방법 은 계산적으로 복잡해지고, 데이터로부터 추출할수 있는 구별하는데 유용한 특성의 잇점을 이용하지 못하게 된다. 더욱이, 기존의 학습 알고리즘을 이러 한 작업을 수행하도록 확장시키는데 아주 많은 노 력이 필요하게 될 것이다. 왜냐하면, 대부분의 학습 알고리즘은 시간에 따라 변하는 데이터를 잘 다룰 수 없기 때문이다.

따라서 본 연구에서는, 특성 혹은 속성이라고 일컬 어지는, 핸드 제스처를 구분하는 데 유용한 정보를 데이터로부터 추출하였다. 이것은 본 연구의 중요 한 부분이며, 간단한 처리과정에 의해 추출될수 있 는 특성에 대해 고찰하고, 평가하였다.

손가락으로부터 나오는 정보는, 각각의 핸드 제스 처 사이에서 일어나는 자동적인 캘리브레이션 때문 에 비교적 깨끗한 반면에, x, y, z와 같은 위치 정 보는 자주 오차가 발생함이 발견되었다. 이러한 오 차는 단지 랜덤하게 잡음이 있을뿐만 아니라 물리 적으로도 설명이 불가능한 에러다. 위치에 대한 잡 음은 있지만 대부분은 1혹은 2센티미터정도였다. 그러나, 어떤 경우에는 반환된 데이터가 1/25초동안 40센티미터 정도를 움직였다고 나타내었는데, 이것 은 핸드 제스처를 사용할때 거의 나타나지 않는 현 상이다. 이러한 현상(glitch)을 제거하기 위하여 다 음과 같은 방법이 채택되었다.

분석되는 각각의 파일은 연속적인 프레임들로 구성 되어 있다. 각 프레임에는 x, y, z의 위치, 손목회 전, 손가락의 구부러짐에 관한 정보가 있다.

$x_i, y_i, z_i$ 를 샘플의 i번째 프레임에 저장되는 x위치, x위치, z위치라고 하자.

마지막 프레임으로부터, 방향의 변화를 다음과 같 이 정의 한다.

$$\Delta x_i = x_i - x_{i-1}$$

$$\Delta y_i = y_i - y_{i-1}$$

$$\Delta z_i = z_i - z_{i-1}$$

벡터(  $\Delta x_i, \Delta y_i, \Delta z_i$ )는 진행되는 방향을 가르키 는 포인터이다. 만일, 이 벡터의 길이를 구하면;

$$\Delta_i^2 = \sqrt{\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2}$$

이것으로부터 속도를 측정할수 있고, 시간에 관한 1차 도함수이다.

마찬가지로 2차 도함수를 정의 할수 있다.

$$\Delta^2 x_i = \Delta x_i - \Delta x_{i-1}$$

$$\Delta^2 y_i = \Delta y_i - \Delta y_{i-1}$$

$$\Delta^2 z_i = \Delta z_i - \Delta z_{i-1}$$

$$\Delta_i^2 = \sqrt{\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2}$$

벡터(  $\Delta^2 x, \Delta^2 y, \Delta^2 z$ ) 또한 방향변화를 나타내는 벡터이다.  $\Delta_i^2$ 은 마찬가지로 그 벡터의 norm의 길 이이다.

#### 3.2 특성 추출

데이터베이스는 5명으로부터 각 수화당 5개의 샘플 을 얻었고, 방법의 타당성을 조사하기 위해, 수집된 데이터에 대하여 테스트되었다.

각각의 특성들은 이미 언급한 학습 알고리즘으로 테스트 되었다.

각경우에 있어서, 5단 교차 확인법(5-fold cross validation)이 사용되었다. 5단 교차 확인법은 샘플 세트가 5개로 분할되어, 분할한 5개 중의 하나는 테스트 세트로 사용되고, 학습자는 나머지 4가지에 대하여 학습된다. 이러한 과정이, 나머지 세트들도 모두 한번씩 테스트세트로 하여 반복하고 평균 오 차율 구한다.

#### 1) 거리, 에너지, 시간 특성

위에서 계산된 데이터의 응용중에서 간단한 한가지 방법은 위의  $\Delta_i$ 를 더하는 것이다. 이것은 핸드 제 스처를 만드는 전체 거리를 계산할수 있도록 해준다. 이것은 하나의 특성이 될 수 있는데, 그것은 어 떤 핸드 제스처는 다른것보다 제스처가 만들어지는 거리가 분명이 길것이기 때문이다.

특성 거리를 다음과 같이 정의 한다.

$$distance = \sum_{i=1}^n \Delta_i$$

여기에서 n은 샘플의 프레임의 수이다.

마찬가지로, 어떤 제스처는 거리는 비슷하더라도 움직임은 다르다.

$$W = \int_0^T F ds$$

위의 식을 이산적으로 근사화 하면;

$$W = \sum_{i=0}^n F \Delta s_i$$

F=ma이고 m은 일정하므로,

$$W \propto \sum_{i=0}^n a \Delta_i$$

$a = \Delta v / \Delta t$ 이고  $\Delta t$ 는 상수라 가정하였다. 또한,

$\Delta v$ 는  $\Delta_i^2$ 와 같다. 따라서,

$$energy = \sum_{i=2}^n \Delta_i^2 \Delta_i$$

## 2) 바운딩 박스 특성

어떤 제스처는 다른 것들보다 움직임이 크고 위치가 다르다. 따라서, 핸드 제스처의 바운딩 박스를 하나의 특성으로 생각할 수 있다. 바운딩박스의 특성을 설명하는데 모두 6개의 특성이 필요하다;  $(x_{min}, y_{min}, z_{min}, x_{max}, y_{max}, z_{max})$ .

하나의 시퀀스  $(x_i, y_i, z_i)$ 가 주어졌을 때, 바운딩 박스는,

$$\begin{aligned} x_{min} &= \min(x_0, \dots, x_n) \\ y_{min} &= \min(y_0, \dots, y_n) \\ z_{min} &= \min(z_0, \dots, z_n) \\ x_{max} &= \max(x_0, \dots, x_n) \\ y_{max} &= \max(y_0, \dots, y_n) \\ z_{max} &= \max(z_0, \dots, z_n) \end{aligned}$$

## 3) 위치 히스토그램 특성

히스토그램이란 이산적인 확률밀도함수라고 생각할 수 있다. 기본적으로, 가능값의 영역을 하위 영역으로 나누고, 각 하위 영역의 인스턴스의 수를 구한다. 핸드 제스처 인식에서는 손의 x, y, z위치에 대한 히스토그램을 구한다.

히스토그램은 다음과 같은 방법으로 계산한다. 영역을 몇 개의 분할로 나눌것인지 그 분할된 영역의 수를  $d$ 라고 하고, 히스토그램의 "기둥(columns)"을  $h_i(0 \leq i < d)$ 라고 하자.  $x$ 위치에 대해서만 생각하면 다음과 같이 된다.

$$h_i = \sum_{j=0}^n \frac{1}{n} r_j(x_j), \quad \forall i, 0 \leq i < a$$

여기에서,

$$r_i(x) = \begin{cases} 1, & \text{if } \frac{i(x_{max} - x_{min})}{d} \leq x < \frac{(i+1)(x_{max} - x_{min})}{d} \\ 0, & \text{otherwise} \end{cases}$$

이것은 다음의 두가지에 의해 표준화 된다.

1) 시퀀스의 길이.  $h_i$ 에 관한 식에서,  $1/n$ 을 포함하므로, 모든  $x_j$ 는 정확히 하나의  $h_i$ 기둥에 속해야 하고,  $\sum_{i=0}^d h_i = 1$ 을 만족해야 한다.

2) 핸드 제스처의 바운딩박스. 기둥분할은 바운딩박스과 관련이 있고, 대부분의  $h_i$ 는 0이 아니다. 이것은 핸드 제스처의 사이즈, 작은 수화의 낮은 해상도와 같은 문제를 제거하는 잇점이 있다.

## 4) 회전, 손가락 위치 히스토그램 특성

손바닥이 위, 아래, 좌, 우 방향으로 향할때에 걸리

는 시간에 비례하여 히스토그램을 만든다.

\* x, y, z위치 정보와 같이 높은 해상도를 가지지 않는다.( x, y, z위치 정보는 각각 256개의 값을 가지는데 반해, 회전에 관한 정보는 12, 손가락은 각각 4개를 갖는다.)

\* 변수의 최대,최소값에 의해 표준화 하는 방법을 적용할수 없다. 예를 들어, 회전값은 0도와 360도 사이의 특정한 값을 나타낸다.(30도 간격) 즉, 상대적인 값이 아닌 절대적인 값을 이용하는 것이다.

손가락은 보통 90퍼센트 정도의 제스처에서 완전히 퍼지거나 구부러진다. 이러한 방법은 정확도가 높은 글러브에서 적용할 때 좋은 성능을 발휘할 수 있으나 실험에 사용된 글러브를 이용해도 정보에 대한 오차는 거의 없다고 할수 있다.

## 4 실험결과 및 결론

1) 글러브의 상태를 나타내는 두 프레임에서 제스처의 방향변화를 구하고, 이 방향변화의 합으로부터 특성거리와 에너지를 이산적으로 근사화하여 핸드 제스처의 한 특성으로 추출하였다.

2) 3차원 공간에서 제스처는 움직임의 크기에 있어서 차이가 나고 위치도 또한 다르기 때문에, 시작점과 끝점 사이에서 각각의 공간좌표들의 원점으로부터 최대 최소의 거리를 구하여 그 최대값과 최소값만을 핸드 제스처를 구별하는 특성으로 사용하였는데 이 실험에서 25에서 35퍼센트의 정확도를 구할수 있었다.

3) x, y축을 중심으로 4개의 영역을 나누어 각각의 영역에 해당하는 각도로 손바닥이 위치하는데 걸리는 시간에 비례하여 히스토그램을 만들어 특성으로 이용하였을 때 40퍼센트의 핸드 제스처 인식률을 보임으로서 x, y, z위치 히스토그램 보다 더 좋은 특성임을 알수 있었다.

6) 학습 알고리즘은 보통 더 많은 예(example)가 주어지면 클래스를 더 잘 구분하는 성질이 있으므로, 한 개의 핸드 제스처당 더 많은 수의 샘플을 얻게 되면 인식율이 높아질 것이라 사료된다.

7) 여러개의 추출된 특성을 결합하여 핸드 제스처 인식에 사용하면, 인식율이 증가할 것이다.

## 참고문헌

- [1] Aha, D. W., Kibler, E. & Albert, M. K. "Instance-based learning algorithms", Draft submission to Machine Learning, 1990.
- [2] Ohira, E., Sagawa, H. & Sakiyama, T. "A Segmentation Method for Sign Language Recognition" IEICE Transactions on Information and Systems, vol. E78-D(1), pp. 49-57. 1995.