

다변량 퍼지 의사결정트리와 사용자 적응을 이용한 손동작 인식

Hand Gesture Recognition using Multivariate Fuzzy Decision Tree and User Adaptation

전 문 진¹, 도 준 형², 이 상 완³, 박 광 현⁴, 변 증 남⁵

Moonjin Jeon¹, Jun-Hyeong Do², Sang-Wan Lee³,
Kwang-Hyun Park⁴, Zeungnam Bien⁵

Abstract While increasing demand of the service for the disabled and the elderly people, assistive technologies have been developed rapidly. The natural signal of human such as voice or gesture has been applied to the system for assisting the disabled and the elderly people. As an example of such kind of human robot interface, the Soft Remote Control System has been developed by HWRS-ERC in KAIST^[1]. This system is a vision-based hand gesture recognition system for controlling home appliances such as television, lamp and curtain. One of the most important technologies of the system is the hand gesture recognition algorithm. The frequently occurred problems which lower the recognition rate of hand gesture are inter-person variation and intra-person variation. Intra-person variation can be handled by inducing fuzzy concept. In this paper, we propose multivariate fuzzy decision tree(MFDT) learning and classification algorithm for hand motion recognition. To recognize hand gesture of a new user, the most proper recognition model among several well trained models is selected using model selection algorithm and incrementally adapted to the user's hand gesture. For the general performance of MFDT as a classifier, we show classification rate using the benchmark data of the UCI repository. For the performance of hand gesture recognition, we tested using hand gesture data which is collected from 10 people for 15 days. The experimental results show that the classification and user adaptation performance of proposed algorithm is better than general fuzzy decision tree.

Keywords : Hand gesture recognition, Multivariate Fuzzy Decision Tree, Model Selection, User Adaptation

1. 서 론

최근 인간-컴퓨터 상호작용(HCI) 기술이 발달하면서 음성이나 동작 등 사람의 자연스러운 신호를 이용해 사람의 의도를 파악하고 시스템을 제어하는 신호로 사용하려는 연구가 다양하게 진행되어 왔다. 이러한 연구의

대표적인 예로 한국과학기술원 인간친화 복지로봇 연구 센터에서는 하지가 불편한 장애인이나 노약자가 간단한 손동작만을 이용해 집안의 가전기기를 제어할 수 있게 하는 소프트리모컨 시스템을 개발했다^[1]. 소프트리모컨 시스템의 손동작 인식 알고리즘은 전체 시스템의 성능을 결정하는 중요한 부분이기 때문에 손동작 인식률을 향상시키기 위한 연구가 수행되어 왔다^[2].

다수의 사용자를 대상으로 한 손동작 인식 실험을 통해 다음과 같은 두 가지 요인에 의해 손동작 인식률이 저하된다는 것을 확인할 수 있었다.

1.1 Inter-person variation

그림 1은 3명의 사용자로부터 수집한 ‘위’, ‘왼쪽’, ‘시계방향 원’ 동작의 손동작 궤적을 나타낸다. 사용자마다

※ 본 연구는 과학기술부/한국과학재단 우수연구센터육성사업의 지원으로 수행되었음(R11-1999-008)

¹ 한국항공우주연구원 다목적 3호체계팀 연구원
(E-mail : mjjeon@kari.re.kr)

² 한국과학기술원 인간친화 복지로봇시스템 연구센터 선임급 위촉 연구원 (E-mail : jhdo@ctrsys.kaist.ac.kr)

³ 한국과학기술원 전자전산학과 박사과정
(E-mail : bigbean@ctrsys.kaist.ac.kr)

⁴ 광운대학교 정보제어공학과 조교수 (E-mail : akaii@robotian.net)

⁵ 한국과학기술원 전자전산학과 교수 (E-mail : zbien@ee.kaist.ac.kr)

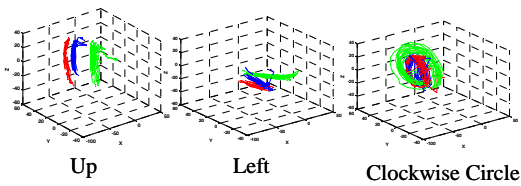


그림 1. 3명의 사용자의 손동작 궤적

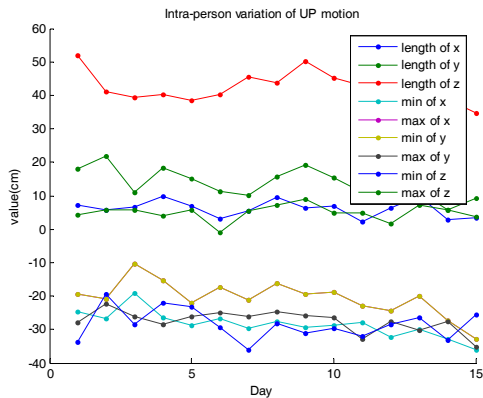


그림 2. 15일 간의 손동작 특성 변화

직선 동작의 길이, 각도, 원 동작의 크기가 다양하다는 것을 알 수 있다. 이러한 손동작의 특성을 **Inter-person variation**이라 하며 이로 인해 인식 모델의 학습 과정에 손동작 데이터를 제공한 사용자의 손동작 인식률은 높은 반면 시스템을 처음 사용하는 사용자의 손동작 인식률은 낮아지게 된다.

1.2 Intra-person variation

손동작 인식률을 저하시키는 두 번째 요인은 같은 사용자라 할지라도 때때로 손동작 특성이 변화한다는 점이며 이를 **Intra-person variation**이라 한다. 그림 2에는 한 사용자로부터 15일간 수집한 손동작의 특성 변화를 나타낸다. 인식 모델을 학습할 당시의 손동작 특성과 시스템을 사용하는 시기의 손동작 특성이 다를 수 있기 때문에 손동작 인식률을 저하시킬 수 있다.

새로운 사용자의 손동작을 인식하기 위해 **UI 모델**(사용자 독립 모델)을 이용한 접근 방식은 다양한 사람으로부터 데이터를 수집할 수 있기 때문에 데이터 수집이 용이하다는 장점이 있는 반면 대체로 낮은 인식률을 갖는다는 단점이 있다. 각각의 사용자의 손동작을 인식하기 위해 해당 사용자의 손동작을 이용해 **UD 모델**(사용자 종속 모델)을 학습하고 인식에 사용하는 방식은 인식률이 높다는 장점이 있는 반면 새로운 사용자마다 데이터를 수집해야 한다는 단점이 있다. 이러한 문제점을

해결하기 위한 최근 연구는 사용자 적응, 개인화된 인식 등의 방법으로 진행되었다. **UI 모델**을 새로운 사용자의 손동작 특성에 적응시키는 사용자 적응 방법이 제시되었고 한국 수화 인식 등에 사용되어 인식률을 향상시킨 결과를 보인 바 있다^[3]. 표정 인식을 위한 한 연구에서는 다양한 인식 모델을 만들고 새로운 사용자에 대해 모델 선택 기법을 이용해 적합한 인식 모델을 선택하는 방법이 제시되었다^[4].

같은 사용자라 할지라도 때때로 변화하는 손동작을 강인하게 인식하기 위해 퍼지 집합의 개념을 의사결정 트리에 적용한 퍼지 의사결정 트리를 이용해 인식률을 향상시킨 바 있다^[5]. 손동작 데이터를 이용해 퍼지 의사결정 트리를 학습할 때 트리의 각 노드를 퍼지 소속함수를 이용해 분기하기 때문에 노드의 수가 증가하게 된다. 의사결정 트리의 노드 수가 증가하면 트리의 구조가 방대해지고 일반화 성능(**generalization performance**)를 저하시킨다^[6]. 즉 학습에 사용한 데이터의 인식률은 노드 수가 증가할수록 높아질 수 있지만 학습에 사용하지 않은 데이터의 인식률은 노드 수의 증가에 따라 낮아지게 된다. 이러한 노드 수 증가에 따른 인식률의 저하는 사용자 적응과정의 효율을 감소시키므로 인식률의 저하 없이 노드 수를 감소시키는 과정이 필요하다. 이 논문에서는 다변량의 개념을 퍼지 의사결정 트리에 적용해 노드 수를 감소시킨 다변량 퍼지 의사결정 트리(**MFDT**)를 제안한다. 학습 과정에서 다수의 **MFDT** 모델을 생성하고 새로운 사용자의 손동작을 인식할 때 최대 우도 모델 비교(**Maximum likelihood model comparison**)를 이용해 가장 적합한 모델을 선택하며 **Gradient descent**에 기반한 사용자 적응을 이용해 사용자가 원할 때 사용자의 손동작에 모델을 적응시킨다.

이 논문의 구성은 다음과 같다. 2장에서는 손동작 인식 시스템에 대해서 설명하고 3장에서는 **MFDT**의 학습 및 분류과정을 설명한다. 4장에서는 모델 선택 및 사용자 적응 과정에 대해 설명하며 5장에서는 벤치마크 데이터를 이용한 분류능력, 손동작 데이터를 이용한 분류능력 및 적응 능력을 평가한다. 6장에서는 이 논문의 결론을 내리며 추후 가능한 연구과제에 관해 고찰함으로 글을 맺는다.

2. 손동작 인식 시스템

이 논문의 대상이 되는 소프트웨어모션 시스템은 하지가 불편한 장애인과 노약자를 위한 시스템이다. 장애인을 대상으로 한 설문조사^[7] 결과에 근거해 장애인들이 동작하기 편한 10가지 손동작을 선정했고 해당 손동작

은 그림 3과 같다.

소프트리모컨 시스템의 손동작 인식과정을 그림 4에 나타내었다. 두 대의 카메라를 이용해 얻은 영상에서 얼굴과 손의 위치를 추출한 후 얼굴을 중심으로 한 손의 상대적인 좌표를 이용해 손의 3차원 궤적을 계산한다. 손의 움직임이 빠를 때 사용자가 손동작을 통해 명령을 내리고 있다는 가정 하에 손동작의 시작점과 끝점을 찾는다. 구해진 궤적을 이용해 특징을 추출한 후 10가지 동작 중에 하나로 분류한다. 의도하지 않은 연속 동작, 즉 up 명령을 하고 나서 손을 내리는 동작 등을 명령으로 인식되지 않도록 하기 위해 state automata를 이용해 의미 없는 동작을 차단한 후 인식된 손동작 정보를 이용해 가전기기 제어 신호를 발생시킨다. 새로운 사용자가 시스템을 사용하기 시작할 때 적합한 인식 모델을 선택하는 과정을 수행하며, 시스템을 사용하는 중에 사용자가 원할 때마다 사용자 적응 과정을 수행한다.

3. 다변량 퍼지 의사결정트리

의사결정트리는 귀납추리를 위해 가장 널리 사용되어 온 방법 중에 하나로써 이산 값을 가지는 타깃 함수를 트리 형태의 학습된 모델로 근사화하기 위한 방법이다⁶⁾. 의사결정트리를 학습하는 기본적인 방법은 각 노드를 분기할 때 정보 이득을 최대화 하는 속성을 찾는 것

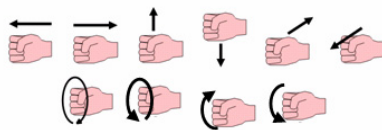


그림 3. 10가지 손동작

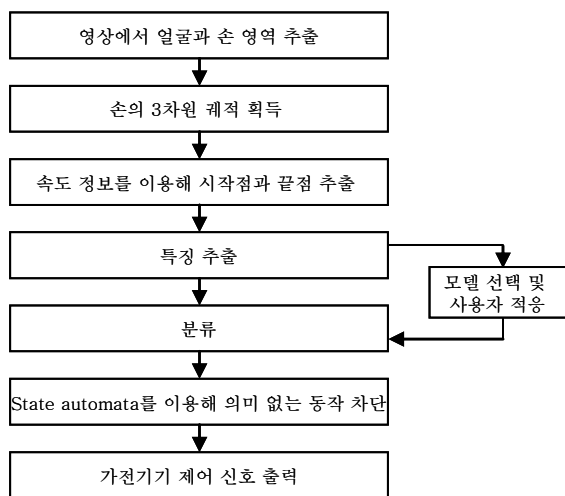


그림 4. 손동작 인식 과정

이며 이러한 방법을 적용한 대표적인 알고리즘으로 ID3, C4.5가 있다.

의사결정트리에 퍼지 집합의 개념이 적용된 퍼지 의사결정트리(FDT)⁸⁹⁾는 정보이득을 계산할 때 퍼지 소속 값을 사용하며 FDT의 각 노드는 하나의 속성 및 퍼지 소속함수 정보를 가지고 있다.

이 논문에서 제시하는 MFDT는 FDT에 다변량의 개념¹⁰⁰⁾이 적용된 것으로 다변량의 개념은 그림 5에 나타내었다. 두 가지의 클래스와 두 가지 특징으로 표현된 데이터 집합을 하나의 속성을 사용해 분기할 경우 6번의 분기가 필요하지만 속성 벡터(다변량 속성)를 사용해 분기할 경우 한 번의 분기로 충분하다. 이 논문에서는 이러한 다변량 개념을 퍼지 의사결정트리에 적용하였고 이를 다변량 퍼지 의사결정트리라고 한다.

3.1 다변량 퍼지 의사결정트리 학습

MFDT 학습 과정에서 노드를 분기할 때 사용하는 속성 벡터는 Linear discriminant analysis(LDA)를 이용해 구할 수 있다. LDA는 분류 문제에서 데이터의 차원을 줄이는데 사용되는 방법으로 LDA를 이용해 구한 벡터 \mathbf{w} 는 LDA의 결과인 discriminant이며 이 벡터로 데이터를 사영했을 때 클래스 분리도가 가능한 최대가 되는 벡터이다¹¹⁾. Iris 데이터에 LDA를 적용하면 그림 6과 같은 벡터 \mathbf{w} 를 구할 수 있다.

MFDT 학습은 다음과 같은 과정을 따른다.

Step 1. 근 노드 생성

근 노드를 생성하고 모든 학습 데이터 \mathbf{x} 를 근 노드에 위치시킨다.

Step 2. 정보이득을 최대화 하는 노드 생성

1) LDA를 이용해 속성 벡터 \mathbf{w} 결정

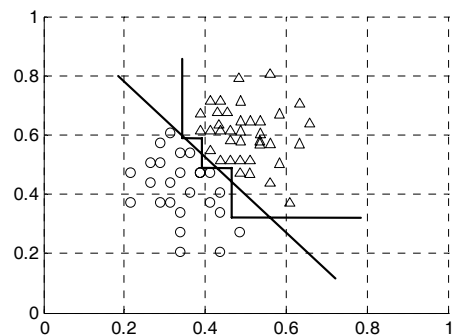


그림 5. 일변량 분기와 다변량 분기

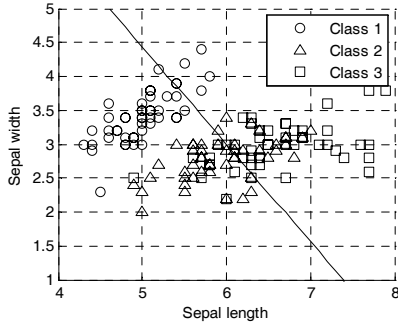


그림 6. LDA를 이용해 계산된 속성 벡터

$$\text{Maximize } J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_B \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}} \quad (1)$$

$$\mathbf{S}_w = \sum_{i=1}^K \mathbf{S}_i \quad (2)$$

$$\mathbf{S}_i = \sum_{\mathbf{x} \in \text{class } i} (\mathbf{x} - \mathbf{m}_i)(\mathbf{x} - \mathbf{m}_i)^T \quad (3)$$

$$\mathbf{m} = \frac{1}{K} \sum_{i=1}^K \mathbf{m}_i \quad (4)$$

∴ \mathbf{w} : $\mathbf{S}_w^{-1} \mathbf{S}_B$ 의 가장 큰 eigenvector

2) \mathbf{w} 를 이용해 속성 값 계산

$$z = \mathbf{w}^T \mathbf{x} \quad (5)$$

3) 현재 노드의 엔트로피 계산

$$\text{Entropy}(S) = - \sum_i P_i^S \log_2 P_i^S \quad (6)$$

$$P_i^S = \frac{N_s^i}{N_S} \quad (7)$$

$$N_S = \sum_i N_s^i \quad (8)$$

N_s^i : i 번째 클래스인 데이터의 개수

S : 현재 노드에 도달한 z 의 집합

4) 현재 노드를 분기할 때 정보이득을 최대화 하는 소속함수 및 정보이득을 계산

생성할 소속함수의 개수를 N 개라고 하면 나열된 z 의 범위에서 클래스가 바뀌는 모든 지점 중에 임의의 $N-1$ 개 지점의 모든 조합을 구하고 각 조합에 대해 소속

함수를 만든다. 선택된 조합으로 나누어지는 N 개의 구역에서 각 구역의 z 값의 평균값이 각각의 삼각형 소속함수의 꼭지점 위치가 된다.

$$m^{v,c} = \frac{1}{n_v} \sum_{j=1}^{n_v} z_j^v, \quad v=1, \dots, N \quad (9)$$

n_v : v 번째 구역에 포함되는 데이터 z 의 개수

z_j^v : v 번째 구역에 포함되는 z 를 오름차순으로 정렬했을 때 j 번째 z 값

$m^{v,c}$: v 번째 소속함수의 꼭지점 위치

만약 현재 노드의 모든 데이터가 N 보다 작은 개수의 클러스터로 완전히 분리될 경우 해당 클러스터의 개수와 같은 개수의 소속함수를 만든다.

모든 구간에서 각 소속함수의 꼭지점 위치를 구한 후 소속함수의 좌측, 우측 값을 구한다.

$$m^{v,l} = m^{v,c} - 0.5(1 + \gamma)(m^{v,c} - m^{v-1,c}) \quad (10a)$$

$$m^{v,r} = m^{v,c} + 0.5(1 + \gamma)(m^{v+1,c} - m^{v,c}) \quad (10b)$$

γ 는 두 퍼지 소속함수 간의 겹친 정도를 나타낸다.

$S_{v|w} = \{(z, \mu_{S_{v|w}}(z)) \mid \mu_{S_{v|w}}(z) : v \text{ 번째 소속함수의 소속 값}\}$

$$C_{S_{v|w}}^i = \sum_{\substack{\text{class of } z=i \\ z \in \text{Supp}(S_{v|w})}} \mu_{S_{v|w}}(z) \quad (11)$$

$$C_{S_{v|w}} = \sum_i C_{S_{v|w}}^i \quad (12)$$

$$P_i^{S_{v|w}} = \frac{C_{S_{v|w}}^i}{C_{S_{v|w}}} \quad (13)$$

$$\text{Entropy}(S_{v|w}) = - \sum_i P_i^{S_{v|w}} \log_2 P_i^{S_{v|w}} \quad (14)$$

$$\text{Gain}(S, \mathbf{w}) = \text{Entropy}(S) - \sum_v \frac{N_{S_{v|w}}}{N_S} \text{Entropy}(S_{v|w}) \quad (15)$$

5) 현재 노드를 한 가지 속성만 사용해 분기하는 경우(univariate case) 가장 큰 정보이득을 갖게 하는 속성 및 소속함수를 구한다.

이 때의 \mathbf{w} 는 하나의 요소만 1이고 나머지는 0인 단위 벡터가 된다.

6) 다변량 속성벡터를 사용한 분기(4번 과정)와 하나의 속성을 사용한 분기(5번 과정)의 정보이득 중 더 큰 정보이득을 갖는 \mathbf{w} 와 소속함수를 사용해 자식 노드를 생성한다.

Step 3. 종료 조건을 만족하면 현재의 노드를 단말 노드로 만들고 클래스를 할당한다. 그렇지 않으면 모든 자식 노드에서 Step 2를 재귀적으로 반복한다.

- 종료 조건

- 1) 현재 노드의 모든 데이터의 클래스가 동일한 경우
- 2) 현재 노드의 깊이가 미리 정의된 최대 깊이를 초과하는 경우

그림 6의 데이터를 이용해 Step 2-4)에서 기술한 방법대로 퍼지 소속함수를 생성한 결과는 그림 7과 같다. 전체 과정을 통해 생성한 MFDT는 그림 8과 같다.

3.2 다변량 퍼지 의사결정트리를 이용한 분류

시험 데이터가 주어지면 학습된 다변량 퍼지 의사결정트리를 이용해 분류할 수 있다. 입력된 데이터를 \mathbf{x} 라 하면 다음과 같은 과정을 통해 분류된다.

Step 1. 근 노드에서 각 단말 노드까지 거치는 노드에서

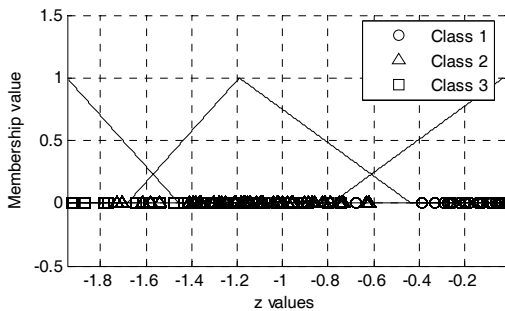


그림 7. 자동으로 생성된 퍼지 소속함수

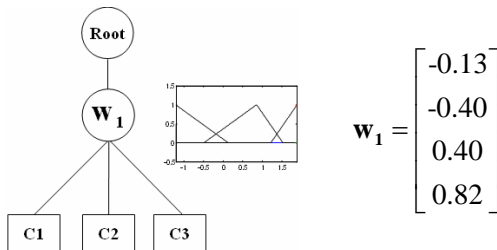


그림 8. 학습된 MFDT

의 소속 함수와 속성 벡터를 이용해 T-norm을 계산한다.

- 1) 각 노드의 속성 벡터 \mathbf{w}_i 의 방향으로 입력 데이터를 사영시킨다.

$$z_i = \mathbf{w}_i^T \mathbf{x} \quad (16)$$

- 2) 근 노드에서 n 번째 단말 노드까지의 T-norm을 계산한다.

$$T_n = \prod_{i=root\ node}^{n_{th}\ leaf\ node} \mu_{s_{v/w}}^i(z_i) \quad (17)$$

위 과정을 모든 단말 노드에 대해 반복하면 각 단말 노드는 0에서 1사이의 T-norm 값을 가지게 된다.

Step 2. 단말 노드 중에 각 클래스에 해당하는 단말 노드들의 T-norm의 평균을 계산하고 이 평균 값이 가장 큰 클래스로 입력 데이터를 분류한다.

A_i : i 번째 클래스를 갖는 단말 노드들의 T-norm의 평균

$$class\ C = \arg\ max_i\ A_i \quad (18)$$

4. 모델 선택 및 사용자 적응

4.1 모델 선택

학습데이터가 주어지면 3.1절의 방법을 이용해 MFDT 모델을 생성할 수 있다. 다양한 사람으로부터 수집한 데이터를 이용해 생성한 다수의 모델은 그림 9와 같은 모델 풀에 저장되어 새로운 사용자의 손동작을 인식하는데 사용된다.

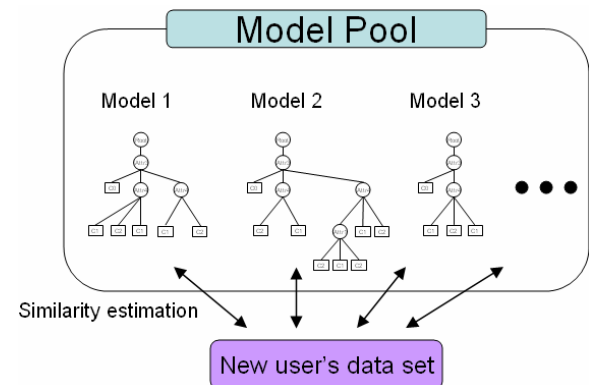


그림 9. 모델 선택

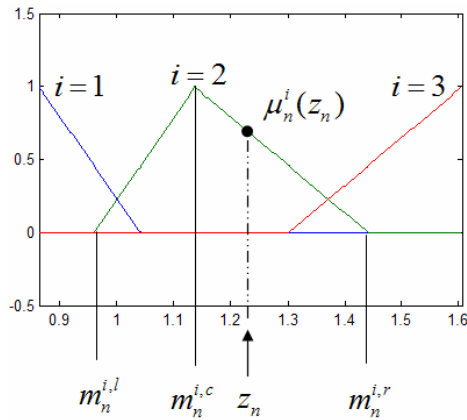


그림 10. MFDT의 각 노드를 구성하는 퍼지 소속함수의 파라미터

새로운 사용자가 시스템을 사용하려 할 때 모델 선택 과정을 통해 사용자에게 가장 적합한 모델을 선택한다. 사용자의 손동작 데이터와 각 모델의 유사도는 최대 우도 모델 비교(Maximum Likelihood Model Comparison)에 근거해 계산한다.

$$P(m_i | D) = \frac{P(D | m_i) p(m_i)}{p(D)} \propto P(D | m_i) p(m_i) \quad (19)$$

$$\text{Selected Model } \hat{m} = \arg \max_{m_i} P(D | m_i) \quad (20)$$

D : data, m_i : i_{th} model

4.2 사용자 적응

모델 선택 과정을 통해 선택된 모델은 사용자의 손동작을 인식하는 분류기로 사용되며 사용자가 원할 때 사용자의 손동작 특성에 적응된다. 클래스 정보를 포함한 손동작 데이터가 하나씩 입력될 때마다 Gradient descent 기반의 적응 과정이 수행되며 이러한 적응과정을 Incremental Adaptation이라 한다^[2].

MFDT 모델은 각 노드에 속성 벡터와 그림 10과 같은 퍼지 소속함수 정보를 가지고 있다. 입력 데이터가 올바르게 분류되기 위해서 분류되어야 할 클래스를 가진 단말 노드들의 T-norm의 평균 값이 다른 클래스의 T-norm의 평균 값보다 커야 한다. 입력 데이터와 같은 클래스를 가진 단말 노드의 T-norm이 1에 가까울수록 에러가 작아지도록 에러 함수를 정의하고 에러를 최소화하는 방향으로 적응시킨다. 사용자 적응의 과정은 입력 데이터와 같은 클래스를 가지는 모든 단말 노드에 대해 다음과 같은 과정을 따른다.

- 1) 근 노드부터 해당 단말 노드까지 이어진 경로 상에 위치한 노드의 속성 벡터를 이용해 입력 데이터를 사영시킨다.

$$z_n = \mathbf{w}_n^T \mathbf{X} \quad (21)$$

\mathbf{w}_n : 근 노드에서 n번째에 위치한 노드의 속성 벡터 이 때 T-norm은 다음과 같이 계산된다.

$$T = \prod_{n=1}^N \mu_n^i(z_n) \quad (22)$$

N : 해당 단말 노드의 부모 노드의 깊이

- 2) Gradient descent를 이용해 각 노드의 소속 함수를 적응시킨다.

$$m_n^{i,l} = m_n^{i,c} - \gamma(m_n^{i,c} - m_n^{i-1,c}) \quad (23)$$

$$m_n^{i,r} = m_n^{i,c} + \gamma(m_n^{i+1,c} - m_n^{i,c}) \quad (24)$$

n : 노드 번호, i : 소속 함수 번호

$$\mu_n^i(z_n) = \begin{cases} \frac{z_n - (1-\gamma)m_n^{i,c} - \gamma m_n^{i-1,c}}{\gamma(m_n^{i,c} - m_n^{i-1,c})}, & z_n < m_n^{i,c} \\ \frac{z_n - (1-\gamma)m_n^{i,c} - \gamma m_n^{i+1,c}}{\gamma(m_n^{i,c} - m_n^{i+1,c})}, & z_n > m_n^{i,c} \end{cases} \quad (25)$$

$$E = \frac{1}{2}(T-1)^2 = \frac{1}{2} \left(\prod_{n=1}^N \mu_n^i(z_n) - 1 \right)^2 \quad (26)$$

$$\frac{\partial E}{\partial m_n^{i,c}} = \left(\prod_{n=1}^N \mu_n^i(z_n) - 1 \right) \cdot \frac{\partial T}{\partial m_n^{i,c}} \quad (27)$$

$$\frac{\partial T}{\partial m_n^{i,c}} = \left(\prod_{\substack{k=1 \\ k \neq n}}^N \mu_k^i(z_k) \right) \cdot \frac{\partial \mu_n^i(z_n)}{\partial m_n^{i,c}} \quad (28)$$

$$\frac{\partial E}{\partial m_n^{i,c}} = \begin{cases} \left(\prod_{n=1}^N \mu_n^i(z_n) - 1 \right) \cdot \left(\prod_{\substack{k=1 \\ k \neq n}}^N \mu_k^i(z_k) \right) \cdot \frac{m_n^{i-1,c} - z_n}{\gamma(m_n^{i,c} - m_n^{i-1,c})^2}, & z_n < m_n^{i,c} \\ \left(\prod_{n=1}^N \mu_n^i(z_n) - 1 \right) \cdot \left(\prod_{\substack{k=1 \\ k \neq n}}^N \mu_k^i(z_k) \right) \cdot \frac{m_n^{i+1,c} - z_n}{\gamma(m_n^{i,c} - m_n^{i+1,c})^2}, & z_n > m_n^{i,c} \end{cases} \quad (29)$$

$$m_n^{i,c} \leftarrow m_n^{i,c} - \eta \frac{\partial E}{\partial m_n^{i,c}} \quad (30a)$$

$$m_n^{i,d} \leftarrow m_n^{i,c} - \gamma (m_n^{i,c} - m_n^{i-1,c}) \quad (30b)$$

$$m_n^{i,r} \leftarrow m_n^{i,c} + \gamma (m_n^{i+1,c} - m_n^{i,c}) \quad (30c)$$

$$m_n^{i+1,d} \leftarrow m_n^{i+1,c} - \gamma (m_n^{i+1,c} - m_n^{i,c}) \quad (30d)$$

$$m_n^{i-1,r} \leftarrow m_n^{i-1,c} + \gamma (m_n^{i,c} - m_n^{i-1,c}) \quad (30e)$$

3) T-norm 값의 변화가 미리 정의된 값보다 작아질 때까지 2)를 반복해 수행한다.

5. 실험 결과

5.1 벤치마크 데이터를 이용한 분류능력 평가

MFDT 학습 및 분류 알고리즘의 일반적 분류 능력을 평가하기 위해 UCI Machine Learning Repository^[13]의 Iris, Wine 데이터를 이용해 분류 능력을 테스트했다. Iris, Wine 데이터의 사양은 표 1과 같다. 학습 및 분류 테스트는 5x2 fold cross validation을 이용했으며 그 결과는 표 2와 같다. 실험 결과 MFDT의 분류 능력이 퍼지 집합의 개념이 포함되지 않은 의사결정트리인 C4.5, C5.0과 일반적인 퍼지 의사결정트리에 비해 우수한 것으로 나타났다.

표 1. 벤치마크 데이터

Data set	Classes	Instances	Features
Iris	3	150	4
Wine	3	178	13

표 2. 의사결정트리 종류별 인식률 비교

Set	C4.5	C5.0	FDT	MFDT
Iris	92.9	92.9	92.8	96.4
Wine	86.6	89.2	88.9	90.6

5.2 손동작 데이터

손동작 데이터는 그림 11과 같은 지능형 주거공간의 천정에 부착된 두 대의 카메라를 이용해 수집했다. 10 frames/sec로 고정된 속도로 이미지를 저장했으며 손동작의 시작점과 끝점은 수동으로 지정했다. 총 10명의 사용자로부터 10가지 손동작을 15일간 수집했으며 수집한 손동작 및 사용한 특징의 종류는 표 3과 같다.

5.3 손동작 분류능력 평가

MFDT의 손동작 분류능력을 평가하기 위해 UD 모델

과 UI 모델을 이용한 인식률을 측정된 결과는 표 4, 표 5와 같다. UD 모델의 경우 MFDT를 이용해 FDT에 비해 높은 평균 인식률을 얻었고, UI 모델의 경우 FDT보다 낮은 평균 인식률을 얻었다. 두 경우 모두 FDT에 비해 적은 수의 노드를 가지는 트리를 형성했다.



그림 11. 지능형 주거공간

표 3. 손동작 클래스 및 특징

10가지 손동작	16가지 특징
1. 위	• x, y, z 축으로의 길이
2. 아래	• x, y, z 축에서의 최소값
3. 왼쪽	• x, y, z 축에서의 최대값
4. 오른쪽	• x, y, z 축에서 최소값일
5. 앞	때의 시간
6. 뒤	• x, y, z 축에서 최대값일
7. 시계방향 원	때의 시간
8. 반 시계방향 원	• 이심률
9. 시계방향 반원	
10. 반 시계방향 반원	

표 4. UD 모델을 이용한 손동작 인식률

User	FDT		MFDT	
	Number of nodes	CR (%)	Number of nodes	CR (%)
User 1	28	90.8	21	88.3
User 2	24	77.7	18	90.8
User 3	16	86.9	12	98.3
User 4	17	66.9	11	90.8
User 5	18	76.5	17	93.3
User 6	23	89.2	15	86.7
User 7	16	74.6	18	86.7
User 8	22	85.0	17	85.0
User 9	21	92.3	17	89.2
User 10	20	90.8	12	96.7
Average	20.5	83.1	15.8	90.6

표 5. UI 모델을 이용한 손동작 인식률

User	FDT		MFDT	
	Number of nodes	CR (%)	Number of nodes	CR (%)
User 1	52	55.8	34	42.0
User 2	52	69.6	34	61.0
User 3	52	77.3	34	69.0
User 4	52	69.2	34	73.0
User 5	52	60.4	34	52.5
User 6	52	71.2	34	72.5
User 7	52	63.1	34	63.5
User 8	52	58.5	34	54.0
User 9	52	73.8	34	52.5
User 10	52	66.2	34	66.0
Average	52	66.5	34	60.6

표 6. 모델 선택 후의 손동작 인식률

User	Model selection		
	Selected model	Number of node	CR (%)
User 1	5	12	71.7
User 2	9	13	80.8
User 3	4	10	92.5
User 4	9	13	81.7
User 5	7	13	67.5
User 6	9	13	81.7
User 7	2	11	84.2
User 8	2	11	75.8
User 9	3	12	93.3
User 10	2	11	85.0
Average		11.9	81.4

5.4 모델 선택 평가

10명의 사용자 중 한 명을 새로운 사용자로 가정하고 새로운 사용자의 손동작을 인식하기 위해 모델 선택 능력을 평가했다. 생성된 10개의 모델 중에 자신의 모델을 제외한 9개의 UD모델과 하나의 UI 모델을 모델 선택에 사용했으며 모델 선택을 위한 입력 데이터는 1세트를 사용했다. 모델 선택 후의 손동작 인식률은 표 6과 같다. UI 모델을 사용하는 경우보다 인식률이 20% 이상 향상되었다는 것을 확인할 수 있다.

5.5 사용자 적응 평가

인식 모델을 새로운 사용자의 손동작 특성에 적응시키는 과정을 통해 좀 더 높은 인식률을 얻을 수 있다. 그림 12는 UI 모델과 선택된 모델을 사용자 특성에 적응시키는 과정에서 측정된 10명의 사용자의 평균 인식률을 나타낸다.

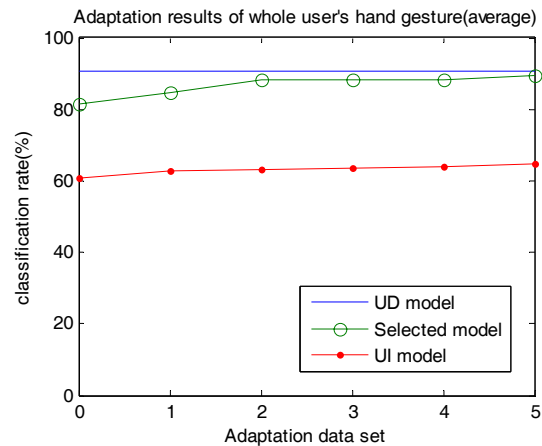


그림 12. 사용자 적응

표 7. UI 모델과 선택된 모델의 사용자 적응 성능 비교(%)

User	UI model	UI model + Adaptation	Selected model	Selected model + Adaptation
User 1	42.0	55.0	71.7	83.3
User 2	61.0	63.0	80.8	90.8
User 3	69.0	72.5	92.5	95.0
User 4	73.0	68.5	81.7	88.3
User 5	52.5	66.5	67.5	86.7
User 6	72.5	72.5	81.7	90.8
User 7	63.5	59.0	84.2	85.8
User 8	54.0	66.5	75.8	84.2
User 9	52.5	54.0	93.3	94.2
User 10	66.0	71.0	85.0	94.2
Average	60.6	64.9	81.4	89.3

선택된 모델을 5일 간의 데이터를 이용해 사용자의 손동작 특성에 적응시킨 결과 UD 모델의 인식률에 근접한 인식률을 얻을 수 있었다. UI 모델의 경우 사용자 적응 후에도 낮은 인식률을 보였는데, 이는 UD 모델에 비해 방대한 노드 구조가 일반화 성능과 적응 효율을 저하시키기 때문이다.

6. 결론 및 추후 과제

6.1 결론

이 논문에서는 강인한 손동작 인식을 위해 다변량 퍼지 의사결정트리의 학습 및 분류 방법을 제안했다. 다변량 퍼지 의사결정트리는 기존의 퍼지 의사결정트리에 비해 적은 수의 노드를 가지기 때문에 더 높은 일반화 능력을 갖는다. 실험을 통해 FDT에 비해 향상된 분류 및 사용자 적응 능력을 보였다. 또한 최대 우도 모델 비

교를 이용한 모델 선택 방법과 gradient descent 기반의 사용자 적응 방법을 제안했으며 선택된 모델의 적응 능력이 UI 모델에 비해 높다는 것을 실험을 통해 보였다.

6.2 추후 연구 과제

의사결정트리의 크기를 줄이는 방법으로 가지치기 (pruning) 방법도 있지만 별도의 검증데이터를 필요로 하기 때문에 이 논문에서는 다루지 않았다. 학습 및 검증 데이터가 충분하다면 가지치기를 통해 필요 없는 노드를 더욱 줄일 수 있으며 더 높은 사용자 적응 성능을 얻을 수 있을 것이라 예상된다.

이 논문에서 제시한 사용자 적응 방법은 각 노드의 퍼지 소속함수의 파라미터만 적응시키는 것이지만 각 노드의 속성 벡터도 사용자 적응의 대상이 될 수 있다. 속성 벡터를 사용자 특성에 맞게 적응시킨다면 의사결정트리의 구조를 변화시키는 효과가 있으므로 모델을 사용자 특성에 더 적합하게 적응시킬 수 있을 것이라 예상된다.

참 고 문 헌

[1] Jun-Hyeong Do, Hyoyoung Jang, Sung Hoon Jung, Jinwoo Jung, Zeungnam Bien, "Soft Remote Control System in the Intelligent Sweet Home," Proc. of IEEE Int. conf. on IROS, pp. 3984-3989, 2005

[2] Sung Hoon Jung, Jun-Hyeong Do, and Z. Zenn Bien, "Adaptive Hand Motion Recognition in Soft Remote Control System," Proceedings of the 7th International Workshop on Human-friendly Welfare Robotic Systems (HWRS 2006), Oct. 22-24, 2006.

[3] 정성훈, 동작 유사도와 추이 예측을 이용한 한국 수화 인식에서의 사용자에 대한 적응, 석사 학위 논문, 한국과학기술원, 2007.

[4] 김대진, 퍼지 신경망을 이용한 영상 기반 개인화 얼굴 표정 인식 시스템, 박사 학위 논문, 한국과학기술원, 2004.

[5] Moonjin Jeon, Jun-Hyeong Do, Sang-Wan Lee, Kwang-Hyun Park, Zeungnam Bien, "Hand Motion Recognition using Fuzzy Decision Tree", Proceedings of the 8th International Workshop on Human-friendly Welfare Robotic Systems (HWRS 2007), Daejeon, Korea, pp. 123-127, Oct. 21-23, 2007.

[6] T. M. Mitchell, Machine Learning, The McGraw-Hill Co., 414p, 1997.

[7] Y. Kim, K.-H. Park, K.-H. Seo, C. H. Kim, W.-J. Lee, W.-G. Song, J.-H. Do, J.-J. Lee, B. K. Kim, J.-O. Kim, J.-T. Lim, and Z. Z. Bien, "A report on questionnaire for developing intelligent sweet home for the disabled and the elderly in Korean living conditions," Proc. of the 8th Int. Conf. on Rehabilitation Robotics (ICORR 2003), Daejeon, Korea, Apr. 22-25, pp. 171-174, 2003.

[8] Cezary Z. Janikow, "Fuzzy Decision Trees : Issues and Methods," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 28, No. 1, Feb 1998

[9] Koen-Myung Lee, Kyung-Mi Lee, Jee-Hyong Lee, Hyung Lee-Kwang, "A Fuzzy Decision Tree Induction Method for Fuzzy Data," 1999 IEEE International Fuzzy Systems Conference Proceedings, August 22-25, 1999, Seoul, Korea

[10] Olcay Taner Yildiz and Ethem Alpaydin, "Linear Discriminant Trees," International Journal of Pattern Recognition and Artificial Intelligence, Vol. 19, No. 3 (2005) 323-353.

[11] Alpaydin, E., Introduction to Machine Learning, The MIT Press (2004).

[12] H.-C. Fu, H.-Y. Chang, Y. Y. Xu, and H.-T. Pao, "User adaptive handwriting recognition by self-growing probabilistic decision-based neural networks," IEEE Transactions on Neural Networks, vol. 11, no. 6, pp. 1373-1384, 2000.

[13] C. J. Merz and P. M. Murphy, "UCI repository for machine learning data-bases," <http://www.ics.u ci.edu/mlearn/MLRepository.html>, Department of Information and Computer Science. University of California, Irvine, 1996.



전 문 진

2006 한양대학교 전자전기컴퓨터공학부(공학사)
 2008 한국과학기술원 전자전산학과(공학석사)
 2008~현재 한국항공우주연구원 다목적3호체계팀 연구원

관심분야: 패턴 인식, 기계 학습, 지능 제어



도 준 형

- 1999 한국과학기술원 전자전산학과(공학사)
- 2001 한국과학기술원 전자전산학과(공학석사)
- 2008 한국과학기술원 전자전산학과(공학박사)

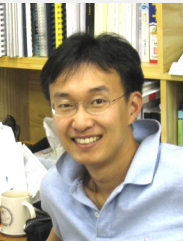
2008~현재 한국과학기술원 인간친화 복지로봇시스템 연구센터 선임급 위촉연구원
 관심분야: 인간-로봇 상호작용/인터페이스, 컴퓨터 비전, 지능 로봇



변 증 남

- 1969 서울대학교 전자공학과(공학사)
- 1972 University of Iowa 전자공학과(공학석사)
- 1975 University of Iowa 수학과(공학석사)

1975 University of Iowa 전자공학과(공학박사)
 1976~1977 University of Iowa 조교수
 1981~1982 University of Iowa 객원부교수
 1987~1988 Syracuse University 객원 연구원
 1988 일본 동경공대 객원교수
 1990~1995 한국퍼지및 지능시스템학회 회장
 2001 대한전자공학회 회장
 2003~2005 국제퍼지시스템학회(IFSA) 회장
 2003~2006 한국로봇공학회 회장
 1977~현재 한국과학기술원 전자전산학과 교수
 1999~현재 한국과학기술원 인간친화복지로봇시스템 연구센터 소장
 2005~현재 한국전력 석좌교수
 관심분야: 지능제어, 학습이론, 소프트컴퓨팅, 서비스 로봇, 재활공학



이 상 완

- 2003 연세대학교 기계전자공학부(공학사)
- 2005 한국과학기술원 전자전산학과(공학석사)
- 2005~현재 한국과학기술원 전자전산학과 박사과정

관심분야: 학습 이론, 패턴 인식, 특징 추출



박 광 현

- 1994 한국과학기술원 전자전산학과(공학사)
- 1997 한국과학기술원 전자전산학과(공학석사)
- 2001 한국과학기술원 전자전산학과(공학박사)

2005~2007 한국과학기술원 전자전산학과 BK 초빙교수
 2008~현재 광운대학교 정보제어공학과 조교수
 관심분야: 학습이론, 지능로봇, 인간-로봇 상호작용, 재활공학