

# 적응적 Multiple Kernels을 이용한 Interval Type-2 Possibilistic Fuzzy C-Means 방법

## A Novel Approach towards use of Adaptive Multiple Kernels in Interval Type-2 Possibilistic Fuzzy C-Means

주원희\* · 이정훈\*\*†

Won-Hee Joo and Frank Chung-Hoon Rhee†

\*한양대학교 전자전기제어계측공학과, \*\*한양대학교 전자통신공학과

† Electronics and Communication Engineering, Hanyang University

### 요 약

본 논문에서는 interval type-2 possibilistic fuzzy C-means(IT2PFCM) 클러스터링 방법에 multiple Gaussian kernels을 기반으로 한 possibilistic fuzzy C-means multiple kernels(PFCM-MK) 알고리즘을 결합하여 적응적인 하이브리드 클러스터링 방법인 multiple kernels interval type-2 possibilistic fuzzy C-means(IT2PFCM-MK) 방법을 제안 하였다. 일반적으로 possibilistic fuzzy C-means(PFCM) 알고리즘은 fuzzy C-means(FCM) 알고리즘의 단점인 노이즈 민감성 및 특이점 문제와 알고리즘 초기 클러스터의 Prototype에 따라 위치가 겹치는 문제를 해결하기 위해 제안 되었다. 하지만 이 방법 역시 퍼지화 파라미터 값에 따라 위와 같은 문제를 여전히 가지고 있기 때문에 이와 같은 문제를 보완하기 위해 interval type-2 퍼지 접근 방법을 이용 하는 interval type-2 possibilistic fuzzy C-means(IT2PFCM) 알고리즘을 제안 하였다. 또한 multiple kernels 함수를 interval type-2 possibilistic fuzzy C-means(IT2PFCM) 알고리즘에 적용하여 분류하기 복잡한 형태의 데이터와 노이즈가 있는 데이터에 대하여 보다 정확하고, 향상된 클러스터링을 수행할 수 있다.

**키워드** : PFCM, Interval Type-2 Fuzzy Sets, Interval Type-2 PFCM, Multiple Kernels, Fuzzy Clustering

### Abstract

In this paper, we propose a hybrid approach towards multiple kernels interval type-2 possibilistic fuzzy C-means(PFCM) based on interval type-2 possibilistic fuzzy c-means(IT2PFCM) and possibilistic fuzzy c-means using multiple kernels(PFCM-MK). In case of noisy data or overlapping cluster prototypes, fuzzy C-means gives poor performance in comparison to possibilistic fuzzy C-means(PFCM). Moreover, to address the uncertainty associated with fuzzifier parameter  $m$ , interval type-2 possibilistic fuzzy C-means(PFCM) is used. Most of the practical data available are complex and non-linearly separable. In such cases using Gaussian kernels proves helpful. Therefore, in order to overcome all these issues, we have integrated multiple kernels possibilistic fuzzy C-means(PFCM) into interval type-2 possibilistic fuzzy C-means(IT2PFCM) and propose the idea of multiple kernels based interval type-2 possibilistic fuzzy C-means(IT2PFCM-MK).

**Key Words** : PFCM, Interval Type-2 Fuzzy Sets, Interval Type-2 PFCM, Multiple Kernels, Fuzzy Clustering

## 1. 서 론

접수일자: 2014년 3월 9일

심사(수정)일자: 2014년 4월 1일

게재확정일자 : 2014년 8월 13일

† Corresponding author

이 논문은 2011년도 정부(과학기술부)의 재원으로 한국 과학재단의 지원을 받아 수행된 연구임  
(No. 2011-0025655)

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

퍼지 클러스터링에 사용되는 여러 가지 방법 중 가장 많이 사용되는 방법은 유클리디안 거리를 이용하여 멤버십을 할당하는 fuzzy C-means(FCM) 알고리즘 이다[1][2][3]. 그러나 FCM 방법은 모든 클래스 멤버십 값 합이 1이 되어야 한다는 조건 때문에 노이즈가 있는 데이터에서는 좋지 않은 결과를 나타낼 수다. 이러한 문제점을 가지고 있는 FCM 방법의 단점을 극복하기 위해 typicality를 고려하는 possibilistic C-means(PCM) 알고리즘이 제안 되었다[4].

PCM 방법은 노이즈엔 강하지만 초기 파라미터에 민감하고, 서로 가까이 위치한 prototype 들을 형성하는 패턴들의 경우에는 최종 prototype의 중심 값의 위치가 겹치는 결과가 나올 수 있다는 단점이 있다. 이러한 문제들을 해결하기 위해 FCM 과 PCM을 가중치 합으로 표현 한 possibilistic fuzzy C-means(PFCM) 알고리즘이 연구 되었다

[5]. 그러나 PFCM 방법 또한 퍼지화 파라미터  $m$  값에 따라 위에서 제시한 문제를 여전히 가지고 있다.

따라서 본 논문에서는 PFCM의  $m$  값에 uncertainty를 부여하여 PFCM 알고리즘의 성능을 개선하는 interval type-2 possibilistic fuzzy C-means(IT2PFCM) 알고리즘을 제안 하였다. 또한 고리형 데이터, 노이즈가 심한 데이터, 또는 많이 복잡한 형태의 겹쳐지는 데이터들에 대하여 더 좋은 성능을 얻기 위해 IT2PFCM 알고리즘에 Gaussian multiple kernels 방법을 적용하여 IT2PFCM multiple kernels 방법을 제안하였다.

일반적으로 Gaussian multiple kernels 함수를 적용하여 입력 속성(feature) 공간이 아닌 커널 속성 공간으로 변환하여 클러스터링을 수행한다[6]. 가우시안 커널 속성 공간으로 정의된 변환 함수에 의한 두 함수 값의 내적(inner product) 커널 함수로 정의된다[7]. 이와 같이 IT2PFCM 방법에 다중 가우시안 커널을 적용하여 데이터를 입력 속성 공간이 아닌 커널 속성 공간으로 변환하여 보다 복잡한 형태의 분포를 갖는 데이터에 정확한 클러스터링이 가능하다.

제안된 내용을 바탕으로 하여 본 논문은 다음과 같이 구성된다. 두 번째 절에서 PFCM 알고리즘에 대하여 설명하고 세 번째와 네 번째 절에서 제안된 IT2PFCM 알고리즘과 IT2PFCM multiple kernels 알고리즘에 대해 소개한다. 다섯 번째 절에서는 제안된 알고리즘과 기존의 알고리즘들 간의 성능을 비교하고 마지막 절에서 결론을 맺겠다.

## 2. PFCM 알고리즘

PCM과 FCM은 패턴에 멤버십을 할당하는 알고리즘이지만 FCM과 달리 PCM은 오직 패턴과 prototype 간의 거리만을 이용해 멤버십을 할당한다. PCM 알고리즘은 하나의 패턴과 하나의 중심 값 간의 절대거리로 typicality를 할당한다. PFCM은 위 두 가지의 특성을 모두 가지고 있다. 따라서 다음과 같이 목적함수(objective function)를 최소화 하는 방향으로 클러스터링 한다.

$$J_{m,\eta}(U, T, V; X) = \sum_{k=1}^n \sum_{i=1}^c (au_{ik}^m + bt_{ik}^\eta) \times \|x_k - v_i\|^2 + \sum_{i=1}^c \gamma_i \sum_{k=1}^n (1 - t_{ik})^\eta \quad (1)$$

$$\sum_{i=1}^c u_{ik} = 1, 0 \leq u_{ik}, t_{ik} \leq 1, m > 1, \eta > 1, \gamma_i > 0 \quad (2)$$

식 (1)에서  $x_k$ 는  $k$ 번째 입력 패턴이고,  $v_i$ 는  $i$ 번째 클러스터 중심 값이다.  $u_{ik}$ 는 입력 패턴  $k$ 가 클러스터  $i$ 에 소속되는 멤버십 값을 나타낸다.  $m$ 은 퍼지화의 정도를 나타내는 상수로  $m \in (1, \infty)$  조건을 만족시키는 상수이다.  $t_{ik}$ 는 입력 패턴  $k$ 가 클러스터  $i$ 에 소속되는 typicality를 나타낸다. 일반적으로 PCM의 경우  $\gamma_i$ 는  $i$ 번째 클러스터의 typicality가 0.5 되는 지점의 거리를 가리키는데 이는 FPCM에서도 동일하게 적용된다. 식 (1)을 최소화하기 위한 멤버십은 다음과 같이 구한다.

$$u_{ik} = \left( \sum_{j=1}^c \left( \frac{d_{ik}}{d_{jk}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (3)$$

식(1)을 최소화하기 위한 typicality 및  $\gamma_i$ 는 아래 (4)와 같이 구한다.

$$t_{ik} = \left( \left( 1 + \frac{b}{\gamma_i} d_{ik}^2 \right)^{\frac{1}{\eta-1}} \right)^{-1} \quad \gamma_i = K \frac{\sum_{k=1}^n u_{ik}^m d_{ik}^2}{\sum_{k=1}^n u_{ik}^m} \quad (4)$$

식 (4)에서  $K=1$  일 경우 클러스터  $i$ 의 평균 내부 거리 값을  $\gamma_i$ 로 할당한다. 이때 최적의 중심 값을 위한 필요조건은 식 (3),(4) 사용하여 다음과 같이 중심 값을 구한다.

$$v_i = \frac{\sum_{k=1}^n (au_{ik}^m + bt_{ik}^\eta) x_k}{\sum_{k=1}^n (au_{ik}^m + bt_{ik}^\eta)} \quad (5)$$

PFCM 알고리즘은 멤버십과 typicality 의 가중치의 합으로  $a$ 와  $b$ 의 값에 따라 가중치를 조정 할 수 있다.  $a$ 와  $b$  값의 변화에 따라 FCM 방법과 PCM 방법의 장점을 최대화 할 수 있다. 적절한  $a, b$  값을 이용하여 데이터 노이즈의 영향을 최소화 하고, 중심 값이 겹쳐지는 문제도 해결할 수 있다.

## 3. Interval Type-2 PFCM 알고리즘

### 3.1 Interval Type-2 Fuzzy sets

일반적으로 패턴인식에서 패턴의 uncertainty를 나타내는데 type-1 fuzzy set(T1FS) 사용 하였다. T1FS에서 퍼지화 상수  $m$ 에 따라 결과가 달라진다. 그림 1과 같이 클러스터의 부피가 다른 경우 하나의  $m$ 값으로는 좋은 성능을 보일 수 없다.

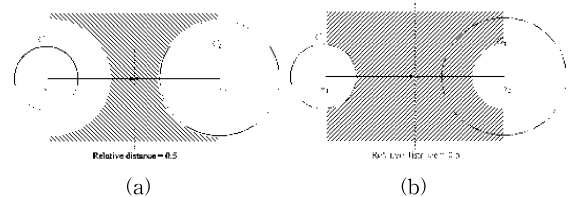


그림 1.  $m$  값에 따른 2개의 클러스터 퍼지 지역  
Fig. 1. Fuzzy cluster region due to  $m$

1(a)는 부피가 다른 2개의 클러스터에서 작은  $m$ 값을 설정했을 때를 보여준다.  $C_2$  클러스터에 퍼지 멤버십 값을 가지는 구간이 겹쳐 있기 때문에 상대적으로  $C_1$  클러스터에 패턴들이 많이 할당이 된다. 그림 1(b)와 같이 큰  $m$  값을 설정했을 때에는 비슷한 멤버십 값이 할당되기 때문에 좋은 성능을 보일 것 같지만,  $C_1$  클러스터의 센터 값이  $C_2$  클러스터로 이동하는 경향을 보인다.

그러나 T1FS 에서 존재하는 퍼지화 상수  $m$ 에 대한 불확실성이 존재 한다. 따라서 이러한 불확실성을 제어하여

더 좋은 결과를 낼 수 있도록 type-2 fuzzy set(T2FS) 으로 확장한다[8-12].

$$\tilde{A} = \int_{x \in X} \mu_{\tilde{A}}(x)/x = \int_{x \in X} \left[ \int_{u \in J_x} f_x(u)/u \right] / x \quad (6)$$

T2FS 이 T1FS 보다 우수한 성능을 나타내지만, 많은 계산 량을 필요로 한다는 단점이 있다. 이러한 단점을 보완하기 위해 secondary membership degree가 1인 interval type-2 fuzzy set(IT2FS)을 사용한다. IT2FS을 나타내면 다음과 같다.

$$\tilde{A} = \left[ \int_{u \in J_x} 1/u \right] / x \quad (7)$$

식 (7)에서 secondary membership degree가 모두 같기 때문에 T1FS 처럼 사용할 수 있다. 이와 같은 IT2FS의 특징을 이용하여  $m$ 값에 따른 퍼지 지역을 설정하면 다음과 같다. 그림 1과 같이 클러스터 크기에 맞게 퍼지 지역을 설정할 순 없기 때문에, 2개의  $m_1, m_2$  값을 이용하여 uncertainty 을 줄으로써 클러스터 부피에 맞는 적당한 퍼지 지역을 형성할 수 있다.

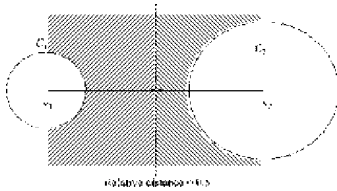


그림 2. Interval type-2  $m$  값에 따른 fuzzy 지역  
Fig. 2. Fuzzy area using interval type-2  $m$  value

### 3.2 Extension to Interval Type-2 Fuzzy sets

PFCM 목적함수에서 퍼지화 파라미터  $m$ 의 값에 따라  $y_i$ 의 값이 변화한다. IT2FS을 이용하여 두 개의  $m$  값을 사용하는 interval type-2 알고리즘으로 확장하여 PFCM 보다 나은 성능을 얻어 낼 수 있다.

IT2FS으로 확장위하여  $m$ 에 대한 uncertainty를 표현하기 위해 primary fuzzy set인 입력 패턴을 IT2FS으로 할당한다. 이를 위해 primary membership 함수를 사용 upper 와 lower membership function 을 만든다. 최종 클러스터링의 퍼지화 정도를 결정하는  $m$ 을 이용하여 upper 와 lower membership function을 구해야 한다.  $m$ 은 퍼지화 파라미터값 으로 최종 클러스터링의 퍼지화 정도를 결정하는 값이다.  $m$ 에 따른 upper 와 lower membership function은 식(8)(9)과 같다.

$$\bar{u}_{ik} = \begin{cases} \frac{1}{\sum_{k=1}^c \left( \frac{d_{ij}}{d_{ik}} \right)^{\frac{2}{m_1-1}}}, & \text{if } \frac{1}{\sum_{k=1}^c \left( \frac{d_{ij}}{d_{ik}} \right)} > \frac{1}{C} \\ 1 \\ \frac{1}{\sum_{k=1}^c \left( \frac{d_{ij}}{d_{ik}} \right)^{\frac{2}{m_2-1}}}, & \text{otherwise} \end{cases} \quad (8)$$

$$u_{ik} = \begin{cases} \frac{1}{\sum_{k=1}^c \left( \frac{d_{ij}}{d_{ik}} \right)^{\frac{2}{m_1-1}}}, & \text{if } \frac{1}{\sum_{k=1}^c \left( \frac{d_{ij}}{d_{ik}} \right)} < \frac{1}{C} \\ 1 \\ \frac{1}{\sum_{k=1}^c \left( \frac{d_{ij}}{d_{ik}} \right)^{\frac{2}{m_2-1}}}, & \text{otherwise} \end{cases} \quad (9)$$

$\bar{u}_{ik}, u_{ik}$ 에 따라  $y_i$ 의 값이 다음과 같이 구해진다.

$$\bar{\gamma}_i = k \frac{\sum_{k=1}^n \bar{u}_{ik} d_{ik}^2}{\sum_{k=1}^n \bar{u}_{ik}}, \quad \gamma_i = k \frac{\sum_{k=1}^n u_{ik} d_{ik}^2}{\sum_{k=1}^n u_{ik}} \quad (10,11)$$

위에서 얻어진  $y_i$  값을 사용하여 upper 와 lower typicality function을 결정한다.

$$\bar{t}_{ik}(x_i) = \left( \left( 1 + \frac{b}{\gamma_i} d_{ik}^2 \right)^{\frac{1}{\eta-1}} \right)^{-1} \quad (12)$$

$$t_{ik}(x_i) = \left( \left( 1 + \frac{b}{\gamma_i} d_{ik}^2 \right)^{\frac{1}{\eta-1}} \right)^{-1} \quad (13)$$

### 3.3 Type Reduction and Defuzzification

입력 패턴들에 대하여 각 클러스터에 대한 upper 와 lower 멤버십을 구하고 나면 각 클러스터들에 대한 중심 값을 갱신해야 한다. 이때 멤버십은 T2FS 이지만 중심 값은 crisp 한 값이기 때문에 위의 방법으로는 값을 구할 수가 없다. 따라서 중심 값 갱신을 위해서 T2FS을 T1FS으로 바꾸는 type reduction 과 type-1의 값을 crisp 한 값으로 바꾸는 defuzzification 이 필요하다. 본 논문에서 KM알고리즘을 이용하여 type reduction을 수행한다.

$$v = 1.0/[v_L, v_R] \\ v_L = \text{least value of left side} \\ v_R = \text{greatest value of right side} \quad (14)$$

$$1.0/v_{i,L} = 1.0 / \frac{\sum_{k=1}^n (au_{ik,L}^m + bt_{ik,L}^\eta) x_k}{\sum_{k=1}^n (au_{ik,L}^m + bt_{ik,L}^\eta)} \quad (15)$$

$$1.0/v_{i,R} = 1.0 / \frac{\sum_{k=1}^n (au_{ik,R}^m + bt_{ik,R}^\eta) x_k}{\sum_{k=1}^n (au_{ik,R}^m + bt_{ik,R}^\eta)}$$

최종 center는 아래와 같이 defuzzification 하여 구한다.

$$Crisp v = \frac{v_L + v_R}{2} \quad (16)$$

<b>Interval Type-2 PFCM Algorithm</b>
Set arbitrary fuzzifier $m_1, m_2$ and typicality $\eta$ ; Set initial arbitrary center $v' = (v'_1, \dots, v'_C)$ ; Set comparison = FALSE; <b>WHILE</b> (comparison = FALSE) Set $v_m = v'$ ; <b>FOR</b> all $C$ centers ( $i = 1, \dots, C$ ) <b>DO</b> Compute distance $d_i(x_k) = \ x_k - v'_i\ $ Compute membership $\bar{u}_{ik}, \underline{u}_{ik}$ and typicality $\bar{t}_{ik}, \underline{t}_{ik}$ using (8)-(13); Compute centroid $v'_{RU}$ and $v'_{RT}$ by (15) and $u_{ik} = \frac{\bar{u}_{ik} + \underline{u}_{ik}}{2}$ and $t_{ik} = \frac{\bar{t}_{ik} + \underline{t}_{ik}}{2}$ Sort all pattern feature indexes in ascending order; (i.e., Sorted feature 1: $x_{11} \leq \dots \leq x_{NM}$ Sorted feature $M$ : $x_{1M} \leq \dots \leq x_{NM}$ )

#### 4. Multiple Kernel Interval Type-2 PFCM

##### 4.1 Multiple Kernel PFCM 알고리즘

일반적으로 커널 방법은 공간 변환 함수를 사용하여 입력 데이터들을 입력 속성 공간에서 커널 함수를 통한 커널 속성 공간으로 변환하여 주는 것이다[13]. 커널 속성 공간 변환을 통하여 입력 속성 공간에서 직선이 아닌 경계면을 갖으며 겹치는 데이터들에 대하여 더 쉽게 구분하기 위해서 커널 속성 공간으로 바꾸기 위한 것이다. 입력 공간에서 데이터를  $X_i, i=1, \dots, N$  이라 한다면 함수를 통해 커널 속성 공간으로 변환된 데이터는  $\Phi(X_i)$ 로 나타낼 수 있다.

일반적인 PFCM과 마찬가지로 Kernel-PFCM의 경우에도 다음의 목적 함수를 최소화 하는 것을 목표로 한다.

$$J^{\Phi} = \sum_{k=1}^n \sum_{i=1}^c (au_{ik}^m + bt_{ik}^n) \times d_{ij}^2 + \sum_{i=1}^c \gamma_i \sum_{k=1}^n (1 - t_{ik})^{\eta} \quad (17)$$

커널  $K$ 에 대한 입력 공간상에서 패턴  $x_j$ 와 cluster prototype  $v_j$ 의 커널 속성 공간상 거리  $d_{ij}$ 는 커널 함수에 의해 (18)과 같이 표현 된다.

$$d_{ij} = \|\Phi(x_j) - \Phi(v_i)\|^2 = \Phi(x_j)\Phi(x_j) + \Phi(v_i)\Phi(v_i) - 2\Phi(x_j)\Phi(v_i) = K(x_j, x_j) + K(v_i, v_i) - 2K(x_j, v_i) \quad (18)$$

일반적으로 커널의 개수가  $S$ 인 multiple kernels을 가정했을 경우 가우시안 커널을 이용한 새로운 Gaussian multiple kernels  $\tilde{k}$  은 다음과 같다[6].

$$\tilde{k}^{(j)}(x_j, v_i) = \sum_{l=1}^s \frac{w_{il}}{\sigma_l} \frac{\exp\left(-\frac{\|x_j - v_i\|^2}{2\sigma_l^2}\right)}{\sum_{t=1}^s \frac{w}{\sigma_t}} \quad (19)$$

최적화 방법을 이용하여 아래의 PFCM 목적함수를 최소화 하여야 한다. 아래의 목적함수의 최소화를 통하여 resolution specific weight  $w_{ij}$  값과, 멤버십 값  $u_{ij}$ , 클러스터 prototype  $v_i$  값들이 결정된다.

$$J_{m,\eta}(U, T, V; X) = 2 \sum_{k=1}^n \sum_{i=1}^c (au_{ik}^m + bt_{ik}^n) \times (1 - \tilde{k}^{(j)}(x_j, v_i)) + \sum_{i=1}^c \gamma_i \sum_{k=1}^n (1 - t_{ik})^{\eta} \quad (20)$$

##### 4.2 Multiple Kernel Interval Type-2 PFCM 알고리즘

일반적인 PFCM 알고리즘에서 퍼지화 상수  $m$  에 존재하는 불확실성을 해결하기 위하여 multiple kernels PFCM 알고리즘을 IT2FS으로 확장한다.  $N$ 개의 데이터와,  $C$ 개의 클러스터,  $S$ 개의 커널이 존재 할 경우 아래와 같은 가우시안 커널 목적 함수를 최소화 하는 방법을 통하여 cluster prototype을 얻을 수 있다.

$$J(U, V, W) = 2 \sum_{i=1}^c \sum_{j=1}^N u_{ij}^m d_{ij}^2 \quad (21)$$

where

$$d_{ij}^2 = \left( \frac{\exp\left(-\frac{\|x_j - v_i\|^2}{2\sigma_i^2}\right)}{\sum_{t=1}^s \frac{w}{\sigma_t}} \right) \quad (22)$$

클러스터 prototype 은 클러스터의 센터  $v_i$ 에 대하여 목적함수를 최적화 하는 방법으로 계산되어진다[14]. 다음에는 IT2FS을 이용하여 각 패턴들에 대하여 가장 작은 멤버십 값과 가장 큰 멤버십 값을 계산한다. 이 최적화된 멤버십 값은 crisp 값인  $v_j$  계산하기 위하여 사용된다.  $v_R$  과  $v_L$  을 계산하기 위하여 멤버십의 upper 또는 lower bound를 결정하는 것이 필요하다. 아래의 주어진 식에 의해 다음과 같이 정리 된다[8][15].

For  $v_L$

IF ( $v(i \leq k)$ ) THEN  $u_{ij} = \bar{u}_{ij}$   
 ELSE  $u_{ij} = \underline{u}_{ij}$

$$v_{iL} = \frac{\sum_{j=1}^N u_{ij}^m \bar{K}^{(i)}(x_j, v_i) x_j}{\sum_{j=1}^N u_{ij}^m \bar{K}^{(i)}(x_j, v_i)} \quad (23)$$

ENDIF

For  $v_R$

IF ( $v(i > k)$ ) THEN  $u_{ij} = \bar{u}_{ij}$   
 ELSE  $u_{ij} = \underline{u}_{ij}$

$$v_{iR} = \frac{\sum_{j=1}^N u_{ij}^m \underline{K}^{(i)}(x_j, v_i) x_j}{\sum_{j=1}^N u_{ij}^m \underline{K}^{(i)}(x_j, v_i)} \quad (24)$$

ENDIF

최종적으로 얻어진  $v_R$  과  $v_L$  을 이용하여 crisp center 값은 아래와 같이 defuzzification 하여 구한다.

$$v_i = \frac{v_{iL} + v_{iR}}{2} \quad (25)$$

이렇게 최적화 함수를 통하여 얻어진 클러스터 prototype  $v_i$  값과, 멤버십 값인  $u_{ij}$  값을 이용하여 resolution-specific weight 값인  $w_{ij}$  값은 다음과 같이 업데이트 된다.

$$w_{il}^{(new)} = w_{il}^{(old)} - \rho \frac{\partial J}{\partial w_{il}} \quad (26)$$

where

$$\frac{\partial J}{\partial w_{il}} = -2 \sum_{i=1}^N \frac{u_{ij}^m}{\sigma_l \sum_{t=1}^S \frac{w_t}{\sigma_t}} (K_l(x_j, v_i) - \tilde{K}^{(i)}(x_j, v_i)) \quad (27)$$

여기서  $\rho$  값은 학습률 파라미터로 gradient descent 방법을 이용하고 IT2PFCM에서 설명 하였던 type-reduction 과 hard partitioning 을 통하여 클러스터링을 수행한다.

### 5. 시뮬레이션 및 결과 고찰

알고리즘의 성능을 알아보기 위해 여러 데이터 패턴과 이미지에 본 논문에서 제안된 multiple kernels을 적용한 IT2PFCM(IT2PFCM-MK) 알고리즘 과 IT2PFCM 알고리즘 및 IT2PCM 알고리즘을 비교 하였다. 모든 실험에서 알고리즘의  $a$  와  $b$  값이 각각 0과 1, 1과 0 또는 1과1로 적용되는 경우의 수를 두어 최적의 값을 찾게 하였다.  $m_1$  과  $m_2$  값을 [1.1, 5] 범위에서 실험을 하였다. Multiple kernels에서  $\sigma$  값은 가장 일반적인 가우스 함수 값으로 고정하였다.

#### 5.1 Iris34 데이터

Iris34 데이터는 세 개의 클러스터로 나누어져 있고, 각 클러스터 마다 50개의 데이터를 가지고 있다. 실험 결과 크기가 다른 세 개의 클러스터 데이터에서 제안된 IT2PFCM-MK 알고리즘이 더 정확한 클러스터링을 하였다. 그림 3을 통하여 보다 정확하게 클러스터링 center를 찾아내는 것을 알 수 있다.

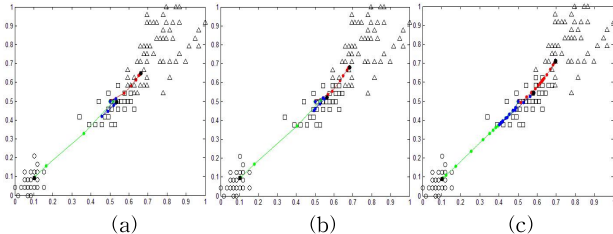


그림 3. Iris34 데이터의 클러스터링 결과:  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘

Fig. 3. Clustering result for Iris34 data  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

#### 5.2 Iris56 데이터

Iris56 데이터는 세 개의 클러스터로 나누어져 있고, 각 클러스터 마다 50개의 데이터를 가지고 있다. 실험 결과 서로 크기가 다른 세 개의 클러스터 데이터에서 제안된 IT2PFCM-MK 알고리즘이 더 정확한 클러스터링을 하였다. 서로 다른 크기를 가진 복잡한 데이터를 클러스터링 하는데 제안된 알고리즘이 더 좋은 성능을 보인다.

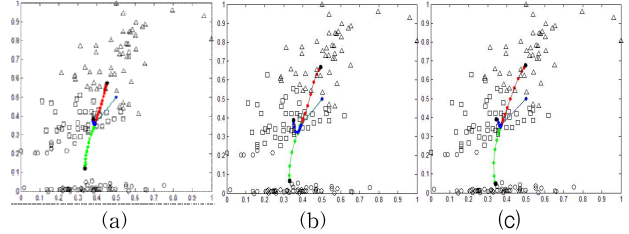


그림 4. Iris56 데이터의 클러스터링 결과:  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘

Fig. 4. Clustering result for Iris56 data  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

#### 5.3 Asymmetric circle 데이터

Asymmetric circle 데이터는 크기가 다른 두 개의 클러스터로 이루어져 있다. 제안된 알고리즘이 IT2PFCM 방법 과 PFCM 방법에 비해 인식률이 더 뛰어난 클러스터링 결과를 나타내는 것을 확인 할 수 있다.

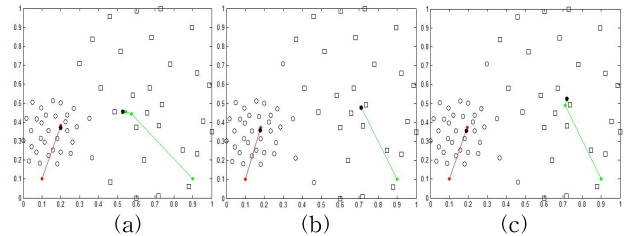


그림 5. Asymmetric circle 데이터의 클러스터링 결과:  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘

Fig. 5. Clustering result for Asymmetric circle data  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

#### 5.4 이미지 데이터 1 (길, 숲, 하늘)

이번 실험은 길, 숲, 하늘 세부분으로 이루어진 200×200 사이즈 영상을 분할해 보겠다.



그림 6. 길, 숲, 하늘 영상:  
(a)Intensity 영상 (b)가우시안 속성 영상

Fig. 6. Road, Forest, Sky feature  
(a)Intensity feature (b)Gaussian feature

위 그림 과 같이 두 개의 속성 데이터를 모두 이용하여 영상을 클러스터링 하였다. 영상 분할을 위해 세부분에서 100개씩의 샘플 데이터를 추출하여 학습 시킨 후 학습된 결과에 의해 영상을 분할하도록 하였다. 주어진 300개의 샘플 데이터로 클러스터링 한 결과와 영상을 분할한 결과는 다음과 같다.

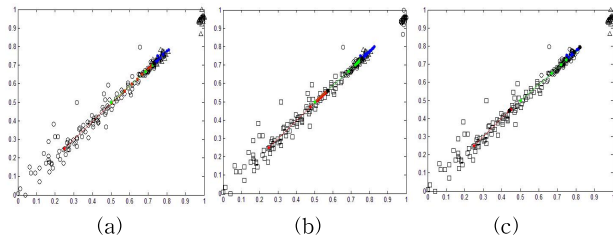


그림 7. 길, 숲, 하늘 영상 데이터의 클러스터링 결과:  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘  
Fig. 7. Clustering result for Image data  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

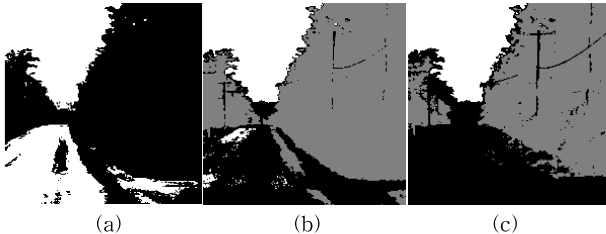


그림 8. 길, 숲 하늘 영상데이터의 영역 분할 결과  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘  
Fig. 8. Clustering result for Image segmentation  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

표 1. 길, 숲, 하늘 영상의 인식률 및 Iteration 비교  
Table 1. Comparison on the basis of classification rate and number of Iterations taken

	IT2PCM	IT2PFCM	IT2PFCM-MK
Recognition(%)	21.42	80.06	86.46
Iterations	14	51	101

### 5.5 이미지 데이터 2 (길, 숲, 하늘)

그림 9와 같이 길, 숲, 하늘 세부분으로 이루어진 다른 종류의 영상으로 200×200 사이즈를 가진다. 실험의 방법은 위의 예제와 동일하다. 세부분에서 100개씩의 샘플 데이터를 추출하여 학습 시킨 후 학습된 결과에 의해 영상을 분할하도록 하였다. 주어진 300개의 샘플 데이터로 클러스터링 한 결과와 영상을 분할한 결과는 다음과 같다.



그림 9. 길, 숲, 하늘 영상:  
(a)Intensity 영상 (b)가우시안 속성 영상  
Fig. 9. Road, Forest, Sky feature  
(a)Intensity feature (b)Gaussian feature

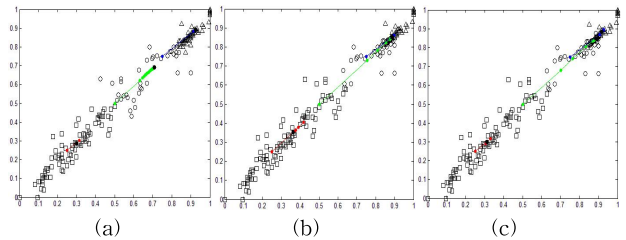


그림 10. 길, 숲, 하늘 영상 데이터의 클러스터링 결과:  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘  
Fig. 10. Clustering result for Image data  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

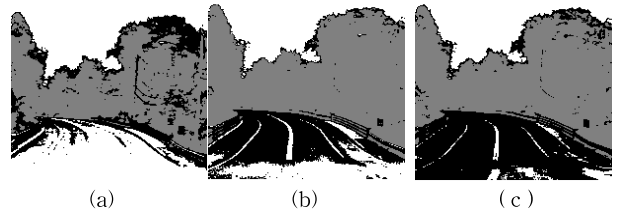


그림 11. 길, 숲 하늘 영상데이터의 영역 분할 결과  
(a)IT2PCM (b)IT2PFCM (c)제안된 알고리즘  
Fig. 11. Clustering result for Image segmentation  
(a)IT2PCM (b)IT2PFCM (c)Proposed Algorithm

표 2. 길, 숲, 하늘 영상의 인식률 및 Iteration 비교  
Table 2. Comparison on the basis of classification rate and number of Iterations taken

	IT2PCM	IT2PFCM	IT2PFCM-MK
Recognition(%)	56.39	80.46	83.80
Iteration	51	32	80

## 6. 결론 및 향후 연구

본 논문에서 제안된 IT2PFCM-MK 방법이 노이즈를 가지고 있는 복잡한 형태의 데이터에 대하여 기존 방법보다 우수한 결과를 나타내는 것을 확인하였다.

IT2FS 방법에서 퍼지화 상수  $m$ 의 결과를 도출 하는데 있어 중요한 역할을 한다. 따라서 향후 연구에서는 최적의  $m$  값에 대한 연구가 필요할 것이다. 또한 PFCM 알고리즘의 특성상 가중치 값인  $a, b$  값이 중요하므로 적응적으로 최적화된  $a, b$  값을 찾아내는 연구를 진행 할 예정이다.



## References

- [1] M. Hung and D. Yang, "An effective fuzzy c-means clustering algorithm," *Proceedings of IEEE*, 2001.
- [2] G. Raju, B. Thomas, S. Tobgay, and S. Kumar, "Fuzzy clustering methods in data mining: a comparative case analysis," *ICACTE*, pp. 489-493, 2008.
- [3] J. Bezdek, *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum, 1981.
- [4] R. Krishnapuram and J. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 2, pp. 98-110, May 1993.
- [5] N. Pal and J. Bezdek, "A possibilistic fuzzy c-means clustering algorithm," *IEEE Trans. Fuzzy Syst.*, vol. 13, pp. 517-530, Aug. 2005.
- [6] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, 1995.
- [7] M. Girolami, "Mercer kernel-based clustering in feature space," *IEEE Trans. Neural Networks*, vol. 13, no. 5, pp. 780-784, May 2002.
- [8] C. Hwang and F. Rhee, "Uncertain fuzzy clustering: interval type-2 fuzzy approach to C-means," *IEEE Trans. Fuzzy Syst.*, vol. 15, no. 1, pp. 107-120, Feb. 2007.
- [9] F. Rhee, "Uncertain fuzzy clustering: insights and recommendations," *IEEE Computational Intelligence Magazine*, vol. 2, no. 1, pp. 44-56, Feb. 2007.
- [10] F. Rhee and C. Hwang, "A type-2 fuzzy C-means clustering algorithm," in *Proc. 2001 Joint Conf IFSA/NAFIPS*, pp. 1926-1919, Jul. 2001.
- [11] C. Hwang and F. Rhee, "An interval type-2 fuzzy C spherical shells algorithm," in *Proc. 2004 Int. Conf. Fuzzy Syst.*, vol. 2, pp. 1117-1122, Jul. 2004.
- [12] F. Rhee and C. Hwang, "An interval type-2 fuzzy perceptron," in *Proc. 2002 Int. Conf. Fuzzy Syst.*, vol. 2, pp. 1331-1335, May 2002.
- [13] B. Scholkopf and A. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT Press, 2001.
- [14] N. Baili and H. Frigui, "Fuzzy clustering with multiple kernels," in *Proc. 2011 Int. Conf. Fuzzy Syst.*, pp. 490-496, June 2011.

- [15] N. Karnik and J. Mendel, "Applications of type-2 fuzzy logic systems to forecasting of time series," *Information Sciences*, vol. 120, pp. 89-111, Nov. 1999.

## 저 자 소개



### 주원희(Won-Hee Joo)

2007년: 숭실대학교 정보통신공학과 졸업  
 2009년: 숭실대학교 정보통신공학과 석사  
 2010년~현재: 한양대학교 대학원  
 전자통신공학과 박사과정

관심분야 : Fuzzy systems, Pattern recognition,  
 Neural networks  
 E-mail : joowonhee@fuzzy.hanyang.ac.kr



### 이정훈(Frank Chung-Hoon Rhee)

1985년: University of Southern California(USC), Dept. of Electrical Eng., BSEE  
 1989년: University of Missouri, Dept. of Electrical & Computer Eng., MSEE  
 1990년~1993년: University of Missouri, Dept. of Electrical & Computer Eng., Ph.D  
 1994년~1995년: 한국전자통신연구원 선임연구원  
 1995년~현재: 한양대학교 전자통신공학부 교수  
 2003년~2009년: 세계 퍼지학회(IFSA) 한국 지부 대표  
 2007년~현재: IEEE Computational Intelligence Society Fuzzy Systems Technical Committee Member

관심분야 : Computer Vision, Pattern Recognition, and all aspects of Computational Intelligence  
 E-mail : frhee@fuzzy.hanyang.ac.kr