학습시간을 개선한 Fuzzy c-means 알고리즘

김 형 철 , 조 제 황
동신대학교 전기전자공학부

The Enhancement of Learning Time in Fuzzy c-means algorithm

Hyung Cheol Kim , Che Hwang Cho
Dept. of Electrical & Electronic Eng., Dongshin Univ.
khc519@hanmail.net , chcho@dongshinu.co.kr

Abstract

The conventional K-means algorithm is widely used in vector quantizer design and clustering analysis. Recently modified K-means algorithm has been proposed where the codevector updating step is as fallows: new codevector = current codevector + scale factor (new centroid - current codevector). This algorithm uses a fixed value for the scale factor. In this paper, we propose a new algorithm for the enhancement of learning time in fuzzy c-means algorithm. Experimental results show that the proposed method produces codebooks about 5 to 6 times faster than the conventional K-means algorithm with almost the same performance.

Ⅰ. 서 론

벡터양자화를 이용한 데이터 압축기법은 단순하고 압축률이 높기 때문에 음성이나 영상의 압축방법으로 많은 관심을 받아왔으며, 부호화 단계에서 주어진 벡터를 양자화하기 위한 기준패턴을 만들어 두어야하는데 이들 패턴들의 집합인 코드북의 설계는 벡터양자화의 성능에 중요한 영향을 미친다[1]-[2]. 따라서 주어진 학습 데이터를 사용해 이 학습 데이터를 가장 잘 대표할 수 있는 코드벡터의 집합인 코드북의 설계는 중요한 문제이며, 이러한 문제는 패턴인식, 음성인식, 그리고 화상인식 등의 분야에서 연구되고 있다. 코드북을 설계하는 알고리즘에는 여러 가지가 있으나 그중 가장 대표적인 방법은 K-means 알고리즘이며, 이 알고리즘은 또한 LBG(Linde, Buzo, and Gray) 알고리즘, GLA (Generalized Lloyd Algorithm) 등으로 불린다. 이 알고리즘은 주어진 초기 코드북에 대하여 최소거리 조건과 중심조건을 이용하여 평균거리 오차가 최소가 되는 코드북을 반복조건에 따라 연속적으로 생성하는 것으로 수렴속도가 비교적 빠르다. 그러나 K-means 알고리즘은 국부적으로 최적화 되고, 그 성능이 초기 코드북에 크게 의존한다는 문제점을 가지고 있어 이를 보완하기 위해 K-means 알고리즘의 초기 코드북을 결정하는 많은 방법들이 제시되었는데, 그 중 splitting 방법이 다른 방법들보다 더 좋은 초기 코드북을 생성하는 것으로 알려져 있다.

K-means 알고리즘과 거의 동일하지만 각 반복과정에서 새로운 코드벡터를 구하는 방법만이 다른 알고리즘을 Jancey가 제안했는데[3], 이 방법은 현재벡터와 새로운 군집의 중심점과 일직선상에 있는 반대편의 점, 즉 거리의 가중치(δ)가 2.0인 점을 새로운 코드벡터로 사용하지만 이 점이 수렴영역의 경계선에 놓여 임의의 데이터에 대하여 수렴이 되지 않는 경우가 있을 수 있다. 이러한 문제를 보완한 것이 D. Lee가 제안한 개선된 K-means 알고리즘이다[4]. D. Lee의 방법은 Jancey가 제안한 방법에서 현재벡터와 새로운 군집의 중심점과 일직선상에 있는 거리의 가중치가 2.0인 점 대신 거리의 가중치가 1.8인 점을 새로운 코드벡터로 사용하는 것으로 기존의 K-means 알고리즘보다 더 좋은 성능을 보인다. 그러나 코드벡터 생성 시 초기 반복학습상태에서는 새로운 코드벡터들이 거리의 가중치에 의해 많은 영향을 받으므로 모든 반복학습과정 동안 가중치를 고정하는 것은 적절하지 않다[5]-[6].

또한, Fuzzy c-means 알고리즘은 기존의 K-means 알고리즘에 비해 보다 우수한 코드북을 설계할 수 있지만[7]-[8], 학습시간이 월등히 길기 때문에 코드북 설계에 주로 K-means 알고리즘을 사용한다. 따라서 본 논문에서는 Fuzzy c-means 알고리즘의 반복 학습시간을 개선하기 위해 퍼지모드의 소속함수에서 퍼지화의 정도를 결

정하는 퍼지화 변수 β를 1에 근접시키고, 가중치를 각각 1.1에서 2.0까지 0.1의 단위로 변화시킨 후 반복 학습시간을 비교하며, 이 알고리즘으로 설계되는 코드북은 그 성능에는 큰 영향을 미치지 않고 반복 학습시간을 줄일 수 있다.

II. K-means 알고리즘

K-means 알고리즘은 최소거리 조건과 중심조건이라는 두 필요조건을 만족하여야 한다. 최소거리 조건은 주어진 학습벡터와 코드벡터 사이의 Euclidean 거리가 최소일 때 학습벡터가 코드벡터에 대응하는 분할에 소속된다는 것을 말하며, 중심조건은 분할된 영역 안에서 학습벡터의 중심이 새로운 코드벡터가 된다는 것이다. K-means 알고리즘은 두 조건을 반복적으로 만족시키면서 평균거리 오차가 최소가 되는 코드북을 연속적으로 생성하는 것이다.

먼저 m차원 영역에서 코드벡터 집합 Y를 크기가 k인 코드북이라 하면 $Y=\{ \boldsymbol{y}_1, \boldsymbol{y}_2, \cdots, \boldsymbol{y}_k \}$이고, M 크기의 학습벡터 집합을 X라 하면 $X=\{ \boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_M \}$이 된다. 코드북 설계는 M개의 학습벡터를 k군집에 할당함으로써 설계된다.

K-means 알고리즘은 최소거리 조건에 근거하여 각 학습벡터를 어떤 한 군집에 할당하는데

$$d(\boldsymbol{x}_i, \boldsymbol{y}_j) = d_{\min}(\boldsymbol{x}_i) = \min_{\boldsymbol{y}_j \in Y} d(\boldsymbol{x}_i, \boldsymbol{y}_j) \quad (1)$$

이면, 학습벡터 $\boldsymbol{x}_i$는 j번째 군집에 할당된다. 여기서, $d(\boldsymbol{x}_i, \boldsymbol{y}_j)$는 $d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \| \boldsymbol{x}_i - \boldsymbol{y}_j \|^2$로 정의되며, 학습벡터 $\boldsymbol{x}_i$와 코드벡터 $\boldsymbol{y}_j$와의 Euclidean 거리의 제곱이다.

최소거리 조건은 식 (2)와 같이 정의되는 crisp 선택함수, 또는 소속함수로 표현될 수 있으며,

$$u_j(\boldsymbol{x}_i) = \begin{cases} 1 & d(\boldsymbol{x}_i, \boldsymbol{y}_j) = d_{\min}(\boldsymbol{x}_i) \text{ 일때} \\ 0 & \text{위 조건이 아닐때} \end{cases} \quad (2)$$

코드북의 코드벡터들은 다음과 같이 정의되는 왜곡측정을 최소화함으로써 얻어진다.

$$J_1 = \sum_{j=1}^{k} \sum_{i=1}^{M} u_j(\boldsymbol{x}_i) \| \boldsymbol{x}_i - \boldsymbol{y}_j \|^2 \quad (3)$$

여기서 k는 코드북의 크기를 나타내며, 주어진 소속함수에 대하여 $\boldsymbol{y}_j$에 관한 $J_1 = J_1(\boldsymbol{y}_j, j=1,2,\cdots,k)$의 최소화는 다음과 같다.

$$\boldsymbol{y}_j = \frac{\sum_{i=1}^{M} u_j(\boldsymbol{x}_i)\, \boldsymbol{x}_i}{\sum_{i=1}^{M} u_j(\boldsymbol{x}_i)} \quad \forall j = 1,2,\cdots,k \quad (4)$$

여기서 정의되는 코드벡터 $\boldsymbol{y}_j$는 유클리드 중심, 즉 j번째 군집에 할당된 모든 학습벡터의 중심이다. 위의 식에 새로운 코드벡터를 구하기 위해 현재 코드벡터와 새로운 군집의 중심점과 일직선상에 있는 거리의 가중치를 적용하면 다음과 같다.

$$\boldsymbol{y}_j^{n+1} = \boldsymbol{y}_j^{n} + \delta(\boldsymbol{c}_j^{n+1} - \boldsymbol{y}_j^{n}) \quad (5)$$

여기서, $\boldsymbol{y}_j^{n}$은 n번 반복시 j번째 코드벡터, $\boldsymbol{y}_j^{n+1}$은 $n+1$번 반복시 j번째 코드벡터, $\boldsymbol{c}_j^{n+1}$은 $n+1$번 반복시 j번째 코드벡터에 대응되는 중심벡터이다. 식 (5)에서 δ=1인 경우 기존의 K-means 알고리즘을 나타내고, δ=2 인 경우는 Jancey의 방법, δ=1.8 인 경우는 D. Lee의 방법으로 위의 방법들 중에서 가장 좋은 성능의 코드북을 설계할 수 있다.

III. 학습시간을 개선한 Fuzzy c-means 알고리즘

c-means 알고리즘은 최소거리 조건에 근거하여 각 학습벡터를 단일 군집에 할당된다는 의미에서 crisp 결정과정을 사용한다. 이 경우에 식 (2)와 같이 소속함수는 crisp 결정에 근거하여 0 아니면 1을 할당하므로, 각 학습벡터가 다른 군집에도 속할 수 있는 확률을 무시하는 경우가 발생한다.

이러한 문제를 해결하기 위해 각 군집을 퍼지 집합으로 생각하고, 0 과 1 사이의 소속값을 갖는 소속함수를 사용하여 각 학습벡터가 한 군집에 소속될 가능성을 계산하는 Fuzzy c-means 알고리즘을 사용한다. 이 소속값은 특정 벡터가 어떤 군집에 어느 정도 소속되는 것으로 간주할 것인가를 나타낸다. 학습벡터의 퍼지 분할은 k개의 군집 안에서 각 학습벡터가 소속되는 정도를 나타낸다. Fuzzy c-means 알고리즘의 유도는 다음과 같은 목적함수의 강제적인 최소화에 기초한다.

$$J_\beta = \sum_{j=1}^{k} \sum_{i=1}^{M} u_j(\boldsymbol{x}_i)^\beta \| \boldsymbol{x}_i - \boldsymbol{y}_j \|^2 \quad (6)$$

여기서 $1 < \beta < \infty$ 이다.

주어진 코드벡터에 대하여 $u_j(\boldsymbol{x}_i) \in [0,1] \ \forall i, j$, $0 < \sum_{i=1}^{M} u_j(\boldsymbol{x}_i) < M$ 과

$$\sum_{j=1}^{k} u_j(\boldsymbol{x}_i) = 1 \quad \forall i = 1,2,\cdots,M \quad (7)$$

의 제한 하에서 $J_\beta = J_\beta(u_j, J=1,2,\cdots,k)$ 을 최소화할 때 다음과 같은 소속함수를 얻는다.

$$u_j(\boldsymbol{x}_i) = \frac{1}{\sum_{l=1}^{k}\left(\frac{d(\boldsymbol{x}_i, \boldsymbol{y}_j)}{d(\boldsymbol{x}_i, \boldsymbol{y}_l)}\right)^{\frac{1}{\beta-1}}} \quad (8)$$

여기서 $d(\boldsymbol{x}_i, \boldsymbol{y}_j) = \| \boldsymbol{x}_i - \boldsymbol{y}_j \|^2$ 이다.

주어진 일련의 소속함수에 대하여 코드벡터는 다음과 같이 $J_\beta = J_\beta(\boldsymbol{y}_j, j=1,2,\cdots,k)$ 을 최소화함으로써 구할 수 있다.

$$\boldsymbol{y}_j = \frac{\sum_{i=1}^{M} u_j(\boldsymbol{x}_i)^\beta \boldsymbol{x}_i}{\sum_{i=1}^{M} u_j(\boldsymbol{x}_i)^\beta} \quad \forall j = 1,2,\cdots,k \quad (9)$$

이 알고리즘으로 생성된 군집의 퍼지화는 1보다 큰 변수 β로 제어되며, 이 변수 β가 1에 접근함에 따라 학습벡터의 공간의 분할이 거의 crisp 결정 과정이 된다. 이 변수를 증가시키는 것은 가장 퍼지한 상태로 만들어서 소속 정도를 낮추는 결과를 낳는다.

군집화 과정은 모든 학습벡터의 퍼지 할당을 갖고 시작하며, $\boldsymbol{\Upsilon}_i^n$ 을 n번째 반복학습 동안 학습벡터 $\boldsymbol{x}_i \in X$를 중심으로 하는 초공간(hypersphere)에 속하는 코드벡터의 집합이라 하면, 학습벡터 $\boldsymbol{x}_i$는 중심이 집합 $\boldsymbol{\Upsilon}_i^n$에 속하는 군집에만 할당될 수 있다. 군집화 과정이 시작될 때 각 학습벡터 $\boldsymbol{x}_i$는 모든 군집에 할당될 수 있으므로, 대응하는 초공간은 모든 군집을 포함하며, $\boldsymbol{\Upsilon}_i^n = Y$ 이다. $\boldsymbol{\Upsilon}_i^n$이 단일 코드벡터를 갖는다면, 학습벡터 $\boldsymbol{x}_i$는 퍼지모드에서 crisp 모드로 전환된다. $\boldsymbol{\Upsilon}_i^n$이 두 개 이상의 코드벡터를 갖는다면, 소속함수 $u_j(\boldsymbol{x}_i)$, $(j=1,2,\cdots,k)$는 0과 1의 사이값을 취한다. $\boldsymbol{y}_j \notin \boldsymbol{\Upsilon}_i^n$이면, $u_j(\boldsymbol{x}_i) = 0$ 이고, $\boldsymbol{y}_j \in \boldsymbol{\Upsilon}_i^n$이면, $\boldsymbol{x}_i$와 $\boldsymbol{y}_j \in \boldsymbol{\Upsilon}_i^n$간의 거리에 의존한다. 따라서 소속함수 $u_j(\boldsymbol{x}_i)$은 거리 $d(\boldsymbol{x}_i, \boldsymbol{y}_j)$이 0에 접근함에 따라 1에 접근하고, $d(\boldsymbol{x}_i, \boldsymbol{y}_j)$이 $d_{\max}(\boldsymbol{x}_i)$에 접근함에 따라 0에 접근한다. 여기서 $d_{\max}(\boldsymbol{x}_i)$는 학습벡터 $\boldsymbol{x}_i$와 코드벡터 $\boldsymbol{y}_j \in \boldsymbol{\Upsilon}_i^n$간의 최대거리를 나타낸다. 군집화가 진행될 때 학습벡터의 일부는 퍼지모드에서 crisp 모드로 전환되고, 모든 학습벡터가 퍼지모드에서 crisp 모드로 전환된 후에는 crisp K-means 알고리즘이 된다.

또한, 소속함수 $u_j(\boldsymbol{x}_i)$, $(j=1,2,\cdots,k)$의 평가는 n번째 반복학습 동안 집합 $\boldsymbol{\Upsilon}_i^n$에 속하는 군집의 중심의 수에 의존한다. 따라서 제안된 알고리즘에서는 학습시간을 감소하기 위해 퍼지화 변수 β를 1에 근접시키고, Fuzzy c-means 알고리즘에서 코드벡터 생성 시 거리의 가중치를 변화시켜 기존의 K-means 알고리즘, D. Lee의 방법과 그 성능을 비교한다.

Ⅳ. 실험 결과

본 실험에서는 제안한 알고리즘과 기존 알고리즘들을 비교하기 위해 256 그레이 레벨을 갖는 512×512 영상 20개를 이용하여 4×4 블록 단위로 블록킹한 후 이를 학습벡터로 사용하고, splitting 방법을 사용하여 얻은 크기가 256이고, 코드벡터가 16차원인 코드북을 사용한다. 입력 영상은 LENA 영상을 이용하고, 원 영상과 복원된 영상을 비교 평가하기 위한 PSNR(peak signal to noise ratio)은 다음과 같다.

$$PSNR = 20\log_{10}\left(\frac{255}{\sqrt{\frac{1}{512^2}\sum_{i=1}^{512}\sum_{i=1}^{512}(f_{ij} - g_{ij})^2}}\right) \quad (10)$$

여기서 f_{ij}는 원 영상의 화소 값이고, g_{ij}는 복원된 영상의 화소 값이다.

본 실험에서는 Fuzzy c-means 알고리즘을 사용하여 학습벡터가 군집에 할당될 때 crisp 모드가 아닌 퍼지모드의 소속함수로써 각 학습벡터가 한 군집에 소속될 가능성을 계산한다. 또한 코드북 생성시 새로운 코드벡터들은 거리의 가중치에 영향을 받으므로 가중치를 일정 간격으로 가변한다.

먼저 퍼지모드의 소속함수에서 퍼지화의 정도를 결정하는 퍼지화 변수 β를 1.1로 고정시키고, 가중치의 값이 2.0을 초과하는 경우 코드북의 성능을 저하시키므로 가중치를 1.1에서 2.0까지 0.1의 단위로 변화시켜 가며, 반복학습회수에 따른 PSNR을 구하고 기존의 K-means 알고리즘, D. Lee의 방법과 제안된 알고리즘을 비교하였다. 또한, 기존의 알고리즘과 제안된 알고리즘의 학습시간의 개선 정도를 비교하기 위해 반복 학습회수에 따른 PSNR을 분석하여, 동일한 PSNR값에서 반복회수가 줄어든 경우 학습시간이 개선된 것으로 간주한다.

그 결과 그림 1의 가중치의 변화에 따른 PSNR의 비교에서와 같이 가중치를 1.7-2.0으로 하였을 때, 그리고 퍼지화 변수 β를 1.1로 하였을 때 코드북의 성능이 향상되었으며, 그림 2의 반복 학습회수에 따른 PSNR의 비교

와 같이 가중치를 1.8-2.0으로 하였을 때 코드북의 성능에는 큰 영향을 미치지 않고 반복 학습시간을 5-6회 정도 줄일 수 있었다.

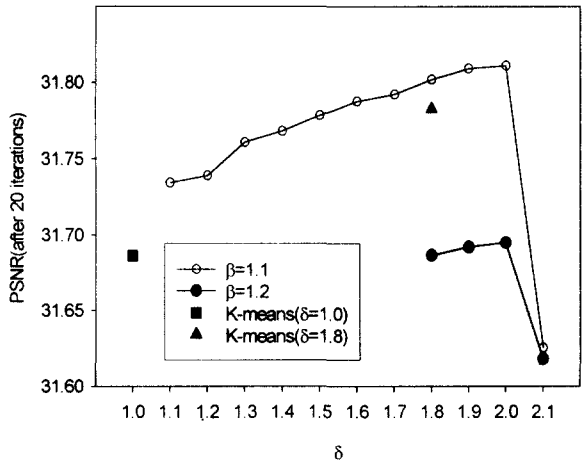

그림 1. 가중치의 변화에 따른 PSNR의 비교

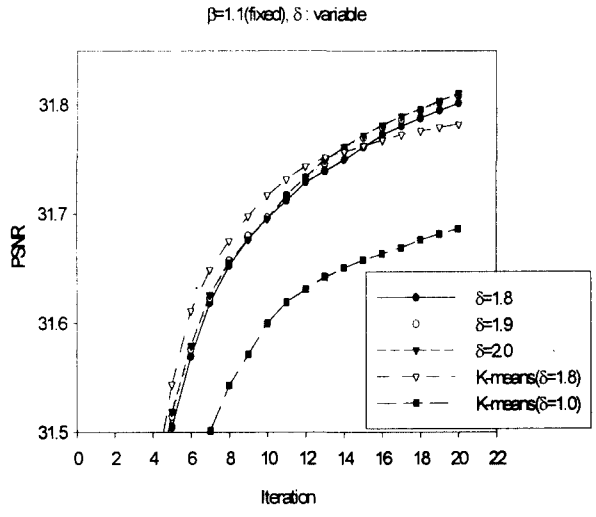

그림 2. 반복 학습회수에 따른 PSNR의 비교

V. 결 론

Fuzzy c-means 알고리즘은 기존의 K-means 알고리즘에 비해 보다 우수한 코드북을 설계할 수 있지만, 학습시간이 월등히 길다는 단점이 있다. 따라서 반복 학습시간의 단축을 위해 퍼지화 변수를 1로 근접시켜 학습이 반복될수록 학습벡터의 공간 분할이 거의 crisp 결정 과정이 되도록 하였으며, 코드벡터 생성 시 새로운 코드벡터들이 거리의 가중치에 의해 많은 영향을 받으므로 가중치의 값을 가변하였다. 그 결과 코드북의 성능에는 큰 영향을 미치지 않고 반복 학습시간을 5-6회 정도 줄일 수 있었으며, Fuzzy c-means 알고리즘에서도 가중치의 변화가 코드북의 성능에 영향을 미침을 알 수 있었다. 따라서 코드벡터 생성 시 새로운 코드벡터들이 거리의 가중치에 의해 많은 영향을 받으므로, 추후 모든 반복학습과정 동안 가중치를 가변하여 가중치의 영향을 분석하고, 실제 시스템에서 새로운 입력 패턴에 대한 코드북의 성능과 학습시간의 개선 여부에 대한 적용이 필요할 것이다.

참 고 문 헌

[1] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," IEEE Trans. Commun., vol. 28, pp. 84-95, Jan. 1980.

[2] A. Gersho and R. M. Gray, Vector Quantization and Signal Compression, KAP, 1992.

[3] M. R. Anderberg, Cluster Analysis for Application, New York: Academic, 1973.

[4] D. Lee, S. Baek, and K. Sung, "Modified K-means algorithm for vector quantizer design," IEEE Signal Processing Letters, vol. 4, pp. 2-4, Jan. 1997.

[5] 김형철, 조제황, "수정된 K-means 알고리즘," 한국음향학회 학술대회 논문집, Vol. 18, No. 2(s), 1999.

[6] K. K. Paliwal and V. Ramasubramanian, "Comments on Modified K-means algorithm for vector quantizer design," IEEE Trans. Image Processing, vol. 9, no. 11, pp. 1964-1967, Nov. 2000.

[7] N. B. Karayiannis and P.-I Pai, "Fuzzy vector quantization algorithms and their application in image compression," IEEE Trans. Image Processing, vol. 4, no. 9, pp. 1193-1201, Sep. 1995.

[8] M. Friedman and A. Kandel, Introduction to Pattern Recognition, World Scientific Publishing, 1999.