논문 2010-47CI-1-10

이미지 데이터베이스에서 매개변수를 필요로 하지 않는 클러스터링 및 아웃라이어 검출 방법

(A Parameter-Free Approach for Clustering and Outlier Detection in Image Databases)

오 현 교*, 윤 석 호*, 김 상 욱**

(Hyun-Kyo Oh, Seok-Ho Yoon, and Sang-Wook Kim)

요 약

이미지 데이터가 증가함에 따라 효율적인 검색을 위해서 이미지 데이터를 구조화해야 할 필요성이 증가하고 있다. 이미지 데이터를 구조화하기 위한 대표적인 방법으로는 클러스터링이 있다. 그러나 기존 클러스터링 방법들은 클러스터링을 수행하기 전에 매개변수로서 클러스터의 개수를 사용자로부터 제공 받아야 되는 어려움이 있다. 본 논문에서는 클러스터의 개수를 사용자에게 제공 받지 않고 이미지 데이터를 클러스터링 하는 방안에 대해서 논의 한다. 제안하는 방안은 객체들 간의 상호 연관관계를 이용하여 매개변수 없이 데이터의 감추어진 구조나 패턴을 찾아내는 방법인 Cross-Association을 기반으로 한다. 이미지 데이터 클러스터링에 Cross-Association을 적용하기 위해서는 먼저 이미지 데이터를 그래프로 변환해야 한다. 그런 후에 생성된 그래프를 Cross-Association에 적용시키고 그 결과를 클러스터링 관점에서 해석한다. 본 논문에서는 또한 Cross-Association을 기반으로 계층적 클러스터링 하는 방법과 아웃라이어 검출 방법을 제안한다. 실험을 통해서 제안하는 방법의 우수성을 규명하고 이미지 데이터를 클러스터링 하는데 적절한 k-최근접 이웃검색에서의 k값과 더 나은 그래프 생성 방법이 무엇인지를 제시한다.

Abstract

As the volume of image data increases dramatically, its good organization of image data is crucial for efficient image retrieval. Clustering is a typical way of organizing image data. However, traditional clustering methods have a difficulty of requiring a user to provide the number of clusters as a parameter before clustering. In this paper, we discuss an approach for clustering image data that does not require the parameter. Basically, the proposed approach is based on Cross-Association that finds a structure or patterns hidden in data using the relationship between individual objects. In order to apply Cross-Association to clustering of image data, we convert the image data into a graph first. Then, we perform Cross-Association on the graph thus obtained and interpret the results in the clustering perspective. We also propose the method of hierarchical clustering and the method of outlier detection based on Cross-Association. By performing a series of experiments, we verify the effectiveness of the proposed approach. Finally, we discuss the finding of a good value of k used in k-nearest neighbor search and also compare the clustering results with symmetric and asymmetric ways used in building a graph.

Keywords: 이미지 데이터 클러스터링, 클러스터링, 아웃라이어 검출, Cross-Association, parameter-free

I. 서 론

최근 들어, 디지털 카메라, 인터넷 등 디지털 이미지를 생성하거나 이용할 수 있는 매체가 대중화 되면서 이미지 데이터가 기하급수적으로 증가하고 있다. 이미지 데이터가 증가하면서 사용자들이 원하는 이미지 데이터를 효율적으로 검색하기 위해 이미지 데이터들을 구조화 하는 연구가 진행되고 있다[1~2].

* 정회원, ** 평생회원(교신저자), 한양대학교 전자컴퓨터통신공학과
(Department of Electronics and Computer Engineering, Hanyang University)

※ 본 연구는 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원(R01-2008-000-20872-0) 및 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업(IITA-2009-C1090-0902-0040)의 지원을 받았습니다.
접수일자: 2009년12월16일, 수정완료일: 2010년1월11일

이미지 데이터를 구조화하기 위해 주로 사용하는 방법으로는 클러스터링이 있다. 클러스터링은 유사한 객체들을 같은 그룹에 포함시키고, 유사하지 않은 객체들을 다른 그룹에 포함시키는 방법으로, 대표적인 알고리즘으로는 K-means[3], CURE[4], BIRCH[5], Chameleon[6] 등이 있다.

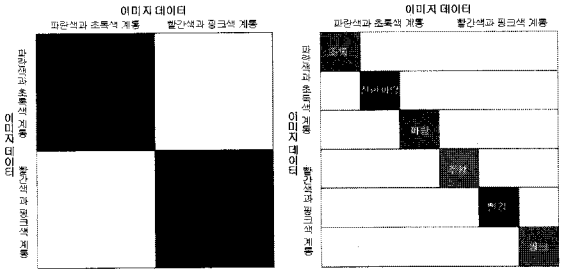
클러스터링 알고리즘은 입력 값으로 주어지는 데이터의 유형과 클러스터링의 목표 및 응용에 따라 분류되는데, 대표적으로 분할 방법(partitioning methods), 계층적 방법(hierarchical methods), 밀도-기반 방법(density-based methods)등으로 분류된다[3].

기존의 클러스터링 방법들은 사용자에 의해서 주어진 클러스터의 개수를 매개변수로 입력 받아서 클러스터링 한다. 그러나 사용자가 적절한 클러스터의 개수를 클러스터링 수행 전에 결정하는 것은 매우 어려운 문제이다[3]. 따라서 본 논문에서는 클러스터의 개수를 사용자에게 매개변수로 입력받지 않고 데이터를 클러스터링하는 방안에 대해 논의하고자 한다.

또한, 데이터를 계층적으로 분해하는 방안을 논의하고자 한다. 이 방안도 사용자로부터 클러스터의 개수를 매개변수로 입력받지 않고 진행된다. 특히, 계층적 클러스터링 방법 중 하나로서 하향식(top-down) 접근방법으로도 불리는 분할적(divisive) 접근방법을 통해 데이터를 분해한다. 계층적 클러스터링은 클러스터의 각 객체가 하나의 클러스터가 되거나 주어진 종료 조건을 만족할 때까지 재귀적으로 분할하는 방법이다[3]. 이 방법을 이용하면 주어진 데이터가 다양한 레벨로 분해되기 때문에 사용자에게 데이터의 전체적인 구조와 패턴뿐만 아니라, 데이터의 세부적인 구조와 패턴과 같은 유용한 정보를 제공할 수 있다.

클러스터링 방법은 아웃라이어 검출(outlier detection)에도 사용된다. 아웃라이어는 데이터의 일반적인 모형이나 행동에 대응하지 못하는 데이터 객체를 의미한다[3]. 사용자는 아웃라이어 검출을 통해 선정된 객체를 제거함으로써 해당 클러스터의 질을 높일 수 있게 된다. 따라서 본 논문에서는 아웃라이어 객체를 찾는 효과적인 방안을 제안한다.

본 논문에서는 클러스터의 개수를 매개변수로 입력받지 않고 클러스터링을 수행하기 위해 Cross-Association(이하 CA)[7]을 이용한다. CA는 객체간의 상호 연관관계를 이용하여 매개변수 없이 데이터의 감추어진 구조나 패턴을 찾아내는 방법이다. 제안하는 방안



(a) 첫 번째 레벨 (b) 두 번째 레벨

(a) First level (b) Second level

그림 1. 계층적 클러스터링 결과

Fig. 1. Results of hierarchical clustering.

은 CA를 수행하기 위해서 이미지 데이터를 그래프 구조로 변환하고 변환된 데이터에 CA를 적용한 후에 그 결과를 클러스터링 관점에서 해석한다.

또한, 데이터에 CA를 적용한 후 클러스터링 관점에서 해석한 결과로 나온 각각의 클러스터에 재귀적으로 CA를 적용하는 계층적 클러스터링을 수행한다. 이는 계층적 클러스터링의 방법 중 하나인 분할적 접근 방법을 이용한다. 이를 통해 사용자는 클러스터 안에 감추어진 하위 클러스터들을 발견 할 수 있다. 예를 들면, 그림 1은 이미지 데이터의 색상 유사도를 이용한 계층적 클러스터링 결과를 나타낸다. 그림 1(a)는 첫 번째 레벨의 클러스터링 결과를 나타낸다. 주어진 이미지 데이터는 첫 번째 레벨에서 파란색과 초록색 계통 그리고 빨간색과 핑크색 계통의 두 개의 클러스터로 분해된다. 첫 번째 레벨에서 발견된 두 개의 클러스터에 재귀적으로 CA를 적용하면 그림 1(b)와 같이 클러스터 안에 감추어진 하위 클러스터들을 발견 할 수 있다. 첫 번째 레벨에서 발견된 파란색과 초록색 계통의 클러스터는 두 번째 레벨에서 초록, 진한 파랑, 파랑 색깔의 하위 클러스터로 분해된다. 또한 첫 번째 레벨에서 발견된 또 다른 클러스터인 빨간색과 핑크색 계통의 이미지 객체들은 두 번째 레벨에서 주황, 빨강, 핑크 색깔의 하위 클러스터로 분해된다. 이와 같이 사용자는 계층적 클러스터링을 통해 다양한 레벨에서 주어진 데이터의 클러스터링 결과를 확인 할 수 있다.

CA를 클러스터링 관점으로 재해석하였기 때문에 이에 따른 새로운 아웃라이어 검출 알고리즘이 필요하다. 본 논문에서는 제안하는 방안을 통해 계층적 클러스터링 결과로 도출된 각각의 클러스터에 포함된 모든 객체들을 대상으로 아웃라이어 등급을 정할 수 있다. 제안

하는 방안은 클러스터 안에 객체들을 순차적으로 제거하면서 클러스터의 정보량의 변화를 관찰하고 그 값에 따라서 객체의 아웃라이어 등급을 정하는 방식이다. 이 중 사용자는 등급이 가장 높은 n개의 객체를 아웃라이어로 검출할 수 있다.

제안하는 방안을 실제 이미지 데이터에 적용하여 여러 가지 의미 있는 결과를 확인하였다. CA를 적용한 색상 유사도 기반의 이미지 데이터 클러스터의 정확도를 5명의 평가자들에게 의뢰한 결과 최대 90.9%의 정확도를 보였고, 또한 계층적 클러스터링을 수행한 결과와 아웃라이어 검출 결과가 타당함을 실험을 통해 보였다.

본 논문의 공헌을 요약하면 다음과 같다.

- 상호 연관관계를 탐사하기 위하여 고안된 CA를 클러스터링에 활용 할 수 있는 방안을 고안
  - 매개변수가 필요 없는 이미지 클러스터링 방법을 제안
- CA를 활용하여 계층적으로 클러스터링 할 수 있는 방안을 고안
  - 매개변수가 필요 없는 계층적 이미지 클러스터링 방법을 제안
- 좋은 이미지 클러스터링 결과를 얻기 위한 그래프 구성 방안을 제시
- 새로운 아웃라이어 검출 알고리즘을 제안
- 실험을 통해서 우수성을 검증

본 논문의 구성은 다음과 같다. 제 Ⅱ장에서는 본 연구와 관련된 기존의 연구들을 기술한다. 제 Ⅲ장에서는 CA를 이용한 이미지 클러스터링 방안과 계층적 클러스터링으로의 확장 방안 그리고 아웃라이어 검출 방안을 다룬다. 제 Ⅳ장에서는 클러스터링 결과에 대한 정확도 평가와 아웃라이어 검출 결과를 비롯한 각종 실험 결과를 보여준다. 끝으로, 제 Ⅴ장에서는 결론을 제시한다.

Ⅱ. 관련 연구

본 장에서는 객체간의 상호 연관관계를 이용하여 매개변수 없이 데이터의 감추어진 구조나 패턴을 찾아내는 방법인 CA와 그 응용들에 대해 소개한다. 1절에서는 CA의 기본원리에 대해 설명하고, 2절에서는 CA를 응용한 커뮤니티 발견(community discovery)에 대해 설명한다. 마지막으로 3절에서는 아웃라이어를 검출하는 방법에 대해 설명한다.

1. Cross-Association

CA는 객체들 간의 상호 연관 관계가 표현된 이진 행렬의 행과 열의 순서를 MDL(Minimum Description Length) 원리[8]를 적용해서 이진 행렬을 표현하는 정보량이 최소화되도록 바꾼다. 결과적으로 연관 관계 패턴이 유사한 객체들이 이진 행렬에서 서로 인접하게 배열된다. 이렇게 인접해 있는 객체들이 행렬에서 차지하고 있는 지역을 Cross-associates라고 한다. 이는 이진 행렬의 숨겨진 구조를 나타내며, 상호 연관 관계에 있는 객체들이 그 지역 안에 모여 있음을 의미한다.

그림 2는 Cross-associates를 찾는 과정이다. 그림에서 검은색 부분은 객체간의 연관 관계가 있음을 나타내고 흰색 부분은 연관관계가 없음을 나타낸다. 그림 2(a)와 같은 기본 행렬에 CA를 적용하면 최종적으로 그림 2(e)와 같이 유사한 객체들이 모인 지역인 Cross-associates를 찾아낸다.

이미지 데이터를 CA를 통해 클러스터링 하기 위해서는 먼저 이미지 데이터를 이진 행렬로 변환하는 과정이 필요하다. 또한 CA를 통해 유사한 객체들의 집합인 Cross-associates를 찾을 수 있지만 동일한 객체가 서로 다른 Cross-associates에 포함될 수 있기 때문에 클러스터링의 클러스터와는 다른 의미를 가진다. 따라서 본 논문에서는 이미지 데이터를 이진 행렬로 표현하는 방법에 대해서 논의하고, CA를 클러스터링 관점으로 해석하여 이미지 데이터를 매개 변수 없이 클러스터링 하고자 한다.

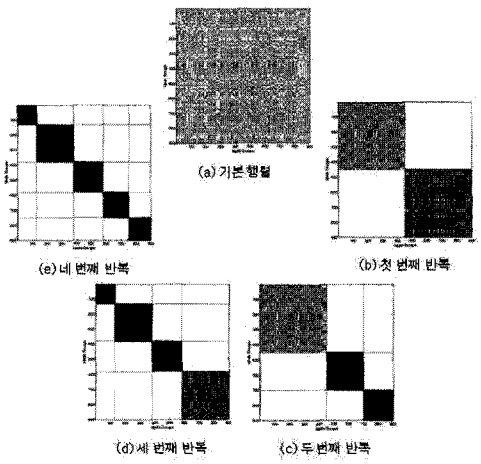


그림 2. Cross-associates를 찾는 과정[7]
Fig. 2. Finding of Cross-associates.

2. 커뮤니티 발견(Community Discovery)

참고문헌[9]에서는 CA를 통해 찾은 Cross-associates를 하나의 커뮤니티로 간주한다. 예를 들면, 저자와 컨퍼런스(conference)간의 관계를 이용해 구성된 그래프에 CA를 적용한 결과인 그림 3(a)에서 왼쪽 상단의 빨간색 지역은 컴퓨터 과학 분야 컨퍼런스에 논문을 제출한적 있는 컴퓨터 과학 분야 저자들의 커뮤니티를 의미한다. 그림 3(a)의 오른쪽 하단의 파란색 지역은 의학 분야 컨퍼런스에 논문을 제출한 적이 있는 의학 분야 저자들의 커뮤니티를 뜻한다. 또한 CA를 통해 찾은 모든 커뮤니티에 재귀적으로 CA를 적용하여 커뮤니티 안의 또 다른 의미 있는 커뮤니티를 계층적으로 찾는다. 그림 3(b)는 두 번째 레벨에서의 찾은 커뮤니티들의 결과이다. 그림 3(b)의 왼쪽 상단의 지역은 컴퓨터 과학 분야의 커뮤니티를 재귀적으로 CA를 적용한 결과를 나타낸다. 두 개의 세부적인 커뮤니티들이 발견되었는데, 발견한 커뮤니티들은 컴퓨터 분야의 컨퍼런스 중에서도 시스템 분야의 컨퍼런스에 논문을 제출한 저자들과 이론 분야에 컨퍼런스에 논문을 제출한 저자들의 커뮤니티이다. 또한 오른쪽 상단의 지역을 통해 의학 분야의 컨퍼런스에 논문을 제출한 컴퓨터 분야의 저자들의 커뮤니티도 발견함을 보여준다. 마지막으로 오른쪽 아래 지역을 통해 의학 분야의 저자들이 의학 분야의 컨퍼런스 중에서도 병리학 분야의 논문을 제출한 저자들과 외과 분야에 컨퍼런스에 논문을 제출한 저자들로 구분되어지는 것을 확인 할 수 있다.

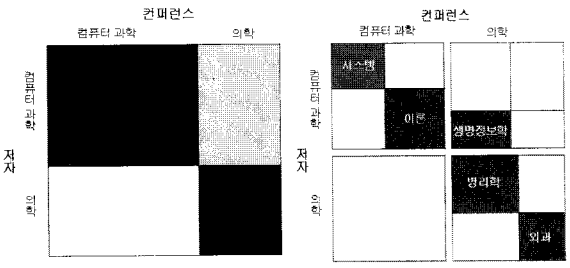
이와 같이 참고문헌[9]에서는 CA를 이용하여 계층적으로 커뮤니티를 찾아내는 유용한 방법을 제안한다. 그러나 2.1절에서 언급한 것처럼 동일한 객체가 서로 다른 커뮤니티에 포함되어 있기 때문에 이는 클러스터링의 클러스터와는 다른 의미를 지닌다. 예를 들면, 그림 3(b)에서 컴퓨터 과학 분야 중 이론 분야 컨퍼런스에 논문을 제출한 저자들과 의학 분야 중 생명정보학 분야 컨퍼런스에 논문을 제출한 저자들은 중복되어서 다른 커뮤니티에 존재함을 확인 할 수 있다. 즉, 각 객체는 하나의 클러스터에 속해야 한다는 클러스터링 방법과는 다른 의미를 갖는 커뮤니티를 찾아낸 것이다. 참고 문헌[9]의 방법은 그림 3의 경우와 같이 이진 행렬을 구성하는 x축과 y축의 객체의 성격이 다르고, 서로 다른 커뮤니티 안에 객체들이 중복되어도 되는 경우에만 사용가능하다. 즉, 어떤 객체가 다른 객체를 포함하는 관계를 지닌 데이터에 대해서만 의미 있는 결과를 보이는 방법이다.

따라서 본 논문에서는 그림 1의 경우와 같이 이진 행렬을 구성하는 x축과 y축의 객체가 같고, 서로 다른 클러스터 안에 객체들이 중복되어 있으면 안 되는 경우에 적합한 새로운 계층적 클러스터링 방법을 제안하고자 한다. 제안하는 방법을 통해 사용자는 다양한 수준에서 클러스터들의 변화를 파악 할 수 있게 된다.

3. 아웃라이어 검출(Outlier Detection)

참고 문헌[10]에서는 CA를 적용한 후에 데이터의 구조를 이용하여 아웃라이어를 검출 하는 방법을 제안한다. CA를 적용한 데이터는 유사한 연관 관계 패턴을 가진 객체들이 모여 있는 여러 개의 Cross-associate로 나누어진다. 객체간의 연관 관계는 이진 행렬상의 간선(edge)으로 표현되며. CA 적용 후 데이터는 다수의 간선이 존재하는 지역과 그렇지 않은 지역으로 구분할 수 있다. 특히, 소수의 간선만이 존재하는 지역은 연관성이 거의 없는 객체들이 모인 지역을 의미하며, 이때의 간선을 아웃라이어 간선(outlier edge)이라 한다. 이 간선은 이진 행렬을 표현하는 정보량인 인코딩 비용(encoding cost)을 증가시킨다.

인코딩 비용은 클로드 섀년이 제안한 정보 엔트로피(information entropy)[10]를 이용해서 구한다. 이 방법은 엔트로피의 개념을 빌려 정보의 양을 설명한 것으로써 Cross-associate의 엔트로피가 높은 경우 인코딩 비용이 높게 측정되고 반대의 경우 인코딩 비용이 낮게 측정된다. 예를 들면, Cross-associate를 과정을 나타내는 그림 2(a) 기본행렬은 CA를 적용하지 않는 상태 즉, 엔트로피가 높은 상태로 인코딩 비용이 높게 측정된다. 엔트로피가 높은 것은 검은색 부분과 흰 부분이 혼재되



(a) 첫 번째 레벨 (a) First level (b) 두 번째 레벨 (b) Second level

그림 3. 커뮤니티 발견의 예
Fig. 3. Community Discovery.

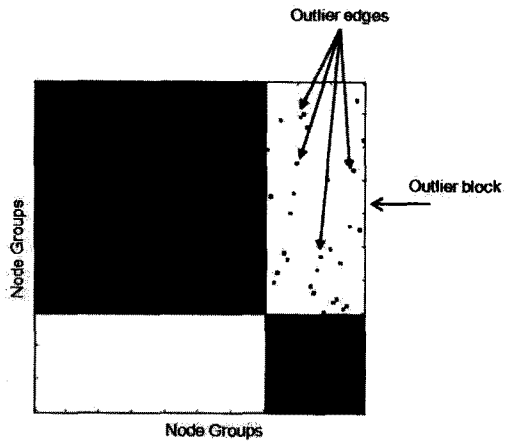


그림 4. 아웃라이어 간선과 아웃라이어 블록[8]
Fig. 4. Outlier edges and an outlier block[8].

어 있음을 통해 알 수 있다. 그림 2(a) 기본행렬은 그림 2(a)-(e) 단계를 거쳐 낮은 엔트로피를 갖는 즉, 인코딩 비용이 낮게 측정되는 구조로 분해된다. 이 과정을 통해 각 지역은 검은색 또는 흰색 부분이 대부분 존재하게 되고 이는 엔트로피가 낮은 상태를 의미하며 따라서 인코딩 비용이 낮게 측정된다.

이와 같은 정보 엔트로피 개념을 이용하여 참고 문헌[10]에서는 어떤 간선을 제거 했을 때 해당 지역의 인코딩 비용이 현저하게 감소하게 되는 경우, 그 간선을 아웃라이어 간선이라고 정의한다. 또한 이러한 아웃라이어 간선들이 존재하는 지역을 아웃라이어 블록(outlier block)이라고 정의한다. 그림 4의 오른쪽 상단의 지역은 아웃라이어 블록과 아웃라이어 간선을 나타낸다.

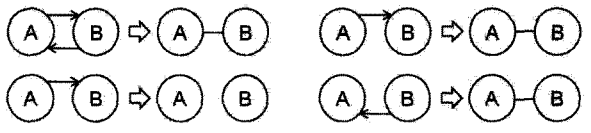
III. 제안하는 방법

본 장에서는 매개 변수 없이 이미지 데이터를 클러스터링 하는 방안과 아웃라이어 검출을 위한 새로운 알고리즘을 제안한다. 1절에서는 CA를 이용한 이미지 클러스터링에 대해서 설명하고, 2절에서는 계층적 클러스터링 방안에 대해서 설명한다. 마지막으로 3절에서는 아웃라이어 검출을 위한 새로운 알고리즘을 제시한다.

1. 이미지 클러스터링 방안

이미지 데이터를 CA에 적용시키기 위해서는 먼저 이미지 데이터를 이진 행렬로 변환해야 한다. 본 논문에서는 이미지 데이터 간의 유사 여부를 이용해서 그래프를 생성하고, 생성된 그래프를 이진 행렬로 다시 변환하고자 한다. 구체적인 방법으로 이미지 객체의 특성을 기반으로 기존의 유사도 계산 방안을 이용하여 객체들 간의 유사도를 계산하고 그래프 생성 방법 중에 하나인 k-최근접 이웃검색 방법[11]으로 그래프를 생성한다. 이렇게 생성된 그래프를 이진 행렬로 변환하면 CA에 적용 가능하게 된다.

이미지 데이터를 가지고 그래프를 생성할 때 고려해야 하는 두 가지 문제가 있다. 첫 번째는 k-최근접 이웃검색의 k값을 설정하는 문제이다. 그래프를 생성할 때 객체 간의 간선은 유사도 기반의 k-최근접 이웃검색을 통해 생성된다. 이때 k값에 따라서 그래프의 위상구조가 바뀌기 때문에 클러스터링의 결과가 달라질 수 있다. 따라서 본 논문에서 실험을 통해서 적절한 k값의 범위를 제시하고자 한다. 두 번째는 그래프를 생성할 때 대칭적인 방법을 이용 할 것이지 또는 비대칭적 방법을 이용할 것인지에 대한 문제이다. 그림 5는 그래프의 두 가지 생성기법을 나타낸다. 화살표는 k-최근접 이웃 검색을 통해 선택된 유사한 객체들을 나타내고, 직선은 생성된 그래프의 간선을 나타낸다. 그림 5(a)는 대칭적 방법을 이용해서 그래프를 생성하는 방법이다. 예를 들면, 객체 A가 객체 B의 유사 객체들 중에 포함되고 반대로 객체 B가 객체 A의 유사 객체들 중에 포함되면 객체 A와 객체 B 사이에 간선을 생성한다. 그림 5(b)는 비대칭적 방법을 이용해서 그래프를 생성하는 방법이다. 예를 들면, 객체 A가 객체 B의 유사 객체들 중에 포함되면 반대의 경우인 객체 B가 객체 A의 유사 객체들 중에 포함되는 지 여부와 상관없이 객체 A와 객체 B 사이에 간선을 생성한다. 대칭적 방법은 객체 간에 서로 유사할 경우에만 간선을 설정한다. 이는 한쪽만 유사하다고 여기는 경우를 노이즈로 간주하고 두 객체들 간에 연관이 없다고 판단하는 것이다. 어떠한 방법을 적용하여 그래프를 생성했을 때 CA를 통한 클러스터링의 결과가 좋은지 실험을 통해 알아보고자 한다.



(a) 대칭적 방법 (b) 비대칭적 방법
(a) Symmetric method (b) Asymmetric method

그림 5. 그래프 생성기법
Fig. 5. Methods of forming a graph.

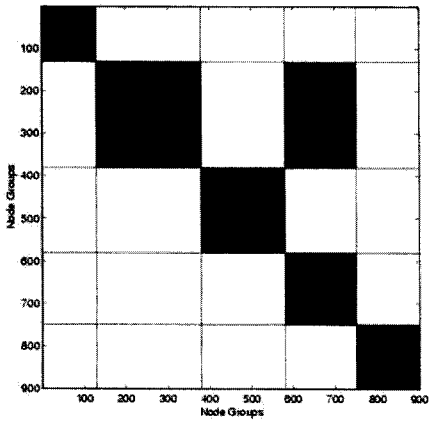


그림 6. CA의 결과[7]
Fig. 6. Results of CA[7].

본 논문에서는 CA를 클러스터링 관점에서 해석한다. 그림 6의 두 번째 행 그룹에는 두 개의 검은색 지역이 존재한다. 검은색 부분은 객체간의 연관 관계가 있음을 나타낸다. 그림 6을 통해 동일한 행 객체들이 서로 다른 두 개의 지역에 포함됨을 알 수 있다. CA는 이러한 지역을 찾고 분석하는데 초점을 맞추고 있다. 그러나 이러한 방식은 서로 다른 클러스터에는 같은 객체들을 포함 할 수 없는 클러스터링과는 다른 의미를 가진다. 따라서 본 논문에서는 이진 행렬의 한 축인 행들의 집합에만 초점을 맞춘다. CA의 결과를 연관 관계 패턴이 유사한 행 객체들이 서로 인접하게 배열되는 것으로 해석하면, 행 집합들 간에는 같은 객체들이 포함하지 않게 된다. 본 실험에서는 이러한 관점에서 CA를 클러스터링으로 해석하여 이미지 데이터를 클러스터링 진행한다.

2. 계층적 이미지 클러스터링 방안

변환된 이미지 데이터를 CA에 적용한 결과를 클러스터링 관점에서 해석하면 같은 객체들을 포함하지 않는 행 집합들을 발견할 수 있다. 이러한 행 집합들과 데이터 상의 존재하는 모든 열들과의 관계를 나타내는 각 지역을 하나의 클러스터로 간주한다.

본 논문에서는 클러스터링 결과로 도출된 각각의 클러스터에 대해 재귀적으로 CA를 적용하는 계층적 클러스터링을 수행하고자 한다. 이를 통해 사용자는 클러스터안의 숨겨진 클러스터의 구조를 발견할 수 있게 된다. 재귀적인 CA를 적용하는 클러스터링은 대상이 되는 클러스터의 정보량이 더 이상 감소하지 않을 때까지 진행된다. 이처럼 정보량이 감소하지 않을 때의 클러스터를 말단 클러스터(leaf cluster)라고 한다. 즉, 말단 클러스터는 더 이상 하위 클러스터를 갖지 않는 클러스터이다. 그림 7은 계층적 클러스터링의 결과를 보여준다. 주어진 이진 행렬 데이터는 첫 번째 레벨에서 두 개의 클러스터로 분해된다. 클러스터 2는 CA를 적용해도 정보량이 더 이상 감소하지 않는 말단 클러스터다. 그림 7의 클러스터 1은 하위 레벨로의 분해가 가능하고 두 번째 레벨에서 두 개의 클러스터로 분해된다. 결과적으로 주어진 이진 행렬은 총 4개의 말단 클러스터를 갖게 된다.

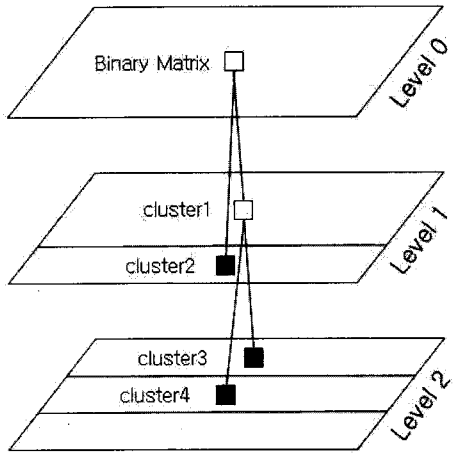


그림 7. 계층적 클러스터링 결과
Fig. 7. Results of hierarchical clustering.

제안하는 계층적 클러스터링 방법은 특정 지역에 존재하는 객체간의 연관관계만을 통해 커뮤니티의 구조를 찾는 2장 2절의 방법과는 달리 행렬상의 존재하는 모든 열 객체가 자신과 관계있는 행 객체의 클러스터링에 관여한다. 따라서 모든 열 객체들과의 연관 관계 패턴이 유사한 행 객체가 같은 클러스터로 모이게 되고 모든 행 객체는 중복되지 않은 상태로 각각의 클러스터에 포함되게 된다.

3. 아웃라이어 검출 방안

클러스터링 관점에서 아웃라이어 검출은 주어진 데이터 중 가장 이질적인 객체를 선택하는 방법이다. 따라서 아웃라이어 간선 검출 방법[10]은 클러스터링 관점에서 적합하지 않다. 본 논문에서는 클러스터링 관점에 부합하는 새로운 아웃라이어 검출 알고리즘을 제안한다. 제안하는 방안을 통해 클러스터의 존재하는 모든 객체에 아웃라이어 등급을 정할 수 있다. 등급이 가장 높은 객체는 클러스터에서 가장 이질적인 패턴을 가지

표 1. 용어 정의 1
Table 1. Notation 1.

용어	정의
Ci	i번째 클러스터
n_{ij}	i번째 클러스터의 j번째 객체
$I(C_i)$	i번째 클러스터의 정보량
CF	비용감소량(CostFall)

는 객체를 의미한다. 사용자는 등급이 가장 높은 n개의 객체를 아웃라이어로 선택 할 수 있다. 이와 같이 객체 자체가 클러스터에 미치는 영향력을 측정하여 등급을 매기는 아웃라이어 노드(outlier node) 검출 방안을 제안한다. 아웃라이어 노드를 검출하기 위해서 각 객체의 아웃라이어의 등급을 정해야 하는데 이를 위해 객체의 아웃라이어 등급을 구하는데 필요한 식을 제안한다.

$$CF = I(C_i) - I(C_i - n_{ij}) \quad (1)$$

(1)은 객체의 비용감소량(CostFall)을 구하는 공식이다. 표 1에서 C_i는 i번째 클러스터를 의미하고 n_{ij}는 i번째 클러스터에 존재하는 j번째 객체를 의미한다. $I(C_i)$는 i번째 클러스터의 정보량을 의미한다. 사용자는 (1)을 통해 객체의 비용감소량을 구할 수 있게 되는데, 이 때 해당 객체의 비용감소량이 크면 클수록 아웃라이어 등급이 높다고 간주한다.

본 논문에서는 (1)을 사용하여 아웃라이어 노드를 검출하기 위한 방안을 제안한다. 제안하는 방안은 아웃라이어의 대상이 되는 객체를 제거하기 전 클러스터의 정보량과 제거한 후 클러스터의 정보량을 측정한 다음 정보량의 차이가 가장 큰 객체가 등급이 가장 높은 아웃라이어가 되는 것이다. 그 이유는 만약 어떤 객체가 제거됨으로써 해당 클러스터의 정보량의 크게 감소한다면 해당 객체는 클러스터의 정보량을 증가시키는 존재임을 나타내기 때문이다. 본 논문에서는 클러스터안의 다른 객체들에 비해 비용 감소량이 큰 상위 n개의 객체들을 아웃라이어 객체로 간주한다. 아웃라이어 검출의 구체적인 절차는 그림 8과 같다.

```
Algorithm outlier detection
1: for 같은 레벨의 모든 클러스터에 대해
2:     for 해당 클러스터의 모든 객체에 대해
3:         do 비용감소량(CostFall) 측정
4: 비용감소량이 큰 상위 n개를 아웃라이어로 검출
5: if 해당 클러스터가 계층적 클러스터링이 되면
6:     then 하위 레벨의 모든 클러스터에 대해
       outlier detection 진행
```

그림 8. 아웃라이어 검출 알고리즘.
Fig. 8. Algorithm outlier detection.

본 논문에서는 이미지 클러스터를 대상으로 아웃라이어 검출 실험을 한다. 실험 결과를 통해 제시하는 알고리즘의 우수성을 증명한다.

Ⅳ. 실 험

1. 실험 환경

본 실험에서는 쉐이더 데이터를 이미지 클러스터링 대상으로 선정한다. 쉐이더 데이터는 형태를 제외한 색상, 질감, 무늬 등 표현된 객체를 의미한다[12]. 각 쉐이더 데이터에 대해 색상을 비롯한 몇 가지의 속성 값을 무작위(random)로 바꾸어 다양한 특성을 갖는 1,000개의 쉐이더 데이터를 생성한다. 본 실험에서는 k-최근접 이웃검색의 유사도 값으로 색상의 RGB 값을 이용한다. 그리고 유사도 측정 함수로는 Histogram quadratic distance[13]를 사용한다.

생성된 이미지 데이터를 다양한 k값에 대해 그리고 대칭적, 비대칭적 방법을 각각 적용하여 그래프로 생성한다. k-최근접 이웃검색의 k값은 20~200 사이로 20씩 증가하면서 설정한다. 어떤 방법으로 그래프를 생성해야 클러스터의 정확도가 높은지 알아보기 위해 생성된 모든 그래프에 CA를 적용하여 클러스터링을 한다. 그리고 실험 결과 높은 정확도를 보이는 그래프를 대상으로 계층적 클러스터링과 아웃라이어 검출을 한다.

2. 실험 방법

이미지 클러스터링을 통해 도출된 클러스터들의 결과가 타당한지를 판단하기 위해서 5명의 평가자들에게 클러스터의 정확도를 평가한다. 평가 방법은 평가자들

표 2. 용어 정의 2
Table 2. Notation 2.

용어	정의
C_i'	이미지 삭제 후 i번째 클러스터
D	전체 데이터(1000개의 이미지)
W_{ci}	$\frac{Count(C_i)}{Count(D)}$

에게 클러스터에 있는 이미지 객체들을 직접 확인하여 유사한 이미지들이 모여 있는지 판단하게 한다. 만약 해당 클러스터에 적합하지 않다고 생각되는 이미지가 있다면 그 이미지를 삭제하도록 한다.

이미지 클러스터의 정확도 =

$$w_{c1}\left(\frac{Count(C_1')}{Count(C_1)}\right)+w_{c2}\left(\frac{Count(C_2')}{Count(C_2)}\right)+\cdots\cdots \quad (2)$$

(2)는 이미지 클러스터의 정확도를 계산하는 공식이다. 표 1의 C_i는 이미지 삭제 전의 i번째 클러스터를 의미하고, 표 2의 C_i'는 평가자에 의해 이미지가 삭제된 후의 i번째 클러스터를 의미한다. D는 전체 이미지 데이터를 의미하고, W_{ci}는 i번째 클러스터안의 이미지 데이터 개수에 대한 가중치를 의미한다. 클러스터마다 이미지의 개수가 다르기 때문에 각 클러스터가 정확도에 미치는 영향이 다르다. 이미지 개수에 따라 클러스터에 미치는 영향력을 반영하기 위해 가중치를 사용한다. 5명의 평가자들의 평가가 끝나면 (1)을 이용하여 각각의 이미지 클러스터의 정확도를 구한 뒤, 평균값을 측정한다.

제안하는 계층적 클러스터링 결과가 타당한지를 판단하기 위해서 이미지 클러스터링 실험 결과 정확도가 높은 그래프를 대상으로 계층적 클러스터링을 수행한다. 그 결과 주어진 그래프가 하위 레벨에서 더욱 구체적으로 분해되는지 실험을 통해 시각적으로 보인다.

아웃라이어 검출 방법에 대한 실험도 이미지 클러스터링 실험 결과 정확도가 높은 그래프를 대상으로 수행한다. 아웃라이어 결과의 타당성을 검증하기 위해 5명의 평가자들을 통해 해당 클러스터의 아웃라이어 검출의 정확도를 평가한다. 평가 방법은 평가자들에게 이미지 클러스터링 결과로 나온 클러스터를 제공한다. 평가자는 제공된 클러스터의 객체들 중에서 아웃라이어로 생각되는 이미지 객체를 개수에 상관없이 선택한다. 5명의 평가자들을 통해 결정된 아웃라이어 중 3명 이상이 공통적으로 아웃라이어로 선택한 이미지 데이터를 정답으로 간주한다. 평가자들에 의해 공통적으로 선택된 n'개의 아웃라이어들이 논문에서 제시하는 알고리즘을 통해 도출된 아웃라이어 중 등급이 높은 n개의 아웃라이어들과 일치하는지의 여부를 실험을 통해 보인다.

3. 실험 결과

그림 9는 그래프 생성 방법의 따른 클러스터의 정확도를 나타낸 것이다. 이 그림은 대칭적 방법과 비대칭적 방법으로 생성된 그래프에 대한 정확도 비교와 k-최근접 이웃검색에서의 k값의 변화에 따른 정확도 비교를 동시에 나타내고 있다. 가로축은 k값을 의미하고, 세로축은 클러스터의 정확도를 의미한다.

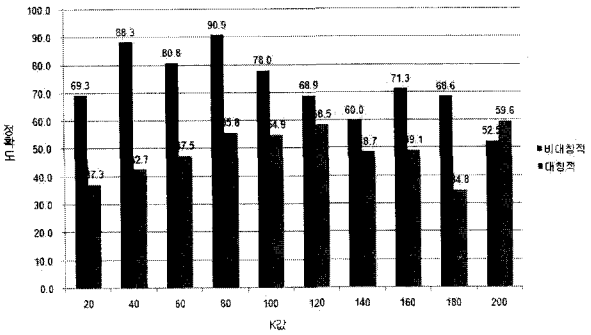
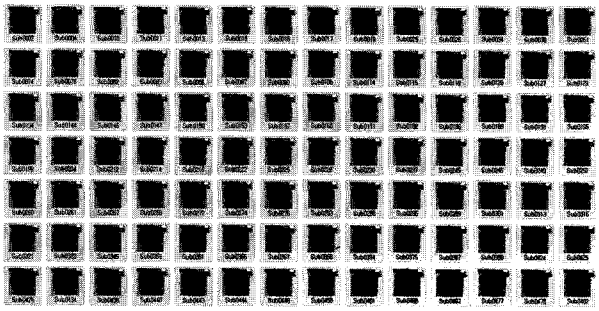


그림 9. 각 데이터 클러스터들의 정확도.
Fig. 9. Accuracy of clusters.

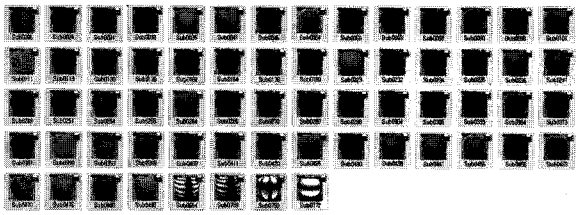
실험 결과 그래프를 대칭적 방법으로 생성했을 때 보다 비대칭적 방법으로 생성했을 때의 클러스터들의 정확도가 일반적으로 높았다. 대칭적 방법은 노이즈 객체에 대한 영향력을 줄이기 위한 방법으로, 이 방법을 이용하면 클러스터의 정확도가 높아 질 것을 예상했다. 그러나 본 논문에서 사용한 이미지 데이터는 노이즈가 적은 경우로서, 이 경우 대칭적 방법은 서로 유사한 객체들 간에 간선이 생성되는 것을 막아서 클러스터의 정확도가 낮아지는 것으로 나타났다.

k-최근접 이웃검색의 k값은 실험 결과 40~80일 때 가장 높은 정확도를 보였다. k값이 20과 같이 아주 작을 때는 실제 유사한 객체들 사이에 간선이 생성되지 않았기 때문에 서로 유사한 객체들이 동일한 클러스터에 포함되지 못했다. 그러나 k가 100이상으로 큰 값을 가질 때에는 k-최근접 이웃검색으로 인해 너무 많은 데이터들이 서로 유사하다고 판단하기 때문에 실제로 유사하지 않는 객체들에 간선 생성되어 같은 클러스터에 포함되었다.

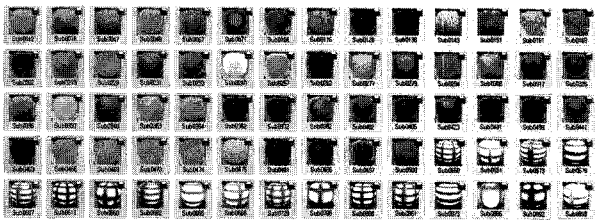
클러스터링의 결과가 가장 정확하게 측정된 경우는 k를 80으로 설정하고 비대칭적인 방법을 이용해서 그래프를 생성했을 때이다. 이때 생성된 클러스터의 개수는 20개였다. 그림 10은 발견된 클러스터들의 일부를 나타낸 것이다. 그림 10을 통해서 데이터간의 유사도를 측정하는데 색상을 이용하였기 때문에 유사한 색상의 이미지들이 같은 클러스터에 존재 하는 것을 볼 수 있다. 그림 10(b)~(f)를 보면 같은 클러스터 안에 무늬가 다른 객체들이 있는 것을 확인 할 수 있다. 이는 본 실험



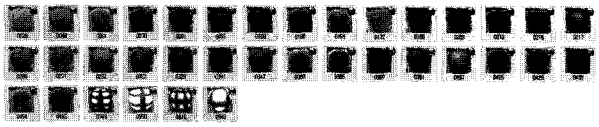
(a) 클러스터 1
(a) cluster 1



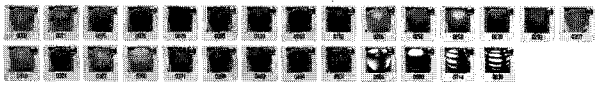
(b) 클러스터 2
(b) cluster 2



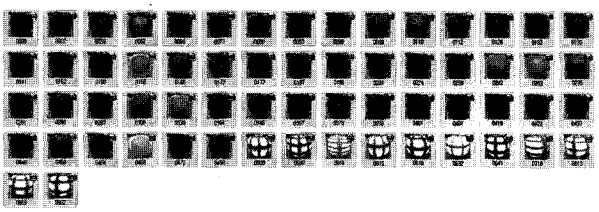
(c) 클러스터 3
(c) cluster 3



(d) 클러스터 4
(d) cluster 4



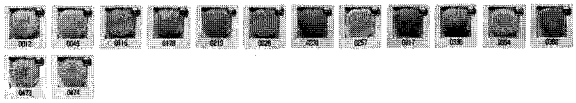
(e) 클러스터 5
(e) cluster 5



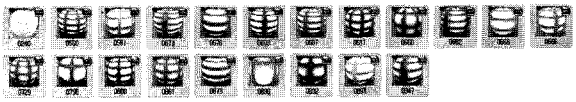
(f) 클러스터 6
(f) cluster 6

그림 10. k=80일 때의 비대칭적 방법을 이용해서 생성한 그래프의 클러스터링 결과.
Fig. 10. Results of clustering(k=80, asymmetric).

(a) 클러스터 3 - 1
(a) cluster 3 - 1



(b) 클러스터 3 - 2
(b) cluster 3 - 2



(c) 클러스터 3 - 3
(c) cluster 3 - 3

(d) 클러스터 3 - 4
(d) cluster 3 - 4

그림 11. 그림 10(c)의 계층적 클러스터링 결과.
Fig. 11. Results of hierarchical clustering about Fig. 10(c).

에서 객체들 간의 유사 여부를 색상으로만 판단했기 때문에 무늬가 다르지만 색상이 유사한 객체들이 같은 클러스터 안에 포함된 것이다.

그림 11은 그림 10(c)를 계층적으로 클러스터링 한 결과이다. 첫 번째 레벨에서의 클러스터 중 하나인 그림 10(c)는 초록색 계통과 파란색 계통이 섞인 이미지 객체들의 모여 있는 클러스터다. 이 클러스터에 계층적 클러스터링을 적용하여 두 번째 레벨로 확장한 결과가 그림 11이다. 두 번째 레벨에서 총 4개의 클러스터가 생성되며 하위 레벨에서 더욱 유사한 이미지 객체가 모여 있음을 시각적으로 확인 할 수 있다. 그림 11(a)는 밝은 초록색에 가까운 이미지들의 집합이며, 그림 11(b)는 하늘색에 가까운 이미지들의 집합이다. 그림 11(c)는 무늬가 있는 이미지들의 집합이다. 해당 이미지들은 흰색을 많이 지니고 있기 때문에 같은 클러스터로 분류가

표 3. 아웃라이어의 정확도
Table 3. Precision of outliers.

클러스터	n′	정확도(precision)
그림 10(d)	7	85.7%
그림 10(e)	4	100%
그림 10(f)	11	90.9%

되었다. 마지막으로 그림 11(d)는 파란색을 지니고 있는 이미지들의 집합이다.

표 3은 그림 10(d)~그림 10(f)의 클러스터를 대상으로 수행한 아웃라이어 검출 실험 결과 중 하나이다. n′는 해당 클러스터에 대해서 5명의 평가자들이 공통적으로 선택한 아웃라이어의 개수이고, 정확도는 평가자들에 의해 공통적으로 선택된 n′개의 아웃라이어를 정답으로 간주하고 본 논문에서 제안하는 방법으로 검출된 n′와 동일한 개수인 상위 n개의 아웃라이어와 일치하는지의 여부를 나타낸 값이다. 표 2를 통해 제안하는 아웃라이어 검출 방법이 높은 정확도를 보임을 알 수

표 4. 아웃라이어 등급
Table 4. Outliers grade.

(a) 그림 10(d) 아웃라이어 등급
(a) Outliers grade about Fig. 10(d).

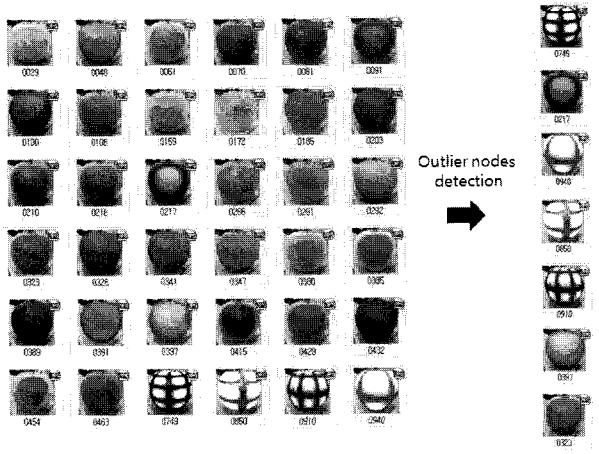
등급	객체 ID	비용감소량(CostFall)
1	749	5.95
2	217	3.86
3	940	3.41
4	850	3.37
5	910	3.24
6	397	1.82
7	323	1.68

(b) 그림 10(e) 아웃라이어 등급
(b) Outliers grade about Fig. 10(e).

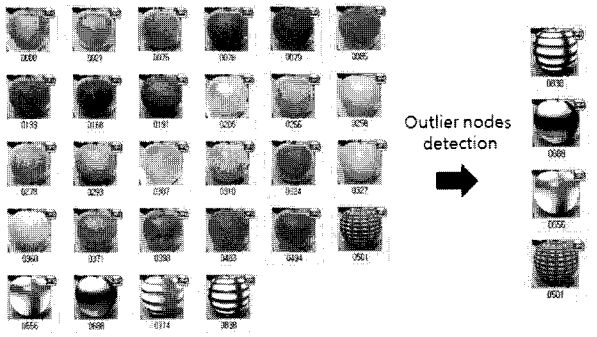
등급	객체 ID	비용감소량(CostFall)
1	838	5.84
2	688	5.83
3	556	5.39
4	501	3.67

(c) 그림 10(f)의 아웃라이어 등급
(c) Outliers grade about Fig. 10(f).

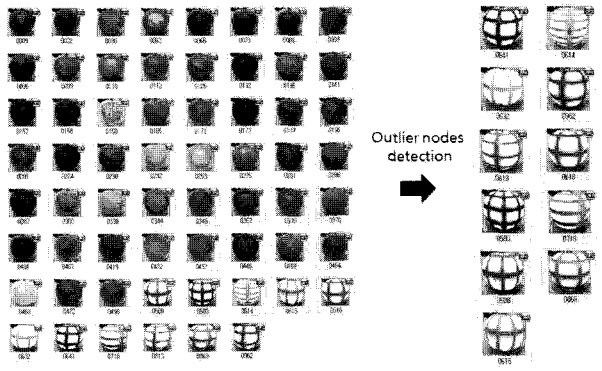
등급	객체 ID	비용감소량(CostFall)
1	641	3.42
2	632	3.09
3	813	2.88
4	580	2.78
5	508	2.69
6	615	2.37
7	614	2.23
8	962	1.98
9	618	1.94
10	718	1.68
11	869	1.50



(a) 그림 10(d)의 아웃라이어 검출 결과
(a) Results of outlier detection about Fig. 10(d)



(b) 그림 10(e)의 아웃라이어 검출 결과
(b) Results of outlier detection about Fig. 10(e)



(c) 그림 10(f)의 아웃라이어 검출 결과
(c) Results of outlier detection about Fig. 10(f)

그림 12. 아웃라이어 검출 결과.
Fig. 12. Results of outlier detection.

있다.

표 4는 본 논문에서 제안하는 방법에 의해 검출된 아웃라이어를 등급별로 나타낸 것이다. 또한 아웃라이어의 객체 ID와 비용감소량도 나타나 있다. 비용감소량은 클러스터의 존재하는 이미지 데이터의 수에 따라 그 값이 상대적이기 때문에 값의 크기보다 값의 순서와 상대적인 값의 차이가 중요하다. 비용감소량의 값의 내림차순에 따라 아웃라이어의 등급이 설정되며, 값이 상대적인 차이가 클수록 더욱 이질적인 아웃라이어임을 의미한다.

그림 12는 아웃라이어의 검출 결과를 나타낸다. 그림 14는 해당 클러스터의 아웃라이어를 표 4에서 나열한 등급별로 보여주고 있다. 그림 12를 통해 제안하는 방법으로 검출한 아웃라이어를 시각적으로 확인할 수 있다. 예를 들면, 그림 12(a)는 파란색 계통의 클러스터다. 이 중 파란색이 거의 포함되어 있지 않는 749번 이미지가 가장 높은 등급의 아웃라이어로 선택됨을 확인 할 수 있다.

V. 결 론

본 논문에서는 클러스터의 개수를 미리 정하지 않고도 이미지 데이터 클러스터링을 수행하기 위해 CA를 이용한 클러스터링 방법을 제안하였다 제안하는 방법은 이미지 데이터를 그래프 생성 기법을 통해서 그래프로 변환하고 변환된 .이미지 데이터를 CA에 적용한 후에 그 결과를 클러스터링 관점에서 재해석하는 방식이다. 실험을 통하여 k가 80이고 비대칭적 방법으로 그래프를 생성해서 CA에 적용했을 때 클러스터의 정확도가 90.9%로 가장 높게 측정되었다. 따라서 본 논문에서는 사용자로부터 매개변수를 요구하지 않는 클러스터링을 위해, CA를 이용하여 이미지 클러스터링을 수행할 때, k를 40~80 사이로 설정하고 비대칭적인 방법을 이용하여 그래프를 생성하는 것을 가이드라인으로 제시한다.

본 논문에서는 또한 클러스터의 정확도가 높은 그래프를 대상으로 재귀적으로 CA를 적용하는 계층적 클러스터링과 아웃라이어 검출 알고리즘을 수행하였다. 다양한 실험을 통하여 제안하는 계층적 클러스터링 방법과 아웃라이어 검출 방법이 타당함을 보였다.

참 고 문 헌

[1] Y. Gdalyahu, D. Weinshall, and M. Werman, "Self-Organization in Vision: Stochastic Clustering for Image Segmentation, Perceptual Grouping, and Image Database Organization," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23, No. 10, pp. 1053-1074, 2001.

[2] Y. Chen, J. Wang, and R. Krovetz, "CLUE: Cluster-Based Retrieval of Images by Unsupervised Learning," *IEEE Trans. Image Processing*, Vol. 14, No. 8, pp. 1187-1201, 2005.

[3] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2006.

[4] S. Guha, R. Rastogi, and K. Shim, "CURE: An Efficient Clustering Algorithm for Large Databases," In *Proc. of ACM SIGMOD Int'l. Conf. on Management of Data*, pp. 73-84, 1998.

[5] T. Zhang, R. Ramakrishnan, and M. Livny, "BIRCH: An Efficient Data Clustering Method for Very Large Databases," In *Proc. of ACM SIGMOD Int'l. Conf. on Management of Data*, pp. 103-114, 1996.

[6] G. Karypis, E. H. Han, and V. Kumar, "Chameleon: A Hierarchical Clustering Algorithm Using Dynamic Modeling," *IEEE Computer*, Vol. 32, No. 8, pp. 68-75, 1999.

[7] D. Chakrabarti, S. Papadimitriou, D. S. Modha, C. Faloutsos, "Fully Automatic Cross-ssociations," In *Proc. Int'l Conf. on Knowledge Discovery and Data Mining*, pp. 79-88, 2004.

[8] P. Grünwald, *A Tutorial Introduction To The Minimum Description Length Principle*, MIT Press, 2005.

[9] S. Papadimitriou, J. Sun, P. S. Yu, C. Faloutsos, "Hierarchical, parameter-free community discovery," In *Proc. of ECML PKDD*, page 170-187, 2008.

[10] D. Chakrabarti, "Autopart: Parameter-free graph partitioning and outlier detection," In *Proc. of ECML PKDD*, pages 112 - 124, 2004.

[11] K. Beyer, J. Goldstein, R. Ramakrishnan, U. Shaft, "When Is Nearest Neighbor Meaningful?," In *Proc. Int'l Conf. on Database Theory*, pp. 217-235, 1999.

[12] 이 재호, 장 민희, 김 두열, 김 상욱, 김 민호, 최 진성, "Shader Space Navigator: 유사 쉐이더 검색 시스템," 대한전자공학회논문지, Vol. 45, No. 3, pp. 198-207, 2008년 5월.

[13] W. Niblack, R. Barber, W. Equitz, M. Flickner, E. H. Clasman, D. Petkovic, P. Yanker, C.

Faloutsos, G. Taubin, "The QBIC Project: Querying Images by Content using Color, Texture, and Shape," In *Proc. of Storage and Retrieval for Image and Video Databases*, pp. 173-187, 1993.

저 자 소 개

오 현 교(정회원)

2008년 한양대학교 정보통신학과 학사 졸업.

2008년~현재 한양대학교 대학원 전자컴퓨터통신공학과 석사과정 재학 중

<주관심분야 : 사회연결망분석, 인터넷 포탈 데이타 분석, e-비지니스, 데이타 마이닝>

윤 석 호(정회원)

2005년 성결대학교 컴퓨터공학과 학사 졸업.

2007년 한양대학교 정보통신대학원 석사 졸업.

2007년~현재 한양대학교 대학원 전자컴퓨터통신공학과 박사과정 재학 중.

<주관심분야 : 사회연결망분석, 인터넷 포탈 데이타 분석, e-비지니스, 데이타 마이닝>

김 상 욱(평생회원)-교신저자

1989년 2월 서울대학교 컴퓨터공학과 학사 졸업.

1991년 2월 한국과학기술원 전산학과 석사 졸업.

1994년 2월 한국과학기술원 전산학사 박사 졸업.

1991년 7월~1991년 8월 미국 Stanford University, Computer Science Department, 방문 연구원.

1994년 3월~1995년 2월 KAIST 정보전자연구소 전문 연구원.

1999년 8월~2000년 8월 미국 IBM T.J. Watson Research Center, Post-Doc.

1995년 3월~2003년 2월 강원대학교 정보통신공학과 부교수.

2003년 3월~현재 한양대학교 정보통신대학 정보통신학부 교수.

2009년 1월~현재 미국 Carnegie Mellon University, Visiting Scholar

<주관심분야 : 데이터베이스 시스템, 저장 시스템, 트랜잭션 관리, 데이터 마이닝, 멀티미디어 정보 검색, 공간 데이터베이스/GIS, 주기억장치 데이터베이스, 이동 객체 데이터베이스/텔레매틱스, 사회 연결망 분석, 웹 데이터 분석>