

효과적인 이미지 검색을 위한 연장 해쉬 (Extendible hash) 기반 인덱싱 및 검색 기법

Indexing and Matching Scheme for Content-based Image Retrieval based on Extendible Hash

탁 윤 식*, 황 인 준**
Yoon-Sik Tak*, Eenjun Hwang**

Abstract

So far, many researches have been done to index high-dimensional feature values for fast content-based image retrieval. Still, many existing indexing schemes are suffering from performance degradation due to the curse of dimensionality problem. As an alternative, heuristic algorithms have been proposed to calculate the result with 'high probability' at the cost of accuracy. In this paper, we propose a new extendible hash-based indexing scheme for high-dimensional feature values. Our indexing scheme provides several advantages compared to the traditional high-dimensional index structures in terms of search performance and accuracy preservation. Through extensive experiments, we show that our proposed indexing scheme achieves outstanding performance.

요 약

보다 빠른 내용 기반 이미지 검색을 위해, 다차원 특징 정보의 효과적인 인덱싱에 대한 다양한 연구들이 수행되고 있다. 하지만, 대부분의 인덱싱 기법들은 특징 정보의 차원이 커질수록 성능이 저하되는 문제를 가지고 있으며, 이를 대체하기 위해서 '높은 확률'로써 사용자가 원하는 결과를 제공해 주기 위한 휴리스틱 (heuristic) 알고리즘을 사용한 기법들이 제안되었다. 본 논문에서는 이러한 다차원 특징 정보를 효과적으로 인덱싱 하기 위해, 연장 해쉬 기반의 새로운 인덱싱 기법을 제안한다. 제안된 인덱싱 기법은 기존의 기법들이 가졌던 문제들을 해결하기 위해, 검색의 정확도에 영향을 주지 않으면서 빠른 검색이 가능하도록 설계되었다. 다양한 실험을 통해, 제안된 기법이 월등한 성능을 가질 수 있음을 보였다.

Key words : Hash, Image Retrieval, Range Search, Shape Feature, Image Indexing

1. 서론

디지털 카메라 및 휴대폰 카메라 등과 같은 휴대용 이미지 촬영 장치의 폭발적인 보급에 따라서 사용자

들은 관리하기 어려울 정도로 많은 양의 이미지들을 저장하고 사용하게 되었다. 이러한 많은 양의 이미지들 중에서 사용자가 원하는 이미지를 효과적으로 제공하기 위해, 다양한 내용 기반의 이미지 검색 기법들이 제안되었다.

정확한 이미지 검색을 위해서는 효과적인 이미지의 특징 정보를 정의해야하는데, 이러한 특징 정보로써 이미지의 색상 (Color), 질감 (Texture) 및 외형 (Shape) 정보가 널리 사용된다. 하지만, 이러한 특징 정보들 중 대부분이 높은 차원 (High-Dimension)을 가지게 됨에 따라, 순차적인 이미지들 간의 비교 방법만으로는 대용량 이미지 데이터베이스를 효과적으

* 高麗大學校 電氣電子電波工學部
(School of Electrical Engineering, Korea University)

★ 교신저자 (Corresponding author)

※ 감사의 글 (Acknowledgment)

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT 연구 센터 지원 사업의 연구 결과로 수행되었음
(NIPA-2010-C1090-1001-0008)

接受日:2010年 12月 3日, 修正完了日: 2010年 12月 29日

로 사용할 수 없기 때문에 보다 빠른 검색을 위한 다양한 인덱싱 기법들이 제안되었다. 대표적인 인덱싱 기법으로는 SS-트리, SR-트리, R-트리 등의 트리 구조를 사용한 기법과 해시를 사용한 기법을 들 수 있다. [1][2]에서는 트리 구조를 사용한 거리 곡선 기반의 이미지 인덱싱 및 검색 기법을 제안하였으며, [3]에서는 거리 곡선과 유사한 타임 시퀀스 (Time Sequence) 기반의 인덱싱 및 검색 기법을 제안하였다. 이러한 트리 기반의 인덱싱 구조에서는 비교적 빠른 시간 안에 유사 이미지를 검색할 수 있지만 해시와 같이 원하는 데이터에 바로 접근할 수 있는 데이터 구조에 비해서는 검색 속도가 느릴 수 있다. [4]에서는 RGB 값을 확장 해시(Extended Hash)에 적용한 인덱싱 및 범위 검색(Range Search) 기법을 제안하였으며, 이러한 기법이 트리 기반의 기법에 비해 월등하게 빠름을 보였다. 하지만, [4]에서 연구된 해시 구조는 RGB 값과 같이 값의 범위가 정해져 있는 (0~255) 경우에 적합하게 구성된 반면, 범위가 정해지지 않은 값으로 구성된 거리 곡선 기반의 이미지 검색에서는 인덱싱이 용이하지 않기 때문에 널리 사용되지 못하였다.

따라서, 본 논문에서는 이미지로부터 추출한 특징 정보가 그 값의 범위가 정해지지 않은 정수 값을 가지더라도 효과적으로 인덱싱 하여 검색에 적용할 수 있도록 새로운 해시 기반의 인덱싱 기법을 제안한다. 또한, 제안된 해시 기반의 인덱싱 구조를 통해 효과적으로 이미지를 검색하기 위한 K-NN 검색 알고리즘을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문과 관련된 기존의 연구들에 대해서 소개하며, 3장에서는 이미지로부터 특징 정보를 추출하여 제안된 해시 기반의 데이터 구조에 인덱싱 하는 일련의 과정에 대해 설명한다. 4장에서는 제안된 인덱싱 구조를 활용한 K-NN 알고리즘에 대해 설명하며, 5장에서는 제안된 기법들의 성능을 평가하기 위한 다양한 실험 결과를 보인다. 마지막으로 6장에서는 본 논문을 결론짓는다.

II. 관련 연구

현재 다차원 특징 정보를 위해 가장 널리 사용되는 인덱싱 구조는 R+-Tree, R*-Tree 등의 변형된 구조를 포함하는 R-Tree이다. 이러한 R-Tree는 [5]에서 다차원 특징 정보를 인덱싱 하기 위한 공간 인덱싱의 목적으로 Guttman에 의해 제안되었다. R-Tree는 공간을 minimum bounding rectangles (MBRs)라 불리

는 영역을 계층적으로 중첩시킴으로써 공간을 표현한다. R-Tree가 다차원 공간 인덱싱에서 좋은 검색 결과를 제공하였지만, 이렇게 중첩된 영역으로 인한 검색 효율의 저하가 발생하였기 때문에 이러한 문제를 줄이기 위해 R+-Tree [6], R*-Tree [7]가 제안되었다.

이러한 R-Tree외에 널리 사용되는 인덱싱 기법들로서 SS-Tree[8]와 SR-Tree[9]가 있다. SS-Tree는

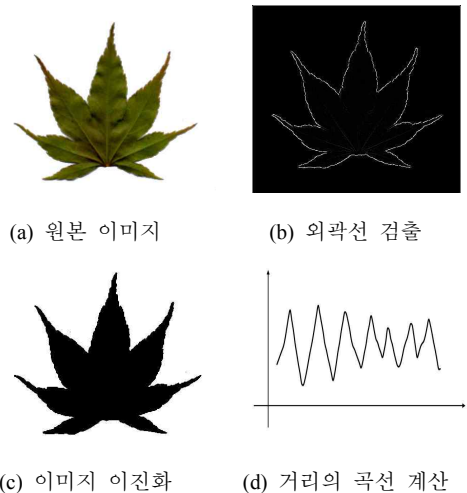


그림 1. 전처리 과정

R-Tree에서 사용되는 minimum bounding rectangles 대신, minimum bounding spheres를 사용한 기법이며, SR-Tree는 이러한 두 개를 혼합하여 사용한 기법이다. 또한, Binary Space Partitioning (BSP)를 사용하는 VP-Tree[10]와 부분 시퀀스 매칭에 사용되는 S2-Tree[11]도 널리 알려진 인덱싱 기법이다.

III. 해시 기반 이미지 인덱싱

1. 외형 기반 이미지 특징정보 추출

보다 정확한 이미지 검색을 위해서는 우선 이미지들을 잘 표현할 수 있는 특징 정보를 추출하여야 한다. 본 논문에서는 이미지에 포함된 주요 객체의 특징 정보로써, 해당 객체의 중심점과 외곽선과의 거리를 계산한 거리 곡선을 사용한다. 본 논문에서는 이미지로부터 이러한 거리 곡선을 계산하기 위해 이미지에 포함된 주요 객체의 중심점과 외곽선을 연결한 거리 곡선을 사용하며, 이러한 거리 곡선을 계산하기 위한 순서는 다음과 같다.

1. 우선 그림 1의 (b)에서와 같이 Canny Edge Detection을 활용해 원본 이미지 (a)에 포함된 주요 객체의 외곽선을 검출한다.
2. 외곽선 정보를 바탕으로 (c)와 같이 객체와 배경을 분리하기 위해 이미지를 이진화 시킨다.
3. 이진화된 영상에서 객체의 중심점으로부터 외곽선과의 거리를 계산하여 연결한 거리 곡선을 (d)와 같이 추출한다.

전처리 과정의 각 단계에 대한 상세한 설명은 [12]에 나타나 있으며, 인덱싱을 위해, 계산된 거리 곡선에 푸리에 변환을 적용하여 계산된 상위 계수 값들을 사용한다.

2. 해쉬 기반의 이미지 인덱싱

제안된 연장 해쉬 기반의 이미지 인덱싱 기법은 이미지로부터 추출된 다차원 특징 정보를 이진화하고, 각 차원으로부터 미리 정의된 길이만큼의 비트를 추출하여 해쉬 주소를 만드는 것을 기반으로 한다. 이를 위해, 본 논문에서는 다음과 같이 몇 개의 항목을 정의하도록 한다.

- **Initial Depth** : 이진화된 다차원 특징정보에서 각각의 차원에서 몇 개의 비트를 추출하여 해쉬 주소를 구성할지를 결정하는 항목이다. 만약 3차원의 특징정보에서 Initial Depth가 (n_1, n_2, n_3) 인 경우, 해쉬 주소는 $n_1 + n_2 + n_3$ 의 길이를 가지게 된다.
- **Initial Bits** : 각각의 이진화된 차원 값에서 오른쪽으로부터 몇 비트 떨어진 곳에서부터 값을 추출할지를 결정하는 항목이다. 만약, Initial Depth가 (n_1, n_2, n_3) 이고, Initial Bits가 (d_1, d_2, d_3) 인 경우, 첫 번째 차원 값으로부터는 오른쪽으로부터 d_1 비트 떨어진 곳에서부터 n_1 개의 비트를, 두 번째 차원 값으로부터는 오른쪽으로부터 d_2 비트 떨어진 곳에서부터 n_2 개의 비트를, 마지막으로 세 번째 차원 값으로부터는 오른쪽으로 d_3 비트 떨어진 곳에서부터 n_3 개의 비트를 추출하여 해쉬 주소를 구성하게 된다.
- **데이터 충돌** : 새로 입력된 이미지로부터 계산된 해쉬 주소에 포함된 데이터 버킷 (data bucket)에 더 이상 새로운 이미지를 넣을 공간이 없을 경우 발생한다. 이러한 데이터 충돌을 없애기 위해서는 기존의 해쉬 주소 계산 법을 변경하여 해쉬 주소의 길이를 더 길게 하여 충돌된 이미지들의 해쉬 주소 값을 다르게 하여야 한다. 이러한 추가적으로

입력 될 해쉬 주소 값을 표현하기 위해 Bit Expansion 항목이 사용된다.

- **해쉬 범위 충돌** : Initial Bits 항목을 두어 각각의 차원 별로 비트를 추출할 위치를 정함으로써 발생한다. 만약, 새롭게 입력된 이미지가 기존의 이미지들에 비해 특정 차원의 비트 수가 더 클 경우, 해당 이미지를 인덱싱 하기 위해서는 기존의 해쉬 주소 계산 방법을 변경해야한다.

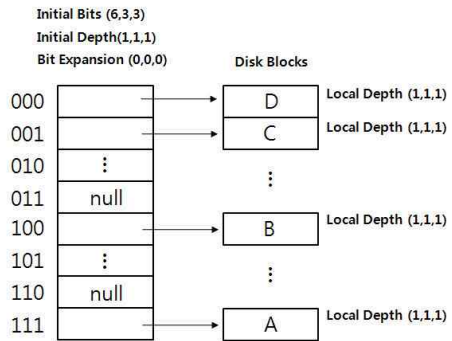


그림 2. 기본적인 이미지 인덱싱을 적용한 예

이러한 내용들을 기반으로, 실질적으로 이미지들을 해쉬 기반의 인덱스 구조에 삽입하는 과정은 예로써 설명하도록 한다.

가. 기본적인 이미지 인덱싱

비어있는 해시 구조에 처음으로 (36, 4, 7: 100100, 100, 111)의 인덱스 값을 가지는 A 이미지가 들어올 경우, 해시의 초기 비트의 수로써 인덱스 값을 구성하는 값들의 비트들의 수(6,3,3)를 Initial Bits에 기록한다. 이후 인덱스 될 이미지들의 인덱스 값들이 초기 비트 수보다 작을 경우, 앞의 비트들을 0으로 채움으로써, 같은 수의 비트로 구성된 특징 정보를 가지도록 한다. Initial Depth가 (1,1,1) 즉, 모든 인덱스 값들의 처음 비트의 수를 해시 주소로 사용하기 때문에, 이미지 A의 해시 주소 값은 (1,1,1)이 되며, 해당 해시 주소는 이미지 A를 가리킨다. 이후, 이미지 B (36, 1, 1: 100100, 001, 001), 이미지 C (31, 3, 7: 01111₍₂₎, 011₍₂₎, 111₍₂₎), 이미지 D (11, 1, 3: 001011₍₂₎, 001₍₂₎, 011₍₂₎)이 차례로 들어올 경우, 모든 이미지들의 인덱스 값들의 비트 수가 Initial Bits의 수보다 크지 않고, 인덱스 주소가 중복되지 않기 때문에 계산된 해시 주소가 차례로 가리키게 된다. 그림 2에서는 이미지 A, B, C, D가 해시 구조에 차례로 인덱싱 된 모습을 나타내고 있다.

나. 데이터 충돌을 일으키는 이미지 인덱싱

기준에 인덱스 된 이미지와 중첩될 경우, 해시를 분할 (Split)해야 한다. 그림 2의 예제에 (48, 1, 3: 110000⁽²⁾, 001⁽²⁾, 011⁽²⁾)의 인덱스 값을 가지는 이미지 E를 인덱스에 추가해야 할 경우, 이미지 E와 기준에

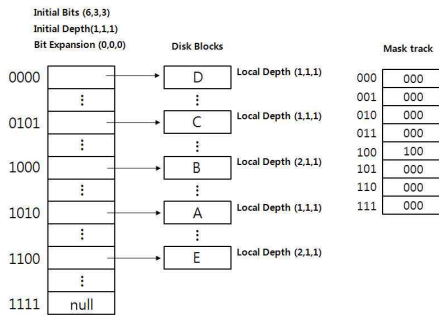


그림 3. 데이터 충돌에 의한 해쉬 분할

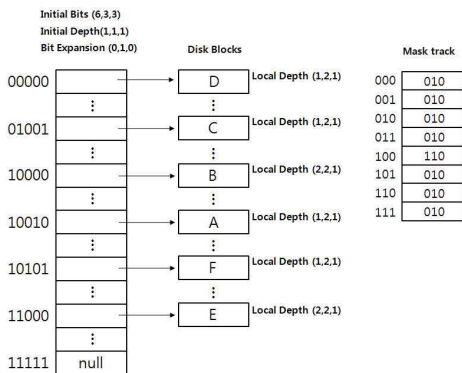


그림 4. 해쉬 범위 충돌에 의한 해시 분할

인덱스 된 이미지 B의 해시 주소 값이 (1, 0, 0)으로 중첩되기 때문에 중첩이 일어나지 않게 하기 위해서는 해시를 분할하여야 한다. 가장 적은 수의 해시 분할을 통해 모든 이미지들을 인덱스 하기 위해서 본 논문에서는 이미지들의 인덱스 값들 중 가장 변화가 큰 값들 순서로 해시를 분할하기 위해 사용된다. 본 예제에서는 첫 인덱스 값의 변화가 가장 크다고 가정한다. 따라서, 이미지 B와 E를 인덱스 하기 위해 첫 인덱스 값을 통해 해시가 분할되며, 이러한 분할을 기록하기 위해, 이미지 B와 E가 인덱스 된 곳의 Local Depth와 해시의 분할을 기록하는Mask Track

의 분할이 일어나게 된 해시 주소 값인 '100'에 첫 인덱스 값을 통해 분할이 되었음을 나타내는 (1,0,0) 값을 기록한다. 그림 3은 이러한 데이터 충돌에 의한 해시 분할 과정을 나타내고 있다.

다. 해쉬 범위 충돌을 일으키는 이미지 인덱싱

그림 3에서 설명한 중첩에 의한 해시 분할 외에, 기준에 인덱스 된 이미지들보다 더 많은 비트로 구성된 인덱스 값을 가지는 이미지가 인덱스 되어야 할 경우, 이를 위한 해시 분할도 이루어져야 한다. 그림 3의 예제에서는 Initial Bits가 (6,3,3)으로 이루어져 있으므로, 이미지들의 인덱스 값들은 6비트, 3비트, 3비트를 초과하는 값을 가질 수 없다. 이 때, (33, 8, 7:100001⁽²⁾, 1000⁽²⁾, 111⁽²⁾)의 값을 가지는 이미지 F가 인덱스 되어야 할 경우, 이미지 F의 인덱스 값들은 6 비트, 4 비트, 3 비트로 구성됨으로써 기존의 인덱스 구조에 비해 2번째 인덱스 값이 1 비트 더 많다. 이러한 이미지 F를 인덱스 하기 위해, Bit Expansion을 (0,1,0)으로 변경하고 Initial Bits + Bit Expansion의 크기인 6bit, 4bit, 3bit 인덱스 값을 사용해서 해시 주소를 계산할 수 있도록 확장한다. 이러한 확장을 기록하기 위해서 해시에 인덱스 된 모든 이미지들의 두 번째 인덱스 값의 Local Depth와 Mask Track의 모든 항목의 두 번째 값을 1 증가시킨다.

이러한 해시의 분할에 따라 해시 함수는 인덱스 값들의 총 비트 수에서 Initial Depth와 Mask Track에서 각각의 인덱스 값의 분할을 나타낸 값들 중 최고 값인 '110'을 더한 값인 (2,2,0)개의 상위 비트를 선택한 값이 된다. 변경된 해시 함수에 따라서, 이미지 F의 해시 주소 값은 '10' + '10' + '1'인 '10101'이 되고, 인덱스 된 모든 이미지들도 변경된 해시 함수에 따라 재 계산된 주소로 이동한다. 그림 4는 이러한 중첩에 의한 해시 분할 과정을 나타내고 있다.

IV. K-NN 검색 기법

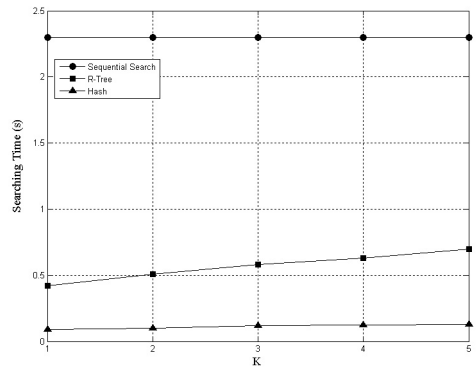
이미지 검색에서 가장 많이 사용되는 검색 기법 중 하나는 질의와의 차이가 작은 순으로 K개의 이미지들을 결과로써 제공하는 K-NN 검색 기법이다. 본 장에서는 3장에서 제안된 해시 기반의 이미지 인덱스를 사용한 새로운 범위 검색 기법을 제안한다. 본 논문에서 제안하는 K-NN 기법은 우선 순위 큐를 사용함으로써 불필요한 계산을 줄이도록 한다. 이러한 우선 순위 큐 기반의 검색 기법은 인덱스 주소 값들 간의

차이를 먼저 계산하고 필요한 경우에 한해 비교적 많은 시간이 걸리는 이미지들 간의 특징 정보간의 차이를 계산함으로써 검색 속도를 빠르게 할 수 있다. 제안된 연장 해쉬 기반의 인덱스 구조를 활용한 K-NN 기법의 실행 순서는 다음과 같다.

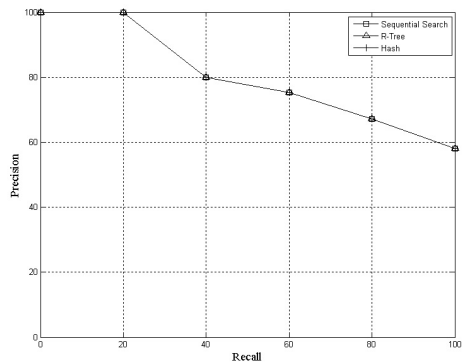
1. Initial Depth와 Initial Bits로 계산되는 각각의 해쉬 주소 값과 질의 영상과의 차이를 계산하여 우선 순위 큐에 넣는다. 이 경우, Mask track을 활용하여 실질적으로 이미지가 인덱싱되어 있지 않은 경우, 이 과정을 수행하지 않는다.
2. Result의 크기가 K일 때까지 다음을 반복한다.
 - 1) 큐의 Top이 실질적인 이미지와 질의 영상간의 다차원 특징정보들의 차이 값을 포함할 경우, Top을 Result에 넣는다.
 - 2) 큐의 Top이 최종적으로 확장된 해쉬 주소를 포함 할 경우, 해당 해쉬 주소의 버킷에 포함된 영상들과 질의 영상과의 특징 정보 간의 차이 값을 계산하여 큐에 넣는다.
 - 3) 그 외의 경우 (Top의 해쉬 주소 값이 확장되는 과정인 경우), Bit Expansion과 Mask Track 정보를 활용하여 질의 영상과 Top의 해쉬 주소 값의 다음 확장 값들 간의 차이를 계산하여 큐에 넣는다.

영상 A와 B의 해쉬 주소간의 차이는 해쉬 주소를 구성하는 각각의 차원 값 간의 차이의 합으로써 계산되며, 해쉬 주소에 포함된 i 번째 차원 값들 간의 차이인 $D(h(A_i), h(B_i))$ 의 계산은 식(1)에 나타나있다. 해당식에서 End_p 는 각 차원에서 해쉬 주소 계산을 위해 추출된 비트 값들의 최 우측에 위치한 비트의 위치 값을 의미한다. (예: 11101에서 111이 추출되었을 경우, $End_p=3$)

$$D(h(A_i), h(B_i)) = \left\{ \begin{array}{l} \text{if } h(A_i) = h(B_i), \\ 0 \\ \text{Otherwise,} \\ (\max(h(A_i), h(B_i)) \\ - (\min(h(A_i), h(B_i)) + 1)) \\ \times 2^{End_p} \end{array} \right\} \quad (1)$$



(a) 검색 시간



(b) 검색 정확도

그림 5. 실험 결과

V. 실험 결과

본 논문에서 제안한 기법의 성능을 측정하기 위해서, 이미지 데이터베이스에 대한 실험을 수행하였다. 실험에 사용된 이미지들은 600개 정도의 일 이미지들이며, 해당 이미지들은 인터넷을 통한 수집되거나 직접 촬영 및 스케치 되었다. 실험 환경으로는 Intel Linfield i5 CPU와 4 GB의 RAM을 가진 시스템이 사용되었으며, 본 논문에서 제안한 해쉬 구조를 통한 이미지 인덱싱 및 검색 기법들은 C# 언어로 구현되었다.

1. 검색 시간

본 논문에서 제안한 기법들의 성능을 평가하기 위해, 우선 검색 시간을 측정하였다. 비교를 위해, i) 인덱스를 사용하지 않은 순차적인 영상 비교 기법 (Sequential Search), ii) R-Tree 인덱스 구조를 사용

한 기법, iii) 본 논문에서 제안한 Hash 인덱스 구조를 사용한 기법의 3가지 경우에 대해 K-NN 검색을 실행하였다. 검색 시간은 그림 5 (a)에 나타나 있다. 그림을 통해서, 본 논문에서 제안하는 인덱스 구조를 활용한 검색을 실시할 경우, 순차 검색 및 R-Tree 등의 기존의 검색 방법에 대해 월등히 빠른 시간 내에 결과를 얻을 수 있음을 알 수 있다.

2. 검색 정확도

다음으로, 검색 정확도를 계산하였다. 비교를 위해, 검색시간에서와 동일한 3가지 기법들에 대해서 검색 정확도를 평가하였으며, 결과는 그림 5 (b)에 나타나 있다. 그림을 통해서, 본 논문에서 제안하는 Hash 기반의 검색 기법이 순차 검색 및 R-Tree와 동일한 검색 정확도를 보임을 알 수 있다. 이러한 결과를 통해서, 본 논문에서 제안하는 해쉬 기반의 인덱싱 기법은 검색의 정확도를 유지하면서 검색 시간의 향상을 얻을 수 있음을 알 수 있다.

VI. 결론

본 논문에서는 효과적인 내용 기반 이미지 검색을 위한 연장 해쉬 기반의 새로운 인덱싱 기법을 제안하였다. 기존의 해쉬 구조가 다차원 특징 정보를 활용한 이미지 검색에 적합하지 않고, 값의 범위가 알려진 특징 정보에 한해서 사용되었던 것에 비해, 본 논문에서 제안된 새로운 해쉬 기반의 인덱싱 기법은 값의 범위가 제한되어 있지 않은 양의 정수 값을 가지더라도 효과적으로 이미지를 인덱싱 할 수 있도록 설계되었다. 또한, 본 논문에서는 이렇게 제안된 인덱싱 구조를 활용한 새로운 K-NN 검색 기법을 제안함으로써 효과적인 이미지 검색이 가능하도록 하였다. 실험을 통해, 본 논문에서 제안하는 해쉬 기반의 검색 기법은 검색의 정확도에 영향을 주지 않으면서, 기존의 기법들에 비해 보다 빠른 검색이 가능함을 보였다.

참고문헌

[1] Eamonn Keogh, Li Wei, Xiaopeng Xi, Sang-Hee lee and Michail Vlachos, "LB_Keogh supports exact indexing of shapes under rotation invariance with arbitrary representations and distance measures," VLDB'06, pp.882 - 893, 2006

[2] Yoon-Sik Tak and Eenjun Hwang, "An indexing scheme for efficient camera angle invariant image

retrieval," CIT'08, pp.143-148, 2008

[3] E. Keogh and C. Ratanamahatana, "Exact indexing of dynamic time warping," Knowledge and Information Systems, Vol.7, pp. 358-386, 2005

[4] Shu Lin, M. Tamer Özsu, Vincent Oria and Raymond T. Ng, "An Extendible Hash for Multi-Precision Similarity Querying of Image Databases," VLDB'01, pp. 221 - 230, 2001

[5] Antonin Guttman, "R-trees: a dynamic index structure for spatial searching," Proceedings of the 1984 ACM SIGMOD international conference on Management of data, pp.47 - 57, 1984

[6] T. Sellis, N. Roussopoulos, and C. Faloutsos, "The R+-Tree: A dynamic index for multi-dimensional objects," In Proc. of the Int. Conference on Very Large Databases, 1987.

[7] Norbert Beckmann, Hans-Peter Kriegel, Ralf Schneider, Bernhard Seeger, "The R*-Tree: An Efficient and Robust Access Method for Points and Rectangles," SIGMOD Conference pp.322 - 331, 1990.

[8] D.A. White and R. Jain, "Similarity indexing with the SS-Tree," In Proc. of the 12th International Conference on Data Engineering, pp.516 - 523,1996

[9] N. Katayama and S. Satoh, "The SR-Tree: an index structure for high dimensional nearest neighbor queries," In Proc. of the ACM SIGMOD International Conference on Management of Data, pp.69 - 380, 1997.

[10] Yianilos and Peter N, "Data structures and algorithms for nearest neighbor search in general metric spaces," Proc. of the fourth annual ACM-SIAM Symposium on Discrete algorithms, pp. 311 - 321, 1993.

[11] H.Wang, C.-S.Perng, "The S2-Tree: an index structure for subsequence matching of spatial objects," In Fifth Pacific-Asic Conference on Knowledge Discovery and Data Mining (PAKDD), 2001.

[12] Y. Tak and E. Hwang, "A Leaf Image Retrieval Scheme Based on Partial Dynamic Time Warping and Two-Level Filtering," CIT'07, pp. 663-638, Oct. 2007

 저 자 소 개

탁 윤 식 (학생회원)

2005년 : 동국대학교 컴퓨터공학과 졸업 (공학사)

2005년 ~ 현재: 고려대학교 대학원 전자컴퓨터공학과 (석·박사통합과정)

<주관심분야> 데이터베이스, 멀티미디어 시스템, 타임 시리즈,

영상 인덱스 및 검색 프레임워크 개발

life993@korea.ac.kr

황 인 준 (정회원)

1998~1999: Bowie State Univ., 조교수

1999~1999: Hughes Research Lab. 연구교수

1999~2003: 아주대학교 정보통신 전문대학원 조교수

2003~2004: 아주대학교 정보통신 전문대학원 부교수

2004~2007: 고려대학교 전기전자전파공학과 조교수

2008~현재: 고려대학교 전기전자전파공학과 교수

<주관심분야>

데이터베이스, 멀티미디어 검색, 정보통합, 전자상거래, 영상처리, 유비쿼터스 컴퓨팅

ehwang04@korea.ac.kr