

이미지 객체의 모양 특징에 기반한 검색 방안

김 영 태^{*}, 엄 기 현
동국대학교 대학원 컴퓨터공학과

Search strategy based on shape feature of image object

Youngtae Kim^{*}, Kyhyun Um
Dept. of Computer Engineering, Graduate School, Dongguk University
{yyoungs, khum}@dgu.ac.kr

요약

이미지 데이터베이스에서 이미지의 객체에 대한 모양 정보는 효율적인 유사성 검색을 위해 매우 중요하다. 본 논문에서는 객체의 지역적인 모양 특징 정보를 이용한 유사성 검색 방안을 제안한다. 이 검색 방안은 사용자의 질의를 이용하여 각 이미지 객체에 대하여 부분 검색 및 전체 검색을 지원한다. 이 때, 유사성 검색을 위해 사용되는 객체의 모양 특징은 지역 특징을 지닌 (거리 r , 각도 θ)의 집합으로 표현되며 같은 객체에 대하여 위치 변화, 크기 변화, 회전시 항상 일정한 값을 지닌다.

1. 서론

대용량의 멀티미디어 데이터베이스에서 사용자의 다양한 요구를 효율적으로 지원하기 위해 적절한 내용 기반 이미지 검색과 그에 따른 유사성 검색이 요구된다. 내용 기반 이미지 검색은 색상, 모양, 질감 등 이미지의 기본 특징을 이용하여 인덱스를 구성함으로써 이미지에 대한 질의를 처리하는 방법이다. 이미지 정보 중에서 모양 정보는 인간이 시각적으로 이미지 객체를 식별하기 위하여 사용되는 중요한 요소이다.

객체의 모양 특징은 이미지 객체의 유사성 검색에서 부분 검색(partial search)과 전체 검색(whole search)을 지원하는 고차원 인덱스 구조에 이용될 수 있어야 한다. 그러나 대부분의 연구에서는 객체 모양의 전역 특징을 이용하고 있으며 각 특징간의 거리는 유클리디언 함수에서 계산된 절대 거리로 산출되므로 객체가 가진 모양에 대한 부분 검색의 지원이 미약하다[3, 4, 5]. 따라서 객체 모양의 지역 특징에 대하여 상대거리를 이용함으로써 이미지 객체의 전체 검색과 부분 검색을 효율적으로 지원하는 연구가 필요하다[2, 6].

각 이미지별로 각 객체의 모양 특징 정보를 저장한 이미지 데이터베이스에서 객체 단위의 유사성 질의 형태는 포인트 질의(point query), 범위 질의(range query), k-최근접 탐색 질의(k-nearest neighbor query)로 나누어진다. 포인트 질의는 주어진 질의와 정확하게 일치하는 객체를 검색하는

질의이고, 범위 질의는 주어진 질의로부터 임의의 유사성 범위 내에 포함되는 객체를 검색하는 질의이며, k-최근접 탐색 질의는 주어진 질의 객체와 가장 유사한 객체 k개를 찾는 질의이다. 이러한 질의 형태는 내용 기반 이미지 검색에서 매우 중요하므로 이를 효율적으로 지원하는 유사성 검색 방안이 요구된다.

본 논문에서는 이미지 객체의 유사성 검색시 부분 검색과 전체 검색을 지원하기 위해 정규화된 객체의 모양 특징을 이용한 유사성 검색 알고리즘을 제안한다. 제안 알고리즘은 재귀적 유평선 근사 알고리즘에서 얻은 시작점을 기준으로 한 객체의 모양 특징의 순서를 고려함으로써 객체간의 유사성을 비교한다.

본 논문의 구성은 다음과 같다. 2장에서는 객체의 모양을 표현하는 방법과 정규화를 제시한 관련 연구를 언급한다. 3장에서는 이미지 객체의 모양 특징 추출을 설명하고 4장에서는 객체의 모양 질의를 위한 유사성 검색 방안을 제안한다. 5장에서는 이미지 객체의 유사성 검색 방법의 실험과 분석 결과에 대해 살펴본다. 6장에서 결론 및 향후 연구 방향에 대하여 기술한다.

2. 관련 연구

2.1. 이미지 객체의 모양 특징 추출

객체의 모양 특징을 표현하는 경계 기반 방법에서 객체의 모양 특징은 직경, 면적, 중심, 전체 모양을 나타내는 사각형 또는 삼각형 등과 같은 전체적인 모양으로부터 추출되는 전역 특징과 다각형 근사, 곡률, 푸리에 서술자, 모멘트, 체인 코드 등과 같은 모양의 지역적인 영역으로부터 추출되

본 연구는 1999년도 한국과학재단 특정기초연구(과제 번호: 1999-1-303-002-3) 내용의 일부임

는 지역 특징이 있다.

특히 경계 기반 방법의 지역적인 특징인 다각형 근사는 객체 모양의 유사성 비교시 윤곽선상의 점들을 줄임으로써 얻은 우세 점들을 이용하여 객체의 모양을 표현하는 방법이다. 그러나 객체의 모양 특징을 표현하는 우세 점들은 크기 변화와 회전에 민감하므로 같은 객체에 대하여 객체의 위치 변화, 크기 변화, 회전시 항상 일정한 값을 유지하도록 하기 위해 정규화를 적용해야 한다. 우세 점들을 이용한 정규화를 진행함으로써 객체의 모양 특징인 (r, θ) 를 얻는다[1].

2.2. 지역 특징을 이용한 검색 방법

대부분의 연구에서는 객체 모양의 절대 거리로 계산된 전역 특징을 먼저 검색한 후, 지역적인 특징을 이용하여 유사성 검색을 하고 있다[3, 4, 5]. 이 방법은 객체 모양의 전역특징을 이용하기 때문에 부분 검색을 지원할 수 없다.

이 검색 방법들은 객체 모양의 두 가지의 전역 특징인 elongation과 compactness를 이용하여 검색 공간을 줄임으로써 빠른 검색을 수행하는 방법[3], 접촉점, 각도 시퀀스의 조합으로 객체의 모양 시그니처를 생성하고 푸리에 변환을 이용하여 모양 시그니처의 차수를 줄임으로써 검색하는 방법[4], 면적, 원형, 최대 회전 축, 대수 모멘트 등의 22개의 전역 특징을 이용한 검색 방법이 있다[5].

객체의 지역적인 특징을 이용하고 각 특징간의 상대 거리를 계산함으로써 부분 검색을 지원하는 방법이 있다[2, 6]. 객체의 모양 특징은 곡률과 각도의 집합으로 구성되며 곡률과 각도를 일련의 토큰으로 표현하여 두 비교 객체의 각 토큰들을 행렬 공간에서 거리를 계산한다[2]. 이 방법은 객체의 모양 특징의 시작점이 유사도 비교 기준이 되지 않으므로 false drop이 커지게 된다. 다각형 근사 알고리즘을 이용하여 객체 모양의 지역 특징인 고정된 차수의 우세 점들을 추출하여 인접한 우세 점간의 각도, 인접한 우세 점간의 거리, 우세 점들의 좌표를 산출하는 방법이다[6]. 두 객체간의 유사성은 editing distance에 의해 계산된다. 이 방법은 모양 특징의 작은 변화에 매우 민감하다.

본 논문에서는 재귀적 윤곽선 근사 알고리즘에서 얻은 우세 점들을 무게 중심과 시작점간의 거리를 기준으로 무게 중심과 우세 점들의 거리에 대한 비율로 거리 r 로 정규화하고 무게 중심을 원점으로 하여 시작점과 각 우세 점간의 각도 θ 로 정규화하여 모양의 지역 특징을 추출한다. 이 모양 특징을 이용하여 두 객체간의 유사성 검색을 할 때, 부분 검색과 전체 검색을 지원하도록 한다.

3. 이미지 객체의 모양 특징 추출

이미지 객체의 윤곽선상의 모든 점을 객체 모양의 식별에 이용한다면 시간이 많이 소모되기 때문에 객체 모양의 특징을 나타내는 적은 수의 우세 점들을 추출하는 것이 필요하다. 따라서 이를 위해 재귀적 윤곽선 근사 알고리즘[1]과 객체의 위치 변화, 크기 변화 및 회전시 그 객체 모양을 표현하는 특징이 항상 일정하도록 하기 위해 우세 점들에 대한 정규화를 적용한다.

3.1 우세 점 추출

객체 모양을 표현하는 윤곽선상의 우세 점들을 추출하는 재귀적 윤곽선 근사 알고리즘을 적용하여 그림 1과 같이 우세 점들을 추출한다.

Algorithm Interest_Points

1. 인덱스 집합과 우세 점 집합을 초기화한다.
2. 객체의 무게 중심을 찾는다.
3. 무게 중심으로부터 최대거리인 시작점을 찾은 후, 시작점을 기준으로 윤곽선 상의 모든 점을 시계 방향으로 재정렬한다.
4. 시작점을 기준으로 최대거리인 끝점을 찾는다.
5. 시작점 위치와 끝점 위치를 인덱스 집합에 삽입한다.
6. 시작점 위치로부터 끝점 위치와 끝점 위치로부터 시작점 위치에 대하여 *Shape_contour* 재귀 함수를 차례로 호출한다.
7. 인덱스 집합내의 인덱스들을 오름차순으로 정렬한 후, w 개의 우세 점 집합 $\{p_{i_1}, p_{i_2}, \dots, p_{i_w}\}$ 을 추출한다.

Function Shape_contour (s, e)

/* s: 시작점 위치, e: 끝점 위치 */

1. 선분 시작점·끝점으로부터 최대거리인 중간점을 찾는다.
2. 만약 선분 시작점·끝점과 중간점간의 거리가 임계값보다 크다면, 중간점을 가리키는 위치를 인덱스 집합에 삽입하고 시작점 위치로부터 중간점 위치와 중간점 위치로부터 끝점 위치에 대하여 *Shape_contour* 재귀 함수를 차례로 호출한다.
3. 만약 선분 시작점·끝점과 중간점간의 거리가 임계값보다 작거나 같다면, 재귀 함수를 종료한다.

그림 1. 재귀적 윤곽선 근사 알고리즘

이 알고리즘에서 얻은 시작점은 객체의 무게 중심으로부터 최대 거리를 가지며 그 위치는 항상 일정하다. 그러므로 같은 객체 모양에 대하여 항상 일정하며 동일한 순서를 가진 우세 점들을 추출할 수 있다.

3.2 우세 점의 정규화

재귀적 윤곽선 알고리즘을 적용하여 추출한 우세 점들은 절대 좌표에 표현되므로 객체의 크기 변화와 회전시 일정한 값을 지니지 않는다. 따라서 모양 특징이 항상 일정하도록 하기 위해 우세 점들의 집합 $U = \{p_{i_1}, p_{i_2}, \dots, p_{i_w}\}$ 에 대하여 그림 2와 같이 정규화를 적용한다.

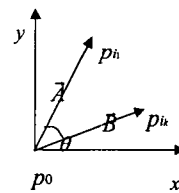


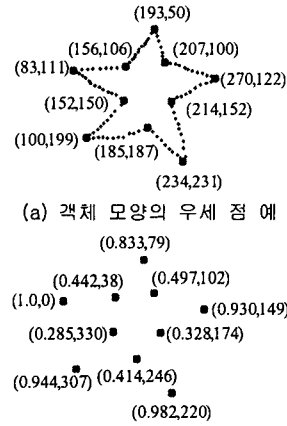
그림 2. 극좌표의 정의 예

먼저 이미지 객체의 무게 중심 p_0 을 원점으로 한 극좌표를 그림 2와 같이 정의한다. 다음과 같이 우세 점들에 대해 정규화를 적용하여 객체의 모양 특징인 거리 r , 각도 θ 를 얻는다.

$$r = \frac{D(p_0, p_{i_k})}{D(p_0, p_{i_1})}, (1 \leq k \leq w)$$

$$\theta = \angle \langle p_{i-1}, p_{i+1} \rangle, (1 \leq k \leq w)$$

그림 3에서는 객체 모양의 특징인 w 개의 우세 점들에 대하여 정규화를 적용함으로써 (r_k, θ_k) 의 집합을 추출한 결과를 보여준다.



(a) 객체 모양의 우세 점 예
(b) 우세 점들에 대한 (r, θ) 의 예
그림 3. 정규화 적용 예

그림 3에서는 객체 모양을 표현하는 우세 점들의 예(a)와 이 우세 점들의 좌표를 이용하여 객체의 모양 특징인 w 개의 (r_k, θ_k) 로 변환되는 예(b)를 보여준다. 모든 이미지 내 객체에 대하여 무게 중심으로부터 최대 거리를 갖는 시작점의 (r, θ) 는 항상 $(1.0, 0)$ 을 가진다.

이 객체의 모양 특징은 같은 객체에 대하여 위치가 변하거나 크기가 변하거나 또는 회전될 경우에도 항상 동일한 값을 유지한다.

4. 객체 모양 질의를 위한 유사성 검색

이 절에서는 질의 객체와 이미지 데이터베이스에 저장된 객체간의 유사성 비교를 위해 각 객체의 모양 특징을 이용한 유사성 검색 알고리즘(Shape_Similarity)을 설명한다(그림 4).

Algorithm Shape_Similarity(A, B)

```

/*  $A = \{ a_1, a_2, \dots, a_w \}$ : 질의 객체의 모양 특징 집합
    $B = \{ b_1, b_2, \dots, b_w \}$ : 데이터베이스에 저장된 객체의 모양 특징 집합
    $w$ :  $A$ 의 모양 특징 개수
    $w$ :  $B$ 의 모양 특징 개수
    $d_{ij}$ :  $A$ 와  $B$ 의 모든 특징간의 차
    $\min(d_{i1}, d_{i2}, \dots, d_{ij})$ :  $i$ 번째 행을 기준으로  $j$ 번째 열에서 가장 최소인 값
    $K = \{1, 2, \dots, i\}$ : 참조된  $i$ 의 집합
    $L = \{1, 2, \dots, j\}$ : 참조된  $j$ 의 집합
    $d(A, B)$ :  $A$ 와  $B$ 간의 유사도 거리
    $\delta$ : 모양 특징의 지역 가중치

```

```

 $\phi$ : 유사도 계수(0.7)
 $\lambda$ : 모양의 전역 가중치
symmetry: 거울 반사된 모양 검사 여부 */
begin
초기화.  $K = L = \emptyset, symmetry = false$ 
단계 1. 집합  $A, B$ 의 모든 특징들에 대하여  $d_{ij}$ 를 계산한다.
    즉,  $d_{ij} = Feature\_distance(a_i, b_j),$ 
         $(1 \leq i \leq w, 1 \leq j \leq w)$ 
단계 2. 만약  $w > w'$ 이라면, 집합  $A$ 의 모든 모양 특징  $(1 \leq j \leq w')$ 을 기준으로 할당되지 않은  $i$ 번째 행의 최소 값  $d_{ij}$ 를 구한다. 이 때 만약  $(i \notin K, j \in L)$ 이라면,  $d_{ij}$ 를 할당하고,  $i$ 와  $j$ 를 각각 집합  $K, L$ 에 삽입한다.
    즉,  $d_i = \min(d_{i1}, d_{i2}, \dots, d_{ij}),$ 
         $K = K \cup \{i\}, L = L \cup \{j\}$ 
단계 3. 만약  $w \leq w'$ 이라면, 집합  $A$ 의 모든 모양 특징  $(1 \leq i \leq w)$ 을 기준으로 할당되지 않은  $j$ 번째 열의 최소 값  $d_{ij}$ 를 구한다. 이 때 만약  $(i \in K, j \notin L)$ 이라면,  $d_{ij}$ 를 할당하고,  $i$ 와  $j$ 를 각각 집합  $K, L$ 에 삽입한다.
    즉,  $d_j = \min(d_{i1}, d_{i2}, \dots, d_{ij}),$ 
         $K = K \cup \{i\}, L = L \cup \{j\}$ 
단계 4. 만약 집합  $K, L$ 내 원소들이 시계방향 또는 반시계방향으로 정렬되어 있지 않다면, 알고리즘을 종료한다.
단계 5. 만약  $(\forall i \in 1, 2, \dots, w, d_i \leq \delta)$ 의 개수  $(k) \geq w \cdot \phi$ 이라면, 식 4.1과 같이  $A$ 와  $B$ 간의 유사도 거리  $d(A, B)$ 를 계산한다.
단계 6. 만약  $symmetry = false$ 이고  $k < w \cdot \phi$ 이라면, 집합  $A$ 를  $a_1$ 을 기준으로 반 시계방향  $(\theta = 360 - \theta)$ 으로 정렬한 후, 단계 1을 수행한다( $K = L = \emptyset, symmetry = true$ ).
end

```

Function Feature_Distance($feature f_i, feature f_j$)

```

/*  $f_i$ : 질의 객체의 모양 특징
    $f_j$ : 데이터베이스에 저장된 객체의 모양 특징
    $\alpha$ : 가중치  $(0 \leq \alpha \leq 1)$  */
begin
단계 1. 두 특징간의 거리의 차를 구한다.
    즉,  $d_r = |r(f_i) - r(f_j)|$ 
단계 2. 두 특징간의 각도의 차를 구한다.
    즉,  $d_\theta = |\theta(f_i) - \theta(f_j)|$ 
단계 3. 두 특징간의 유사도 거리를 구한다.
    즉,  $d_{ij} = \alpha d_r + (1 - \alpha) d_\theta$ 
end

```

그림 4. Algorithm Shape_Similarity

이 검색 알고리즘은 먼저 각 객체의 시작점을 기준으로 각각의 특징들의 거리를 계산한다. 이 때 함수 $Feature_Distance$ 는 삼각 불일치를 만족하기 때문에 특징 공간 (r_k, θ_k) 은 행렬 공간(metric space)이다.

단계 1은 두 개의 객체의 모양 A, B 에 대하여 모든 특징

들의 거리를 계산한다. 단계 2는 A의 모양 특징 개수가 B의 모양 특징 개수보다 크다면 행렬 d_{ij} 의 j열을 기준으로 할당되지 않은 i번째 행의 최소 값을 구한다. 즉, i, j 는 각각 집합 K, L내에 존재하지 않아야 한다. 그리고 i, j 를 각각 집합 K, L에 삽입한다. 단계 3은 A의 모양 특징 개수가 B의 모양 특징 개수보다 작거나 같다면 행렬 d_{ij} 의 i행을 기준으로 할당되지 않은 j번째 열의 최소 값을 구한다. i, j 를 각각 집합 K, L에 삽입한다. 단계 4는 각 객체가 시작점을 기준으로 시계 방향으로 정렬되어 있거나 반 시계 방향으로 정렬(거울 반사된 모양)되어 있다면 단계 5를 수행한다. 그렇지 않다면 알고리즘을 종료한다. 단계 5는 모든 d_{ij} ($1 \leq i \leq w$)에 대하여 지역 가중치 (δ)보다 작거나 같은 d_{ij} 의 개수를 구한다. 이 때 가중치 내에 존재하는 개수 k가 A의 모양 특징 개수 w의 70%이상이라면 B를 유사 후보로 지정하고 유사도 함수 $d(A, B)$ 를 적용한다. 그렇지 않다면 두 객체가 유사하지 않다고 판정하고 알고리즘을 종료한다. 객체간의 유사성 비교를 위해 유사도 함수 $d(A, B)$ 를 다음과 같이 정의하였다.

$$d(A, B) = \lambda \cdot \frac{\sum_{i=1}^k d_i}{k}, (\lambda = 1 + (w - k)/w) \quad (식 4.1)$$

λ 는 모양의 전역 가중치로서 같은 모양의 유사도 비교시 $k = w$ 이므로 $\lambda = 1$ 이 되며 $k = w \cdot \phi$ 일 경우 $\lambda = 1.3$ 이 된다. 즉, λ 는 $1 \leq \lambda \leq 1.3$ 의 범위 내에 존재한다.

단계 6은 객체의 유사성 검색에서 false drop을 줄이기 위한 단계이며 거울 반사된 모양을 기준으로 유사성 검색을 처음부터 진행한다.

본 논문에서 제안한 유사성 검색 알고리즘은 객체 모양의 크기 변화, 위치 변화, 회전시 항상 일정한 값을 갖는 지역적인 특징을 이용하기 때문에 질의 객체에 대한 전체 검색 및 부분 검색을 지원한다.

5. 실험과 분석

지금까지 앞 절에서 설명한 검색 알고리즘이 두 객체의 유사성 검색시 부분 검색과 전체 검색을 적절하게 지원하는지를 확인하기 위해 200개의 다양한 모양을 가진 객체에 대하여 실험 및 분석을 하였다.

이미지 객체의 유사성 검색을 실험 및 분석을 하기 위한 시스템을 펜티엄 III(450MHz) 윈도우 NT 4.0의 PC에서 JDK 1.3을 사용하여 구현하였다.

그림 5에서는 질의 객체(a), 이미지 객체 1(b)와 이미지 객체 2(c)에 대하여 제각기 모양 특징의 개수는 10, 8, 15개이며 이 특징들을 이용하여 각 객체간의 유사도 거리를 측정하였다. 이 때 적용된 지역 가중치 (α)와 전역 가중치 (δ)를 각각 0.2, 10으로 하였다.



(a) 질의 객체 (b) 이미지 객체 1 (c) 이미지 객체 2

그림 5. 유사성 검색 비교 분석

표 1에서는 Feature_Distance 함수를 적용함으로써 그림 5a의 질의 객체와 그림 5b의 이미지 객체 1 사이의 각 특징간의 최소 거리를 보여준다. 이 때 질의 객체의 모양 특징은 표 1의 행에 해당하며 이미지 객체 1은 표 1의 열에 해당된다. 질의 객체의 시작점을 기준으로 6개의 모양 특징에 대하여 특징 비교 최소 값이 시계 방향으로 분포되어 있다. 또한, 그림 5a의 질의 객체와 그림 5c의 이미지 객체 2사이의 각 특징 비교에서는 질의 객체의 시작점을 기준으로 9개의 모양 특징에 대하여 시계 방향으로 분포된다.

	1	2	3	4	5	6	7	8
1	0	82.489	113.613	128.936	171.215	192.929	245.621	264.145
2	30.511	52.022	83.298	98.424	140.895	162.418	215.29	233.634
3	63.233	19.256	50.42	65.702	108.017	129.696	182.412	200.912
4	81.7	0.811	32.087	47.235	89.684	111.229	164.079	182.445
5	119.214	36.875	5.6	9.722	52.001	73.715	126.407	144.931
6	139.334	56.844	25.721	10.401	32.118	53.604	106.513	124.811
7	176.003	93.686	62.409	47.332	4.812	16.926	69.617	88.142
8	196.917	114.427	83.304	68.018	25.701	4.012	48.896	67.228
9	245.611	163.278	132.002	116.924	74.404	52.918	0.000	18.534
10	264.143	181.653	150.529	135.207	92.927	71.213	18.521	0.002

표 1. 질의 객체와 이미지 객체 1 사이의 각 특징간의 거리

질의 객체와 이미지 객체 1의 유사도 거리는 다음과 같다. 7개의 모양 특징간의 최소거리의 합은 15.246, $k = 7$, $\lambda = 1.3$ 이다. 따라서 $d(A, B) = 2.831$ 이다. 질의 객체와 이미지 객체 2의 유사도 거리에 대하여 10개의 모양 특징간의 최소거리의 합은 5.135, $k = 10$, $\lambda = 1$ 이며 $d(A, B) = 0.513$ 이다. 따라서 제안 검색 알고리즘은 질의 객체에 대하여 이미지 객체 1보다 이미지 객체 2가 더 유사함을 나타내고 있다.

6. 결론

본 논문에서는 이미지 객체의 모양 특징을 이용한 유사성 검색 알고리즘을 제안하였다. 이 검색 알고리즘은 이미지 질의에 대하여 부분 검색과 전체 검색을 모두 지원한다.

향후 연구 과제로 효율적인 유사성 검색을 위해 객체의 모양의 지역 특징인 w개의 (r, θ) 값에 적합한 효율적인 인덱스 구조의 구축을 위한 연구가 필요하다.

참고논문

- [1] 김영태, 엄기현, "이미지 DB의 인덱싱을 위한 객체의 모양 특징 표현 방안", Korean DataBase Conference, 2001.
- [2] Stefano Berretti, Alberto Del Bimbo, Pietro Pala, "Retrieval by Shape Similarity with PerceptualDistance and Effective Indexing", IEEE Transactions on Multimedia, Vol.2, No.4, pp.225-239, 2000.
- [3] J. Wang, W. Chang, and R. Acharya, "Efficient and Effective Similar Shape Retrieval", IEEE Multimedia Computing and Systems, pp.875-879, 1999.
- [4] Maytham Safar, Cyrus Shahabi, "Two Optimization Techniques to Improve the Performance of MBC-based Shape Retrieval", Multimedia Information System 2000.
- [5] C. Faloutsos, M. Flickner, W. Niblack, D. Petkovic, W. Equitz, and R. Barber, "The Qbic Project: Efficient and Effective Querying by Image Content," IBM Res. Div. Almaden Res. Center, Res. Rep. 9453, Aug. 1993.
- [6] W.I. Grosky, P. Neo, R. Mehrotra, "A pictorial index mechanism for model-based matching", Data Engineering, Fifth International Conference, pp.180-187, 1989.