

# 이미지 데이터베이스에서의 응답 시간 향상을 위한 그리드 기반 매칭 기법

남윤영<sup>○</sup> 박진규<sup>\*\*</sup> 황인준<sup>\*\*</sup> 위영철<sup>\*</sup> 김동윤<sup>\*</sup>  
아주대학교 정보통신전문대학원<sup>\*</sup>  
고려대학교 전자컴퓨터공학과<sup>\*\*</sup>

youngman<sup>○</sup>@ajou.ac.kr<sup>\*</sup>, saanin@mil.korea.ac.kr<sup>\*\*</sup>, ehwang04@korea.ac.kr<sup>\*\*</sup>, ycwee@ajou.ac.kr<sup>\*</sup>,  
dykim@ajou.ac.kr

## A Grid-based Matching Algorithm for Improving Response Time in Image Database

Yunyoung Nam<sup>○</sup> Jinkyu Park<sup>\*\*</sup> Eenjun Hwang<sup>\*\*</sup> Youngcheul Wee<sup>\*</sup> Dongyoon Kim<sup>\*</sup>  
Graduate School of Information and Communication, Ajou University<sup>\*</sup>  
Department of Electronics and Computer Engineering, Korea University<sup>\*\*</sup>

### 요 약

내용기반의 이미지 검색방법은 객체의 내부의 정보를 이용한 검색방법으로 색상, 모양, 질감과 같은 특징을 사용한다. 이러한 특징 중에 모양은 검색에 사용될 수 있는 점을 추출하여 유사도 계산에 사용한다. 유사도 계산은 점의 개수가 증가할수록 검색의 응답시간도 함께 증가한다는 문제점이 있다. 본 논문은 응답시간 향상을 위하여 특징점들에 대한 그리드 기반의 유사도 매칭 기법을 제안한다. 그리드 기반의 유사도 매칭 기법은 점들을 그리드로 나누어 검색의 범위를 점함으로써 매칭하는 횟수를 줄이는 방법이다. 특징점으로 사용된 점들은 이미지의 선으로부터 MPP(Minimum Perimeter Polygons) 알고리즘으로 추출하였으며, 특징 점들간의 거리값의 합을 유사도로 계산하였다. 실험에서는 400여개의 식물 잎 이미지로부터 점들을 추출하여 검색 시간을 비교하였다.

이 방법은 점의 개수가 많을수록 더욱 효과적이며, 점을 추출하는데 사용한 MPP알고리즘에 그리드 검색을 적용하기가 용이하다는 장점이 있다.

### 1. 서 론

이미지와 비디오 같은 멀티미디어 데이터는 다양하고 많은 정보를 담고 있다. 이미지의 경우 색상, 질감, 외곽선, 객체들간의 관계 등 여러 가지 특징정보를 가지고 있다. 검색시간은 사용되는 데이터의 양에 비례하기 때문에 이러한 다양한 정보를 이용한 검색은 많은 시간이 소요되며, 검색에서의 응답시간을 줄이기 위해 지금까지 여러 가지 인덱싱 기법과 매칭 기법이 개발되었다.

본 논문에서는 이미지 검색에서 검색에서 소요되는 시간을 줄이기 위하여 그리드 기반 검색 기법을 제안한다. 질의 이미지와 데이터베이스에 저장된 이미지가 있을 때 각 이미지에서 추출한 점들은 2차원의 형태로 x좌표와 y좌표 값으로 표현하여 관계형 데이터베이스에 저장되어 있다. 질의 이미지에서 추출한 점들의 개수를  $n$ 이라고 하고 데이터베이스에서 저장되어 있는 이미지로부터 추출한 점들의 개수를  $m$ 이라고 하면, 어떠한 인덱싱 기법이나 매칭 기법을 사용하지 않을 때 매칭의 시간 복잡도는  $O(nm)$ 이다. 이러한 매칭 횟수를 줄이기 위해 점들의 좌표를 그리드로 나누어 각 셀의 위치정보를 인덱싱으로 이용하여 배열에 적용하면 이상적인 경우, 한 개의 점에서 매칭하는 횟수는 최대 9번을 넘지 않는다.

### 2. 관련 연구

매칭은 사용자가 질의한 이미지(U)와 데이터베이스의 이미지(V)의 우세점을 추출한 후, 얼마나 일치하는가에 대한 유사도(similarity)를 측정하게 된다. 유사도를 측정하는 방법 중에 가장 널리 사용하는 방법은 아래의 식 1인 Lp Distance (Minkowski Distance)이다. 이 식에서  $p$ 의 값이 1이면 맨하탄 거리(Manhattan Distance)이고,  $p$ 의 값이 2이면 유클리드 거리(Euclidean distance)이며,  $p$ 의 3이면 Max Matrix가 된다.

$$D(U, V) = \left( \sum_{i=1}^k |u_i - v_i|^p \right)^{1/p} \dots \dots \dots (1)$$

※  $u_i, v_i$  : feature vectors,

$k$  : the number of dimension

이외에도 최소 가중치 거리와 최소 편차 거리를 이용하는 방법이고 1:1 대응 함수에서만 사용되는 Bottleneck Distance 함수가 있으며, 1:1 대응이 존재하지 않은 다른 크기의 두 점 집합에 사용되는 Hausdorff Distance가 있다.

※ 본 연구는 과학기술부 국책연구 개발 사업인 유전자원지원 활용사업단의 연구비 (no. BDM0100211)의 지원에 의해 수행되었습니다.

본 논문에서는 항수식 2인 유클리드 거리를 사용한다.

$$D(U, V) = \sqrt{\sum_{i=1}^k (u_i - v_i)^2} \dots \dots \dots (2)$$

3 이미지 매칭과 추출

3.1 MPP 알고리즘

이미지에서 MPP를 찾기 위해서는 우선 그 이미지를 셀(Cell)로 쪼개어 좀 더 단순화된 형태의 매트릭스(Matrix)로 만든다. 이렇게 단순화된 매트릭스에서 특정한 각도 이상의 점들을 찾아서 볼록점(convex)과 오목점(concave)들을 찾은 후, 이 점들을 순회(travel)하면서 점들의 오목하고 볼록한 특징을 통해 이미지의 형태를 단순화한다. 이때, 이미지의 형태를 결정하는데 중요하지 않은 점들은 제거하며, 단순화된 형태의 매트릭스에서 (X,Y)좌표의 집합이 바로 MPP가 된다.

식물의 잎 모양 이미지에서, 식물의 잎은 잎맥을 제외한 잎의 외곽선이 시작점과 끝점이 서로 연결되어있는 폐쇄형(closed-loop)이므로, 식물의 잎 모양 이미지에 대해 MPP 알고리즘을 적용할 수 있다.

MPP 알고리즘은 MATLAB[5]으로 구현하였으며, 아래 표 1의 작업 순서대로 점들의 시퀀스(sequence)를 찾는다.

표 1 MPP에 의해 시퀀스를 추출하는 단계

1. 셀로 쪼개어진 간단한 형태의 매트릭스로 변환한다.
2. 1에서 구한 매트릭스에서 폐쇄형 내부 영역을 구한다.
3. 내부 영역의 경계를 구한다. 이것은 4-connected sequence의 형태이다.
4. 경계에서 프리먼 체인 코드(Freeman Chain Code)를 구한다.
5. 체인코드에서 볼록점(convex)과 오목점(concave)을 구한다.
6. 볼록점들을 연결하여 초기 다각형을 구하고, 다각형의 외부에 있는 오목점은 모두 제거한다.
7. 제거 후 남은 볼록점과 오목점을 연결하여 다각형을 만든다.
8. 연결된 다각형에서 볼록점 중에 실제로는 오목점들을 제거한다.
9. 7번과 8번을 더 이상의 변화가 없을 때 까지 반복하며 점의 각도가 180도인 것들도 제거한다.
10. 최종적으로 남은 점들이 MPP의 시퀀스이다.

이러한 MPP 알고리즘을 식물의 잎 이미지에 적용하면 그림 1과 같은 이미지를 얻을 수 있다.

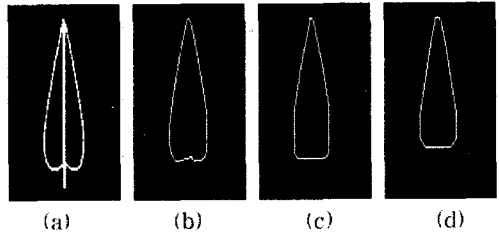
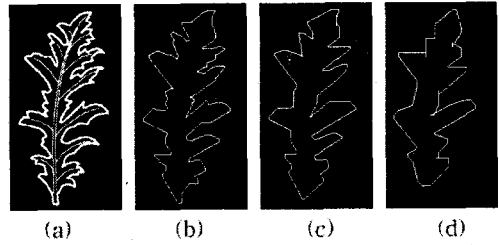


그림 1 식물의 잎에 적용한 MPP 알고리즘의 예  
(a) 원본이미지 (b) Cell Size=2  
(c) Cell Size=3 (d) Cell size=5



3.2 우세점 추출

MPP 알고리즘은 점과 점 사이를 연결하는 선을 기준으로 볼록한 점(180° 이하)과 오목한 점(180° 이상)으로만 점들을 찾기 때문에 추출한 점들 중에는 불필요한 점들이 많이 포함되어 있다. 이러한 점들을 간결화하기 위해 각 점들의 각도를 계산하여 임계값의 범위를 벗어나는 점들은 병합하였다. 아래의 표 2는 이러한 점들의 병합 여부를 판단하는 알고리즘이다.

표 2 불필요한 점들을 병합하는 알고리즘

```

Input
point : 점들의 (X, Y)좌표
N : 점들의 개수
threshold : 불필요한 점들의 기준 각도

find_sequence(point, N, threshold)
{
for ( i=0; i<N; i++)
{
a=get_distance(point[i-1], point[i+1])
b=get_distance(point[i], point[i-1])
c=get_distance(point[i], point[i+1])
angle=acos( ( b^2 + c^2 - a^2 ) /
( 2*b*c ) )
if(angle < threshold)
add_point(result, point[i])
}
return result
}
    
```

표 3은 임계값이 160°일 때, 그림 1-(d)의 MPP에서 불필요한 점들을 병합하여 추출한 결과를 보여주고 있다.

표 3 개선된 MPP 알고리즘을 이용한 예제

i	X[i]	Y[i]	각도(°)	병합여부
0	51	16	108	
1	55	16	108	
2	60	31	176	병합
3	65	51	180	병합
4	70	71	166	병합
5	70	85	135	
6	65	90	135	
7	41	90	135	
8	36	85	135	
9	36	71	166	병합
10	41	51	180	병합
11	46	31	176	병합

3.3 이미지 매칭과 추출

이미지 표현 알고리즘을 통해 시퀀스를 추출하여 이미지 인덱싱이 이루어지면 사용자가 질의한 이미지가 데이터베이스의 이미지와 얼마나 일치하는지를 계산한다. 이러한 유사도 계산을 통하여 결과값을 얻어낸 후, 가장 비슷한 이미지부터 순서대로 보여준다.

유사도를 계산하기 위해 식 2의 유클리드 거리를 사용하였으며, 아래의 식 5-7과 같이 질의한 이미지의 점들(U)와 데이터베이스에 저장된 이미지의 점들(V)과의 거리에서 최소값을 가지는 거리를 이용하여 계산하였다.

$$U = \{u_1, \dots, u_n\} \text{ with } u_i = (u_x^i, u_y^i) \dots\dots (3)$$

$$V = \{v_1, \dots, v_m\} \text{ with } v_j = (v_x^j, v_y^j) \dots\dots (4)$$

$D_{ij}(U, V)$ 는 두 점들간의 거리이다.

$$D_{ij}(U, V) = \sqrt{(u_x^i - v_x^j)^2 + (u_y^i - v_y^j)^2} \dots\dots (5)$$

$D_i(U, V)$ 는 두 점들간의 최소의 거리이다.

$$D_i(U, V) = \min\{D_{i1}(U, V), D_{i2}(U, V), \dots, D_{im}(U, V)\} \dots\dots (6)$$

$$S(U, V) = \sum_{i=1}^n D_i(U, V) \dots\dots (7)$$

4. 실험결과

본 논문에서 제안하는 알고리즘의 성능을 분석하기 위해서 모양 기반의 검색 시스템을 개발하였다. 검색 시스템은 인텔 펜티엄 CPU 1.8GHz, 256M RAM의 PC에서 실험하였으며, 데이터베이스는 Microsoft SQL Server

2000을 사용하였다. 스케치 입력은 애플릿으로 구현하였고 매칭 및 검색은 C언어로 구현하였다.

실험에서는 데이터베이스에 저장된 422개의 원 스케치 이미지를 사용하였고, 이미지로부터 추출한 점의 수는 26,455개이다.

질의는 가로와 세로의 크기가 500픽셀인 이미지 5개를 5번 질의하여 검색에 소요된 시간과 매칭횟수를 계산하였다. 이때 시간은 100번 수행하여 평균값으로 계산하였다. 질의 이미지의 점의 수는 Query1은 37개, Query2는 25개, Query3은 23개, Query4는 11개, Query5는 12개이다.

표 4와 표 5는 매칭횟수와 검색시간에 대한 분석결과이다. 표에서도 알 수 있듯이 매칭의 횟수는 약 13%로 크게 줄었다. 매칭의 횟수에 비해 검색시간이 약 2배 줄인 것은 인접한 셀에서 더 가까운 점이 있는지 검색하고 확인하는 시간 때문이다.

표 4 매칭횟수

질의	simple matching(A)	Grid matching(B)	감소비율(%) (B/A)
Query1	978,835	125,010	12.77
Query2	661,375	86,383	13.06
Query3	608,465	80,545	13.24
Query4	291,005	38,941	13.38
Query5	317,460	42,496	13.39

표 5 검색시간 단위:msec

질의	simple matching(A)	Grid matching(B)	성능향상 비율(A/B)
Query1	70.060	31.259	2.24
Query2	47.428	26.701	1.78
Query3	43.103	24.306	1.77
Query4	20.749	12.220	1.70
Query5	22.506	11.896	1.89

그림 2-4는 그리드 크기별 검색시간을 비교분석한 것이다. 둥근 점으로 된 선이 brute-force이고, 사각형 점으로 된 선이 그리드 기반 검색이다. 삼각형 점으로 된 선은 데이터베이스의 저장된 데이터를 하나의 프레임이 모두 표시한 후 검색한 방법이다.

그림에서 알 수 있듯이, brute-force 방법보다 검색 시간을 줄일 수 있었다. 크기에 따라 검색시간이 다르게 나타나지만 모두 공통적으로 그리드 기반의 검색이 검색 시간이 가장 짧았다.

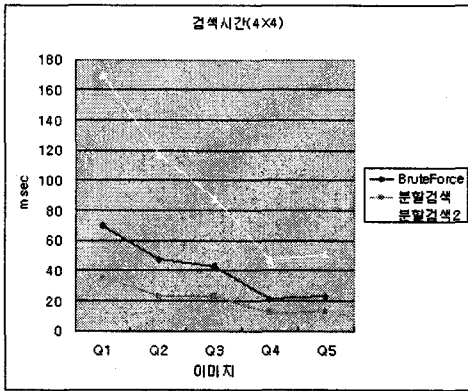


그림 2 매칭기법에 따른 검색시간결과  
(그리드 사이즈 : 4x4)

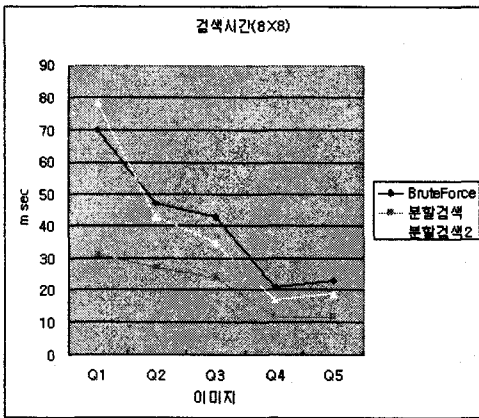


그림 3 매칭기법에 따른 검색시간결과  
(그리드 사이즈 : 8x8)

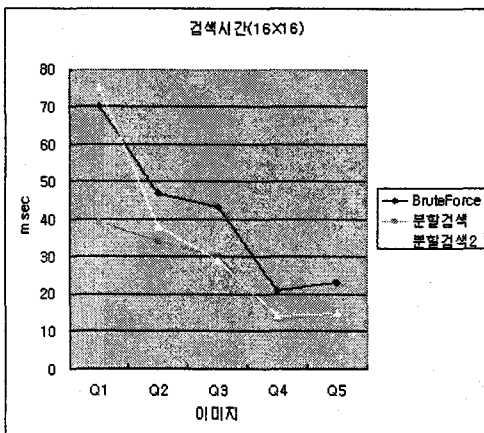


그림 4 매칭기법에 따른 검색시간결과  
(그리드 사이즈 : 16x16)

### 5. 결론 및 향후 연구 과제

본 논문에서는 내용기반 이미지 검색에서 대용량 특징정보로 인해 매칭시간이 기하급수적으로 증가하는 것을 줄이기 위하여 그리드 기반 검색 기법을 제안하였다. 일반적인 brute-force 알고리즘은 매칭에 이용되는 데이터의 수와 함께 비례하지만, 그리드 기반 검색 기법은 그리드 단위로 나누어 인덱싱하고 매칭의 범위를 줄임으로써 매칭횟수는 13%로 줄었으며 검색시간은 2배정도 단축시킬 수 있었다. 특히, 이미지의 특징이 다양하고 복잡한 경우에 검색시간을 줄이는데 더욱 효과적이며, 셀 크기단위로 우세점을 추출하는 MPP알고리즘에 그리드 검색을 적용함으로써 기존의 검색방법보다 검색의 성능을 향상시킬 수 있었다.

향후 연구로는 매칭횟수의 감소비율에 비해 검색의 시간이 증가하게 한 원인이었던 인접 셀에 더 가까운 점이 있는지 검색하는 시간을 줄일 수 있는 방법을 고안하는 것이다.

### 참고 문헌

- [1] Loncaic, S., "A survey of shape analysis techniques," Pattern Recognition, Vol.31, No.8, pp.983-1001, 1998.
- [2] Petrakis, E., Diptaros, A. and Milios, E., "Matching and Retrieval of Distorted and Occluded Shapes Using Dynamic Programming", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.24, No.11, pp.1501-1516, 2002.
- [3] Sklansky, Chazin et al. "Minimum perimeter polygons of digitized silhouettes," 1972.
- [4] Veltkamp, R., "Shape matching: similarity measures and algorithms," Technical Report UU-CS-2001-03, the Netherlands, 2001.
- [5] The MathWorks - MATLAB and Simulink for Technical Computing <http://www.mathworks.com>
- [6] 이창복, 대한식물도감, 향문사, 서울, 1982.
- [7] Y. Nam and E. Hwang, "A Shape-based Retrieval Scheme for Leaf Images," Lecture Notes in Computer Science, Springer-Verlag, Vol. 3767, pp. 876-887 Nov, 2005.
- [8] Y. Nam, E. Hwang, and K. Byeon, "ELIS: An Efficient Leaf Image Retrieval System," Lecture Notes in Computer Science, Springer-Verlag, Vol. 3687, pp. 589-597, Aug, 2005.
- [9] 남윤영, 황인준, "모양기반 식물 잎 이미지 검색을 위한 표현 및 매칭 기법," 한국정보과학회 논문지, 제32권, 제11호, pp.1013-1020, Nov. 2005.