

모양 기반의 식물 잎 이미지 인덱싱

남윤영^o 손정민^{*} 황인준^{**}

아주대학교 정보통신공학과^{*}, 고려대학교 전자컴퓨터공학과^{**}
 {youngman^o, fogony}@ajou.ac.kr^{*}, ehwang04@korea.ac.kr^{**}

Shape-based Leaf Image Indexing

Yunyoung Nam^o Jungmin Son^{*} Eenjun Hwang^{**}

Graduate School of Information and Communication, Ajou University^{*}
 Department of Electronics and Computer Engineering, Korea University^{**}

요 약

최근, 디지털 카메라와 디지털 캠코더처럼 디지털 장비가 대중화됨에 따라, 이미지 데이터가 급증하게 되었다. 이와 함께, 이미지 검색에 대한 요구도 증가하게 되었으며, 단순한 텍스트 검색이 아닌 이미지의 특징에 기반한 검색이 요구되고 있다. 특징 기반의 검색은 색상, 질감, 모양 등과 같은 특성에 기반한 검색으로 사람이 일일이 주석을 입력하는 방식보다 자동화가 가능하며, 빠르게 인덱싱할 수 있는 장점이 있다. 본 연구에서는 모양을 이용하여 이미지를 인덱싱 하였으며, 스케치된 식물의 잎 모양의 이미지를 이용하였다. 또한, 식물의 잎에 뻗어있는 잎맥의 모양을 이용하여 검색의 정확도를 높였다.

1. 서 론

식물의 잎을 관찰하여 동정(同定)이나 식물명을 결정할 때, 식물의 색상이나 질감보다는 잎의 모양을 이용하는 것이 그 식물을 구별하는데 더욱 효과적이다. 또한, 잎 모양 뿐만 아니라, 잎몸 안에 평행선이나 그물 모양으로 뻗어 있는 잎맥의 모양을 이용하면 검색의 정확성을 높일 수 있다.

따라서, 이미지의 검색 방법 중에 모양(shape)을 이용한 검색은 객체를 구별하는 데에 있어서 효과적인 검색의 방법중에 하나이다. 모양을 표현하는 방법은 외부 윤곽선을 이용하는 방법과 내부 골격(skeleton)을 이용하는 방법으로 나눌 수 있다. 외부 윤곽선은 곡률(curvature), 중심거리, 복소수 좌표 함수등을 통하여 추출해 내며, 푸리에 변환을 거쳐 특징으로 이용된다. 내부 골격은 평면 영역의 구조적 형태를 그래프로 변환시켜 만든다.

본 연구에서는 식물의 잎 모양을 기반으로 이미지를 인덱싱하였다. 사용된 이미지는 스케치된 식물의 잎 이미지며, 잎 모양뿐만 아니라, 잎에 뻗어있는 잎맥을 인덱싱에 고려하여 검색의 정확도를 높였다.

2. 관련연구

2.1 내용기반 검색

이미지 검색 기법에는 특징 기반 검색, 유사 검색, 공간 관계 검색, 의미 검색 등으로 나누어 볼 수 있다. 특

징 기반 검색은 이미지의 색상, 질감(texture), 모양, 윤곽선 등과 같은 이미지의 특성(feature)에 기반한 검색 방법으로써 이러한 정보는 자동 및 수동으로 생성될 수 있으며 데이터베이스에 이미지 데이터를 저장할 때 함께 저장되어 검색 시 이용된다. 유사 검색은 사용자가 예제 이미지를 주거나 개략적인 정보를 주면 그와 가장 유사한 이미지 데이터를 데이터베이스에서 검색하는 기법이다. 이 검색은 질의에 정확히 일치하는 하나의 결과를 찾아주는 것이 아니라 질의의 유사 범위 내에 속하는 결과 집합을 찾아주게 된다. 공간 관계 검색은 이미지 내에 존재하는 객체간의 공간 관계를 이용한 질의이다. 의미 검색은 이미지에 존재하는 객체에 의미 레이블을 부여해서 검색 시 이를 이용하여 결과를 찾아주는 기법이다.

2.2 이미지 표상(Representation)

데이터를 간결하게 표현하는 것은 검색의 효율성, 정확성, 계산비용 면에서 중요한 요인이 되며, 이러한 표현법은 지금까지 많은 연구에서 제안되었다. 시작점에서의 길이와 방향을 이용한 체인코드(chain code)[1]는 잡음에 민감하며, 임의의 각도에 대해서는 불가능하다. 푸리에 변환(Fourier transform)[2]은 윤곽선 기반의 검색방법 중 많이 사용되며, 푸리에 변환을 통해 생성된 복소수 형태의 상관계수들을 모양 기술자(descriptor)로 이용한다. 다각형 근사(polygonal approximation)[3]는 폐곡선(closed curve)에 대해서 객체 모양의 지역 특징인 고정된 차수의 우세점(interest point)들을 추출하여 인접한 우세점 간의 각도, 거리, 좌표들을 산출하는 방법이다. 이 방법은 객체의 모양을 정확하게 표현하고 신속하게 찾기 위한 우세점을 추출하는 알고리즘이 중요하며, 본 연구에서는 MPP(Minimum Perimeter Polygon) 알고리즘[4,5]을 사용하였다. 또한, 영역기반의 표현 방법인 골격

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

화(skeletonizing)[4,5]는 평면 영역의 구조화된 모양을 그래프로 변환시킨다. 이러한 골격화는 MAT(medial axis transformation)[6]을 통해 만들어질 수 있으며, 영역(region) R과 경계(border) b가 있을 때, R안에 각점 p에 대해서 b의 가장 가까운 이웃을 찾는다. 만약 p가 하나 이상의 이웃을 갖게 되면, 이것을 중앙축(medial axis)이라 하며 골격(skeleton)이라고 한다.

3. 이미지 인택싱

이미지의 형태를 구조화된 표현으로 바꾸는 것은 윤곽선검색과 이미지 표상방법을 통해 이루어져야 한다. 본 연구에서는, 이미지에서 윤곽선을 찾아내는 방법으로 Canny Edge Detection 알고리즘[7]을 사용하였으며, 이미지 표상은 MPP를 사용하였다.

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 일정 각도 이상의 점들을 찾아서 볼록점(convex)과 오목점(concave)들을 찾은 후, 이 점들을 순회(travel)하면서 점들의 오목하고 볼록한 특성을 이용하여 이미지의 형태를 결정하는데 중요하지 않은 점들을 제거한다. 이렇게 단순화된 형태의 매트릭스에서 (X,Y)좌표의 집합이 바로 MPP가 된다.

인택싱 및 검색에 사용된 데이터는 식물의 잎 모양 이미지이며, 식물의 잎은 잎맥을 제외한 잎의 외곽선이 시작점과 끝점이 서로 연결되어있는 폐쇄형(closed loop)이므로, 식물의 잎 모양 이미지에 대해 MPP 알고리즘을 적용할 수 있다. 그림 1은 연구에 사용된 식물의 잎에 대한 스케치된 이미지이다.

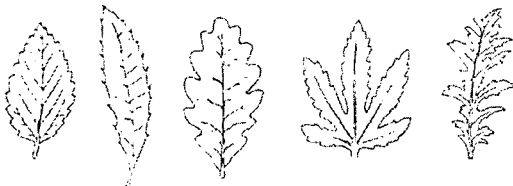


그림 1 다양한 식물의 잎

MPP 알고리즘은 MATLAB으로 구현하였으며, 다음 표 1의 작업 순서대로 MPP를 찾는다. 이러한 MPP 알고리즘을 식물의 잎 이미지에 적용하면 그림 2와 같은 이미지를 얻을 수 있다.

표 1 MPP를 찾기위한 작업 단계

1. 셀로 쪼개어진 간단한 형태의 매트릭스로 변환한다.
2. 1에서 구한 매트릭스에서 폐쇄형 내부 영역을 구한다.
3. 앞에서 구한 내부 영역의 경계를 구한다. 이것은 4-connected sequence의 형태이다.

4. 이렇게 구한 경계에서 프리먼 체인 코드(Freeman Chain Code)를 구한다.
5. 체인코드에서 볼록점과 오목점을 구한다.
6. 볼록점들을 연결하여 초기 다각형을 구하고 이 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 꼭지점이다.

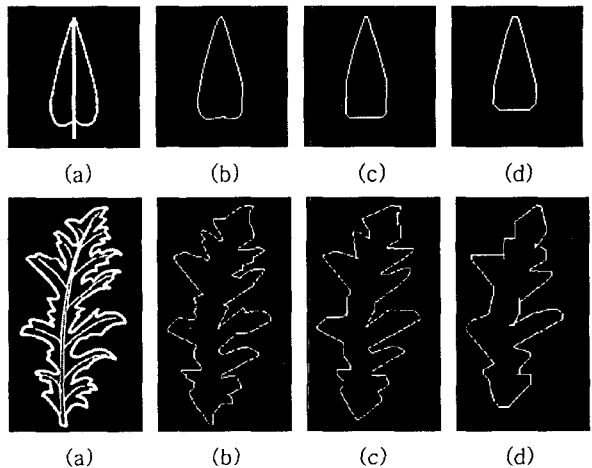


그림 2 식물의 잎에 적용한 MPP 알고리즘의 예

- (a) 원본이미지 (b) Cell Size=2
(c) Cell Size=3 (d) Cell size=5

3.2 유효한 점들의 추출

MPP 알고리즘은 점과 점 사이를 연결한 선을 기준으로 어떤 점이 볼록한 점(180도 이하)인지 오목한 점(180도 이상)인지의 정보만을 이용하기 때문에 점들의 각도 정보의 이용이 많지 않았다. 따라서, MPP를 이용하여 추출한 점들 중에는 불필요한 점들이 많이 포함되어 있다.

이러한 점들을 간결화 시키기 위해서 각 점들의 각도를 계산하여 임계값(threshold)의 범위를 벗어나는 점들은 병합시키는 기법을 사용하였다. 아래의 표 2는 이러한 각도 계산과 점의 병합 여부를 판단하는 알고리즘이다.

알고리즘을 도출하기 위해 $\triangle ABC$ 의 세 점 A, B, C의 각도를 구하기 위해 다음의 식 1을 사용하였다.

$$\cos A = \frac{b^2 + c^2 - a^2}{2bc} \dots \dots \dots (1)$$

$$(a = \overline{BC}, b = \overline{AC}, c = \overline{AB})$$

점 B의 좌표를 (x_1, y_1) 이라 하고, 점 C의 좌표를 (x_2, y_2) 이라 하면 \overline{BC} 의 길이 a는 다음과 같다.

$$a = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \dots \dots \dots (2)$$

표 2 불필요한 점들을 병합하는 알고리즘

```

Input : point - Sequence의 점들의 (X, Y)좌표
       N - Sequence에 있는 점의 개수
       threshold - 불필요한 점의 기준 각도

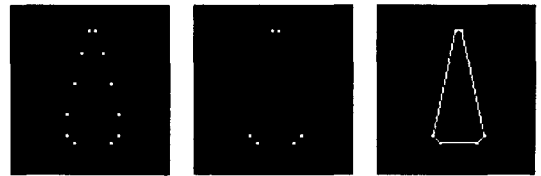
find_sequence(point, N, threshold)
{
  for ( i=0; i<N; i++)
  {
    a=get_distance(point[i-1], point[i+1])
    b=get_distance(point[i], point[i-1])
    c=get_distance(point[i], point[i+1])
    angle=acos( ( b^2 + c^2 - a^2 ) /
                ( 2*b*c ) )
    if(angle < threshold)
      add_point(result,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

표 3과 그림 3은 임계값이 160°일 때, 그림 2-(d)의 MPP에서 불필요한 점들을 병합하여 구한 결과를 보여주고 있다. 이러한 방식으로 외곽선 뿐만 아니라 잎맥의 특징점도 추출한다.

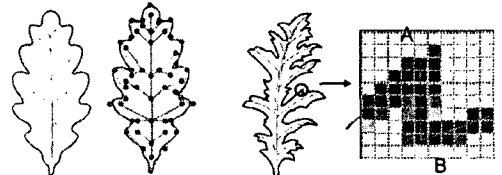
표 3 개선된 MPP 알고리즘을 이용한 예제

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

그림 4는 잎의 외곽선뿐만 아니라 잎맥에 대해서 특징점을 찾은 결과를 보여주고 있다. (a)는 외곽선과 잎맥의 특징점 시퀀스(sequence)이며, (b)는 윈도우 크기가 10x10에서 두 개의 특징점 A, B를 나타내고 있다.



(병합 전) (병합 후) (점 연결)
그림 3 개선된 MPP 알고리즘을 통해 추출한 시퀀스



(a) 잎맥의 그래프화 (b) 추출한 특징점 A, B (window size = 10x10)

그림 4 외곽선과 잎맥의 시퀀스

4. 결론 및 향후 연구 과제

본 연구에서는 식물의 잎 모양을 개선된 MPP 알고리즘을 사용하여 이미지의 외곽선을 간략히 표현하였으며, 검색의 정확성을 높이기 위해 잎 모양뿐만 아니라 잎맥을 이용하여 이미지 인덱싱을 하였다. 향후 연구 과제로는 좀 더 개선된 알고리즘을 개발하고, 웹 상에서 사용자로부터 입력받은 스케치를 통해 검색할 수 있도록 구현하는 것이다.

참고문헌

- [1] D.H. Ballard C.M. Brown, Computer Vision, Prentice-Hall, 1982
- [2] Cheng Chang, et al. "Image retrieval based on region shape similarity," Proceedings of SPIE Vol.4315, pp.31-38, 2001,
- [3] Kurozumi Y., Davis W.A., "Polygonal approximation by the minimax method," Computer Vision, Graphics and Image Processing, pp. 248-264, 1982.
- [4] Sklansky, Chazin et al. "Minimum perimeter polygons of digitized silhouettes," 1972
- [5] Sklansky J., "Finding the Convex Hull of a Simple Polygon", Pattern Recognition Letters, Vol.1(2), pp.79-84, 1982
- [6] Lee, D.T., "Medial axis transformation of a planar shape," IEEE Transactions On Pattern Analysis and Machine Intelligence, pp. 363-369, 1982
- [7] Michael Heath, et al., "A Robust Visual Method for Assessing the Relative Performance of Edge Detection Algorithms," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19(12), pp.1338-1359, 1997.
- [8] M. Egenhofer, "Spatial-Query-by-Sketch" VL'96: IEEE Symposium on Visual Languages, pp. 60-67, 1996.