

이미지 경계값을 이용한 영역 추출 방법

이승재, 김창화
강릉대학교 컴퓨터공학과
silverree@moira.co.kr, kch@kangnung.ac.kr

Region Extraction Methodology Using Edge Values of Image.

SeungJae Lee, ChangHwa Kim
Dept. of Computer Science, Kangnung National Univ.

요 약

본 논문에서는 내용기반 이미지 검색 시스템을 제작하기 위하여 필수적으로 선행되어야 하는 이미지의 영역구분에 대한 새로운 방법인 경계값을 이용한 영역추출 방법을 소개한다. 빠르고 정확한 이미지 검색엔진을 구현하기 위하여 질의의 결과가 될 이미지들은 전처리기에 의하여 모든 영역을 추출한 뒤 각각의 영역에 따른 특성(feature)을 저장하고 있어야 한다. 정확한 질의 결과를 얻기 위하여는 정확히 영역을 추출할 수 있고 그 특성도 추출할 수 있는 전처리를 사용하여야 한다. 또한 정확도만을 중시하여 너무 복잡한 알고리즘을 사용하면 그 또한 실용적이지 못하게 된다. 경계값을 이용한 영역추출 방법은 이미지의 각 점에 대한 경계값(edge value)을 이용하여 그 경계값이 작은 점으로부터 시작하여 경계값이 큰 점들을 병합해 가면서 인접한 영역간의 크기, 색상 등을 고려하여 각각의 영역을 구분해 낸다. 이 방법의 가장 큰 특징은 텍스처(texture)를 제외한 일반적인 영역뿐 아니라 텍스처 포함하는 영역도 추출할 수 있는 점과 빠른 처리 속도에 있다.

1. 서론

최근 쏟아지는 정보 홍수 속에서 각종 매체가 결합된 멀티미디어 검색기술에 관한 연구가 활발히 진행되고 있다. 이미지는 멀티미디어 데이터의 가장 기본적인 데이터 중 하나로, 하나 이상의 의미 있는 시각적 영역들의 집합으로 볼 수 있다. 따라서 많은 이미지 정보들을 효과적으로 데이터베이스를 구축하고 검색 가능하게 하기 위하여, 정확하고 빠른 영역 추출 기법이 요하게 된다.

영역은 그 구분 방법에 따라 여러 가지로 나눌 수 있지만 본 논문에서는 비슷한 색상 분포를 가지는 리전영역과 텍스처로 구성 되어 있는 텍스처영역 두 가지로 구분하고 이 두 가지를 통칭하여 영역이라고 부르기로 하였다.

주어진 이미지에서 리전(region)영역을 찾아내는 방법으로는 임계값에 의한 영역구분, region growing algorithm과 California 대학의 Netra[5]에서 사용된 Edge Flow[2, 5]와 같은 방법들이 이미 연구되어 있고 텍스처영역을 찾아내는 방법은 B.S. Manjunath's group의 UCSB 시스템[3]과 IBM의 QBIC[3]에서 사용되고 있으며 특히 Multiple Gabor filter[1, 6]를 이용한 방법은 상당히 만족스런 결과를 나타낸다.

하지만 이러한 방법들은 리전영역과 텍스처영역을 따

로 따로 구분해 내는 방법으로 리전영역과 텍스처영역을 동시에 추출해 내지는 못한다. 그리고 Gabor filter[7]를 이용한 방법이 비록 그 결과는 좋게 나오지만 많은 연산으로 인하여 처리속도 면에서는 상당히 성능이 떨어지게 된다.

본 논문에서는 Sobel Edge Detection에 근거하여 각 점마다 edge value를 구하고, 하나의 리전영역으로 구분될 수 있는 부분일수록 edge value의 값이 작다는 점을 이용하여 이 부분들을 병합하면서 리전영역을 구한 후 리전영역으로 포함되지 않은 부분을 대상으로 edge density, 색상유사도 등을 이용하여 텍스처영역을 구하는 과정을 설명한다.

먼저 2장에서는 영역구분에 관한 기존의 관련연구를 살펴보고 3장에서는 경계값을 이용한 영역구분방법을 자세히 살펴본 후 4장에서는 결론 및 앞으로의 연구 과제를 기술한다.

2. 관련연구

Buñing algorithm이라고도 불리는 region growing algorithm은 주어진 이미지에서 리전영역을 구분하는 방법으로 seed 점을 구한 후 주변 점들과의 동질성(homogeneity)을 검사하여 점점 넓혀 가면서 비슷한 영역끼리 다시 합하는 방법이다. 이 방법은 비교적 알고리

증이 간단하다는 장점이 있으나 초기에 씨드점의 개수를 결정할 때 색상 히스토그램(color histogram)에서 피크를 이루는 점의 개수를 이용하기 때문에 정확한 영역을 구분하기 어렵고 특히 텍스처영역은 구분해 내지 못하는 단점이 있다.

Netra에서 사용한 멀티 가보 필터를 이용한 텍스처 분리 방법은 샘플이 되는 여러 가지 텍스처의 특성을 저장한 후 multiple Gabor 필터를 사용하여 질의 이미지에 존재하는 텍스처와의 유사도를 이용하여 텍스처들을 구분해 내는 방법으로 그 결과가 다른 방법보다 상당히 좋게 나오나 많은 필터링으로 인한 계산 속도가 느리고 리전영역은 구분하지 못하며 텍스처영역만 구분 가능하다는 단점이 있다.

또 IBM의 QBIC과 ARBIRS[3]의 서브시스템으로 사용된 Texture Detection Subsystem에서 사용된 텍스처 추출 방법은 리전영역을 추출하지 못하고 텍스처영역만 추출할 수 있다. 하지만 추출된 텍스처영역도 서로 다른 텍스처를 구분하지는 못하고 한 이미지 내에서 텍스처 영역과 그렇지 않은 영역을 구분할 뿐이다.

3. 경계값을 이용한 영역 구분 방법

이미지에서 리전영역으로 구분할 수 있는 영역과 텍스처영역으로 구분할 수 있는 영역의 가장 큰 특징은 하나의 리전영역은 칼라의 변화가 상대적으로 작기 때문에 리전영역에 포함되어 있는 점들의 edge value 값은 대체로 작은 값을 가지게 되며 색상도 유사하다. 그러나 텍스처영역의 점들의 edge value 값의 분포가 매우 다양하고 색상도 다양한 분포를 가질 수 있다.

리전영역을 구할 때는 seed 영역으로부터 시작하여 특히 영역을 병합할 때 각 영역의 색상을 사용하는데 일반적인 RGB 값을 사용하지 않고 hue 값을 사용한다. RGB 값을 사용할 경우 아주 밝은 영역 혹은 아주 어두운 영역들이 서로 다른 색상으로 분리되에도 불구하고 RGB값은 비슷한 값을 가지게 되어 서로 다른 영역으로 구분되지 않고 병합되는 현상이 발생된다. 그러므로 밝기에 관계없이 색상이 다른 두 개의 영역을 구분하기 위하여 영역의 색상은 HSL 색상영역의 hue 값을 사용한다[4].

이러한 특성에 착안하여 작은 edge value 값을 이용하여 리전영역을 먼저 추출한 후 나머지 구역에서 텍스처영역을 추출하는 방법을 사용한다.

3.1 Edge value 계산

영역구분의 대상이 되는 이미지의 모든 점을 대상으로 <그림 1>과 같은 Sobel edge detector mask를 적용하여 edge value $E_v(x, y)$ 를 구한다.

$$\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

<그림 1>

3.2 Seed 영역추출

리전영역을 추출하기 위하여 먼저 이미지에서 seed 영역을 추출하여야 한다. 이 때 edge value 값이 0인 점들의 집합을 seed 영역으로 사용하는 방법도 가능하지만 이 경우 너무 많은 seed 영역이 만들어지고 리전영역을 넓혀가는 과정에서 병합조건이 까다로와 지면 많은 seed 영역으로 인해 계산 시간이 현저히 증가한다.

그러므로 edge value 값이 0인 점들을 사용하기보다 특정값 α 이하의 edge value 값을 가진 점들의 집합을 seed 영역으로 사용하면 앞의 문제들이 해결된다.

필자는 다양한 이미지를 재료로 하여 실험한 결과 그 래픽 툴로 작업하지 않은 자연 이미지들의 edge value 값이 플랑크분포를 이룬다는 사실을 발견하였다. 또 이 분포값들의 분산값에 로그를 취하면 해당 이미지의 seed 영역을 결정할 수 있는 edge value 값을 구할 수 있음을 발견하였다.

$$r = a \log(bv + c), \quad v: \text{분산} \quad (1)$$

이렇게 구해진 r 값은 이미지 전체 영역에서 seed 영역이 될 리전영역의 면적비가 된다. 즉 특정 값 α 이하의 edge value 값을 가지는 영역들의 전체 이미지에 대한 영역비가 r 이상이 되는 최소값을 α 로 하여 α 이하의 edge value 값을 가지는 점들의 집합을 seed 영역으로 삼는다.

3.3 Edge value에 의한 리전영역 추출 및 병합

이제 $0 < E_v \leq \alpha$ 인 모든 점을 취하여 인접한 점들끼리 병합하여 임계값 이하의 작은 영역들은 버리고 그 이상의 크기를 가진 영역들을 seed 리전으로 취하고 이들을 각각 독립적인 리전영역으로 간주한다.

이제 α 에서 d 만큼 증가시킨 값 이하의 edge value를 가지는 점들 찾아 하나의 영역으로 간주한 후 그 인접한 주위에 리전영역이 있으면 새로운 영역을 리전영역에 병합시키고 그렇지 않으면 버린다. 이때 새로운 영역이 두 개 이상의 리전영역과 인접할 경우 인접한 부분이 많은 영역으로 병합시킨다.

다음 과정으로는 다음의 식(2)에 의하여 위와 비슷한 과정을 반복한다.

$$a < E_v \leq b \quad (a = a, a + d, \dots, T_R - d, b = a + d) \quad (2)$$

Seed 영역을 구할 때 경우에 따라서는 하나의 리전영역으로 묶여야 할 영역이 두 개 이상으로 나뉘어 지는 현상이 발생할 수 있으므로 이 경우 병합 과정에서 다시 하나의 영역으로 만들어 주어야 한다.

두 개의 인접한 영역 R_i, R_j 가 다음 조건을 만족하면

하나의 리전영역으로 병합시킨다.

$$|R_{i_{hue}} - R_{j_{hue}}| < T_{Hue} \left(\frac{MAX(R_i, R_j)}{MIN(R_i, R_j)} \right) \quad (3)$$

R_{i_A}, R_{j_A} : 해당영역의 넓이
 $R_{i_{hue}}$: 해당영역의 hue 값

3.4 텍스처영역 추출

텍스처영역은 그 특성상 좁은 구간 내에서도 각 점들의 E_v 값이 많은 차이를 보인다. 이러한 이유로 3.3 절에서 다루었던 방법으로 텍스처영역을 보면 많은 영역들이 좁은 구역 내에 분포하는 것처럼 보이게 된다. 그러므로 3.3 절에서 다른 방법을 계속해서 적용시키면 텍스처영역들이 리전영역으로 병합되어 버리게 된다.

그러므로 텍스처영역을 추출할 때는 이미 구해진 리전영역은 완성된 리전으로 간주하고 더 이상 병합을 하지 않으면서 리전영역으로 구분되지 않은 구역만을 대상으로 계산되어야 한다.

이미 리전영역으로 구분된 구역을 제외한 나머지 구역, 즉 $E_v > T_R$ 를 만족하는 구역에 대한 E_v 의 분포 그래프를 그려 하위 피크점이 발생하는 E_v 값들을 찾는다.

$$E_p = \{E_i | i = 0, 1, 2, \dots, n\}, \quad E_0 = T_R \quad (4)$$

$E_i \sim E_{i-1}$ 까지의 E_v 값을 가지는 점들을 하나의 텍스처영역 R_i 로 간주하고 이 영역에 포함되면서 영역으로 구분되지 않은 구역은 모두 영역 R_i 에 병합시키면서 이러한 과정을 $E_n \sim E_{n-1}$ 까지 계속한다.

이제 마지막으로 텍스처리전으로 추출된 영역끼리 인접하면서 $|R_{i_{hue}} - R_{j_{hue}}| < T_{Hue}$ 인 텍스처영역들을 병합시킨다.

4. 결론 및 향후 연구과제

효율적인 이미지 검색 엔진을 제작하기 위해서는 검색 대상이 되는 이미지의 특성(feature)을 정확하게 추출할 수 있어야 한다.

그러나 하나의 이미지는 하나의 개체만을 포함하는 것이 아니라 여러 가지의 개체를 포함하기 때문에 각각을 구분할 수 있어야 한다. 가장 기본이 되는 구분 방법을 이미지를 각각의 영역으로 구분하는 방법이 될 것이다. 이미지를 영역으로 구분하여 각각의 영역에 대한 형태(shape), 칼라히스토그램(color histogram), 위치 정보(spatial information) 등의 특성을 추출할 수 있다. 이러한 특성을 추출하기 위해서는 주어진 이미지에서 정확하게 영역을 추출해 내는 알고리즘은 필수적이라

하겠다. 전술한 바와 같이 영역은 리전영역과 텍스처영역으로 구분지을 수 있으나 현재까지 많은 알고리즘들은 이 두 가지 영역을 모두 추출해 낼 수 없었다.

이에 지금까지 본 논문에서는 간단한 Sobel edge detector mask를 적용하여 리전영역에서의 경계값은 텍스처영역에서의 그것보다 상대적으로 작다는 점에 착안하여 경계값이 T_R 이하인 점을 대상으로 씨드영역을 구한 후 계속해서 병합해 나가면서 인접한 영역에 대해서는 그 상대적인 크기와 hue 값을 이용하여 병합해 리전영역을 추출하였고, 경계값이 T_R 이상인 점들을 대상으로 텍스처영역을 구하여 빠르면서도 리전영역과 텍스처영역을 동시에 추출할 수 있는 방법에 대하여 알아 보았다.

그러나 리전영역을 추출하기 위하여 한계값으로 사용한 T_R 의 경우 여러 가지 이미지를 대상으로 실험한 결과 최적값이 경미하기는 하지만 이미지에 따라서 다소의 차이를 보이고 있다. 또 무채색으로 이루어진 점들에 대하여 hue 값과 lightness 값을 동시에 고려해야 하는 복잡성을 제거하기 위해 이미지에서 무채색으로 된 모든 점들의 RGB를 전체 이미지의 특성을 변화시키지 않는 범위에서 매우 조금만 변화시키는 필터를 사용할 수도 있고, 두 개의 영역을 병합할 때 충분조건이 되어야 하는 식(2), (3)에서 영역의 최소 크기를 결정하는 T_A 와 병합할 영역간의 크기에 따라 hue 값의 허용 범위를 결정하는 함수 T_{Hue} 를 각각의 이미지에 따라 정확하게 찾아낼 수 있다면 이미지의 영역을 구분하는데 있어서 보다 더 만족한 결과를 얻을 수 있을 것이다.

5. 참고문헌

- [1] Thomas P. Weldon and William E. Higgins, "Integrated approach to texture segmentation using multiple Gabor filters", Proc ICIP-96, pp. 1-3, 1996.
- [2] W.Y. Ma and B.S. Manjunath, "Edge flow: A framework of boundary detection and image segmentation", IEEE Int. Conf. on Computer Vision and Pattern Recognition, pp. 1-5, 1997.
- [3] Yihong Gong, "Intelligent Image Databases:", Kluwer Academic Pub., pp. 19-30, 69-76, 1997.
- [4] Alberto Del Bimbo, "Visual Information Retrieval", Morgan Kaufmann Publishers, Inc., pp. 81-116, 1999.
- [5] W. Y. Ma and B. S. Manjunath, "NeTra: A toolbox for navigating large image databases", <http://vivaldi.ece.uscb.edu/Netra>, pp. 5-14, 1997.
- [6] Thomas P. Weldon and William E. Higgins, "An Algorithm for Designing Multiple Gabor Filters for Segmenting Multi-Textured Images", -, pp. 1-4, -.
- [7] B.S. Manjunath and W.Y. Ma, "Texture Features for Browsing and Retrieval of Image Data", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 18, No. 8, August 1996