

# 색상-공간정보를 고려한 내용기반 영상검색

정희원 장 정 동\*, 이 태 홍\*\*

## Content-based Image Retrieval Considering Color and Spatial Information

Jungdong Jang\*, Taihong Yi\*\* *Regular Members*

### 요 약

최근 정보통신기술의 발전과 함께 영상매체의 급속한 증가로 영상의 효율적인 관리와 검색을 수행하기 위한 내용기반 영상검색은 핵심기술로 대두되고 있다. 내용기반 영상검색에서 영상의 특징을 표현하기 위해 색상 히스토그램을 많이 사용하고 있으나, 색상만을 고려하는 것은 많은 단점을 지니고 있으므로 본 논문에서는 영상의 특징으로 색상과 공간정보를 함께 고려하기 위한 순차영역분할(sequential clustering)기법을 도입하며, 분할된 영역의 색상평균값, 분산값과 영역의 크기를 특성벡터로 제안한다. 제안된 방법의 특성의 수가 18개로 타 방법보다 매우 작은 저장공간을 가지면서도 검색효율이 8.8% 이상 개선되었다. Precision 대 Recall에서도 각 질의영상에서 대부분의 Recall값에서 제안한 방법이 우수함이 확인되었으며, 시각적으로도 양호한 검색결과를 얻을 수 있었다.

### ABSTRACT

Recent advances in information technology have made an explosive increase in the number of images. In order to manage and retrieve these images efficiently, content-based image retrieval of color image is appeared to be the core technology. Color histograms are widely used for content-based retrieval, but considering only color has many disadvantages. In this paper we considered color and spatial information of the features in the image and we proposed to use Sequential Clustering method for homogeneous property regions. Also we proposed to use color mean, variation and size of each divided regions for feature vector which has 18 features and of which storage space is smaller than conventional methods. However, our experimental results showed that retrieval efficiency not only improved 8.8 percents better than conventional methods but also saved much computation and data storage. In addition, this method showed excellent Precision vs. Recall evaluation and subjective fine judgment of human perception.

### I. 서 론

컴퓨터의 발달 특히, 최근 영상산업의 급속한 발전과 더불어 인터넷, 인트라넷 등의 대중화에 따라 문자이외에 음성, 영상 등의 양은 급속하게 증가하고 있다. 그러나 영상자료는 기존의 문자 자료에 비하여 용량이 클 뿐 아니라 비정형적인 특성을 가지고 있으며, 포함하고 있는 내용이 복잡하여 기존의

문자기반 데이터베이스로는 신속하고 효율적인 검색과 효과적인 관리에 커다란 어려움이 있다.

최근, 이러한 문제점을 해결하고자 다양한 영상검색기법이 연구되고 있으며, 이러한 영상검색은 사용자의 간단한 명령으로 원하는 영상자료를 신속하게 찾아 볼 수 있도록 하는데 초점을 두고 있다. 영상자료의 효율적인 검색을 수행하기 위한 접근방법은 세 가지로 분류할 수 있다. 즉, 문자기반 접근방법, 의미

\* 한국전력공사 전자통신처(jangjd@kepeco.co.kr)

\*\* 영남대학교대학원 전자공학과 영상통신연구실(tachong@yu.ac.kr)  
논문번호 : K01012-0108, 접수일자 : 2001년 1월 8일

론적 접근방법과 내용기반 접근방법으로 나눈다<sup>[1]</sup>.

문자기반 접근방법은 영상의 내용에 따라 주제어 또는 주석을 사용자가 부여해야 하므로 사용자에게 따라 한 영상에 대한 주제어 또는 주석이 다를 수 있으며, 주제어를 설정할 때 개인의 주관에 개입될 수 있다는 단점이 있다. 의미론적 접근방법은 영상을 추상화된 특징들로 표현하는 방법으로, 데이터의 특징을 추출하기 위한 방법이 적용 분야에 따라 달라질 수 있어 제한된 환경 내에서만 사용된다. 내용기반 접근방법은 영상의 내용 즉, 색상, 질감, 물체의 모양, 공간적인 관계 등과 같은 영상의 특징과 속성을 자동으로 추출할 수 있다는 장점이 있다. 그러나 영상의 특징을 표현하기 위해서 색상 히스토그램을 많이 사용하고 있으나<sup>[2]</sup>, 영상내의 특징을 색상만으로 나타내는 것은 히스토그램의 많은 장점에도 불구하고 공간정보를 고려하지 못한다는 단점을 가지게 되어 유사한 히스토그램을 가진 다른 영상을 검색하게되는 결과를 가져온다.

본 논문에서는 이러한 문제점을 해결하기 위하여 색상-공간정보를 함께 고려하기로 한다. 색상-공간정보를 고려하기 위한 방법으로 영상을 분할하는데 있어 고정분할 방법을 사용하지 않고 영상의 색상이 동일한 영역을 적응적으로 처리하여 영역을 분할하는 기법을 사용하고자 한다. 분할된 영역에서 포함하고 있는 평균색상과 해당 영역의 중심좌표를 계산하여 영역의 분산을 계산하고, 영역의 크기를 계산하여 분리된 객체의 모양을 비교할 수 있도록 함으로써 색상과 공간을 함께 고려할 수 있는 방법을 제안한다.

본 논문의 구성은 II장에서 공간정보를 고려한 영상검색에 대한 기존 연구에 관하여 살펴보고 이들에 대한 문제점을 파악해 본다. III장에서는 본 논문에서 제안하는 색상-공간정보를 고려하기 위한 순차영역분할과 특성추출 방법에 대하여 서술하였고, IV장에서는 유사도 측정방법과 성능평가에 대해 기술하였다. 그리고 V장에서는 실험결과를 기존의 방법과 비교하여 분석하였으며, VI장에서 결론과 향후 연구 방향에 대하여 기술하였다.

## II. 공간정보를 고려한 영상검색

영상정보의 중요성을 인식하고 많은 사람들이 내용기반 영상검색에 대한 연구를 하여 왔다. 내용기반 영상검색 중에서 가장 대표적인 방법은 Swain에 의한 히스토그램 인터섹션이다<sup>[2]</sup>. 이 방법은 같은

색상을 가진 화소들의 개수를 색상 빈(bin)의 빈도수에 따라 계수(count)하여 색상 히스토그램으로 표현한다. Hafner 등은 영상의 색상분포 뿐 아니라 히스토그램의 각 구간사이의 거리와 최대거리에 대한 비율을 유사행렬을 이용하여 히스토그램 사이의 유사성을 구분하였다<sup>[3]</sup>. 이러한 방법은 영상내의 색상 히스토그램만을 비교하므로 계산이 복잡하지 않다는 장점이 있으나, 포함된 물체의 위치나 공간적인 관계를 나타내지 못하고 색상의 분포만을 고려함으로써 유사한 히스토그램을 가지는 서로 다른 영상을 구별하지 못하고 있다.

영상의 물체나 공간적인 정보를 고려하기 위한 다양한 연구가 진행되어 왔다. Hsu 등은 영상의 엔트로피(maximum entropy discrimination)를 계산하고 추출된 영역을 사각형 꼴로 표현한 후, 영역의 대표칼라와 공간정보를 분석하여, 통합된 색상-공간 정보에 기반한 검색방법을 제안하였다<sup>[4]</sup>. Belongie 등은 영상을 “blob”이라는 형식으로 표현하여 영상 검색에 적용하고 있다. 이는 자연영상을 blob (region)으로 나누고 각 영역에서 중앙과 주변에서 나타나는 대표색상을 특성벡터로 이용하고 있다<sup>[5]</sup>.

영상의 영역이나 객체들을 기반으로 하는 색상정보가 공간적인 분포를 고려하기 위한 방법으로 매우 유용하지만, 영역을 분할하는데 어려움이 있으므로 간단한 고정 크기의 분할 방법이 다양하게 이용되고 있다. Stricker 등은 영상을 중앙의 타원형과 겹치는 영역(fuzzy regions)을 포함하여 5영역으로 구분하고, 각 영역에서 히스토그램 및 1, 2, 3차 모멘트를 추출하여 비교하였다<sup>[6]</sup>. 이들은 히스토그램과 모멘트를 사용하여 계산량을 줄이는 효과를 가져온 반면, 영상에 나타나는 색상정보를 상세히 반영하지 못하는 단점이 있다.

영상을 고정 또는 가변 블록으로 분할하지 않고 공간적인 색상 분포를 표현하기 위한 다른 접근 방법이 있다. Pass 등은 히스토그램 방법의 단점을 보완하기 위해 영상내에 포함하고 있는 같은 색상의 응집(coherence)여부를 이용한 CCV(color coherence vectors)<sup>[11]</sup>를 제안하며, Huang 등은 주어진 칼라의 화소가 임의의 거리에 존재할 확률을 계산한 테이블을 작성하여 색인하는 color correlogram이라는 방법을 제안하였다<sup>[7]</sup>.

이와 같이 영상이 지니고 있는 다양한 특징 중에서 영상의 전체적인 색상 히스토그램과 모멘트들을 이용하면 다른 정보와 비교하여 상대적으로 비교적 쉽게 추출할 수 있을 뿐 아니라, 영상의 크기나 방

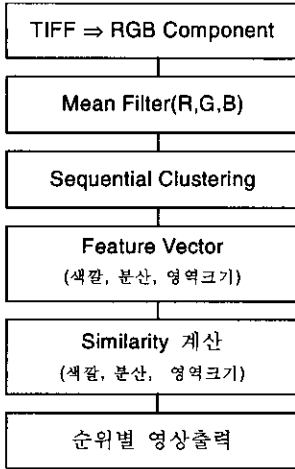


그림 1. 제안한 영상검색방법의 알고리즘

향의 변화에 강한 성질을 지니고 있어 영상검색에 매우 유용한 정보를 제공하고 있지만, 영상내의 공간 정보를 충분하게 고려하지 못하고 있다. 이로 인해 영상의 개수가 많아지면 검색결과가 좋지 않게 되는 경향이 있으므로 본 논문에서는 이러한 문제점을 해결하기 위하여 색상-공간정보를 함께 고려하기로 한다. 색상-공간정보를 고려하기 위한 방법으로 영상을 분할하는데 있어 고정분할 방법을 사용하지 않고 영상의 색상이 동일한 영역을 적응적으로 처리하여 영역을 분할하는 기법을 사용하고자 하며, 분할된 영역에서 포함하고 있는 평균색상과 해당 영역의 중심좌표를 이용한 영역의 분산, 그리고 영역의 크기를 계산하여 분리된 객체의 모양을 비교할 수 있도록 함으로써 색상과 공간을 함께 고려할 수 있는 알고리즘을 그림 1과 같이 제안한다.

### III. 순차영역분할과 특성추출

#### 1. 순차영역분할 (Sequential Clustering)

영상의 특성이 동일한 영역으로 분할하기 위해 패턴인식 분야에서 사용되고 있는 순차영역분할 기법<sup>[8]</sup>을 영상검색에 맞도록 변형하여 도입하려고 한다. 이 방법은 연속되는 자료를 군집화(clustering)하는 기법으로 최초의 자료를 첫 번째 군집영역(clustering region)의 대표값으로하고, 이 대표값과 뒤이은 자료와의 거리를 계산하여 임의의 설정치(threshold)와 비교한다. 이때 설정한 범위를 벗어나면 새로운 군집영역을 생성하게 되고, 범위 안에 있으면 해당 군집영역에 할당하게 함으로서 연속되는 자료를 군집

화한다.

즉, 입력되는 영상의 화소 값인  $f$ 와 군집영역  $C$  간의 거리를  $d(f, C)$ 로 나타내면

$$d(f, C) = d(f, \bar{C}) \quad (1)$$

가되며, 이때  $\bar{C}$ 는 군집영역  $C$ 의 대표값(평균값)을 나타내며, 생성된 군집영역에 입력영상의 화소의 값  $f$ 가 할당될 때마다 해당 군집영역의 대표값(평균값)을 식(2)와 같이 적응적으로 변화시킨다.

$$\bar{C}_k^{new} = \frac{(nC_k^{new} - 1)\bar{C}_k^{old} + f}{nC_k^{new}} \quad (2)$$

여기서  $nC_k^{new}$ 는 입력화소  $f$ 가  $k$ 번째 군집영역에 할당된 상태에서 계산된 화소의 수를 나타내며,  $\bar{C}_k^{new}$  및  $\bar{C}_k^{old}$ 는 각각 입력화소  $f$ 가 군집영역  $C_k$ 에 할당되기 전과 후의 대표값(평균값)을 나타낸다.

본 논문에서는 RGB공간에서 순차영역분할 알고리즘을 적용하며, 군집영역의 구분을 더욱 정확히 하기 위해서 군집영역결정 루틴과 영역분할 루틴으로 나누어 최초 루틴에서는 각 영역의 대표값 즉, 색상 평균값을 결정하고, 다음 루틴에서 영역을 분할하는 기법을 사용한다. 영역의 대표값은 다음과 같이 계산한다.

$$\bar{R}_k = \frac{\sum r_k}{n_k}, \quad \bar{G}_k = \frac{\sum g_k}{n_k}, \quad \bar{B}_k = \frac{\sum b_k}{n_k} \quad (3)$$

$$R_k = |r - \bar{R}_k|, \quad G_k = |g - \bar{G}_k|, \quad B_k = |b - \bar{B}_k| \quad (4)$$

$$D = \sqrt{R_k^2 + G_k^2 + B_k^2} \quad (5)$$

$$Min(TH, D) = \begin{cases} C_k, & TH \leq D \\ C_{k+1}, & TH > D \end{cases} \quad (6)$$

여기에서  $k$ 는 순차영역분할 알고리즘에 의해 생성된 군집 영역의 수를 나타내며,  $n_k$ 는  $k$ 번째 영역의 화소수를 나타낸다.  $\bar{R}_k, \bar{G}_k, \bar{B}_k$  및  $r_k, g_k, b_k$ 는 각 영역의 RGB 3채널의 평균치 및 화소의 값을 나타내며,  $r, g, b$ 는 입력되는 화소값이다.  $R_k, G_k, B_k$ 는 입력되는 화소와 각 군집영역의 평균값과의 거리를,  $D$ 는 RGB 3채널의 유클리디안 거리를 나타내며,  $TH$ 는 영역을 구분하는 설정치를 나타낸다. 이상의 알고리즘을 요약하면 그림 2와 같고,

순차영역분할에 의한 분할된 영상을 표시하면 그림 3과 같다.

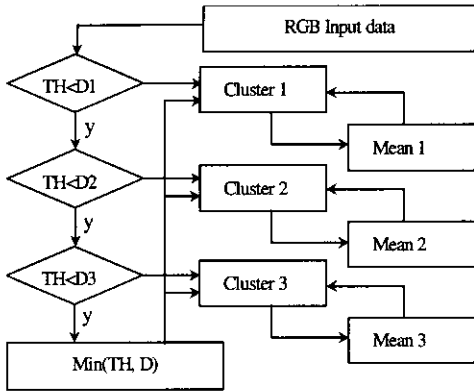
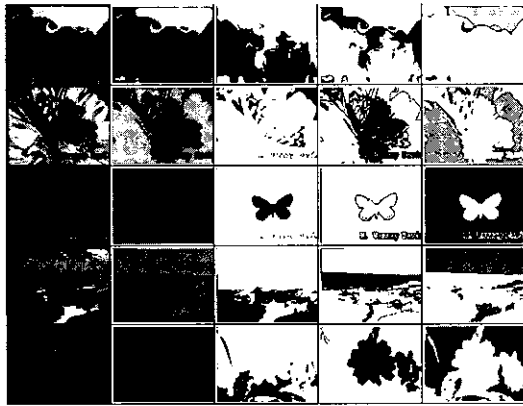


그림 2. 순차 영역분할 알고리즘



(a)원영상 (b)합성영상 (c)영역1 (d)영역2 (e)영역3

그림 3. 순차영역분할의해 분할된 영상

2. 특성 추출

분할된 영역으로부터 영상이 지니고 있는 특성을 추출하기 위해 각각의 영역에서 RGB 3채널의 색상의 평균값, 영역의 중심점으로부터 화소들의 분산정도, 그리고 영역의 크기를 다음과 같이 계산한다.

$$R_k = \frac{\sum r_k}{255 \cdot N_k}, G_k = \frac{\sum g_k}{255 \cdot N_k}, B_k = \frac{\sum b_k}{255 \cdot N_k} \quad (7)$$

$$XV_k = \frac{\sqrt{\sum(x_i - \bar{X})^2}}{N_k}, YV_k = \frac{\sqrt{\sum(y_i - \bar{Y})^2}}{N_k} \quad (8)$$

$$SIZE_k = N_k \quad (9)$$

여기에서 N은 화소 수, R, G, B는 평균색상값,

$\bar{X}$ ,  $\bar{Y}$ 는 영역의 중심좌표, 그리고 XV, YV, SIZE는 각각 분산과 영역의 크기를 나타낸다. 이러한 결과로 색상 평균값 9개와 분산 값 6개, 영역의 크기 값 3개 즉, 18개의 특성 값을 특성벡터로 하여 색인 정보로 저장한다.

IV. 유사도 측정 및 성능평가 방법

1. 유사도 측정

일반적인 검색시스템과는 달리 내용기반 영상검색은 정확한 검색을 요구하는 것이 아니고, 질의영상과의 유사성의 순서에 따라 데이터베이스 내에 있는 영상을 순위별로 검색하여 보여 주기 위하여 질의벡터와 색인벡터간의 유사한 정도를 계산해야 한다.

영상검색 시스템에서 가장 많이 사용되는 유사도 측정방법은 다음 식과 같이 벡터간의 거리를 계산하는  $L_p$  계산(norm)방법이다.

$$L_p(\bar{Q}, \bar{T}) = \left| \sum_i (Q_i - T_i)^p \right|^{1/p} \quad (10)$$

두 영상간의 유사성을 측정하기 위해서는 가장 많이 사용되는 함수인 유클리드 거리 함수를 이용한다. 유클리드 거리 함수의 특징은 정규적이고 변환하에서는 그 거리가 항상 보존된다는 것이다. 그리하여 두 벡터간의 거리를 유클리드 거리로 사용하면, 위 식에서 P의 값은 2로 할 수 있다. 즉,

$$Distance = \left| \sum_i (Q_i - T_i)^2 \right|^{1/2} \quad (11)$$

$$Similarity = 1 - Distance \quad (12)$$

$$Distance(Q, T) = \frac{w_1 D_1 + w_2 D_2 + w_3 D_3}{w_1 + w_2 + w_3} \quad (13)$$

과 같다, 질의영상의 벡터와 색인벡터간의 차이를 나타내는 거리를 식 (11)에서 구한 후, 그 값의 역으로 식(12)와 같이 유사도를 측정한다. 특성벡터의 요소가 복합적인 요소로 구성되어 있으면 벡터간의 가중치를 주어 식(13)과 같이 정규화하여 유사도에 의한 순위를 결정한다.

본 논문에서는 두 영상간의 유사도를 측정하기 위해 특성벡터로 저장된 색상 평균값, 분산값, 영역의 크기의 유사도를 계산하고 이를 통합한 전체 유사도를 다음과 같이 계산한다.

$$dC(Q, T) = \sum_{k=1}^3 \sqrt{Rc_k^2 + Gc_k^2 + Bc_k^2} \quad (14)$$

$$dV(Q, T) = \sum_{k=1}^3 \sqrt{Xvar_k^2 + Yvar_k^2} \quad (15)$$

$$dS(Q, T) = \sum_{k=1}^3 \left| \frac{SIZE_{Q,k} - SIZE_{T,k}}{SIZE_{Q,k} + SIZE_{T,k}} \right| \quad (16)$$

$$S_D = \frac{w_1 \cdot dC + w_2 \cdot dV + w_3 \cdot dS}{w_1 + w_2 + w_3} \quad (17)$$

여기서  $w_1, w_2, w_3$ 은 특성벡터간의 가중치를 나타내며,  $Q$  및  $T$ 는 각각 질의(query)영상과 검색 대상(target)을 나타낸다. 또한  $Rc_k, Gc_k, Bc_k$ 는 RGB 3 채널에서 거리를 나타내며,  $Xvar_k, Yvar_k$ 은 XY좌표에서 영역의 중심점으로부터 분산값을 표시한다. 그리고  $dC, dV$  및  $dS$ 는 각각 색상, 분산 및 영역의 크기에 대하여 그리고  $S_D$ 는 전체적인 유사도를 나타낸다.

## 2. 성능 평가

본 논문에서 다루고 있는 내용기반 영상검색은 정보검색에서와 같이 정확한 검색을 요구하는 것이 아니라 유사한 영상을 검색하게 되므로 검색결과를  $k$ 번째 순위까지로 결정한다면 다음과 같은 값들을 구할 수 있다.

$$A_k = \sum_{i=0}^{k-1} V_i, B_k = \sum_{i=0}^{k-1} (1 - V_i), C_k = \sum_{i=0}^{k-1} V_i - A_k \quad (18)$$

여기서 순위  $l$ 에 들어가는 정보의 집합을  $V_l$ 이라 하면,  $V_l \in \{0, 1\}$ 이며,  $A_k$ 는  $k$ 번째까지 검색된 유사한 영상의 수,  $B_k$ 는 유사하지 않은 영상의 수,  $C_k$ 를 유사한 영상이  $k$ 번째 이후에 검색된 영상이다. 이러한 값들로부터 영상검색의 성능평가를 하기 위한 표준척도로 사용되는 recall 및 precision은 다음과 같이 계산한다.

$$Recall : R_k = \frac{A_k}{A_k + C_k} \quad (19)$$

$$Precision : P_k = \frac{A_k}{A_k + B_k} \quad (20)$$

위의 식에 의하여 영상검색시스템의 효율성을 측정할 수 있으며, 모든 recall 값에 대하여 precision의 값이 높으면 더 나은 시스템이라 평가할 수 있다<sup>[9]</sup>. precision과 recall에 의해서 영상검색시스템을

평가할 때 고려해야할 가장 중요한 문제가 주관적으로 유사한 영상을 판단해야한다는 점이다. 영상의 개수가 많아지면 실험을 하기 위해 유사한 영상을 일일이 구분한다는 것은 매우 어렵기 때문에 다음과 같은 식을 사용하여 성능을 평가하면 동일한 신뢰성을 가지면서 성능평가에 소요되는 시간을 줄일 수 있다.

$$Eff = \begin{cases} \frac{n}{N}, & \text{if } N \leq T \\ \frac{n}{T}, & \text{otherwise} \end{cases} \quad (21)$$

여기에서  $n$ 은 검색된 유사한 영상의 개수,  $N$ 은 전체 유사한 영상의 개수이며,  $T$ 는 검색결과로 고려할 순위(rank)를 나타낸다. 식에서 보는 바와 같이  $N \leq T$ 이면 recall값과 같으며,  $N > T$ 이면 앞서 언급한 precision과 같은 값을 가지게 된다.

## V. 실험결과 및 분석

본 논문에서 구현한 색상-공간정보를 고려한 내용기반 영상검색시스템은 Windows 98을 운영체제로 하는 Pentium III 650Mhz에서 실험하였다. 사용된 언어는 MicroSoft Visual C++ 6.0 컴파일러를 사용하였으며, 실험에 사용된 영상은 Berkeley 대학의 『BlobWorld』 시스템<sup>[10]</sup>에서 사용한 총 418개의 192×128크기의 TIFF영상을 실험영상으로 사용하였다. 그리고 질의 방법은 예제영상에 의한 질의 방법(query by example image)으로 53개의 질의 영상을 선택하여 실험하였다. 순차 영역분할을 하기 위한 설정치는 실험에 의한 값인 55로 하였으며, 영역의 개수는 3개로 고정하였다. 그리고 색상평균값, 분산의 정도, 영역의 크기에 관한 유사도의 가중치인  $w_1, w_2, w_3$ 는 각각 4, 2.5, 2를 주어 실험하였다.

본 논문에서는 제안한 시스템을 평가하기 위해 주관적인 평가와 객관적인 평가를 병행하였다. 주관적인 평가는 검색된 결과를 영상으로 직접 시각적으로 확인하는 방법을 사용하였으며, 객관적인 평가 도구는 식 (19), (20) 및 (21)을 사용하였다. 또한 평가의 객관성을 유지하기 위해 동일한 영상자료로서 제안한 방법과 Swain에 의한 방법, CCV 및 Correlogram에 의한 방법들과 비교하였다.

제안한 방법에서 사용하는 특성벡터는 3영역에서 각각 평균색상 3, 영역의 분산값의 좌표 2, 그리고

영역의 크기를 나타내는 값 1을 합쳐 총 18개의 요소를 특성벡터로 하고 있는 반면, Swain의 방법은 opponent RGB공간에서 rg축과 by축을 각각 16 등분, wb축을 8등분하여  $16 \times 16 \times 8 = 2048$ bin(bins)이 되도록 양자화하여 이를 특성벡터로 사용하고 있다. 그리고 Pass 및 Huang에 의한 CCV, Correlogram의 방법은 공히 RGB공간을  $4 \times 4 \times 4 = 64$ bin이 되도록 양자화하고 있으며, CCV에서는 색상의 응집 정도(coherence, incoherence)를 구분하므로 128개의 요소를 특성벡터로 사용하고 있다. Correlogram에서는 거리(distance set) 즉,  $D = \{1, 3, 5, 7\}$ 을 주어 계산하므로 256개의 요소로 이루어진 특성벡터를 사용하고 있다. 이러한 실험영상과 질의영상의 분류와 평가에 사용된 방법을 비교하면 다음 표와 같다.

표 1. 실험영상과 질의영상의 분류

종류	수량	관련	종류	수량	관련
새	34	2	꽃	56	6
나비	50	9	동물	83	5
풍경	68	17	바다	16	4
원주민	38	4	고기	12	2
나무	48	3	기타	12	1

표 2. 평가에 사용된 방법들의 비교

방법	색상 공간	특성수	거리합 수	영역 분할	용량 (byte)
제안방법	RGB	18	L2	3	52,672
CCV	RGB	128	L1	없음	226,560
Correlo	RGB	256	L1	없음	547,584
Swain	opp RGB	2048	H1	없음	3,456,800

주) 색인용량은 총 418개의 영상을 인덱스하는데 소요되는 용량이며, 파일명은  $10\text{byte} \times 418 = 4,180\text{byte}$ 가 각각 소요되고 있음

시스템의 성능평가 방법으로는 식(21)에서 계산한 Eff를 사용하였으며, 평가의 객관성을 확보하기 위해 임의로 선택한 53개의 질의영상에 대한 검색결과를 5위, 10위, 15위, 20위의 4가지에 대한 평균을 다음 표3에 나타내었다.

표 3에서 보는 바와 같이 본 논문에서 제안한 방

법이 타 방법보다 전체적으로 검색된 순위에 관계 없이 우수한 성능을 보이고 있다. 특히 20위까지 고려한 실험에서는 제안한 방법에서 유사한 영상을 91.7%까지 검색하는 반면, Correlogram은 82.7%, CCV는 77.6%, Swain에 의한 방법은 70.0%를 보이고 있어 제안한 방법의 성능이 매우 우수함을 보이고 있다.

또한 질의영상 중 “불가사리”(y0096), “꽃”(y0373)에 대한 Precision vs. Recall의 그래프를 그림 5와 같이 표시하였다. 그림에서 보는 바와 같이 전반적으로 Precision 대 Recall의 실험에서도 제안한 방법이 우수함을 알 수 있다.

주관적인 평가를 하기 위한 검색성능을 시각적으로 확인하고자 다음과 같이 “나비”영상에 대해 검색된 결과를 표시하였다. 이 그림에서 좌측 첫 번째 영상은 질의영상이며, 우측으로 나열된 순서에 따라 검색된 순위는 높아지게 된다.

표 3. Eff에 의한 성능 비교

방법	5th	10th	15th	20th
제안방법	81.7	77.4	80.6	91.5
Swain'	63.6	58.4	62.0	70.0
CCV	70.9	65.6	68.8	77.6
Correlo	74.6	68.7	73.1	82.7

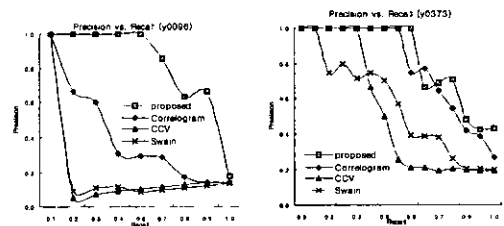
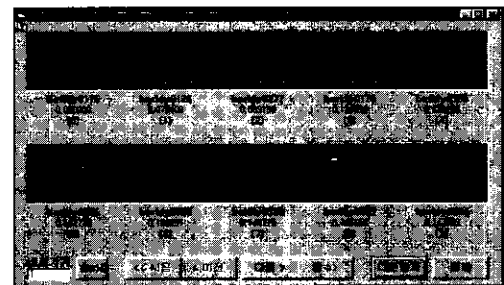
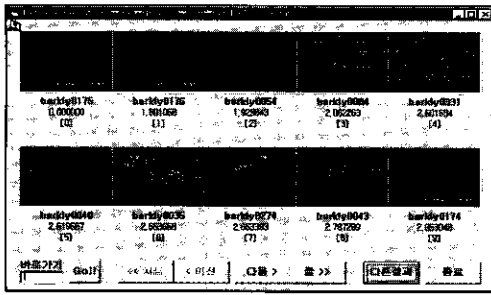


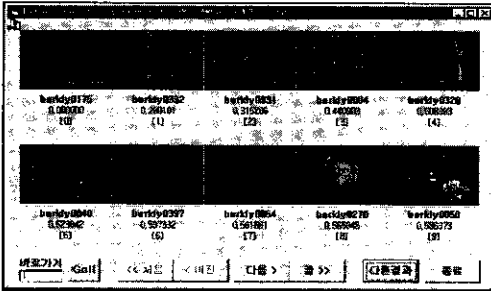
그림 5. Precision vs. Recall



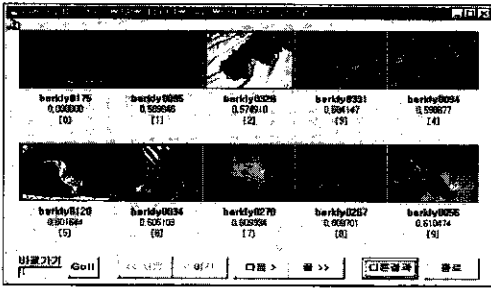
(a) 제안한방법



(b) Correlogram



(c) CCV



(d) Swain

그림 6. “나비”영상에 대한 검색결과

그림 6에서 보는 바와 같이 제안한 방법은 유사한 영상 모두가 4위권 내에서 검색되었지만, Correlogram에서는 y0176영상은 2위, y0174는 10위, y0177은 20위에 검색되었으며, CCV에서는 22위, 41위, 60위로, 그리고 Swain의 방법은 11위, 221위, 71위로 각각 검색되었다. 따라서 제안한 방법은 “나비”영상을 상위순위에서 모두 검색하였지만, 다른 방법에서는 상위순위의 범위를 벗어나는 영상이 많았다. 그러므로 시각적인 면에서도 제안한 방법이 우수함을 알 수 있었다.

## VI. 결 론

본 연구에서는 칼라영상의 내용기반 영상검색에

관하여 연구하였다. 내용기반 영상검색을 하기 위해서 색상과 공간정보를 고려하는 것은 매우 중요한 일로서, 본 논문에서는 순차 영역 분할 기법을 도입하여 분할된 영역의 색상과 공간정보를 고려하기 위한 특성추출 방법을 제안하였다. 색인되는 영상의 특징으로는 분할된 영역의 색상평균값, 영역의 중심 좌표로부터 분산값과 해당 영역의 상대적 크기를 특성벡터로 색인하였다. 색인되는 특징의 수가 영상당 18개로 타 방법보다 매우 작은 저장공간을 가지면서도 검색성능에는 우수한 효과를 보이고 있었다.

제안한 방법의 유효성을 증명하기 위한 방법으로 주관적인 방법과 객관적인 방법을 사용하였으며, 두 가지 평가방법에서 모두 제안한 방법이 우수함을 증명하였다. 객관적인 방법으로는 기존의 방법 중에서 색상정보만 고려한 Swain의 방법과 색상과 공간정보를 함께 고려한 두 가지 방법들에 대한 비교실험을 통해 제안한 방법의 유효성을 증명하였다. 여기에서 제안한 방법에서 추출된 특성들의 수가 Swain과는 1/117, correlogram과는 1/14, CCV는 1/7임에도 불구하고 검색성능 비교에서 Swain은 70.0%, CCV는 77.6%, Correlogram은 82.7%인 반면, 제안한 방법은 91.5%를 나타내어 제안한 방법이 매우 우수함을 알 수 있었다. 그리고 Precision 대 Recall 평가에서도 각 질의영상에서 대부분의 Recall값에서 제안한 방법이 모두 우수함을 알 수 있었다. 또한 주관적인 평가에서도 제안한 방법의 우수성을 검색결과를 통해 시각적으로 확인할 수 있었다.

향후 연구해야할 과제로는 제안한 방법이 다른 색상공간에서 특성을 추출할 경우 검색성능에 미치는 영향과 각 영역에서 추출되는 특성을 표현하는데 있어 기하학적인 방법을 도입하여 표현함으로써 더욱 향상된 검색성능을 얻을 것으로 기대한다.

## 참 고 문 헌

- [1]. Greg Pass, Ramin Zabih, and Justin Miller, “Comparing Images Using Color Coherence Vectors,” *ACM Conference on Multimedia*. Boston, Massachusetts, November 1996.
- [2] M. J. Swain and D. H. Ballard, “Color Indexing,” *International Journal of Computer Vision*, 7(1), pp. 11-31, 1991.
- [3] Jane Hafner, H. S. Sawhney, W. Equitz, M. Flickner, and W. Niblack, “Efficient color

Histogram Indexing for Quadratic form Distance functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Volume 17, Number 7, pp. 729-736, July, 1995.

[4] W. Hsu, T. S. Chua, and H. K. Pung, "An Integrated Color-Spatial Approach to Content-based Image Retrieval," *Proceedings of ACM Multimedia*, pp. 305-313, Sanfrancisco, November, 1995.

[5] S. Belongie, C. Carson, H. Greenspan, and J. Malik, "Color- and Texture-Based Image Segmentation Using the Expectation-Maximization Algorithm and Its Application to Content-Based Image Retrieval," *International Conference on Computer Vision (ICCV'98)*, Bombay, India, Jan 1998.

[6] M. A. Stricker and A. Dimai, "Color indexing with weak spatial constraints," *Storage and Retrieval for Still Image and Video Databases*, volume 2670, pages 29-40, 1996.

[7] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image Indexing Using Color Correlograms," *IEEE Computer Vision and Pattern Recognition Conference*, San Juan, Puerto Rico, June 1997.

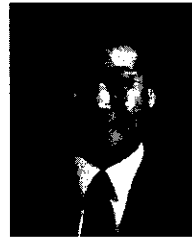
[8] Sergios Theodoridos and Kostantinos Koutroumbas, *Pattern Recognition*, Athens: Academic Press, 1999.

[9] J. R. Smith, "Image Retrieval Evaluation," *Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries*, pp. 112-113, Santa Barbara, California, 21 June 1998.

[10] Chad Carson, Megan Thomas, Serge Belongie, Joseph M. Hellerstein, and Jitendra Malik, "Blobworld: A System for Region-Based Image Indexing and Retrieval (long version)," University of California, Berkeley, 1999. <http://elib.cs.berkeley.edu/p-hotos/blobworld>

장 정 동(Jungdong Jang)

정회원



1990년 2월 : 방송통신대 전자계산공학과 졸업

1996년 2월 : 영남대학교 산업대학원 전자공학과 석사

1999년 8월 : 영남대학교 전자공학과 박사과정 수료

1970년 3월~현재 : 한국전력 공사 전자통신처 근무  
<주관심 분야> 영상통신, 영상검색, 영상복원 등

이 태 흥(Taihong Yi)

정회원

vol.25, no.1B 참조