논문96-1-1-11

그래프 표현을 이용한 컬러 영상 데이터베이스 검색기법

박 인 규, 윤 일 동, 이 상 욱

Color Image Retrieval from Database Using Graph Representation

In Kyu Park, Il Dong Yun, and Sang Uk Lee

요 약

본 논문에서는 컬러 영상 데이터베이스에서의 효과적인 영상검색 및 비교를 위한 새로운 알고리듬을 제시한다. 이를 위하여 전처리 과정에서 조명의 영향을 배제하기 위한 새로운 컬러 균일성 보정 알고리듬을 제안하여 조명의 변화에 민감한 기존의 방법과는 달리 기준 영상의 조명조건으로 입력 영상의 조명조건을 변화시킴으로써 히스토그램을 이용한 컬러 영상 검색에 있어서 적절하게 이용된다. 또한 제안하는 영상 정합 기법은 컬러 영상를 노드와 에지로 표현되는 그래프 형식으로 표현하여 그래프 정합방법을 통해 두 영상의 유사도를 측정한다. 제안하는 알고리듬은 기존의 방법에 의해 영역화 과정이 없고 기하학적 정보추출을 위한 시간이 필요하지 않으므로 빠르면서도 우수한 검색 성능을 나타낸다.

Abstract

In this paper, an effective color image retrieval algorithm is proposed based on the graph representation. Also we propose a color constancy algorithm to remove the effect of illumination change. Illumination condition of an image can be transformed to that of reference image using the proposed color constancy algorithm, so that the effect of dirrerent lighting is significantly alleviated. Then, we represent a color image as a graph with several nodes and edges in the histogram space, and finally two images are matched by compared two graphs representing them. The simulation results show that the proposed 3-step algorithm performs well for various conditions, including different lighting, translation, rotation, and scaling of the object in the image. In addition, the proposed algorithm is very fast compared to the geometry-based matching technique.

I. 서 론

데이터 기록 및 저장장치의 용량이 증가함에 따라서 많은 영상을 가지고 있는 영상 데이터 베이스에서 원하는 영상을 찾아내는 검색방법이 중요해지고 있다[1]. 특히 데이터베이스에 저장된 영상의 수가 매우 많고 영상의 크기가 큰 경우 기존의 영역화(segmentation) 또는 기하학적 특징추출을 통한 두 영상간의 비교를 통한 검색은 시간이 너무 많이 소요되기 때문에 적당하지 못하다. 또한 영상마다 고유의 키워드를 부여하고 검색하는 경우 검색시간은 줄어들지만 많은 수의 영상의 경우 사전의 분류작업이 필요하고 같은 키워드를 가지는 다른 영상이 존재할 가능성이 높아지게 되는 단점을 가지고 있다. 따라서 영상 검색시 입력영상내의 물체의 변형, 조명의 변화와 영상의 취득시점의 변화에 강건하게 동작하는 기법에 관한 관심이 크다고 할 것이다. 다음에는 이러한 목적에서 제안된 기존의 컬러 영상 검색 기법의 특성을 간략하게 언

서울대학교 전기공학부
School of Electrical Engineering, Seoul Nat'l Univ.

급한다.

Swain은 컬러 영상의 경우 밝기 영상에 비해 정보량이 많음을 이용하여 영상의 통계학적 특징을 이용한 3차원 히스토그램을 생성한 후에 히스토그램 공간상에서 모델과 입력영상의 정합을 시도하는 인덱싱 알고리듬을 제시하였다[2]. 이 알고리듬은 그 간단명료함에 비하여 좋은 결과를 보여주었는데 물체의 변형, 일부의 절취, 영상을 취득한 시점의 변화, 물체 크기의 변화 및 회전과 이동에 대해 비교적 강건한 정합결과를 나타내었다. 그러나 잡음의 첨가나 조명의 변화와 같이 히스토그램 자체를 변환시키는 요인에는 매우 민감히 반응하였다. 특히 조명의 변화는 그 영향이 심각하여서 알고리듬이 거의 동작하지 않는 결과를 나타내었다[3]. 그러므로 조명의 변화에 강건한 정합을 위해서 필연적으로 컬러 균일성 보정(color constancy) 알고리듬이 필요하다[4, 5, 6, 7]

Funt와 Finlayson은 Swain의 알고리듬의 단점을 보완하여 컬러 균일성이 보정된 새로운 히스토그램 인덱싱 알고리듬을 제시하였다[3]. 이 알고리듬에서는 어떤 조건하에서는 인접해 있는 화소에 대한 센서의 입력값의 비는 그 위치에서의 알베도(albedo)의 비가 되어 일정함을 보였고, 이 모델을 이용하면 물체의 화소값의 로그값의 미분은 항상 일정하게 되어 이 값을 이용한 히스토그램간의 정합을 통해 조명의 영향이 배제된 영상간의 정합이 가능하였다. 그러나 이 알고리듬은 미분을 이용하기 때문에 잡음에 민감하고 로그함수의 비선형적 특성상 큰 화소값에서는 많은 정보가 소실되어 잘 동작하지 않는 단점이 있다.

Stricker는 히스토그램을 이용한 정합기법에 기하학적 정보를 포함시키기 위해서 경계 히스토그램(boundary histogram)을 제시하고 인접한 두 색간의 길이를 추정하여 새로운 히스토그램을 생성하였다[8]. 이 알고리듬은 히스토그램의 크기가 작고 잡음의 영향에 강건한 결과를 나타내었다.

일반적으로 영상의 검색성능을 결정하는 요소로는 정합율, 정합속도 그리고 정합가능한 모델의 수를 들수 있다. 정합율에 영향을 미치는 요소로는 물체의 변형, 회전 및 이동, 잡음의 영향, 조명의 변화 및 영상의 취득시점의 변화가 있고 정합속도에 영향을 미치는 요소로는 히스토그램의 수, 입력영상의 크기와 해상도, 사용하는 알고리듬의 효율성등이 있다. 기존의 연구들이 대부분 히스토그램을 이용한 정합을 이용하는데, 이는 히스토그램 표현이 영상내 물체의 이동, 회전등의 변화에 불변하며 어느 정도의 잡음이나 크기 변화(scaling)에 강건하기 때문이다.

본 논문의 구성은 다음과 같다. 제 Ⅱ장에서는 본 연구가 제안하는 전처리과정 및 히스토그램 생성 알고리듬과 그래프 생성 및 정합 알고리듬에 대해 서술한다. 그리고 제 Ⅲ장에서는 제안하는 알고리듬에 대한 실험결과 및 다른 알고리듬과의 비교를 서술하고 마지막으로 제 Ⅳ장에서는 결론을 제시한다.

Ⅱ. 제안하는 알고리듬

제안하는 알고리듬은 전처리 과정으로 컬러 균일성 보정 알고리듬과 영상의 그래프 기반표현, 그리고 그래프 정합 알고리듬의 세 단계로 구성된다. 첫 번째 단계에서는 입력영상을 비교하는 모델 영상이 취득된 조명조건과 같은 조명에서 취득한 영상으로 변환하며 이 과정에서 물체의 크기변화에 의한 영상의 밝기 왜곡이 보정된다. 기존의 정합에 있어서는 별도의 컬러 균일성 보정 알고리듬을 사용하였다는 가정하에 정합을 수행하였음에 비하여 본 논문에서 제안하는 알고리듬은 히스토그램을 이용한 정합에 알맞은 컬러 균일성 보정 알고리듬을 제안함으로써 실제 시스템으로의 구현이 용이하도록 한다. 두 번째 단계는 그래프 생성단계로서 양자화된 히스토그램의 각 원소를 하나의 노드로, 인접하는 화소의 수를 노드와 노드를 잇는 에지의 강도로 정의하여 영상에 대한 고유한 그래프를 생성한다. 마지막으로 이러한 방법으로 정의된 그래프간의 정합을 통해 두 이미지의 유사도를 정량적으로 표현할 수 있다. 전체적인 알고리듬의 구조는 그림1에 제시한다.

1. 컬러 균일성 보정 알고리듬

일반적으로 컬러 균일성 보정 알고리듬은 컬러 영상을 분석하여 물체의 각 위치에서 알베도를 추정해 내는 것이다[4]. 그러므로 컬러 균일성이 보정된 후 만일 어떤 알려진 주변광이 주어진다면 우리의 눈에 실제로 비춰질 영상을 얻을 수 있는 것이다. 그러나 영상 취득시의 조명과 입력센서의 파장에 대한 특징을 영상과 함께 취득하고 그 정보를 저장하기는 매우 어렵다. 따라서 본 논문에서 제안하는 알고리듬은 알베도를 직접 추정하기보다는 특정한 조명에서 취득한 영상으로의 변환을 수행하여 컬러 영상 정합이라는 특수한 목적에 사용될 수 있게 한다. 영상의 위치 (x, y)에서의 k번째 센서의 화소값은 식(1)과 같이 주변광 E, 알베도 $S^{(x,y)}$, 그리고 센서의 입력 특성 R_k의 곱의 일차결합으로 이루어진다[5].

$$\rho_k^{(x,y)} = \sum_l E(\lambda_l) S^{(x,y)} = (\lambda_S) R_k(\lambda_S) \tag{1}$$

여기서 영상에 비추어지는 조명의 변화는 물체표면의 알베도의 변화에 비해 매우 작다고 가정하였다. 또한 Funt의 가정과 마찬가지로 센서의 입력특성의 파장에 대한 분포가 매우 좁다고 가정하면 식(1)은 다음과 같이 근사된다[3].

$$\rho_k^{(x,y)} = E(\lambda_s) S^{(x,y)} = (\lambda_S) R_k(\lambda_S) \tag{2}$$

여기서 λ_S는 센서의 입력특성곡선에서의 지배적 파장(dominant wavelength)을 의미한다. 이제 식(2)의 양변에 Fourier 변환을 취하면 다음과 같은 결론을 얻는다.

$$\begin{aligned} F\{\rho_k^{(x,y)}\} &= F\{E(\lambda_S) S^{(x,y)}(\lambda_S) R_k(\lambda_S)\} \\ &= E(\lambda_S) R_k(\lambda_S) F\{S^{(x,y)}(\lambda_S)\} \\ &= E(\lambda_S) R_k(\lambda_S) \mid S^{(u,v)}(\lambda_S) \mid exp(j\Phi_S(u, v)) \end{aligned} \tag{3}$$

여기서 $S_F^{(x,y)}$는 $S_F^{(x,y)}$의 Fourier 변환을 나타내며 Φ_S는 $S_F^{(x,y)}$의 Fourier 변환을 나타낸다.

식 (3)을 고찰해 보면 결국 영상의 Fourier 변환의 위상(phase)은 조명과 센서 입력 특성에 관계없고 오직 물체의 알베

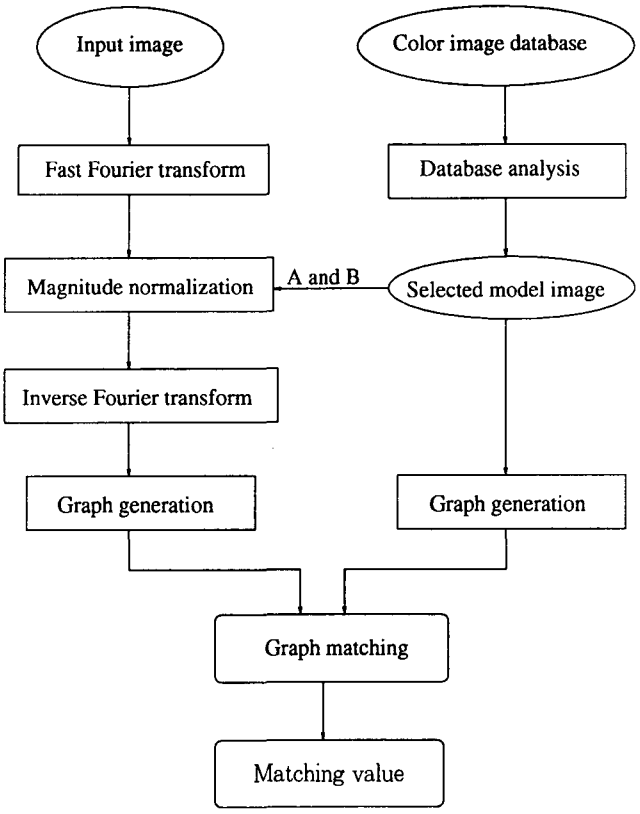

그림 1. 제안하는 알고리듬의 전체구조

도에 의존함을 알 수 있다. 즉 취득한 영상에서의 조명은 위상에는 영향을 미치지 않고 절대값에만 영향을 미친다. 따라서 영상의 Fourier 변환의 절대값(magnitude)를 적절히 변화시켜 줌으로써 균일한 인위적인 조명하에서 영상을 취득한 것과 같은 효과를 얻을 수 있다. 즉, 입력 영상의 Fourier 변환의 절대값의 평균이 m_k 이고 분산이 σ_k^2이라면 식 (4)와 같은 변환을 통해 Fourier 변환의 절대값의 평균이 AB이고 분산이 B인 영상의 Fourier 변환으로 바꿀 수 있다.

$$\rho'^{(x,y)}_k = \frac{\rho^{(x,y)}_k - m_k + \sigma_k \cdot A}{\sigma_k} \cdot B \qquad (4)$$

여기서 A, B는 모델영상으로부터 얻어지는 상수이다. 이제 위의 정규화 과정에 의해 정규화된 후 역 Fourier 변환을 수행하면 입력영상은 데이터베이스에서의 모델영상과 같은 조명에서 취득한 것과 같게 된다.

이제 모델영상과 비교하여 평행이동, 회전이동, 크기변화등이 일어난 경우를 생각해보자. 우선 입력영상에서 물체의 평행이동이 일어난 경우 Fourier 변환의 절대값에는 영향을 미치지 않고 위상의 변화만 가져오므로 절대값보정을 통해 조명의 영향을 없애는 것이 가능하다. 또한 영상에서 회전이동이 일어난 경우 Fourier 변환도 같은 각도만큼 회전하고 절대값및 위상에는 영향을 미치지 않는다[12]. 따라서 절대값에 대해 정규화과정을 거쳐 모델영상과 같은 조명에서 취득한 영상으로의 변환이 가능하다. 다음으로 물체의 크기에 변화가 일어나 모델영상에서의 물체의 면적에 비해 n배가 되었다고 할 때 각 주파수 대역의 값을 결정하는 경계의 길이는 $\sqrt{n}$배가 될것이므로 Fourier 변환의 절대값이 $\sqrt{n}$ 배가 된다. 따라서 이 경우 모델영상의 Fourier 변환의 절대값을 $\sqrt{n}$ 배로 변환하여 주어 크기변환에 따른 보정을 수행한다. 이와 같이 제안하는 컬러보정 알고리듬은 영상의 회전, 이동, 크기 변환등에 영향을 미치지 않으므로 히스토그램 표현기법의 특징을 그대로 이용하는 장점을 갖는다.

2. 컬러 영상의 그래프 표현

영상간의 정합을 위한 특징공간에서의 표현방식으로서 본 논문에서는 컬러 영상의 그래프 표현을 제안한다. 그래프는 노드와 에지로 구성이 되는데 여기서는 양자화된 히스토그램의 한원소를 노드로 본다. 노드정보는 히스토그램의 해당 원소가 포함하는 화소의 수이며 에지정보로는 두 히스토그램 원소들에 해당하는 색들의 인접화소의 수를 이용한다. 즉 인접하고 있는 색의 경계가 많을수록 두 노드간의 연결강도는 커지고 경계가 적을수록 두 노드간의 연결강도가 작아진다.

그런데 실제 영상에 대한 그래프 표현은 히스토그램 원소의 수

만큼 노드를 가지기 때문에 매우 복잡하다. 만일 히스토그램의 생성단계에서 R, G, B의 각 축을 8개의 구간으로 나누어 양자화한다고 할 때 그래프의 노드의 수는 모두 512개가 되고 에지의 수는 130186($=_{512}C_2$)개가 된다. 따라서 모든 노드에 대해 정합을 시도할 경우 정합시간이 많이 소요되기 때문에 일정한 수 이상의 화소를 포함하는 노드를 유효한 노드로 보고 그 노드들에 대한 정보만을 데이터 파일에 저장하여 다음 단계의 정합을 위한 정보로 이용하도록 한다. 실제 영상 그림 2의 경우 그래프 생성을 위한 유효노드를 가려내기 위한 문턱치는 전체 화소수의 0.5%로 설정하면 23개의 유효노드가 생성된다. 표 1에 그림 2에 대한 그래프 표현을 제시한다. 표 I에서 제 1행과 제 1열은 히스토그램 원소의 인덱스를 의미하고 행렬의 각 원소는 대각원소인 경우 해당 히스토그램 원소의 화소수를, 비대각 원소인 경우 두 히스토그램 원소가 의미하는 두 색이 인접한 화소수를 나타낸다. 이러한 그래프 표현은 히스토그램을 행렬로 효과적으로 표현할 수 있기 때문에 데이타베이스의 저장공간을 효율적으로 이용하게 한다.

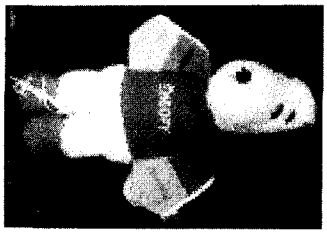

그림 2. 실제 영상(Swain, 1991)

3. 그래프 표현을 이용한 정합

컬러 영상에 대한 그래프가 생성되면 두 컬러 영상간의 유사도 측정도 두 래프간의 유사도 측정으로 바뀌게 된다. 우선 다음과 같이 두 그래프 J, H간의 연산을 정의한다.

- J의 행렬표현 : 각 행과 각 열은 노드를 의미하며 대각원소는 노드의 값을, 비대각 원소는 두 노드를 연결하는 에지의 값을 의미하는 상삼각행렬.
- $J \oplus H$: H에는 존재하고 J에는 존재하지 않는 노드를 J에 추가시키고 노드값과 연결된 모든 에지값을 0으로 만든다.
- $|J|$: 그래프 J의 행렬표현의 절대값 : 여기서 절대값은 행렬의 모든 원소의 절대값의 합을 의미한다.

위의 연산을 이용하여 정합함수를 다음과 같이 제안하여 정합의 결과를 정량적으로 나타낸다.

$$C(J, H) = |(J \oplus H) - (H \oplus J)| \tag{5}$$

위 함수는 두 단계의 연산을 각각의 그래프에 수행하는데, 먼저

표 1. 그림 2의 그래프를 행렬로 표현한 예

Index	0	147	148	149	163	166	171	193	194	228	235	243	300	301	365	373	437	438	446	506	507	510	511
0	82	64	0	1	0	0	0	0	0	5	0	0	0	3	0	0	0	0	4	0	0	0	0
147	64	98	10	1	3	2	0	6	3	94	2	0	48	9	16	5	8	9	9	0	0	1	2
148	0	10	128	56	0	2	0	4	3	10	0	0	3	11	1	0	0	0	0	0	0	0	0
149	1	1	56	180	0	3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
163	0	3	0	0	80	0	26	0	0	0	18	6	0	0	0	0	1	0	0	0	0	0	0
166	0	2	2	3	0	74	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	2
171	0	0	0	0	26	0	104	0	0	0	40	7	0	0	0	0	0	0	0	0	0	0	0
193	0	6	4	0	0	0	0	48	22	0	0	0	0	0	1	0	0	0	0	0	0	0	0
194	0	3	3	0	0	0	0	22	82	0	0	0	0	0	3	0	1	2	0	0	0	0	0
228	5	94	10	1	0	4	0	0	0	264	0	0	117	34	27	8	8	18	6	0	0	2	3
235	0	2	0	0	18	0	40	0	0	0	388	41	4	0	0	2	2	0	0	1	0	0	0
243	0	0	0	0	6	0	7	0	0	0	41	118	0	0	0	0	0	0	0	6	1	0	0
300	0	48	3	0	0	0	0	0	0	117	4	0	216	77	129	14	14	13	4	0	0	1	3
301	3	9	11	0	0	0	0	0	0	34	0	0	77	80	112	6	0	16	3	0	0	5	5
365	0	16	1	0	0	0	0	1	3	27	0	0	129	112	782	155	61	46	13	0	0	8	13
373	0	5	0	0	0	0	0	0	0	8	2	0	14	6	155	186	57	62	8	0	0	7	1
437	0	8	0	0	1	0	0	0	1	8	2	0	14	0	61	57	174	114	21	0	0	29	2
438	0	9	0	0	0	0	0	0	2	18	0	0	13	16	46	62	114	396	89	0	0	64	68
446	4	9	0	0	0	0	0	0	0	6	0	0	4	3	13	8	21	89	102	0	0	88	37
506	0	0	0	0	0	0	0	0	0	0	1	6	0	0	0	0	0	0	0	716	143	0	0
507	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	143	322	2	0
510	0	1	0	0	0	0	0	0	0	2	0	0	1	5	8	7	29	64	88	0	2	628	205
511	0	2	0	0	0	2	0	0	0	3	0	0	3	5	13	1	2	68	37	0	0	205	3388

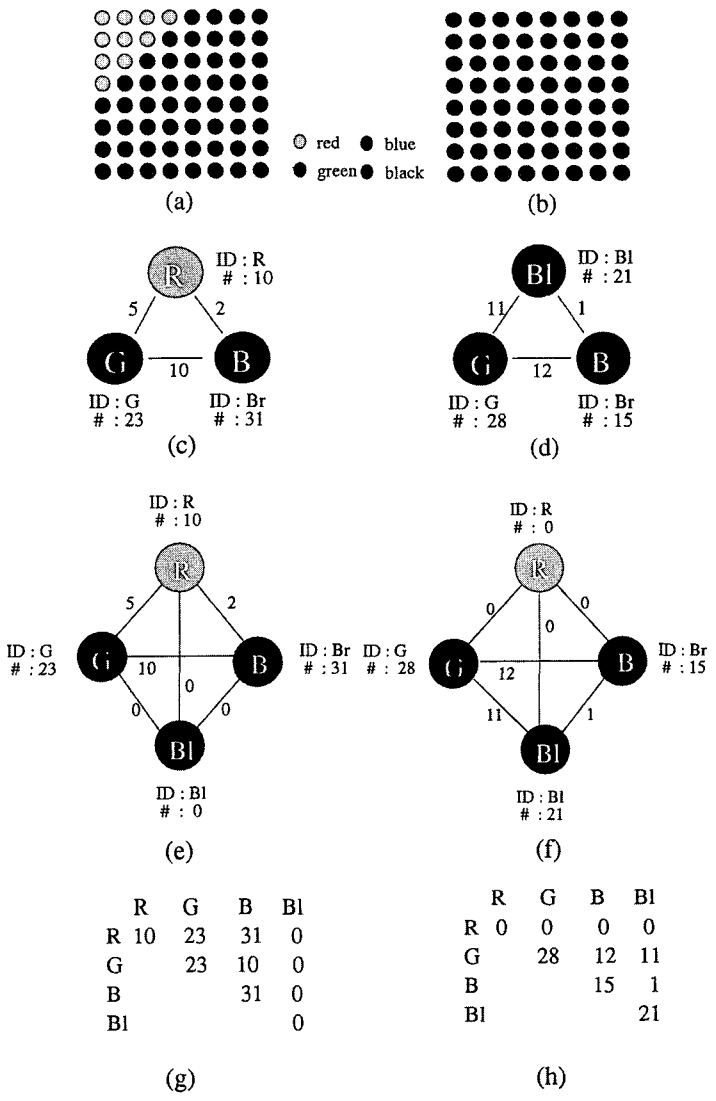

그림 3. 영상의 그래프 표현과 정합 :
(a) (b) 예제 영상, (c) (d) 그래프 표현 (e) (f)⊕연산수행 후의 그래프, (g), (h) 결과 그래프의 행렬 표현

비교하고자 하는 두 그래프의 노드의 수를 서로 일치시킨 후에 이들간의 유사도를 계산하는 단계로 이루어진다. 그림 3에 간단한 두 영상에 대한 위 연산들을 이용한 정합의 결과를 제시한다. 그림 3의 (a)와 (b)는 8×8 예제 영상을 나타내고 (c)와 (d)는 (a)와 (b)를 그래프로 각각 표현한 것이다. 그리고 (e)와 (f)는 (c)와 (d)의 그래프간에 각각 ⊕연산을 수행한 결과 그래프이며, (g)와 (h)는 그래프 (e)와 (f)를 행렬로 표현한 것이다. 이 때의 정합 결과 정합값은 두 행렬의 차의 모든 원소의 합으로 정의되므로 120이 된다. 그래프가 유사할 수록 정합값은 0에 가까워지고 만약 공통의 노드가 하나도 존재하지 않는다면 정합값은 두 행렬의 모든 원소의 합이 된다.

이제 제안하는 그래프 정합 알고리듬의 계산 복잡도를 고려해 보자. 제안하는 알고리듬에 대하여 세 가지 종류의 계산 복잡도 t_h, t_g와 t_m를 고려할 수 있다. 여기서 t_h, t_g와 t_m는 각각 입력 영상에 대한 히스토그램 생성시간, 입력 영상의 히스토그램에서의 그래프 생성시간, 그리고 그래프 정합시간을 각각 의미하고 평균 계산량은 다음과 같다.

$$t_h = O(d_r \cdot d_g \cdot d_b)$$
$$t_g = O(n_i)$$
$$t_m = O(M \cdot \frac{n_m + n_i}{2})^2) \quad (6)$$

여기서 d_r, d_g와 d_b는 히스토그램 양자화시의 R, G, B의 각 좌축표의 분할수를 의미한다. 그리고 M, n_m과 n_i는 데이터베이스의 모델 영상의 수, 모델 영상의 그래프 표현에서의 평균 노드수, 그리고 입력 영상의 그래프 표현에서의 노드수를 의미한다. 이 때 전체 계산량은 $t_h+t_g+t_m$으로 근사된다. 이 계산량은 Swain의 알고리듬보다 상재적으로는 많은 양이지만 실험결과 절대적인 검색시간은 실시간 검색에 충분할 정도로 작음을 알 수 있다. 즉, 100개의 모델 영상을 가진 데이터베이스에 대한 검색 결과, 하나의 입력 영상에 대한 검색시간은 SUN Sparc 20에서 약 1초정도가 소요되었다.

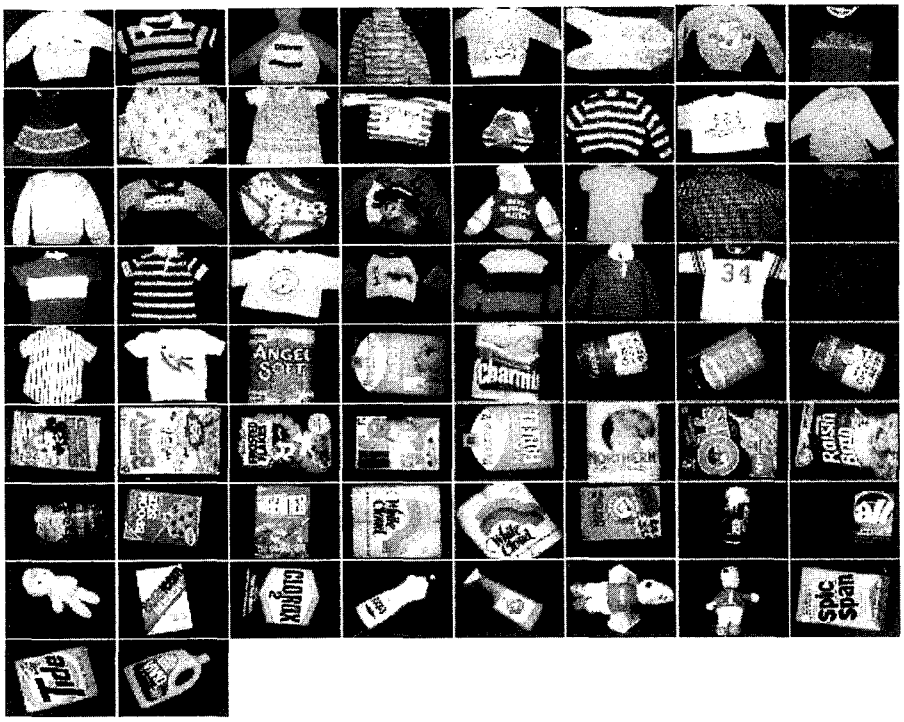

그림 4. Swain의 컬러 영상(Swain, 1991)

그림 5. Newsweek 컬러 영상

Ⅲ. 실험결과

제안하는 알고리듬은 컬러 균일성 보정과 영상 데이터베이스에서의 영상 검색에 있어서 우수한 성능을 나타내었다. 실험을 위하여 제작된 영상 데이터 베이스는 그림 4와 그림 5의 두 영상집합으로 구성된 100개의 영상으로 이루어져 있다. 그림 4에 나타낸 영상들은 Swain의 실험에 쓰인 영상들로서 128×90의 해상도를 가지는 66개의 영상들로 이루어져 있고 그림 5에 나타낸 Newsweek영상들은 직접 취득한 128×120의 해상도를 가지는 34개의 영상으로 이루어져 있다. 이러한 영상 데이터 베이스에 대하여 제안하는 알고리듬에 대한 실험을 수행하였다.

1. 컬러 균일성 보정 알고리듬

제안하는 컬러 균일성 보정 알고리듬에 대한 실험을 위해 데이터 베이스의 영상과 다른 조명조건에서 취득한 입력영상들의 조명조건을 데이터 베이스의 해당 모델영상의 조명조건으로 변환시키는 실험을 수행하였다.

이를 위해 데이터 베이스의 Newsweek영상에서 5개의 영상들을 추출하였다. 이 영상들은 형광등 조명에서 취득된 것들이며 이 영상들에 대해 4가지의 다른 조명, 즉 붉은색 조명, 초록색 조명, 푸른색 조명 그리고 백열등아래에서 취득한 20개의 영상에 대한 컬러 균일성 보정 실험을 수행하였다. 그림 6에 추출된 영상들을 나타내었고 그림 7에 컬러 균일성 보정을 하기 전과 보정을 한 후의 영상들을 각각 나타내었다. 그림 7에서 알 수 있듯이 서로 상이한 조명하에서 취득한 영상이 데이터 베이스의 영상들과 유사한 조명에서 취득한 영상으로 변환되었다. 영상들간의 유사도를 정량적으로 표현하기 위하여 보정전과 보정후의 모델영상에 대한 시트로그램간 공간상의 두 히스토그램의 거리(L1 norm)를 계산하여 표 Ⅱ에 나타내었고 결과적으로 컬러 균일성 보정후에 히스토그램 공간상의 거리가 상당히 감소함을 알수 있다. 이와 같이 컬러 균일성이 보정된 후, 변환된 영상들에 대해 Ⅱ장에서 기술한 그래프기반 정합을 수행하여 영상을 검색하였고 이에 대한 결과는 2절에서 서술한다.

2. 컬러 영상의 그래프기반 표현 및 정합

컬러 균일성이 보정된 영상에 대해 히스토그램을 생성한 후 이

그림 6. 형광등 조명에서 취득한 모델 영상

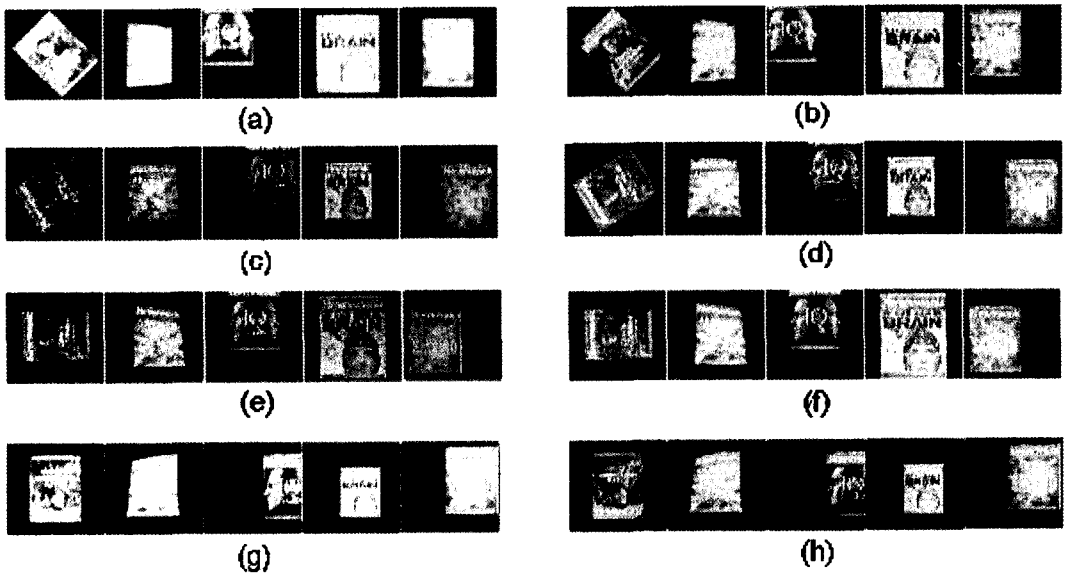

그림 7. (a)(c)(e)(g) 붉은 색, 초록 색, 푸른 색, 백열등 조명에서 취득한 입력영상 (b)(d)(f)(h) 컬러 균일성이 보정된 영상

표 2. 컬러 균일성 보정전과 보정후의 히스토그램 공간상의 거리변화(입력 영상 20개, 모델 영상 5개)

모델번호	조명종류	보정전 거리	보정후 거리	조명종류	보정전 거리	보정후 거리
1	적색조명	10048	4282	녹색조명	7984	3368
	청색조명	4495	2057	백열조명	9214	3898
2	적색조명	10355	4718	녹색조명	9656	3501
	청색조명	6359	3799	백열조명	10302	3986
3	적색조명	6912	5910	녹색조명	5172	2844
	청색조명	4299	2009	백열조명	6379	5913
4	적색조명	12603	5405	녹색조명	8989	3303
	청색조명	12026	5908	백열조명	8821	4391
5	적색조명	11210	4234	녹색조명	10466	2964
	청색조명	6109	4259	백열조명	9922	3398

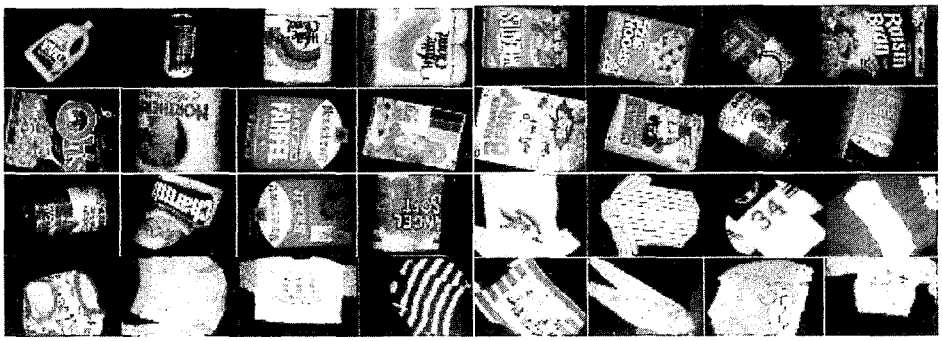

그림 8. 변형 및 왜곡이 일어난 입력영상(Swain, 1991)

그림 9. 조명의 변화가 일어난 입력영상

를 이용하여 영상을 그래프로 표현하여 정합에 이용하였다. 우선 그림 4와 그림 5로 이루어진 영상 데이터 베이스에 대한 입력으로서 그림 8에 나타낸 Swain의 입력영상과 그림 9에 나타낸 Newsweek 입력영상으로 이루어진 50개의 영상들에 대한 정합을 수행하였다. Swain의 입력영상들은 해당 모델영상에 비해 물체의 형태가 변화한 경우, 영상 취득 시점이 변화한 경우, 물체의 일부가 가리워진 경우등을 포함하는 왜곡된 영상들이다. 그러나 Swain의 입력영상들은 모델영상과 같은 조명조건에서 취득했기 때문에 제안하는 전처리 알고리듬의 정합에 대한 성능평가를 할 수는 없다. 따라서 영상 데이터 베이스내의 Newsweek 영상에 대해 조명조건을 달리하고 또한 동시에 평행이동, 회전이동, 물체의 크기변화를 수반하는 18개의 입력영상을 제작하여 입력영상에 포함시킨 후 결과적으로 생성된 50개의 입력영상에 대한 실험을 수행하였다. 정합결과는 1순위로 정합되는 경우를 성공적인 정합으

로 보았고 성공적인 정합이 일어나는 영상과 전체 입력영상과의 비율을 정합 성공율로 보았으며 전체적인 순위를 고려한 평균 정합율은 아래 식(7)에 의해 계산된다[2].

$$\eta=\frac{1}{M}\sum_{i=1}^{M}\frac{N-R_i}{N-1} \qquad (7)$$

여기서 N과 M은 각각 데이터베이스내의 영상과 입력 영상의 수를 의미하고 R_i은 i번째 입력영상의 정합 결과의 순의를 의미한다. 모든 입력 영상이 1순위로 정합되는 경우에는 η 값은 1이되며 순위가 내려갈수록 0에 가까운 값을 가진다. 따라서 정합 성공율이 높아도 하위 순위로 정합되는 영상이 존재한다면 평균정합율은 감소할 것이다.

각각의 알고리듬에 대한 성능평가를 위하여 각각의 정합결과를 표 Ⅲ과 Ⅳ에 제시하였고 또한 전체 정합결과를 표 Ⅴ에 제시하였다. 또한 객관적인 성능측정을 위하여 Swain의 알고리듬및 Funt의 알고리듬과의 성능 비교를 수행하였고 그 결과를 함께 제시하였다. 실제 구현에 있어서 히스토그램은 R, G, B좌표계 상에서 생성하였으며 Swain의 알고리듬은 512개의 히스토그램 원소를 가지고 Funt의 알고리듬에서 쓰인 미분연산자는 Laplacian 연산자를 이용하였다.

또한 영상의 그래프 표현에 있어서 유효노드를 찾는 문턱치는 전체 화소수의 0.01%이상으로 설정하였다. 표 Ⅲ에 나타낸 바와 같이 제안하는 알고리듬은 Swain의 영상에 대해 우수한 정합율을 보이며 기존의 알고리듬에 비해 우수한 정합 성능을 보였다. 컬러 균일성 보정 성능을 측정하기 위한 Newsweek 영상에 대해서도 표 Ⅳ에 제시한 바와 같이 제안하는 알고리듬은 기존의 알고리듬에 비해 우수한 성능을 나타내었다. 조명조건이 변화한 입력영상에 대해 Swain의 영상은 거의 동작하지 않음을 알 수 있다. 여기서 Funt의 알고리듬은 정합성능면과 컬러 균일성 보정에 있어서 잘 동작하지 않았는데 이는 Funt의 알고리듬이 포화된(Saturated)영상에 대해서는 로그함수의 특성상 많은 정보가 소실되기 때문이다. 실제로 Swain의 영상에는 다수의 포화된 영상을 포함하고 있고 Newsweek의 입력 영상 역시 R, G, B의 성분중 하나가 포화된 영상들로 이루어져 있으며 이는 일반적인 실제 영상에서 보이는 현상이다. 표 Ⅴ에 나타난 결과는 표 Ⅲ과 표 Ⅳ에 각각 나타낸 결과를 종합하여 제시한 것이다. 결과적으로 제안하는 알고리듬은 기존의 알고리듬에 비해 우수한 성능을 보이며 만족할 만한 정합율을 보임을 알 수 있다.

Ⅳ. 결 론

본 논문에서는 컬러 영상 데이터베이스에서 원하는 영상을 검색하기 위한 방법으로서 영상의 그래프 형식의 표현을 통한 정합 방법을 제시하였다. 또한 입력영상에 대한 조명의 변화의 영향을 줄이기 위해 새로운 컬러 균일성 보정 알고리듬을 제시하였다. 제시하는 컬러 균일성 알고리듬은 입력영상을 비교하는 모델영상이 취득된 조명조건과 같은 조명으로 변화시켜 조명의 변화에 관계없이 원하는 영상을 검색할 수 있었다. 다음으로 각 영상들은 그래프 형식 표현을 통해 영상의 의미있는 색들을 하나의 노드로 추상화하고 두 노드가 의미하는 두 색의 인접도를 노드간을 잇는 에지의 값으로 보아 그래프 정합을 통해 영상간의 유사도를 측정하였다. 실험결과에서 보인 바와 같이 제안하는 알고리듬은 기존의 알고리듬에 비해 우수한 정합성능을 나타냈으며 보다 큰 규모의 영상 데이터베이스에 대해서도 적용이 가능하다. 본 논문에서 제시한 그래프 형식의 영상표현기법은 노드정보와 에지정보로서 여러 가지 다른 정보를 포함시킬 수 있기 때문에 확장이 용이하고 추후 문맥기반(context-based) 영상 검색기법의 바탕이 될 수 있을 것이다.

표 3. Swain의 입력영상에 대한 정합결과(입력 영상 32개, 모델 영상 100개)

정합 알고리듬	순위 1	순위2	순위3	순위4이하	정합 성공율	평균 정합율
제안하는 알고리듬	29	2	1	0	0.9063	0.9981
Swain의 알고리듬으로 정합	27	3	1	1	0.8438	0.9962
Funt의 알고리듬으로 정합	22	2	4	4	0.6875	0.9865

표 Ⅳ. Newsweek 입력영상에 대한 정합결과(입력 영상 18개, 모델 영상 100개)

정합 알고리듬	순위1	순위2	순위3	순위4이하	정합 성공율	평균 정합율
제안하는 알고리듬	17	1	0	0	0.9444	0.9983
Swain의 알고리듬으로 정합	2	3	0	13	0.1111	0.6515
Funt의 알고리듬으로 정합	6	2	1	9	0.3333	0.8603

표 Ⅴ. 전체 입력영상에 대한 정합결과(입력 영상 50개, 모델 영상 100개)

정합 알고리듬	순위1	순위2	순위3	순위4이하	정합 성공율	평균 정합율
제안하는 알고리듬	46	3	1	0	0.92	0.9982
Swain의 알고리듬으로 정합	29	6	1	14	0.58	0.7412
Funt의 알고리듬으로 정합	28	4	5	13	0.56	0.9117

참 고 문 헌

[1] W. Niblack, R. Berber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic, and P. Yanker, "The QBIC project : Querying images by content using color, texture, and shape," *SPIE 1908, Storage and Retrieval for Images and Video Dbases,* Feb. 1993

[2] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision,* vol. 7, No. 1, pp. 11-32, 1991

[3] B. Funt and G. Finlayson, "Color constant color indexing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 5m pp. 522-529, May. 1995

[4] L. T. Maloney and B. A. Wandell, "Color constancy : a method for recovering surface spectral reflectance," *Journal of Optical Society of America* A. vol. 3, no. 1, pp 29-33, 1986.

[5] B. A. Wandell, "The synthesis and analysis of color images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 9, no. 1, pp. 2-13, 1987.

[6] J. Ho, B. V. Funt and M. S. Drew, "Separating a color into illumination and surface reflectance components : theory and applications," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, no. 10, pp. 966-977, Oct. 1990.

[7] D. A. Forsyth, "A novel algorithm for color constancy," *International Journal of Computer Vision,* vol. 5, no. 1, pp.5-36, 1990.

[8] M. A. Stricker, "Color and geometry as cues for indexing," *Technical Report CS92-22 Univ. of Chicago,* Nov. 1992.

[9] J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner, and W. Niblack, "Efficient color histogram indexing for quadratic form distance functions," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 17, no. 7, pp. 729-736, July. 1995.

[10] M. Stricker and M. Swain, "The capacity of color histogram indexing," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,* vol. 2, pp. 704-708, 1994.

[11] W. K. Pratt, *Digital Image Processing,* John Wiley & Sons, 1991.

[12] J. C. Russ, *The Image Processing Handbook,* 2nd edition, CRC Press, 1995.

저 자 소 개

박 인 규
1972년 2월 28일생
1995년 2월 서울대학교 제어계측공학과 졸업(공학사).
1995년 3월- 현재 서울대학교 제어계측공학과 석사과정 재학중.
*주관심 분야는 컴퓨터비젼, 컴퓨터 그래픽스등임.

윤 일 동
1965년 7월 21일생.
1989년 2월 서울대학교 제어계측공학과 졸업(공학사).
1991년 2월 서울대학교 제어계측공학과 졸업(공학석사).
1991년 3월- 현재 서울대학교 제어계측공학과 박사과정 재학중.
*주관심 분야는 영상 신호처리, 컴퓨터비젼, VLSI 신호처리등임.

이 상 욱
1949년 8월 11일생.
1973년 2월 서울대학교 전기공학과 졸업(공학사).
1976년 5월 미국 Iowa State Univ.(공학석사),
1980년 1월 미국 Univ. of Southern California 졸업(공학박사).
1973년 3월 - 1974년 8월 한국 과학 기술 연구소 (KIST) 공중선 연구실 연구원.
1980년 1월 - 1981년 9월 미국 General Electric Co. 연구원.
1981년 10월- 1983년 3월 미국 M/A-COM 연구소 선임 연구원.
1983년 4월 - 1995년 11월 서울 대학교 제어계측공학과 조교수, 부교수, 교수.
1995년 12월- 현재 서울 대학교 전기공학부 교수.