

색상 서열 비교를 통한 영상의 유사도 분석 기법

정인준, 우균
부산대학교 컴퓨터공학과
e-mail:spd1335@nate.com

Method of Image Similarity Analysis Using Sequence Alignment of Colors

Jung In-Joon, Woo Gyun
Dept of Computer Science and Engineering, Pusan National University

요 약

영상처리를 이용한 영상간의 유사도 비교 기법은 영상의 검색 및 영상의 자동 인식 등을 위한 연구로 최근 각광받고 있다. 최근 영상 처리 기법은 화소의 질적 향상 및 처리시간 최적화, 효율적인 특정 요소의 추출 등 다양한 방법으로 시도되고 있다. 특히, 영상의 유사도 비교는 유사 영상 검색과 같은 경우에 많이 쓰인다. 영상의 유사도를 비교하기 위한 기법으로는 영상 데이터의 특징에 따라 대상 영역을 여러 영역으로 나누는 영역분할 기법과 군집화, 퍼지, 유전자 알고리즘 등이 있다. 본 논문에서는 영상을 HSV 색공간으로 변환한 후 색상 값에 대하여 전역 정렬 기법을 사용하는 유사도 측정 방법을 제시한다. 전역 정렬 기법은 유전자 서열 비교 기법 중 하나로서 두 유전체의 유사도를 측정하는데 사용된다. 유사도 측정 효율을 높이기 위해 색상 값을 8단계로 양자화하여 영상의 서열을 생성하였다. 실험결과 제시한 방법을 영상 회전이나 대칭, 글자 삽입 등의 간단한 연산에 크게 영향을 받지 않는 것으로 드러났다.

1. 서론

영상처리를 이용한 영상간의 유사도 비교 기법은 영상의 검색 및 영상의 자동 인식 등을 위한 연구로 최근 각광받고 있다. 인터넷상의 영상 검색에서는 단순한 영상의 검색을 넘어 영상간의 유사도 비교를 이용한 유사 이미지 검색 기능은 흔히 사용되고 있다. 그리고 생산성 향상 및 작업공정의 자동화를 위한 물체의 인식에서 영상간의 유사성을 이용하는 기법이 일반적으로 사용된다. 이때 영상간의 유사도 비교 알고리즘이 전체 프로그램의 성능을 결정짓는다.

최근 영상의 유사도 비교는 다양한 방법으로 연구되고 있다[1]. 영상의 시각적 정보를 추출하여 영상의 유사도를 비교하여 분류하는 방법으로 정보추출과 구성이 쉬운 색, 질감을 데이터로 이용하는 방법이 활발하게 연구되고 있다. 이러한 영상의 시각적 정보를 이용하는 방법은 이미지가 내포하고 있는 의미적인 정보를 추출하여 이용하기에는 한계가 있다. 이러한 방법은 이미지가 가지고 있는 저차원의 시각적인 정보를 추출하여 이용하므로, 이미지가 가지고 있는 의미적인 정보 추출이 정확하지 않다[6].

본 논문에서는 유전자 서열 비교 알고리즘인 전역정렬 기법[7]을 이용하여 영상의 유사도를 분석한다. 유전자

알고리즘의 전역정렬 기법은 서로 동일한 종류의 핵산이나 단백질 서열을 이어서 비교하는 기법이다. 동일한 기능을 가진 단백질이나 핵산으로 높은 점수가 나타내는 경우에는 비교한 부분과 유사한 부분을 나타내게 된다. 영상간의 서열을 구할때에는 영상의 색상 영역만을 이용하며, 색상 영역을 양자화하여 서열을 만든다. 만들어진 서열을 전역 정렬 기법을 이용하여 비교한다.

2. 서열 정렬

서열 정렬(sequence alignment)은 단백질 서열이나 핵산서열 사이의 상관관계를 나타내는 방법 중 하나이다. 단백질의 서열간의 관계는 서열들이 기능적으로나 진화적으로 어느 정도 연관성이 있는지를 나타낸다. 서열 정렬 기법에서 전역 정렬(global alignment)과 국부 정렬(local alignment)이 있다. 전역정렬과 국부정렬은 서로 다른 기준에 의해 다른 방식의 계산을 해서 다른 결과를 보여주기 때문에, 적용된 알고리즘이 서로 다르고, 사용되는 영역도 다르다[3, 4].

서로 동일한 종류의 핵산이나 단백질 서열을 전체적으로 비교하여 최대의 상동성(homology)이 나타나도록 정렬하는 경우, 그러한 상동성을 전역 상동성(global homol-

ogy)이라 하며, 전역 상동성을 검사하는 정렬을 전역 정렬이라 한다. 전역 정렬 기법을 이용하여 단백질의 시퀀스간의 유사도를 측정하였을 경우 점수가 높다면 두 유전자간에 기능상의 유사성이 있을 가능성이 높다고 본다.

두 서열의 어떤 부분 서열이 높은 상동성을 가지는가를 나타내기 위해 정렬하는 경우, 그러한 상동성을 국부 상동성(local homology)이라 하고, 그 정렬을 국부 정렬이라 한다. 기능적으로 일부분만이 관련되어 있으며, 진화적으로도 일부분만이 상동성을 보이는 경우에는, 국부 정렬을 실행하는 것이 효과적이다. 또한 전역 정렬 기법의 점수가 낮아서 쉽게 상동성을 나타내지 못하지만, 일부분에서 높은 점수를 보이는 경우도 국부 정렬을 실행하는 것이 효과적이다.

3. HSV 색공간

HSV 색공간이란 색을 표현하는 방법 중 한가지이다. RGB 색공간과 HSV 색공간은 그림 1과 같은 차이를 보인다. HSV 색공간에서는 그림 1과 같이 색상(Hue), 채도(Saturation), 명도(Value)의 좌표를 써서 특정한 색을 나타낸다[5]. 색상의 범위는 0~360°이고, 채도와 명도의 범위는 0~100%이다.

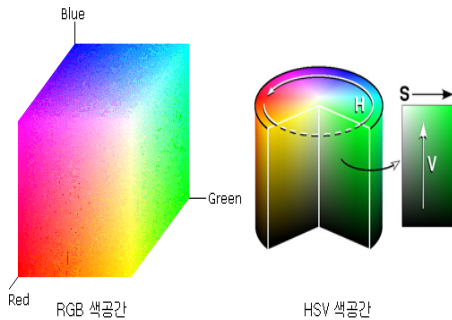


그림 1. RGB 색공간과 HSV 색공간

본 논문에서는 영상의 유사도 분석을 위한 전처리로 RGB 색공간을 가지는 영상을 HSV 색공간으로 변환하여 이용하였다. 영상에서 밝기와 채도 성분은 많은 차이를 보이지만 전체적으로 영상이 유사한 경우가 있다. 같은 사물을 촬영하더라도 다른 카메라로 촬영하여를 경우 대무문 밝기와 채도의 성분값이 차이가 나게 된다. RGB 색공간의 영상을 HSV 색공간으로 변환하여 영상의 유사도를 비교하는데 이용할 경우 명도와 채도에 관계없이 같은 색상값만을 이용한 비교가 가능하다.

4. 구현

4.1 HSV 변환

RGB 색공간에서 HSV로 변환하기 위한 과정은 다음과 같다. 우선 픽셀의 R, G, B 세 값을 최솟값이 0, 최댓값이 1인 값으로 변환한다. 그리고 RGB의 최댓값과 최솟값을 구한다. 최댓값과 최솟값의 차를 δ 라고 할 때, 색상

값은 R이 최댓값일 때 $((G-B) / \delta) * 60$ 이고, B가 최댓값일 때 $((B-R) + 2.0 / \delta) * 60$, 그리고 G가 최댓값일 때 $(B-R) + 2.0 / \delta) * 60$ 이다. 그리고 만약 색상 값이 0보다 작을 경우 값에 360을 더해준다. 본 논문에서는 색상값만을 연산에 이용하기 때문에 채도값과 명도값은 변환하지 않는다.

4.2 색상영역 양자화

RGB 색공간에서 HSV 색공간으로 변환을 하면 영상의 색상 범위는 0~359가 된다. 즉, 360가지 색상이 사용된다. 이러한 색상의 범위를 비교가 가능하도록 시퀀스를 만들기 위해서는 양자화한다. 논문에서는 색상의 범위인 0~359를 균일하게 8단계로 나누어 양자화 하였다. 그림 2는 원본 그림(좌측)과 양자화된 그림(우측)을 비교하여 보여준다.

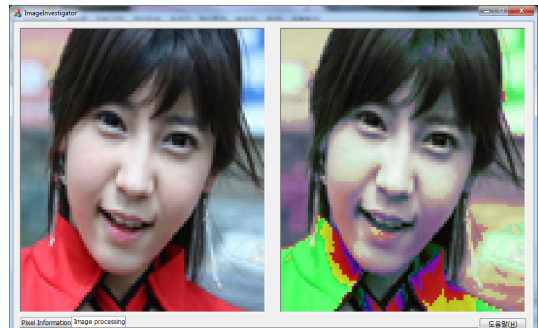


그림 2. 색상영역을 양자화한 그림

4.3 시퀀스 만들기

양자화된 영상 픽셀의 색상 값에 하나의 값을 대입하여 시퀀스를 만든다. 영상 영상의 색상은 8가지 색상으로 양자화되었으므로 문자는 A~H까지의 8가지 문자를 사용한다. 본 논문에서는 영상을 입력하여 입력된 영상의 시퀀스를 그림 3과 같은 텍스트 파일로 출력하여 사용하였다.

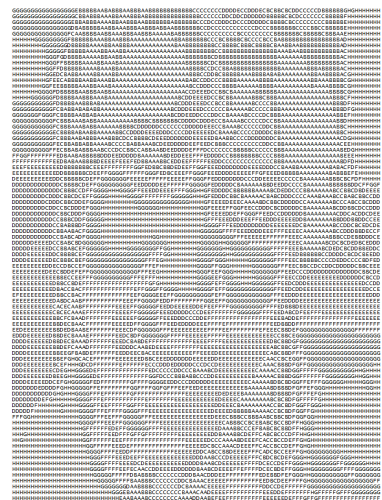


그림 3. 영상의 시퀀스

4.4 유사도 비교

양자화된 영상의 시퀀스를 비교한다. 전역 정렬 방법을 이용하였다. 전역 정렬 방법은 전체 시퀀스의 유사도를 이용하는 방법으로서 아래 그림 4와 같이 비교한다. 전역 정렬 기법은 두 개의 시퀀스가 비슷할수록 점수가 높아진다. 그림 4는 매치시 8점, 갭에서 -3점 미스매치에서 -6점을 주고 16자리의 시퀀스를 비교한 예제이다.

0	G	G	G	G	G	G	C	B	B	B	B	A	B	B	B	B	B	
0	0	-6	-12	-18	-24	-30	-36	-42	-48	-54	-60	-66	-72	-78	-84	-90	-96	-102
G	-6	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22	-25	-28	-31	-34	-37	-40
G	-12	5	16	13	10	7	4	1	-2	-5	-8	-11	-14	-17	-20	-23	-26	-29
G	-18	2	13	24	21	18	15	12	9	6	3	0	-3	-6	-9	-12	-15	-18
G	-24	-1	10	21	32	29	26	23	20	17	14	11	8	5	2	-1	-4	-7
D	-30	-4	7	18	29	26	23	20	17	14	11	8	5	2	-1	-4	-7	-10
B	-36	-7	4	15	26	23	20	17	14	11	8	5	2	-1	-4	-7	-10	-13
B	-42	-10	1	12	23	20	17	14	11	8	5	2	-1	-4	-7	-10	-13	-16
B	-48	-13	-2	9	20	17	14	11	8	5	2	-1	-4	-7	-10	-13	-16	-19
B	-54	-16	-5	6	17	14	11	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22
A	-60	-19	-8	3	14	11	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22	-25
B	-66	-22	-11	0	11	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22	-25	-28
B	-72	-25	-14	-3	8	5	2	-1	-4	-7	-10	-13	-16	-19	-22	-25	-28	-31
B	-78	-28	-17	-6	5	2	-1	-4	-7	-10	-13	-16	-19	-22	-25	-28	-31	-34
B	-84	-31	-20	-9	2	-1	-4	-7	-10	-13	-16	-19	-22	-25	-28	-31	-34	-37
B	-90	-34	-23	-12	-1	-4	-7	-10	-13	-16	-19	-22	-25	-28	-31	-34	-37	-40
B	-96	-37	-26	-15	-4	-7	-10	-13	-16	-19	-22	-25	-28	-31	-34	-37	-40	-43
C	-102	-40	-29	-18	-7	-10	-13	-16	-19	-22	-25	-28	-31	-34	-37	-40	-43	-46

```

GGGGGGCCBBBBB-BBBBB-
!- - ! ! ! X ! ! ! ! ! - ! ! ! ! ! -
G- - G G G D B B B B A B B B B B B C
Score : 94
Press any key to continue . . .
    
```

그림 4. 글로벌 얼라이먼트를 이용하여 시퀀스 비교한 예

영상에서는 시퀀스가 2차원으로 생성된다. 본 논문에서는 그림 5와 같이 한 행의 시퀀스를 상대 영상의 각 행의 시퀀스에 비교하였다.

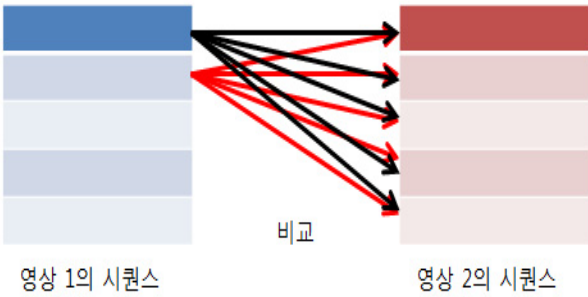


그림 5. 시퀀스 비교

영상의 시퀀스를 행별로 상대 영상의 시퀀스의 전체 행과 비교하였다. 시퀀스 비교는 (원 영상의 행의 수) × (상대 영상의 행의 수)만큼 이루어진다. 각 행을 비교하여 나온 최댓값들의 합을 최종 점수로 생성한다. 유사도 R은 수식 1과 같이 구할 수 있다.

$$R = (S_{now} - S_{min}) / (S_{max} - S_{min})$$

$$\text{if } \begin{cases} S_{max} = \text{최선의 점수} \\ S_{min} = \text{최악의 점수} \\ S_{now} = \text{현재의 점수} \end{cases}$$

(1)

수식 1의 R은 유사도이다. 만약, 매치시 점수가 8점이고 미스 매치시 점수가 -4점, 갭의 점수가 -2점일 때, 100개의 시퀀스 2개를 비교한다면 최선의 점수는 모두가 매치가 되는 800점이 된다. 그리고 최악의 점수는 -400점이다. 이 경우 만약 현재의 점수가 200점이라면 유사도는 $(200 - (-400)) / (800 - (-400))$ 으로 0.5이다. 즉, 유사도는 50%가 된다.

5. 실험 및 결과

5.1 실험

실험은 Intel사의 Core2Duo T9400 CPU와 6GB의 DDR3 RAM, 320GB의 S-ATA II방식의 HDD를 가진 랩톱 컴퓨터에서 이루어 졌다. 실험에 사용된 시스템의 운영체제는 Microsoft사의 Windows 7을 이용하였다. Microsoft사의 Visual Studio 2008 SP1 개발 툴을 사용하여, Visual C++ 프로그래밍 언어를 사용하여 실험을 위한 프로그램을 구현하였다.

실험에는 원본 영상 1가지와 비교를 위한 영상 5가지를 사용하였다. 사용한 영상은 그림 6~11과 같다.



그림 6. 원본 영상



그림 7. 뒤집힌 영상



그림 8. 글이 추가된 영상



그림 9. 전혀 다른 영상



그림 10. 흐리게 한 영상



그림 11. 더욱 흐리게 한 영상

그림 6는 원본 영상이며, 그림 7은 원본 영상을 다른 편집 없이 세로로 뒤집은 영상이다. 그림 8은 원본 영상에 글을 추가한 영상이다. 그리고 그림 9는 원본과는 전혀 다른 영상이다. 영상 10은 원본영상을 흐리게 만든 영상이

고, 영상 11은 영상 10보다 원본 영상을 더욱 흐리게 만든 영상이다. 영상을 흐리게 하기 위해서 가우시안 블러 (Gaussian Blur) 알고리즘을 사용하였고 그림 10와 11은 각각 가우시안 블러 값 1.0과 2.5를 적용하였다. 영상은 가로 100픽셀, 세로 100픽셀의 사이즈에 24비트 색공간을 사용하는 영상으로 통일 하였다.

5.2 결과

그림 6에서 11까지의 여섯 가지 영상을 원본 영상인 그림 6의 영상과 비교한 실험 결과는 표 1과 같다. 비교는 처음으로 원본영상인 그림 6과 그림 6을 비교하여 유사도가 정확하게 나오는지 확인하였고 그 다음 원본영상과 그림 7~11번 영상을 순서대로 각각 비교하였다. 시퀀스 비교할 때 시퀀스의 매치 점수는 8점으로 하였다. 그리고 미스 매치 점수는 -6점으로 하였고 갭은 -3점으로 하여 총 점수와 유사도를 계산하였다.

	일치 점수	유사도
그림6/그림6	80000	100.00%
그림6/그림7	8000	100.00%
그림6/그림8	64866	89.19%
그림6/그림9	18297	55.93%
그림6/그림10	58020	84.30%
그림6/그림11	43305	73.79%

표 1 실험 결과

원본영상과 원본영상을 비교하였을 때는 유사도가 100%가 나타나서 같은 영상임을 확인할 수 있다. 그리고 원본 영상과 뒤집힌 영상을 비교하였을 때 역시 유사도 100%로 같은 영상임을 확인할 수 있다. 시퀀스를 행별로 비교를 하기 때문에 영상 내에서 같은 행이 반드시 나타나게 되는 세로로 뒤집힌 영상에서는 100%의 유사도가 나타난다. 원본 영상에 글자와 같은 다른 영상이 일부분 덮여 썩어져 있을 경우에도 89.19%의 높은 유사도를 나타나서 유사한 영상임을 확인할 수 있었다. 영상을 흐리게 만든 영상과 비교를 하였을 때에도 흐림의 정도에 따라 다르지만 70%이상의 높은 유사도가 나타났다. 그리고 원본 영상과 전혀 다른 영상을 비교하였을 경우에는 55.93%의 낮은 유사도가 나타나 영상에서 유사한 부분이 적음을 확인할 수 있었다.

6. 결론

본 논문에서는 유전자 시퀀스 비교 방법을 응용하여 영상간의 유사도를 비교하는 실험을 하였다. 먼저 영상의 시퀀스를 구하기 위하여 색공간을 RGB 색공간에서 HSV 색공간으로 변환을 하였다. 그리고 구해진 색상값을 3단계로 양자화한 다음 각각의 색상에 A~H까지의 값을 대입하여 시퀀스를 완성하였다. 완성된 영상의 시퀀스를 이용하여 다른 영상의 시퀀스와 행별로 비교를 하였다.

실험 결과 원본영상을 다른 편집 없이 그냥 뒤집은 연산을 할 경우 유사도 100%로 정확히 원본 영상과 동일한 영상임을 인식하였다. 그리고 영상에 글과 같은 다른 잡음이 추가되더라도 높은 유사도를 보였다. 그리고 영상의 흐려짐이 심해짐에 따라 유사도나 낮아졌다. 하지만 영상을 흐리게 변환한 영상에서도 전혀 다른 영상보다 상대적으로 높은 유사도를 나타내었다.

실험 결과 유전자 시퀀스 비교 방법이 영상간의 유사도 비교에 적합함을 알 수 있었다. 본 논문에서 사용한 유전자 비교 방법을 사용한다면 유사한 영상을 찾거나 영상간의 유사도를 비교하는 등 여러 영역에서 사용될 수 있을 것이라 생각된다.

참 고 문 헌

- [1] Kim Donggi, Lee seonggyu, Lee Moonwook and Kang esok, "An Image Segmentation Method and Similarity Measurement Using Fuzzy Algorithm for Object Recognition", *대한기계학회논문집 A권*, Vol. 28, No. 2, pp. 109-213, 2004.
- [2] Yoon Dongwoo, Park Hyeyoung and Lee Kwanyong, "Image Data Classification using a Similarity Function based on Second Order Tensor", *정보과학회논문지 : 소프트웨어 및 응용*, Vol. 36, No. 8, pp. 664-672, 2009.
- [3] Kun-Mao Chao, Louxin Zhang and Andreas Dress, "Detecting Faces in Images: A Survey", *Sequence Comparison: Theory and Methods - SEQUENCE COMPARISON*, Springer, 2008.
- [4] Neil C. Jones and Pavel A. Pevzner, *An Introduction to Bioinformatics Algorithms*, Bradford Book, 2004.
- [5] Gary Rost Bradski and Adrian Kaehler, *Learning OpenCV*, O'REILLY, 2008.
- [6] Kwangsu Hwang, Hongryoul Yi and Pankoo Kim, "Auto-Classification of Annotated Images using Similarity between Concepts", *한국컴퓨터종합학술대회*, Vol. 34, No. 1(C), pp. 370-375, 2007.
- [7] Shuba Gopal, Anne Haake, Rhys Price Jones and Paul Tymann, *Bioinformatics A Computing Perspective*, McGraw-Hill Higher Education, 2009.