

닮은꼴을 이용한 새로운 동영상 디코딩 처리방법

진현수*, 한군희*

*백석대학교, 정보통신학부
e-mail:jhs1020@bu.ac.kr

New Moving Picture Decoding Process using Picture Resemblance

Hyun-Soo Jin*, Kun-Hee Han*

* Baek Seok University. Div. of Information & Communication

요약

본 논문에서는 MPEG2(Moving Picture Expert Group 2)비디오 디코더에서 영상데이터를 디코딩하는 기술에 관한 것으로, 종래의 동영상 처리기에서는 허프만 코드화 처리를 하여 정지영상을 부호화한 후 분할기반 부호화 방식의 동영상 처리기법을 사용하였으나 1프레임의 코드화시에도 많은 계산 알고리즘이 필요하고 이미지의 닮음정도를 객관적수치로 표현할 수 없으므로 비슷한 이미지를 중복 코드화 작업 및 전송 작업을 하는 결함이 있었다. 본 논문에서는 이를 해결하기 위하여 이미지의 닮음정도를 수치화하고 기준값 이하의 영상은 코드화 작업을 하지않고 계산된 영상 변화값에 의해 구현된 영상으로 대체하여 저장 및 전송하도록 하여 시각적으로 전혀 거부감이 없는 동영상을 구현하였다.

I. 서론

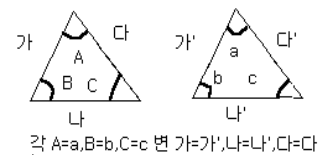
'80년대에 들어와 컴퓨터, 정보처리기술, 반도체기술의 눈부신 발전은 인간의 실생활에 많은 변화를 안겨 주었다, 특히 정보 이용의 차원은 기존의 정형화된 데이터 형태인 문자에만 의존하는 단계를 벗어나 실제 인간의 실생활에서 접하는 텍스트, 그래픽, 정지화상, 동화상, 음성, 음악등 다양한 형태의 뉴미디어 정보들을 있는 그대로 컴퓨터 및 컴퓨터 통신망 기술을 이용해 제공하고자 하는 노력이 이루어 지고 있다. 특히 화상의 저장, 표시 및 정보 교환을 원하는 경우, 화상정보를 있는 그대로 이용하여서는 엄청난 양의 데이터 저장공간, 매우 빠른 데이터 처리속도를 갖는 광대역 통신망이 요구되게 되어 정보 이용, 비용면에서 볼 때 동화상 정보 이용의 실용화는 거의 불가능하다고 할 수 있다.

다행히 이차원 화상신호에는 상당량의 중복성 정보가 포함되어 있어서 감축된 양의 정보로도 비교적 정확한 화상의 표시 및 화상 전송 통신선로의 대역폭 감소가 가능하다. 따라서 화상신호의 중복성을 이용한 동화상 신호의 데이터 압축기술은 화상 정보처리 시스템 실현을 위한 핵심 기술이라 할 수 있겠다.

II. 영상의 유사성

2.1 이미지 닮은꼴값

그림(Picture)의 닮음정도를 평가하는 작업은 지금까지는 기하적(Geometric)형태로서만 평가 하여서 그림 1과 같은 임의의 그림이 기하적으로 같을수 있다.



[그림 1] 닮은꼴 기하그림

그러나 단순한 영상(Image)차원에서 닮음을 계산하고자 할 때는 이 영상을 기하적(Geometric)적으로 분할하거나 복원하는 차원으로 해결할수가 없으므로 화소단위(Pixel)로 나뉘어 그림 2와 같이 어떤 화면이 본래의 화면과 더욱 많은 닮음정도의 화소를 포함하고 있는가를 나타내면 어떠한 형태의 영상도 서로 유사성이 있음을 발견 할 수가 있게된다.

영상(image)-1	영상(image)-2
① ② ③ ④	① ② ③ ④
① 0 0 0 0	① 1 1 1 1
② 0 0 0 0	② 1 1 1 1
③ 0 0 0 0	③ 1 1 1 1
④ 0 0 0 0	④ 1 1 1 1

[그림 2] 임의의 영상의 뒀음꼴

즉, 그림 2에서 이미지의 크기가 4x4화소 단위로 구성된 그림을 사용한다면 영상-1은 흑(Black)의 화소(Pixel)가 하나도 없는 흰색(White)의 화소를 나타내고 있고 영상-2는 반대로 흑의 화소만 나타나는 그림으로서 두 그림을 비교한다면 전혀 영상의 내용으로서는 관련이 없는 그림임을 알 수 있다. 이를 뒀음꼴 비교값이라는 수치로서 나타낸다면 무한대(∞)를 나타낸다.

2.2 두개의 2진 영상비교

위의 과정을 디지털 이진영상으로 비교하는 과정을 구체적으로 나타내면 다음 그림 3이다.

1화면	2화면	3화면
0 0 0 1 0 0 0	0 0 0 1 0 0 0	1 1 1 1 1 ①1
0 0 1 1 1 0 0	0 0 1 1 1 1 0	1 1 1 1 ①0 ①
0 1 1 1 1 1 0	0 1 1 1 1 1 0	1 1 1 1 1 ①1
1 1 1 1 1 1 1	1 1 1 1 1 1 1	1 1 1 1 1 1 1
0 0 0 0 0 0 0	0 0 0 0 0 0 0	1 1 1 1 1 1 1
3화면	1,3화면 비교값	
0 0 0 1 0 0 0	①0 0 1 0 ①1	
1 0 1 1 1 1 0	0 ①1 1 ①0 ①	
0 1 1 1 1 1 0	①1 1 1 1 ①0	
1 1 1 1 1 1 1	1 1 1 1 1 1 1	
0 0 0 0 0 0 0	1 1 1 1 1 1 1	

[그림 3] 영상의 비교값

첫 번째, 동영상 구현하기 위하여 우선 정지영상 코드를 한다. 기본이 되는 영상(기준화면을)도 1에서 가장 첫 번째 입력되어지는 BS(Beat Stream)영상으로 정하면 도2와 도3의 1화면으로 나타내어질수 있는데, 기준 화면을 도1의 12번 항목인 프레임 메모리에 16x16영상면을 데이터화 한다. 화소(pixel)가 있을 경우'1'로 데이터화 하고 화소가 없을 경우 '0'로

데이터화 하여 전에 데이터화 한 화면중 기준화면 다음에 들어오는 화면도 기준화면과 비교하기 위하여 16x16크기로 분할하여 기준화면과 비교되어지는 데 우선, 그림3에서와 같이 기준화면인 1화면과 다음번 화면인 2화면을 중첩하여 화소가 중첩이 될 경우 '1'로 데이터화하고 중첩이 되지 않을 경우 '0'로 데이터화하여 비교화면 데이터를 그림3의 1,2화면 비교값, 1,3화면 비교값 처럼 생성한다. 그림3에서는 화면을 7x7로 나타내어 실행하였다.

III. 이진 영상의 비교값 구하기

실제로 2진영상의 비교값을 구하여 보면 그림 4(a),(b)와 같이 기준 영상 데이터를 1화면이라하고 비교할 데이터 화면을 2,3화면이라 할 때 비교된 1,2 비교 데이터 화면과 1,3비교 데이터화면의 각 화소 데이터 가운데

1화면	2화면	3화면
0 0 0 1 0 0 0	0 0 0 1 0 0 0	0 0 0 1 0 0 0
0 0 1 1 1 0 0	0 0 1 1 1 1 0	0 0 1 1 1 1 0
0 1 1 1 1 1 0	0 1 1 1 1 1 0	0 1 1 1 1 1 0
1 1 1 1 1 1 1	1 1 1 1 1 1 1	1 1 1 1 1 1 1
0 0 0 0 0 0 0	0 0 0 0 0 0 0	0 0 0 0 0 0 0
4화면	5화면	
0 0 0 1 0 0 0	0 0 0 1 0 0 1	
1 0 1 1 1 1 0	1 0 1 1 1 1 0	
0 0 1 1 1 1 0	1 0 1 1 1 1 0	
1 1 1 1 1 1 1	1 1 1 1 1 1 1	
0 0 0 0 0 0 0	0 0 0 0 0 0 0	

(a)

1화면	2화면	3화면
0 0 0 1 0 0 0	1 1 1 1 1 1 1	1 1 1 1 1 ①1
0 0 1 1 1 0 0	1 1 1 1 1 1 1	1 1 1 1 ①0 ①
0 1 1 1 1 1 0	1 1 1 1 1 1 1	1 1 1 1 1 ①1
1 1 1 1 1 1 1	1 1 1 1 1 1 1	1 1 1 1 1 1 1
0 0 0 0 0 0 0	1 1 1 1 1 1 1	1 1 1 1 1 1 1

3화면	1,3화면 비교값
①0 0 0 1 ① 1	①1 1 1 1 1 ①'
0 ①0 1 ① 1 ①	0 ①1 1 1 ① 0
①1 1 1 1 ① 1	①1 1 1 1 1 ①
1 1 1 1 1 1 1	1 1 1 1 1 1 1
0 0 0 0 0 0 0	1 1 1 1 1 1 1

[그림 3] 영상의 뒀음꼴 비교값 구하기

이때 그림4에서는 다음 계산값을 통해 기준 화면과 2,3화면과의 절대 뉘른꼴 계산을 다음과 같이 수행한다, 2화면의 뉘른꼴계수는 1,2화면 비교값에서 등급(class)2로 인한 화소인 ①이 4곳 있고 각각 떨어져 있는 거리가 1x+1y이므로 총거리로 인한 값은 4x(1x+1y)=4x+4y이고 1,3화면의 뉘른꼴 계수는 등급(class)2의 화소(pixel)군이 2개가 생기고 두 계수값을 비교하면 3화면이 2화면보다 거리계수값이 크므로 1화면과 덜 뉘른었다는 것을 나타내고 있다. 화면당 좌표값을 해당 화면의 '절대 뉘른꼴치'라 설정한다. 지금 까지 과정을 도1의 영상 비교기가 수행하게 된다. 도5에서는 위의 절대 뉘른꼴 계산 과정을 순서도로 표시하였다.

IV. 동영상 처리과정

위와 같은 처리과정을 거쳐 동영상으로 복원하기 위해서 위의 과정을 연결하여 진행 한다면 세 번째, 이와 같은 정지영상의 뉘른꼴값을 계산한 후 ,그림6의 13번 항목인 동영상 복원기 실행과정을 그림 5에서 설명하였다, 기준화면인 1화면 외에 4개의 정지화면이 입력이 되어 '0'이고 1,3화면의 뉘른꼴 계산값은 4x+4y이고 1,4화면의 뉘른꼴 계산값은 15x+2y이어서 화소의 평균 변화값은 0+(4x+4y) +(17x+2y)/3=7x+2y 이고 화면전송의 가부판정 임계값으로 쓰이는 수치는 (20x+20y)/(7개의 화소/16개의 화소)=15x+15y이므로 화면2에서 화면 4까지는 직접 전송하지 않고 평균변화값의 변화분으로 대체된 화면으로 복원하게 되고, 나머지 1,5화면의 뉘른꼴 계산은 등급1인 화소 ①이 5곳이 있으므로 거리값 10x+2y이고 등급 2인 화소를 위해서 위의 과정을 연결하여 진행 하면된다. 세 번째, 이와 같은 정지영상의 뉘른꼴값을 계산한 후 ,그림6의 13번 항목인 동영상 복원기 실행과정을 그림 5에서 설명하였는데, 기준화면인 1화면 외에 4개의 정지화면이 입력이 되어 '0'이고 1,3화면의 뉘른꼴 계산값은 4x+4y이고 1,4화면의 뉘른꼴 계산값은 15x+2y이어서 화소의 평균 변화값은 0+(4x+4y) +(17x+2y)/3=7x+2y이고 화면전송의 가부판정 임계값으로 쓰이는 수치는 (20x+20y)/(7개의 화소/16개의 화소)=15x+15y이므로 화면2에서 화면 4까지는 직접 전송하지 않고 평균변화값의 변화분으로 대체된 화면으로 복원하게 되고, 나머지 1,5화면의 뉘른꼴 계산은 등급1인 화소 ①'가 2곳 있으므로 거리값 30x(1x+1y)=30x+30y이므로 총 뉘른꼴 계산 값은

40x+32y으로 임계값을 초과하여 직접 전송하여야 하므로 다음 기준화면으로 사용하기 위하여 프레임메모리에 저장되어진다.

	1화면	2화면
(0,0)	(7,0)	
0 0 0 1 0 0 0	0 0 0 1 0 0 1	
0 0 1 1 1 0 0	0 0 1 1 1 1 0	
0 1 1 1 1 1 0	0 1 1 1 1 1 0	
1 1 1 1 1 1 1	1 1 1 1 1 1 1	
0 0 0 0 0 0 0	0 0 0 0 0 0 0	
(0,7)	(7,7)	

	3화면	4화면
0 0 0 1 0 0 0	0 0 0 1 0 0 1	
0 0 1 1 1 1 0	0 0 1 1 1 1 0	
0 1 1 1 1 1 0	0 0 1 1 1 1 0	
1 1 1 1 1 1 1	0 1 1 1 1 1 1	
0 0 0 0 0 0 0	0 0 0 0 0 0 0	

(a)

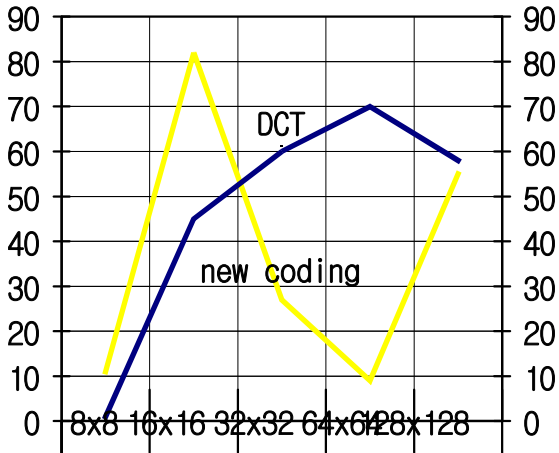
[그림 4] 동영상 처리화면 결정도

메모리 사이즈는 전송 팩터에서 1×10^6 으로 나누면 각 블록 사이즈별로 메모리 사이즈가 나오게 된다

블록의 수가 많을수록 즉 보내야할 화상의 크기가 클수록 메모리 사이즈의 축소되는 면이 작다고 할 수가 있다. 따라서 많은수의 화상을 많은양의 화상크기로 전송할 경우가 가장 뉘른꼴 전송방법에서 유리하다고 볼수 있다.다음은 각 방법별로샘플링 히스토그램을 관찰한 결과가 그림11에 나타내었는데 블록 사이즈에 관계없이 일정한 크기변화를 보이는 것이 뉘른꼴 디코딩 방법으로서 많은 블록 즉, 많은 화면일수록 샘플링수가 줄어들음을 볼수 가 있다.

V. 결론

본 논문에서는 기존의 화면당 모든 화소들을 인정하여 보내야할 처음의 화면부터 마지막 화면까지 각 픽셀 하나 하나까지 트랜스 폼 하는 방식을 DCT 방식이라하여 정지화상 및 동영상에 이르기까지 사용하여왔으나 이방법은 알려진 대로 화상의 존재를 인정하는 것이 아니고 즉 화상이 아닌 픽셀을 인정하는 방식이라하여 아무리 같은 화상이라 할지라도 모



두 디코딩 하는 방법을 사용하였으나 이방법은 매우 반복되어져서 보내 저장할 화면에서는 비교할수 있는 방법이 없어 무조건 다 전송하는 원시적인 방법이라 할 수가 있다.그러나 새로운 디코딩 방식은 한 픽셀을 설정하여 기준 화면으로 정한후 임계값으로 결정한 값을 비교하여 모든 화면의 픽셀들을 디코딩 하는 방법이다.화면의 특성상 중복되는 픽셀들을 임계값으로 결정하여 그 값 이하는 유사성 화면으로 대체되는 방법을 통하여 메모리의 절감 효과를 가져왔다. 기존의 DCT방법보다 14%정도의 샘플링 절감 효과를 가져와서 정지영상및 동영상의 전송시 다소 복잡한 알고리즘의 구축과정만 발전시킨다면 매우 훌륭한 영상 전송 방법이라 할수 있다.

참고문헌

[1] P.H. Westerink,D.E.Boekee,J.Biemand and J.W.Woods,"Subband Coding of Image Using Vector Quantization",IEEE Trans.Commun., vol.36,pp.713-719,June 1988

[2] P.A.Chau,T.Lookabaugh,and R.M. Gray,"Entropy-Constrained Vector Quantization", IEEE Trans. Acoust.,Speech and Signal Process., vol. ASSP-37,pp.31-42,Jan.1989

[3] A.Gersho,"Asymptotically Optimal Block Quantization,"IEEE trans.Inform.Theory,vol. IT-27,pp.280-291,May 1981.

[4] J.W.Modestino and Y.H.Kim,"Adaptive Entropy-Coded Predictive Vector Quanti - zation of Images,"to appear in IEEE Trans.

Acoust.,Speech, and Signal Process.,June 1992.

[5] Y.H.Kim and J.W.Modestino,"Adaptive Entropy-Coded Pruned Tree-Structured Predictive Vector quantization of images," to appear in IEEE Trans. on Commun.

[6] J.B.Anderson,T.Aulin,andC.E.Sundberg, Digital Phase Modulation,Plenum Press, New York,1986