PVR 에서 실시간 브라우징을 위한 클러스터링

장경훈, 이동호

*한양대학교 전자컴퓨터공학부

Image Clustering for Real-time Browsing in DTV PVR

kyoung-Hoon Lee*, Dong-Ho Lee*

*School of Electrical and Computer Engineering of Hanyang Univ

khjang@image.hanyang.ac.kr, dhlee@image.hanyang.ac.kr

요 약

현재 수 많은 공중파 채널을 비롯하여 각종 케이블 TV 와 위성 TV 등의 채널이 대량으로 디지털화 되어 가고 있지만 각국의 디지털 TV 의 규격은 MPEG2 라는 표준 규격으로 통일 되어 우리에게 전해지고 있다. 이런 수 많은 동영상 정보들은 디지털이라는 특성상 기록과 보관이 용이하므로 사용자는 수 많은 정보를 저장, 관리 할 수 있게 되었다. 따라서 방대한 데이터의 바다 속에서 원하는 정보를 검색하는데 있어서 기존의 방식 즉 일정 속도를 가진 검색 엔진을 사용하는 것 보다는 MPEG2 규격의 표준들을 보다 효율적으로 이용하여 사용자들이 동영상 정보 중에 이동하기 원하는 부분으로 이동하기 위한 비선형 검색엔진을 소개한다. 이를 이용한 PVR(Personal Video Recorder)은 현재 hardware 부분은 완료되었고 일부 software 엔진을 적용하여 개발 중이다.

I. 서 론

기존의 동영상 검색 방식은 전체 동영상을 일정한 시간 간격으로 해당하는 key 영상만을 찾아가는 방식이며 응용할 수 있는 부분은 시간 간격의 장,단만을 결정할 수 있을 뿐이다. 이러한 방식으로는 사용자가 동영상의 대략적인 내용을 확인하기 힘들 뿐 아니라 수 많은 의미 없는 key 영상의 남발이나 주요 key 영상을 지나칠 가능성이 있다. 이에 반하여 비선형 검색 방식은 카메라의 장면 전환 단위인 shot 을 찾아서 이를 동영상의 의미 있는 구성인 scene 으로 clustering 을 하여 각 scene 의 첫 shot 만을 key 영상으로 선정하여 사용자가 보다 나은 검색을 가능케 하며 화면의 복잡도나 중요성을 고려하여 브라우징 속도를 달리 할 수 있다. 또한 H/W 로 실시간으로 들어오는 동영상을 parsing 하여 저장과 동시에 scene change 여부와 브라우징 속도, 화면에 대한 정보등을 tag 로 만들어 같이 저장하게 하여 보다 빠르고 용이한 검색이 가능케 했다. I 장에서는 I,P,B picture 에서 필요한 정보와 추출 방법을 소개한다. 나날이 VLSI 기술의 발전으로 CPU 의 속도와 기능이 향상되고 있지만 화면데이터를 생성하는 역할을 담당하는 부분은 MPEG2 VLD (Variable Length Coding) parsing 의 많은 load, 계산의 용이성과 Bandwidth 문제, 추출할 정보의 양자화 등을 고려할 때 Hardware 로 설계하는 것이 더 나은 선택이며 이에 대한 구현 방법을 소개한다. 양자화된 각 화면의 데이터를 이용하여 shot change detection 과 Scene clustering 을 하기 위한 알고리듬 적용은 차후 응용 가능성에 대비하여 software 를 이용하여 설계되며 그 방법은 II 장에 소개한다. 이는 보다 나은 알고리듬의 개발과 동영상의 내용에 따라 다른 장면 변환 검출 방법이 적용되어 져야 하기 때문이다. III 장에서는 hardware 로 추출된 화면정보와 software 장면 추출 알고리듬을 이용하여 실제 simulator 를 이용한 non linear 브라우징 결과를 보여준다.

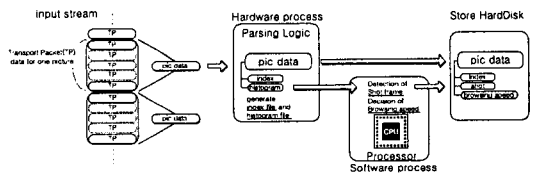

그림 1. PVR 중 clustering 의 개략도

II. 화면정보 추출 H/W 설계

H/W part 는 MPEG2 stream 을 최소한의 logic 과 bandwidth 로 parsing 하기 위하여 각 picture 마다 다른 logic 으로 설계 되지만 공통적으로 picture size, frame number, picture type 과 picture structure 를 parsing 한다. 또한 각 화면의 주요 데이터인 DC 값, 모션벡터와 매크로블록타입들은 매크로블록 단위로 측정되어 각각의 단위에 따라 양자화 시켜 histogram 화 되어 S/W load 를 줄이게 된다. Logic 의 구현은 VHDL 을 이용하였다.

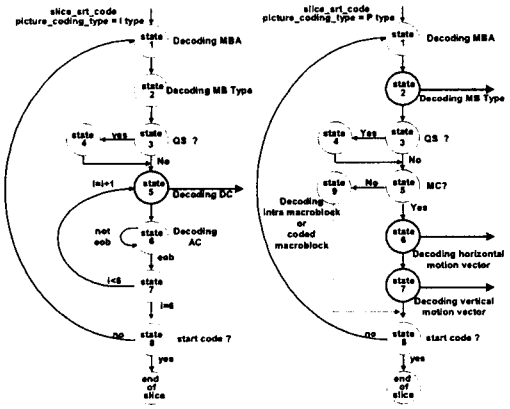

그림 2. Intra, Predicted picture 의 state machine

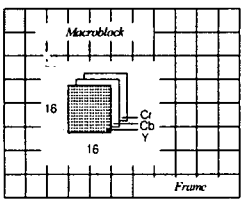

그림 3. 화면내의 매크로 블록

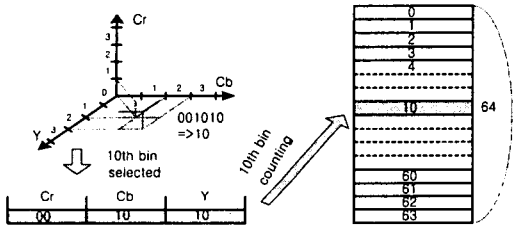

그림 4. DC 값의 양자화와 histogram 화

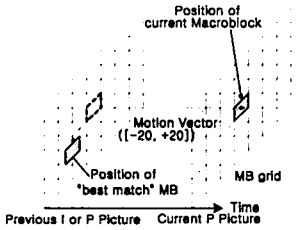

그림 5. 모션벡터의 추출

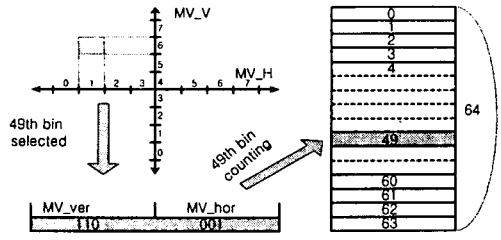

그림 6. 모션벡터값의 양자화와 histogram 화

I(Intra coded) picture 는 모든 매크로 블록이 움직임 보상 없이 표현되므로 모든 화면이 부호화 된다. 각 매크로 블록의 DC 값인 휘도(Y),색도(Cb,Cr)를 각각 4 개의 단위로 양자화 시킨 후 Y, Cb, Cr 을 축으로 하는 3 차원 histogram 화 하여 64 개의 메모리 번지에 저장한다. P(Predicitive) picture 는 순방향 예측을 이용하여 과거의 화면에서 움직임 보상을 하며, 이 picture 에서는 각 매크로 블록의 타입과 움직임 보상이 사용된 매크로블록에서의 모션벡터를 추출해 낸다. 모션벡터는 x 축, y 축이 각각 8 개의 단위로 양자화 되어 64 개의 2 차원 히스토그램화 된다. 또한 7 개의 매크로 타입은 그 개수를 count 하여 메모리에 저장된다. B(Bidirectionally predictive) picture 는 양방향 예측을 이용하여 과거와 미래의 화면에서 움직임 보상을 하며, 매크로블록 타입을 1~11 까지의 11 개로 양자화 한다.

이와 같이 양자화 된 데이터는 내부 메모리로 만들어 256X16(512byte) two-port ram 두 개를 통하여 실시간으로 parsing 되어 양자화된 데이터를 picture 별로 번갈아 가며 bin counting 을 하며 저장하고 저장이 끝난 화면의 데이터는 검색 엔진 S/W 에 전송된다.

표 1. 매크로 블록 타입의 히스토그램화

MB type	Predicted picture	Bi-Predicted picture
10		Intra, Quant
9		Bwd, Coded, Quant
8		Fwd, Coded, Quant
7		Interp, Coded, Quant
6	Intra, Quant	Intra
5	No Mc, Coded,Quant	Fwd, Coded
4	MC, Coded, Quant	Fwd,Not coded
3	Intra	Bwd, Coded
2	MC, Not coded	Bwd, Not Coded
1	No MC, Coded	Interp, Coded
0	MC, Coded	Interp, Not coed

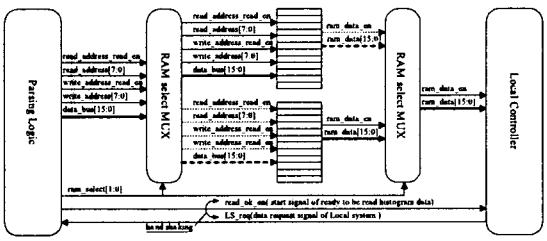

그림 7. 내부 메모리 인터페이스

III. S/W 에 의한 데이터 가공

Clustering 을 위하여 장면 전환 화면을 찾을 때 가장 먼저 화면의 DC 정보가 담긴 I picture 의 밝기차를 비교하는 것이 가장 효과적이며 단지 이 알고리듬만을 적용할 수도 있다. S/W 검색 엔진은 GOP(Group of Pictures)를 이끄는 I picture 와 I picture 간의 DC histogram 의 차

이가 사용자의 요구에 맞는 특정 임계치를 넘는 GOP 를 shot change 의 후보 구간으로 삼으며 그렇지 않은 경우에는 다음 GOP 에 대하여 알고리듬을 적용한다

$$Difference = \sum_{k=1}^{NB} | Hist(P_k) - Hist(C_k) |$$

NB : number of bins(64)
Hist(P_k) : k-th bin of color histogram for previous picture
Hist(C_k) : k-th bin of color histogram for current picture

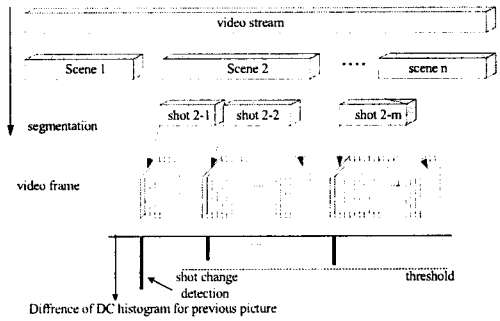

그림 8. 장면 변화 구간의 설정

더욱 더 자세한 화면 전환 구간을 찾기 위해 선정된 한 GOP 안에서 P picture 의 Intra coded 매크로 블록의 개수를 비교하여 특정 임계치를 넘는 P picture 사이를 장면 전환 구간으로 선정한다. P picture 라 하더라도 장면 전환 구간에서는 Intra coding 된 비율이 많으므로 이를 이용한 것이다. 마지막으로 선택된 구간의 B picture 에서는 미래의 영상에서 예측을 한 backward macroblock 의 비율이 가장 많다면 이전 화면과의 관계가 적은 것으로 추측 되므로 그 화면을 shot change 가 일어난 화면으로 결정한다. 하지만 대개의 shot change 는 I picture 에서 발생하므로 GOP 안에서 shot change 결정의 임계치는 더 엄격히 적용되어야 하며 특정 임계치를 넘지 않을 시에는 shot change 화면으로 선정하지 않고 GOP 의 I picture 가 선정된다. 이렇게 각 scene 단위로 clustering 된 구간의 브라우징 속도는 모션벡터의 크기에 의해 결정된다. 모션벡터를 사용하는 P,B picture 에서 작은 모션벡터를 이용하거나 이전 화면과 변화가 없어 그대로 가져온 매크로블록이 많다는 것은 상대적으로 intra coded 매크로블록과 큰 모션벡터를 이용하는 화면보다는 화면의 복잡도가 떨어진다는 것을 의미하며 브라우징의 속도를 빨리 할 수 있다.

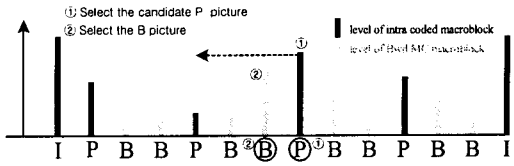

그림 9. GOP 안에서의 shot change 결정

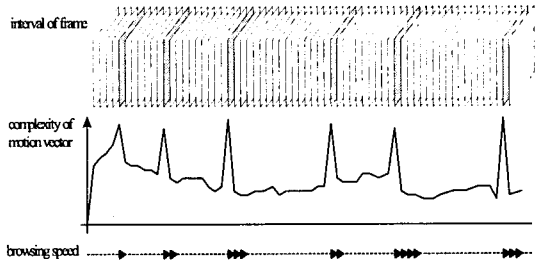

그림 10. 브라우징 속도의 결정

IV Clustering 결과

예제로 쓰인 그림 11 의 스트림은 1500 개의 영상으로 이루어진 뉴스의 일부분이다. 그림 12 는 연속되는 I picture 들의 DC 히스토그램 차이를 보여주는 과정이고 빨간 막대 그래프는 DC 히스토그램의 분포량을 보여준다. 실험에서 알 수 있듯이 한 scene 안에서의 차이 값은 새로운 scene 을 대표하는 shot 이 나왔을 때와는 많은 차이가 남을 알 수가 있고 검색엔진은 이를 Shot change 로 간주하고 새로운 scene 으로 결정하며 일정 임계값 이상의 차이가 일어나기 전까지는 동일 scene 으로 판단한다. 뉴스의 경우 주요한 scene 은 앵커가 나오는 shot 과 그 뒤에 따르는 기사로 볼 수 있는데 기사안의 shot change 로 발생하는 차이값에 비해 앵커 shot 이 나오며 발생하는 차이값이 월등히 큼을 알 수 있다.

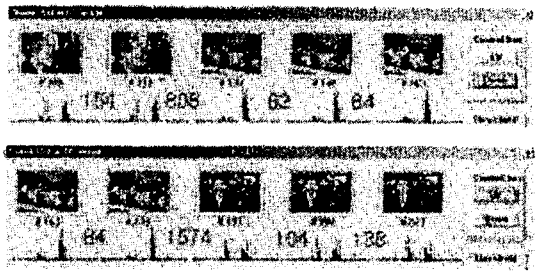

그림 12. DC 히스토그램의 비교

그림 11. 1500 frame 으로 이루어진 뉴스 프로그램의 주요 key frame (3 개의 주요 scene 존재)

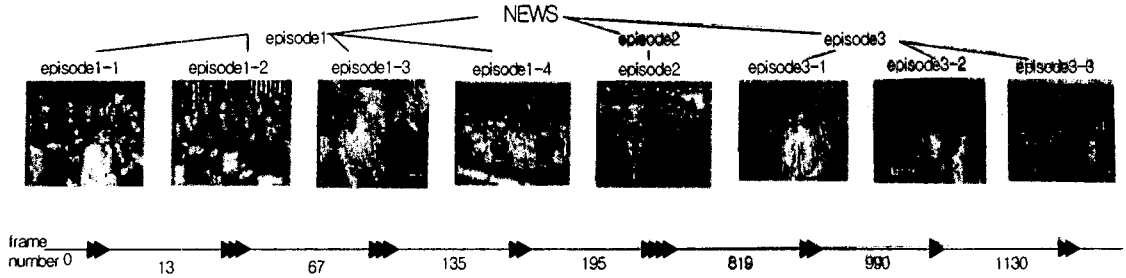

그림 13. clustering 과 각 episode 간의 브라우징 속도 결정

위의 shot change 결정 과정으로 뉴스는 크게 세 개의 scene 으로 나눌 수 있으며 각각의 브라우징 속도 결정 과정은 각 scene 을 대표하는 shot 이 시작하면 이전의 속도는 무시되고 intra coded 매크로 블록의 비율에 따라 새로운 속도가 결정된다. 특히 앵커 scene 에서는 화면의 변화가 거의 없으므로 대부분의 매크로 블록이 움직임 보상을 통해 이루어 짐을 알 수 있고 첫 번째 scene 또한 사람들이 앉아있는 상황으로 움직임이 적으므로 상대적으로 브라우징 속도를 매우 빠르게 결정할 수 있다. 반면 움직임이 많은 세 번째 scene 의 경우 intra coded 매크로블럭의 비율이 더 많으므로 브라우징 속도는 더 느리게 결정된다.

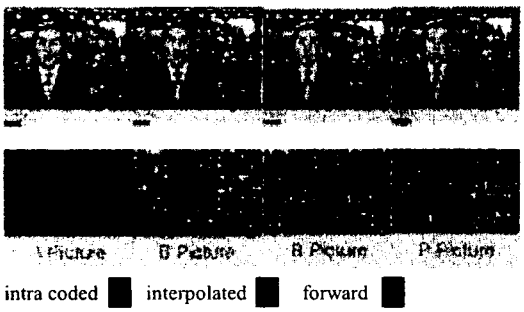

그림 14. scene 안에서의 매크로 블록 분포도

V. 결 론

현재 개발 중인 PVR 의 주요 목적은 사용자가 원하는 영상물의 저장이고 non linear browsing 은 대형화된 저장 공간에서 보다 효율적인 검색을 가능케 한다. 이를 위해 주로 사용한 방법은 화면의 밝기 정보를 비교하여 shot 의 변화를 찾는 것이다. 실시간이라는 제약이 있기 때문에 MPEG2 stream 의 H/W 적 parsing 이 요구 되고, 방송용 스트림을 HDD 에 전송, 저장 하며 브라우징 정보를 전송하므로 bus bandwidth 의 한계가 있기 때문에 간략화 된 데이터 전송이 필요하다. 때문에 화면 정보들은 가장 효율적인 양자화가 되었다.

본 논문에서 주장한 브라우징 방식은 기존의 아날로그적 저장장치에 적용되던 정속 위주의 linear browsing 방법을 보안하여 clustering 을 통한 non linear 브라우징이 데이터의 검색에 더욱 용이함을 알 수 있다.

VI. 참고 문헌

[1] ISO 13818-2:1994 *Information technology – Coding of Moving Pictures and Associated Audio – Part 2 : Video*

[2]ISO 13818-2:1994 *Information technology – Coding of Moving Pictures and Associated Audio – Part 1 : System*

[3] John R. Smith and Shih-Fu Chang, *"Visually Searching the Web for Content"*, 1997 IEEE.

[4] J.Berens, G.D.Finlayson and G.Qiu, *"Image indexing using compressed colour histgrams"*, Image Signal Process., Vol 147, No.4, August 2000

[5] Ullas Gargi, Rangachar Kasturi, and Susan H. Strayer, *"Performance Characterization of Video-Shot-Change Detection Methods"*, IEEE 2000

[6] Alan Hanjalic, Reginald L. Lagendijk and Jan Biemond, *"Automated High-Level Movie Segmentation for Advanced Video-Retrieval Systems"*, IEEE 1999

[7] Minerva M. Yeung and Boon-Lock Yeo, *"Video Visualization for Compact Presentation and Fast Browsing of Pictorial Content"*, 199 IEEE

[8] Barry G. Haskell, Atul Puri and Arun N. Netravali, *"Digital Video: An introduction to MPEG-2"*, Chapman and Hall

[9] Joan L. Mitchell, William B. Pennebaker, Chad E. Fogg and Didier J. LeGall," MPEG video Compression Standard" , Chapman and Hall