

히스토그램의 상관성을 이용한 Video Thumbnail 추출 기법

*서찬원 **한종기 ***배진수

세종대학교

*sachonwon@gmail.com, **hjk@sejong.ac.kr, ***baej@sejong.ac.kr

Extraction Method for Video Thumbnail using Correlations of Histograms

*Seo, Chan-Won **Han, Jong-Ki ***Bae, Jinsoo

Sejong University

요약

최근 통신 환경의 진화와 고사양 모바일 단말기의 대중화로 인하여 모바일 통신 환경에서 동영상 서비스를 이용하는 사용자가 급증하고 있다. 사용자들은 제한적인 데이터 사용량을 이용하여 필요한 영상을 효율적으로 검색 및 탐색하기 위하여 미리보기 서비스 또는 Thumbnail을 이용한다. 이 중 Thumbnail은 동영상의 복호화 과정에서 제공되는 서비스이며, 이를 효과적으로 추출하는데 있어서 낮은 복잡도의 기술들이 사용되어야 한다. 본 논문에서는 영상의 히스토그램의 특징을 이용한 Thumbnail 추출 기법을 제안한다. 제안하는 추출 기법에서는 각 영상의 히스토그램을 비교 및 분석하여 하나의 video를 다수의 구간으로 분할하고, 각 구간의 대표 Thumbnail을 추출하는 방법을 사용한다. 다양한 영상을 이용한 실험 결과는 제안하는 알고리즘이 효과적으로 Thumbnail을 추출함을 보여준다.

1. 서론

Thumbnail은 해상도가 높은 이미지를 낮은 해상도의 이미지로 표현함으로써 많은 수의 이미지를 쉽고 빠르게 볼 수 있도록 해준다. 최근에는 그림 1와 같이 동영상에서도 Thumbnail 이미지를 널리 사용하고 있다. 특히, 인터넷을 통한 비디오 스트리밍 서비스에서는 영상의 주요 장면을 선택적으로 미리 보여줌으로써 사용자가 원하는 장면부터 시청할 수 있도록 도움을 준다.



그림 1. 비디오 서비스에서의 Thumbnail 사용의 예

비디오에서 Thumbnail을 추출하는 기법은 많은 논문을 통해 연구되어왔다 [1]-[4]. [1]과 [2]에서는 지금까지 발표된 기존 추출 기법들을 정리하고 있으며, 다양한 방법을 소개하고 있다. [3]에서는 Video

이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2011-0011401)

Time Density Function (VTDF)를 이용하여 비디오의 시간적 특성을 분석하고 그 결과를 토대로 영상을 분할하는 방법을 사용한다. 또한 [4]에서는 H.264/AVC [5]로 부호화 된 비트스트림을 복호화할 때 적합한 Thumbnail 추출 기법을 제안한다.

기존 방법들에서 효과적으로 비디오의 Thumbnail을 추출하는 기법을 제안하였다. 그러나 기존 방법들은 다양한 비용함수 및 움직임 정보등을 이용하여 상대적으로 낮은 알고리즘 복잡도를 요구하는 모바일 환경에서는 적합하지 않다. 따라서 본 논문에서는 히스토그램의 특징을 이용한 간단한 Thumbnail 추출 기법을 제안한다.

본 논문의 제 2장에서는 제안하는 Thumbnail 추출 기법을 제안하고, 제 3장에서는 제안하는 알고리즘을 적용한 실험 결과를 보인다. 그리고 마지막 제 4장에서는 결론을 서술한다.

2. 제안하는 Thumbnail 추출 기법

일반적으로 비디오의 Thumbnail을 추출하기 위해서는 영상을 shot 별로 분류하는 작업이 필요하다. 이는 각 shot에서 해당 구간을 대표하는 Thumbnail이 불필요하게 많이 선택되는 것을 방지하기 위함이다. 본 논문에서는 각 영상의 히스토그램을 비교 및 분석하여 shot의 경계를 인식하고, 해당 구간에서 대표 Thumbnail을 선택하는 기법을 제안한다.

가. 비디오 영상의 구간 분할

히스토그램의 분석을 통해 영상의 특성을 분류할 수 있다는 것은 이미 오래전부터 널리 알려져 있으며 [6], 그림 2는 영상과 시간에 따

른 히스토그램을 보여준다. 그림 2에서 S1와 S2는 서로 다른 영상이며, t 는 현재 영상을, $t-1$ 은 바로 직전의 과거 영상을 나타낸다. 그림에서 확인 할 수 있듯이, 같은 shot에서 시간적으로 인접한 영상의 히스토그램은 서로 비슷하여 상관성이 높은 반면, 다른 shot에 존재하는 영상인 경우에는 히스토그램의 분포가 확연히 다른 것을 알 수 있다. 따라서 제안하는 구간 분할 기법에서는 시간적으로 근접한 영상들의 히스토그램을 비교하여 각 구간을 분할한다.

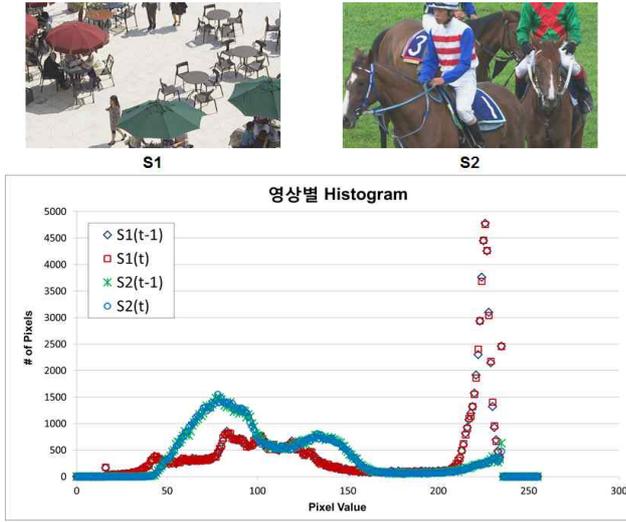


그림 2. 영상별 히스토그램

히스토그램의 특성을 이용하여 구간을 분할하기 위하여 본 논문에서는 각 영상의 히스토그램의 시간적 상관성을 측정한다. 이 때, fade-in 또는 fade-out과 같이 shot은 변하지 않고 화소 값만 변하는 경우에 대하여 효과적으로 대처하기 위해, 각 영상의 평균 화소값이 0이 되도록 다음과 같이 히스토그램을 이동시킨다.

$$H_s(i) = H_o(i - P_{avg}) \quad (1)$$

위의 수식에서, H_s 는 원본 히스토그램인 H_o 에서 평균 화소값 P_{avg} 만큼 이동시킨 히스토그램을 나타낸다. 이렇게 얻어진 H_s 을 이용하여 과거 영상의 히스토그램과 현재 영상의 히스토그램의 상관관계 계수 (correlation coefficient)를 아래와 같이 계산한다.

$$\rho_{t,t-1} = \frac{\sum_{i=-P_{max}}^{P_{max}-1} H_s^t(i) \times H_s^{t-1}(i)}{2P_{max} \sigma_t \sigma_{t-1}} \quad (2)$$

수식 (2)에서 P_{max} 는 입력 영상의 최대 화소값을 의미하며, 8 bit 영상에서는 256이다. 그리고 σ 는 각 영상의 표준 편차를 나타낸다. t 는 현재 영상의 시간적 위치를 나타내어, $t-1$ 은 현재 영상의 바로 직전에 존재하는 영상을 의미하게 된다. 이 상관 계수는 YUV color format 4:2:0에서 Y 성분뿐만 아니라, U, V에 대하여 계산할 수 있으며, 이 값들을 수식 (3)과 같이 가중치를 고려한 합을 계산한다.

$$c_t = 2 \times \rho_{t,t-1}^Y + \rho_{t,t-1}^U + \rho_{t,t-1}^V \quad (3)$$

위의 수식에서 밝기 신호의 상관관계 계수에 더 큰 가중치를 두는 이유는, 사람이 시각적으로 색차 신호보다는 밝기 신호에 더 민감하기 때문이다. 각 상관관계 계수가 $-1 \sim 1$ 범위의 값을 가지므로, c_t 는 -4 에서부터 4 사이의 값을 갖는다. 일반적인 경우에는 음수의 상관관계 계수는 좀처럼 발생하지 않으며, $0 \sim 4$ 사이의 값을 갖게 된다.

제안 하는 구간 분할 방식에서는 수식 (3)에서 계산한 수식이 특정 임계값 보다 작은 경우 새로운 shot이 발생했다고 판단하고 그 영상부터 새로운 구간으로 구분한다. 각 구간의 시작점의 집합을 아래와 같이 나타낼 수 있다.

$$D = \{t | c_t \leq \alpha_c \text{ or } t = 0\} \quad (4)$$

위의 수식에서 D 는 각 구간의 시작점의 집합을 의미하며, α_c 는 구간을 나누는 임계값을 나타낸다. 이 때, Video의 가장 첫 번째 영상은 과거 영상이 없으므로, c_t 값을 계산할 수 없다. 이에 따라 첫 번째 영상을 집합 D 에 포함시키기 위하여 $t = 0$ 의 조건이 추가되었다.

나. 구간별 대표 Thumbnail 추출

시간적으로 연속되는 영상에서, 영상 사이의 변화가 크지 않은 영상보다는 변화가 많은 영상이 보는 사람에게 더 많은 정보를 주게 될 확률이 높다. 이는 수식 (3)에서 계산한 c_t 값이 나타내는 의미와 유사하다. c_t 값이 최대인 4에 가깝다면, 현재 영상의 히스토그램은 이전 영상의 히스토그램과 Y 성분을 비롯하여 U와 V까지 모두 상관도가 높음을 의미하게 되고, 이는 보는 이에게 제공되는 정보의 양이 많지 않음을 나타낸다. 반면, c_t 의 값이 작은 경우에는 해당 영상이 이전 영상과 비교하여 많은 변화가 일어나서 현재 구간에서 주목할 만한 사건이 일어날 가능성이 높음을 의미하게 된다. 이는 이미 큰 변화가 일어나는 것을 반영하여 구간을 분할한 후 측정하게 됨으로써 shot의 변화보다는 현재 shot에서 발생하는 움직임이나 변화를 나타내게 된다. 따라서 제안하는 구간별 대표 Thumbnail 추출은 다음 수식에 따라 결정한다.

$$t_j^* = \operatorname{argmin}_{D(j) < t < D(j+1)} [c_t] \quad (5)$$

위의 수식에서 $D(j)$ 는 수식 (4)을 통해 얻어진 j 번째 구간의 시작점을 나타내고, t_j^* 는 해당 구간에서 가장 변화가 많이 일어난 영상으로써, 구간 j 에서 Thumbnail로 선택되는 영상의 시간적 위치를 표현한다.

3. 실험

본 실험에서는 네 가지 종류의 실험영상을 이용해 제안하는 알고

리즘을 검증하였다. 첫 번째 실험영상(Video 1)은 4개의 서로 다른 영상을 조합하여 다양한 shot이 일정하지 않은 간격으로 발생하는 영상으로 많은 scene change를 포함하고 있다. 그리고 2 개의 실험 영상(Video 2, Video 3)은 음악 방송 촬영 영상으로 다양한 화면과 움직임, 그리고 밝기 변화를 다수 포함하고 있다. 마지막 실험 영상(Video 4)은 자동차 게임 영상으로 가운데 자동차는 크게 움직이지 않고 주변 배경이 변화하는 특징을 가진다. 본 실험에서는 수식 (4)의 임계치인 α_c 를 2.5로 설정하였다.



그림 3. Video 1의 Thumbnail



그림 4. Video 2의 Thumbnail



그림 5. Video 3의 Thumbnail



그림 6. Video 3의 Thumbnail

그림 3은 scene change가 많이 일어나는 영상에서 추출함에 따라 추출된 Thumbnail이 서로 다른 영상을 보여준다. 그림 4와 그림 5는 음악 방송에서 볼 수 있는 다양한 장면들을 각 shot에 따라 Thumbnail이 적절하게 선택되었음을 확인할 수 있다. 마지막 그림 6에서는 가장 중요한 물체의 변화가 적고 주위 환경만 변함에 적은 수의 Thumbnail이 선택되는 것을 볼 수 있다.

4. 결론

본 논문에서는 모바일 단말기를 위한 간단한 비디오 Thumbnail 추출을 위하여 영상 사이의 히스토그램을 비교 및 분석하여 상관관계에 따라 영상의 구간을 분할하고, 해당 구간에서 히스토그램의 변화가 가장 많은 영상을 Thumbnail로 추출하였다. 실험을 통하여 제안하는 알고리즘이 각 비디오의 Thumbnail을 적절하게 추출하는 것을 확인할 수 있음을 보였다.

5. 참고 문헌

- [1] C. Cotsaces, N. Nikolaidis and I. Pitas. "Video shot boundary detection and condensed representation: a review," IEEE Signal Processing Magazine, vol.23, no.2, Mar. 2006.
- [2] B.-T. Truong and S. Venkatesh. "Video abstraction: a systematic review and classification," ACM Trans. on Multimedia Computing, Comm. and Applications, vol.3, no.1, Feb. 2007.
- [3] J. Jiang and X. P. Zhang, "Video thumbnail extraction using video time density function and independent component analysis mixture model," IEEE International Conference on Acoustic, Speech and Signal Processing (ICASSP), 2011 pp. 1417-1420.
- [4] E. S. Kim, T. W. Um, and S. J. Oh, "A fast thumbnail extraction method in H.264/AVC video streams," IEEE Transactions on Consumer Electronics, vol. 55, issue 3, pp. 1424-1430, Aug. 2009.
- [5] ITU-T Rec. H.264 and ISO/IEC 14496-10, "Advanced Video Coding for Generic Audiovisual Services", Nov. 2007.
- [6] H. Zhang and S. S. A. Kankanalli, "Automatic partitioning of full-motion video," ACM Multimedia Systems, vol 1, no 1, pp. 10 - 28, Jan. 1993.