Video Key-Frame Extraction using Unsupervised Clustering and Mutual Comparison

키 프레임 추출은semantic 개념 기반 비디오 인덱싱 및 검색에서 중요한 단계 중 하나이며 비디오 개념 탐지의 정확성은 키 프레임 추출 방법의 효과에 크게 좌우됩니다. Shot의 두드러지는 content와 clustering의 결과에 의존하여 key-frame이 추출됩니다. 하지만 보통, visual complexity와 비디오의 내용 때문에 출력에서 동일한 의미론적 내용을 가진 중복되거나 반복적인 key frame에 접근하는 경향이 있어 key frame 추출의 정확도가 떨어집니다. 따라서 이 논문에서는 unsupervised clustering과 mutual comparison(color component(HSV histogram)에 70%의 가중치, texture (GLCM)에 30%가중치)에 기반을 두며 클러스터링에 사용되는 결합된 프레임 유사성 인덱스를 계산하는 novel key-frame extraction 방법을 제시합니다. 또한 중복되는 key-frames을 제거하기 위한 mutual comparison을 제시합니다.

Introduction

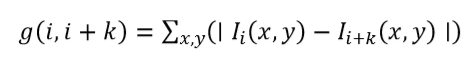
비디오 데이터의 실질적인 서로 다른 특성(ex 비디오 길이, 구조화되지않은 format)때문에 비디오에 효율적으로 접근하는 것은 매우 어려운 일입니다. 이러한 문제를 해결하기 위해 abstraction과 summarization 기술이 필요합니다. Video segmentation은 shot이라고도 불리며 shot boundary detection method의 목적은 비디오 sequence를 multiple shots로 나누는 것입니다. 우선 video 가 나누어 지고 나면 key frame 추출이 가능해집니다. Key-frame extraction methods는 video 처리 문제를 image 처리 문제로 전환합니다. Key-frame은 shot의 대표적인 shot이 되어야 하며 이는 shot contents를 가장 잘 반영하는 frame이 되어야 합니다. 대부분은, shot의 middle frame이 key-frame으로 식별됩니다. 하지만 key-frame을 식별하는 다른 많은 방법들이 있습니다. Shot이 항상 하나의 frame을 대표할 필요는 없지만 몇몇 경우에 visual complexity 때문에 multiple key-frame들이 하나의 shot으로 대표될 수 있습니다. Key-frame들은 shot의 index를 찾는데 사용될 수 있습니다. 따라서 key-frame의 사용은 video indexing에 필요한 정보들을 상당히 줄일 수 있습니다.

몇몇 효과적인 shot boundary detection algorithm들이 존재하지만 대부분의 접근들은 다양한 visual content에 효과적으로 적용되지 못합니다. 이 논문에서 효과적이고 효율적인 clustering에 기반한 접근법을 제시합니다.

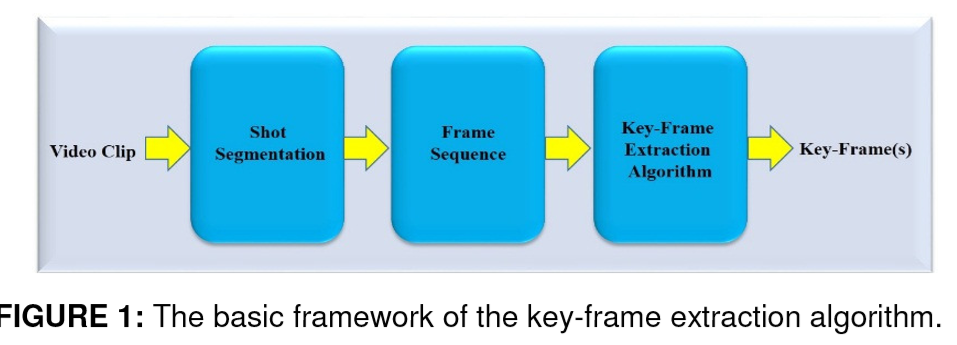
Shot Boundary Detection

Shot boundary detection 또는 video segmentation은 key-frame extraction의 첫 단계이며 video retrieval과 video browsing에 많이 쓰입니다. Video shot은 video stream의 기본 unit이며 single camera로 만들어진 frame의 unbroken string입니다. Editing stage에서는 hard cuts 또는 dissolve, fade-in, fade-out, wipe 등 점진적 전환을 사용하여 shot을 결합합니다. Shot boundary detection methods는 cut boundary detection(컷 경계 검출) 과 gradual transition detection(점진적 전이 검출)으로 분류됩니다.

Shot boundary detection method 의 중요한 task는 video stream으로부터 visual features(color, shape, texture, motion) 또는 그 특징들의 결합을 뽑아내고 frame들 사이의 similarity를 측정하는 것이다. g(I,i+k)가 I와 i+k frame의 difference 또는 discontinuity를 나타낸다. 이때 g(I,i+k)를 계산하기 위한 대안들이 video sequence에 존재하며, 가장 간단한 것은 frame 사이의 절대 차이이며 다음의 방정식에 의해 주어진다.



이때 I(x,y)는 image pixel의 (x,y) 좌표의 intensity level을 나타낸다. 절대 차이를 기반으로 하는 방법은 frame sequence의 중요한 차이를 테스트하기 위해 설정된 임계 값과 차이를 비교합니다. 하지만 g(I,i+k)는 intensity의 변화나 object, camera motion에 매우 민감하여 false detection 확률이 증가할 수 있습니다.



Shot detection에 사용되는 가장 주된 방법은 pixel-difference, statistical-difference, histogram comparison, edge-difference, motion vector methods입니다. Frame difference를 계산하기 위해 가장 선호되는 방법은 histogram-based method이며 histogram은 frame의 color 분배를 나타냅니다. Shot detection의 효율은 연속적인 frame 사이의 similarity measure의 적합한 선택에 의존합니다.??

Color histogram 기술은 frame sequence내의 object에 대해 독립적입니다. 기존의 histogram(RGB, HSV 등) 기반 기술은 강력하고 효과적인 것으로 나타났습니다. 두 이미지의 color histogram이 계산되고 similarity가 histogram intersection 기법에 의해 계산됩니다. 만약 두 histogram 사이의 similarity가 특정한 threshold에 있다면 a shot boundary가 가정됩니다. Color histogram의 문제는 비슷한 histogram을 가진 이미지들이 서로 다른 visual appearance를 가질 수 있다는 것입니다.

Key-Frame Extraction

Video stream이 shot segmentation 과정을 거치고 나면 shot의 frame들은 서로 매우 비슷할 것입니다. 따라서 shot의 content를 가장 잘 반영하는 key-frame을 선택해야 합니다. 최근의 key-frame extraction 접근법들은 6개의 categories로 나누어집니다. (sequential comparison-based, global comparison-based, reference frame-based, clustering-based, curve-simplification-based, object/event based)

1. Sequential comparison-based

* 이전에 선택된 key-frame은 similarity가 전혀 다른 key-frame이 선택될 때까지 다음의 frame과 비교하며 진행합니다. 이 algorithm의 이점은 simplicity, low computational complexity, 그리고 하나의 shot에 여러 key-frames을 적용할 수 있다는 것입니다. 이 algorithm의 한계는 key-frame이 shot의 global한 properties가 아닌 local properties를 나타낸다는 것과 key-frame이 불규칙적으로 분포되고 key-frame 수가 가변적이므로 일부 응용 프로그램에는 적합하지 않다는 것입니다. 또한 내용이 반복적으로 발생하면 key-frame간에 중복 가능성이 있습니다.

1. Global comparison-based

* 이 접근 방식은 어플리케이션에 따라 사전 정의된 objective function을 최소화하여 key-frame을 분배한다. 일반적으로 objective function은 다음의 4가지를 따른다 1) Even temporal variance 2) Maximum coverage 3) Minimum correlation 4) Minimum reconstruction error 이 algorithm의 이점은 1) key-frame이 global characteristics를 반영한다 2) key-frames의 수가 제한된다 3) key-frame간의 중복이 최소화된다는 것이며, 단점은 비교적 계산비용이 많이 든다는 것이다.

1. Reference frame-based

* Reference frame이 생성되며 shot frames을 reference frame과 비교하며 key-frame extraction을 진행한다. 이 algorithm은 이해하고 시행하기 쉽지만 key-frame의 정확도가 reference frame의 정확도에 의존하게 된다.

1. Clustering

* Shot frame들이 clustered 되며 이후 cluster의 center에 가장 가까운 frame을 key-frame으로 선택한다. 이 방법의 가장 큰 이점은 추출된 key-frame이 global characteristics를 반영한다는 것이며 단점은 extraction의 정확도가 clustering의 결과의 정확도에 의존한다는 것이다.

1. Curve simplification-based

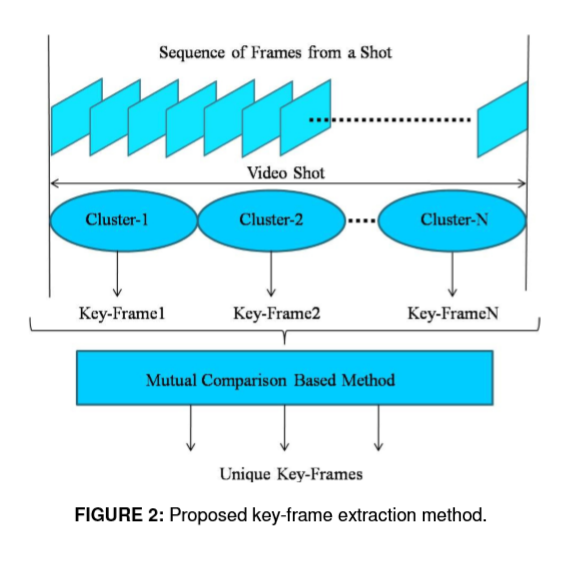
* Shot의 각각의 frame은 feature space안의 point로 표시되며 trajectory curve(궤적 곡선)을 얻기 위해 순차적으로 연결된다. 이후 curve의 모양을 가장 잘 나타내는 point의 그룹을 찾아간다. 이 algorithm의 장점은 sequential 정보가 유지된다는 것이며 단점은 최적의 curve를 찾는데 계산 복잡도가 매우 높아진다는 것이다.

1. Object/Events

* 많은 video processing applications에서 우리는 몇몇 object또는 events에 관심이 있을 수 있다. 이 algorithm은 우선 object 또는 event를 찾아낸 뒤 key-frame extraction을 진행하여 key-frame이 필요로 하는 object 또는 events 정보를 가지고 있다. 따라서 이 방법의 장점은 key-frame이 의미적으로 풍부한 정보를 가진다는 것이며 단점은 object/events detection이 응용 프로그램의 특성에 따라 지정된 heuristic rules에 크게 의존한다는 것이다.

Key-frame 정의의 주관성으로 인해 key-frame extraction에 이용 가능한 균일한 평가 방법이 없다는 것을 알고 있어야 한다.

Clustering based approach and proposed method



 Clustering은 매우 효과적인 기술이며 패턴인식과 information retrieval 등 많은 부분에 사용되고 있다. Clustering은 크게 두가지 types으로 나누어질 수 있으며 supervised 와 unsupervised가 있다. Supervised clustering은 cluster의 수를 사전에 알 수 있을 때 좋으며 unsupervised는 cluster의 수를 확신할 수 없을 때 유용하다. Supervised clustering에서 shot boundary detection algorithm으로 얻어진 video shot을 라 하자. N개의 frame들을 M개의

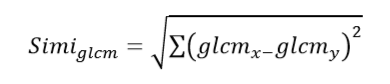
라 불리는 cluster로 분류한다. 임의의 객체 또는 frame의 두드러진 content는 해당 객체 또는 frame의 색상, texture 또는 shape가 될 수 있는 객체 또는 frame으로 정의됩니다. 두 frame 사이의 similarity는 visual content의 similarity를 계산함으로써 구할 수 있다. 이 논문에서 visual content를 나타내기 위해 frame의 color와 texture 요소의 가중치 조합을 선택한다. Color feature은 HSV color space의 global level histogram을 사용하였고 texture feature은 GLCM을 사용했다. GLCM에서 사용된 실질적인 feature들은 1. Contrast 2. Correlation 3. Energy 4. Homogeneity이며 이러한 feature들을 계산한 뒤 HSV histogram 과 GLCM texture feature각각에 대해 frame i와 j 사이의 similarity index를 찾는다. Histogram similarity는 color bin들의 최소값의 합을 사용하는 histogram intersection method를 통해 계산될 수 있다. Frame i와 j 사이의 histogram similarity index는 다음과 같이 정의된다.



hx와 hy는 frame x와 y 각각의 HSV histogram이며 C는 histogram 안의 color bin의 수이다.

Simi\_hsv는 두 frame에서 공통적인 총 pixel 수를 제공한다.

GLCM texture features를 이용한 frame similarity index는 다음의 식을 사용한 Euclidean distance method를 사용하여 계산된다.

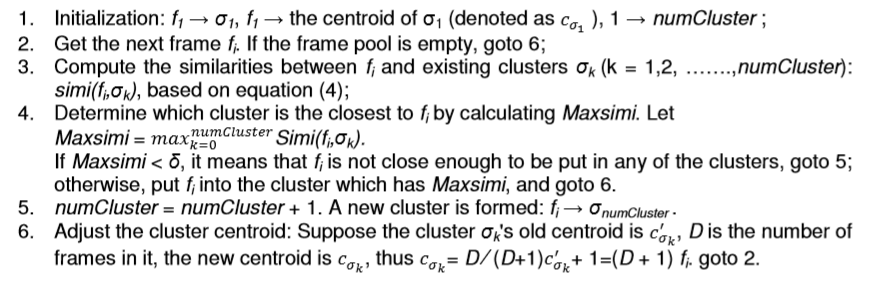


glcmx 와 glcmy는 각각 frame x,y의 GCLM texture feature이다. Feature를 선택할 때 모두가 video frame의 visual content를 표현하는데 똑같이 효과적이진 않다. 따라서 feature들의 중요성에 기반을 두고 적절하게 weights를 할당해야 한다. 이 논문에서는 color component가 texture보다 major salient content를 나타낸다고 생각해 color histogram에 70%의 weightage, GLCM texture에 30%의 weightage를 준다. 다음은 이러한 feature 들을 결합하고 결합된 frame similarity index를 찾는 것이다. Frame x,y 사이의 결합된 식은 다음과 같다.



위의 frame similarity index를 사용하여 clustering이 진행된다. 모든 clustering algorithm에는 clustering density를 제어하는 임계 값 parameter δ가 있다. δ가 높을수록 더 많은 cluster가 생긴다. Human learning 과 recognition system에서도 이러한 것을 볼 수 있는데 예를 들어 임계 값이 낮으면 사람들은 자동차를 분류할 때 wagons, mini-vans 들을 vehicles 로 분류할 것이다. 하지만 임계 값이 높으면 wagons, mini-vans등을 서로 다른 categories로 분류할 것이다. 따라서 임계 값 parameter는 classification의 밀도를 제어하게 해준다. 새로운 frame이 특정 cluster로 분류되기 전 node와 cluster의 중심 사이의 유사성이 계산된다. 만약 value가 δ보다 낮으면 이는 노드가 cluster에 추가되기에 충분하지 않다는 것을 나타낸다.

Unsupervised clustering algorithm은 다음과 같이 요약된다.



Cluster가 구성되고 나면 다음단계는 key-frame을 선택하는 것이다. 이 논문에서는 충분히 크고 key-cluster라고 고려되는 cluster만 선택하였으며 이 cluster로부터 대표적인 frame을 key-frame으로 뽑았다. 충분히 큰 기준은 cluster내부의 frame 수가 min\_clust\_size = shot의 전체 frame수의 10% 보다 크면 충분히 큰 것으로 잡았다. 해당하는 key-cluster의 key-frame은 cluster의 중심에 가장 가까운 것이며 이는 key cluster의 주요 visual content를 나타낼 수 있을 것이다. 만약 min\_clust\_size를 줄이면 cluster의 수가 증가하고 over-segmentation이 생길 수 있다. 반대로 min\_clust\_size를 높이면 under-segmentation이 생길 수 있다.

Key-frame을 추출하고 나서 over-segmentation의 결과로 추정되는 몇몇 key-frame들이 중복되는 것을 발견했다. 이를 막기위해 min\_clust\_size를 줄였지만 under-segmentation의 결과가 나와 따라서 이러한 문제를 해결하기 위해 mutual comparison을 사용하였다. 각각의 key-frame을 다른 key-frame과 비교하여 similarity를 찾고 만약 similarity가 특정한 임계 값보다 높으면 중복된 key-frame으로 판단하여 제거하였다.