

MPEG 도메인상에서 카메라 움직임 정보 추출 방법

강명규, 박성한
한양대학교 컴퓨터공학과

An Extraction Method of Camera Operations In MPEG Domain

Myeong-gyu Kang, Sung Han Park
Department of Computer Science and Engineering
Hanyang University
E-mail: {mgkang, shpark}@cse.hanyang.ac.kr

Abstract

본 논문에서는 MPEG 압축 도메인상에서 카메라 움직임 정보를 추출하는 효과적인 방법을 제안한다. 카메라 움직임 정보는 동영상에서의 주요 장면과 프레임 간의 관계를 기술할 수 있는 실마리를 제공한다. 본 논문에서는 MPEG Video 의 모션벡터를 이용하여 카메라 움직임 정보를 추출한다. 카메라 움직임에 따라 모션벡터는 특징적으로 분포하는 특성이 있다. 본 논문에서는 이러한 특징들을 이용하여 MPEG 모션벡터의 방향성과 크기를 이용하여 각 모션벡터끼리 교차점과 평행성분을 구한다. 그리고 이것을 이용하여 같은 교차점과 평행성분끼리 모션벡터 Clustering 을 수행한다. 본 논문에서는 클러스터링 된 모션벡터를 Fuzzy inference rule 을 이용하여 카메라 움직임이 Zoom, Pan, Tilt 인지 여부를 판단한다. 실험은 전통적인 방법 중에 하나인 Affine Model 방법과 비교하며 본 논문의 방법이 어느 정도 우수함을 입증한다.

I. 서론

최근 디지털 TV 방송과 디지털 매체 기술의 발전으로 멀티미디어에 대한 관심이 높아지고 있다. 특히 앞으로 디지털 비디오 자료가 대중화 됨에 따라 방대한 양의 자료를 효과적으로 이용 및 관리하기 위한 비디오 인덱싱 기술 연구가 대두되고 있다. 색상, 텍스처 정보,

자막 및 얼굴 정보 등 비디오 내의 Content 정보가 비디오 인덱싱에 유용하게 쓰일 수 있다. 특히 최근 들어 프레임간의 정보를 기술해 줄 수 있는 카메라 움직임 정보 추출 연구가 활발히 진행되어가고 있다[1]-[4]. 전통적으로 카메라 움직임 정보 추출은 비압축 도메인에서 Optical flow 를 계산하여 Affine Model 을 이용하여 구했다. 그러나 비압축 도메인상에서의 계산은 MPEG 동영상의 경우 비압축 도메인으로 Decoding 하는 추가적인 오버헤드가 따르게 된다. 그래서 최근에는 MPEG 압축 도메인상에서 카메라 움직임 정보를 추출하는 연구가 활발히 진행되고 있다. A. Akutsu 는 MPEG 도메인상에서 모션 정보를 Hough Transform 을 이용하여 각 방향 별로 누적을 하여 카메라 움직임 정보를 추출하였다[2]. Hough Transform 방법은 움직임에 대한 판별근거가 모호하다는 문제가 있다. Y. Tan 는 MPEG 도메인에서 모션 벡터를 Affine 모델에 직접 적용하여 카메라 움직임 정보를 추출 하였다[3]. Affine 모델 방법은 잘못된 모션 벡터가 하나만 있더라도 움직임 정보를 제대로 판별 못하는 단점이 있다. 본 논문은 이러한 문제점들을 해결하기 위해 MPEG 비디오 데이터에 존재하는 모션벡터가 카메라 움직임에 따라 특징적으로 분포하는 특성과 Fuzzy Inference 을 이용하여 카메라 움직임 정보를 추출하는 방법을 제안한다.

본 논문에서는 먼저 2 절에서 본 논문에서 사용된 배경 아이디어를 소개한다. 3 절에서는 본 논문에서 제안

하는 MPEG 도메인상에서 카메라 움직임 정보 추출 방법을 소개한다. 4 절에서는 실험 결과를 보이고, 마지막으로 5 절에서는 결론을 맺는다.

2. 연구 배경

2.1 MPEG 에서의 모션벡터

MPEG Video 데이터는 I, B, P 프레임으로 이루어져 있다. 특히 B 와 P 프레임에는 프레임간의 움직임 보상을 위한 모션벡터란 정보가 존재한다. 실제로 MPEG Video 데이터의 B, P frame 에 존재하는 모션 벡터는 아래 그림과 같다.

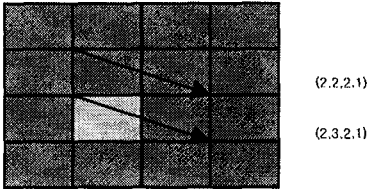


그림 2.1 모션 벡터

연속해서 여기서 관심을 가질 것은 실제 MPEG 데이터를 조사해 보면 P frame 의 매크로 블록에는 모션벡터가 (x', y', dx', dy') 형식으로 기록되어 있다. dy'/dx' 을 계산하면 현재 모션벡터의 기울기를 얻어낼 수 있다. 그리고 모션 벡터는

$$y = \frac{dy'}{dx'}x + (y' - \frac{dy'}{dx'}x') \text{-----}(식 2.1)$$

형태의 직선의 방정식으로 표현될 수 있다.

2.2 카메라 움직임에 따른 모션 벡터 분포

카메라 움직임에 따라 MPEG Video 의 모션벡터는 특징적으로 분포하는 특성이 있다. 카메라가 Zoom 인 경우 모션벡터는 한 점으로 향하는 특성이 있다. 그리고 카메라가 Pan, Tilt 인 경우는 한쪽 방향으로 평행하게 분포한다. 본 논문에서는 모션벡터의 이러한 특성을 이용하여 MPEG 데이터의 모션벡터로부터 카메라 움직임 정보를 추출한다.

3. 카메라 움직임 정보 추출 방법

본 절에서는 MPEG 도메인 상에서 카메라 움직임

정보를 추출하는 방법에 대해서 설명한다. MPEG Video 데이터는 I, B, P 프레임으로 이루어져 있는데 본 연구에서는 P 프레임만을 대상으로 한다. 아래 그림은 본 논문에서 제안하는 방법의 전체 구성도 이다.

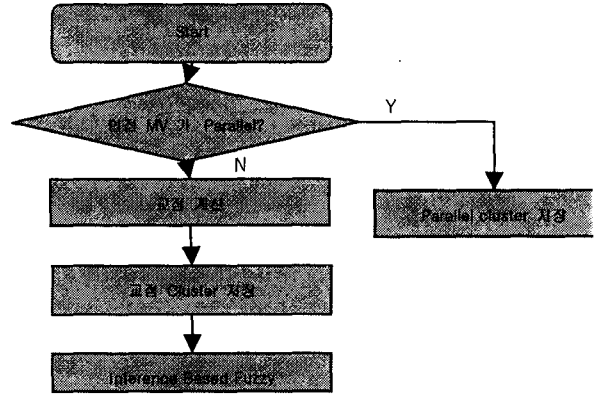


그림 3.1 전체 구성도

3.1 인접 MV 간에 Parallel 여부 조사

평행 여부 조사는 P-frame 에 존재하는 각각의 모션 벡터에 대해 왼쪽 상단에서부터 오른쪽 아래 방향으로 진행해 나간다. 현재 위치가 (x_p, y_p) 이고 이 위치에 해당되는 모션벡터가 (x_p, y_p, dx_p, dy_p) 일 때 기울기는

$$d_p = \frac{dy_p}{dx_p} \text{-----}(식 3.1)$$

이다. (x_p, y_p, dx_p, dy_p) 와 인접한 모션벡터들은

$$(x_p + 1, y_p, dx_1, dy_1), (x_p, y_p + 1, dx_2, dy_2), (x_p + 1, y_p + 1, dx_3, dy_3)$$

이다. d_p 와 인접 모션벡터의 기울기 값을 서로 비교한다. 만약 d_p 와 인접 모션벡터의 기울기 값이 서로 같으면 평행이고 다르면 평행이 아니다. 서로 평행인 모션 벡터인 경우 기울기 값이 같은 Parallel Cluster 에 저장하고 평행이 아닌 경우 교점을 구하여 서로 교점이 같은 Intersection Cluster 에 저장한다.

3.2 교점 계산

평행이 아닌 두 개의 모션 벡터간에는 교점을 구하게 되는데 그 방법은 아래와 같다. 두 개의 모션벡터와

그에 해당하는 직선의 방정식이 아래와 같을 때

$$(a_1, b_1, da_1, db_1), y = \frac{db_1}{da_1}x + (b_1 - \frac{db_1}{da_1}a_1) \text{----(식 3.2)}$$

$$(a_2, b_2, da_2, db_2), y = \frac{db_2}{da_2}x + (b_2 - \frac{db_2}{da_2}a_2) \text{----(식 3.3)}$$

두 직선의 방정식의 교점을 (a', b') 이라 하면

$$\frac{db_1}{da_1}a' + b_1 - \frac{db_1}{da_1}a_1 = \frac{db_2}{da_2}a' + b_2 - \frac{db_2}{da_2}a_2 \text{에 의해}$$

$$\therefore a' = \frac{1}{\frac{db_1}{da_1} - \frac{db_2}{da_2}} \left(\frac{db_1}{da_1}a_1 - \frac{db_2}{da_2}a_2 + b_2 - b_1 \right) \text{----(식 3.4)}$$

(단, $\frac{db_1}{da_1} - \frac{db_2}{da_2} \neq 0$)

a' 을 두 개의 모션 벡터의 직선의 방정식 중에 하나에 대입하면 마찬가지로 b'도 구할 수 있다.

3.3 Fuzzifier

그림 3.1, 3.2 는 실제로 모션벡터끼리 평행여부를 판별하여 저장된 Cluster 결과를 나타낸다. 이 결과를 바탕으로 현재의 프레임이 어떠한 카메라 움직임 정보가 있는지를 판별해야 한다.

Parallel Cluster	Intersect Cluster
0 : 547	0 : 34
1 : 2	1 : 2
	2 : 1
	3 : 1
	...
	147 : 1

그림 3.1 Pan, Tilt 인 경우 Cluster 들

Parallel Cluster	Intersect Cluster
0 : 79	0 : 99
1 : 6	1 : 2
2 : 42	2 : 1
3 : 10	3 : 1
4 : 6	...
5 : 30	95 : 1
6 : 11	

그림 3.2 Zoom 인 경우 Cluster 들

본 논문에서는 Fuzzy IF-THEN Rule 에 기반한 유추를 사용한다. 그러기 위해서는 우선 Fuzzy 도메인으로 변환할 수 있는 Fuzzifier 가 필요하다. 아래 식은 본 논문에서 사용하는 Fuzzifier 이다.

$$x_1 : \arg \max_{i \in \text{Parallel}} (N_i) / \sum_{i \in \text{Parallel}} -p_i \log p_i \text{----(식 3.5)}$$

$$x_2 : \arg \max_{i \in \text{Inter}} (N_i) / \sum_{i \in \text{Inter}} -p_i \log p_i \text{----(식 3.6)}$$

3.4 Fuzzy Rules

그림 3.3 은 변환된 Fuzzy 데이터를 Inference 하기 위한 Fuzzy Rules 이다. x1 과 x2 는 3.3 절에서 소개한 fuzzifier 를 통해 얻어진 값이 Fuzzy Inference 의 입력으로 쓰인다.

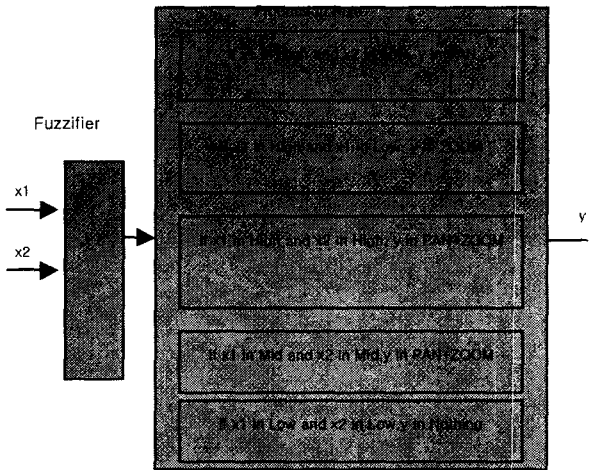


그림 3.3 Fuzzy Production Rule

4. 실험 결과

실험은 MPEG-1 IBBP 형식으로 인코딩 된 비교적 다양한 장르의 스포츠 영상으로 실험을 한다. 비교방법으로는 Y. Tan[3] 이 제안한 방법을 이용한다. 아래 그림은 실제 MPEG 영상의 모션벡터들을 캡처한 장면이다.

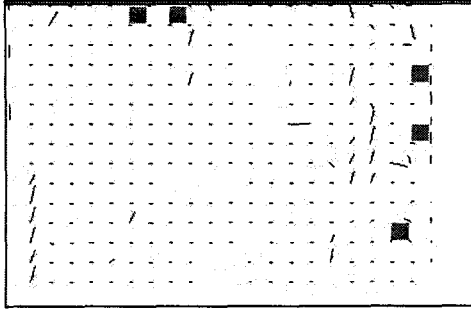


그림 4.1 Tracking 의 경우 모션벡터 분포

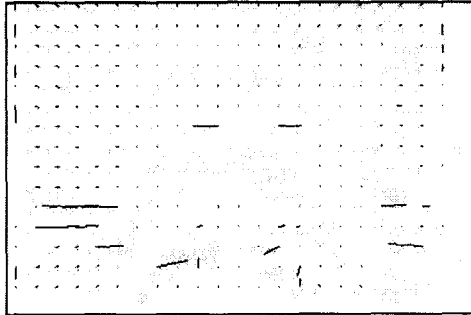


그림 4.2 Zoom 인 경우 모션벡터 분포

표 4.1 은 본 논문에서 제안하는 방법으로 구현한 것과 Y. Tan[3]가 제안하는 MPEG 압축 도메인상에서 Affine 모델을 이용하여 실험한 결과를 비교한 것이다. 표 4.1 의 Missed 항목은 카메라 움직임이 존재하는데 찾아내지 못하는 구간을 의미하며, False 항목은 카메라 움직임을 잘못 판별한 것을 의미한다. 표 4.1 에서 판별이 안되는 경우는 특정 크기 이상의 물체가 화면에서 움직일 때이다. 따라서 본 논문과 비교논문 모두 특정 크기 이상이 물체가 화면에서 움직일 때 제대로 판별이 안되는 문제점을 갖고 있다.

표 4.1 실험 결과

종류	길이	방법	Total	Missed	False	Precision (%)	Recall (%)
야구	5 분 22 초	본방법	15	2	1	80	92
		Affine	15	2	3	67	85
축구	5 분 46 초	본방법	20	3	1	80	94
		Affine	20	3	2	75	88
골프	2 분 45 초	본방법	16	2	2	75	86
		Affine	16	2	3	69	79

5. 결론

본 논문은 MPEG 비디오 데이터에 존재하는 모션 벡터가 카메라 움직임에 따라 특징적으로 분포하는 특성을 이용하여 카메라 움직임 정보를 추출하는 방법을 제안한다. 움직임이 많은 대부분의 동영상 예를 들면, 스포츠 영상의 경우 카메라의 움직임이 거의 Zoom, Pan, Tilt 들로 이루어져 있다. 따라서, 실험 결과에서 알 수 있듯이 기존의 방법에 대해 제한된 상황에서 좋은 성능을 보임을 알 수 있다. 클러스터링 된 모션벡터는 기존의 카메라 추출방법과는 달리 영상 내에서 객체의 움직임 정보 추출을 위한 실마리도 제공한다. 향후 연구 계획으로는 크기가 큰 객체를 중심으로 카메라 움직임 있는 경우 MPEG Video 데이터 중 I-frame 을 고려한 연구가 이루어져야 할 것이다.

참고문헌

- [1] S. Jeannin, B. Mory, "Video Motion Representation for Improved Content Access", *IEEE Transactions on Consumer Electronics*, August 2000, vol. 6, no. 3, pp. 645-655.
- [2] A. Akutsu, Y. Tonomura, H. Hashimoto, Y. Ohba, "Video Indexing using motion vectors", *The Proceeding of SPIE VCIP '92*, vol. 1818, pp. 1522-1530.
- [3] Y. Tan, D. D. Saur, S. R. Kulkarni, "Rapid Estimation of Camera Motion from compressed Video with application to Video Annotation", *IEEE Transactions on Circuits and Systems for Video Technology*, Feb. 2000, vol. 10, no. 1, pp. 133-146.
- [4] S. Jeannin, A. Divakaran, "MPEG-7 Visual Motion Descriptors", *IEEE Transactions on Circuits and Systems for Video Technology*, June 2001, vol. 11, no. 6, pp. 720-724.