

論文99-36S-8-8

# 비디오 데이터 분석에 의한 카메라의 동작 추출

(Extracting Camera Motions by Analyzing Video Data)

張錫宇\*, 李根壽\*\*, 崔濬一\*

(Seok Woo Jang, Keun Soo Lee, and Hyung Il Choi)

## 요 약

본 논문에서는 비디오 데이터를 분석하여 다양한 카메라의 동작을 정량적으로 추출하는 방법을 제안한다. 본 논문에서 제안하는 카메라의 동작 추출 방법은 어파인 모델을 이용한 방법으로 인접 영상으로부터 추출한 동작 벡터를 어파인 모델에 적용하고 회귀분석법을 통해 어파인 모델을 구성하는 파라미터를 구한다. 그런 다음, 파라미터들의 크기를 분석하고 상호 관계를 해석하여 카메라의 동작을 추출한다. 본 논문에서는 잡음이 포함된 동작 벡터를 필터링하여 사용하므로 잡음에 강건한 결과를 얻을 수 있다. 그리고 어파인 모델을 구성하는 파라미터만을 분석함으로써 카메라의 다양한 동작을 간단하면서도 비교적 정확하게 추출한다. 실험 결과는 카메라의 동작을 정확하게 추출하고 있음을 보여준다.

## Abstract

This paper presents an elegant method, an affine-model based approach, that can qualitatively estimate the information of camera motion. We define various types of camera motion by means of parameters of an affine-model. To get those parameters from images, we fit an affine-model to the field of instantaneous velocities, rather than raw images.

We correlate consecutive images to get instantaneous velocities. The size filtering of the velocities are applied to remove noisy components, and the regression approach is employed for the fitting procedure. The fitted values of the parameters are examined to get the estimates of camera motion. The experimental results show that the suggested approach can yield the qualitative information of camera motion successfully.

## I. 서 론

대용량의 저장 매체, 방대하게 통합된 통신망 및 고속 프로세서 등의 급속한 발달에 따라 문자나 숫자 등과 같이 단일한 형태의 자료인 모노미디어로부터 동영상과 같이 다양한 형태의 자료인 멀티미디어로 관심이 집중되고 있다. 특히, 최근에 정보 서비스의 주역으로

떠오르고 있는 비디오 데이터는 멀티미디어 데이터 중에서도 동영상과 함께 오디오 및 텍스트 정보를 포함하고 있는 복합적 성격의 데이터로서 영상 매체의 발달과 더불어 그 중요성이 점차 증가하고 있다. 따라서, 급증하고 있는 비디오 데이터를 효율적으로 분석하여 보다 계층적이고 구조적인 형태로 재구성하고 이를 관리해야 할 필요성이 절실히 요구되고 있다. 이에 따라 자막 인식, 장면 전환 검출, 키 프레임 추출, 비디오 색인 및 검색 등과 같이 비디오 데이터와 관련된 많은 연구가 활발히 진행되고 있다<sup>[1-4]</sup>.

본 논문에서 추출하고자 하는 카메라의 동작은 컴퓨터 비전이나 비디오 색인 및 검색 등의 여러 분야에서 사용될 수 있는 강력하고 중요한 요소이다. 예를 들면,

\* 正會員, 崇實大學校 컴퓨터學部

(School of Computing Soongsil University)

\*\* 正會員, 韓京大學校 컴퓨터工學科

(Dept. of Computer Eng., Hankyong Nat'l Univ.)

接受日字:1999年2月18日, 수정완료일:1999年6月30日

서버 장치를 구동하여 시스템에 부착된 카메라를 움직이면서 동적으로 변화하는 이동 물체를 추적하는 시스템에서 물체의 정확한 추적에 필요한 카메라의 동작 정보를 분석하는 작업은 매우 중요하다. 또한, 비디오 데이터를 분석하여 패닝하는 장면, 틸팅하는 장면, 줌인하는 장면과 같이 카메라의 동작에 따른 장면 전환을 검출하는 데도 활용될 수 있다.

카메라의 동작을 추출하기 위한 기존의 연구 방법에는 히스토그램을 이용한 방법, 호프 변환(Hough transformation)을 이용한 방법, X-ray 영상을 이용한 방법, 시공간적인 패턴을 이용한 방법 등이 있다<sup>[3,5-7]</sup>. [3]에서는 동작 벡터에 대한 히스토그램을 구하여 카메라의 동작을 추출하므로 각 카메라의 동작에 대한 히스토그램 특성을 알아야 하고 카메라의 동작이 혼합될 경우에는 히스토그램 분석이 어렵다. [5]에서는 블록 정합을 이용하여 추출된 동작 벡터를 호프 공간으로 변환하고 이들의 분포를 분석함으로써 카메라의 동작을 추출하지만 잡음에 민감한 동작 벡터를 직접 이용하므로 오류 발생의 확률이 높다. [6]에서는 에지 영상을 구하고 수평 방향 및 수직 방향의 X-ray 영상을 추출한 후 각 영상에 대한 에지 방향의 공간적인 분포를 이용하여 카메라의 동작을 구한다. 그러나 매 영상에 대한 에지 추출 작업을 수행해야 하므로 시간이 많이 소요된다. [7]에서는 영상의 특정 부분에 해당하는 레이블 맵의 시공간적인 패턴에 대한 방향성을 분석하여 카메라의 동작을 추출한다. 그러나 시공간적인 패턴의 방향에 대한 분포만을 이용하여 카메라의 단일한 동작을 구하므로 여러 가지 카메라 동작이 혼합된 경우에는 개별적인 카메라 동작을 구할 수 없다.

카메라의 동작 측정이 중요하고 다양한 목적으로 사용될 수 있지만 자동화된 카메라의 동작 추출에 대한 연구, 특히 비디오 데이터에 초점을 맞춘 연구는 상당히 미흡한 수준이다. 따라서 본 연구에서는 기존 방법의 문제점을 최대한 배제하면서 비디오 데이터를 중심으로 정량적인 카메라의 동작을 자동하는 추출하는 방법을 제안한다. 본 논문에서는 시간적인 문제를 고려하여 화소 단위가 아니라 셀 단위의 동작 벡터를 추출하여 이용하고, 잡음이 포함된 동작 벡터를 직접 이용할 경우 발생하는 오류를 제거하기 위해 카메라의 동작에 따라 복잡하게 생성되는 동작 벡터를 전체 구조를 유지하면서 강건하게 필터링하였다. 그리고 서로

다른 카메라의 동작을 구하기 위해 두 가지 이상의 프로시저를 혼합하여 사용하지 않고 카메라의 동작에 대한 동작 벡터를 하나의 모델인 어파인 모델만을 이용하여 일관성있게 분석하였다. 또한, 어파인 모델을 구성하는 파라미터들을 분석함으로써 카메라의 여러 동작이 혼합된 경우에도 개별적인 카메라의 동작을 간단하게 확인할 수 있다.

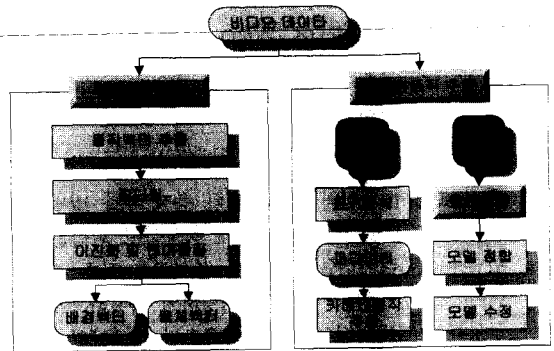


그림 1. 카메라 동작 추출 방법의 구조  
Fig. 1. Overall block diagram.

그림 1은 본 논문에서 제안하는 카메라 동작 추출 방법의 전체적인 구조를 보여주는데, 크게 동작 벡터 추출 단계와 카메라의 동작 추출 단계로 구성된다. 동작 벡터 추출 단계에서는 비디오 데이터를 받아들여 인접 영상으로부터 상관관계(correlation)에 기반한 방법을 사용하여 동작 벡터를 추출한다. 먼저, 영상을 셀 단위로 분할하고 셀 단위로 칼라 특징을 정의한 후 정의된 특징에 의해 구성되는 정합 척도를 일정한 범위 내의 셀들에 적용하여 가장 높은 정합 척도를 갖는 셀을 대응셀로 정의한다. 대응셀이 확인되면 동작 벡터는 각 셀을 대표하는 좌표점의 차이로 계산된다. 그러나 영상의 균일한 칼라 값을 갖는 영역에서 동작 벡터를 추출할 경우에는 부정확한 셀 정합으로 인해 필연적으로 잡음이 많이 포함된다. 따라서 본 논문에서는 다양한 카메라의 동작에 따라 방향과 크기가 매우 다른 동작 벡터의 전체 구조를 유지하면서 강건하게 필터링하여 잡음이 포함된 부정확한 동작 벡터를 배제하고 정확하게 추출된 동작 벡터만을 고려한다. 그런 다음, 필터링된 동작 벡터를 이진화 및 레이블링하여 이동 물체에 해당하는 동작 벡터는 물체 벡터로 할당하고 배경에 해당하는 동작 벡터는 배경 벡터로 할당한다. 일반적으로, 카메라를 움직이면서 촬영한 비디오 데이터의 물체 영역에는 물체의 움직임과 더불어 카메라

라의 동작도 포함되어 있으므로 카메라의 순수한 동작은 배경에만 포함된다. 따라서 본 논문에서는 이동 물체에 해당하는 동작 벡터는 제외하고 배경에 해당하는 동작 벡터만을 받아 들여 카메라의 동작을 추출하므로 보다 정확한 결과를 얻을 수 있다.

카메라의 동작 추출 단계에서는 동작 벡터 추출 단계에서 추출한 물체 벡터를 이용하여 물체 모델을 생성하고 이동 물체를 계속 추적하면서 카메라의 동작을 구한다. 이동 물체를 추적하는 이유는 연속적으로 입력되는 비디오 데이터에서 이동 물체 부분을 배제하고 카메라의 동작을 구하기 위함이다. 이동 물체의 추적 작업은 예측 알고리즘인 칼만 필터를 이용하고 이동 물체의 모델을 정합 및 수정하면서 진행한다. 카메라의 동작을 구하기 위해서는 6개의 파라미터로 구성되는 어파인 모델과 통계적인 방법인 회귀분석법을 이용한다. 서로 다른 카메라의 동작에 따라 매우 복잡한 구조를 형성하는 동작 벡터를 하나의 모델만을 사용하여 일관성있게 분석하기 위해 필터링된 동작 벡터를 어파인 모델에 적용하고, 어파인 모델을 구성하는 파라미터를 추출하기 위해 통계적인 방법인 회귀분석법을 사용한다. 그런 다음, 파라미터들의 크기를 분석하고 상호 관계를 해석하여 카메라의 동작을 유도한다. 어파인 모델을 구성하는 파라미터들이 카메라의 동작을 표현하므로 여러 가지 카메라의 동작이 혼합된 경우에도 파라미터들을 분석하여 간단하게 개별적인 카메라의 동작을 추출할 수 있다.

1장에서는 연구 동기와 카메라 동작 추출의 전체적인 개요를 설명하였다. 2장에서는 동작 벡터를 추출하는 방법에 대해 설명하고, 3장에서는 카메라의 동작과 어파인 모델의 구조에 대해 설명한다. 4장에서는 회귀분석법을 통한 파라미터 추출 방법에 대해 설명하고, 5장에서는 실험 결과를, 그리고 6장에서는 결론에 대해 기술한다.

## II. 동작 벡터 추출

본 장에서는 입력 영상으로부터 동작 벡터를 추출하는 방법에 대해 설명한다. 동작 벡터는 영상 좌표의 수평 및 수직 이동을 나타내는 2차원 벡터로서 본 논문에서는 상관관계(correlation)에 기반한 동작 벡터 추출 방법을 사용한다. 이 방법에서는 t 시점에서 촬영된 영상의 한 점 또는 영역에 대응되는 점을  $t + \Delta t$

시점에서 촬영된 영상에서 확인하여 동작 벡터를 구한다. 본 논문에서는 각 영상을 일정한 크기의 셀로 분할한 후 셀 단위로 특징을 정의하고 정의된 특징에 의해 구성되는 정합 척도를 일정한 범위 내의 셀들에 적용하여 가장 높은 정합 척도를 갖는 셀을 대응 셀로 정의한다. 대응셀이 확인되면 동작 벡터는 각 셀을 대표하는 좌표점의 차이로 계산된다. 그리고 잡음이 포함된 동작 벡터를 제거하기 위해 필터링을 수행한다.

본 논문에서는 입력 영상을 받아 들여 일정한 크기의 셀로 분할한 후 셀 단위의 정합을 위해서 칼라 특징을 사용한다. 일반적으로, 칼라 영상을 코딩하기 위해서 RGB 칼라 공간을 많이 사용하지만, 영상 처리를 수행하려 할 때 세 가지의 RGB 칼라 값을 모두 처리해야 하기 때문에 영상 처리에 비효율적이다. 그리고 RGB 칼라 공간은 영상의 밝기 변화에 매우 민감하게 작용할 뿐만 아니라 조명의 크기나 방향에 따라 칼라 값의 차이가 크게 발생할 수 있으므로 동영상 처리에 부적절한 것으로 알려져 있다<sup>[8]</sup>. 따라서 본 논문에서는 조명의 변화에 강건한 칼라 특징을 추출하기 위해 YIQ 칼라 공간을 사용한다. YIQ 칼라 공간은 NTSC에서 정한 합성 칼라 비디오 표준 방식으로 밝기값의 변화에 대해 색상이나 채도가 변하지 않는 특징을 가지고 있다<sup>[9, 10]</sup>.

YIQ 칼라 공간은 YUV 칼라 공간에서 유도된 것으로 Y(Luminance) 요소는 밝기값을 나타내고, I(Inphase) 요소와 Q(Quadrature) 요소는 두 요소를 합성하여 색상과 채도를 나타낸다. 식 (1)은 I, Q 요소와 색상, 채도와의 관계를 나타내며, 식 (2)는 RGB 칼라 공간을 YIQ 칼라 공간으로 변환하는 식을 나타낸다<sup>[11]</sup>.

$$\text{색상} = \tan^{-1}\left(\frac{Q}{I}\right) \quad (1)$$

$$\text{채도} = \sqrt{I^2 + Q^2}$$

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.30 & 0.59 & 0.11 \\ 0.60 & -0.27 & -0.32 \\ 0.21 & -0.52 & 0.31 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

본 논문에서는 YIQ 칼라 공간에서 밝기값을 나타내는 Y 요소를 제외한 I와 Q 요소만을 이용하여 조명의 변화에 강건한 칼라 특징을 추출한다. 본 논문에서 사용하는 칼라 특징은 셀에 포함되는 영상 화소들의 I와 Q 요소의 평균과 표준편차를 이용하여 정의된

다. 따라서 본 논문에서 사용하는 칼라 특징은 식 (3)과 같이 정의된다.

$$C = [\mu_I \ \sigma_I \ \mu_Q \ \sigma_Q] \quad (3)$$

식 (4)와 식 (5)는 영상 좌표  $(x,y)$ 에서 시작하는 셀에 포함되는 영상 화소들의 I와 Q 요소의 평균과 표준 편차를 구하는 식을 보인다. 식 (4)와 식 (5)에서 N과 M은 셀의 높이와 너비를 나타낸다.

$$\mu_I = \frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} I(x+i, y+j) \quad (4)$$

$$\mu_Q = \frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} Q(x+i, y+j)$$

$$\sigma_I = \sqrt{\frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I(x+i, y+j) - \mu_I)^2} \quad (5)$$

$$\sigma_Q = \sqrt{\frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (Q(x+i, y+j) - \mu_Q)^2}$$

본 논문에서는 상관관계에 기반한 방법으로 동작 벡터를 추출한다. 따라서 식 (3)과 같은 특징을 정의하고, 정의된 특징에 의하여 구성되는 정합 척도를 설정하여 대응셀을 확인함으로써 동작 벡터를 구한다. 본 논문에서는 식 (6)과 같은 정규화된 상관관계 척도를 정합 척도로 사용한다.

$$CM_{t, t+\Delta t}(I) = \frac{\sigma_{t, t+\Delta t}(I)}{[\sigma_t(I) \cdot \sigma_{t+\Delta t}(I)]^{1/2}} + \frac{\sigma_{t, t+\Delta t}(Q)}{[\sigma_t(Q) \cdot \sigma_{t+\Delta t}(Q)]^{1/2}} \quad (6)$$

$$\sigma_t(I) = \frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_t(x+i, y+j) - \mu_{I_t})^2$$

$$\sigma_{t+\Delta t}(I) = \frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_{t+\Delta t}(x+i, y+j) - \mu_{I_{t+\Delta t}})^2$$

$$\sigma_t(Q) = \frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (Q_t(x+i, y+j) - \mu_{Q_t})^2$$

$$\sigma_{t+\Delta t}(Q) = \frac{1}{NM} \cdot \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (Q_{t+\Delta t}(x+i, y+j) - \mu_{Q_{t+\Delta t}})^2$$

$$\sigma_{t, t+\Delta t}(I) = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_t(x+i, y+j) - \mu_{I_t})$$

$$\cdot (I_{t+\Delta t}(x+i, y+j) - \mu_{I_{t+\Delta t}})$$

$$\sigma_{t, t+\Delta t}(Q) = \frac{1}{NM} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (Q_t(x+i, y+j) - \mu_{Q_t})$$

$$\cdot (Q_{t+\Delta t}(x+i, y+j) - \mu_{Q_{t+\Delta t}})$$

이와 같이 정의된 정합 척도를 일정한 범위 내의

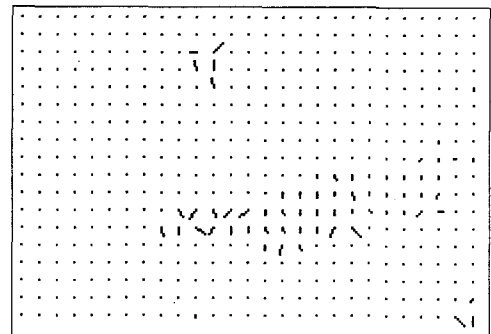
셀들에 적용하여 가장 높은 값을 갖는 셀을 대응셀로 확정한다. 대응셀이 확정되면 동작 벡터는 각 셀을 대표하는 좌표점의 차이로 계산된다. 그림 2는 입력 영상으로부터 동작 벡터를 추출한 예를 보여준다.



(a) Input Image t



(b) Input Image t+Δt



(c) Motion vector

그림 2. 동작 벡터의 예

Fig. 2. Example of motion vector.

그림 2에서와 같이 상관관계에 기반한 방법으로 추출된 동작 벡터에는 잘못된 동작 벡터가 많이 포함되어 있다. 이는 균일한 칼라 값을 갖는 영역에서 셀 간의 정합 척도를 측정할 경우 유사한 값을 나타내게 되고 이로 인해 부정확한 셀 정합이 발생했기 때문이다. 따라서 본 논문에서는 잘못된 동작 벡터가 포함된 동

작 벡터를 어파인 모델에 직접 적용하지 않고 필터링 과정을 수행한 후 적용한다. 일반적으로, 카메라의 다양한 동작에 따라 방향과 크기가 서로 다른 동작 벡터들이 추출되고 동작 벡터가 복잡한 구조를 형성하므로 동작 벡터의 전체 구조를 유지하면서 강건하게 필터링하는 작업은 매우 어렵다.

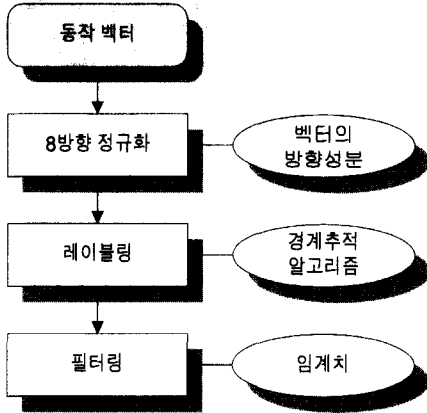


그림 3. 동작 벡터 필터링 과정  
Fig. 3. Process of motion vector filtering.

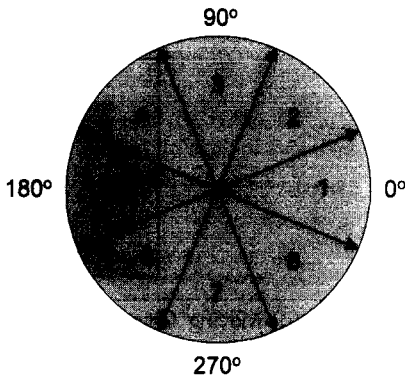
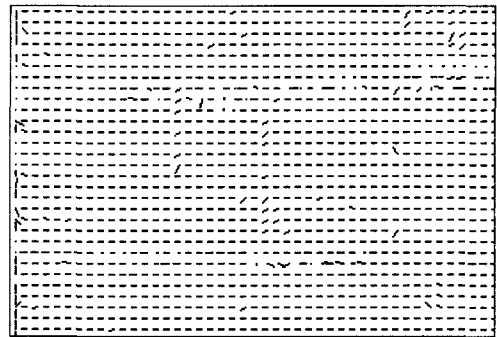


그림 4. 8방향 정규화  
Fig. 4. Normalization with 8 direction.

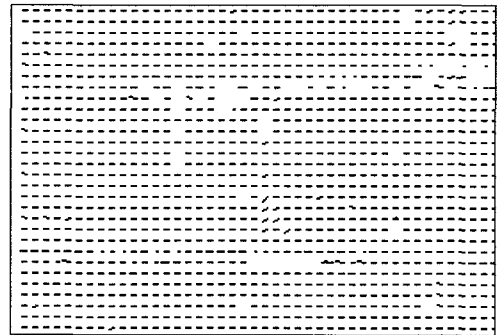
잘못 추출된 동작 벡터들은 올바르게 추출된 주위의 동작 벡터와는 다른 방향 성분을 가지면서 작은 범위에 걸쳐 분포합니다. 따라서 본 논문에서는 방향 성분에 따라 동작 벡터를 정규화하고 정규화된 동작 벡터를 레이블링하여 크기가 작은 레이블들을 잘못 추출된 동작 벡터로 고려하고 이들을 제거합니다. 이와 같이 동작 벡터의 방향 성분을 이용하여 간단하면서도 강건하게 동작 벡터를 필터링하는 전체 과정은 그림 3과 같다. 8방향 정규화 과정에서는 동작 벡터의 방향 성분을 8방향으로 정규화한다. 8 방향은 그림 4와 같이

360°를 -22.5°에서부터 시작하여 등간격으로 8등분하여 사용한다. 레이블링 과정에서는 8방향으로 정규화된 동작 벡터를 레이블링하여 8진화된 벡터를 생성한다. 본 논문에서는 동작 벡터를 8진화하기 위해 이진화된 데이터를 레이블링하기 위해 사용하는 경계 추적에 의한 1 패스 레이블링 알고리즘<sup>[12]</sup>을 8진화된 데이터를 레이블링할 수 있도록 수정하여 사용한다. 그리고 잘못된 동작 벡터 제거 과정인 필터링 과정에서는 레이블링된 동작 벡터의 레이블 크기를 계산하여 크기가 작은 레이블을 제거한다.

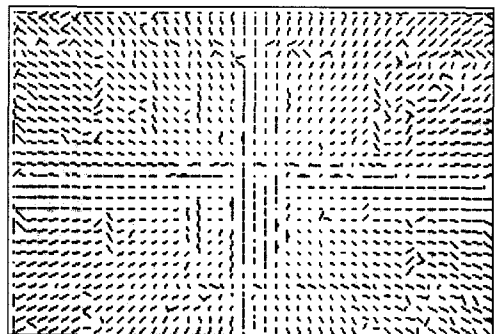
그림 5에서는 카메라의 패닝 동작과 줌인 동작이 발생했을 경우 동작 벡터를 필터링한 결과를 보여준다.



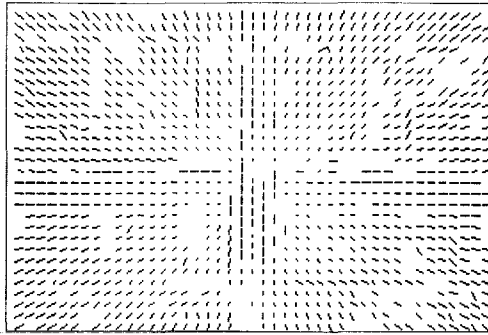
(a) Motion vector 1



(b) Filtered motion vector 1



(c) Motion Vector 2



(d) Filtered motion vector 2

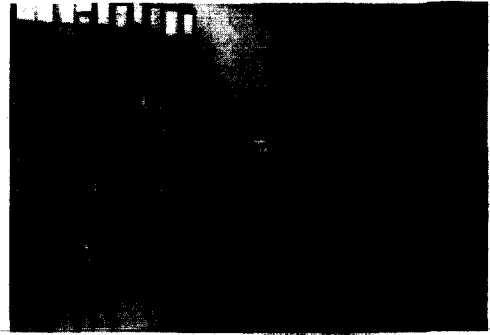
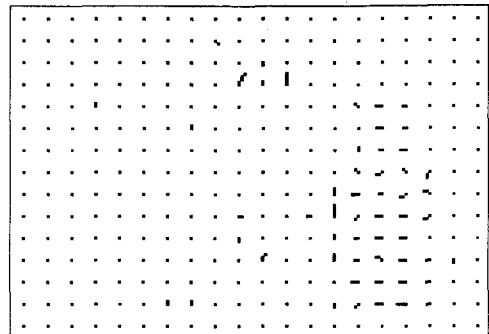
(b) Input Image  $t+\Delta t$ 

그림 5. 동작 벡터 필터링

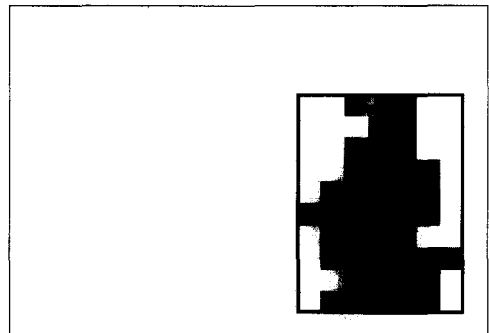
Fig. 5. Motion vector filtering.

필터링한 동작 벡터는 이진화 및 레이블링 과정을 통해 이동 물체 영역에 해당하는 동작 벡터와 배경에 해당하는 동작 벡터로 분리한다. 일반적으로, 입력 영상으로부터 추출된 동작 벡터 중에서 이동 물체에 해당하는 동작 벡터는 배경에 해당하는 동작 벡터보다 그 수가 적으며 배경에 해당하는 동작 벡터와는 그 크기가 다르다. 이런 사실을 근거로 동작 벡터의 크기에 대한 히스토그램을 그린 후 히스토그램을 이진화할 수 있다. 그런 다음, 이진화된 동작 벡터를 레이블링한 후 배경에 해당하는 동작 벡터는 배경 벡터로 할당하고 이동 물체에 해당하는 동작 벡터는 물체 벡터로 할당한다. 배경 벡터는 단순한 카메라의 동작을 추출하기 위해 사용되며 물체 벡터는 이동 물체를 계속 추적하기 위해 사용된다.

본 논문에서 수행하는 이동 물체 추적 작업은 칼만 필터를 이용하고 물체 모델을 정합 및 수정하면서 진행된다<sup>[13, 14]</sup>. 그림 6은 동작 벡터의 이진화 및 레이블링을 통해 배경 벡터와 물체 벡터를 분리한 결과를 보여주고 있다.



(c) Motion vector



(d) Moving Object

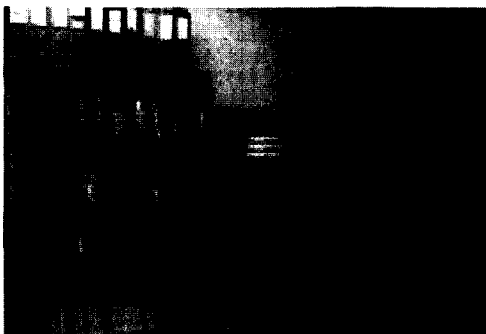
그림 6. 동작 벡터 분리

Fig. 6. Separation of motion vector.

### III. 카메라의 동작과 어파인 모델

본 장에서는 카메라 동작의 종류에 대해 설명하고, 영상의 움직임을 나타내는 동작 벡터를 표현하는 어파인 모델의 구조에 대해 설명한다. 그리고 카메라의 동작과 어파인 모델을 구성하는 파라미터 사이의 관계에 대해 설명한다.

카메라는 단일하게 또는 복합적으로 동작하여 다양한 장면을 연출할 수 있지만 기본적으로 그림 7과 같이 정지, 패닝, 틸팅, 줌잉, 롤링, 수평 이동, 수직 이동, 돌링의 8 가지 동작으로 분류할 수 있다. 그림 7

(a) Input image  $t$

의 a에서와 같이 수평 이동, 수직 이동, 돌림은 카메라의 몸체가 직접 이동하는 동작이다. 수평 이동과 수직 이동은 수평과 수직 방향으로 이동하는 동작이며 돌림은 깊이 방향으로 이동하는 동작이다. 수평 이동은 트래킹(tracking), 그리고 수직 이동은 붐잉(booming)이라는 표현을 사용하기도 한다<sup>[5]</sup>. 그림 7의 b에서와 같이 패닝, 틸팅, 줌잉, 롤링은 카메라의 몸체가 고정된 상태에서 머리 부분만이 회전하는 동작이다. 패닝과 틸팅은 수평과 수직 방향으로 회전하는 동작이며, 줌잉은 카메라의 초점 거리를 조절하는 동작, 그리고 롤링은 관측 방향 즉, 깊이 방향으로 회전하는 동작이다.

카메라의 동작 중 정지 동작은 매우 일반적으로 사용되고 있으며, 패닝과 틸팅은 이동하는 물체를 추적하거나 주변의 풍경을 묘사하기 위해서 카메라를 좌우 또는 상하로 회전할 때 사용한다. 수직 이동과 수평 이동은 이동하는 물체를 추적하기 위해 사용하며, 줌잉과 돌림은 장면을 강조하거나 카메라 시계의 범위를 넓히기 위해 사용한다. 그리고 롤링은 카메라를 회전시켜 특수한 장면을 연출할 때 사용한다. 이런 동작 외에도 여러 동작이 혼합되거나 특수한 장면을 연출하기 위한 카메라의 동작이 있을 수 있다.

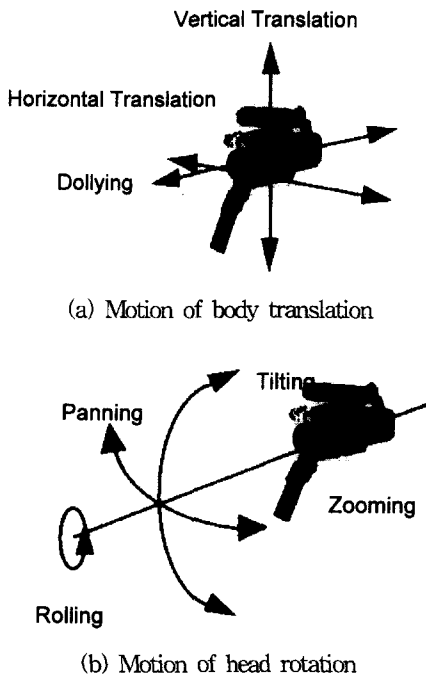


그림 7. 카메라 동작의 종류  
Fig. 7. Camera motions.

다양하게 카메라를 움직이면서 촬영한 영상은 포함된 카메라의 동작에 따라 서로 다른 동작 벡터를 형성한다. 본 논문에서는 동작 벡터를 이용하여 카메라의 동작을 추출하기 위해 기본적으로 어파인 모델을 활용한다. 일반적으로, 어파인 모델은 스케일(scale), 회전(rotation), 전이(translation), 그리고 이들의 조합을 포함하는 모델로 이동 물체의 크기나 모양이 변하는 동적인 환경에서 물체의 경계를 분할 및 추적하면서 물체의 경계가 형성하는 변형을 어파인 모델을 사용하여 적합시키는 것과 같이 기하학적인 다양한 변화를 수용할 수 있는 모델을 필요로 할 때 많이 사용한다<sup>[15]</sup>.

본 논문에서도 영상을 분석하여 영상 전체의 움직임을 나타내는 동작 벡터를 파라미터 모델인 어파인 모델에 적합시키고, 어파인 모델을 구성하는 파라미터를 추출하여 분석함으로써 동적으로 변화하는 다양한 카메라의 동작을 추출한다.

영상 전체의 움직임을 하나의 파라미터 모델로 표현하는 것은 영상 내에서 동작 벡터의 공간적인 변화를 가정하는 것이다. 즉, 영상 전체를 하나의 평면으로 가정하고 영상의 움직임을 모델링하는 것이다. 본 논문에서는 하나의 평면으로 가정된 영상의 움직임을 식 (7)과 같이 6개의 파라미터로 구성된 어파인 모델로 표현한다.

$$u(x-x_0, y-y_0) = a_1 + a_2(x-x_0) + a_3(y-y_0) \quad (7)$$

$$v(x-x_0, y-y_0) = a_4 + a_5(x-x_0) + a_6(y-y_0)$$

여기서,  $u(x, y)$ 와  $v(x, y)$ 는 영상 좌표  $(x, y)$ 에서의 동작 벡터의 수평 및 수직 성분을 나타내고,  $a = (a_1, a_2, a_3, a_4, a_5, a_6)$ 는 측정될 파라미터 벡터를 나타낸다. 그리고 영상 좌표  $(x, y)$ 는 영상의 중심  $(x_0, y_0)$ 를 기준으로 평행 이동된다.

식 (7)에서 파라미터  $a_i$ 는 영상의 움직임, 즉 카메라의 동작을 정량적으로 표현할 수 있다. 예를 들면,  $a_1$ 은 동작 벡터의 수평 성분에서 변수  $x$ 와  $y$ 에 독립적인 상수로서 수평 성분의 위치를 수평 양의 방향 또는 수평 음의 방향으로 이동시키는 효과를 나타낸다. 따라서  $a_1$ 은 영상의 수평 이동, 즉 카메라의 패닝 동작에 해당한다.

$a_4$ 는 동작 벡터의 수직 성분을 구성하는 상수로서 수직 성분의 위치를 수직 양의 방향 또는 수직 음의

방향으로 이동시키는 효과를 나타낸다. 따라서  $a_4$ 는 영상의 수직 이동, 즉 카메라의 틸팅 동작에 해당한다. 그리고 파라미터  $a_i$ 의 조합을 이용하여 식 (8)과 같이 발산(divergence), 회전(curl) 및 변형(deformation)을 정의할 수 있다.

$$divergence = a_2 + a_6 = (u_x + v_y) \quad (8)$$

$$curl = -a_3 + a_5 = -(u_y - v_x)$$

$$deformation = a_2 - a_6 = (u_x - v_y)$$

식 (8)에서 첨자  $x$ 와  $y$ 는 영상 속도의 부분 도함수 (partial derivative)를 나타낸다. 그리고 발산은 등방향(isotropic)의 확대 및 축소를 나타내고, 회전은 관측 방향과 수직이 되는 회전을 나타내며, 변형은 수평 방향이나 수직 방향으로 늘이거나 줄이는 효과를 나타낸다.

따라서 확산은 일반적인 카메라의 동작 중 줌인 (zooming)과 동일한 효과를 나타내고, 회전은 카메라의 회전에 해당하는 롤링(rolling), 그리고 변형은 특수 효과를 발생시키기 위해 카메라의 시계를 수평 또는 수직 방향으로 밀폐시키는 효과를 나타내는데, 이런 효과를 그림으로 표현하면 그림 8과 같다. 그림 8에서 가는 실선은 변형되기 전의 영상 영역을 나타내고 굵은 실선은 변형된 후의 영상 영역을 나타낸다. 그리고 + 기호와 - 기호는 위에서 정의한 각 파라미터 또는 파라미터들의 조합이 양의 값 또는 음의 값을 나타내고, 값의 부호가 변경되면 해당되는 카메라의 동작 방향이 반대가 된다.

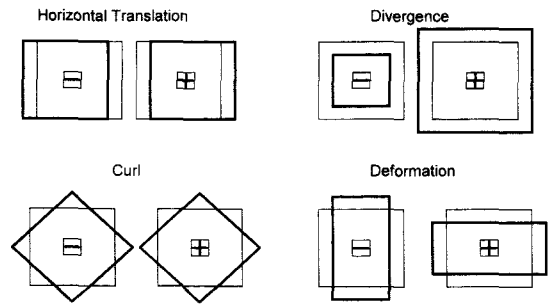


그림 8. 영상의 움직임 효과  
Fig. 8. Effect of image movement.

표 1은 각각의 파라미터 또는 파라미터들의 조합이 나타내는 값의 부호에 따라 표현되는 다양한 카메라의 동작을 보여준다.

표 1. 파라미터가 표현하는 카메라의 동작  
Table 1. Camera motions by means of parameters.

파라미터	부 호	카메라의 동작
$a_i$	$a_i = 0$	Stationary
$a_1$	$a_1 > 0$	Translation(Panning) to the Left
	$a_1 < 0$	Translation(Panning) to the Right
$a_4$	$a_4 > 0$	Translation(Tilting) Up
	$a_4 < 0$	Translation(Tilting) Down
$a_2 + a_6$	$a_2 + a_6 > 0$	Divergence(Zooming) In
	$a_2 + a_6 < 0$	Divergence(Zooming) Out
$-a_3 + a_5$	$-a_3 + a_5 > 0$	Curl(Rolling) to the Right
	$-a_3 + a_5 < 0$	Curl(Rolling) to the Left
$a_2 - a_6$	$a_2 - a_6 > 0$	Deformation In
	$a_2 - a_6 < 0$	Deformation Out

IV. 회귀분석법에 의한 파라미터 추출

본 장에서는 회귀분석법을 이용하여 어파인 모델을 구성하는 파라미터를 추출하는 방법에 대해 설명한다. 회귀분석법은 측정값들을 가장 잘 만족시키는 모델의 파라미터를 구하는 문제로, 변수와 변수 사이의 관계를 규명하고자 할 때 사용되는 통계적 방법 중의 하나이다. 이 분석 방법에서는 먼저 변수들 간의 관계를 나타내는 타당한 수학적 모델을 설정하고, 변수들의 측정된 값을 이용하여 그 모델을 추정한다. 다음, 추정된 모델에 의해 변수들 간의 관계를 설명하거나 또는 예측 등의 분석에 응용하게 된다.

본 논문에서는 6 개의 파라미터로 구성되는 어파인 모델을 수학적 모델로 사용하고, 동작 벡터의 위치를 나타내는 영상 좌표를 변수들의 측정값으로 받아들이는 회귀분석법을 이용한다. 그리고 회귀분석법을 이용하여 추출한 어파인 모델의 파라미터를 해석하여 카메라의 동작을 구한다.

회귀분석법에서는 변수들 간의 관련성 정도와 관련성 형태 조사 및 예측에 그 목적이 있다. 회귀분석법에서 변수들 간의 관계를 나타내는 수학적 모델을 회귀식(regression equation)이라 하고, 서로 관계를 가지고 있는 변수들 중에서 다른 변수에 의해 영향을 받



는 변수를 종속 변수(dependent variable)라 하며, 종속 변수에 영향을 주는 변수를 독립 변수(independent variable)라 한다. 본 논문에서는 어파인 모델이 회귀식이 되고, 동작 벡터를 구성하는 수평, 수직 요소가 종속 변수가 되며, 동작 벡터의 위치를 나타내는 영상 좌표가 독립 변수가 된다.

본 논문에서는 회귀분석법을 이용하여 어파인 모델의 파라미터를 구하기 위해 회귀식에서 계산된 예측값과 측정값 사이의 오차를 최소화할 수 있는 방법을 이용한다.

본 논문에서는 이러한 방법들 중에서 가장 널리 사용되는 방법으로 오차의 제곱의 합을 최소로 하는 최소제곱법(method of least squares)을 이용한다. 본 논문에서 사용하는 최소제곱법의 기본 수식은 식 (9)와 같이 표현된다.

$$\begin{aligned}
 E &= \sum_{i=1}^n ((u_i, v_i) - (a_1 + a_2x_i + a_3y_i, a_4 + a_5x_i + a_6y_i))^2 \\
 &= \sum_{i=1}^n (u_i - (a_1 + a_2x_i + a_3y_i))^2 \\
 &\quad + \sum_{i=1}^n (v_i - (a_4 + a_5x_i + a_6y_i))^2 \quad (9)
 \end{aligned}$$

식 (9)에서  $i$ 는 측정값들을 나타내는 첨자이다. 최소제곱법을 이용하여 식 (9)의  $E$ 를 최소화하는 파라미터  $a = (a_1, a_2, a_3, a_4, a_5, a_6)$ 를 추출하기 위해서는 식 (9)의 좌변을 파라미터  $a_1, a_2, a_3, a_4, a_5, a_6$  각각에 대해 편미분하여 0으로 놓고 풀면 된다. 식 (9)의 좌변을 각 파라미터에 대해 편미분하면 식 (10)과 같다.

$$\frac{\partial E}{\partial a_1} = 2 \sum_{i=1}^n (u_i - u_0) \cdot (-1) \quad (10)$$

$$\frac{\partial E}{\partial a_2} = 2 \sum_{i=1}^n (u_i - u_0) \cdot (-x_i)$$

$$\frac{\partial E}{\partial a_3} = 2 \sum_{i=1}^n (u_i - u_0) \cdot (-y_i)$$

$$\frac{\partial E}{\partial a_4} = 2 \sum_{i=1}^n (v_i - v_0) \cdot (-1)$$

$$\frac{\partial E}{\partial a_5} = 2 \sum_{i=1}^n (v_i - v_0) \cdot (-x_i)$$

$$\frac{\partial E}{\partial a_6} = 2 \sum_{i=1}^n (v_i - v_0) \cdot (-y_i)$$

식 (10)에서  $u_0$ 는  $a_1 + a_2x_i + a_3y_i$ 를 나타내고,  $v_0$

는  $a_4 + a_5x_i + a_6y_i$ 를 나타낸다. 식 (10)의 각 식을 0으로 놓고 수식을 정리하면 6개의 파라미터로 구성된 6개의 방정식이 생성된다. 따라서 가우시안 소거법을 이용하여 방정식의 해를 구하면 파라미터의 값을 추출할 수 있다. 본 논문에서는 일반적인 가우시안 소거법의 성능을 개선한 스케일 컬럼 피벗(scaled column pivoting) 알고리즘을 사용한다<sup>[16]</sup>.

### V. 실험 결과

본 장에서는 어파인 모델을 이용하여 카메라의 동작을 추출한 실험 결과를 보인다. 실험에서는 이동 물체가 존재하지 않는 실내의 책장 앞에서 카메라를 움직이면서 연속적으로 촬영한 다양한 비디오 데이터와 이동 물체가 있는 사무실 장면을 촬영한 비디오 데이터를 실험 영상으로 사용하였으며, 마이크로소프트의 Visual C++ 컴파일러를 이용하여 Windows 98 운영체제 환경에서 실험하였다.

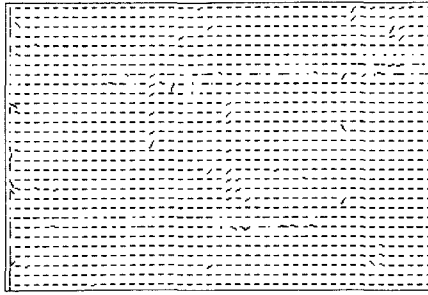
그림 9는 실내의 책장을 촬영한 영상으로 카메라가 오른쪽으로 수평 이동한 경우, 즉 카메라의 패닝 동작이 발생한 경우의 실험 결과를 보여준다.



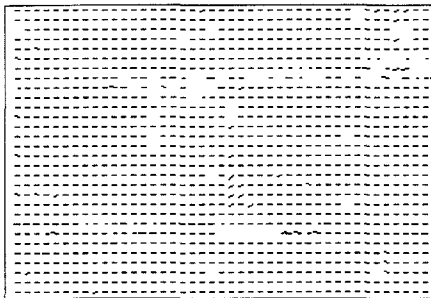
(a) Input image t



(b) Input image t+Δt



(c) Motion vector



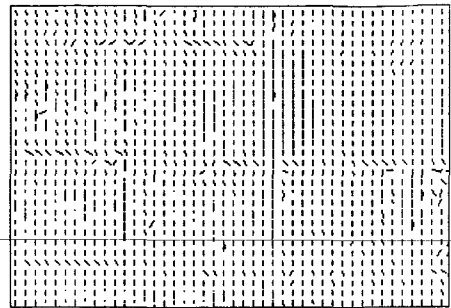
(d) Filtered motion vector

그림 9. 카메라의 패닝 동작  
Fig. 9. Camera Pan.

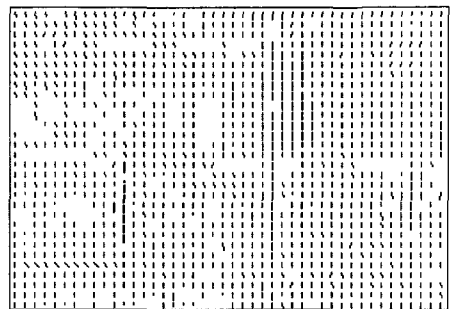
그림 9의 (a)와 (b)는 인접한 입력 영상  $t$ ,  $t+\Delta t$ 를 나타내고, (c)는 입력 영상 (a)와 (b)로부터 상관관계 방법을 기반으로 셀 단위로 추출한 동작 벡터를 보여 주고 있다. 본 논문에서는 셀의 크기를  $7 \times 7$ 로 설정하였다. 그림 9의 (c)에서와 같이 영상의 경계 부분에서는 부정확한 셀 정함으로 인하여 잡음을 많이 포함하고 있다. 그리고 영상에서 칼라값이 균일한 영역에서도 셀 간의 유사도가 동일하여 정확한 셀 정함이 이루어질 수 없으므로 동작 벡터가 정확히 추출되지 못했다. 그림 9의 (d)에서는 동작 벡터를 8방향으로 정규화한 후 레이블링과 잡음 제거 과정을 통해 필터링한 결과를 보여준다. 본 논문에서는 실험적으로 20이라는 임계치를 사용하여 잡음을 제거하였다. 그림에서 보는 바와 같이 동작 벡터에 포함되었던 잡음들이 대부분 제거되었음을 확인할 수 있다.

그림 10은 카메라가 위쪽으로 수직 이동한 경우, 즉 카메라의 틸팅 동작이 발생한 경우의 실험 결과이고, 그림 11은 카메라의 줌인 동작, 즉 화면이 축소되는 동작이 발생한 경우의 실험 결과이다. 그리고 그림 12는 카메라가 왼쪽으로 수평 이동하고 위쪽으로 수직 이동하면서 화면이 확대되는 동작인 패닝, 틸팅, 줌인 동작이 혼합된 경우의 실험 결과이고, 그림 13은 카메

라의 회전 동작이 발생한 경우의 실험 결과이다. 각각에 대한 실험 결과는 그림 9와 동일하다.

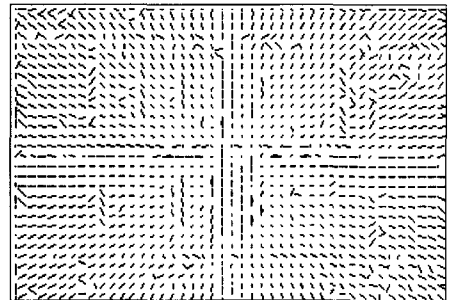


(a) Motion vector

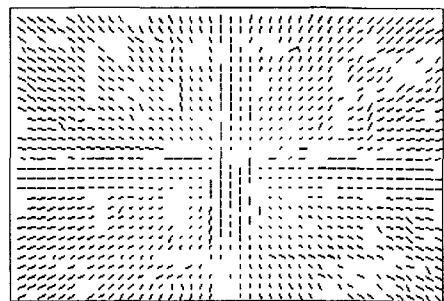


(b) Filtered motion vector

그림 10. 카메라의 틸팅 동작  
Fig. 10. Camera tilt.

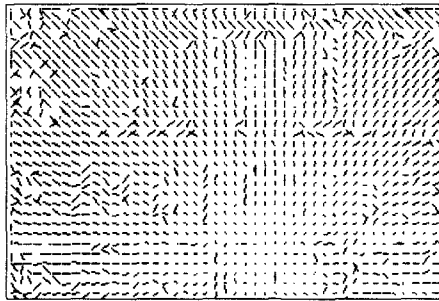


(a) Motion vector

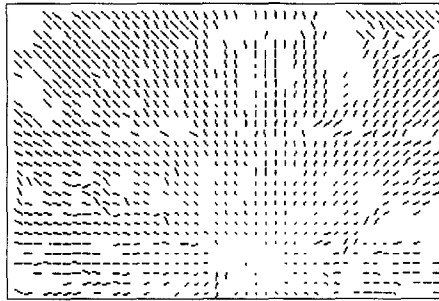


(b) Filtered motion vector

그림 11. 카메라의 줌인 동작  
Fig. 11. Camera zoom.

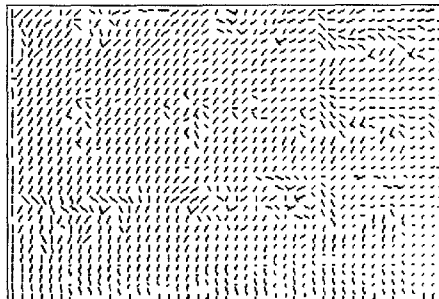


(a) Motion vector

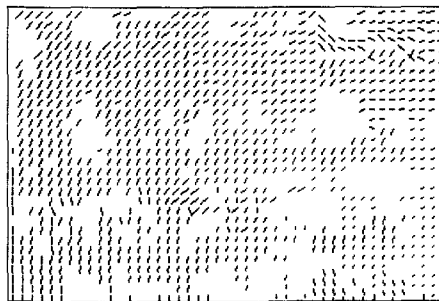


(b) Filtered motion vector

그림 12. 카메라의 패닝, 틸팅, 줌잉이 혼합된 동작  
Fig. 12. Camera pan, tilt, and zoom.



(a) Motion vector



(b) Filtered motion vector

그림 13. 카메라의 회전 동작  
Fig. 13. Camera roll.

표 2는 그림 9, 그림 10, 그림 11, 그림 12, 그림 13에서 보여주는 카메라의 동작으로부터 추출한 동작

벡터를 어파인 모델에 적용하여 어파인 모델을 구성하는 파라미터를 추출한 결과를 보여주고 있다.

표 2. 카메라의 동작에 대한 파라미터 추출 결과

Table 2. Result of parameter extraction.

		$a_1$	$a_2$	$a_3$
그림 9	t+1 $\Delta$ t	-2.0334	-0.0026	-0.0035
	t+2 $\Delta$ t	-1.9268	-0.0027	-0.0041
	t+3 $\Delta$ t	-2.4388	-0.0042	-0.0054
그림 10	t+1 $\Delta$ t	+0.0876	+0.0001	+0.0006
	t+2 $\Delta$ t	+0.1142	-0.0001	-0.0002
	t+3 $\Delta$ t	+0.2939	+0.0015	+0.0013
그림 11	t+1 $\Delta$ t	+1.5210	-0.0278	+0.0184
	t+2 $\Delta$ t	+1.5737	-0.0283	+0.0198
	t+3 $\Delta$ t	+1.4468	-0.0185	+0.0101
그림 12	t+1 $\Delta$ t	-1.4577	+0.0172	-0.0154
	t+2 $\Delta$ t	-1.0720	+0.0174	-0.0131
	t+3 $\Delta$ t	-0.3915	+0.0219	-0.0218
그림 13	t+1 $\Delta$ t	-0.6868	-0.0040	-0.0505
	t+2 $\Delta$ t	-0.5131	-0.0070	-0.0930
	t+3 $\Delta$ t	-0.5845	-0.0045	-0.0204

		$a_4$	$a_5$	$a_6$
그림 9	t+1 $\Delta$ t	-0.0011	+0.0001	-0.0001
	t+2 $\Delta$ t	-0.0068	+0.0001	-0.0001
	t+3 $\Delta$ t	-0.0115	+0.0001	-0.0001
그림 10	t+1 $\Delta$ t	+1.0250	+0.0066	+0.0047
	t+2 $\Delta$ t	+1.0074	+0.0073	+0.0057
	t+3 $\Delta$ t	+1.1999	+0.0047	+0.0130
그림 11	t+1 $\Delta$ t	+1.4395	+0.0106	-0.0303
	t+2 $\Delta$ t	+1.2860	+0.0125	-0.0310
	t+3 $\Delta$ t	+1.1713	+0.0049	-0.0164
그림 12	t+1 $\Delta$ t	-1.2837	-0.0138	+0.0133
	t+2 $\Delta$ t	-0.9694	-0.0082	+0.0142
	t+3 $\Delta$ t	-0.6275	-0.0220	+0.0308
그림 13	t+1 $\Delta$ t	+0.5321	+0.0570	+0.0091
	t+2 $\Delta$ t	+0.8189	+0.0305	+0.0098
	t+3 $\Delta$ t	+0.6037	+0.0260	+0.0071

본 논문에서는 매우 작은 움직임이나 잡음이 포함된 데이터를 배제하고 카메라의 동작을 구하기 위해 표 3과 같은 임계치를 실험적으로 구하여 사용한다.

표 3. 실험적인 임계치

Table 3. Experimental threshold.

파라미터	임 계 치	카메라의 동작
$a_1$	$a_1 < 0.5$	Translation to the Left
	$a_1 < -0.5$	Translation to the Right
$a_4$	$a_4 > 0.5$	Translation Up
	$a_4 < -0.5$	Translation Down
$a_2 + a_6$	$a_2 > 0.01, a_6 > 0.01$	Divergence In
	$a_2 < -0.01, a_6 < -0.01$	Divergence Out
$-a_3 + a_5$	$a_3 < -0.01, a_5 > 0.01$	Curl to the Right
	$a_3 > 0.01, a_5 < -0.01$	Curl to the Left
$a_2 - a_6$	$a_2 > 0.01, a_6 < -0.01$	Deformation In
	$a_2 < -0.01, a_6 > 0.01$	Deformation Out

표 2에 나와있는 개별적인 파라미터들을 표 3과 같은 임계치를 고려하여 본 논문에서 카메라의 동작을 구하기 위해 실질적으로 사용하는 파라미터들을 중심으로 표현하면 표 4와 같다. 표 4에서 +는 양의 큰 값을 나타내고 -는 음의 큰 값, 그리고 0은 매우 작은 값을 나타낸다.

그림 14는 이동 물체가 있는 사무실 장면을 입력 데이터로 받아 들어 실험한 메인 화면을 나타낸다. 그림 14에서 보는 바와 같이 메인 화면은 5개의 윈도우로 구성된다. 좌측 상단 윈도우는 입력 영상이 계속해서 표시되는 윈도우이고, 좌측 중앙 윈도우는 감지 및 추적된 이동 물체를 표시하는 윈도우이며, 좌측 하단 윈도우는 입력 영상으로부터 추출된 동작 벡터가 표시되는 윈도우이다. 그리고 우측 상단 윈도우는 영상의 배경으로부터 추출한 동작 벡터를 분석하여 카메라의 동작을 구한 결과를 표시하는 윈도우이고, 우측 하단 윈도우는 프로그램의 진행 결과를 라인 단위로 표시하는 윈도우이다. 이동 물체가 포함된 입력 영상의 실험에서는 이동 물체의 추적 작업도 수행하므로 시간적인 부하가 크다. 따라서 이동 물체를 감지한 후에는 그림 14의 좌측 하단 윈도우와 같이 공간적인 간격을 유지하면서 동작 벡터를 추출하여 시간적인 부하를 감소시켰다.

그리고 그림 15는 연속적으로 입력되는 영상을 분석해서 동작 벡터를 추출하면서 이동하는 물체를 계속

추적하는 과정을 보여준다. 그림 15에서와 같이 이동 물체를 계속 추적함으로써 이동 물체 영역을 제외한 배경 영역에서만 동작 벡터를 추출하여 카메라의 동작을 구한다.

		$a_1$	$a_4$	$a_2 + a_6$	$-a_3 + a_5$	$a_2 - a_6$
그림 9	t+1 $\Delta$ t	-	0	0	0	0
	t+2 $\Delta$ t	-	0	0	0	0
	t+3 $\Delta$ t	-	0	0	0	0
그림 10	t+1 $\Delta$ t	0	+	0	0	0
	t+2 $\Delta$ t	0	+	0	0	0
	t+3 $\Delta$ t	0	+	0	0	0
그림 11	t+1 $\Delta$ t	+	+	-	0	0
	t+2 $\Delta$ t	+	+	-	0	0
	t+3 $\Delta$ t	+	+	-	0	0
그림 12	t+1 $\Delta$ t	-	-	+	0	0
	t+2 $\Delta$ t	-	-	+	0	0
	t+3 $\Delta$ t	-	-	+	0	0
그림 13	t+1 $\Delta$ t	-	+	0	+	0
	t+2 $\Delta$ t	-	+	0	+	0
	t+3 $\Delta$ t	-	+	0	+	0

표 4. 임계치를 반영한 파라미터

Table 4. Parameter considering threshold.

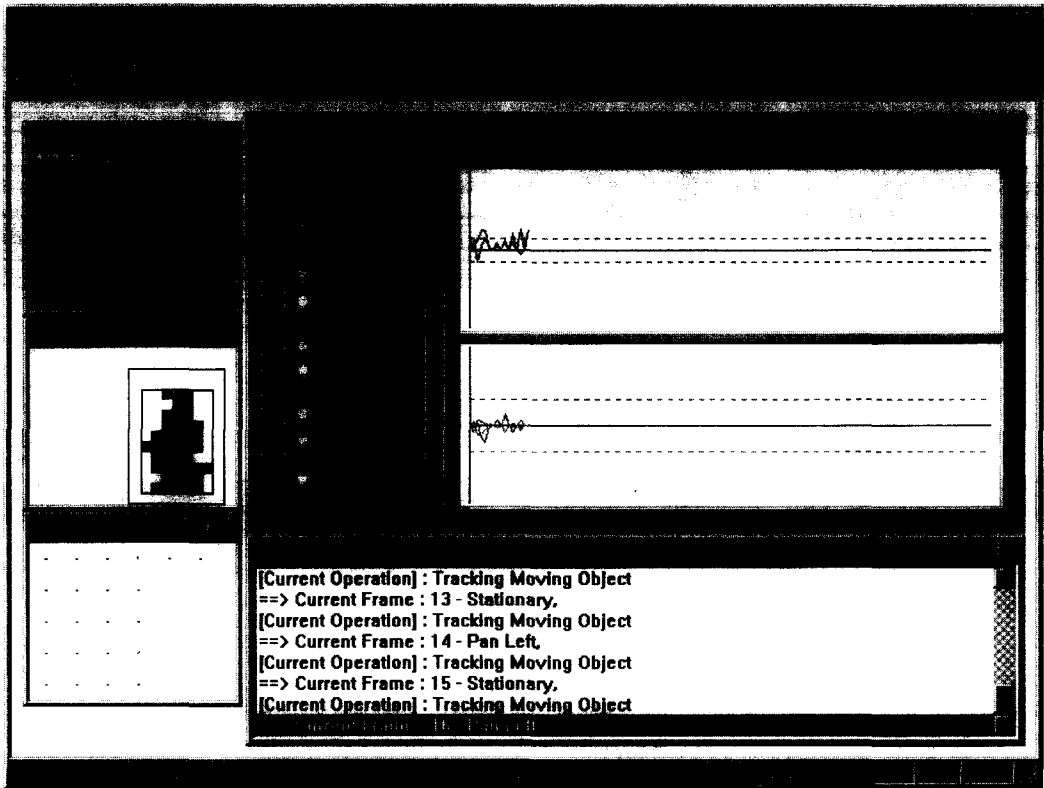
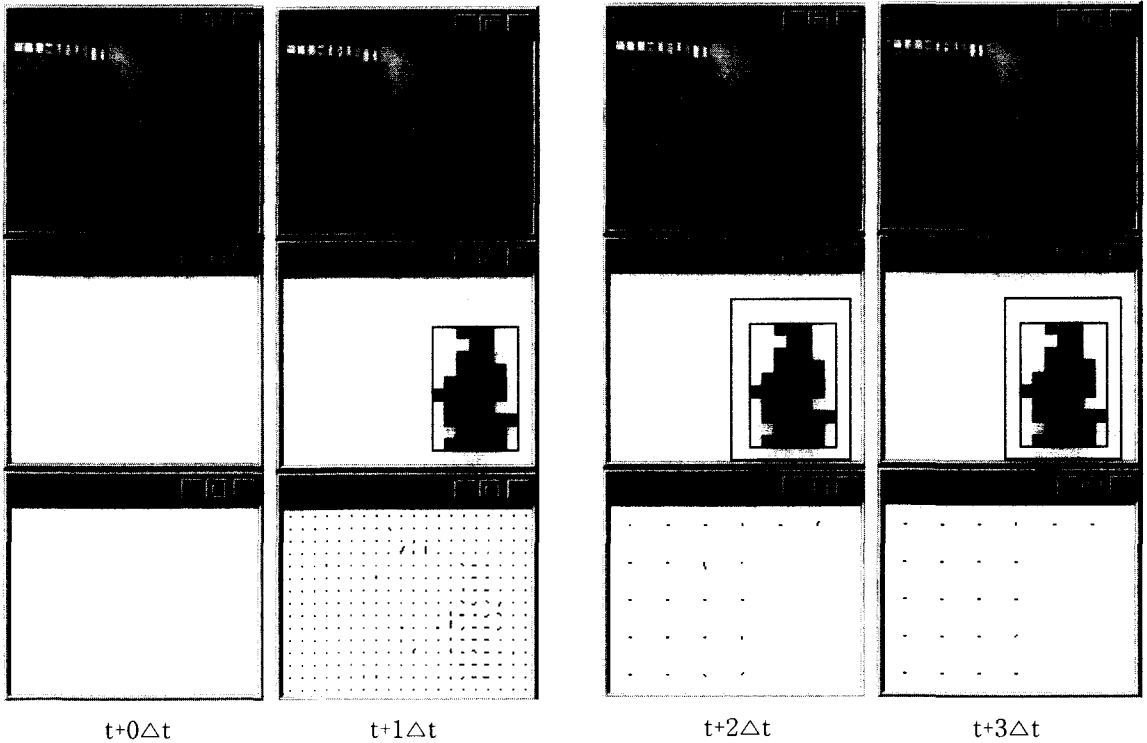


그림 14. 프로그램의 메인 화면

Fig. 14. Main window of program.



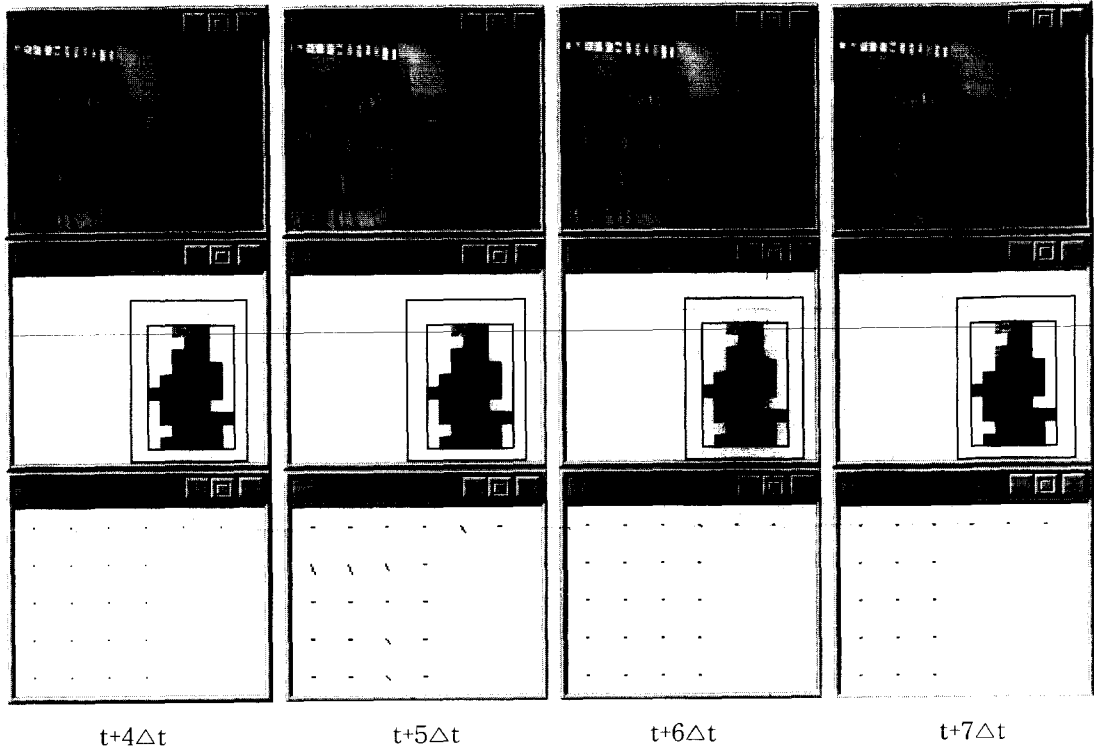


그림 15. 프로그램 진행 과정

Fig. 15. Processing result of program.

## VI. 결론

본 논문에서는 어파인 모델을 이용하여 다양한 카메라의 동작을 정량적으로 추출하는 방법을 제안하였다. 입력 영상으로부터 카메라의 움직임 정보를 획득하기 위해 기본적으로 셀 단위의 동작 벡터를 추출하였고, 영상에서 칼라값이 균일한 영역에서 동작 벡터를 추출할 경우 부정확한 셀 정합으로 인해 필연적으로 발생하게 되는 잡음들을 제거하기 위해 간단하면서도 강건하게 동작 벡터를 필터링하였다. 그리고 카메라의 다양한 동작에 따라 매우 복잡한 구조를 형성하는 동작 벡터를 하나의 모델만을 사용하여 일관성있게 분석하기 위해 어파인 모델을 사용하였다. 통계적인 방법인 회귀분석법을 이용하여 어파인 모델을 구성하는 파라미터를 구하였으며, 파라미터들의 크기 및 상호 관계를 해석하여 카메라의 동작을 확인하였다. 어파인 모델을 구성하는 파라미터들이 카메라의 동작을 표현하므로 여러 가지 카메라의 동작이 혼합된 경우에도 파라미터만을 분석하여 간단하게 개별적인 카메라의 동

작을 추출할 수 있다.

본 논문에서 제안하는 카메라 동작 추출 방법을 개선하기 위해서는 다음과 같은 문제점이 보완되어야 한다. 하나의 특징인 칼라 정보만을 이용하여 동작 벡터를 추출할 경우 균일한 칼라값을 갖는 영역에서 필연적으로 셀 간의 오정합이 발생하여 부정확한 동작 벡터가 추출되므로 이를 개선할 수 있는 방법에 대한 연구가 필요하다. 그리고 본 논문에서 제안하는 방법은 카메라 동작의 정량적인 분석, 즉 입력 영상만을 분석하여 패닝, 틸팅, 줌인 동작 등과 같이 카메라 동작의 종류를 판명하고 카메라 동작의 수량적인 분석, 즉 카메라 동작의 크기는 추출하지 않았다. 따라서 카메라 동작의 종류와 더불어 카메라 동작의 크기에 대한 추출이 요구된다. 이를 위해서는 퍼지 추론 등의 추론 방법을 이용해서 카메라의 동작을 수량화할 수 있을 것으로 기대되며, 카메라 동작의 수량화에 대한 연구를 계속 진행할 것이다.

※ 본 논문은 정보통신연구진흥원의 산·학·연 공동 기술개발사업의 지원과 첨단정보기술연구센터를 통하여 과학재단의 일부 지원을 받았다.

### 참 고 문 헌

- [ 1 ] Rainer Lienhart, "Automatic Text Recognition for Video Parsing", Proceedings ACM Multimedia 96, pp. 11-20, 1996.
- [ 2 ] Yi Wu and David Suter, "A Comparison of Methods for Scene Change Detection in Noisy Image Sequence", First International Conference on Visual Information Systems, pp. 459-468, 1996.
- [ 3 ] Omer N. Gerek and Yucel Altunbasak, "Key Frame Selection from MPEG Video Data", Visual Communications and Image Processing 97, SPIE Vol. 3024, pp. 920-925, 1997.
- [ 4 ] Jonathan D. Courtney, "Automatic Video Indexing via Object Motion Analysis", Pattern Recognition, Vol 30, No. 4, pp. 607-625, 1997.
- [ 5 ] A. Akutus, Y. Tonomura, H. Hashimoto and Y. Ohba, "Video Indexing Using Motion Vectors", Proc. of the SPIE - Visual Communications and Image Processing, Vol. 1818, pp. 1522-1530, 1992.
- [ 6 ] A. Akutsu and Y. Tonomura, "Video Tomography : An Efficient Method for Camerawork Extraction and Motion Analysis", ACM Multimedia 94, pp. 349-356, 1994.
- [ 7 ] F. Idris and S. Panchanathan, "Detection of camera operations in compressed video sequences", Proc. of the SPIE Storage and Retrieval for Image and Video Database V, Vol. 3022, pp. 493-505, 1997.
- [ 8 ] Frank Perez and Christof Koch, "Toward Color Image Segmentation in Analog VLSI : Algorithm and Hardware", International Journal of Computer Vision, Vol. 12, No. 1, pp. 17-42, 1994.
- [ 9 ] Ying Dai and Yasuaki Nakano, "Extraction of Facial Images from Complex Background Using Color Information and SGLD Matrices", International Workshop on Automatic Face and Gesture Recognition, Zurich, pp. 238-242, 1995.
- [ 10 ] Yu-Ich Ohta, Takep Kanade and Toshiyuki Sakai, "Color Information for Region Segmentation", Computer Vision, Graphics, and Image Processing, Vol. 13, pp. 222-241, 1980.
- [ 11 ] Christopher C. Yang and Jeffrey J. Rodriguez, "Efficient Luminance and Saturation Processing Techniques for Bypassing Color Coordinate Transformations", IEEE Systems, Man and Cybernetics, pp. 667-672, 1995.
- [ 12 ] Hironobu Takahashi and Fumiaki Tomita, "Fast Region Labeling with Boundary Tracking", IEEE ICIP'89, Singapore, pp. 369-373, 1989.
- [ 13 ] Dae-Sik Jang, Gye-Young Kim, and Hyung-Il Choi, "Model-based Tracking of Moving Object", Pattern Recognition, Vol. 30, No. 6, pp. 999-1008, 1997.
- [ 14 ] 장석우, 문철호, 최형일, "전경과 배경을 동시에 고려하는 이동 물체 추적", 한국 퍼지 및 지능 시스템 학회 '98 추계 학술대회 학술발표 논문집, pp. 511-515, 1998
- [ 15 ] 김동근, 황치정, "활성 모델에 의한 경계선 분할 및 모델 적합", 한국정보과학회 논문지(B), Vol. 23, No 2, February, 1996
- [ 16 ] Faires Burden, Numerical Methods, PWS publishing company, 1993.

저 자 소 개



張錫宇(正會員)

1995년 2월 숭실대학교 전자계산학과 졸업(공학사). 1997년 2월 숭실대학교 대학원 전자계산학과 졸업(공학석사). 1997년 3월 ~ 현재 숭실대학교 대학원 컴퓨터학과 박사과정 재학중. 주관심분야는 컴퓨터 비

전, 영상처리, 동작이해, 비디오 검색 등임



崔滢一(正會員)

1979년 2월 연세대학교 전자공학과 졸업(공학사). 1982년 6월 미시간대학교 전산공학과 졸업(공학석사). 1987년 6월 미시간대학교 전산공학과 졸업(공학박사). 1988년 ~ 현재 숭실대학교 컴퓨터학부 교수. 주관

심분야는 컴퓨터 비전, 퍼지 및 신경망이론, 패턴인식, 지식기반시스템 등임



李根壽(正會員)

1983년 2월 숭실대학교 전자계산학과 졸업(공학사). 1988년 2월 숭실대학교 대학원 전자계산학과 졸업(공학석사). 1993년 8월 숭실대학교 대학원 전자계산학과 박사과정 졸업(공학박사). 1989년 2월 ~ 현재 한

경대학교 전자계산학과 부교수. 주관심분야는 패턴인식, 컴퓨터 비전, 동작이해, 비디오 검색 등임