

회전 및 병진운동 추정을 통한 디지털 영상안정화

Digital Image Stabilization Using Simple Estimation of Rotational and Translational Motion

석호동**, 강길순*, 유준***
(Ho-Dong Seok, Kil-Soon Kang, and Joon Lyoo)

Abstract - This paper presents a simple method of rotational and translational motion estimation for digital image stabilization. The scheme first computes the rotation center by taking least squares of selected local velocity vectors, and the rotational angle is found from special subset of motion vectors. And then translational motion can be estimated by the relation among movement of rotation center, rotation angle and translation movement. To show the effectiveness of our approach, the synthetic images are evaluated, resulting in better performance.

Key Words : Digital image stabilization, Motion estimation, Block matching algorithm,

1. 서론

기동 플랫폼에 탑재된 영상센서를 활용하여 독립적으로 움직이는 물체를 탐지, 인식, 추적하고자하는 분야에서 동적 영상해석 및 영상전시, 가시화 등을 위해서 필수적으로 선행되어야 될 사전 준비작업이 영상안정화이다. 또한, 무인차량 시스템을 원격 제어할 경우, 원격 조종자에게 흔들림이 제거된 안정된 영상을 제공하는 것은 필수적인 요소이며, 이동 차량과 독립적으로 움직이는 표적 추적 등을 위해서 영상센서 출력의 모션을 제거하게 되면 배경화면이 고정되어 있는 정지 영상과 같게 되어 표적추적이 간단히 처리될 수도 있다.

기존에 사용된 안정화 방식으로서는 관성센서와 구동메카니즘을 결합한 기구적인 안정화 방식이 주류였으나 최근에 기구적 안정화 장치에 비해 가격이 저렴하고 형상을 최소화 할 수 있는 디지털 영상안정화 장치가 제안되고 있다. 디지털 영상안정화는 순수한 영상신호처리에 의한 움직임량의 추정을 통하여 영상 재현영역을 재구성함으로써 영상을 안정화 하는 기법이다.

이 분야의 기존연구들은 카메라의 병진운동만 다루거나^[1], 회전과 병진운동을 동시에 고려하더라도 움직임 추정에 수평선과 같은 특징점을 필요로 하거나^[2] 계산량이 많아 충분한 대역폭을 확보하기 어려운 등 제한사항을 가지고 있다.^[3]

본 연구는 디지털 영상안정화에 적용할 수 있는 병진과 회전움직임을 빠르고 간단하게 추정하는 기법을 제안하였다. 제안된 알고리즘은 먼저 영상 각 화소의 지역 속도 벡터로부터 속도벡터의 회전중심을 찾고 회전 중심점 정보를 이용한 움직임 벡터들의 부분집합으로부터 회전각도를 추정하고 회전 중심의 움직임량, 회전각도와 병진운동량의 관계로부터 병진운동량을 추정하였다.

제안된 알고리즘을 합성영상에 적용하여 시험한 결과 모션변수에 대하여 추정정능이 우수하였다.

2. 회전 및 병진 움직임 추정

2.1 영상화소의 국부적 움직임 계산

영상의 움직임 크기는 영상 각화소의 국부적인 움직임량을 기반으로 구할 수 있는데, 국부적인 움직임 추정을 위해서 본 연구에서는 블록 정합법을 사용하였다. 블록 정합법은 적용방법이 용이하고 실제 시스템을 구현할 때 하드웨어가 비교적 간단하기 때문에 움직임 추정에 가장 많이 사용하는 방법 중의 하나이다. 정합 기준으로는 최소 평균 자승 오차법 (minimum mean square error : MSE), 최소 평균 절대차법 (minimum mean absolute difference : MAD), 최대 정합 화소수법(maximum matching pixel count : MPC) 등이 있으나 본 논문에서는 가장 보편적으로 사용하는 MAD를 적용하였다. 지역 움직임을 추정 할 때 MAD를 이용한 오차는 다음 식과 같이 정의 된다^[4].

$$e = \frac{1}{N_1 N_2} \sum_{k=1, \dots, n} |s(x, y, k) - s(x + d_1, y + d_2, k + 1)| \quad (1)$$

움직임 추정은 다음 식에 의해 구해진다.

$$[d_1, d_2]^T = \arg \min_{(d_1, d_2)} MAD(d_1, d_2) \quad (2)$$

가장 정합이 잘 이루어지는 위치를 찾기 위해서는 각 화소에 대해 움직임 벡터가 될 수 있는 모든 점에 대해 정합기준을 최적화하는 과정이 필요하다. 이는 각 화소에 대해 영상의 모든 점에 정합과정을 수행 하여야 하나 이는 매우 많은 시간을 필요로 한다. 따라서 계산량을 줄이기 위해 영상의 동특성을 고려하여 탐색창의 크기와 영역을 제한하였다.

블록정합의 결과는 점 (x, y) 에서 각 블록의 움직임 벡터 $V(x, y) = (u(x, y), v(x, y))$ 가 된다. 일반적으로 실영상에서 구한 움직임 벡터는 영상에 포함된 잡음의 영향으로 훼손될 가능성이 높다. 따라서 잡음에 의한 영향을 줄이기 위해 지역 움직임 벡터들에 대해 메디안 필터를 적용함으로써 잡음을 제거하였다.

2.2 영상의 움직임 추정

먼저 순수한 회전 움직임만 있는 경우 지역 모션벡터는 그림 1과 같이 회전중심을 기준으로 동심원을 그리게 된다.

저자 소개

* 學生會員 : 忠南大學 電子工學科 碩士課程

** 準會員 : 忠南大學 電子工學科 博士課程

***正會員 : 忠南大學 電子工學科 教授 · 工博

본 연구는 한국과학재단 목적기초연구(R01-2003-000-10857-0) 지원으로 수행되었음.

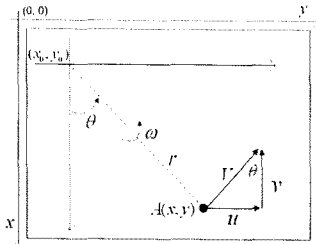


그림 1. 지역모션벡터 특성

임의의 점 $A(x, y)$ 에서 모션벡터는 다음 식과 같이 표현된다.

$$u(x, y) = w(y - y_0), \quad v(x, y) = w(x - x_0) \quad (2)$$

여기서 u, v 는 각각 x 축, y 축 방향의 모션벡터 성분이다. 그리고 (x_0, y_0) 는 회전 중심점, w 는 회전 각속도이다.

회전운동과 병진운동이 동시에 발생하게 되면 지역 모션벡터의 동심원의 중심은 병진운동량과 회전운동량크기에 따라 이동하게 되는데 수식으로 표현하면 다음과 같다.

$$\Delta x = w(y - y_0), \quad \text{즉 } y = y_0 + \frac{\Delta x}{w}$$

$$\text{또한, } \Delta y = w(x - x_0), \quad \text{즉 } x = x_0 + \frac{\Delta y}{w} \quad (3)$$

수식에서 보듯이와 같이 지역 모션벡터의 동심원의 중심은 $\frac{\Delta y}{w}$, $\frac{\Delta x}{w}$ 만큼 x 축, y 축으로 이동하게 된다. 따라서 연속된 두 영상의 지역모션벡터로부터 모션벡터의 회전중심과 각도를 추정하면 영상간의 회전각도, 병진운동량을 추정할 수 있다.

2.2.1 지역모션벡터의 회전중심 추정

영상의 모든 점에서 움직임 벡터는 동심원의 접선 방향과 일치하게 된다. 따라서 임의의 모든 점에서 움직임 벡터에 대한 법선들은 영상의 회전 중심에서 만나게 된다. 그림 2는 임의의 두 점에 대한 경우를 설명하고 있다.

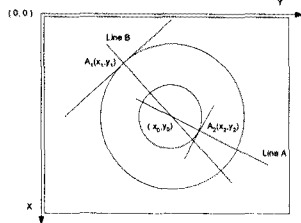


그림 2. 회전중심 추정

점 $A_1 = (x_1, y_1)$, $A_2 = (x_2, y_2)$ 에서 움직임 벡터에 대한 법선 방정식은 식 (4-5)과 같이 표현할 수 있다.

점 A_1 의 경우,

$$y = a_1 x + b_1 \quad (4)$$

여기서 $a_1 = \frac{u_1}{v_1}$, $b_1 = y_1 - a_1 x_1$ 이며, u_1, v_1 는 점 A_1 에서

속도 벡터이다. 점 A_2 의 경우는

$$y = a_2 x + b_2 \quad (5)$$

여기서 $a_2 = \frac{u_2}{v_2}$, $b_2 = y_2 - a_2 x_2$ 이며, u_2, v_2 는 점 A_2 에서 속도 벡터이다.

두 화소에 대한 움직임 벡터만 구하면 회전중심을 계산할 수 있으나 영상에 잡음이 포함되거나 프레임간 충분한 대조비(contrast)가 확보되지 않는 경우에는 오차를 유발하게 된다. 따라서 실제 계산시는 잡음과 대조비의 변화 등에 대한 강인성을 고려하여 3 화소 이상에 대한 움직임 추정결과를 활용한다.

n 개의 화소점의 움직임 벡터를 활용해서 회전중심 (x_0, y_0) 을 찾는 경우는 식 (6)과 같이 표현할 수 있다.

$$\begin{bmatrix} -a_1 & -a_2 & \dots & -a_n \\ 1 & 1 & \dots & 1 \end{bmatrix}^T [x_0, y_0]^T = [b_1, b_2, \dots, b_n]^T \quad (6)$$

식 (6)을 행렬과 벡터 형태로 표현하면 다음과 같다.

$$Ax = b \quad (7)$$

이 경우 회전중심은 최소 자승법을 적용하여 식 (8)로부터 구할 수 있다.

$$x = (A^T A)^{-1} A^T b \quad (8)$$

2.2.2 지역모션벡터의 회전각도 추정

계산된 회전중심점을 이용하면 회전각도 θ 를 다음과 같이 구할 수 있다.

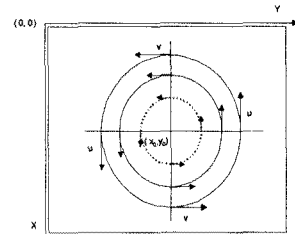


그림 3. 회전각도 추정

그림 2에서 보는 바와 같이 회전중심에서 y 축에 대한 수직선에 위치하는 모든 점은 y 축 방향 움직임 벡터만 갖고 있다. 또한, 회전중심에서 y 축에 대한 수평선에 위치하는 모든 점은 x 축 방향 움직임 벡터만 갖고 있다. 따라서 회전각도는 다음 식으로부터 구할 수 있다.

수평선상의 모든 점에 대해서는 $\theta = \tan^{-1}(du/dx)$ 이며 수직선상의 모든 점에 대해서는 $\theta = \tan^{-1}(dv/dy)$ 이다.

여기서 dx, dy 는 회전중심과 각 점간의 거리이며 du, dv 는 수평선과 수직선상 각 점에서의 움직임 벡터 값이다. 매 프레임당 카메라의 움직임 각도는 매우 작으므로 회전각도는 $\theta = du/dx$, $\theta = dv/dy$ 로 근사화 할 수 있다.

회전중심 추정의 경우와 마찬가지로 회전각도 추정에도 최소 자승법을 적용하였다. 회전각도 추정을 위해 n 개의 점에 대해 식을 정리하면 다음과 같다.

$$\begin{bmatrix} dv_1 & dv_2 & \dots & dv_N & du_1 & du_2 & \dots & du_N \end{bmatrix}^T = \begin{bmatrix} dx_1 & dx_2 & \dots & dx_N & du_1 & du_2 & \dots & du_N \end{bmatrix}^T \theta \quad (9)$$

여기서 dx_1, dx_2, \dots, dx_N 는 회전중심과 수평선상의 점들과의 거리이며, dy_1, dy_2, \dots, dy_N 는 회전중심과 수직선상의 점들과의 거리이다. dv_1, dv_2, \dots, dv_N 는 수평선상의 각 점에

서 움직임 벡터의 크기이며, du_1, du_2, \dots, du_N 는 수직선상의 각 점에서 움직임 벡터의 크기이다.

식 (9)를 행렬과 벡터형태로 표현하면 다음과 같다.

$$b = A\theta \quad (10)$$

따라서 회전각도는 식(11)로부터 구할 수 있다.

$$\theta = (A^T A)^{-1} A^T b \quad (11)$$

3. 실험

제안된 알고리즘을 검증하기 위해 그림 3과 같은 표준영상을 알고 있는 값으로 회전과 병진을 시킨 영상을 생성하여 합성영상을 만들고, 제안된 알고리즘을 적용하여 모션을 추정하여 실제값과 비교하였다. 실험은 연속된 2개의 영상에 대한 것과 40프레임의 연속된 영상에 대해 실험하였다.

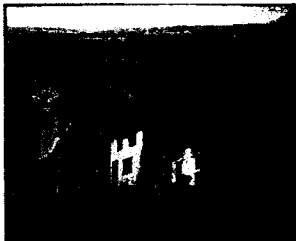


그림 4. 실험에 사용된 표준영상

2개의 연속된 합성영상에 대한 실험결과는 표1과 같다. 표 1의 내용에서 보는 바와 같이 제안된 알고리즘은 회전운동과 병진운동을 잘 추정하였다.

표 1 두개의 연속영상에 대한 실험결과

회전각도(도)		병진운동 X(pixel)		병진운동 Y(pixel)	
실제값	추정값	실제값	추정값	실제값	추정값
1	0.9877	2	1.8964	2	1.9481
1	0.9654	2	1.8199	3	2.8815
2	1.9986	2	1.7099	2	2.1635
2	1.8653	5	4.0384	5	4.7549
2	1.9265	4	3.5654	2	2.1527

40프레임의 연속된 영상은 표준영상을 회전각도변화와 x 축 및 y축의 병진을 다음 식을 적용하여 영상을 생성하였다.

$$\theta = 0.4 \sin(2\pi 0.75 k T) \quad (12)$$

$$x_{move} = 20 \sin(2\pi 0.75 k T) \quad (13)$$

$$y_{move} = 20 \sin(2\pi 0.75 k T) \quad (14)$$

여기서 θ 는 회전각도(radian), x_{move} 는 x축방향으로 이동한 pixel수 y_{move} 는 y축방향으로 이동한 pixel수, k 는 프레임 number, T 는 주기이며 실험에서 1/60을 적용하였다. 병진운동은 pixel 단위로 라운드 처리하였다.

제안된 알고리즘을 적용하여 추정한 결과와 실제값과 비교한 결과는 그림 4와 같다. 제안된 알고리즘은 실제 회전각도와 병진값을 잘 추정하였다. 20 프레임 근처에서 추정오차가 다소 큰 것은 병진 및 회전운동량이 매우 작은 결과에 기인하는 것으로 판단된다.

4. 결론

본 연구에서는 회전 및 병진 운동을 하는 영상에 대하여 안정화 알고리즘을 제안하였고 성능평가를 통하여 제안방식의 성능이 우수함을 보였다.

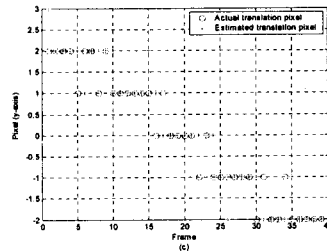
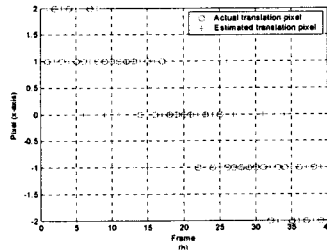
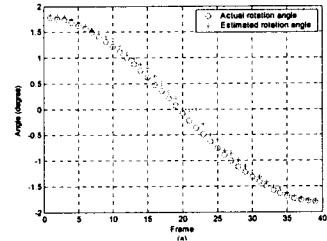


그림 3. 40프레임 연속영상 실험결과 (a) 회전각도 (b)x축 병진 (c)y축 병진

제안된 알고리즘은 먼저 영상 각 화소의 지역 속도 벡터로부터 속도벡터의 회전중심을 찾고 회전 중심점 정보를 이용한 움직임 벡터들의 부분집합으로부터 회전각도를 추정하고 회전 중심의 움직임량, 회전각도와 병진운동량의 관계로부터 병진운동량을 추정하였다.

제안된 알고리즘을 합성영상에 적용하여 시험한 결과 모션 변수에 대하여 추정성능이 우수하였다.

참 고 문 헌

- [1] S. J. Ko, S. H. Lee, S. W. Jeon, and E. S. Kang, "Fast digital image stabilizer based on gray-coded bit-plane matching," IEEE Transactions on Consumer Electronics Vol. 45, pp. 598-603, 1999.
- [2] C. Morimoto, and R. Chellappa, "Fast electronic digital image stabilization for off-road navigation," Real-Time Imaging, Vol. 2, pp. 285-296, 1996.
- [3] J. Y. Chang, W. F. Hu, M. H. Cheng and G. S. Chang, "Digital image translational and rotational motion stabilization using optical flow technique." IEEE Transactions on Consumer Electronics, Vol. 48, No. 1, pp.108-115, 2002.
- [4] A. Murat Tekalp, Digital Video Processing, Prentice Hall, 1995.