

영상 기반 롤 안정화 장치 구현 및 실험

The realization of Image Based Roll Stabilization and Experiments

석호동, 곽휘권, 조재현, 유 준

(Ho-Dong Seok, HwY-Kuen Kwak, Jae-Hyun Cho, Joon Lyou)

Abstract - This paper presents realization and evaluation of digital image stabilization for roll motion. The proposed image stabilizer has been implemented on the DSP based hardware and DSP/BIOS software. The proposed image stabilizer is tested in the field. The results show good performance with the sufficient bandwidth. The proposed image stabilizer is combined with mechanical stabilizer and tested to evaluate whether that is feasible for 3-axis stabilization of tank sight system. The results also show good performance.

Key Words : Image stabilization, Motion estimation, Block matching algorithm, Peak signal to noise ratio

1. 서론

전차와 같이 기동차량에 탑재된 조준경은 차량이 정지한 상태 또는 주행 상태에서 입력되는 모든 회전성분의 외란을 제거하는 정밀한 안정화 성능이 요구된다.

기존의 조준경 장치는 엔진, 냉각fan, 불규칙한 노면 등으로 인해 입력되는 외부 외란을 제거하기 위해 관성센서를 이용하여 기구적으로 피치와 요 방향으로만 안정화를 하고 있다. 2축 안정화 장치만을 적용하고 있는 이유는 단순히 기구적으로 3축 안정화 장치를 구성할 경우 중량 및 부피가 커지고 가격이 상승이 주요 요인중의 하나이다. 그러나, 최근의 사격통제장치는 원거리의 표적획득 능력 및 명중률을 높이기 위해 피치, 요, 롤 3축의 매우 정밀한 안정화 장치를 요구하고 있다. 최근 중량, 부피 및 가격측면에서 유리한 영상 기반 롤 안정화 기법이 제안되었으나[1] 전산 모사에 의한 실험만(off-line simulation)으로 제안된 기법의 실 시스템 적용가능성을 보였다. 본 연구에서는 DSP(Digital Signal Process)를 적용하여 실시간 작동 영상기반 롤 안정화 장치를 구현하였다. 영상의 움직임 추정에 처리속도 향상을 위한 기법이 고안되었으며 움직임보정 부에서는 영상이 지속적으로 흔들리더라도 설정한 초기 영상으로 안정화되는 기준영상과 입력영상간 안정화를 구현하였다. 제안된 안정화장치는 야외실험을 통해 실영상에 대한 작동성을 검증하였으며 조준경의 3축안정화장치에 적용가능성을 보이기 위해 기구적 안정화장치와 결합한 실험을 수행하였다. 실험결과 16Hz의 대역폭을 가지면서 안정화성능이 우수하였다.

2. 영상기반 롤 안정화 장치 구현

디지털 영상안정화는 움직임 추정과 움직임 보상으로 나누어지며 움직임 추정은 카메라에서 획득된 연속된 영상에서 영상의 움직임을 추정하는 것이고, 움직임 보상은 움직임 추정결과에 의해 영상변화를 제거하는 과정이다.

2.1 하드웨어 구성

구현된 영상안정화장치의 구성은 그림 1과 같다. 원시영상을 획득하는 CCD 카메라와 회전 및 병진 흔들림을 발생하는 PAN/TILT장치 및 슬라이딩 패드로 구성된 흔들림 영상입력부, 영상의 움직임 추정 및 보상알고리즘을 수행하는 DSP 보드로 구성된 영상신호처리 및 안정화부, 입력된 흔들림 영상과 안정화된 영상을 전시하는 전시부, 프로그램 작성 및 디버깅등을 수행할 수 있는 JTAG 에뮬레이터 보드를 장착한 팬텀 4 PC로 구성된 컨트롤부가 주요 장치이다. 영상안정화 알고리즘이 실제 구현되는 메인 프로세서인 DSP보드는 TI(Texas Instrument)사의 DM642 EVM 보드를 사용하였으며 주요 사양은 표 5.1과 같다.

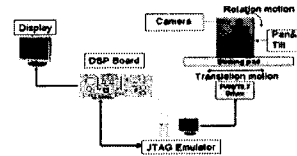


그림 1. 영상안정화 장치 하드웨어 구성

2.2 알고리즘 구현

제안된 영상안정화 알고리즘은 C6000 Code Composer Studio를 이용한 C 언어로 프로그램 하여 구현하였다. 스케일러블한 실시간 커널로서 실시간 스케줄링과 동기화, 호스트와 타겟보드 간 통신 등을 필요로 하며 선점형 멀티쓰레딩, 하드웨어 추상화, 실시간 분석 및 설정도구를 제공하는 DSP/BIOS의 TSK 모듈을 이용하였다. 영상안정화 프로그램 흐름도는 그림 2와 같다.

저자 소개

- * 석호동 : 忠南大學 電子工學科 博士課程
 - ** 곽휘권 : 忠南大學 電子工學科 碩士課程
 - *** 조재현 : 忠南大學 電子工學科 碩士課程
 - ****유 준 : 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 2 영상안정화 흐름도

2.2.1 DSP 초기화 및 영상 획득

프로그램을 시작하기 위해서는 DSP/BIOS의 Startup Sequence를 구현해야하며 main() 함수를 호출한 후 종료하면서 BIOS_start() 함수를 호출하여 DSP/BIOS를 시작시키고 idle 상태에 들어간다. 이때부터 DSP/BIOS에 의해 스케줄링된 작업들이 우선순위에 따라 수행되므로 영상 안정화 알고리즘의 메인 루틴이 무한루프를 돌면서 실행된다. 입력 영상의 최대 크기는 720×480까지 선택할 수 있다. 본 논문에서는 720×480 크기의 전체 영상에서 중앙 부분 256×256 크기의 영상을 잘라서 사용하였다.

2.2.2 영상의 지역 움직임 벡터 추정

영상의 지역 움직임 벡터는 블록 정합법을 사용하여 식 (1), (2)에 의해 추정된다.

$$MAD(d_1, d_2) = \frac{1}{N_1 N_2} \sum_{(x,y) \in B} |s(x,y,k) - s(x+d_1, y+d_2, k+1)| \quad (1)$$

여기서 B 는 $N_1 \times N_2$ 크기의 정합 블록을 의미하며 $s(x,y,k)$ 는 k 차 영상프레임에서 (x,y) 지점 화소의 영상 밝기이며 $s(x+d_1, y+d_2, k+1)$ 은 $k+1$ 차 영상프레임에서 $(x+d_1, y+d_2)$ 지점 화소의 영상 밝기이다.

$$[\hat{d}_1, \hat{d}_2]^T = \arg \min_{(d_1, d_2)} MAD(d_1, d_2) \quad (2)$$

여기서 \hat{d}_1, \hat{d}_2 은 화소의 움직임 벡터 추정 값이다.

프로그램 구현은 DSP 보드와 함께 제공되는 이미지 처리용 라이브러리에 포함된 MAD 계산 함수를 사용하였다. 라이브러리에서 제공되는 블록 크기는 8x8과 16x16 두 가지이며 함수 자체가 어셈블리어로 되어 작성되었고 파이프라인을 이용한 병렬처리를 사용하기 때문에 속도 면에서 수행시간을 크게 단축할 수 있다. 동일한 방법으로 직접 라이브러리를 생성할 경우 다양한 블록크기를 사용할 수 있다.

2.2.3 대표영역 설정

계산된 모든 지역 모션 벡터를 사용하여 회전 중심과 회전 각도를 추정하면 정확도 면에서는 유리할 수 있으나 계산량이 많아진다. 또한 선택된 대표점 근처의 지역 움직임 벡터만을 계산하면 되므로 계산 시간이 단축할 수 있다. 본 논문에서는 그림 3과 같은 방법으로 대표점을 선택하였다. 이외에도 다양한 방법으로 대표영역을 설정할 수 있다. 영상의 외부 경계 영역은 흔들림 발생시 영상을 벗어날 수 있는 영역이므로 대표점 선택에서 제외하였으며 계산량을 줄이면서 전체 영상을 대표할 수 있는 영역선택 방법 중 하나를 선정하였다.

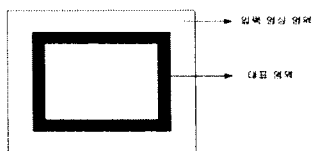


그림 3 영상의 전역움직임 추정을 위한 대표영역 설정

2.2.4 영상의 전역움직임 추정

영상의 전역움직임 추정 즉 회전중심과 회전각도는 그림 4와 5와 같이 지역움직임 벡터의 기하학적 특성을 이용하여 식 (3), (4)에 의해 계산된다[1].

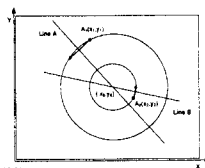


그림 4. 회전중심 추정

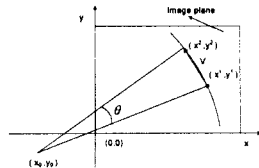


그림 5. 회전각도 추정

$$x = (A^T A)^{-1} A^T b \quad (3)$$

$$\theta = \tan^{-1} \frac{(y_2 - y_0)(x_1 - x_0) - (x_2 - x_0)(y_1 - y_0)}{(x_2 - x_0)(x_1 - x_0) + (y_2 - y_0)(y_1 - y_0)} \quad (4)$$

여기서 $x = (x_0, y_0)$ 는 회전중심 추정결과이며 θ 는 회전 각도 추정결과이다. 회전중심 추정은 최소자승법을 적용함으로 행렬의 계산을 필요로 한다. MATLAB과 달리 DSP 프로그램에서는 행렬계산을 직접할 수 없으므로 어레이를 사용한 계산을 수행하였다. 본 영상안정화 장치에 사용한 DSP보드는 부동소수점을 처리해주는 연산기가 별도 장착되어 있지 않아 부동소수점 연산시에 계산시간이 과도하게 소요된다. 따라서 연산시간 단축을 위해 높은 정밀도가 필요한 경우를 제외하고는 고정소수점 연산방식으로 변환하여 적용하였다. 고정소수점 연산을 위해서 정수부분과 소수부분을 분리하여 정수로 변환하는 Q포맷을 사용하였다.

2.2.5 움직임 보상 및 영상 출력

움직임 보상은 안정화 결과를 가시적으로 볼 수 있는 입력 영상 프레임간 안정화를 구현하였다. 매 입력 영상프레임에 대해 영상의 전역 움직임 추정결과를 식 (5)에 의해 누적 계산하여 매 영상프레임마다 기준영상으로 움직임 보상을 결과를 출력하였다.

$$\begin{bmatrix} x^1 \\ y^1 \end{bmatrix} = R_1 R_2 \cdots R_{k-2} R_{k-1} \begin{bmatrix} x^k \\ y^k \end{bmatrix} + R_1 R_2 \cdots R_{k-2} \begin{bmatrix} K_{k-1} \\ L_{k-1} \end{bmatrix} \\ + R_1 R_2 \cdots R_{k-3} \begin{bmatrix} K_{k-2} \\ L_{k-2} \end{bmatrix} + \cdots + R_1 \begin{bmatrix} K_2 \\ L_2 \end{bmatrix} + \begin{bmatrix} K_1 \\ L_1 \end{bmatrix} \quad (5)$$

여기서 x^k, y^k 은 k 번째 영상 프레임의 좌표이며,

$R_k = \begin{bmatrix} \cos \theta_k & -\sin \theta_k \\ \sin \theta_k & \cos \theta_k \end{bmatrix}$ 는 $k-1$ 번째 영상과 k 번째 영상간의 회전각도 추정결과이다. $R_k^{k-2} = R_k R_{k-1} \cdots R_{k-2}$ 와 식 (5)에서 우측 3번째 이하 항은 $k-1$ 번째 영상입력시 이미 계산된 값이다.

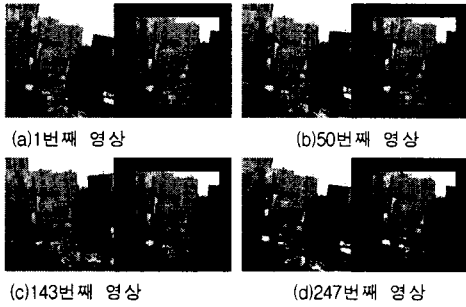
영상의 출력은 RGB모니터를 출력장치로 사용하였기 때문에 그레이 영상을 RGB포맷으로 재구성하였다.

3. 실험

실험은 영상안정화 장치만에 의한 실험과 조준경의 3축안정화장치에 적용가능성을 보이기 위해 기구적 안정화장치와 결합한 실험을 수행하였다.

3.1 영상안정화 장치만의 실험

구현된 안정화 장치를 적용하여 야외 영상에 대해 실험하였다. 팬/틸터 장치위에 장착된 카메라에 회전 흔들림을 발생시켜 영상안정화 장치의 성능을 확인하였다. 화면에 전시되는 입력영상과 안정화결과 영상은 그림 6과 같다.



(a)1번째 영상 (b)50번째 영상
(c)143번째 영상 (d)247번째 영상

그림 6. 영상안정화 장치만의 실험결과 영상

또한 안정화 성능을 수치적으로 확인하기 위해 식 (6)에 정의된 PSNR을 계산하였으며 영상안정화를 하지 않은 경우인 LB(Lower Bound)값과 비교한 결과는 그림 7과 같다.

$$PSNR(I_1, I_0) = 10 \log \frac{255^2}{MSE(I_1, I_0)} \quad (6)$$

여기서, MSE는 평균자승오차이며 안정화된 영상간의 PSNR이 크다는 것은 안정화 성능이 우수하다는 것을 의미한다.

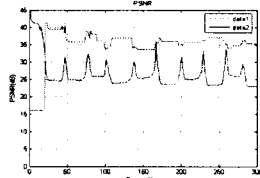


그림 7. 영상안정화 장치만의 실험결과(PSNR)

그래프에서 처음 20프레임까지 안정화된 영상의 PSNR이 낮은 걸 확인할 수 있는데 이 부분은 영상 안정화 알고리즘을 작동시키지 않은 부분이다. 안정화된 영상의 평균 PSNR은 34.7774이며 안정화되지 않은 부분은 평균 26.5957이다. 실험은 7번을 수행하였고 모두 우수한 성능을 나타내었다. 실험결과 안정화 성능을 정량적으로 확인할 수 있었으며 대략 초당 16프레임 비율의 처리속도를 보였다.

3.2 기구적안정화장치와 결합 실험

본 연구의 궁극적인 목표는 기구안정화와 영상안정화를 결합한 복합 안정화 장치 구현에 있다. 따라서 기구안정화 연구결과와 결합된 실험장치에서 영상안정화 성능을 확인하였다. 그림 8은 기구안정화 장치와 결합된 실험장치 모습이다.

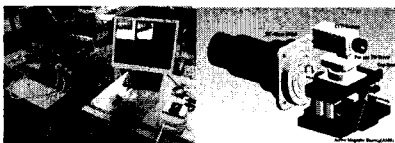


그림 8. 기구안정화장치와 결합된 영상안정화장치

실험장치를 야외에 이동하기 불가능하여 실내에서 사진을 실제카메라로 영상을 획득하는 것으로 실험하였다. 그림 8은 입력영상과 보상된 영상이다.



(a)1번째 영상 (b)92번째 영상



(a)189번째 영상 (b)252번째 영상

그림 8. 영상안정화 장치만의 실험결과 영상

기구부에서 Dual Mode Stage를 통해 전차 차량의 실제 회전 외란을 출력하도록 그림 1과 같이 카메라와 축을 평행하게 구성하였으며, 듀얼 스테이지 시스템에서는 플랫폼의 김발의 연성에 의한 외란을 제거하기 위해 슬라이딩 모드 제어를 이용하였다. 전차 차량의 실제 외란을 입력한 후의 실험결과 는 그림 9와 같다.

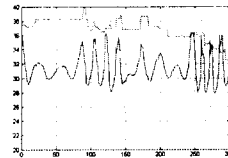


그림 9. 영상안정화 장치만의 실험결과(PSNR)

4. 결론

전차에 탑재되는 조준경의 롤 축 안정화에 적용 가능한 영상 안정화장치를 실제 구현하고 실험하였다. 영상안정화 장치만의 실험결과 초당 16프레임의 처리속도를 보이면서 안정화 성능도 우수하였다. 또한, 조준경의 3축 안정화장치에 적용가능성을 보이기 위해 기구적 안정화장치와 결합한 실험을 수행하였다. 실험결과 기구적 안정화 장치와 영상안정화 장치가 결합된 상태에서도 우수한 안정화 성능을 보였다. 이로서 전차에 탑재되는 조준경 및 기동 플랫폼에 장착된 카메라의 안정화 장치에 적용가능하리라 기대된다.

참 고 문 헌

- [1] H. D. Seok, H. K. Kwak and J. Lyou, "A Rotational Motion Estimation Method for Digital Image Stabilization", *IEEE international conference region 10, 2005/11/21*, 1568963922
- [2] H. D. Seok, J. Lyou and D. J. Kim, "영상 안정화를 위한 회전중심 및 각도 추정기법", *Journal of Control, Automation and Systems Engineering*, 1994.
- [3] A. Murat Tekalp, *Digital Video Processing*, Prentice Hall, 1995.
- [4] C. N and R. Cellappa. "Evaluation of image stabilization algorithms," *Proceedings of IEEE International Conference on*, Vol. 5, pp. 2789-2792, 1998.
- [5] K. Uomori, A. Morimura, and H. Ishii, "Electronic image stabilization system for video cameras and VCRs," *Journal of the Society of Motion Picture Television Engineers*, Vol. 101, pp. 66-75, 1992.
- [6] 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.