

실시간 적응형 Motion Estimation 알고리즘 및 구조 설계

이 준 환, 김 재 석
연세대학교 전기전자공학과
전화 : 02-361-4018 / 핸드폰 : 016-245-7580

A Adaptive Motion Estimation Using Spatial correlation and Slope of Motion vector for Real Time Processing and Its Architecture

Joonhwan Lee, Jaesook Kim
Dept. of Electronic engineering , Yonsei University
E-mail : montana@asic.yonsei.ac.kr

Abstract

This paper presents a new adaptive fast motion estimation algorithm along with its architecture. The conventional algorithm such as full-search algorithm, three step algorithm have some disadvantages which are related to the amount of computation, the quality of image and the implementation of hardware, the proposed algorithm uses spatial correlation and a slope of motion vector in order to reduce the amount of computation and preserve good image quality, The proposed algorithm is better than the conventional Block Matching Algorithm(BMA) with regard to the amount of computation and image quality. Also, we propose an efficient architecture to implement the proposed algorithm. It is suitable for real time processing application.

I. 서론

최근 위성통신과 광통신 시스템의 도입으로 고속 디지털 회선 등의 이용이 대중화됨에 따라 동화상 통신과 그에 관련된 기술의 개발에 대한 관심이 크게 증가하게 되었다. 디지털 영상은 아날로그 영상보다 화질이 깨끗하고 외부의 잡음에 대한 강한 데이터의 보존

성 및 다양한 디지털 신호 처리기술을 쉽게 적용할 수 있어 컴퓨터 그래픽 응용과 공중 전파망을 통한 깨끗한 화질의 HDTV(High-Definition Television) 방송 등에도 실용화되는 추세이다[1-4].

그러나 이러한 장점에도 불구하고 디지털 영상은 그 자체가 방대한 데이터 량을 요구하므로 동영상의 전송과 처리 및 저장 시에 하드웨어로 구현함에 있어서 방대한 하드웨어 부담을 필요로 하게 된다.

이러한 단점을 극복하고 적은 저장공간과 대역폭을 차지하면서도 적당한 화질을 유지하려는 효과적인 영상 데이터의 압축에 대한 연구가 진행되고 있다.

이러한 사실을 토대로, 본 논문에서는 BMA(Block Matching Algorithm)을 이용하여, 동영상에서의 인접된 프레임의 상관관계를 통하여 화상의 움직임을 예측하고 예측된 동영상 모션벡터를 전송함으로써 시간축의 방향으로의 영상압축을 수행하는 Motion Estimation의 새로운 알고리즘 개발 / 검증 및 구현을 수행하였다.

II. 제안된 알고리즘

2.1 기존의 알고리즘

Motion Estimation(ME)에서 널리 쓰이는 기존의 알

고리듬에는 Full-search 알고리듬과 TSS 알고리듬, NTSS 알고리듬, 2D-LOGS 알고리듬 등, 여러 가지 알고리듬이 있다[1-3]. 이러한 기존의 알고리듬은 실시간으로 적용되기에는 각각 단점이 있다. Full-search 알고리듬의 경우, 화질은 좋지만, 연산량 과다라는 단점이 있다[1].

연산량을 줄이고자 제안된 TSS 알고리듬 등은 국소정합(Local Minimum)이라는 단점이 존재한다. 즉, 위의 알고리듬들은 Full-search와는 달리, 탐색영역의 모든 후보블록에 대한 검색을 하지 않고, 일부 후보블록만 샘플링 하여 검색한다. 따라서 샘플링에서 누락된 후보블록들 중, 정합 블록이 존재하는 경우라면, 이들 알고리듬들은 탐색영역 전체의 정합블록이 아닌, 샘플링된 후보블록들 중의 정합블록을 찾게되는 국소 정합의 문제점이 발생한다.

이러한 문제점을 개선하기 위한 알고리듬으로는 Jie-Bin Xu의 알고리듬이 있다[2]. Jie-Bin Xu의 알고리듬은 주위 블록과 기준블록의 상관관계를 이용하여, 탐색영역의 중앙을 적응적으로 보정함으로써, 국소정합의 발생확률을 최소화하는 알고리듬이지만, 국소정합의 문제점이 완전히 해결된 것은 아니다.

2.2 제안된 알고리듬

본 논문은 화질은 우수하지만 연산량이 많은 단점을 가지고 있는 Full-Search 알고리듬을 수행하는 과정에서 얻어지는 여러 가지 정보를 이용하여, 연산량을 최소화하는 동시에 우수한 화질을 유지하는 것에 주안점을 두고 수행하였다. 탐색영역을 적응적으로 변환시켜, 연산량을 줄였으며, 그로 인한 화질저하의 문제는 탐색영역의 중심을 재설정함으로써 극복하였다. 탐색영역의 중심을 재설정할 때에는, 초기화 작업을 통해 얻어진 영상정보를 사용하게 된다. 제안된 알고리듬을 단계별로 보면 다음 그림과 같다.

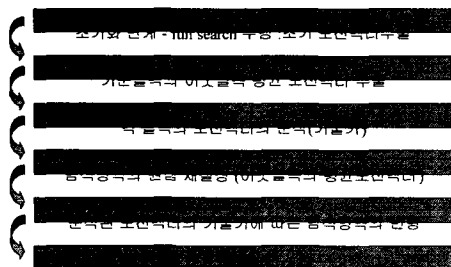


그림2-1 제안된 알고리듬의 처리 순서

2-2-1 초기화 단계

초기화 단계는 제안된 알고리듬에서 필요로 하는 각종 정보를 얻어내는 단계로서, 기존의 Full-Search 알고리듬과 동일하다. 각 프레임의 일정크기(N×N)의 블록으로 분할하여, 정해진 영역의 탐색영역을, 분할된 블록 단위로 탐색을 한다. 각각의 기준블록에 대한 최적의 정합 블록을 정하는 판단기준은 MAD(Mean Absolute Difference)를 사용했다.

$$MAD(k, l) = (1/N^2) \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |C(i, j) - R(i+k, j+l)| \quad (식 2-1)$$

2-2-2 기준블록의 이웃블록 평균모션벡터추출단계

이 단계에서는 하나의 블록을 기준으로 삼고, 전 단계(초기화 단계)에서 얻어진 공간적인 정보를 이용하여, 기준 블록의 이웃블록 평균모션벡터를 얻어내는 단계이다. 여기서 말하는 이웃블록이란 기준블록의 움직임에 영향을 끼치는 블록들로서, 이들 블록의 움직임은 우리가 찾고자 하는 블록과 많은 상관관계를 가지고 있다.[2]

이러한 상관관계를 이용하여, 연산량을 줄여 나가는 것이 이 알고리듬의 주된 내용이므로, 상당히 중요한 특성이라 할 수 있겠다. 이러한 관계는, 다음 그림을 참조하면, 쉽게 이해할 수 있다.

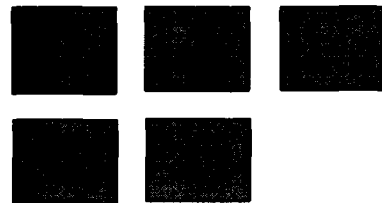


그림 2-2 기준블록(B0) 및 이웃블록

여기서 기준블록은 B0가 되며, 나머지 블록은 B0의 이웃블록이 된다.

즉, 이 단계는 기준블록(B0)을 탐색하기 위한 탐색영역의 원점을, 이웃블록(B1, B2, B3, B4)의 모션벡터의 평균으로 정하게 된다. 따라서 탐색영역의 원점을 설정하는 데 필요한 정보를 제공하는 단계라 할 수 있겠다.

2-2-3 각 벡터의 모션벡터분석(기울기) 단계

위에서 구한 이웃블록의 모션벡터평균의 기울기를 구하여, 기준블록의 움직임 성향을 예측하여 탐색영역을 적절하게 감소시킴으로써 연산량을 줄일 수 있다.

이웃블록과 기준블록과의 사이에는 상관관계가 존재하므로 이웃블록의 평균적인 움직임 성향으로 기준블록의 움직임 성향을 예측할 수 있다. 따라서 이웃블록의 평균모션벡터의 기울기를 구함으로써, 기준블록의 수직/수평의 움직임 성향을 예측하여, 적절하게 탐색영역을 축소함으로써 연산량을 줄일 수 있게 된다.

2-2-4 탐색영역의 원점 재설정 단계

이 단계에서는 이전 프레임의 이웃블록의 평균모션벡터로 하여, 블록의 움직임을 미리 예측하여, 블록의 예측된 이동위치에 탐색영역의 원점을 위치시켜 탐색하는 과정을 거치게 되는데, 이 과정에서 필요한 탐색영역의 원점을 재설정 하는 단계이다.

2-2-5 분석된 모션벡터의 기울기에 따른 탐색영역의 변형단계

모션벡터의 기울기를 분석하여 기울기의 크기에 따라 탐색영역의 수평방향 및 수직방향 성분을 감소시켜, 탐색영역의 면적을 줄임으로써 연산량을 줄일 수 있다.

2-2-6 변형된 탐색영역 내에서 Full-search 실행

위 과정을 거치면서 결정된 각 블록별 탐색영역 내에서 full-search 실행하는 단계이다. 전체적인 과정을 나타내면, 아래 그림 2-3 와 같다.

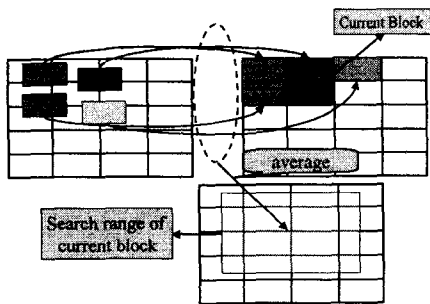


그림 2-3 제안된 알고리즘의 그림설명

III. Hard Ware 구조 제안

3.1 제안된 하드웨어의 전체구조

위의 소개된 알고리즘을 구현한 하드웨어의 전체

구조는 그림 3-1과 같다

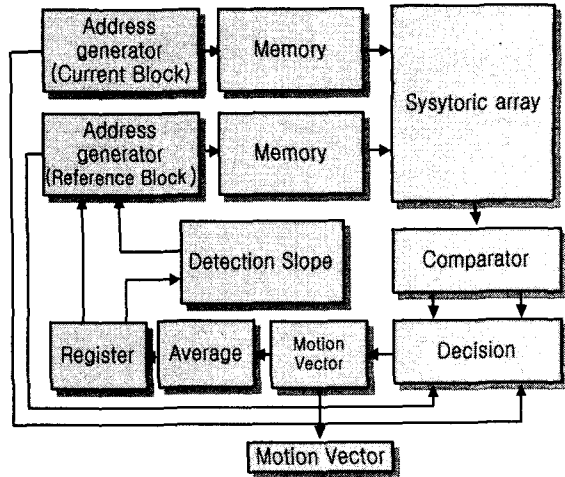


그림 3-1 전체 구조의 블록 다이어그램

위의 그림의 각 모듈은 독립적인 기능을 담당한다. 연속된 프레임을 각 메모리에 저장시킨 후, 현재 블록을 불러오는 주소 발생기와 후보 블록을 불러오는 주소발생기를 통하여 각각의 블록을 불러온 후, 이를 시스토픽 어레이에 입력으로 사용한다. 시스토픽 어레이에서 각 블록의 MAD를 계산하여, 비교기에 입력시키면, 비교기는 한 탐색영역의 범위에 해당하는 MAD 값을 저장하여, 그 중 최소의 값을 갖는 블록의 주소를 주소 결정모듈로 전달한다.

주소 결정모듈에서 모션 벡터가 추출되어 나오며, 이 모션벡터는 다음 단계인 평균기와 기울기 결정모듈, 이를 저장하는 레지스터 등을 통하여 처리된 데이터들은 후보 주소 발생기의 입력으로 사용되어, 탐색영역의 원점을 보정하게 되고, 탐색영역을 적응적으로 축소 시킨다. 다음 그림은 주소 발생기의 블록 다이어그램이다.

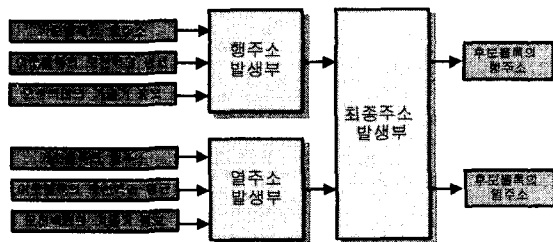


그림 3-2 후보 주소 발생기의 블록 다이어그램

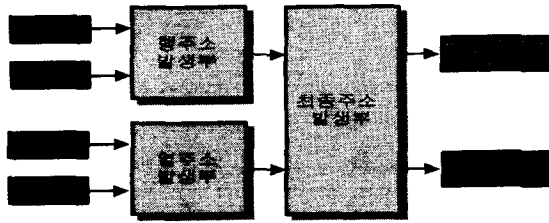


그림 3-3 기준블록의 주소발생기 블록 다이어그램

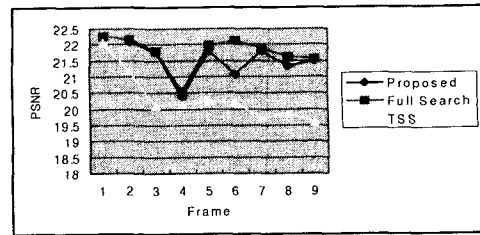


그림 4-1 각 알고리즘에 따른 프레임별 PSNR 비교

IV. 알고리즘 및 하드웨어의 검증

본 단원에서는 위의 알고리즘 및 하드웨어 설계를 검증하도록 하였다.

4-1 시뮬레이션 과정 / 결과

실험방법은 다음과 같이 하였다.

- ① 첫 번째 프레임, 두 번째 프레임의 full-search 알고리즘을 통한 화면 재구성
- ② 위의 단계에서 얻어진 재구성된 화면과 세 번째 프레임의 full-search 알고리즘을 통한 화면 재구성
- ③ 1단계에서 얻어진 정보를 이용하여 제안된 알고리즘을 적용, 화면을 재구성
- ④ 2단계와 3단계의 결과물 비교 (각 화면의 PSNR 비교/ 전체 프레임)

위의 과정을 거친 실험결과는 아래 표와 같이 얻을 수 있었다.

또한 알고리즘 별 프레임당 PSNR은 다음 그림 4-1과 같았다.

(표 4-1) 각 알고리즘별 연산량과 화질비교

	Full- search Algorithm	TSS algorithm	Proposed Algorithm
연산량/블록	$F_s = N^2(2p+1)^2$	$F_{TSS} = 25N^2$	$0.25F_s$ $\sim 0.5F_s$
연산량/프레임	$F_s = VHN^2(2p+1)^2$	$F_{TSS} = 25VHN^2$	$0.25F_s$ $\sim 0.5F_s$
PSNR	22.15db	21.18db	22.09db

블록의 크기 : $N \times N$, VH : 프레임내의 블록 개수,

p : 탐색영역의 범위, PSNR : 3rd frame의 결과

V. 결론

본 논문은 영상처리에 있어서, 속도 및 화질면에 크게 영향을 끼치는 모션벡터 추출을 최소의 연산량으로 추출함과 동시에, 우수한 화질을 유지하는 알고리즘의 개발 및 하드웨어 모듈화를 목적으로 진행되었다.

연산량은 Full-search의 최대 25%를 절감할 수 있지만, 이 경우는 화면 내의 모든 블록이 일정한 방향으로 움직이는 특수한 경우에 제한되며, 실제로 일반적인 화면에서의 연산량 절감은 50%가 된다. 이에 따른 화질의 저하는 약 0.05db 정도로, 거의 동일한 수준의 화질을 얻을 수 있었다.

본 논문을 통하여 개발된 알고리즘으로 인해 모션 벡터를 빠르고 정확하게 찾을 수 있을 것으로 기대되며, 화상전화 및 HDTV encoder 등의 다양한 분야에 응용될 수 있을 것으로 기대된다.

참고문헌(또는 Reference)

- [1] Gagan Gupta, "Architectures for Hierarchical and Other Block Matching Algorithms", IEEE. Trans. On Circuit and Systems For Video Technology, Vol. 5, No. 6, December 1995
- [2] Jie-Bin Xu, Lai-Man Po and Chok-Kwan Cheung, "Adaptive Motion Tracking Block Matching Algorithms for Video Coding", IEEE Trans. On Circuit and Systems For Video Technology, Vol. 9, No. 7, October 1999
- [3] 이태영, "움직임 예측기를 위한 새로운 1차원 부분 탐색 VLSI 알고리즘에 관한 연구", 연세대학교 석사학위 논문, 1995
- [4] JAE S. Lim, "Two-Dimensional Signal and Image Processing", Prentice hall, 1990