

논문 2013-50-1-15

H.264/AVC를 위한 고성능 움직임 예측 하드웨어 설계

(A Design of High Performance Motion Estimation Hardware for H.264/AVC)

박 승 용*, 류 광 기**

(Seungyong Park and Kwangki Ryoo)

요 약

본 논문에서는 고성능 H.264/AVC 부호기 설계를 위해 낮은 연산 복잡도를 가지는 움직임 예측 알고리즘과 하드웨어 구조를 제안한다. 제안하는 움직임 예측 알고리즘은 주변 매크로블록들의 움직임 벡터와 방향성으로 유동적인 초기 탐색점과 탐색 패턴으로 정확한 초기 탐색점을 설정한다. 주변 매크로블록들의 움직임 벡터를 사용하여 적은 수의 탐색점으로 움직임 예측이 가능하며, 적은 수의 탐색점으로 인해 연산량과 수행 사이클을 감소시킨다. 제안한 움직임 예측 하드웨어를 TSMC 0.18um CMOS 표준 셀 라이브러리 이용해 합성한 결과 217.92k 개의 로직 게이트로 구현되며 최대동작 주파수는 166MHz이다. 제안한 움직임 예측의 하드웨어 구조는 하나의 매크로 블록을 부호화 하는데 312사이클 소요되어 기존 하드웨어 구조대비 성능이 69% 향상됨을 확인하였다.

Abstract

In this paper, a new motion estimation algorithm with low-computational complexity is proposed to improve the performance of H.264/AVC. The proposed architecture uses the directions of the median motion vector which is computed by the motion vectors of the three neighbor macroblocks in Integer Motion Estimation. By using the directions of the vector, the proposed architecture has a single computational level instead of multi-computational levels in Integer Motion Estimation. The proposed motion estimation is synthesized using the TSMC 0.18um standard cell library. The synthesis result shows that the gate count is about 217.92K at 166MHz and it was improved about 69% compared with previous one.

Keywords : Motion Estimation, H.264/AVC, Inter Prediction, 부호화기

I. 서 론

최근 광대역 전송의 발전으로 유선 또는 무선의 광대역 접속회선이 고속화되고 저장매체가 대용량화되었다. 그러나 문자, 음성, 사진, 동영상과 같은 멀티미디어들

역시 보다 좋은 색상, 화질이 요구되어 사용되는 정보량은 시대에 따라 증가하고 있다. 특히 영상인 경우에는 음성에 비해 발생하는 정보량이 많기 때문에, 영상 데이터를 직접 전송하거나 저장하는데 많은 어려움이 있다. 따라서 영상 압축 효율이 높은 기술을 요구하게 되었다^[1]. H.264/AVC는 ITU-T와 ISO/IEC가 공동으로 진행하여 개발한 국제 영상 압축 표준으로써, MPEG-4 비디오 표준에 비해 약 2배 정도 향상된 압축 효율을 제공한다^[2].

H.264/AVC에서 움직임 예측 기법은 프레임의 상관관계를 이용하여, 프레임 사이의 시간적 중복성을 제거하기 위해서 사용한다. 움직임 예측 기법은 영상 압축

* 학생회원, ** 정회원, 한밭대학교 정보통신공학과
(Department of Information and Communication Engineering, Hanbat National University)

※ 본 논문은 교육과학기술부와 한국연구재단의 지역혁신인력양성사업 및 지식경제부 출연금으로 수행한 ETRI SW-SoC 융합 R&BD 센터와의 공동 연구의 결과입니다.

접수일자 2012년5월10일, 수정완료일 2012년12월27일

에서 일반적으로 연산량의 60% 이상을 차지하고 있는 중요한 처리 과정이다^[3].

연산량을 줄이기 위한 움직임 예측 알고리즘에 대한 연구가 지금까지 활발히 진행되고 있다. Choi^[4]가 제안하는 하드웨어 구조의 움직임 예측 알고리즘은 Full search 방식을 사용한다. Full search 방식은 정밀한 움직임 예측을 수행하지만, 검색 영역에서 모든 화소를 비교하기 때문에 높은 연산량을 가진다.

Tsai^[5]가 제안한 하드웨어 구조에서 사용하는 움직임 예측 알고리즘은 HEXBS 알고리즘에 기반을 가지는 EIMD와 PHS를 사용하였으며, 5개의 탐색점을 가지는 SPHSP 패턴과 7개의 탐색점을 가지는 LPHSP 패턴을 사용하여 중심점부터 단계별로 움직임 예측을 수행하기 때문에 최대 1,236개의 탐색점을 사용한다.

Ndili^[6]이 제안하는 움직임 예측 알고리즘은 HMDS로 13개의 탐색점을 가지는 LDSP 패턴과 17개의 탐색점을 가지는 변경한 다이아몬드 검색 패턴을 가진다. 총 3단계로 움직임 예측을 수행하기 때문에 최대 43개의 탐색점을 가진다. 이와 같은 움직임 예측 알고리즘 및 하드웨어 구조는 탐색점이 많기 때문에 높은 연산량과 수행 시간을 가진다^[4-6].

하동원^[7]이 제안하는 움직임 예측 알고리즘은 주변 매크로 블록들의 움직임 벡터를 이용하여 초기점을 설정하고, 13개의 탐색점을 가지는 다이아몬드형 탐색 패턴으로 움직임 예측을 수행한다. 이와 같은 초기점 설정 방식은 고속의 움직임 예측을 수행하기 위해 사용되며, 연산량 및 연산 시간을 최소화할 수 있다.

본 논문에서는 움직임 예측시 주변 매크로블록의 움직임 벡터들의 중간값과 방향성을 고려하여 초기 탐색점을 예측하는 알고리즘을 제안한다. 제안하는 알고리즘은 주변 매크로블록들의 움직임 벡터를 이용하기 때문에 초기 탐색점을 보다 정확하게 예측하여 적은 탐색점으로 연산량을 감소시키고, 영상의 손상을 최소화시킨다. 또한 하드웨어 구현시 적은 연산량을 가지고, 주변 매크로블록의 움직임 벡터만 저장하여 하드웨어 면적도 감소하는 구조로 설계한다.

본 논문의 구성은 다음과 같다. II장에서는 움직임 예측 알고리즘에 대해서 기술한다. III장에서는 제안하는 움직임 예측 알고리즘을 기술하며, IV장에서는 하드웨어 구조를 기술한다. V장에서는 제안한 움직임 예측 알고리즘과 기존 움직임 예측 알고리즘의 결과를 비교

분석하고, 제안하는 하드웨어 구조와 기존 하드웨어 구조의 합성결과를 비교 분석한다. 마지막으로 VI장에서는 본 논문의 결론을 맺는다.

II. 움직임 예측 알고리즘

움직임 예측은 프레임 간의 시간적 상관성을 이용하여 움직임을 압축하는 기술로서 이전 프레임과 현재 프레임 간의 시간적 유사성을 이용하여 움직임 정보를 예측한다. 움직임 보상은 이전 프레임에서 움직임 정보를 포함시켜 참조 프레임을 생성하고, 참조 프레임과 현재 프레임의 차이값을 구한다. 움직임 정보와 두 프레임 간의 차이값만 부호화하여 영상 정보 데이터를 크게 줄일 수 있다.

움직임 예측은 IME(Integer-pel Motion Estimation)과 FME(Fractional-pel Motion Estimation)으로 구분된다. 영상에서 물체의 움직임은 정수 단위로만 일어나지 않기 때문에 정화소 단위에서의 움직임 벡터가 실제의 움직임 벡터가 아닐 수도 있다. 이로 인해 H.264/AVC에서의 움직임 추정은 IME를 통해 정화소 단위에서 움직임 추정을 수행하여 블록 매칭 오차가 최소가 되는 지점을 중심으로 FME를 수행하여 최소 블록 매칭 오차 지점을 찾는다.

최소 블록 매칭 오차 지점을 측정하는 가장 일반적인 방법으로는 SAD(Sum of Absolute Difference)이며, 곱셈을 요구하지 않는 간편성 때문에 널리 사용된다. 현재 프레임을 $M \times N$ 크기로 분할하였을 때 (x, y) 에 위치한 현재 매크로블록과 $(x + v_x, y + v_y)$ 에 위치한 블록 간의 SAD는 식(1)으로 표현할 수 있다.

$$SAD(v_x, v_y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |F_t(x+m, y+n) - F_{t-i}(x+v_x+m, y+v_y+n)| \quad (1)$$

여기에서 F_t 는 현재 프레임이고, F_{t-i} 는 참조 프레임이다. 식 (1)을 통해 알 수 있듯이, SAD를 한 번에 구하기 위해서는 $M \times N$ 번의 뺄셈이 필요하다. 만약 탐색 변위가 $S_x \times S_y$ 일 때 전역 탐색 기법을 사용한다면 매크로블록 한 개당 $S_x \times S_y \times M \times N$ 번의 뺄셈이 필요한데, 이는 엄청난 연산량이라고 할 수 있다.

III. 제안하는 움직임 예측 알고리즘

본 논문에서는 고성능 H.264/AVC 인코더를 위해 움직임 예측의 수행시간 단축과 연산량 감소를 위한 새로운 움직임 예측 알고리즘 및 효율적인 하드웨어 구조를 제안한다.

제안하는 움직임 예측 알고리즘은 주변 매크로블록의 움직임 벡터와 방향성을 이용하여, 초기 탐색점 모드를 설정한다. 초기 탐색점 모드는 주변 매크로블록 움직임 벡터의 방향성을 고려하여 유동적인 초기 탐색점과 탐색 패턴을 가진다. 제안하는 알고리즘은 적은 탐색점으로 연산량을 감소시켰고, 유동적인 초기 탐색점과 탐색 패턴으로 정확한 초기 탐색점을 설정한다.

제안하는 움직임 예측 알고리즘에서 사용하는 주변 매크로블록의 움직임 벡터는 그림 1과 같다.

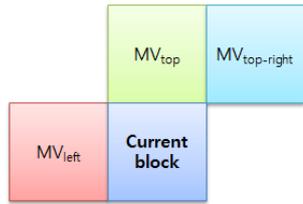


그림 1. 주변 매크로블록들의 움직임 벡터
Fig. 1. Motion vector of neighbor macroblocks.

초기 탐색점은 3개의 주변 매크로블록의 움직임 벡터를 식 (2)에 대입하여 중간 움직임 벡터 값(mvp)을 구한다.

$$mvp = \text{median}(MV_{left}, MV_{top}, MV_{top-right}) \quad (2)$$

식 (2)의 결과 값인 중간값(mvp)으로부터 탐색점 모드를 결정한다. 탐색점 모드 결정 방법은 식 (3)과 같이 중간값(mvp)의 x 좌표와 y 좌표를 비교하여 결정한다.

```

if(mvp.x == 0)
    if(mvp.y == 0) mode = 0;
    else if(mvp.y > 0) mode = 4;
    else mode = 3;
else if(mvp.x > 0)
    if(mvp.y == 0) mode = 1;
    else if(mvp.y > 0) mode = 6;
    else mode = 5;
else
    if(mvp.y == 0) mode = 2;
    else if(mvp.y > 0) mode = 8;
    else mode = 7;
    
```

(3)

계층적 움직임 예측 수행에 따른 많은 연산량을 감소시키기 위해 9가지 모드를 갖는 단일 계층 움직임 예측을 제안한다. 단일 계층 움직임 예측은 계층적 움직임 예측 수행에 따른 제어로직이 불필요하며, 적은 탐색점으로 연산량을 감소 시켰다. 또한 중간값(mvp)의 방향성으로 초기 탐색점을 설정하여 적은 수의 탐색점으로 넓은 탐색 영역을 가진다.

탐색점 모드는 방향성에 따라 총 9개의 모드로 구분되며, 각 모드에 다른 초기 탐색점의 설정은 그림 2와 같다. 탐색점 모드 0은 주변 매크로블록의 움직임 벡터가 없을 때 설정되며, 정확한 움직임 예측을 위해 다른 모드에 비해 넓은 탐색점을 가진다. 모드 1~4는 중간값(mvp)이 직선 방향성을 가질 때 다이아몬드 탐색 패턴을 사용하고, 중간값(mvp)의 방향성으로 초기 탐색점을 이동하여 설정한다. 모드 5~8는 중간값(mvp)이 대각선 방향성을 가질 때 사각형 탐색 패턴을 사용하고, 중간값(mvp)의 방향성으로 탐색점을 이동하여 설정한다. 이와 같은 방향성을 고려한 탐색점 모드 알고리즘은 적은 연산량을 가지며, 주변 매크로블록의 움직임 벡터를 사용하기 때문에 HD급 해상도에서도 적은 연산량 대비 PSNR의 손실이 미비하다.

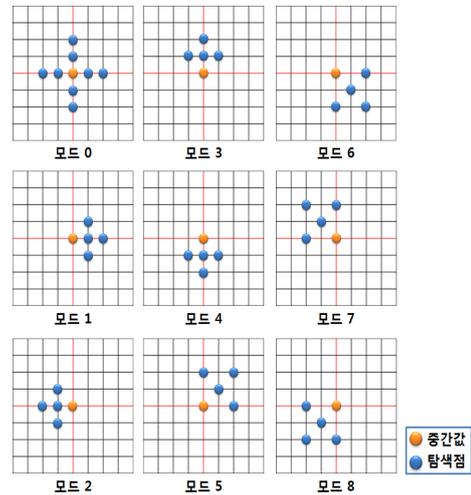


그림 2. 초기 탐색점 모드
Fig. 2. Initial search-point mode.

IV. 제안하는 움직임 예측 하드웨어 구조

제안하는 움직임 예측 알고리즘 기반 하드웨어 구조는 그림 3과 같이 ME Controller와 IME, FME,

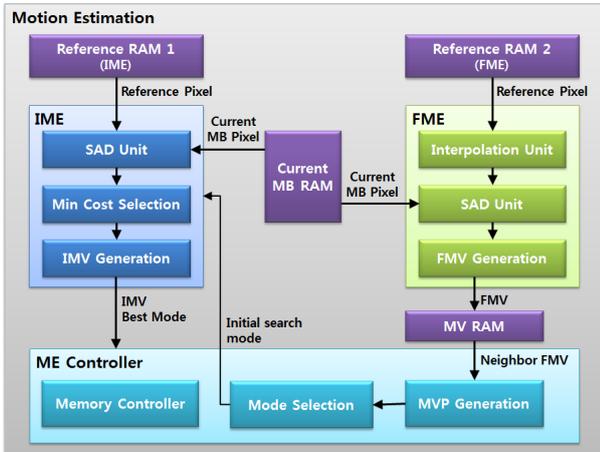


그림 3. 제안하는 움직임 예측 구조
Fig. 3. Architecture of the Proposed Motion Estimation.

Current MB RAM, MV RAM, 2개의 Reference RAM 으로 구성된다.

1. ME Controller 모듈

ME Controller 모듈은 MVP Generation, Mode Selection 모듈, Memory Controller 모듈로 구성된다. MVP Generation 모듈은 제안하는 중간 움직임 벡터 값을 구하는 모듈로서 주변 매크로블록 움직임 벡터 값을 비교기를 통해 중간값(mvp)을 출력한다. Mode Selection 모듈은 제안한 알고리즘을 기반으로 중간값(mvp)의 방향성에 따라 9가지 모드로 설정되며, 각 모드에 따라 최대 9개의 탐색점에서 최소 5개의 탐색점으로 설정한다. Memory Controller 모듈은 외부 메모리에 접근하여 현재 매크로블록의 데이터와 참조 프레임의 데이터를 읽고, 내부 메모리에 저장하는 기능을 수행한다.

2. IME 모듈

제안하는 IME 하드웨어 구조는 그림 4와 같다.

IME 모듈은 SAD Unit 모듈, Min Cost Selection 모듈, IMV Generation 모듈로 구성되며 16x16, 16x8, 8x16, 8x8, 8x4, 4x8, 4x4 크기를 갖는 가변 매크로블록들에 대한 움직임 예측을 수행한다.

제안하는 IME는 각 매크로블록을 1 사이클에 처리하기 위해 그림 5와 같이 4x4 매크로블록 단위로 SAD를 연산한다. SAD Unit 모듈은 16개의 4x4 SAD 모듈로 구성되며, 2 사이클 만에 현재 픽셀과 참조 픽셀의 차

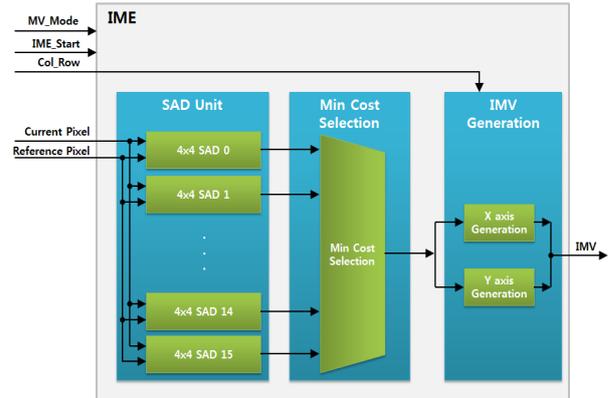


그림 4. 제안하는 IME 구조
Fig. 4. Architecture of the Proposed IME.

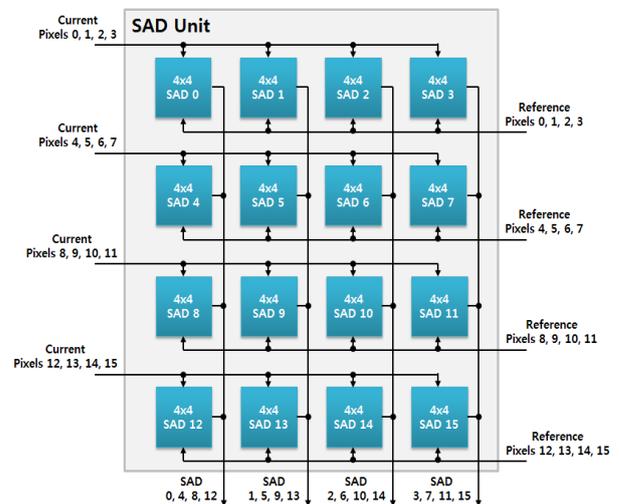


그림 5. 제안하는 SAD Unit 구조
Fig. 5. Architecture of the Proposed SAD Unit.

이값을 구하고 각 매크로블록 크기에 해당하는 차이값들을 더하여 1개 탐색점에 대한 SAD를 구한다.

Min Cost Selection 모듈은 SAD Unit 모듈에서 출력된 각 매크로블록 크기의 최소 SAD를 비교하여 저장한다. 최대 9개 탐색점에서 최소 5개 탐색점의 최소 SAD를 각 매크로블록 크기 마다 저장하고, 최소 SAD 값을 갖는 매크로블록을 정하여 출력한다. IMV Generation은 최소 SAD 값을 갖는 매크로블록의 움직임 벡터를 생성하며, 최소 SAD 값을 갖는 매크로블록 크기가 16x16일 경우 1개의 움직임 벡터를 생성하고, 4x4일 경우 16개의 움직임 벡터를 생성한다.

3. FME 모듈

제안하는 FME 하드웨어 구조는 그림 6과 같다.

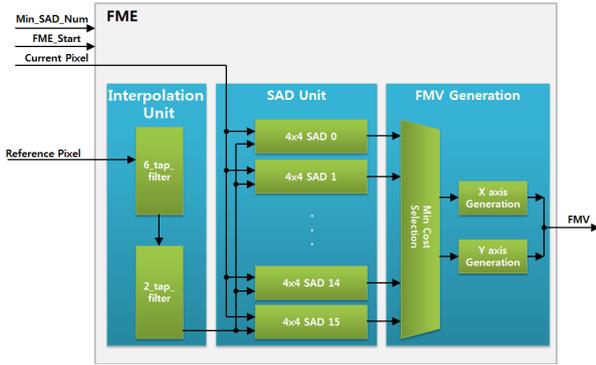


그림 6. 제안하는 FME 구조
Fig. 6. Architecture of the Proposed FME.

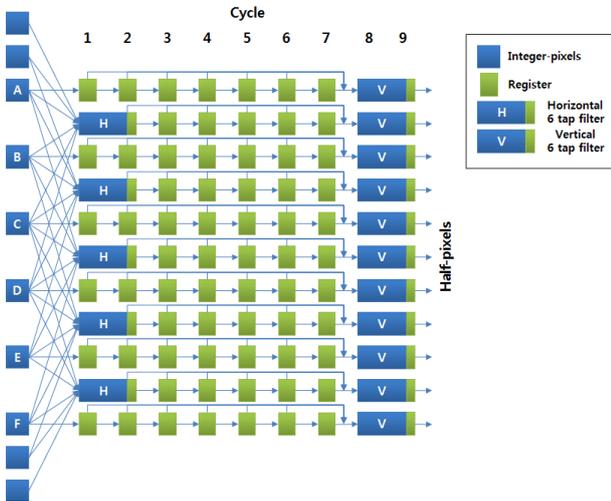


그림 7. 하프픽셀 보간 구조
Fig. 7. Architecture of the Half-pixel interpolation.

FME 모듈은 Interpolation Unit 모듈, SAD Unit 모듈, FMV Generation 모듈로 구성된다. Interpolation Unit 모듈은 부화소 움직임을 예측을 위해 참조 프레임의 픽셀들을 쿼드픽셀 단위로 보간하는 기능을 수행한다.

그림 7은 하프픽셀 보간의 하드웨어 구조이다. IME에서 출력된 IMV에 해당하는 참조 프레임의 픽셀을 입력받아 9 사이클에 가로에 해당하는 하프픽셀을 출력하고, 12 사이클 동안 FME에 필요한 8x8 하프픽셀을 출력하게 된다.

출력된 8x8 하프픽셀은 쿼드픽셀 보간을 위해 2 tap filter에 입력되며, 4x4 크기의 쿼드픽셀 보간을 수행하는 4개 모듈을 이용하여 16x16 쿼드픽셀을 출력한다.

쿼드픽셀 보간의 결과인 16x16 쿼드픽셀 데이터는 SAD Unit 모듈로 입력되며, 1 사이클 동안 SAD 연산을 수행한다. SAD Unit은 4x4 SAD 모듈 16개로 구성된다.

FMV Generation 모듈은 SAD Unit에서 출력된 SAD 값을 저장하고, 최소 SAD 값을 가지는 부화소 움직임 벡터를 생성한다. 생성된 부화소 움직임 벡터는 MV RAM에 저장한다.

V. 실험 및 고찰

본 논문에서는 HD급 영상(1280x720, 1920x1080, 300 프레임, 4:2:0)을 이용하여 H.264/AVC 표준 참조 소프트웨어 JM18.0^[8]에서 움직임 벡터를 추출하였고, 제안한 움직임 예측 하드웨어를 시뮬레이션 한 결과와 참조 소프트웨어에서 추출한 움직임 벡터를 비교하여 정상적으로 움직임 예측 수행을 확인하였다.

표 1은 H.264/AVC 참조 소프트웨어 JM18.0에서 테스트 한 기존 알고리즘인 EPZS와 제안하는 알고리즘과의 PSNR 결과값이다. EPZS와 제안하는 알고리즘과의 PSNR 차이는 평균 0.061dB 감소하였다.

표 2는 제안하는 움직임 예측 하드웨어 구조를 IDEC에서 지원하는 CAD tool을 사용하여, TSMC 0.18 μ m 공정 셀 라이브러리로 합성한 결과이다.

표 1. 제안하는 알고리즘의 PSNR 비교
Table 1. Comparison of the Proposed algorithm.

입력영상	QP	EPZS	Proposed	차이값
shields (720p)	22	41.119	41.114	0.005
	28	37.669	37.616	0.053
	32	36.008	35.934	0.074
	38	33.519	33.506	0.013
stockholm (720p)	22	41.759	41.751	0.008
	28	38.121	38.116	0.005
	32	36.605	36.586	0.019
	38	34.420	34.397	0.023
sunflower (1080p)	22	44.310	44.302	0.008
	28	41.274	41.224	0.05
	32	39.436	39.338	0.098
	38	37.875	37.504	0.371
평균		38.509	38.449	0.061

표 2. 움직임 예측 하드웨어 비교
Table 2. Comparison of the Motion Estimation architecture.

비교 구분	Choi ^[4]	Proposed	차이값
최대 지원 해상도	SD	FHD	-
움직임 예측 알고리즘	Full search	Proposed	-
공정	0.18 μ m	0.18 μ m	-
동작 주파수	54MHz	166MHz	-
매크로블록 당 수행 사이클 수	1.024	312	712
게이트 (k)	284	217.92	66.08

제안하는 하드웨어 구조의 최대 동작 주파수는 166MHz이고 게이트 수는 217.92k이다. 제안한 구조의 게이트 수는 Choi^[4] 대비 66.08k 만큼 감소하였다. 또한 제안한 구조는 하나의 매크로블록을 부호화 하는데 312 사이클 소요되어 Choi^[4]에 비하여 성능이 약 69% 향상됨을 확인하였다.

VI. 결 론

본 논문에서 제안하는 움직임 예측은 계층적 움직임 예측 수행에 따른 많은 연산량을 감소시키기 위해 9가지 모드를 갖는 단일 계층 움직임 예측을 수행한다. 단일 계층 움직임 예측은 계층적 움직임 예측 수행에 따른 제어로직이 불필요하며, 적은 수의 탐색점으로 연산량을 감소시켰다. 또한 중간값(mvp)의 방향성으로 초기 탐색점을 설정하여 적은 수의 탐색점으로 넓은 탐색 영역을 가진다. 제안하는 움직임 예측 알고리즘은 표준 참조 소프트웨어 JM18.0의 EPZS 알고리즘과 비교하여 PSNR 차이는 평균 0.061dB 감소로 미비하였다. 제안한 움직임 예측 방법을 하드웨어로 설계하여 TSMC 0.18 μ m 공정 셀 라이브러리로 합성한 결과 최대 동작 주파수는 166MHz이고 게이트 수는 217.92k이다. 제안한 구조는 하나의 매크로 블록을 부호화 하는데 312사이클 소요되어 기존 하드웨어 구조대비 성능이 69% 향상됨을 확인하였다.

참 고 문 헌

- [1] 정제창, *H.264/AVC 비디오 압축 표준*, 홍릉과학출판사, 2005.
- [2] J. V. Team, *Advanced Video coding for generic audiovisual services*, ITU-T Recommendation H.264 and ISO/IEC 14496-10 AVC, May 2005.
- [3] J. Ostermann, T. Wedi, et al., "Video Coding with H.264/AVC : tools, performance, and complexity", *IEEE Circuits and System Magazine*, vol. 4, pp. 7-28, 2004.
- [4] J. H. Choi, Y. H. Jung, Y. S. Lee and J. S. Kim, "New Motion Vector Prediction for Pipelined Motion Estimation in H.264/ACV," *International Conference on Electronics, Information, and Communication (ICEIC)*, Uzbekistan, June 2008.
- [5] T. H. Tsai and Y. N. Pan, "High Efficiency Architecture Design of Real-Time QFHD for H.264/AVC Fast Block Motion Estimation," *IEEE Circuits and Systems for Video Technology*, vol. 21, no. 11, November, 2011.
- [6] O. Ndili and T. Ogunfunmi, "Algorithm and Architecture Co-Design of Hardware-Oriented, Modified Diamond Search for Fast Motion Estimation in H.264/AVC," *IEEE Circuits and Systems for Video Technology*, vol. 21, no. 9, September, 2011.
- [7] 하동원, 조효문, 이중화 "고속 움직임 추정을 위한 시공간적 상관관계 기반의 효율적인 부분 왜곡 탐색 알고리즘," *대한전자공학회*, 제 47권 SD편, 제 1호, pp. 79-85, 2010년 1월
- [8] Joint Video Team(JVT) Reference Software JM 18.0.

저 자 소 개



박 승 용(학생회원)
 2010년 한밭대학교 공과대학
 정보통신공학과 공학사
 2012년 한밭대학교 정보통신전문
 대학원 정보통신공학과
 공학석사

2012년~현재 한밭대학교 정보통신전문대학원
 정보통신공학과 박사과정
 <주관심분야 : SoC 플랫폼 설계 및 검증, 영상신
 호처리>



류 광 기(정회원)
 1986년 한양대학교 전자공학과
 공학사
 1988년 한양대학교 전자공학과
 공학석사
 2000년 한양대학교 전자공학과
 공학박사

1991년~1994년 육군사관학교 교수부 전자공학과
 전임강사
 2000년~2002년 ETRI 시스템IC설계팀
 선임연구원
 2010년~2011년 Univ of Texas at Dallas
 방문교수
 2003년~현재 한밭대학교 정보통신공학과 교수
 <주관심분야 : SoC 플랫폼 설계 및 검증, 하드웨
 어/소프트웨어 통합설계 및 검증, 멀티미디어 코
 렉 설계>