

주파수 영역에서의 움직임 예측 및 보상을 위한 재귀 방정식을 이용한 웨이브프런트 어레이 프로세서

정회원 이주흥*, 류철**

A Wavefront Array Processor Utilizing a Recursion Equation for ME/MC in the Frequency Domain

Jooheung Lee*, Chul Ryu** *Regular Members*

요 약

본 논문은 DCT(Discrete Cosine Transform) 기반의 움직임 예측 및 보상을 위한 새로운 연산 아키텍처를 제안한다. 기존 방식들의 경우 연산 시간의 단축을 위하여 2차원 DCT 계수의 희소성을 충분히 활용하지 못하고 있다. 본 논문에서는 DCT 영역에서의 효율적인 움직임 예측을 위한 재귀 방정식을 유도하고, 이를 바탕으로 PE로 구성된 WAP를 개발한다. 또한, 재귀 방정식을 이용하여, 움직임 예측된 영상이 저주파 성분부터 고주파 성분까지 다양한 주파수 대역을 갖는 것이 가능함을 보인다. WAP는 아키텍처의 수정 없이 로그형 탐색이나 3단계 탐색과 같은 다양한 움직임 예측 알고리즘들을 수행할 수 있으며, 이러한 특성들은 비디오 부호화와 복호화에 필요한 전력 소모를 줄이기 위하여 이용될 수 있다. 본 논문에서 제안한 WAP 아키텍처는 계산의 복잡도와 연산 시간을 효과적으로 감소시키며, SAD기준을 이용한 DCT 영역에서의 움직임 예측 및 보상 방식은 SAD 또는 SSD 기준을 이용한 공간 영역에서의 움직임 예측 및 보상 방식보다 높은 PSNR과 압축률을 제공함을 보여준다.

Key Words : Motion Estimation and Compensation, Recursion Equation, Sparse Matrix Computation, Video Coding, Wavefront Array Processor

ABSTRACT

This paper proposes a new architecture for DCT-based motion estimation and compensation. Previous methods do not take sufficient advantage of the sparseness of 2-D DCT coefficients to reduce execution time. We first derive a recursion equation to perform DCT domain motion estimation more efficiently; we then use it to develop a wavefront array processor (WAP) consisting of processing elements. In addition, we show that the recursion equation enables motion predicted images with different frequency bands, for example, from the images with low frequency components to the images with low and high frequency components. The wavefront array processor can reconfigure to different motion estimation algorithms, such as logarithmic search and three step search, without architectural modifications. These properties can be effectively used to reduce the energy required for video encoding and decoding. The proposed WAP architecture achieves a significant reduction in computational complexity and processing time. It is also shown that the motion estimation algorithm in the transform domain using SAD (Sum of Absolute Differences) matching criterion maximizes PSNR and the compression ratio for the practical video coding applications when compared to the motion estimation algorithm in the spatial domain using either SAD or SSD.

* 펜실베이니아 주립대학교 전기공학과 (jzl116@psu.edu)

** 동국대학교 정보통신공학과 (cryu@dgu.ac.kr)

논문번호 : KICS2006-04-174, 접수일자 : 2006년 4월 17일, 최종논문접수일자 : 2006년 9월 1일

I. 서론

영상 부호화 기술은 HDTV, 단말기 영상 통신 및 인터넷 상에서의 비디오 스트리밍 등 다양한 영역에서 광범위하게 사용되고 있으며, 이러한 기술들의 호환성을 위하여 H.26X와 MPEG 계열과 같은 영상 부호화 표준들이 활용되고 있다. 대부분의 영상 부호화기는 DCT, 움직임 예측 및 보상, 그리고 엔트로피 코딩 등으로 구성되어, 영상 데이터에 존재하는 공간적, 시간적, 통계적인 중복성을 효과적으로 제거하는데 사용된다. 움직임 예측 알고리즘은 가장 많은 계산을 필요로 하는 부분으로, 전체 계산 시간의 약 60%~80%를 차지하는 것으로 알려져 있다^[1]. 또한, 사용된 움직임 예측 알고리즘에 따라 복원된 영상의 화질에 큰 영향을 주며, 보다 정확한 예측은 압축률을 높이고 주어진 비트율에서 PSNR (Peak Signal to Noise Ratio)을 향상시킨다^{[2],[3]}. 움직임 예측 알고리즘은 영상 부호화 표준에 명시되어 있지 않지만, 서로 다른 하드웨어 복잡도와 전력 소비량, 그리고 수행 시간을 갖는 다양한 알고리즘들이 현재 개발되어 있다^[4].

움직임 예측 알고리즘은 크게 공간 영역을 기반으로 한 방식과 주파수 영역을 기반으로 한 방식으로 구분될 수 있다. 공간 영역 알고리즘 방식은 matching 알고리즘과 gradient 기반 알고리즘으로 구성되며^{[5]-[7]}, 주파수 영역 알고리즘은 phase correlation 알고리즘과 웨이블릿 변환 기반 알고리즘, 그리고 DCT-기반 알고리즘으로 구성된다^{[8]-[10]}.

주파수 영역에서의 phase correlation 알고리즘은 정확한 움직임 예측을 제공하는 반면, 넓은 탐색 영역과 높은 연산량을 갖는 FFT에 기반하므로, 기존 DCT-기반의 비디오 부호화 표준과 호환성이 떨어

지는 단점이 있다. 그러나 DCT 기반 움직임 예측 알고리즘의 경우 일반적인 DCT-기반의 영상 부호기를 단순화할 수 있는 장점이 있다. 일반적인 DCT-기반 영상 부호기에서의 피드백 루프는 DCT, IDCT, 양자화기, 역 양자화기, 그리고 공간 영역에서의 움직임 예측 및 보상과 같은 함수들로 구성되어 있다. 그러나, 움직임 예측 및 보상이 DCT 영역에서 수행될 경우 IDCT는 피드백 루프에서 제거될 수 있으며, 그 결과 피드백 루프는 양자화기, 역 양자화기, 그리고 변환 영역에서의 움직임 예측 및 보상의 함수들만으로 구성이 가능하다. 그림 1-(a)는 일반적인 DCT-기반의 비디오 부호기를, 그림 1-(b)는 변환 영역에서의 움직임 예측 및 보상을 이용한 단순화된 비디오 부호기를 나타내고 있다. 그러나, 그림 1-(b)의 DCT-기반 움직임 예측 및 보상 방식의 경우, 높은 연산량을 필요로 하는 단점을 가지고 있다.

본 논문에서는 2차원 DCT 계수의 회소성에 기초로 한 DCT-기반의 움직임 예측 알고리즘으로부터 연산량을 크게 줄일 수 있는 재귀 방정식을 유도한다. 이를 기반으로 하여, VLSI 설계에 적합하고 이전의 구조보다 더 높은 처리량을 제공하는 WAP(Wavefront Array Processor)를 개발한다. 재귀 방정식은 움직임 예측된 영상이 저주파 성분을 갖는 영상부터 저주파와 고주파 성분을 갖는 영상까지 다양한 주파수 대역을 갖는 것이 가능함을 보이며, WAP은 아키텍처의 수정 없이 로그형 탐색이나 3단계 탐색과 같은 다양한 움직임 예측 알고리즘들을 실시간으로 수행할 수 있는 장점을 가지고 있다. 이러한 특성은 전력 소모에 따라 적응적으로 움직임 예측 및 보상을 수행하는 영상 부호기의 설계 시 효율적으로 이용될 수 있다.

본 논문의 구성은 다음과 같다. 제2장에서 DCT-

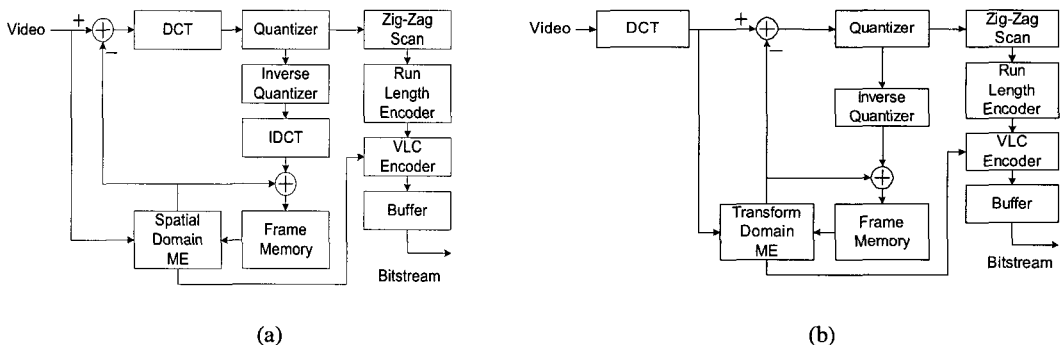


그림 1. 영상 부호기 아키텍처. (a) 공간 영역에서의 움직임 예측 및 보상을 이용한 비디오 부호기. (b) DCT 영역에서의 움직임 예측 및 보상을 이용한 비디오 부호기

기반의 움직임 예측 및 보상 알고리즘을 설명하고, 제3장은 VLSI 구현을 위한 재귀 방정식을 유도한다. 제4장에서는 2-D DCT 성분의 회소성을 이용하여 높은 처리량을 제공하는 새로운 아키텍처를 제안한다. 제5장은 변환 영역에서의 움직임 예측 및 보상을 위한 제안된 아키텍처가 2차원 DCT 계수의 회소성을 활용하여, 계산의 복잡도와 연산 시간을 효과적으로 감소시키는 것을 보여주며, SAD(Sum of Absolute Differences) 기준을 이용할 경우, SAD 또는 SSD(Sum of Squared Differences) 기준을 이용한 공간영역에서의 움직임 예측 및 보상 방식보다 PSNR과 압축률의 측면에서 더 좋은 결과를 나타내고 있음을 보여준다. 마지막으로 제6장에서 본 논문에서 제안된 방법을 간단히 정리하고 끝을 맺는다.

II. 주파수 영역에서의 움직임 예측과 보상

크기가 $0 \leq n \leq N-1$ 인 1-D 시퀀스 $x(n)$ 의 DCT는 다음과 같이 주어지며,

$$y(n) = \sum_{k=0}^{N-1} x(n)t(k,n), \quad 0 \leq k \leq N-1 \quad (1)$$

$N \times N$ 의 DCT 행렬 $T = \{t(k,n)\}$ 은 k 번째 행과 n 번째 열의 행렬 성분인 $t(k,n)$ 로 표기된다.

$$t(k,n) = \begin{cases} \frac{1}{\sqrt{N}}, & k=0, 0 \leq n \leq N-1 \\ \sqrt{\frac{2}{N}} \cos \frac{\pi(2n+1)k}{2N}, & 1 \leq k \leq N-1, 0 \leq n \leq N-1 \end{cases} \quad (2)$$

DCT 변환을 이용한 정지 영상과 동영상 압축 표준들은 8×8 크기의 DCT 블록을 사용하므로 본 논문에서도 N 이 8인 경우를 고려한다. Plompen이 제안한 이동 행렬(shifting matrix) 기반의 기법은 DCT 영역에서 이전 블록의 행렬과 이동 행렬의 곱셈을 사용하여 정수 화소의 정밀도를 가진 움직임 예측 및 보상을 수행할 수 있음을 보여준다^[14]. 이 논문은 변환 영역에서의 움직임 예측 및 보상 기법이 공간 영역에서의 움직임 예측 및 보상 기법보다 더 좋은 화질을 제공함을 보여준다.

Plompen이 제안한 이동 행렬 기반 알고리즘의 기본 이론은 다음과 같다. 예측된 블록 f_{pred} 는 주변 4개 블록인 f_0, f_1, f_2, f_3 의 수평, 수직 변환된 형

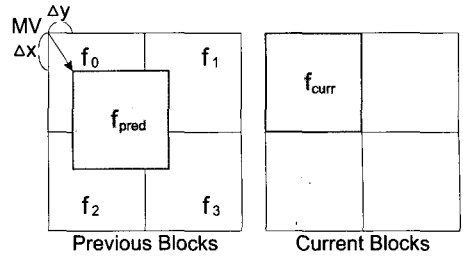


그림 2. 움직임 예측

태의 합으로써 표현될 수 있고, 그림 2는 현재 블록인 f_{curr} 에 대하여 가장 정확히 예측된 f_{pred} 와 그에 따른 움직임 벡터의 관계를 설명한다. 예측된 블록 f_{pred} 는 식 (3)에서 보는 바와 같이 수직 변환 행렬 V_i 와 수평 변환 행렬 H_i 를 이용하여 계산된다.

$$f_{pred} = \sum_{i=0}^3 V_i f_i H_i \quad (3)$$

수직 변환 행렬 V_i 와 수평 변환 행렬 H_i 는 각각 다음과 같이 주어진다.

$$V_0 = D_{8-\Delta x}, \quad V_1 = D_{8-\Delta x}, \quad V_2 = D_{\Delta x}^t, \quad V_3 = D_{\Delta x}^t \quad (4)$$

$$H_0 = D_{8-\Delta y}, \quad H_1 = D_{\Delta y}, \quad H_2 = D_{8-\Delta y}^t, \quad H_3 = D_{\Delta y}^t \quad (5)$$

식 (4)와 (5)에서 사용된 변위 행렬 D_n 은

$$D_n = \begin{bmatrix} | & & | \\ 0 & I_n & \\ | & & | \\ \hline | & & | \\ 0 & 0 & \end{bmatrix} \quad (6)$$

로 정의되며, I_n 은 $n \times n$ 단위 행렬이다.

이전 프레임과 현재 프레임간의 시간적 중복성을 보다 효과적으로 제거하기 위해 정수 화소가 아닌 부화소(sub-pixel) 정밀도를 가지는 움직임 예측 기법들이 많이 연구되어 왔다. H.264와 MPEG-4와 같은 동영상 부화소 표준은 부화소 정밀도를 이용한 움직임 벡터를 제공하고 있으며, 이러한 연산은 더욱 높은 계산의 복잡도와 메모리 크기의 증가를 필요로 한다. 일반적으로 반화소 위치에서의 화소값들은 가장 근접한 정수 화소 위치의 화소값들을 이용한 양선형 보간법에 의하여 얻어진다. 그러나, 본

논문에서 유도한 식 (7)-(10)의 일반화된 수직 변환 행렬 V_i 와 수평 변환 행렬 H_i 를 사용할 경우에는 추가비용 없이 변환 영역에서의 부하소 정밀도를 가지는 움직임 예측을 수행할 수 있다.

$$V_0 = V_1 = \frac{1}{2}(D_{8-\lfloor \Delta x \rfloor} + D_{8-\lceil \Delta x \rceil}) \quad (7)$$

$$V_2 = V_3 = \frac{1}{2}(D_{\lfloor \Delta x \rfloor}^t + D_{\lceil \Delta x \rceil}^t) \quad (8)$$

$$H_0 = H_2 = \frac{1}{2}(D_{8-\lfloor \Delta y \rfloor} + D_{8-\lceil \Delta y \rceil}) \quad (9)$$

$$H_1 = H_3 = \frac{1}{2}(D_{\lfloor \Delta y \rfloor} + D_{\lceil \Delta y \rceil}) \quad (10)$$

여기서 $\lfloor x \rfloor$ 는 버림 (floor) 함수를, $\lceil x \rceil$ 는 올림 (ceiling) 함수를 나타낸다. 또한, 식 (12)와 식 (13)은 DCT가 갖는 직교성과 분리성을 보여주고 있다.

$$DCT(A) \cong \hat{A} = TAT^t \quad (11)$$

$$T \cdot T^t = I \quad (12)$$

$$DCT(AB) = DCT(A)DCT(B) = \hat{A}\hat{B} \quad (13)$$

따라서 식 (3)의 예측된 블록 f_{pred} 의 DCT 계수들을 식 (12)와 식 (13)을 이용하여 다음과 같이 전개할 수 있다.

$$\begin{aligned} \hat{f}_i &= DCT(f_{pred}) = DCT\left(\sum_{i=0}^3 V_i f_i H_i\right) \quad (14) \\ &= \sum_{i=0}^3 DCT(V_i)DCT(f_i)DCT(H_i) \\ &= \sum_{i=0}^3 (TV_i T^t)(Tf_i T^t)(TH_i T^t) \\ &= \sum_{i=0}^3 \hat{V}_i \hat{f}_i \hat{H}_i \end{aligned}$$

$$\hat{V}_i = TV_i T^t, \hat{f}_i = Tf_i T^t, \hat{H}_i = TH_i T^t \quad (15)$$

식 (14)는 예측된 블록 f_{pred} 의 DCT 계수들을 이전의 주변 블록 f_0, f_1, f_2, f_3 의 DCT 계수들과 미리 계산된 수평 변환 행렬 H_i 와 수직 변환 행렬 V_i 의 DCT 계수들을 이용하여 구할 수 있음을 설

명한다. 변환 영역에서의 움직임 예측과 보상 시, 현재 DCT 계수 블록 \hat{f}_{curr} 과 가장 정확히 예측된 DCT 계수 블록 \hat{f}_{pred} 의 상대적 위치로부터 움직임 벡터를 구하고, \hat{f}_{curr} 과 \hat{f}_{pred} 간의 차이값은 다시 부호화 과정을 거치게 된다.

III. 효율적인 VLSI 아키텍처를 위한 재귀 방정식의 유도

주파수 영역에서 사용되는 움직임 예측 알고리즘의 주된 단점은 높은 연산량에 있다. 이전 프레임의 주변 블록 f_0, f_1, f_2, f_3 의 DCT 계수들을 이용하여 예측된 블록 f_{pred} 의 DCT 계수들을 구하기 위해서는 8개의 행렬 곱셈과 3개의 행렬 덧셈이 요구된다. 하지만 DCT 변환의 특성을 이용하면 많은 연산량을 감소시킬 수 있다. 이는 주파수 영역에서 에너지가 집중되는 특성과 DCT 변환 후 양자화된 블록의 희소성을 활용한 새로운 방법이다.

$\hat{f}_i(m,n)$ 은 블록 \hat{f}_i 의 m 번째 행과 n 번째 열의 행렬 성분이며, 열 벡터 $\hat{V}_i(m)$ 은 행렬 \hat{V}_i 의 m 번째 열을 그리고, 행 벡터 $\hat{H}_i(n)$ 은 행렬 \hat{H}_i 의 n 번째 행을 나타낸다. DCT 변환 후 양자화된 블록 \hat{f}_i 의 특성에 기반을 두어 계산상의 복잡도를 줄이기 위해 식 (14)를 다음과 같이 유도할 수 있다.

$$\begin{aligned} \hat{f}_{pred} &= \sum_{i=0}^3 \hat{V}_i \hat{f}_i \hat{H}_i \quad (16) \\ &= \sum_{i=0}^3 \sum_{m=0}^7 \sum_{n=0}^7 \hat{f}_i(m,n) \hat{V}_i(m) \hat{H}_i(n) \\ &= \sum_{m=0}^7 \sum_{n=0}^7 \sum_{i=0}^3 \hat{f}_i(m,n) \hat{V}_i(m) \hat{H}_i(n) \end{aligned}$$

$$\begin{aligned} \hat{f}_{pred} &= \sum_{m=0}^7 \sum_{n=0}^7 \sum_{i=0}^3 \hat{f}_i \hat{H}_i(m) \hat{V}_i(n) \quad (17) \\ &= \hat{f}_0(0,0) \hat{V}_0(0) \hat{H}_0(0) + \hat{f}_1(0,0) \hat{V}_1(0) \hat{H}_1(0) \\ &\quad + \hat{f}_2(0,0) \hat{V}_2(0) \hat{H}_2(0) + \hat{f}_3(0,0) \hat{V}_3(0) \hat{H}_3(0) \\ &\quad + \hat{f}_0(0,1) \hat{V}_0(0) \hat{H}_0(1) + \hat{f}_1(0,1) \hat{V}_1(0) \hat{H}_1(1) \\ &\quad + \hat{f}_2(0,1) \hat{V}_2(0) \hat{H}_2(1) + \hat{f}_3(0,1) \hat{V}_3(0) \hat{H}_3(1) \\ &\quad + \hat{f}_0(1,0) \hat{V}_0(1) \hat{H}_0(0) + \hat{f}_1(1,0) \hat{V}_1(1) \hat{H}_1(0) \\ &\quad + \dots \\ &\quad + \hat{f}_0(7,7) \hat{V}_0(7) \hat{H}_0(7) + \hat{f}_1(7,7) \hat{V}_1(7) \hat{H}_1(7) \\ &\quad + \hat{f}_2(7,7) \hat{V}_2(7) \hat{H}_2(7) + \hat{f}_3(7,7) \hat{V}_3(7) \hat{H}_3(7) \end{aligned}$$

식 (16)에서 \hat{f}_{pred} 는 $\hat{V}_i(m)$ 과 $\hat{H}_i(n)$ 의 외적에 의하여 얻어진 베이스스 영상의 선형 조합으로 간주될

수 있고, $\hat{f}_i(m,n)$ 이 0값을 갖는 외적의 경우에는 계산에서 제외될 수 있기 때문에 \hat{f}_i 의 희소성을 이용하여 \hat{f}_{pred} 을 효과적으로 계산할 수 있게 된다. 이는 일반적으로 저주파 대역에 있는 소수의 $\hat{f}_i(m,n)$ 만 0이 아닌 값을 갖고, 고주파 대역에 있는 $\hat{f}_i(m,n)$ 의 대부분이 0의 값을 갖기 때문이다. 따라서 식 (16)을 다음과 같은 지그재그 스캔 순서로 정리할 필요가 있다. 식 (17)에서 0이 아닌 $\hat{f}_i(m,n)$ 만으로 스케일링된 $\hat{V}_i(m)$ 과 $\hat{H}_i(n)$ 의 외적은 \hat{f}_{pred} 의 최종 결과를 얻기 위하여 다음과 같이 나타낼 수 있다.

$$\hat{f}_{pred}^k = \hat{f}_{pred}^{k-1} + \hat{f}_i(m,n)\hat{V}_i(m)\hat{H}_i(n) \quad (18)$$

여기서 \hat{f}_{pred}^0 은 8×8 의 영행렬이며, k 는 1부터 f_0, f_1, f_2, f_3 의 0이 아닌 DCT 계수들의 전체 개수까지 증가한다.

식 (18)에서 유도된 재귀 방정식은 다음의 두 가지 중요한 점을 보여주고 있다. 첫째, 다른 주파수 대역을 갖는, 예를 들면 저주파 성분으로 구성된 영상부터 저주파와 고주파 성분으로 구성된, 움직임 예측 및 보상된 영상은 $\hat{f}_i(m,n)$ 의 개수를 조절하여 효율적으로 생성할 수 있으며, 이러한 특성은 압축 효율성과 연산량 간의 상호 trade-off 관계를 효율적으로 제어할 수 있는 방법을 제공한다. 둘째, V_i, H_i, \hat{f}_i 의 $N \times N$ 크기가 증가하고, \hat{f}_i 에서 0이 아닌 성분의 총 개수 m 이 감소할수록, 식 (18)의 재귀 방정식은 연산량을 효과적으로 줄일 수 있다. 식 (14)와 (17)의 연산량을 계산해 보면 다음과 같다. 식 (14)의 일반적인 행렬 계산에서 전체 곱셈량은 $8N^3$ 이고, 덧셈량은 $(8N-5)N^2$ 이 된다. 그러나, 식 (17)에서는 \hat{f}_i 의 각 0이 아닌 성분의 총 곱셈량은 $\hat{f}_i(m,n)$ 와 $\hat{V}_i(m)$ 의 곱셈량이 N 이고, $\hat{f}_i(m,n)\hat{V}_i(m)$ 과 $\hat{H}_i(n)$ 의 곱셈량이 N^2 이 되어 N^2+N 이 된다. 따라서, $0 \leq i \leq 3$ 이고 각 \hat{f}_i 에서 0이 아닌 성분의 개수가 m 인 경우, 식 (17)에 필요한 전체 곱셈량과 덧셈량은 각각 $4mN^2+4mN$ 과 $(4m-1)N^2$ 이 된다. 그러므로 $N=8$ 이고 \hat{f}_i 의 0이 아닌 성분의 총 개수 m 이 15보다 작을 경우, 식 (18)의 재귀 방정식은 일반적인 방법에 비하여 DCT 변환 후 양자화된 블록 \hat{f}_i 의 희소성을 충분히 이용함으로써 곱셈과 덧셈

모두에서 효율적으로 연산량을 줄일 수 있게 된다.

IV. 2-D DCT 계수의 희소성을 이용한 아키텍처 구현

제 4장에서는 식 (18)의 재귀 방정식을 이용하여 효율적으로 주파수 영역에서의 움직임 예측 연산을 수행할 수 있는 WAP 아키텍처를 제안한다. 그림 3에서 ROM에 저장되어 있는 DCT 변환된 이동 행렬의 값들은 컨트롤러의 제어 신호에 따라서 PE와 MUL 블록들로 구성된 아키텍처의 위쪽과 왼쪽으로 입력된다. 입력된 데이터는 제안된 아키텍처의 위쪽 좌측 모서리에서 아래쪽 우측 모서리까지 순차적으로 클럭에 동기되어 전달된다. 이러한 구조에서, 하나의 웨이브 프론트는 한 번의 재귀 연산 즉, 0이 아닌 $\hat{f}_i(m,n)$ 에 의하여 스케일링된 $\hat{H}_i(n)$ 과 $\hat{V}_i(m)$ 의 외적 연산을 수행하게 된다. 1개의 곱셈기와 2개의 출력 레지스터로 구성된 각각의 MUL은 0이 아닌 $\hat{f}_i(m,n)$ 과 $\hat{H}_i(n)$ 의 곱셈에 사용되며, MAC(Multiplier-Accumulator) 블록과 2개의 출력 레지스터로 구성된 각각의 PE는 $\hat{f}_i(m,n)$ 에 의하여 스케일링된 $\hat{H}_i(n)$ 과 $\hat{V}_i(m)$ 의 외적을 누적 연산하는데 사용된다. 따라서, 웨이브 프론트의 누적 연산을 통하여 순차적으로 저주파 성분부터 고주파 성분까지 \hat{f}_{pred} 를 구할 수 있으며, 최종 결과값은 8×8 PE 배열에 저장된다. 컨트롤러는 선택된 움직임 예측 알고리즘에 따라 필요한 ROM의 주소를 생성하며, SAD를 바탕으로 계산에 필요한 PE들을 지그재그 스캔 순서에 따라 결정한 후, 사용 여부에 따라서 각 PE의 ON/OFF를 제어한다. 식 (17)의 최종 결과를 얻은 후에는, 컨트롤러의 mux_sel 신호 (mux_sel0 ~ mux_sel7)를 이용하여 각 PE의 MAC에 저장된 데이터 값이 상위 PE의 MAC으로 전달된다. 컨트롤러의 output_sel 신호는 8:1 MUX로 구성된 출력 버퍼 모듈에서 8×8 PE 배열 첫 번째 행의 MAC들에 저장되어 있는 값들을 선택하기 위하여 사용된다. 즉, 8×8 PE 배열에 저장된 \hat{f}_{pred} 는 컨트롤러의 mux_sel 신호와 output_sel 신호에 의하여 지그재그 정렬되어 출력 버퍼에 전달된다. 그림 4에서는 이러한 timing diagram을 보여주고 있다. 식 (17)에서 \hat{f}_{pred} 이 \hat{f}_i 의 희소성을 이용하여 어떻게 계산되는지를 다음의 예를 들어 살펴보자. 8×8 주변 블록들의 DCT 계수가 양자화 과정을 거친 후 다음의

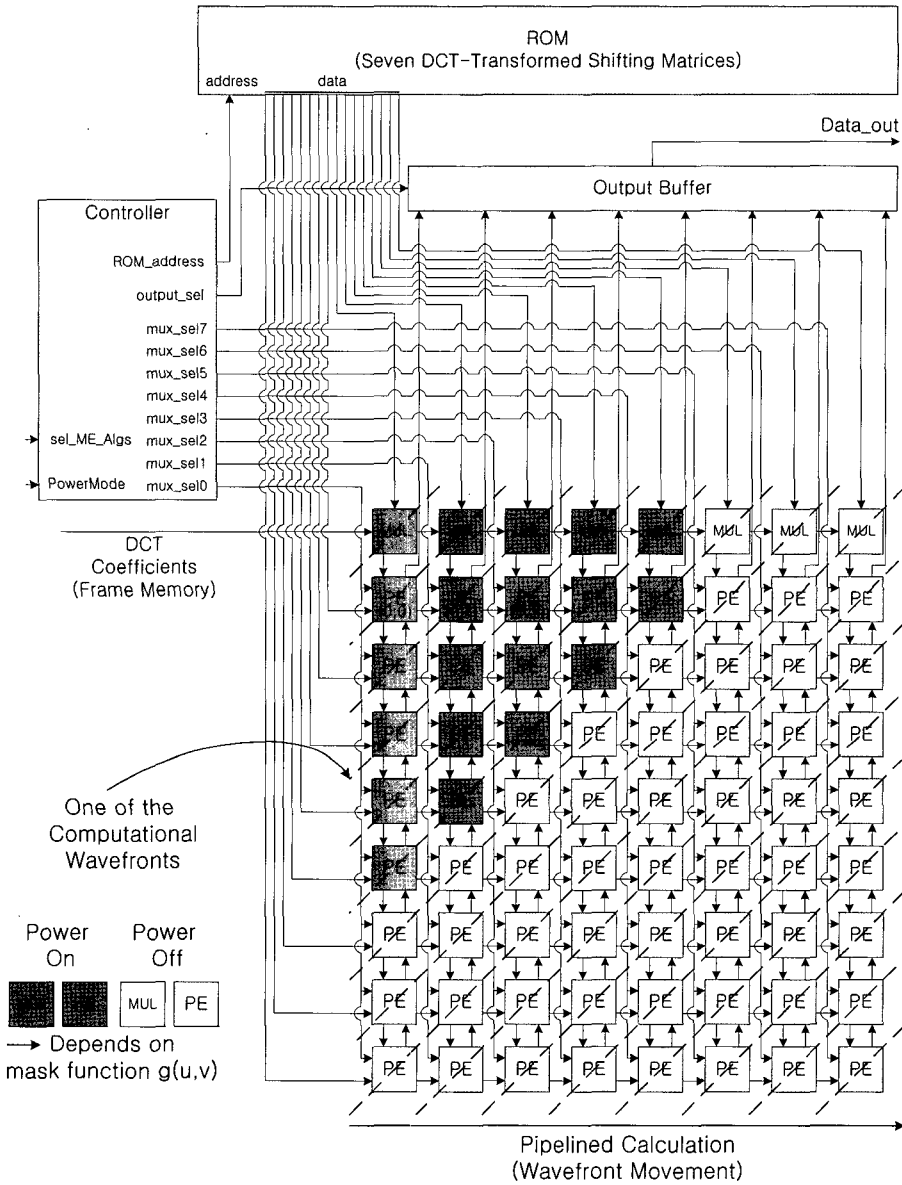


그림 3. WAP 아키텍처

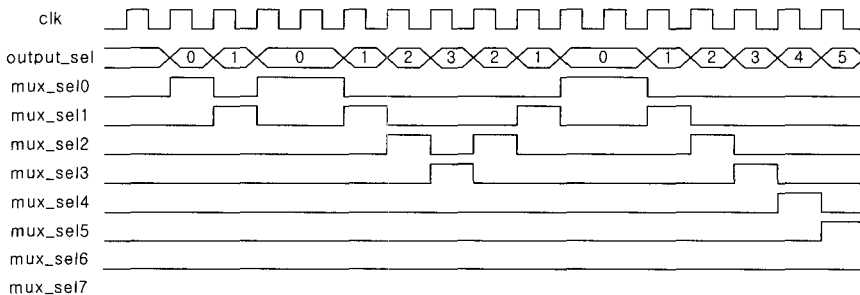


그림 4. 컨트롤러의 Timing Diagram

값을 갖는다고 가정한다.

$$\begin{aligned} \hat{f}_0 &= \begin{bmatrix} \hat{f}_0(0,0) & \hat{f}_0(0,1) & 0 & \dots & 0 \\ \hat{f}_0(1,0) & 0 & 0 & \dots & \vdots \\ 0 & 0 & 0 & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 \end{bmatrix} \\ \hat{f}_1 &= \begin{bmatrix} \hat{f}_1(0,0) & \hat{f}_1(0,1) & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & \vdots \\ 0 & 0 & 0 & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 \end{bmatrix} \\ \hat{f}_2 &= \begin{bmatrix} \hat{f}_2(0,0) & 0 & 0 & \dots & 0 \\ \hat{f}_2(1,0) & 0 & 0 & \dots & \vdots \\ 0 & 0 & 0 & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & 0 \end{bmatrix} \\ \hat{f}_3 &= \begin{bmatrix} \hat{f}_3(0,0) & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & \vdots \\ 0 & 0 & \hat{f}_3(2,2) & 0 & \dots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & \dots & \dots & \dots & 0 \end{bmatrix} \end{aligned} \quad (19)$$

양자화된 DCT 계수들은 지그재그 순서로 정렬된 후 RLC (Run-Length Coding) 과정을 거치되므로, 0이 아닌 DCT 계수의 위치 정보는 RLC 블록에서 알 수 있다. DCT 계수의 위치 정보 (6 비트)는 프레임 메모리에서 역 양자화된 0이 아닌 DCT 계수 (12 비트)와 함께 저장된다. 0이 아닌 DCT 계수들만이 연속적으로 WAP로 입력되어 연산을 수행하므로, 효율적으로 처리 시간을 단축시킬 수 있으며, 0이 아닌 DCT 계수의 위치 정보는 컨트롤러가 ROM에서 $\hat{V}_i(m)$ 과 $\hat{H}_i(n)$ 의 값들을 얻기 위한 주소

를 생성할 때 사용된다. 표 1은 대표적인 PE들과 이들이 수행하는 연산을 나타낸다.

V. Matching Criterion에 대한 고찰

움직임 예측 및 보상 알고리즘의 성능은 주로 탐색 방식, 탐색 영역, 최적의 움직임 벡터를 찾기 위한 비교 조건과 같은 요인들에 영향을 받는다. 일반적으로 움직임 벡터를 결정하기 위한 기준 조건으로는 SAD와 SSD를 많이 사용하고 있다. 제 5장에서는 공간 영역 및 변환 영역에서의 움직임 예측 및 보상 방식에서 SAD 또는 SSD와 같은 기준 조건들이 성능에 미치는 영향을 알아보려고 한다.

$$SAD(\Delta x, \Delta y)_{sp} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} |f_{curr}(u,v) - f_{pred,\Delta x,\Delta y}(u,v)| \quad (20)$$

$$SAD(\Delta x, \Delta y)_{tr} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} |\hat{f}_{curr}(u,v) - \hat{f}_{pred,\Delta x,\Delta y}(u,v)| \quad (21)$$

$$SSD(\Delta x, \Delta y)_{sp} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} (f_{curr}(u,v) - f_{pred,\Delta x,\Delta y}(u,v))^2 \quad (22)$$

$$SSD(\Delta x, \Delta y)_{tr} = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} (\hat{f}_{curr}(u,v) - \hat{f}_{pred,\Delta x,\Delta y}(u,v))^2 \quad (23)$$

다양한 비디오 시퀀스를 사용하여 실험을 한 결과 (제 6장 참조), SAD_{tr} 을 사용한 변환 영역에서의 움직임 예측 방식이 SSD_{tr} 을 사용한 변환 영역에서

표 1. 대표적인 PE들의 연산 예

	PE(0,0)	PE(0,1)	PE(0,2)
1	$f_{pred}^1(0,0) = f_0(0,0) \hat{V}_0(0,0) \hat{H}_0(0,0)$		
2	$f_{pred}^2(0,0) = f_{pred}^1(0,0) + f_1(0,0) \hat{V}_1(0,0) \hat{H}_1(0,0)$	$f_{pred}^1(0,1) = f_0(0,0) \hat{V}_0(0,0) \hat{H}_0(0,1)$	
3	$f_{pred}^3(0,0) = f_{pred}^2(0,0) + f_2(0,0) \hat{V}_2(0,0) \hat{H}_2(0,0)$	$f_{pred}^2(0,1) = f_{pred}^1(0,1) + f_1(0,0) \hat{V}_1(0,0) \hat{H}_1(0,1)$	$f_{pred}^1(0,2) = f_0(0,0) \hat{V}_0(0,0) \hat{H}_0(0,2)$
4	$f_{pred}^4(0,0) = f_{pred}^3(0,0) + f_3(0,0) \hat{V}_3(0,0) \hat{H}_3(0,0)$	$f_{pred}^3(0,1) = f_{pred}^2(0,1) + f_2(0,0) \hat{V}_2(0,0) \hat{H}_2(0,1)$	$f_{pred}^2(0,2) = f_{pred}^1(0,2) + f_1(0,0) \hat{V}_1(0,0) \hat{H}_1(0,2)$
5	$f_{pred}^5(0,0) = f_{pred}^4(0,0) + f_0(0,1) \hat{V}_0(0,0) \hat{H}_0(1,0)$	$f_{pred}^4(0,1) = f_{pred}^3(0,1) + f_3(0,0) \hat{V}_3(0,0) \hat{H}_3(0,1)$	$f_{pred}^3(0,2) = f_{pred}^2(0,2) + f_2(0,0) \hat{V}_2(0,0) \hat{H}_2(0,2)$

의 움직임 예측 방식이나 SAD_{sp} 또는 SSD_{sp} 를 사용한 공간 영역에서의 움직임 예측 방식보다 더 좋은 성능을 제공하였으며, SSD 를 기준으로 채택할 경우에는 공간 영역에서의 움직임 예측과 변환 영역에서의 움직임 예측이 동일한 성능을 보여 주었다. SSD 기준 조건과 움직임 예측 방식과의 상관관계를 수학적으로 명확하게 밝히기 위하여 Frobenius norm을 식 (24)와 같이 정의하면,

$$\|A\|_F = \left(\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |a_{i,j}|^2 \right)^{\frac{1}{2}}, \quad A \in \mathbb{R}^{N \times N} \quad (24)$$

DCT 변환과 같은 직교 변환은 Frobenius norm의 크기를 변환시키지 않는다. 그러므로 공간 영역에서의 움직임 예측을 통한 왜곡의 크기와 변환 영역에서의 움직임 예측을 통한 왜곡의 크기 사이에서 식 (25)와 같은 수학적 관계를 유도해 낼 수 있다. 즉, SSD 기준에 기반한 공간 영역에서의 움직임 예측을 통한 왜곡의 크기 SSD_{sp} 는 변환 영역에서의 움직임 예측을 통한 왜곡의 크기 SSD_{tr} 과 같게 된다. 그러므로 공간 영역과 변환 영역에서의 움직임 예측은 동일한 움직임 벡터를 생성하게 되고, PSNR과 압축률의 측면에서 동일한 성능을 제공하게 된다. 이러한 관찰 결과는 공간 영역과 변환 영역에서의 움직임 벡터 결정을 위한 조건 선택 시 유용하게 사용될 수 있으며, 이에 대한 자세한 고찰은 다음 논문에서 다루고자 한다.

$$\begin{aligned} SSD_{sp} &= \|f_{curr} - f_{pred}\|_F^2 & (25) \\ &= \|T(f_{curr} - f_{pred})T^t\|_F^2 \\ &= \|Tf_{curr}T^t - Tf_{pred}T^t\|_F^2 \\ &= \|\hat{f}_{curr} - \hat{f}_{pred}\|_F^2 = SSD_{tr} \end{aligned}$$

VI. 실험 결과

DCT 영역에서의 움직임 예측 및 보상 알고리즘

실험을 위하여 H.263 비디오 부호기를 사용하였다^[15]. 비디오 부호화기의 처리율은 그림 1에서 보여지듯이 주로 피드백 루프의 복잡도에 의해 제한되어진다. 그 중에서도, 움직임 예측 및 보상은 가장 많은 계산을 필요로 하는 부분으로, 이 연산을 위해 필요한 프레임 메모리는 전력 소모의 주된 원인 중 하나이다.

DCT는 공간 영역의 영상 신호를 주파수 영역으로 변환하여, 에너지를 저주파대로 집중시킨다. 따라서 양자화 과정 후에는 대부분의 고주파 성분의 DCT 계수들은 0이 되고, 몇몇 저주파 성분의 DCT 계수들만 존재하게 된다. 이러한 현상은 움직임 예측 및 보상을 수행하는 P-프레임 부호화에서 더욱 뚜렷이 나타나며, 양자화 파라미터가 커짐에 따라 0이 아닌 DCT 계수의 개수 또한 상당히 감소하게 된다. 제안된 파이프라인의 구조를 갖는 WAP 아키텍처는 0이 아닌 소수의 DCT 계수들과 이들의 위치 정보만을 이용하여 움직임 예측 및 보상을 수행하기 때문에 공간 영역에서의 움직임 예측 및 보상 방식과 비교 시, 보다 높은 연산 처리율을 얻을 수 있으며, 또한 프레임 메모리의 크기와 액세스를 크게 줄일 수 있다. 변환 영역에서의 움직임 예측은 DCT 계수들을 비교하여 움직임 벡터를 찾기 때문에 IDCT 블록이 필요하지 않으며, ROM으로부터 DCT 변환된 수직 변환 행렬과 수평 변환 행렬의 데이터를 읽어오기 위한 메모리 주소만 탐색 영역 위치에 따라 생성하면 되므로, 구조적인 추가 수정 없이 다양한 알고리즘들을 수행할 수 있다. 표 2는 제안된 변환 영역에서의 움직임 예측 기법과 공간 영역에서의 움직임 예측 기법을 비교하여 보여주고 있다.

본 실험에서는 공간 영역과 DCT 영역에서 전역 탐색 기법(Full Search Method)과 3단계 탐색 기법(3-Step Search Method)을 사용하여 성능 결과를 비교하였다. 움직임 벡터를 결정하기 위한 기준으로

표 2. 공간 영역과 변환 영역에서의 움직임 예측 기법 비교

	프레임 메모리	탐색 위치별 처리 시간	IDCT	데이터 플로우
변환 영역	$\frac{P_h \times P_v}{N^2} \times m \times (12+6)$	$15 + 4 \times m$	불필요	탐색 방식에 독립적
공간 영역	$P_h \times P_v \times 9$	$N \times N$	필요	탐색 방식에 의존적

$P_h \times P_v$: 영상 크기 (QCIF = 176×144, CIF = 352×288),
 $N \times N$: 블록 크기 (8×8), m : 블록별 0이 아닌 DCT 계수들의 숫자

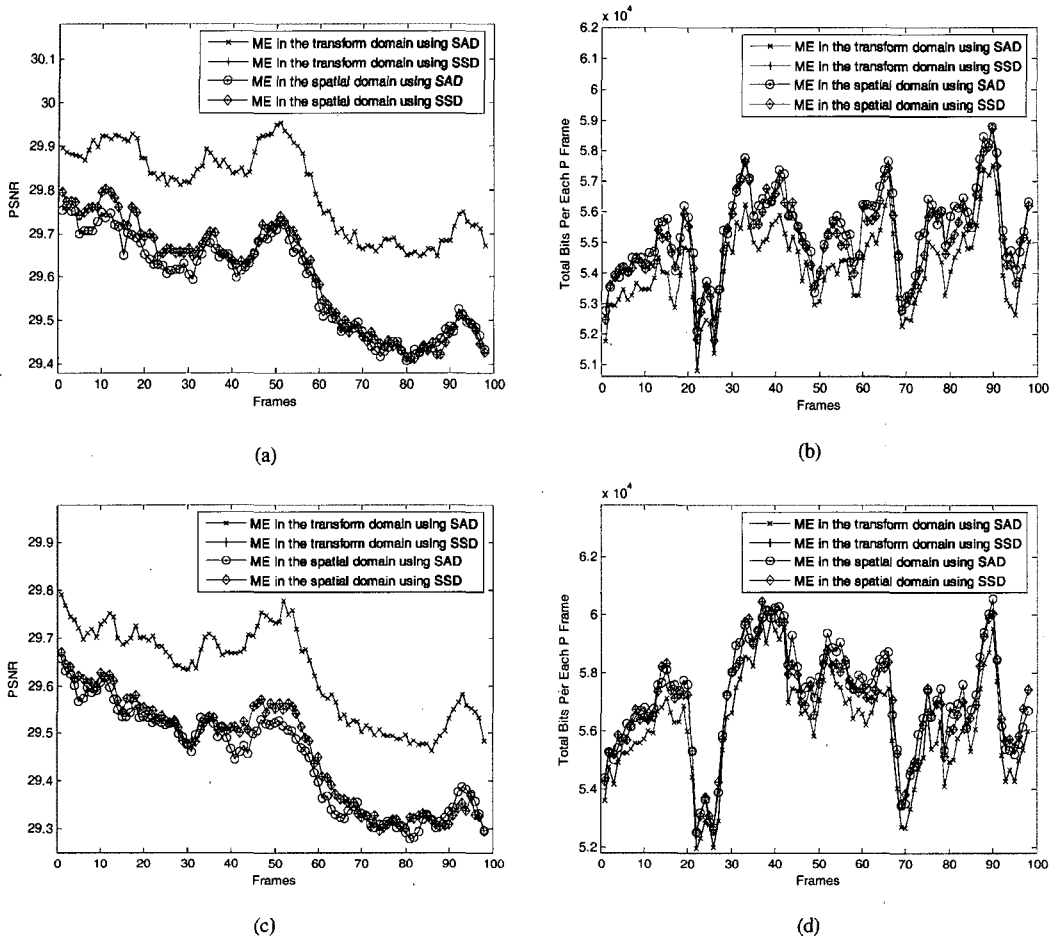


그림 5. paris CIF (352×288) 비디오 시퀀스를 사용한 움직임 예측 기법의 성능 비교
 (a) PSNR (전역 탐색) (b) 각 P-프레임별 총 비트수 (전역 탐색)
 (c) PSNR (3단계 탐색) (d) 각 P-프레임별 총 비트수 (3단계 탐색)

는 SAD와 SSD를 사용하였다. 고정 양자화 파라미터를 사용한 VBR (Variable Bit Rate) 모드에서 다양한 비디오 시퀀스들을 테스트하였으며, DCT 영역에서 움직임 예측을 수행하는 경우에는 블록 \hat{f}_i 의 0이 아닌 모든 DCT 계수들을 사용하였다. 그림 5-(a), 5-(b)는 전역 탐색 기법에 의한 공간 영역에서의 움직임 예측과 변환 영역에서의 움직임 예측 결과를 나타내며, 그림 5-(c), 5-(d)는 3단계 탐색 기법에 의한 공간 영역에서의 움직임 예측과 변환 영역에서의 움직임 예측 결과를 나타낸다. 그림 5에서의 성능 비교처럼, SAD_{tr} 를 사용한 변환 영역에서의 움직임 예측 기법이 SSD_{tr} 를 사용한 변환 영역에서의 움직임 예측 기법이나 SAD_{sp} 또는 SSD_{sp} 를 사용한 공간 영역에서의 움직임 예측 기법보다 더 좋은 PSNR과 압축률을 제공하였다. paris CIF

(352×288) 비디오 시퀀스를 사용하여 전역 탐색을 수행하였을 때에는 평균적으로 PSNR의 경우 0.20 dB 증가하였으며, 각 P-프레임별 총 비트수는 1022 비트 감소하였다. 동일 비디오 시퀀스를 사용한 3단계 탐색을 수행하였을 때에는 평균적으로 PSNR의 경우 0.17 dB 증가하였으며, 각 P-프레임별 총 비트수는 863 비트 감소하였다.

변환 영역에서의 SAD를 계산하기 위하여 사용되는 DCT 계수 \hat{f}_{pred} 의 수는 복원된 영상의 PSNR과 압축률에도 영향을 준다. 입력 블록의 DCT 계수와 움직임 예측된 블록의 DCT 계수간의 차이가 양자화되고, 지그재그 순서로 Run-Length 부호화되기 때문에, 계산상의 복잡도와 전력 소비를 줄이기 위하여 저주파에 속하는 몇 개의 DCT 계수들만을 이용하여 SAD를 계산할 수 있다. 이러한 경우, 압축

율이 낮아지는 단점이 있지만, 고주파대에 있는 DCT 계수들을 계산하기 위해 사용되는 PE와 MUL 블록들을 정지시킬 수 있는 장점이 있다. 그러므로, 식 (26)에 추가된 새로운 마스크 함수 $g(u,v)$ 는 전력 소모를 줄이기 위하여 그림 3과 같이 몇몇 PE와 MUL 블록들을 정지시키기 위해 효과적으로 사용될 수 있다.

$$SAD(\Delta x, \Delta y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} |\hat{f}_{curr}(u,v) - g(u,v) \times \hat{f}_{pred,\Delta x,\Delta y}(u,v)| \quad (26)$$

$$g(u,v) \in \{0,1\}$$

VII. 결론

본 논문에서는 주파수 영역에서의 움직임 예측 및 보상을 위한 새로운 알고리즘과 연산 아키텍처를 제안하였다. 제안된 움직임 예측 및 보상 알고리즘은 재귀 방정식을 이용하여 계산상의 복잡도를 크게 감소시켰으며, 파이프라인의 구조를 갖는 Array Processor에 의하여 높은 연산 처리율을 얻을 수 있었다. 또한, 제안된 구조는 공간 영역 방식과 비교하여 프레임 메모리의 크기와 메모리 액세스를 줄일 수 있으며, ROM으로부터 DCT 변환된 수직 변환 행렬과 수평 변환 행렬의 데이터를 읽어오기 위한 메모리 주소만 탐색 영역 위치에 따라 생성하면 되므로, 구조적인 추가 수정 없이 다양한 알고리즘들을 수행할 수 있다. 또한, 본 논문에서 유도된 재귀 방정식은 $\hat{f}_i(m,n)$ 의 개수를 조절하여 다른 주파수 대역을 갖는, 예를 들면 저주파 성분으로 구성된 영상부터 저주파와 고주파 성분으로 구성된 움직임 예측 및 보상된 영상을 효율적으로 생성할 수 있다. 이러한 특성은 압축 효율성과 연산량 간의 상호 trade-off 관계를 효율적으로 제어할 수 있는 방법을 제공한다. 다양한 특성을 갖는 비디오 시퀀스들을 이용한 모의실험을 통하여, SAD 기준을 적용한 변환 영역에서의 움직임 예측 및 보상 방식은 SAD 또는 SSD 기준을 이용한 공간영역에서의 움직임 예측 및 보상 방식보다 높은 PSNR과 압축률을 보여주었다.

참고 문헌

[1] P. Kuhn, *Algorithms, Complexity Analysis and VLSI Architectures for MPEG-4 Motion Estimation*, Kluwer Academic Publishers, Boston, 1999.

[2] V. Bhaskaran and K. Konstantinides, *Image and Video Compression Standards (Algorithms and Architectures)*, Kluwer Academic Publishers, Boston/Dordrecht/London, 1996.

[3] K. R. Rao and J. J. Hwang, *Techniques and Standards for Image, Video, and Audio Coding*, Prentice Hall, New Jersey, 1996.

[4] S. Cheng and H. Hang, "A Comparison of Block-Matching Algorithms Mapped to Systolic-Array Implementation," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 7, No. 5, pp. 741-757, Oct. 1997.

[5] J. Zan, M. O. Ahmad, and M. N. S. Swamy, "Pyramidal Motion Estimation Techniques Exploiting Intra-Level Motion Correlation," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 50, No. 2, pp. 83-93, Feb. 2003.

[6] Y. Keller and A. Averbuch, "Fast Gradient Methods Based on Global Motion Estimation for Video Compression," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 4, pp. 300-309, Apr. 2003.

[7] C. Du, Y. He, and J. Zheng, "PPHPS: A Parabolic Prediction-Based, Fast Half-Pixel Search Algorithm for Very Low Bit-Rate Moving-Picture Coding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 6, pp. 514-518, Jun. 2003.

[8] J. Zan, M. O. Ahmad, and M. N. S. Swamy, "Comparison of wavelets for multiresolution motion estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 16, No. 3, pp. 439-446, Mar. 2006.

[9] N. Merhav and V. Bhaskaran, "A fast algorithm for DCT-domain inverse motion compensation," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 4, pp. 2307-2310, May 1996.

[10] J. Song and B. Yeo, "A Fast Algorithm for DCT-Domain Inverse Motion Compensation Based on Shared Information in a Macroblock," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 10, No. 5, pp. 767-775, Aug. 2000.

[11] R. H. J. M. Plompen, B. F. Schuurink, and J. Biemond, "A New Motion-Compensated Transform Coding Scheme," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, Vol. 1, pp. 371-374, 1985.

[12] ITU-T Recommendation H.263, *Video coding for low bit rate communication*, 1998.

이 주 흥 (Jooheung Lee)

정회원



1996년 2월 인하대학교 전자공학과 졸업

1998년 2월 한양대학교 전자통신공학과 석사

2006년 8월 Pennsylvania State University 전기공학과 박사

2006년 8월~현재 University of

Central Florida, Electrical Engineering and Computer Science Department, 조교수

<관심분야> Signal processing, multimedia systems, power-aware and reliable VLSI systems design

류 철 (Chul Ryu)

정회원



1989년 3월 미국 Florida Institute of Technology 컴퓨터 공학과 졸업

1991년 3월 미국 Polytechnic University 전자공학과 석사

1997년 10월 미국 Polytechnic University 전자공학과 박사

1999년 3월~현재 동국대학교 정보통신공학과 부교수
<관심분야> Multimedia and visual data communications, compression, encryption, watermarking