

방향성을 고려한 적응적 고속 움직임 추정 기법

*오창조의불 **이강준 ***정제창

한양대학교 전자통신컴퓨터공학과

*changburi@ece.hanyang.ac.kr

Fast Adaptive Motion Estimation Considering Directivity

*Oh, Chang-Jo-Ui-Bull **Lee, Gang-Joon ***Jeong, Je-Chang

Department of Electronics and Computer Engineering, Hanyang University

요약

본 논문에서는 영상 압축에 있어서 사용되는 새로운 고속 움직임 추정 기법에 대해 소개한다. 전역 탐색 블록 정합 알고리즘(FSBMA)은 최상의 PSNR을 갖는 화질로 움직임 벡터를 추정할 수 있지만 높은 계산량으로 인하여 실시간 구현에 부적합하다는 단점을 가지고 있다. 그러므로 계산량을 낮추면서 유사한 화질을 유지할 수 있는 많은 고속 탐색 기법들이 제안되어 왔다. 본 논문에서는 기존의 잘 알려진 고속 블록 정합 알고리즘을 수정 보완한 새로운 알고리즘을 제안한다. 제안한 방법에서는 움직임 추정에서의 고속 블록 정합 알고리즘에 있어 변형된 다이아몬드(Diamond) 탐색 기법을 이용하여 영상이 갖는 서로 다른 움직임 패턴에 대해 움직임의 방향성에 따라 적응적으로 탐색 방향과 패턴을 달리하면서 움직임을 예측하여 다이아몬드(Diamond)탐색 알고리즘과 유사한 화질을 유지하면서 보다 적은 계산량을 가지고 움직임 벡터를 추정할 수 있다. 또한 이를 PMVFAST(Predictive Motion Vector Field Adaptive Search Technique)와 결합함으로써 보다 좋은 화질을 가질 수 있다.

1. 서론

고용량 고화질 동영상의 빠르고 효율적인 전송을 위해 동영상 데이터의 압축은 반드시 필요한 기술이다. 이를 위해 사용되어 지는 움직임 추정 기술은 동영상을 고효율로 압축할 수 있는 가장 중요한 기술의 하나이다. 시간 축으로 연속된 픽처들은 주로 화면의 원점 부분에서 움직임 벡터를 가지고 있기 때문에 움직임 보상 방법에 있어서는 영상의 이러한 성질을 이용하여 시간 축 상에 존재하는 데이터의 중복성을 제거하여 압축 효율을 높인다. 이렇듯 시간 축에 존재하는 이전 참조 픽처와 현재 픽처와의 가장 유사한 블록을 탐색하는 단계를 움직임 예측(motion estimation)이라고 하며 이동한 변위를 나타내는 것을 움직임 벡터(motion vector)라고 한다.

움직임 추정에는 블록 정합 방식(BMA: Block Matching Algorithm)이 사용되어진다. 가장 일반적인 블록 정합 방식에는 전역 탐색 기법 FSBMA(Full Search Block Matching Algorithm)가 있으며 이 기법은 현재 프레임의 기준 블록과 이전 프레임의 동일 위치에 대한 탐색 범위 내에서 정합 오류를 계산하여 최소의 오류를 갖는 블록에 해당하는 움직임 벡터를 찾아내는 것이다. 이 방법은 탐색 범위의 모든 좌표에 대한 SAD(Sum of Absolute Difference) 값을 계산해야 하는 것이므로 많은 계산량을 필요로 하며 실시간 압축에 있어 어려움이 따른다. 이를 해결하기 위해 NTSS(New TSS), DS(Diamond Search), HEXBS(Hexagon-based Search), PMVFAST(Predictive Motion Vector Field Adaptive Search Technique) 등의 많은 고속 알고리즘 방식이 제안되어 왔다. [1-4]

※ 본 논문은 정보통신부의 출연금으로 수행한 IT SoC 핵심설계인력양성사업의 수행결과입니다.

본 논문에서 제안한 고속 움직임 추정 기법은 탐색점수(SP: Search Point) 대비 복원된 영상과 원영상과의 화질 차이를 비교하는 PSNR(Peak Signal to Noise Ratio)에 있어 우수한 방식으로 평가되는 다이아몬드 탐색 기법(DS: Diamond Search)을 기본으로 하여 개선하였으며, 또한 장기간에서 말의 움직임을 따서 각 모드에 이름을 붙였다. 제안한 기법은 다이아몬드 탐색 기법과 비교하여 동일한 화질을 유지하면서 보다 적은 탐색점수를 갖으며 또한 탐색 범위를 임의로 설정함에 따라 보다 좋은 PSNR을 갖도록 조절 가능한 방식을 제안하였다. 또한 이 방식을 개선된 PMVFAST에 적용시킬 경우 보다 좋은 PSNR을 갖을 수 있다.

제안한 방식은 방향성을 갖는 움직임이 큰 영상의 경우에 있어 보다 좋은 화질 향상의 결과를 가지며 움직임이 적은 영상의 경우에 있어서는 보다 적은 탐색 점수로 움직임 추정을 함으로서 계산량을 상당수 감소시킬 수 있다. 실험 결과 제안된 방식은 다이아몬드 탐색 기법과 비교하여 동일한 화질을 유지하면서 보다 적은 탐색점수를 갖으며 또한 탐색 범위가 커질수록 다이아몬드 탐색 기법보다 좋은 성능을 갖음을 알 수 있다.

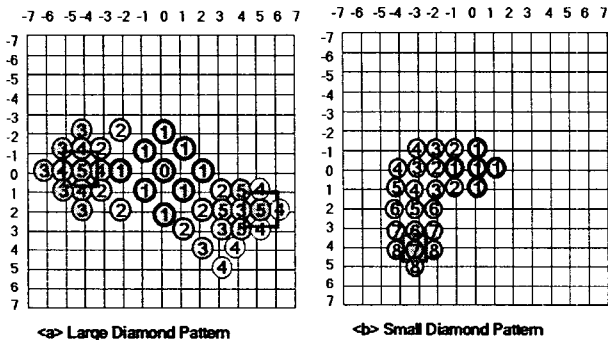
2. 기존의 움직임 추정 방식

움직임 추정방식 중 가장 일반적이고 기본적인 형태의 블록 정합 기법은 전역 탐색 기법 FS(Full Search)이 있다. 전역 탐색 기법의 경우 블록 정합에 대한 가장 정확한 움직임 벡터 값을 구할 수 있는 방식이지만, 현재 프레임의 기준 블록에 대한 참조 프레임의 탐색 범위 내의 모든 영역에 대한 탐색이 이루어지므로 계산량이 커지고 실시간 처

리에는 적합하지 않다. 그러므로 여러 다양한 방식의 고속 움직임 추정 기법이 제안되어 왔다. 그 중 대표적인 기법으로서 NTSS (New TSS), DS (Diamond Search), HEXBS (Hexagon-Based Search), PMVFAST(Predictive Motion Vector Field Adaptive Search Technique) 등을 들 수 있다.[1-4]

이 중 가장 대표적인 기법 중 하나인 다이아몬드 탐색 기법 (DS)의 경우 고속 탐색 기법의 성능을 측정하는 척도로서 사용되는 원영상과 움직임 추정에 의해 재구성되어 복원된 영상과의 픽셀값의 차이를 통해 화질 차이를 비교하는 PSNR(Peak Signal to Noise Ratio)과 최소 SAD(Sum of Absolute Difference) 값을 구하기 위해 소요되는 탐색 점수를 나타내는 SP(Search Point)의 대비에 있어서 우수한 성능을 나타낸다고 볼 수 있다.

우선 다이아몬드 탐색 기법에 대해 살펴보면, <그림1>의 (a),(b)에서 Large Diamond 패턴과 Small Diamond 패턴을 들 수 있다. 시작 단계에 있어서 원점을 중심으로 이웃하는 주변 8개의 점에 대하여 기본적인 Large Diamond 형태의 탐색을 수행한다. 그리고 원점이 최소 SAD값이 아니고 원점에서 직각 방향의 가장자리의 점이 최소값을 지닐 경우 동일 방향으로 확장해서 5점의 검색을 더 수행하게 된다. 그리고 만약 가장자리가 아닌 점이 최소값일 경우 동일 방향으로 3점을 더 확장해서 탐색을 수행하게 된다. 그리고 3점 혹은 5점을 확장해서 탐색을 수행하였을 시 최소 SAD값에 변화가 없을 경우 중심점을 해당 블록의 움직임 벡터 값으로 정하고 마지막으로 <그림1, b>의 Small Diamond 패턴을 한번 더 검색한 후 탐색을 종료하게 된다.



<그림1> 다이아몬드 탐색 기법

또한 육각 패턴을 이용하는 HEXBS (Hexagon-based Search) 탐색 기법이 있으며 이는 움직임이 주로 원점 주변에 있는 성질을 이용하여 육각 패턴으로 탐색을 하여 움직임 벡터를 추정하게 된다. 이 탐색 기법의 경우 다이아몬드 탐색 기법보다 다소 낮은 화질을 가지지만 더 적은 탐색점수를 가지고 고속으로 탐색을 수행하게 된다. [2]

3. 제안한 탐색 패턴

제안하는 방식은 네 가지의 모드로 분류할 수 있다. <그림2>

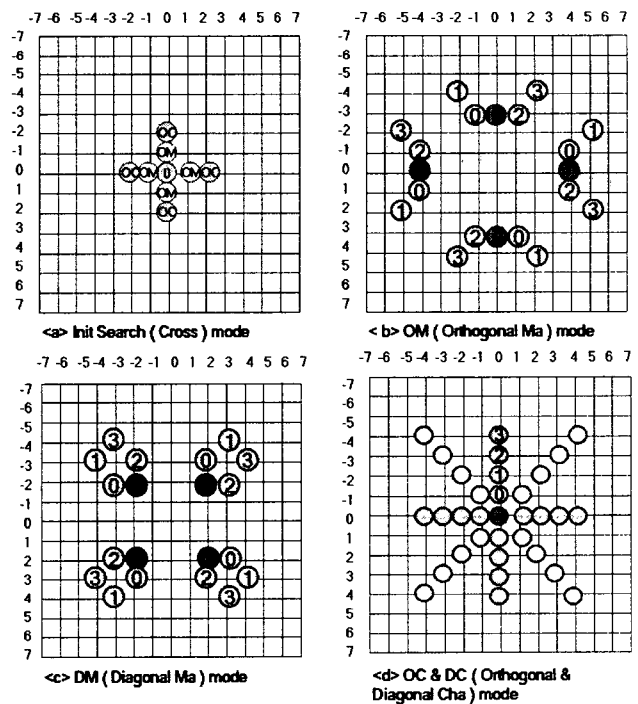
첫 번째 모드는 초기 모드(Init Search)로서 기본적으로 십자 형태(Cross Pattern)로 탐색을 시작하게 된다. 이 중 <그림2, a>에서 표기된 십자 형태의 중간 지점들에 해당하는 OM 표기된 점이 최소 SAD값을 지닐 경우 다음에서 설명하는 두 번째 모드인 OM모드로 탐색을 수행하게 되며, OC 표기된 점이 최소 SAD값을 지닐 경우 네 번째 모드인 OC 모드로 탐색을 수행하게 된다. 그리고 원점이 최소값을 지닐 경우 원점을 움직임 벡터로 지정하고 탐색을 종료하게 된다.

두 번째 모드는 직교 마(OM: Orthogonal Ma) 모드로서 <그림2, b>에 나타내었다. 장기에서 마의 움직임 방향과 유사성을 두고 있으며

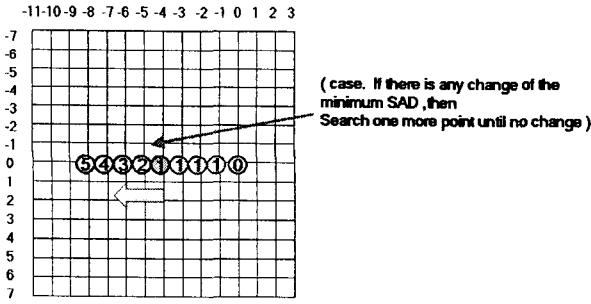
이 모드의 경우 다이아몬드 탐색 기법과 비교하여 보다 적은 움직임 값을 갖는 영상에 대하여 보다 적은 탐색점수를 가지기 위해 고안되었다. 만약 최소 SAD값이 <그림2, b>에 표기된 짝수 번호의 인덱스(0,2)에 해당될 경우 DM 모드를 수행하게 되고 만약 홀수 인덱스(1,3)에 해당될 경우 해당 변위의 OM 모드를 수행함으로써 확장된 범위의 움직임 탐색을 수행하게 된다.

세 번째 모드는 대각 마 (DM: Diagonal Ma) 모드로서 <그림2, c>에 나타내었다. 역시 장기에서 마의 움직임 방향과 유사성을 두고 있으며 대각 방향을 갖는 중간 정도의 움직임을 갖는 영상을 위해 고안하였다. 만약 짝수 인덱스(0,2)가 최소SAD값을 지닐 경우 선택된 지점에 대해 왼쪽 방향(0번 인덱스의 경우) 혹은 오른쪽 방향(2번 인덱스의 경우)으로 90도 변위된 DM 모드를 한 번 더 수행하게 된다. 그리고 그 중 최소의 SAD값을 갖는 지점을 움직임 벡터로 지정하게 된다. 그리고 만약 홀수 인덱스(1,3)가 최소SAD값을 가지게 될 경우는 동일 변위를 갖는 DC 모드를 수행하게 된다.

네 번째 모드는 직교와 대각 차 (OC & DC: Orthogonal and Diagonal Cha) 모드로서 <그림2, d>에 나타나 있으며 역시 장기에서 차의 움직임을 본 따서 만들었다. 그리고 <그림3>에서 이 모드의 탐색 방식에 대해 설명하였다. 만약 OC 혹은 DC 모드가 선택되었을 경우 설정된 탐색 범위내의 점의 최소 SAD값 변화 여부를 검사하고 만약 탐색범위 내에서 최소SAD값의 변화가 발생할 경우 동일 방향으로 최소 SAD값의 변화가 없을 때까지 한 점씩 더 확장해서 탐색한다. 이는 직진성을 갖는 움직임이 큰 영상에서 보다 더 정확한 최소 SAD지점을 찾기 위해 고안되었으며 그림에서 표기된 탐색 범위는 4로 선택되어 있으나 이는 선택에 따라 보다 큰 값으로 늘릴 수 있다. 그리고 이 모드에서의 탐색 범위를 증가시키에 따라서 증가된 PSNR 값을 얻을 수 있으며 보다 높은 해상도의 영상이거나 혹은 직진성을 갖는 움직임이 많은 영상에 있어서 다이아몬드 탐색 기법보다 더 향상된 PSNR을 가질 수 있다.



<그림2> 제안한 방식의 네 가지 모드



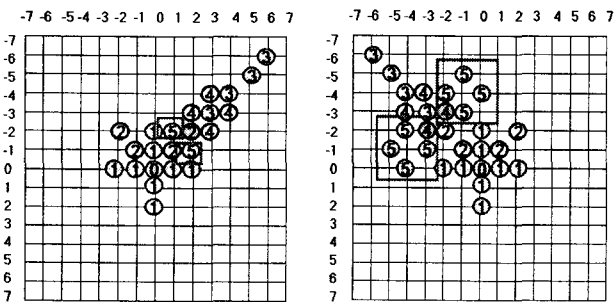
<그림 3> OC 와 DC 모드의 탐색 패턴

<그림4>는 제안한 방법의 탐색 과정을 보인다

<그림4, a>의 경우는 초기 (Init Search) 단계에서 OM (Orthogonal Ma) 모드가 선택이 되고 그 중 홀수 인덱스 (그림 2의 b에서 1,3 으로 표기된 점)가 최소 SAD값을 지닐 경우 원점과 인덱스 점간의 대각 방향으로 확장해서 DC (Diagonal Cha) 모드를 수행하여 정해진 범위만큼의 탐색을 수행하게 된다. 그리고 탐색 범위 이내에 최소 SAD값이 발견되지 않을 경우 홀수 인덱스를 갖은 점에서 역시 같은 대각 방향으로 DM (Diagonal Ma) 모드를 선택하고 탐색을 수행하게 된다. 그리고 이 모드에서도 최소 SAD값이 발견되지 않을 경우, 인근의 두 점을 더 탐색하고 그 중 최소 SAD값을 선택하게 된다.

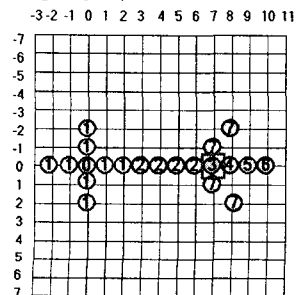
그리고 만약 위의 DM모드 단계에서 짝수 인덱스 (0,2)가 선택되었을 경우의 예는 <그림4, b>에서 나타내었다. 이 경우 위의 각 모드에 대한 설명에서도 말했듯이 역방향으로 90도 변위된 DM 모드를 선택하여 한 번 더 탐색을 수행하고 그 중 최소 SAD값을 움직임 벡터로 지정하게 된다. 또한 홀수 인덱스 (1,3)이 선택되었을 경우는 다시 시작점에서 이전 단계의 모드에 따라서 OC 혹은 DC모드를 수행하여 계속 탐색을 수행하게 된다.

그리고 <그림4, c>의 경우는 초기 (Init Search) 단계의 가장자리의 네 점 혹은 OM 모드 이후 최소 SAD값의 변화가 없을 경우 혹은 그림에 표시된 예에서 OC 모드 이후 최소SAD값의 변화가 없을 때까지 한 점씩 계속 확장한 이후 다시 OM 모드를 수행하고 이 경우에도 변화가 없을 경우 그 점을 움직임 벡터로 지정하게 되고 변화가 있을 경우 다시 상기의 방법대로 탐색을 수행하게 된다.



<a> Break Condition 1 (There is no change after DM, more search two neighbor point)

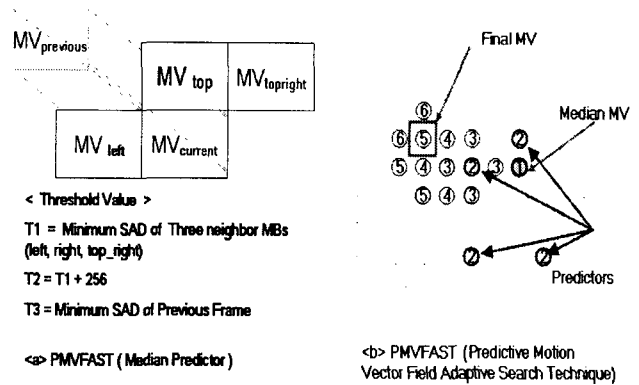
 Break Condition 2 (There is no change after DM of two iteration)



<c> Break Condition 3 (There is no change after OM)

<그림4> 제안한 방법의 탐색 과정

4. 개선된 PMVFAST와의 결합



<그림5> PMVFAST (Predictive Motion Vector Field Adaptive Search Technique)

기존의 제안된 방식 중 가장 우수한 화질과 적은 탐색점수를 가지는 블록 정합 방식의 하나로서 PMVFAST(Predictive Motion Vector Field Adaptive Search Technique)를 들 수 있다.[3-4] <그림5>에서 나타내었듯이 이는 원점을 예측기로서 움직임 벡터를 구하는 방식이 아닌 인접한 움직임 벡터들의 값을 이용하여 중간값을 구하고 이 값을 예측값으로 지정하여 정해진 임계값과 주어진 5개의 예측기 (즉 좌측, 상단, 우측, 상단의 움직임 벡터와 원점, 그리고 이전 프레임의 움직임 벡터 값) 과의 정해진 조건을 만족하는 경우 바로 움직임 벡터를 정하거나 혹은 다이아몬드 탐색 기법을 이용하여 블록 정합을 하는 방식이다. 이 방식의 주요 알고리즘에 대해서는 <그림 6>에 설명을 하였다

step1) Calculate Threshold and Median MV (Median Rule)

Break if Median SAD < T1
 Break if Median MV = MV-1 and SAD of Median MV < T3

step2> Check All Predictors
 [(0,0), MVleft, MVtop, MVtopright, MVprevious frame]
 Best Predictor (a vector with minimum SAD among 5 predictors)

Break if Best MV = MV-1 and minimum < T3
 Break if minimum SAD < T1

Step3> Perform Diamond Search

Perform Large Diamond Search if Median MV = (0,0)
 and MV left = MV top = MV topright and T2>1536
 Perform Small Diamond Search only one time
 if MV left = MV top = MV topright = MV-1
 Otherwise Perform Small Diamond Search

<그림 6> PMVFAST 알고리즘

제안하고자 하는 방식은 위 PMVFAST의 알고리즘 중 단계 1의 임계치 설정 부분과 단계 2 부분의 중간값 구하는 부분을 수정하는 것이다. 또한 PMVFAST에 쓰이는 다이아몬드 탐색 기법 대신에 상기에 제안한 알고리즘으로 대체함으로써 화질에 있어 보다 우위를 가질 수 있도록 한다.

이 PMVFAST 알고리즘의 수정 부분에 대해서는 <그림7>에서 나타내었다. 단계 1에서의 임계치 부분을 그림에 설명된 것과 같이 변경하고 중간값 규칙을 적용시켜서 Median Vector를 구하는데 있어 인접한 세 개의 움직임 벡터 (즉 MVleft, MVtop, MVtopright)에 의해 구해진 중간값을 단계 2에서 원점 (0,0) 그리고 이전 프레임의 움직임 벡터들과 중간값 규칙을 적용하여 중간값을 구한다. 그리고 이로부터 얻어진 중간값을 다시 좌측과 우측 상단의 움직임 벡터(MVleft, MVtopright) 값들 사이에서 중간값을 취함으로써 최종 중간값을 구한다. 그리고 이 값으로부터 다이아몬드 탐색 기법 대신 제안한 탐색 기법을 적용하여 최종 움직임 벡터 값을 구하게 된다.

step1>
Median SAD < T1 → Median SAD < (T3 + T2 + T1) / 3 - 256

step2>
Median MV1 = [MV left, MV top, MV topright]
Median MV2 = [0, Median MV1, Previous MV]
Median MV3 = [MV left, Median MV2, MV topright]

step3> When Proposed PMVFAST is used, alternate diamond search to Proposed Search

<그림 7> 제안한 PMVFAST에서 변형된 부분

5. 실험 결과

실험에 사용된 영상은 CIF (352 X 288) 테스트 영상으로서 stefan, coastguard, mobile, akiyo, container (각300프레임), hall_monitor(330프레임), waterfall, tempete(각260프레임) bus(150프레임), flower1(250프레임), flower2(246프레임), bridge_close(2000프레임)이 사용되었다. 그리고 매크로블록(MB: Macro Block)의 크기는 16 X 16이 사용되었으며 또한 탐색 범위는 표1)의 기존의 BMA 방식들인 FS(Full Search), DS(Diamond Search), HEXBS(Hexagon-based Search)와 표2)의 제안된 방식1 의 경우 X축, Y축 각각에 대하여 [-16,15]의 조건으로 하였다. 그리고 표2)의 제안된 방식2, 표3)의 실험에서는 정해진 탐색 범위를 주지 않았다. 그리고 제안된 방식에서 CR의 값은 OC & DC모드의 탐색 범위를 나타낸다.

실험은 기존의 고속 움직임 추정 방식 중 전역 탐색 기법(FS)과 다이아몬드탐색 기법(DS), 그리고 HEXBS (Hexagon-based Search)와의 비교를 통한 실험 결과를 나열하였다. 표에서 음영처리 된 부분은 제안한 방식이 PSNR이 더 높거나 혹은 탐색점수(SP)가 더 적은 경우를 말한다. 그리고 굵게 표기된 숫자는 HEXBS 보다도 제안한 방식의 탐색점수가 적을 경우를 나타낸다.

실험 결과에서 보면 알 수 있듯이 표2) 에서 제안한 방식은 다이아몬드 탐색 기법과 비교하여 OC와 DC모드의 검색 범위가 7일 경우 거의 유사한 PSNR 값을 갖는 것을 볼 수 있다. 반면 탐색 점수(SP)는 매크로 블록 당 평균 1점에서 4점 까지 감소하는 것을 볼 수 있다. 그리고 일부 시퀀스에 대해서는 고속 알고리즘인 HEXBS보다 더 좋은 성능을 나타내는 것을 볼 수 있다. 그리고 검색 범위가 10으로 확장된 경우 더 좋은 PSNR을 가질 수 있다.

그리고 표3)의 경우에는 PMVFAST와 개선된 PMVFAST (Changed PMVFAST(DS))와 이 개선된 PMVFAST에서 다이아몬드 기법 대신 상기에서 제안한 탐색 기법으로 변경하였을 경우의 실험 결과를 나타낸다. 결과에서 알 수 있듯이 영상이 직진성을 가지고 움직임을 갖을 경우 0.1에서 0.7dB 향상된 PSNR을 갖음을 알 수 있다.

	Full Search		Diamond Search		HEXBS	
	PSNR	SP(MB)	PSNR	SP(MB)	PSNR	SP(MB)
STEFAN	24.64	299.83	22.55	19.59	22.48	15.52
COASTGUARD	30.48	299.83	30.37	17.08	30.33	13.73
BUS	25.07	299.83	22.26	20.81	22.14	16.40
HALL_MONITOR	34.89	299.83	34.84	13.27	34.76	11.21
MOBILE	24.60	299.83	24.51	13.74	24.53	11.52
TEMPETE	26.88	299.83	26.60	13.83	26.74	11.65
WATERFALL	34.57	299.83	34.57	12.66	34.55	11.00
CONTAINER	38.33	299.83	38.23	12.80	38.24	11.14
AKIYO	42.94	299.83	42.93	12.67	42.59	11.03
FLOWER1	25.74	299.83	25.61	16.54	25.32	13.43
FLOWER2	25.07	299.83	25.04	14.71	25.00	12.23
BRIDGE_CLOSE	35.53	299.83	35.53	12.68	35.53	11.06

<표1 : 기존의 BMA 방식들 비교 >

	Diamond Search		Proposed1, CR=7		Proposed2, CR=10	
	PSNR	SP(MB)	PSNR	SP(MB)	PSNR	SP(MB)
STEFAN	22.55	19.59	22.57	18.74	22.66	21.24
COASTGUARD	30.37	17.08	30.37	17.33	30.40	19.38
BUS	22.26	20.81	22.27	21.26	22.40	24.01
HALL_MONITOR	34.84	13.27	34.82	10.17	34.82	10.63
MOBILE	24.51	13.74	24.51	12.04	24.80	12.51
TEMPETE	26.60	13.83	26.58	11.15	26.76	11.75
WATERFALL	34.57	12.66	34.56	9.25	34.56	9.47
CONTAINER	38.23	12.80	38.24	9.17	38.24	9.51
AKIYO	42.93	12.67	42.90	8.97	42.90	9.18
FLOWER1	25.61	16.54	25.60	16.00	25.63	17.58
FLOWER2	25.04	14.71	25.05	13.01	25.05	13.78
BRIDGE_CLOSE	35.53	12.68	35.53	8.90	35.53	9.12

<표2 : 제안된 방식과 다이아몬드 탐색 기법과의 비교>

	PMVFAST		Changed PMVFAST(DS)		Proposed PMVFAST CR=16	
	PSNR	SP(MB)	PSNR	SP(MB)	PSNR	SP(MB)
STEFAN	23.57	6.77	23.58	7.07	24.51	10.16
BUS	22.23	5.81	22.55	6.43	22.98	10.17
FLOWER1	25.27	2.91	25.66	3.15	25.68	5.38

<표3 : 개선된 PMVFAST 방식과 제안된 PMVFAST 방식 비교>

6. 결론

본 논문에서는 영상의 압축 효율을 높이는 움직임 추정 기법에 있어서 기존에 제안된 BMA 방식보다 좋은 화질을 유지하면서 보다 고속으로 전송하는 방식을 제안하였다. 다이아몬드 탐색 기법보다 비슷한 화질을 유지하면서 탐색 점수를 줄이므로 보다 고속으로 움직임 추정이 가능하게 하며 또한 탐색 범위를 임의로 설정할 수 있게 함으로써 영상의 화질과 전송 속도 사이의 비교우위에 따라 화질과 탐색 속도 조절을 가능하게 하였다. 또한 가장 고속 알고리즘으로 볼 수 있는 PMVFAST를 개선하고 이 PMVFAST에 제안한 방식을 삽입함으로써 보다 좋은 화질을 얻을 수 있다. 그러나 제안한 알고리즘을 PMVFAST와의 접목시킬 경우에 보다 좋은 화질을 위해 탐색 범위를 설정할 시 발생할 수 있는 탐색점수의 증가를 줄일 수 있는 더욱 개선된 고속 탐색 기법의 연구가 필요하다.

7. 참고 문헌

- [1] Shan Zhu and Kai-Kuang Ma, "A New Diamond Search Algorithm For Fast Block Matching Motion Estimation." IEEE Trans. Image Processing. vol.9. no.2. February 2000
- [2] Ce Zhu, Xiao Lin and Lap-Pui Chau, "Hexagon-Based Search Pattern for Fast Block Motion Estimation," IEEE Trans. On Circuits and System for Video Tech. Vol. 12, No. 5, May 2002.
- [3] Alexis Michael Tourapis, Oscar C. Au, Ming L. Liou, "Predictive Motion Vector Field Adaptive Search Technique (PMVFAST) Enhancing Block Based Motion Estimation," Proceedings of Visual Communications and Image Processing 2001 (VCIP'01)
- [4] International Organisation For Standardisation, "Coding of Moving Pictures and Audio", ISO/IEC JTC1/SC29/WG11 N3675, La Baule, October 2000

SESSION 4

영상부호화 III 및 음성신호처리

15:40~17:00

좌장 : 서 정 일 (ETRI)

Session 4-1

Dependent Quantization for Scalable Video Coding

Danu Pranantha, 김문철, *함상진, *이근식, *박근수 (한국정보통신대학교, *한국방송)

Session 4-2

감성적 화질 개선을 위한 영상의 블록현상 제거 기법

이상우, 박상주 (홍익대학교)

Session 4-3

MP3의 음질 향상을 위한 고주파 복원

허소영, 김인철 (서울시립대학교)

Session 4-4

가상 음향 공간 구현에 관한 연구

윤재연, *박준선, **김충석, ***진용욱 (통일부, *방송위원회, **KT, ***경희대학교)