

---

# 움직임 방향 치우침 특성을 이용한 십자형 다이아몬드 탐색

류 권 열\*

Cross Diamond Search Using Motion Direction Biased Characteristics

Kwon-yeol Ryu\*

## 요 약

본 논문에서는 MVP 분포의 움직임 방향 치우침 특성을 이용한 방향성 탐색 패턴을 디자인하고, 탐색점의 이동 방향에 따라 탐색 패턴을 적응적으로 변화시키는 방향 적응적 십자형 다이아몬드 탐색 방법을 제안 한다. 제안한 방법은 이웃 매크로 블록으로부터 모션 벡터를 예측하고, 예측한 모션 벡터를 이용하여 초기 움직임 방향을 설정한다. 탐색 과정에서 BMP의 이동 방향에 따라 제안한 탐색 패턴을 적응적으로 변화시킴으로써 탐색 효율을 향상시킨다. 실험 결과, 제안한 방법은 움직임 추정에 소요되는 계산량을 기존의 십자형 다이아몬드 탐색에 비해 0.43% ~ 1.76% 감소시킴으로써 고속 움직임 추정이 가능함을 알 수 있었다.

## ABSTRACT

In this paper, we design directional search pattern using motion direction biased characteristics of MVP distribution, and proposes a direction applied cross diamond search method that adaptively change search pattern according to moving direction of search point. Proposed method predict motion vectors from neighbor macro blocks, and define initial motion direction by using predicted motion vectors. It improve search efficiency by using alternately proposed search pattern according to motion direction of BMP in search process. The simulation results show that proposed method is able to fast motion estimation compared with conventional cross diamond search, according as it reduce computational complexity that is required of motion estimation with 0.43% ~ 1.76%.

## 키워드

block matching algorithm, motion vector, motion estimation, directional search pattern

---

\* 위덕대학교 소프트웨어공학과

접수일자 2009. 02. 18  
심사완료일자 2009. 04. 24

## I. 서 론

동영상은 연속한 프레임간의 높은 상관성으로 인하여 많은 시간적 중복성을 지니고 있다. 따라서 연속된 프레임간의 움직임 정보를 찾고, 이를 이용하여 시간적 중복성을 효과적으로 제거함으로써 동영상 부호화 성능을 향상시킨다. 시간적 중복성 제거에 이용되는 움직임 추정은 많은 계산량이 요구되기 때문에 실시간 부호화를 위해서는 움직임 추정에 소요되는 계산량 감소가 필수적이다. 움직임 추정에 소요되는 계산량을 감소시키는 방법으로 블록 매칭 알고리듬(BMA: block matching algorithms)이 제안되었으며, BMA는 계산량이 적고, 하드웨어 구현이 용이하기 때문에 H.261, H.263, MPEG-1, MPEG-2, MPEG4/AVC 등과 같은 동영상 표준에 적용되고 있다[1,2].

BMA는 입력 영상을 매크로 블록(MB: macro block) 단위로 분할하고, 현재 프레임의 매크로 블록과 블록 정합 오차(BMD: block matching distortion)가 최소인 탐색 점(BMP: best matching point)을 이전 프레임의 탐색 영역에서 찾은 후, 두 블록간의 움직임 정보를 이용하여 모션 벡터(MV: motion vector)를 설정한다.

설정한 모션 벡터를 이용하여 움직임 보상 영상을 생성하며, 움직임 보상 영상과 현재 프레임의 차이를 부호화 한다. BMA를 기반으로 하는 연구로는 전역 탐색(FS: full search), 새로운 3단계 탐색(NTSS: new three step search), 4단계 탐색(FSS: four step search), 다이아몬드 탐색(DS: diamond search), 십자형 다이아몬드 탐색(CDS: cross diamond search) 등과 같이 다양한 형태의 탐색 패턴을 이용하여 최적의 매칭 블록을 찾는 방법들이 제안되었다[3-7]. FS는 탐색 영역 내의 모든 블록을 대상으로 탐색하므로 움직임 벡터를 정확하게 찾을 수 있으나, 많은 계산량이 요구된다. NTSS는 움직임이 적은 영상에서는 3단계 탐색보다 효율적이지만, 모든 영상에서 고정된 탐색 패턴을 적용하기 때문에 움직임이 큰 영상에서 국부적 최적점에 빠지는 단점이 있다.

FSS는 국부적 최적점에 빠지는 단점을 보완하기 위해 제안되었으나, 2회소 간격의 9개 탐색점으로 탐색 패턴을 구성하여 4단계로 진행하므로 많은 계산량이 요구된다. DS는 탐색 영역의 중심에 모션 벡터가 위치할 확률(MVP: motion vector probability)이 높다는 가정을 이

용한 방법으로 화질과 계산량에서 좋은 성능을 보이지만, 움직임이 빠른 영상에서는 NTSS보다 비효율적이다. CDS는 움직임이 적은 영상에서는 DS보다 좋은 성능을 나타내지만, 움직임 방향에 따른 MVP 분포 특성을 고려하지 않았고, 초기 탐색점 수가 9개이므로 모션 벡터가 탐색 영역의 중심에서 반경  $\pm 2 \sim \pm 3$ 에 위치할 경우 비효율적인 면이 있다.

본 논문에서는 영상의 움직임 방향에 따른 MVP 분포 특성을 분석하고, 움직임 방향 특성에 적합한 탐색 패턴을 설정하며, 탐색 과정에서 BMP의 이동 방향에 따라 탐색 패턴을 적응적으로 변화시키는 방향 적응적 십자형 다이아몬드 탐색 방법을 제안한다. 본 논문의 구성은 2장에서 움직임 방향에 따른 MVP 분포 특성을 나타내고, 3장에서 제안한 방법에 대한 탐색 패턴과 탐색 과정을 나타내며, 4장 및 5장에서 실험 결과 및 결론을 나타낸다.

## II. 움직임 방향 치우침 특성

동영상 부호화 과정에서 움직임 추정에 적용되는 탐색 패턴의 모양과 크기는 계산량 감소의 중요한 요인이 되기 때문에 고속 움직임 추정을 위해서는 탐색 영역의 MVP 분포 특성에 가장 적합한 탐색 패턴을 사용하여야 한다. NTSS, DS, CDS는 MVP 분포 특성 중에서 중심 치우침(center biased) 특성을 이용한 방법으로 탐색 패턴의 모양을 적게 하고, 탐색 시작점을 탐색 영역의 중심으로 설정하여 고속 움직임 추정을 가능하게 하였으나, 움직임 방향 치우침(motion directional biased) 특성은 고려하지 않았다. 움직임 방향 치우침 특성은 두 가지로 구분할 수 있으며, 첫째는 객체의 움직임 방향에 따른 특성을 나타내는 것으로 사람, 기차, 자동차처럼 수평 방향으로 움직이는 경우와 탁구공, 농구공처럼 수직 방향으로 움직이는 경우에 따라 MVP 분포가 다르게 나타난다. 둘째는 탐색 과정에 적용하는 탐색 패턴의 모양과 탐색점의 이동 방향에 따라 탐색 효율이 다르게 나타나는 특성이다. 352x288 크기의 “salesman”, “football”, “tennis” 영상에 대해 FS를 이용하고, 탐색 영역을  $\pm 7$ 로 하였을 때의 평균 MVP 분포는 그림 1과 같다.

Radius (vert.)	Radius (hor.)							
	0	1	2	3	4	5	6	7
0	0.5439	0.0782	0.0133	0.0071	0.0045	0.0021	0.0062	0.0029
1	0.1571	0.0277	0.0140	0.0043	0.0020	0.0012	0.0007	0.0012
2	0.0440	0.0201	0.0084	0.0023	0.0014	0.0005	0.0010	0.0007
3	0.0373	0.0137	0.0092	0.0031	0.0012	0.0017	0.0009	0.0019
4	0.0086	0.0089	0.0029	0.0019	0.0124	0.0007	0.0017	0.0008
5	0.0069	0.0076	0.0012	0.0005	0.0004	0.0018	0.0012	0.0003
6	0.0041	0.0030	0.0010	0.0007	0.0006	0.0005	0.0089	0.0006
7	0.0073	0.0019	0.0021	0.0017	0.0012	0.0016	0.0009	0.0239

그림 1. 탐색 영역( $\pm 7$ )의 수평 & 수직방향 MVP 분포

Fig. 1 MVP distribution of horizontal & vertical direction in search window ( $\pm 7$ ).

그림 1에서와 같이 탐색 영역의 중심으로부터 수평 방향의 MVP 합은 11.43%, 수직 방향 MVP 합은 26.53% 이므로, 자연 영상의 MVP 분포에는 움직임 방향 치우침 특성이 존재함을 나타낸다. 그리고 탐색 영역의 중심 ( $r=0$ )에서 MVP 합이 54.39%이고, 중심으로부터 1픽셀 이내( $r=\pm 1$ )인 3x3 블록의 MVP 합이 80.69%이므로, 자연 영상의 MVP 분포에는 중심 치우침 특성이 있음을 알 수 있다. 탐색 영역 중심으로부터 반경 2픽셀( $r=\pm 2$ ) 이내의 MVP 분포는 그림 2와 같다.

그림 2에서 CDS의 CSP(cross search pattern) 영역에 대한 MVP 합을  $P_{csp}$ , 수평 움직임 방향 특성을 반영한 탐색 패턴 영역의 MVP 합을  $P_{hcsd}$ , 수직 움직임 방향 특성을 반영한 탐색 패턴 영역의 MVP 합을  $P_{vcsd}$ 로 표현하면,  $P_{csp}, P_{hcsd}, P_{vcsd}$ 는 식 (1), 식 (2) 및 식 (3)과 같다.

Radius (vert.)	Radius (hor.)				
	-2	-1	0	1	2
2	0.0025	0.0070	0.0062	0.0041	0.0016
1	0.0041	0.0073	0.0422	0.0062	0.0025
0	0.0081	0.0438	0.5439	0.0344	0.0052
-1	0.0039	0.0076	0.1149	0.0066	0.0035
-2	0.0018	0.0038	0.0378	0.0052	0.0025

그림 2. 중심에서 반경  $\pm 2$  픽셀 이내의 MVP 분포

Fig. 2 MVP distribution inside of  $\pm 2$  pixel from center.

$$\begin{aligned} P_{csp} &= MVP(-2,0) + MVP(-1,0) + MVP(0,0) \\ &\quad + MVP(1,0) + MVP(2,0) + MVP(0,2) \\ &\quad + MVP(0,1) + MVP(0,-1) + MVP(0,-2) \\ &= 0.8365 \end{aligned} \quad (1)$$

$$\begin{aligned} P_{hcsd} &= MVP(-2,0) + MVP(-1,0) + MVP(0,0) \\ &\quad + MVP(1,0) + MVP(2,0) + MVP(0,1) \\ &\quad + MVP(0,-1) \\ &= 0.7925 \end{aligned} \quad (2)$$

$$\begin{aligned} P_{vcsd} &= MVP(-1,0) + MVP(0,0) + MVP(1,0) \\ &\quad + MVP(0,2) + MVP(0,1) + MVP(0,-1) \\ &\quad + MVP(0,-2) \\ &= 0.8232 \end{aligned} \quad (3)$$

식 (1)에서  $P_{csp}$ 의 탐색점 수는 9개이고, MVP 합은 0.8365이며, 식 (2)에서  $P_{hcsd}$ 의 탐색점 수는 7이고 MVP 합은 0.7925이며, 식 (3)에서  $P_{vcsd}$ 의 탐색점 수는 7이고 MVP 합은 0.8232이다.  $P_{csp}$ 에 대한  $P_{hcsd}$ 의 비율은 94.74%이고,  $P_{vcsd}$ 의 비율은 98.41% 이므로 그림 2의 MVP 분포는 수직 방향 움직임 특성이 있음을 나타내며,  $P_{vcsd}$ 는  $P_{csp}$ 에 근접하는 MVP 합을 유지하면서 탐색점 수가 2개 감소한다. 따라서 영상의 움직임 방향 특성에 적합한 탐색 패턴을 설정하고, 움직임 방향에 따라 적응적으로 탐색 패턴을 변화시킴으로써 고속 움직임 추정이 가능함을 알 수 있다.

### III. 제안한 방법

#### 3.1 제안한 탐색 패턴

기존의 고속 탐색 방법인 NTSS, DS 및 CDS는 탐색 영역 내의 MVP 분포에 대한 중심 치우침 특성을 이용하여 고속 움직임 추정을 가능하게 하였으나, 움직임 방향 치우침 특성을 반영하지 않은 탐색 패턴을 사용함으로써 움직임이 크고 빠른 영상에서는 탐색 효율이 떨어지는 경우가 발생한다. 본 논문에서는 움직임 방향 치우침 특성을 반영한 탐색 패턴을 설정하고, 탐색 과정에서 BMP의 이동 방향에 따라 탐색 패턴을 적응적으로 변화시킴으로써 움직임 추정에 소요되는 계산량을 감소시키는 방법을 제안한다. 움직임 방향 치우침 특성을 적용한 탐색 패턴은 그림 3과 같다.

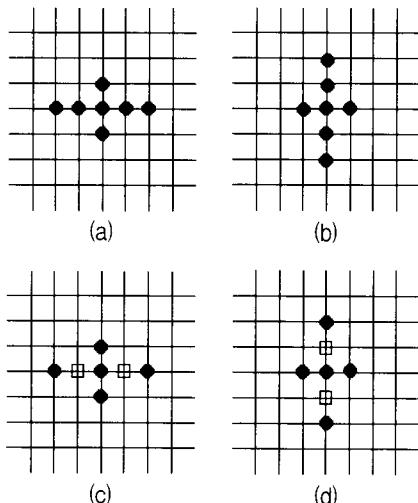


그림 3. 제안한 탐색 패턴  
 (a) HCSP (b) VCSP (c) HDSP (d) VDSP  
 Fig. 3 Proposed search pattern  
 (a) HCSP (b) VCSP (c) HDSP (d) VDSP

그림 3에서 HCSP(horizontal cross search pattern) 및 VCSP(vertical cross search pattern)는 수평 및 수직 방향 움직임 특성을 반영한 크로스 다이아몬드 탐색 패턴을 나타내고, HDSP (horizontal diamond search pattern) 및 VDSP (vertical diamond search pattern)는 수평 및 수직 방향 움직임 특성을 반영한 다이아몬드 탐색 패턴을 나타낸다.

HCSP와 VCSP는 제안한 방법의 첫 번째 단계에 적용되고, HDSP와 VDSP는 중간 탐색 단계에 적용되며, 탐색 과정에서 BMP의 이동 방향에 따라 탐색 패턴을 적응적으로 변화시킴으로써 움직임 추정에 소요되는 계산량을 감소시킨다.

첫 번째 단계의 탐색 효율  $e_1$ 은 식 (4)와 같다.

$$e_1 = \frac{p_1}{n_1} \quad (4)$$

식 (4)에서  $p_1$ 은 탐색 패턴 영역의 MVP 합이고,  $n_1$ 은 탐색 패턴을 구성하는 탐색 점의 수를 나타내며, 그림 2의 MVP 분포를 기준으로 계산한 탐색 효율은 표 1과 같다.

표 1. 탐색 효율 비교  
 Table. 1 Comparison of search efficiency

탐색 패턴	탐색 효율(%)
LDSP in DS	0.8642/9 = 9.69
CSP in CDS	0.8365/9 = 9.29
VCSP in proposed method	0.8232/7 = 11.76

표 1에서와 같이 제안한 탐색 패턴인 VCSP의 탐색 효율은 LDSP에 비해 4.77% 향상되고, CSP에 비해 2.47% 향상됨을 알 수 있다.

#### 3.2 제안한 탐색 과정

제안한 방법의 탐색 과정은 3단계로 구성된다. 1단계는 현재 매크로 블록의 모션 벡터를 이웃 매크로 블록으로부터 예측하여 탐색 영역을 설정하고[8], 예측한 모션 벡터를 이용하여 수평 수직 성분에 대한 위상각도  $\Theta$ 를 식 (5)와 같이 계산한다.

$$\Theta_i = \arctan \frac{V_i}{H_i} \quad (5)$$

식 (5)에서  $V_i$  및  $H_i$ 는 예측한 모션 벡터의 수직 및 수평 성분을 나타내며, 위상각도  $\Theta$ 가 45보다 적거나 같

으면 HCSP를 적용하여 탐색을 시작하고,  $\theta$ 가 45보다 크면 VCSP를 적용하여 탐색을 시작한다. 탐색 결과 BMP가 탐색 패턴의 중심에 위치하면 탐색을 멈추고, BMP를 이용하여 모션 벡터를 생성하며, 그리하지 않으면 2단계로 진행한다.

2단계는 현재 BMP 위치가 이전 단계 탐색 패턴 중심을 기준으로 수평 방향으로 이동했으면 HDSP를 적용하고, 수직 방향으로 이동했으면 VDSP를 적용하여 새로운 BMP를 탐색한다. 새로운 BMP가 탐색 패턴 중심에 위치하면 3단계로 진행하고, 아니면 2단계 과정을 반복한다.

3단계는 현재 BMP를 그림 3의 사각 표시(□) 탐색점 중에서 한 개와 비교하여 새로운 BMP를 탐색하고, 새로운 BMP의 위치 정보를 이용하여 모션 벡터를 생성한다. HDSP에서는 수평방향 탐색점 2개 중에서 BMD가 작은 쪽의 사각 표시 탐색점을 선택하고, VDSP에서는 수직 방향 탐색점 2개 중에서 BMD가 작은 쪽의 사각 표시 탐색점을 선택한다.

제안한 방법에서 움직임 추정에 소요되는 탐색점 수는 식(6)과 같다.

$$N_{\text{search}}(mv_x, mv_y) = \begin{cases} 7 & \text{If } MV = 0 \\ 7 + 3 \times n + 1 & \text{If } MV \neq 0 \end{cases} \quad (6)$$

식(6)에서  $(mv_x, mv_y)$ 는 모션 벡터의 위치를 나타내고,  $n$ 는 중간 단계를 나타내며, 마지막 단계에서 탐색점 수가 한 개 증가함을 나타낸다.

제안한 방법에서 BMD가 “0”인 이상적인 경우에 각 위치에 소요되는 탐색점 수는 그림 4와 같다.

	0	1	2	3	4	5	6	7
0	9	11	19	19	25	25	29	29
1	11	17	17	23	23	28	28	29
2	19	17	22	22	26	26	30	30
3	19	23	22	25	25	29	29	30
4	25	23	26	25	28	28	31	31
5	25	28	26	29	28	31	31	31
6	29	28	30	29	31	31	32	32
7	29	29	30	30	31	31	32	30

	0	1	2	3	4	5	6	7
0	7	10	11	11	15	15	17	17
1	11	13	14	17	17	20	19	20
2	11	16	14	20	17	23	19	23
3	15	17	17	20	20	23	23	23
4	15	20	23	23	26	26	28	26
5	18	20	20	23	23	26	26	27
6	18	22	25	25	28	28	30	29
7	19	21	21	25	29	28	31	29

(a)

(b)

그림 4. 탐색점 수 (a) CDS 방법 (b) 제안한 방법

Fig. 4 Number of search point  
(a) CDS method (b) Proposed method

그림 4의 탐색점 수 및 그림 1의 MVP 분포를 기준으로 계산한 평균 탐색점 수는 CDS가 12.5회이고, 제안한 방법이 9.7회이다. 따라서 제안한 방법은 움직임 추정에 소요되는 블록 당 평균 탐색점수가 CDS에 비해 2.8회 감소함으로써 고속 움직임 추정이 가능함을 알 수 있다.

제안한 방법에서 모션 벡터가 (1,3)에 위치할 때 탐색과정은 그림 5와 같다.

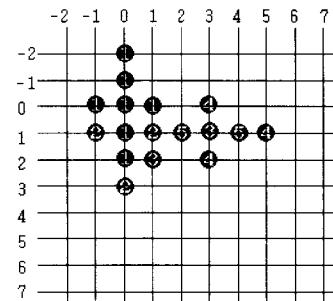


그림 5. 제안한 방법의 탐색 과정  
Fig. 5 Search process of proposed method

#### IV. 실험 결과

제안한 방법에 대한 실험은 352x288 크기의 “stefan”, “coastguard”, “news”, “salesman” 영상을 사용하였고, 각 영상의 프레임 수는 60프레임으로 하였다. “stefan” 및 “coastguard” 영상은 움직임이 크고 빠르며 불규칙적인 특징이 있고, “news”, “salesman” 영상은 움직임이 적고 느린 특징이 있다. 움직임 추정을 위한 블록 크기는 16x16으로 하였으며, 탐색 영역은  $\pm 7$ 로 지정하였다. BMD 측정은 비교적 계산량이 적은 MAD(mean absolute difference)를 사용하였고, MAD는 식(7)과 같다.

$$MAD(x, y) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} |H_c(u+i, v+j) - H_p(u+x+i, v+y+j)| \quad (7)$$

식(7)에서  $H_c$  및  $H_p$ 는 현재 프레임 및 이전 프레임의 화소값을 나타내고,  $u, v$ 는 현재 블록의 좌측상단 좌표를 나타내며,  $x, y$ 는 현재 블록이 이동한 거리이며, MAD가 최소 일 때의  $x, y$  값을 이용하여 현재 블록

의 모션 벡터를 생성한다.

제안한 방법에 대한 실험 결과, 매크로 블록 당 평균 탐색 점 수(ANSP : average numbers of search points)는 그림 6과 같았다.

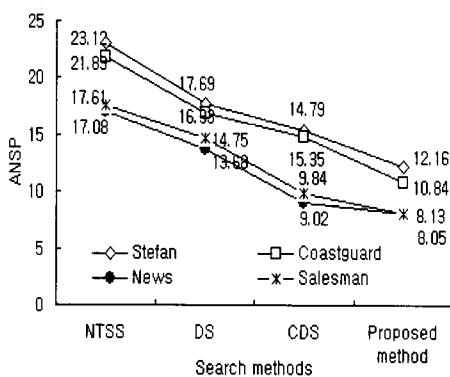


그림 6. 제안한 방법의 ANSP 비교

Fig. 6 Comparison of ANSP for proposed method

그림 6에서 제안한 방법의 ANSP는 CDS에 비해 영상에 따라 차이는 있으나 0.97~4.51개 감소함을 알 수 있다. 전역 탐색에서 필요한 평균 탐색 점 수, 즉 225 ( $= 15^2$ )를 100%로 설정하였을 때, 제안한 방법에 대한 상대적 계산량은 표 2와 같았다. 표 2에서 제안한 방법의 상대적 계산량은 영상에 따라 차이가 있으나, NTSS에 비해 4.02%~4.87%, DS에 비해 2.46%~2.94%, CDS에 비해 0.43%~1.76% 감소하며, 움직임이 많은 영상에서 계산량 감소 효과가 크게 나타남을 알 수 있다.

표 2. 상대적 계산량 비교

Table. 2 Comparison of relative computational complexity

	NTSS (%)	DS (%)	CDS (%)	proposed method(%)
stefan	10.27	7.86	6.82	5.4
coastguard	9.72	7.52	6.57	4.81
news	7.59	6.08	4.0	3.57
salesman	7.82	6.55	4.37	3.61

표 2에서 제안한 방법의 상대적 계산량은 영상에 따라 차이가 있으나, NTSS에 비해 4.02%~4.87%, DS에 비

해 2.46%~2.94%, CDS에 비해 0.43%~1.76% 감소하며, 움직임이 많은 영상에서 계산량 감소 효과가 크게 나타남을 알 수 있다.

움직임이 큰 “coastguard” 영상에 대한 연속적인 평균 탐색 점 수에 대한 실험 결과는 그림 7과 같다.

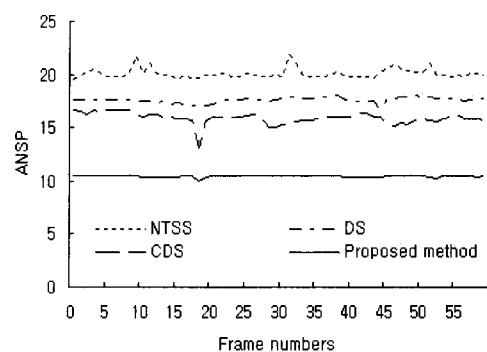


그림 7. 프레임 단위 ANSP 비교

Fig. 7 Comparison of ANSP at each frames

그림 7에 나타나는 바와 같이 영상의 움직임이 빠르고 큰 경우, 기존의 방법들은 평균 탐색 점 수의 변화가 크지만, 제안한 방법은 움직임 방향 치우침 특성을 반영한 탐색 패턴을 적용하기 때문에 평균 탐색 점 수의 변화가 일정하게 유지됨을 알 수 있다.

## V. 결론

동영상 부호화에서 움직임 추정에 적용되는 탐색 패턴의 모양과 방향은 계산량 감소의 중요한 요인이 되기 때문에 MVP 분포 특성에 가장 적합한 탐색 패턴을 사용하여야 한다. 기존의 고속 탐색 방법인 NTSS, DS, CDS는 탐색 영역 내의 MVP 분포에 대한 중심 치우침 특성을 이용하여 고속 움직임 추정을 가능하게 하였으나, 움직임 방향 치우침 특성을 고려하지 않은 탐색 패턴을 사용함으로써 움직임이 크고 빠른 영상에서는 탐색 효율이 떨어지는 경우가 발생하였다. 본 논문에서는 움직임 방향 치우침 특성을 적용한 탐색 패턴을 설정하고, 탐색 과정에서 BMP의 이동 방향에 따라 탐색 패턴을 적용적으로 변화시킴으로써 움직임 추정에 소요되는 계산량을 감소시키는 방법을 제안하였다. 실험 결과 제안한 방

법은 NTSS에 비해 4.02%~4.91%, DS에 비해 2.46%~2.94%, CDS에 비해 0.43%~1.76%의 계산량을 감소시킴으로써 고속 움직임 추정이 가능함을 알 수 있었다.

### 참고문헌

- [1] "Information Technology-Generic Coding of Moving Pictures and Associated Audio Information: Video," ISO/IEC 13818-2 (MPEG-2 Video), 2000.
- [2] "Draft IYU-T recommendation and final draft international standard of joint video specification (ITU-T rec. H.264/ISO?IEC 14 496-10 AVC)," in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-G050, 2003.
- [3] M. J. Chen, L. G. Chen, and T. D. Chiueh, "One-dimensional full search motion estimation algorithm for video coding," IEEE Transactions CASVT, vol. 4, no. 5, pp. 504-509, Oct. 1994.
- [4] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," IEEE Transactions CASVT, vol. 4, no. 4, pp. 438-443, Aug. 1994.
- [5] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Transactions CASVT, vol. 6, no. 3, pp. 313-317, June 1996.
- [6] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast matching motion estimation," IEEE Transactions Image Processing, vol. 9, pp. 287-290, Feb. 2000.
- [7] C. H. Cheung and L. M. Po, "A new cross diamond search algorithm for block motion estimation," IEEE Transactions CASVT, vol. 12, pp. 1168-1177, Dec. 2002.
- [8] 류권열, "적응적 탐색 영역 예측을 이용한 고속 움직임 추정" 한국해양정보통신학회논문지, vol. 12, no. 7, pp. 1187-1192, July 2008.

### 저자소개

류 권 열(Kwon-yeol Ryu)



경북대학교 전자공학과 공학사  
경북대학교 산업공학과 공학석사  
부경대학교 전자공학과 공학박사  
1982년~1985년 : 삼성전자  
기술연구소 근무

1986년~1995년 : 포항공과대학교 전자계산소 근무  
1996년~현재 : 위덕대학교 소프트웨어공학과 근무  
※ 관심분야: 디지털영상처리, 멀티미디어 저작권보호