

특집논문-05-10-3-10

고속 블록 정합 움직임 측정을 위한 십자-다이아몬드-삼각 탐색 알고리즘

김 성 훈^{a)†}, 신 재 민^{a)}, 오 승 준^{a)}, 안 창 범^{a)}, 박 호 종^{a)}, 심 동 규^{a)}

A Cross-Diamond-Triangle Search Algorithm for Fast Block-Matching Motion Estimation

Seong-Hoon Kim^{a)†}, Jae-Min Shin^{a)}, Seung-Jun Oh^{a)}, Chang-Beom Ahn^{a)}, Ho-Chong Park^{a)}, and Dong-Gyu Sim^{a)}

요 약

본 논문에서는 CDS(Cross-Diamond Search algorithm) 및 CDHSs(Cross-Diamond-Hexagonal Search algorithms)의 성능을 향상시키기 위하여 탐색 영역의 위치에 따라 최적의 탐색 형태를 적용하는 CDTS(Cross-Diamond-Triangle Search algorithm)를 제안한다. CDTS에서 탐색 영역의 위치에 따라 최적의 탐색 형태를 적용하기 위하여 다양한 실험 영상들의 움직임 분포를 먼저 분석하였다. 분석된 움직임 분포에 근거하여 CDTS 구현 시 탐색 영역 중심부에서는 십자 탐색 형태와 다이아몬드 탐색 형태를 적용하고, 나머지 탐색 영역에서는 삼각 탐색 형태와 다이아몬드 탐색 형태를 적용한다. CDTS에서 제안하는 삼각 탐색 형태는 기존 탐색 알고리즘에서 사용하는 대칭형 구조와는 다르게 비대칭형 구조를 가지고 탐색점을 더욱 감소시키는 효과를 제공하면서도 화질을 유지할 수 있다. CDTS를 CDS 및 CDHSs와 비교하였을 때 화질 열화 없이 각각 평균 16.22% 및 3.09% 정도의 탐색점 감소 효과가 나타났다.

Abstract

In this paper, we propose a new motion search algorithm called CDTS (Cross-Diamond-Triangle Search algorithm) that uses optimal search pattern according to the position of a search area to improve the performance of CDS(Cross-Diamond Search algorithm) as well as CDHSs(Cross-Diamond-Hexagonal Searches algorithms). We analyze motion distributions in various test video sequences to apply optimal search pattern according to a position of search area. Based on the result of this analysis, we propose a new triangle-shaped search pattern whose structure is asymmetric while previous search patterns are generally symmetric in conventional algorithms. In CDTS, we apply cross- and diamond-shaped search patterns to central search areas, and triangle- and diamond-shaped patterns to the other areas. Applying CDTS to test video sequences, the proposed scheme can reduce search points more than CDS and CDHSs by 16.22% and 3.09%, respectively, without any visual quality degradation.

Keywords : Cross-center-biased property, Block motion estimation, Cross-diamond-triangle-search.

a) 광운대학교 VIA-Multimedia Center

VIA-Multimedia Center, Kwangwoon University

※ 연구는 2005년도 한국소프트웨어진흥원 IT-SoC 핵심실계인력양성사업, 한국전자통신연구원 위탁과제(0201-2005-0080), 한국과학재단 목적기초연구(R01-2002-000-00179-0)로 수행되었음

I. 서론

블록 정합 움직임 탐색은 ISO/IEC MPEG-1/2/4^[1-3], ITU-T H.261^[4], H.263^[5] 그리고 H.264^[6-7] 같은 오늘날의 영상 부호화 기술 및 표준에서 영상의 시간적 중복성을 제거할 수 있는 매우 유용한 방법이다. 움직임 탐색을 위한 대표적인 방법인 전역 움직임 탐색은 탐색 영역 내의 모든 위치를 검사하기 때문에 신뢰할만한 결과를 얻을 수 있으나 많은 연산량을 필요로 한다. 고속 움직임 탐색은 전역 움직임 탐색과 유사한 결과를 얻을 수 있으면서도, 전역 움직임 탐색을 수행할 때 발생하는 많은 연산량을 줄일 수 있는 방법이다. 이러한 고속 움직임 탐색을 위한 대표적인 방법으로 3SS(three-step search)^[8], 2DLOG(2D-logarithmic search)^[9], OSA(orthogonal search algorithm)^[10], N3SS(new three-step search)^[11], 4SS(four-step search)^[12], BBGDS(block-based gradient descent search)^[13], DS(diamond search)^[14-16], HEXBS(hexagon-based search)^[17], CDS(cross-diamond search)^[18], CDHSS(cross-diamond-hexagonal search algorithms)^[19] 등이 있으며, 이러한 방법에서는 일반적으로 대칭형 움직임 탐색 형태를 사용하였다. 대칭형 움직임 탐색 형태는 특정 탐색 위치에서 각 방향으로의 탐색 비중이 같으므로 결과적인 탐색 궤적이 균일하게 나타난다. 균일한 탐색은 그 결과의 신뢰도를 높일 수 있으나, 실제 영상의 움직임 분포가 탐색 영역 전반에 걸쳐 균일하지 않다는 것을 고려할 때, 기존의 대칭형 탐색 형태를 실제 영상의 움직임 분포에 알맞게 보다 비대칭형으로 변형시키는 것이 바람직하다.

기존의 고속 움직임 탐색 방법 중 CDS와 CDHSS는 실제 영상의 움직임 분포가 탐색 영역 중심부에 대부분 분포한다는 것을 고려하여 전체 탐색 과정을 크게 2단계로 나눈다. 이러한 단계적 방법은 실제 영상의 움직임 분포를 고려한 것이므로 탐색점을 매우 효과적으로 감소시킨다. 더욱이, CDHSS는 CDS와 달리 전반부 탐색에서 후반부 탐색을 위하여 수평 또는 수직의 방향 설정을 하고, 후반부 탐색에서 미리 설정된 방향을 사용할 수 있도록 설계된 비대칭형 탐색 형태를 사용하여 추가적으로 탐색점을 감소시킨다. 후반부 탐색에서의 비대칭형 탐색 형태의 사용으로 인하여

탐색점 감소 효과가 있다는 것은 탐색 과정 초기에 나타난 움직임의 방향이 결과적인 움직임의 방향일 확률이 높다는 것을 의미한다. 그러나, CDHSS의 비대칭형 탐색 형태는 기존의 대칭형 탐색 형태의 탐색점 수를 줄이기 위하여 설계된 것이기 때문에, 실제 영상의 움직임 분포를 분석하여 그 특징을 충분히 고려한 비대칭형 탐색 형태의 설계로서 추가적인 탐색점 감소를 기대할 수 있다.

제안하는 방법인 CDTS는 이러한 단계적 탐색 방법과 비대칭 탐색 형태를 같이 사용한다. CDTS에서 사용하는 비대칭 탐색 형태는 이등변 삼각형 구조로서 기존의 비대칭 탐색 형태보다 실제 영상의 움직임 분포가 더 많이 반영되어 탐색점을 더욱 감소시키도록 설계되었으며, 전반부 탐색에서 설정된 방향이 최적의 움직임 방향과 달라서 발생할 수 있는 탐색점 수의 증가를 보다 적게 할 수 있도록 설계되었다.

본 논문은 2장에서 전역 탐색으로 얻은 동영상의 움직임 분포 특성을 분석하여 움직임 탐색 과정을 전반부 탐색과 후반부 탐색으로 구분하고, 3장에서 기존에 제안되었던 탐색 형태 및 CDTS에서 사용한 탐색 형태를 비교 설명하며, 4장에서 CDTS의 알고리즘 및 이론적 탐색 횟수 이득을 설명한다. 5장에서 실험 결과를 분석하고, 6장에서 결론을 맺는다.

II. 움직임 분포 특성

기존의 대칭형 탐색 형태는 동영상의 움직임 분포 특성을 고려하였다기보다는 전체 탐색 영역을 고르게 탐색할 수 있도록 설계된 것이라고 할 수 있다. 그러므로 보다 효과적으로 탐색점 수를 줄임으로써 SAD 연산 수를 감소시키기 위하여 동영상이 가지는 움직임 분포 특성을 보다 면밀하게 분석하여 그 특성을 반영시키는 것이 요구된다. 상기한 동영상의 움직임 분포 특성을 조사하기 위하여 본 논문에서는 표 1에 나열된 33개의 대표적인 동영상들이 가지는 움직임 분포를 전역 움직임 탐색으로 분석하였다. 표 2는 분석된 움직임 분포 중 탐색 범위가 7인 CIF/QCIF 영상의 결과를 백분율로 나타낸 것이다.

표 1. 분석에 사용된 동영상 목록

Table 1. Video sequences used for analysis

프레임 포맷	동영상 시퀀스 종류 (300 프레임)
QCIF(176x144) CIF(352x288) 4CIF(interpolated from CIF)	akiyo, coastguard, container, foreman, hall, mobile, mother_daughter, news, silent, stefan, table

표 2에서 볼 수 있는 움직임 분포 특성은 크게 두 가지로 분류된다. 하나는 탐색 영역의 중심부에 대부분의 움직임이 분포한다는 것이고, 다른 하나는 나머지 탐색 영역에서 탐색 영역의 수직/수평축 또는 수직/수평축에서 가까운 곳에 대부분의 움직임이 분포한다는 것이다. 이러한 분석 결과를 바탕으로 본 논문에서 제안하는 CDTS는 전반부 탐색과 후반부 탐색 두 단계로 구성된다.

표 2에서 탐색 범위 0에서 1 사이에는 17.99%, 탐색 범위 1에서 2 사이에는 9.19%의 움직임 분포 증가가 나타나지만, 탐색 범위 2에서 3 사이에는 3.76%, 탐색 범위 3에서 4 사이에는 2.33%와 같이 움직임 분포 증가가 상대적으로 급격히 감소된 것을 알 수 있다. 그리고 86.43%의 움직임 분포가 탐색 범위 2 이내에 존재 하므로 이 영역을 탐색하

는 것을 전반부 탐색이라고 한다. 분석된 자료에 따르면 전반부 탐색 단계에서 대부분의 움직임 탐색이 종료되겠지만, 만약 전반부 탐색에서 움직임 탐색이 종료되지 않으면 전반부 탐색 결과에서 후반부 탐색을 위한 방향을 설정하고 계속해서 후반부 탐색을 한다. 분석 결과를 근거로 한 전반부 탐색 형태는 3.1절에서 설명한다.

표 2에서 탐색 범위 2인 전반부 탐색 영역을 제외한 나머지 영역 중 탐색 범위 중심점을 기준으로 수평/수직축의 주변이 다른 위치에 비해 상대적으로 움직임 분포가 큰 것을 볼 수 있다. 따라서, 탐색 범위 중심점을 기준으로 대각선 방향 보다는 수평/수직축 방향에 더 많은 비중을 두고 탐색 범위 2 이내를 제외한 영역에 대해 후반부 탐색을 수행한다. 후반부 탐색 형태는 3.2절에서 설명한다.

III. 탐색 형태

1. 전반부 탐색 형태

전반부 탐색 영역은 작은 움직임이 분포하는 곳이므로

표 2. 분석에 사용된 동영상의 탐색 영역 내 움직임 분포도

Table 2. The motion distribution in the search area of video sequences used for analysis

	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7
-7	0.07%	0.02%	0.02%	0.02%	0.02%	0.03%	0.04%	0.16%	0.04%	0.03%	0.03%	0.03%	0.03%	0.02%	0.13%
-6	0.02%	0.01%	0.01%	0.01%	0.01%	0.01%	0.02%	0.11%	0.06%	0.02%	0.02%	0.01%	0.01%	0.01%	0.04%
-5	0.03%	0.01%	0.01%	0.01%	0.01%	0.02%	0.03%	0.09%	0.03%	0.02%	0.02%	0.01%	0.02%	0.01%	0.04%
-4	0.03%	0.01%	0.01%	0.02%	0.02%	0.02%	0.04%	0.16%	0.05%	0.03%	0.03%	0.02%	0.02%	0.01%	0.04%
-3	0.03%	0.01%	0.02%	0.02%	0.04%	0.05%	0.07%	0.21%	0.09%	0.06%	0.05%	0.03%	0.02%	0.02%	0.04%
-2	0.04%	0.02%	0.03%	0.04%	0.08%	0.11%	0.14%	0.37%	0.16%	0.10%	0.08%	0.05%	0.03%	0.03%	0.07%
-1	0.13%	0.04%	0.05%	0.06%	0.11%	0.18%	0.62%	1.92%	0.47%	0.22%	0.16%	0.09%	0.06%	0.05%	0.14%
0	0.55%	0.19%	0.40%	0.50%	0.56%	1.79%	7.97%	59.25%	3.54%	4.74%	1.23%	0.46%	0.32%	0.18%	0.48%
1	0.14%	0.06%	0.09%	0.09%	0.14%	0.24%	0.65%	2.13%	0.69%	0.19%	0.12%	0.07%	0.06%	0.06%	0.16%
2	0.05%	0.02%	0.03%	0.03%	0.07%	0.10%	0.16%	0.41%	0.17%	0.11%	0.09%	0.04%	0.03%	0.03%	0.08%
3	0.03%	0.02%	0.02%	0.02%	0.04%	0.05%	0.06%	0.17%	0.07%	0.06%	0.10%	0.03%	0.02%	0.02%	0.07%
4	0.03%	0.01%	0.02%	0.02%	0.02%	0.03%	0.04%	0.14%	0.06%	0.04%	0.04%	0.02%	0.02%	0.02%	0.08%
5	0.03%	0.01%	0.01%	0.02%	0.02%	0.02%	0.03%	0.09%	0.08%	0.03%	0.03%	0.02%	0.01%	0.01%	0.06%
6	0.03%	0.02%	0.02%	0.02%	0.02%	0.02%	0.03%	0.11%	0.02%	0.02%	0.04%	0.02%	0.01%	0.01%	0.07%
7	0.09%	0.03%	0.03%	0.02%	0.03%	0.03%	0.04%	0.13%	0.04%	0.03%	0.07%	0.06%	0.05%	0.05%	0.23%

작은 움직임을 효과적으로 찾을 수 있는 탐색 형태가 필요하다. 또한, 전반부 탐색 영역의 움직임 분포는 후반부 탐색 영역의 움직임 분포에 비하여 상당히 높기 때문에 전반부 탐색 형태는 제안하는 방법의 전체적인 탐색 성능 및 결과에 매우 큰 영향을 줄 수 있다. 이러한 점을 고려하여 설계된 CDTS의 전반부 탐색 형태는 [19]를 참조하여 그림 1과 같이 세 가지 형태로 분류된다.

● Cross-Shaped Pattern Size 1 : CSP-S1

표 2에서 보여 주듯이 (0,0), (±1,0), (0,±1)의 분포는 매우 높으므로, 초기 탐색에서 이 위치들을 포함하는 탐색 형태인 그림 1(a)와 같은 CSP-S1 탐색을 하여 최소의 탐색점으로 최적의 움직임 위치를 찾는다.

● Cross-Shaped Pattern Size 2 : CSP-S2

CSP-S1에 추가적으로 (±2,0), (0,±2)가 포함된 분포는 전체 탐색 영역에서 CSP-S1를 제외하면 높은 분포를 보이므로, CSP-S1 탐색 이후에 그림 1(b)와 같은 CSP-S2 탐색을

하여 최소의 탐색점으로 최적의 움직임 위치를 찾는다.

● Diamond-Shaped Half-Pattern Size 2 : DSHP-S2

CSP-S2에 추가적으로 (±1,±1)가 포함된 분포는 전체 탐색 영역에서 CSP-S2를 제외하면 역시 높은 분포를 보이므로, CSP-S2 탐색 이후에 CSP-S1과 CSP-S2 탐색의 결과로 얻어진 최소 SAD 위치에 따라서 상/하/좌우 중 한 가지 방향을 설정하고, 그 방향에 따라서 (±1,1), (±1,-1), (-1,±1), (1,±1) 중 하나를 선택한 후 DSHP-S2 탐색을 하여 최소의 탐색점으로 최적의 움직임 위치를 찾는다. 그림 1(c)은 선택된 위치가 (1,±1)인 경우에 사용하는 DSHP-S2이다.

2. 후반부 탐색 형태

후반부 탐색 영역은 큰 움직임이 분포하는 곳이므로 큰 움직임을 효과적으로 찾을 수 있는 탐색 형태가 필요하다. 후반부 탐색 형태 설계는 “① 후반부 탐색 영역의 움직임이 탐색 영역의 수직/수평축 또는 수직/수평축에서 가까운 곳

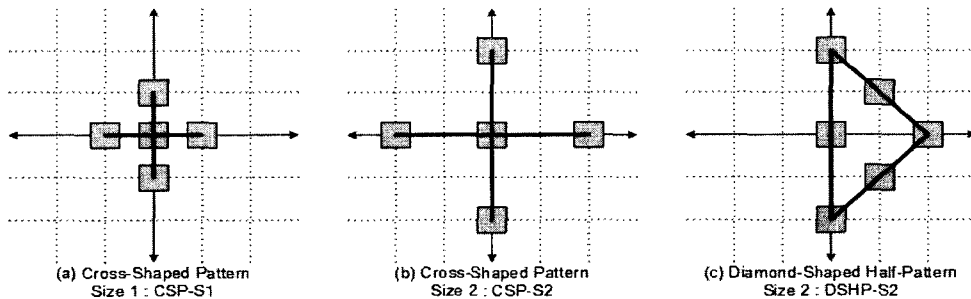


그림 1. 세 가지 전반부 탐색 형태
Fig. 1. Three search shapes on the first-half process

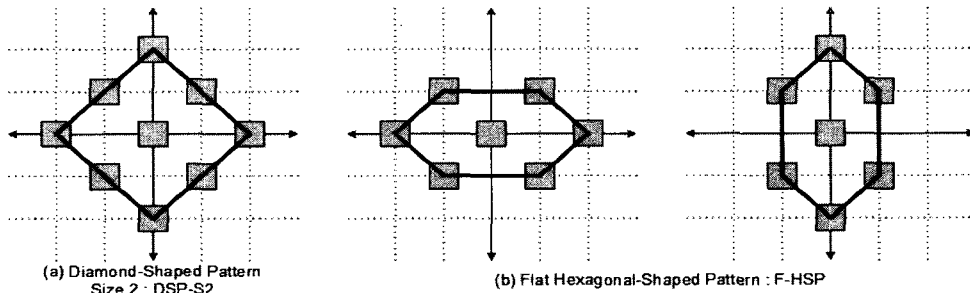


그림 2. DSP-S2와 F-HSP : CDS와 CDHSs의 각 후반부 탐색 형태
Fig. 2. DSP-S2 and F-HSP : the search pattern of second-half process in CDS and CDHSs, respectively

에 대부분 분포한다는 것”과, “② 후반부 탐색은 큰 움직임을 탐색하면서 발생하는 많은 탐색점의 수를 최소화하기 위하여 최적의 움직임이 위치하는 곳까지 빠르게 이동할 수 있도록 해야 한다는 점”을 고려해야 한다. 그림 2(a)는 CDS의 후반부 탐색 형태인 DSP-S2이고, 그림 2(b)는 CDHSS의 후반부 탐색 형태 중 납작한(Flat) 형태중 하나인 F-HSP이다. 이들 탐색 형태는 후반부 탐색 형태 설계의 고려사항 중 ①을 충분히 고려하여 설계되었다고 볼 수 있지만, ②를 충분히 고려하여 설계되었다고 보기는 어려우므로 개선될 필요가 있다. 따라서 이러한 점들을 고려하여 설계된 CDTS에서 사용하는 후반부 탐색 형태는 TSP (Triangle-Shaped Pattern)로서 그림 3과 같다.

TSP는 그림 3과 같이 상/하/좌/우의 방향을 가진 TSP-U, TSP-D, TSP-L, TSP-R의 4가지로 구분되는 비대칭 탐색 형태로서 최대 탐색 범위가 3이다. 이러한 구성을 가진 TSP는 전반부 탐색에서 설정된 방향으로 기존의 후반부 탐색 형태보다 빠르게 움직임 탐색을 할 수 있으므로, 움직임 탐색 과정 초기에 나타난 움직임의 방향이 결과적인 움직임의 방향일 확률이 높다는 것과¹⁹⁾ 후반부 탐색 형태 설계의 고려사항 모두를 충분히 고려했다고 볼 수 있다. 또한, TSP는 움직임 탐색 과정에서 탐색 방향이 변경될 수 있도록 설계되었다. 이러한 설계 의도는 전반부 탐색에서 설정

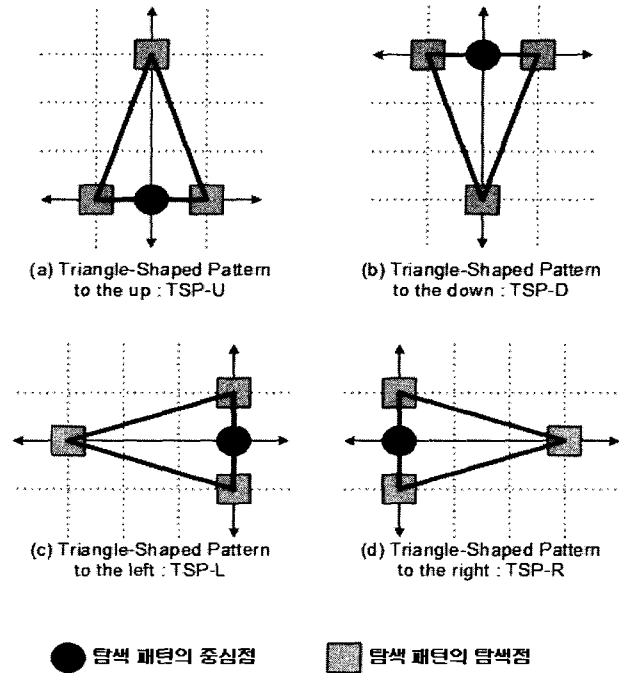


그림 3. CDTS의 후반부 탐색 형태 : TSP
Fig. 3. The second-half search patterns of CDTS : TSP

된 움직임 방향이 최적의 움직임 방향과 달라서 발생할 수 있는 탐색점 수의 증가를 보다 적게 하려는 것이다. TSP를 사용하면서 방향을 변경하는 예를 그림 4에 보였다.

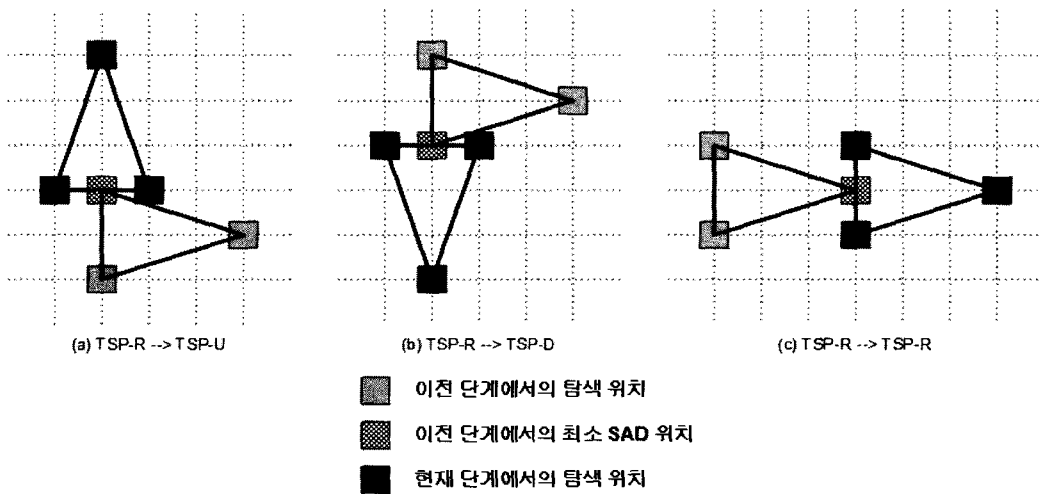


그림 4. TSP의 방향 전환 예시
Fig. 4. Three examples of a direction change of TSP

IV. CDTS

1. 알고리즘

CDTS는 CDS, CDHSs 등과 유사하지만, 후반부 탐색에서 동영상의 움직임 분포 특성을 충분히 반영한 탐색 형태인 TSP를 사용하여 더욱 우수한 성능을 발휘한다는 점에서 이러한 방법들과 구분된다. CDTS는 그림 5와 같이 5단계로 수행된다.

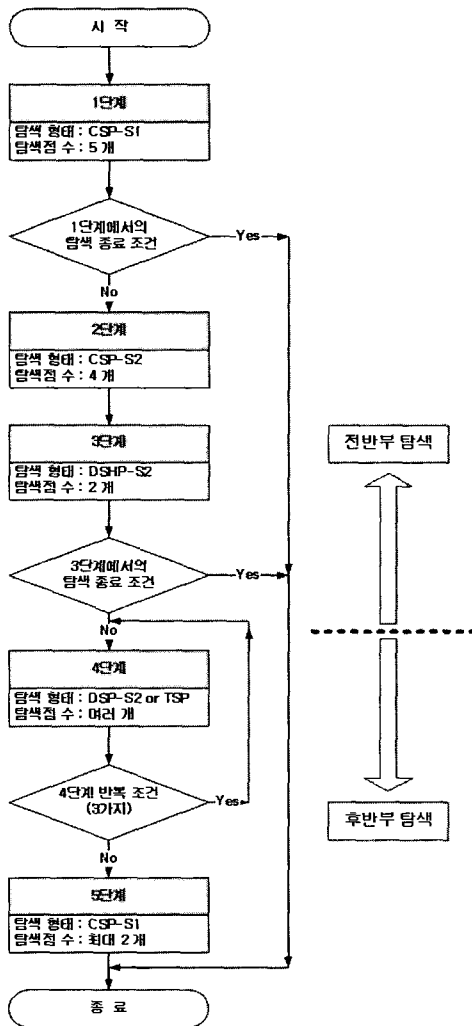


그림 5. CDTS 수행 흐름도
Fig. 5. Flow Chart of CDTS

- 1단계 : 탐색 영역 중심에서 CSP-S1 탐색으로 5개의 위치를 탐색한다. 이 때, 그림 6(a)와 같이 5개의 위치 중 (0,0)에서 최소 SAD가 나타나면, 이 위치를 최적의 움직임 탐색 결과로 채택하고 움직임 탐색의 모든 과정을 종료한다. 그렇지 않으면 나머지 4개의 위치 중 최소 SAD가 나타난 위치에 따라 방향을 설정하고 계속해서 2단계를 수행한다. 예를 들어 그림 6(b)와 같이 최소 SAD가 (1,0)에서 나타났으면 우향으로 설정한다.

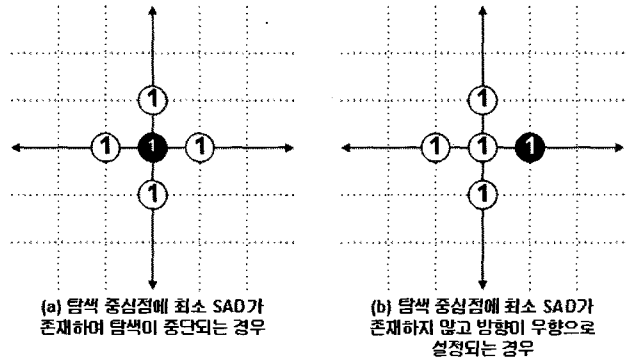


그림 6. 1단계 예제
Fig. 6. The example of the 1st step

- 2단계 : 탐색 영역 중심에서 CSP-S2 탐색을 하여 1단계 이후 추가적인 4개의 위치인 $(\pm 2,0)$ 과 $(0,\pm 2)$ 를 탐색한다. 이 때, 1단계에서 보다 작은 SAD가 존재하는 위치가 나타나면 적절히 방향을 수정하고 계속해서 3단계를 수행한다. 그림 7은 2단계에서 나타날 수 있는 세 가지 탐색 결과의 예를 나타낸 것이다.
- 3단계 : 탐색 영역 중심에서 앞서 설정된 방향을 기준으로 DSHP-S2 탐색을 한다. 즉, 추가적인 2개의 위치를 탐색한다. 예를 들어, 방향이 우향이었던 3단계에서 탐색되는 위치는 그림 8에 삼각형 도형으로 표시된 위치인 $(1, \pm 1)$ 이다. 이 때, $(\pm 1,0)$ 또는 $(0,\pm 1)$ 위치에 최소 SAD가 존재한다면, 그 위치를 최적의 움직임 탐색 결과로 채택하고 움직임 탐색의 모든 과정을 종료한다. 그렇지 않으면 계속해서 4단계를 수행한다. 그림 8은 3단계에서 나타날 수 있는 세 가지 탐색 결과의 예를 나타낸 것이다.

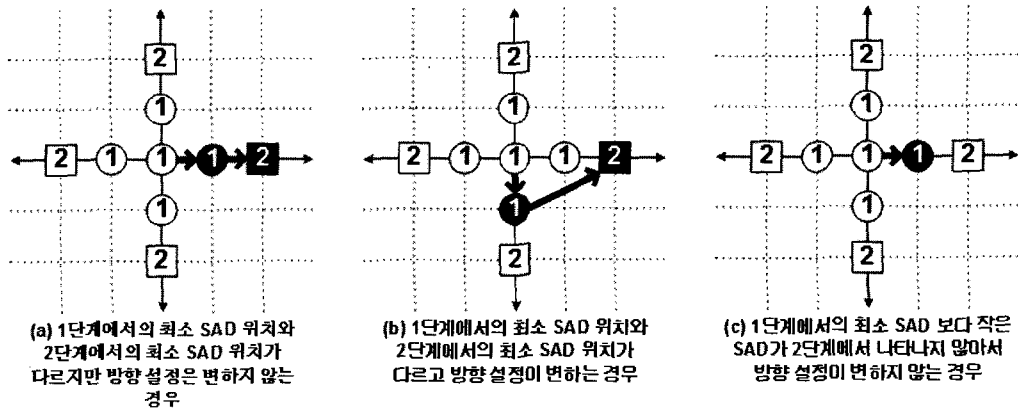


그림 7. 2단계 예제
Fig. 7. The example of the 2nd step

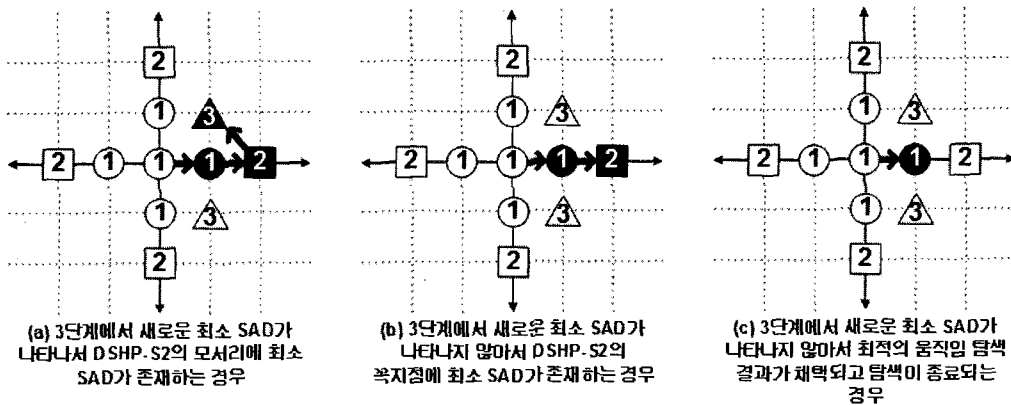


그림 8. 3단계 예제
Fig. 8. The example of the 3rd step

■ 4단계

경우-(1) : 이전 단계에서 DSHP-S2 또는 DSP-S2 탐색을 하였고, 새로운 최소 SAD의 위치가 DSHP-S2 또는 DSP-S2의 모서리에서 나타났으면, 그 위치를 중심으로 하는 DSP-S2 탐색을 한다. 그림 9(a)는 경우-(1)을 보여 준다.

경우-(2) : 이전 단계에서 DSHP-S2 또는 DSP-S2 탐색을 하였고, 새로운 최소 SAD의 위치가 DSHP-S2 또는 DSP-S2의 꼭지점에서 나타났으면, 그 위치를 탐색의 중심으로 하고 주어진 방향을 가진 TSP 탐색을 한다. 그림 9(b)는 경우-(2)를 보여 준다.

경우-(3) : 이전 단계에서 TSP 탐색을 하였고, 새로운 최소 SAD의 위치가 나타났으면, 그 위치를 탐색의 중심으로 하고 새로운 최소 SAD의 위치에 따라 변경된 방향을 가진 TSP 탐색을 한다. 그림 9(c)(d)(e)는 경우-(3)을 보여 준다.

기타 : 4단계 탐색에서 새로운 최소 SAD 위치가 발견되면 4단계를 반복하고, 그렇지 않으면 계속해서 5단계를 수행한다.

■ 5단계 : 이전 단계에서 나타난 최소 SAD 위치를 탐색의 중심으로 하여 1단계에서 사용했던 CSP-S1 탐색을 한다. 이 때, TSP의 구조적 특징 때문에 최대 2개의 추가적

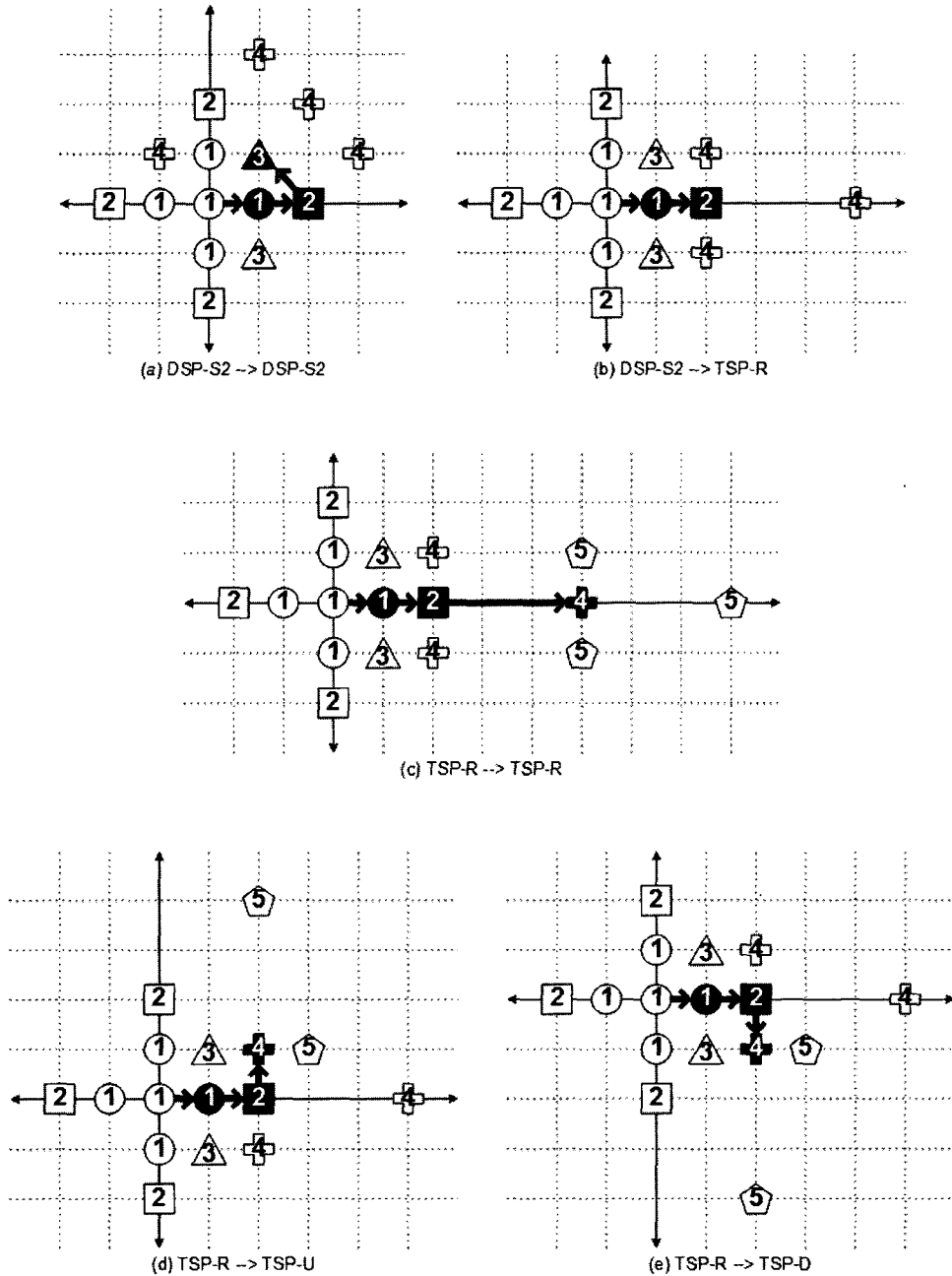


그림 9. 4단계 예제
 Fig. 9. The example of the 4th step

인 탐색점이 발생하므로, 이렇게 탐색된 3개의 위치 중 최소 SAD가 나타난 위치를 최적의 움직임 탐색 결과로 채택하고 움직임 탐색의 모든 과정을 종료한다.

2. 이론적 탐색 횟수 이득

그림 10은 움직임 탐색 범위가 7인 CIF/QCIF의 경우에

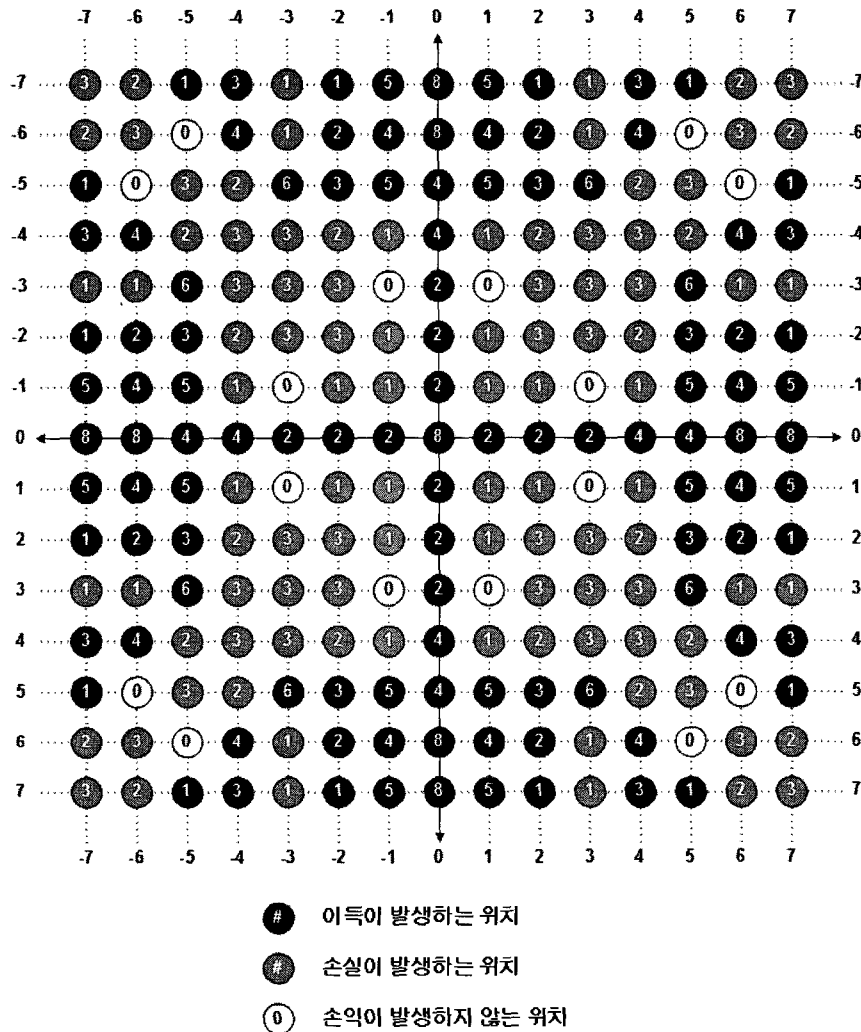


그림 10. DS에 대한 CDTS의 이론적 탐색 횟수 이득
 Fig. 10. The gain in search points using CDTS over DS

이론적인 최단거리 검색을 수행하여 얻은 DS 탐색에 대한 CDTS의 탐색 횟수 이득이다. 그림 10은 살펴보면 탐색 영역 중심부와 탐색 영역의 수직/수평축 또는 수직/수평축에서 가까운 곳의 이득이 크게 나타난다. 이러한 분포는 표 2의 움직임 분포와 매우 유사한 것을 알 수 있으며, [19]에서 CD와 CDHSs를 비교한 것보다 더욱 표 2와 유사한 것을 알 수 있다.

또한, 이론적인 탐색점 수의 이득을 N_s , 각 위치에서의 움직임 존재 확률을 P_p , 각 위치에서의 이득을 n_p , 각

위치를 p , 탐색 영역내의 각 위치의 집합을 W 라고 할 때의 최대 이득을 다음과 같은 수식으로 정리하였다. 이 수식에 따라 분석에 사용된 영상을 기준으로 하는 이론적인 탐색점 수의 최대 이득을 표 2와 그림 10을 참고하여 계산하면 하나의 블록에 대하여 약 5.46 이다.

$$N_s = \sum_{p \in W} n_p P_p$$

V. 실험 결과 및 분석

CDTS의 성능 실험을 위한 비디오 부호화기는 MPEG-2를 사용하였고, 비트율 제어(Rate Control) 모듈을 사용하지 않은 VBR(Variable Bit Rate) 방법으로 부호화 하였다. 실험 영상은 표 1에 나열된 33개 영상의 휘도 성분만 사용하였다. 탐색 범위는 CIF/QCIF가 7이고, 4CIF는 15이다. N=15, M=1 이고, 반화소 움직임 측정을 하였으며, 순차 영상으로 부호화하였다. 실험은 부호화 과정에서 SAD 연산 횟수를 측정하는 것이며, CDS 및 CDHSs를 같은 환경에

서 실험하여 그 결과를 비교하였다.(본 실험의 CDHSs는 F-HSP를 사용한 것으로서 [19]에서의 실험 결과를 고려한 것이다.)

표 3은 CDHSs를 CDS와 비교한 것이고, 표 4는 CDTS를 CDS와 비교한 것이며, 표 5는 CDTS를 CDHSs와 비교한 것이다. 실험 결과에서 음수로 표기된 것은 SAD 연산 수가 감소되었다는 뜻으로서, 두 방법 모두 CDS 보다 모든 영상에 걸쳐 SAD 연산 수가 감소되었다. 더욱이, 후반부 탐색에서 제안하는 방법과 유사하게 이미 설정된 방향을 사용하는 CDHSs와 CDTS를 비교한 표 5에서 CDTS는

표 3. CDS에 대한 CDHSs의 비교 실험 결과

Table 3. The experimental result about CDHSs over CDS

실험 동영상 \ 동영상 포맷	QCIF	CIF	4CIF	QCIF	CIF	4CIF
akiyo	-98,732	-406,366	-1,560,555	-22.21%	-21.42%	-19.54%
coastguard	-31,690	-199,638	-1,524,686	-6.39%	-7.40%	-10.97%
container	-98,944	-411,226	-1,455,706	-22.20%	-21.47%	-17.61%
foreman	-71,829	-294,627	-1,559,008	-12.64%	-10.93%	-11.26%
hall	-94,745	-367,468	-1,111,129	-21.09%	-18.43%	-11.51%
mobile	-90,422	-99,166	-478,662	-20.14%	-4.76%	-4.62%
mother_daughter	-88,537	-317,331	-1,039,522	-19.15%	-15.05%	-10.32%
news	-92,184	-386,086	-1,547,854	-20.38%	-19.42%	-17.59%
silent	-93,273	-386,455	-1,515,531	-20.02%	-18.81%	-16.13%
stefan	-73,730	-364,286	-1,907,173	-11.78%	-12.77%	-13.43%
table	-79,426	-351,884	-1,618,111	-16.21%	-15.62%	-15.56%

표 4. CDS에 대한 CDTS의 비교 실험 결과

Table 4. The experimental result about CDTS over CDS

실험 동영상 \ 동영상 포맷	QCIF	CIF	4CIF	QCIF	CIF	4CIF
akiyo	-98,732	-406,866	-1,579,676	-22.21%	-21.45%	-19.78%
coastguard	-34,179	-345,042	-2,552,636	-6.89%	-12.79%	-18.37%
container	-99,053	-414,055	-1,505,275	-22.22%	-21.62%	-18.21%
foreman	-85,686	-355,208	-2,200,963	-15.07%	-13.18%	-15.90%
hall	-95,133	-380,136	-1,329,898	-21.18%	-19.06%	-13.78%
mobile	-90,459	-102,174	-775,721	-20.15%	-4.90%	-7.48%
mother_daughter	-89,673	-336,712	-1,287,156	-19.40%	-15.97%	-12.78%
news	-92,707	-399,358	-1,668,972	-20.49%	-20.09%	-18.96%
silent	-95,743	-400,047	-1,693,270	-20.55%	-19.47%	-18.03%
stefan	-103,252	-502,694	-2,604,964	-16.50%	-17.63%	-18.34%
table	-83,301	-387,410	-1,867,618	-17.00%	-17.20%	-17.96%

표 5. CDHSs에 대한 CDTS의 비교 실험 결과

Table 5. The experimental result about CDTS over CDHSs

실험 동영상	동영상 포맷	QCIF	CIF	4CIF	QCIF	CIF	4CIF
akiyo		0	-500	-19,121	0.00%	-0.03%	-0.30%
coastguard		-2,489	-145,404	-1,027,950	-0.54%	-5.82%	-8.31%
container		-109	-2,829	-49,569	-0.03%	-0.19%	-0.73%
foreman		-13,857	-60,581	-641,955	-2.79%	-2.52%	-5.22%
hall		-388	-12,668	-218,769	-0.11%	-0.78%	-2.56%
mobile		-37	-3,008	-297,059	-0.01%	-0.15%	-3.00%
mother_daughter		-1,136	-19,381	-247,634	-0.30%	-1.08%	-2.74%
news		-523	-13,272	-121,118	-0.15%	-0.83%	-1.67%
silent		-2,470	-13,592	-177,739	-0.66%	-0.81%	-2.26%
stefan		-29,522	-138,408	-697,791	-5.35%	-5.56%	-5.68%
table		-3,875	-35,526	-249,507	-0.94%	-1.87%	-2.84%

CDHSs보다도 SAD 연산 수가 더 감소되었다는 것을 알 수 있으며, 움직임이 큰 foreman, stefan과 같은 영상에서는 다른 영상 보다 SAD 연산 수가 더 많이 감소되었다. 특히, akiyo 같이 움직임이 매우 작은 영상에서도 CIF 이상의 해상도에서 SAD 연산 수가 감소되었고, CIF 이상의 해상도에서

는 움직임이 일정한 방향으로 일관되게 나타나는 coastguard 영상에서 SAD 연산 수가 가장 많이 감소되었다.

또한, 그림 11/12/13은 실험 결과 중 CIF 영상의 CDHSs와 CDTS의 각 프레임별 PSNR[dB] 차이를 그래프로 나타

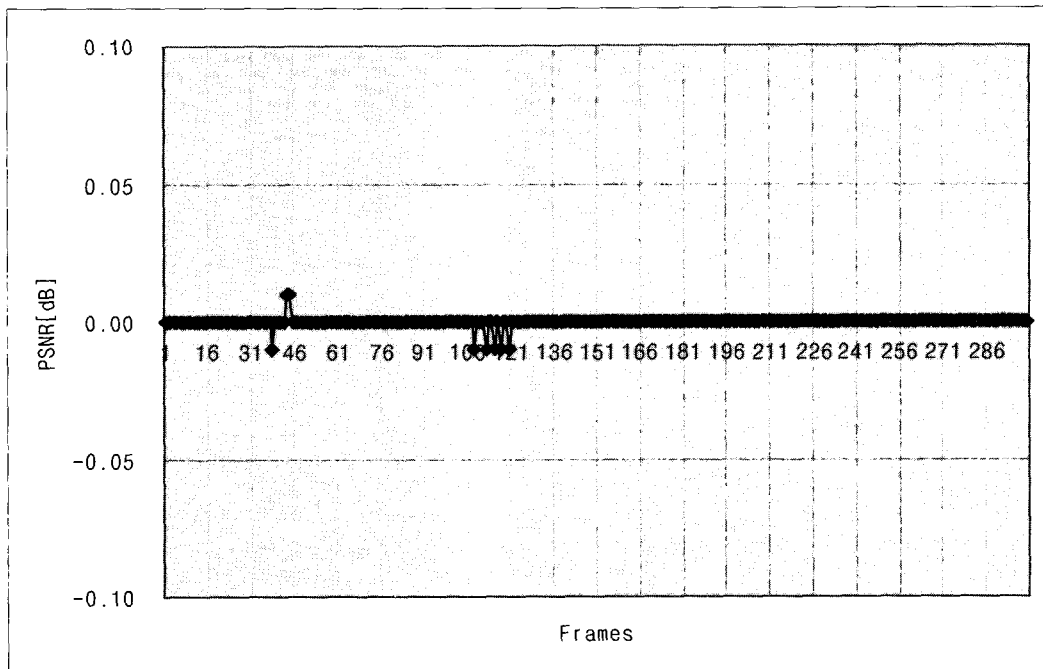


그림 11. CDHSs와 CDTS의 PSNR[dB] 차이 : akiyo

Fig. 11. PSNR[dB] difference between CDHSs and CDTS : akiyo

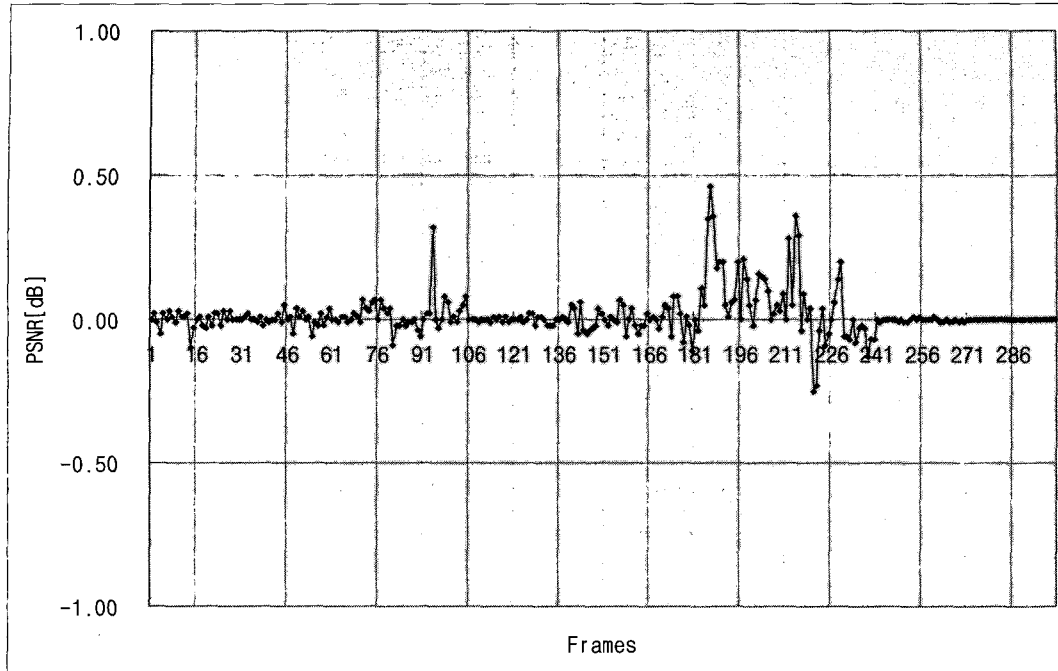


그림 12. CDHSs와 CDTs의 PSNR[dB] 차이 : foreman

Fig. 12. PSNR[dB] difference between CDHSs and CDTs : foreman

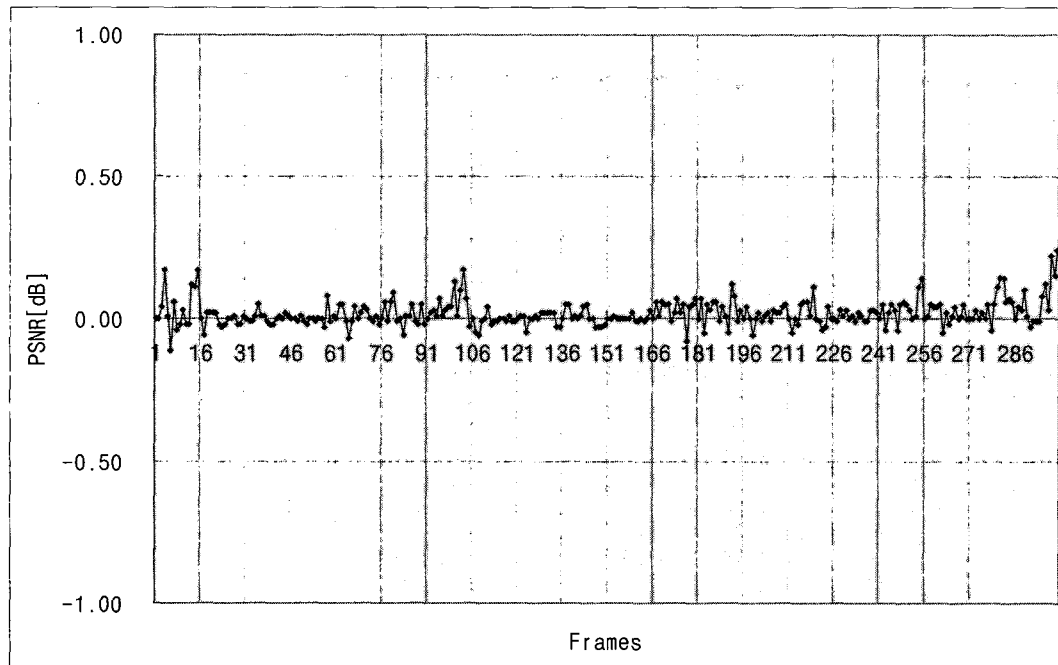


그림 13. CDHSs와 CDTs의 PSNR[dB] 차이 : stefan

Fig. 13. PSNR[dB] difference between CDHSs and CDTs : stefan

낸 것이다. 그림 11은 작은 움직임을 가진 akiyo, 그림 12는 중간 움직임을 가진 foreman, 그림 13은 큰 움직임을 가진 stefan 영상이다. 그래프에서 양수로 표기된 것이 CDTS의 PSNR이 높아졌다는 것을 의미하는 것으로서, 대부분의 경우에 CDHSs와 CDTS의 PSNR 차이가 거의 없음을 보여 준다.

V. 결론

본 논문에서는 실제 영상의 움직임 분포를 분석하여 여러 가지 탐색 형태를 적절히 사용하는 CDTS를 제안하였다. CDTS는 전반부 탐색과 후반부 탐색으로 나누어 진행되는데, 전반부 탐색은 작은 움직임들이 탐색 영역 중심에 높게 분포한다는 것을 고려한 것이며, 후반부 탐색은 큰 움직임들이 탐색 영역의 수직/수평축이나 수직/수평축에서 가까운 곳에 높게 분포한다는 것을 고려한 것이다. 특히, 후반부 탐색에서 사용하는 TSP는 기존에 제안되었던 방법들과 비교하여 영상의 화질 열화가 거의 없으면서도 탐색 점의 수를 상당히 줄일 수 있는 상대적인 비대칭 탐색 형태이다.

실험 결과가 보여 주듯이 CDTS의 성능을 CDS와 비교하였을 때 영상의 화질 열화를 거의 발생시키지 않는 상황에서 SAD 연산 수를 평균 약 16.93% 정도 감소시킬 수 있었으며, CDS를 변형한 다른 방법들과의 비교에서도 SAD 연산 수가 감소하였다. TSP 탐색 방법의 특성상 실험 영상 시퀀스 중에서 foreman, stefan 같이 움직임이 큰 영상에서 SAD 연산 수를 더 감소시킬 수 있었고, CIF 이상의 해상도에서는 움직임이 일정한 방향으로 일관되게 나타나는 coastguard 영상에서 가장 많이 SAD 연산 수를 감소시킬 수 있었다. 또한, 움직임이 매우 작은 akiyo 영상에서도 CIF 이상의 해상도에서 SAD 연산 수를 감소시킬 수 있었다.

참고 문헌

- [1] "Information Technology-Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s-Part 2: Video," JTC1/SC29/WG11, ISO/IEC 11 172-2 (MPEG-1 Video), 1993.
- [2] "Generic Coding of Moving Pictures and Associated Audio Information-Part 2: Video," ITU-T and ISO/IEC JTC 1, ITU Rec. H.262 - ISO/IEC 13 818-2 (MPEG-2 Video), 1994.
- [3] "Information Technology-Coding of Audio Visual Objects-Part 2 Visual," JTC1/SC29/WG11, ISO/IEC 14 469-2 (MPEG-4 Visual), 2000.
- [4] "Video Codec for Audiovisual Services at p□□64 kbit=s," ITU-T SG15, ITU-T Rec. H.261, 2 ed., 1993.
- [5] "Video Coding for Low Bit Rate Communication," ITU-T SG16, ITU-T Rec. H.263, 3rd ed., 2000.
- [6] H. Schwarz and T. Wiegand, "The emerging JVT/H.26L video coding standard," in Proc. of IBC 2002, Amsterdam, The Netherlands, Sep. 2002.
- [7] "Draft ITU-T Rec. and Final Draft International Standard of Joint Video Specification (ITU-T Rec. H.264-ISO/IEC 14 496-10 AVC)," Joint Video Team (JVT) of ITU-T and ISO/IEC JTC1, Geneva, JVT of ISO/IEC MPEG and ITU-T VCEG, JVT-G050r1, 2003.
- [8] T. Koga, K. Iinuma, A. Hirano, Y. Iijima, and T. Ishiguro, "Motion compensated interframe coding for video conferencing," in Proc. Nat. Telecommun. Conf., New Orleans, LA, Nov 1981, pp. G5.3.1 - G5.3.5.
- [9] J. R. Jain and A. K. Jain, "Displacement measurement and its application in interframe image coding," IEEE Trans. Commun., vol. COM-29, pp. 1799 - 1808, Dec. 1981.
- [10] A. Puri, H. M. Hang, and D. L. Schilling, "An efficient block matching algorithm for motion compensated coding," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, 1987, pp. 1063 - 1066.
- [11] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 4, pp. 438 - 443, Aug. 1994.
- [12] L. M. Po and W. C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 6, pp. 313 - 317, Jun. 1996.
- [13] L. K. Liu and E. Feig, "A block-based gradient descent search algorithm for block motion estimation in video coding," IEEE Trans. Circuits Syst. Video Technol., vol. 6, no. 4, pp. 419 - 423, Aug. 1996.
- [14] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," in Proc. 1997 Int. Conf. Information, Communications and Signal Processing (ICICS), vol. 1, Sep. 9 - 12, 1997, pp. 292 - 296.
- [15] S. Zhu and K. K. Ma, "A new diamond search algorithm for fast blockmatching motion estimation," IEEE Trans. Image

- Process., vol. 9, no. 2, pp. 287 - 290, Feb. 2000.
- [16] C. Zhu, X. Lin, L. P. Chau, K. P. Lim, H. A. Ang, and C. Y. Ong, "A novel hexagon-based search algorithm for fast block motion estimation," in Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, May 2001.
- [17] "CCITT SGXV, "Description of reference model 8 (RM8)", " in Document 525, Working Party XV/4, Specialists Group on Coding for Visual Technology, Jun. 1989.
- [18] C. H. Cheung and L. M. Po, "A novel cross-diamond search algorithm for fast block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 12, no. 12, pp. 1168 - 1177, Dec. 2002.
- [19] C. H. Cheung and L. M. Po, "A novel cross-diamond-Hexagonal search algorithms for fast block motion estimation," IEEE Trans. Multimedia, vol. 7, no. 1, pp. 16 - 22, Feb. 2005.
- [20] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, "A novel unrestricted center-biased diamond search algorithm for block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 8, pp. 369 - 377, Aug. 1998.
- [21] Ce Zhu, Senior Member, IEEE, Xiao Lin, Lappui Chau, and Lai-Man Po, "Enhanced Hexagonal Search for Fast Block Motion Estimation", IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 10, Oct. 2004

저 자 소 개



김 성 훈

- 2001년 2월 : 광운대학교 전자공학부 졸업 (학사)
- 2001년 3월~2003년 8월 : 광운대학교 대학원 전자공학과 졸업(석사)
- 2003년 9월~현재 : 광운대학교 대학원 전자공학과 박사과정
- 주관심분야 : Real Time System, DMB



신 재 민

- 1999년 8월 : 광운대학교 전자공학과 졸업 (학사)
- 2003년 9월 ~ 현재 : 광운대학교 대학원 전자공학과 석사과정
- 주관심분야 : Real Time System, DMB



오 승 준

- 1980년 2월 : 서울대학교 전자공학과 졸업(학사)
- 1982년 2월 : 서울대학교 전자공학과 대학원 졸업(석사)
- 1988년 5월 : 미국 Syracuse University 전기/컴퓨터공학과 졸업(박사)
- 1982년 3월~1992년 8월 : 한국전자통신연구원 근무(멀티미디어연구실 실장)
- 1986년 7월~1986년 8월 : NSF Supercomputer Center 초청 학생연구원
- 1987년 5월~1988년 5월 : Northeast Parallel Architecture Center 학생연구원
- 1992년 3월~1992년 8월 : 충남대학교 컴퓨터공학부 겸임교수
- 1992년 9월~현재 : 광운대학교 전자공학부 교수 (VIA-Multimedia Center 멀티미디어연구실)
- 2002년 3월~현재 : SC29-Korea 의장 및 MPEG Forum 부의장
- 주관심분야 : 비디오 데이터 처리, 비디오 코덱 시스템, 비디오 및 영상압축, 멀티미디어서비스

저 자 소 개



안 창 범

- 1981년 2월 : 서울대학교 전자공학과 졸업(학사)
- 1983년 8월 : 한국과학기술원 전기 및 전자공학과 졸업(석사)
- 1986년 8월 : 한국과학기술원 전기 및 전자공학과 졸업(박사)
- 1986년 9월~1991년 4월 : 미국 University of California, Irvine 연구조교수
- 1991년 4월~1992년 8월 : 생산기술연구원 전자정보센터 부교수
- 1992년 9월~현재 : 광운대학교 전기공학과 교수 (신호처리연구실)
- 2001년 9월~2002년 8월 : 미국 University of California, Irvine 객원연구원
- 2004년 9월~현재 : 광운대학교 산학협력단 VIA-Multimedia Center
- 1989년 : Sylvia Sorkin Greenfield Award 수상
- 1997년 : IR52 장영실상 수상
- 주관심분야 : 다차원 신호처리, 영상 압축, 영상시스템



박 호 중

- 1986년 2월 : 서울대학교 전자공학과 (학사)
- 1987년 12월 : University of Wisconsin-Madison, 전기/컴퓨터공학과 (석사)
- 1993년 5월 : University of Wisconsin-Madison, 전기/컴퓨터공학과 (박사)
- 1993년 9월~1997년 8월 : 삼성전자 선임연구원
- 1997년 9월~ 현재 : 광운대학교 전자공학과 부교수
- 주관심분야 : 음성/오디오 신호처리, 멀티미디어시스템



심 동 규

- 1993년 2월 : 서강대학교 전자공학과 (학사)
- 1995년 2월 : 서강대학교 전자공학과 (석사)
- 1999년 2월 : 서강대학교 전자공학과 (박사)
- 2000년 9월 : 현대전자, 기반기술 연구소, 선임연구원
- 2002년 3월 : 바로비전, 뉴미디어 연구소, 선임연구원
- 2005년 2월 : University Washington, Senior research engineer
- 2005년 3월 ~ 현재 : 광운대학교 전자정보공과대학, 컴퓨터공학과 조교수
- 주관심분야 : 영상처리, 컴퓨터비전, 멀티미디어