

정합 검증 패턴을 이용한 적응형 육각 탐색의 성능 개선

김명호*, 박경완*, 오영걸*, 곽노윤*

*천안대학교 정보통신학부

e-mail: nykwak@cheonan.ac.kr

A Performance Improvement of Adaptive Hexagonal Search Using Matching Verification Pattern

Myoung-Ho Kim*, Kyoung-Wan Park*, Young-Geol Oh*, No-Yoon Kwak*

*Division of Information and Communication Engineering,
Cheonan University

요 약

본 논문은 육각 탐색에 기반한 고속 블록 정합 알고리즘의 성능 개선에 관한 것으로, 육각 탐색 과정에서 추출한 정합점에 대해 정합 검증 패턴을 이용하여 정합도를 검증함으로써 국부 최소 문제를 효과적으로 개선한 고속 움직임 벡터 추정 기법에 관한 것이다. 제안된 방법은 우선, 적응형 육각 탐색에 있어서 차순위 국부 정합점을 이용하여 탐색 패턴을 확장한다. 이후 이렇게 확장된 탐색 패턴에서 추출한 임시 전역 정합점을 대상으로 정합 검증 패턴을 이용하여 정합도의 비교우위를 재차 검증한다. 이 검증 결과에 따라 추가 탐색 과정을 계속적으로 진행할 것인지 또는 현 임시 전역 정합점을 최종 전역 정합점으로 확정할 것인지 여부를 결정하는 과정을 반복적으로 수행함으로써 움직임 보상 화질을 개선한 것이다. 제안된 방법에 따르면, 정합 검증 패턴을 적용한 검증 과정에서 추가적인 연산량 증가가 초래되지만 이를 상호타협적으로 보상할 수 있는 화질 측면에서의 성능 개선 효과를 기대할 수 있다.

1. 서론

블록 정합 알고리즘(BMA: Block Matching Algorithm)은 MPEG-1,2,4, H.261, H.263, H.264 등으로 대표되는 동영상 부호화 국제 표준들의 프레임간 부호화 방식[1-5]으로 널리 채택되어 있다. 다수의 블록 정합 알고리즘들 중 전역 탐색 방식은 최적의 성능을 제공하는 반면에 정합될 수 있는 경우의 수가 탐색 영역의 크기에 따라 기하급수적으로 증가하는 단점이 있다. 이에 따라 단계적으로 예상 범위를 줄여나감에 따라 정합 연산량을 감소시키고 동시에 정확한 움직임 벡터를 탐색하려는 의도에서 출발한 고속 블록 정합 알고리즘들이 다수 제안되어 있다. 대표적인 것으로는 삼단계 탐색(TSS; Three Step Search), 2-D 대수적 탐색(2-D logarithmic search), 공액축 탐색(conjugate direction search), 병렬 계층 1-D 탐색(parallel hierarchical 1-D search)[1][2], 마름모꼴 탐색(diamond search)[6], 1D 기울기 감소 탐색(1D gradient descent search)[7], 계층적 탐색(hierarchical search) [8][9][10], 육각 탐색(hexagonal search)[11-14] 등이 있다. 그러나, 이러한 고속 블록 정합 알고리즘들은 전역 탐색에 비해 추정 오차가 크고, 추정된 움직임 벡터의 신뢰성이 현저히 떨어지는 문제점을 내포하고 있다. 추정된 움직임 벡터에 대한 신뢰성을 최대한 유지하면서 탐색 영역과 연산량에 대한 문제를 적응적으로 해결하기 위한 일환으로 적응형 육각 탐색(adaptive hexagonal search) 방법[15]이 제안되어 있다. 하지만 적응형 육각 탐색[15]은 빠른 수렴에 기초하여 고속 움직임 추정이 가능한 반면에 그 동작 특성상 통상의 육각 탐색의 단점인 국부 최소 문제를 근본적으로 해결하지 못하는 한계가 있다.

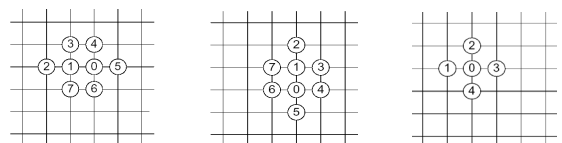
본 논문에서는 적응형 육각 탐색[15]의 문제점을 최소화하기 위해 적응형 육각 탐색의 최적 국부 정합점에 의해 형성된 기존의 탐색 패턴에 차순위 국부 정합점을 중심으

로 새롭게 형성한 탐색점들을 추가하여 탐색 패턴을 적응적으로 확장한 후, 이렇게 확장된 탐색 패턴에서 추출한 임시 전역 정합점을 대상으로 정합 검증 패턴을 이용하여 검증함으로써 움직임 추정 성능을 개선한 고속 움직임 탐색 알고리즘을 제안하고자 한다.

2. 기존의 적응형 육각 탐색

2.1 기본 탐색 패턴

그림 1은 기존의 적응형 육각 탐색을 이용한 블록 정합 알고리즘[15]에서 사용하는 기본 탐색 패턴들을 나타낸 것이다. 그림 1(a)와 그림 1(b)의 각 패턴을 보면 8개의 후보 탐색점들로 구성되어 있다. 각 탐색점들은 수평 및 수직 방향으로 편향된 패턴(biased pattern) 상에 분포된다. 그림 1(a)는 수평축에 편향된 방향으로 후보 영역을 조사하는 수평 편향 패턴(Horizontal Biased Pattern; HBP)을 나타낸 것이고, 그림 1(b)는 수직축에 편향된 방향으로 후보 영역을 조사하는 수직 편향 패턴(Vertical Biased Pattern; VBP)을 나타낸 것이다.



(a)수평 편향 패턴 (b)수직 편향 패턴 (c)초기 탐색 패턴
그림 1. 적응형 육각 탐색의 기본 탐색 패턴

그림 1(a)과 그림 1(b)와 같이 각각 수평 및 수직 방향으로 편향된 육각 패턴은 기울기 감소 방향에 가중치를 두면

서 전역 최소점으로 탐색하기 위한 것이다. 편의상 각 패턴을 구성하는 탐색점들은 정점(vertex), 표면점(surface point) 및 내부점(inner point)으로 구분한다. 정점은 '2'와 '5' 위치이고 표면점은 '3', '4', '6', '7' 위치이며 내부점은 '0'과 '1'이 된다. 특히, '0'은 중심점으로 정의한다. 그림 1(c)는 초기 탐색 패턴을 나타낸 것으로, '0'은 중심점이고 '1', '2', '3', '4'는 정점들이다.

2.2 적응형 육각 탐색

그림 2는 이후 단계에서 적용될 탐색 패턴을 선택하는 과정을 예시한 것이다. 적응형 육각 탐색은 현재 탐색 패턴에서 최소 SSD(Sum of Square Difference) 값 또는 SAD(Sum of Absolute Difference)를 갖는 위치가 다음 탐색 패턴의 중심점으로 선택된다.

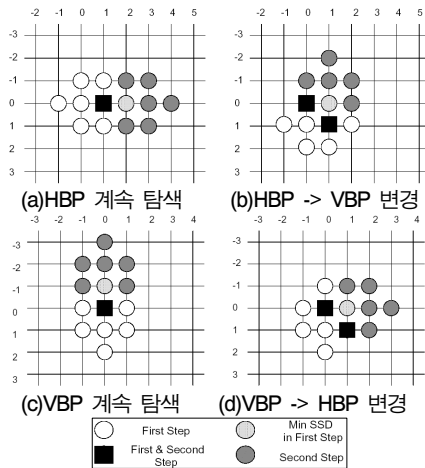


그림 2. 각 위치에 대한 HBP와 VBP의 탐색 예시

예측 평가 척도가 SSD일 경우, 최소 SSD 값을 갖는 위치가 정점이면 현재 탐색 패턴의 방향을 그대로 유지하면서 다음 단계의 탐색 후보점들을 선택하고, 표면점이면 탐색 패턴의 방향을 변경하여 탐색 후보점을 선택하며 내부점일 경우에는 탐색을 종료하고 해당 위치를 최종 정합점으로 삼는다.

최소 SSD 값을 갖는 위치는 수평 편향 패턴과 수직 편향 패턴 중 어떤 패턴을 사용할지 여부를 결정함과 동시에 다음 탐색 단계에 중심점으로 사용한다. 각 패턴에 있는 정점들은 현재 탐색 단계의 기울기 감소 방향과 일치한다. 따라서, 최소 SSD 값을 갖는 위치가 정점들에서 발생할 경우, 탐색 패턴의 방향은 그대로 유지된다. 반면에 최소 SSD 값을 갖는 위치가 표면점들에서 발생할 경우, HBP에서 VBP로 또는 VBP에서 HBP로 탐색 패턴의 방향을 변경한다.

예컨대, 그림 1(a)의 HBP 경우, 최소 SSD가 2번과 5번 위치에서 발생한다면 다음 탐색 단계에 대한 탐색 패턴은 HBP가 된다. 이와 반대로 3번, 4번, 6번, 7번 위치에서 발생한다면 다음 탐색 패턴은 HBP에서 VBP로 변경된다.

그림 2는 수평 및 수직에서 발생할 수 있는 최소 SSD 값을 갖는 위치에 따라 탐색 패턴을 적응적으로 선택하는 전략을 도시한 것이다. 이러한 선택 전략의 목적은 현재의 탐색 과정에서 사용된 탐색 패턴에 따라 발생하는 탐색 영역의 협소함을 보완하기 위한 것이다. 탐색 과정에서 탐색 영역은 균일한 분포를 이룸으로써 후보 점들의 선택 확률을 균등하게 배분하는 것이 바람직하고, 이를 위해 탐색 패턴을 원형(circle)으로 구성하는 것이 이상적이다. 적응형 육각 탐색 알고리즘에서 사용하고 있는 수평 편향 패턴과 수직 편향 패턴에 의해 탐색 영역의 비균등성이 발생하는 것을 보완하기 위해 기울기 감소 방향에 적합하도록 탐색 패턴의 방향을 변경함으로써 현재 탐색 과정에서 제외된

유효 탐색 영역을 다음 탐색 과정에서 포함시켜 탐색한다.

3. 제안된 적응형 육각 탐색

3.1 개요

통계적인 측면에서 볼 때 일반적인 동영상 시퀀스들의 움직임 벡터 분포는 탐색 영역의 중심 부근에 집중적으로 분포하는 것이 일반적이다[14]. 따라서 탐색 영역의 중심 부근에서 고속으로 움직임을 추정할 수 있는 탐색 패턴을 사용하면 양호한 추정 화질을 획득하면서도 탐색 시간을 현저하게 경감시킬 수 있다. 육각 탐색 알고리즘들은 이러한 분포 특성을 효과적으로 이용하고 있는 것이다. 하지만 탐색 영역의 중심 부근에서의 빠른 수렴에 기초하여 고속 움직임 추정이 가능하다는 이점이 있는 반면에 상대적으로 움직임이 큰 동영상 시퀀스에서는 그 동작 특성상 탐색 시간이 증가할 뿐만 아니라 그 편차가 유동적이고 국부 최소 수렴할 가능성이 높은 것이 단점이다.

전술한 기존의 적응형 육각 탐색(adaptive hexagonal search) 방법[15]은 육각 탐색 패턴의 방향성에 적응성을 부여하여 고속 움직임 추정이 가능하면서도 상대적으로 개선된 성능을 제공하고 있다. 하지만 육각 탐색에 기반한 탐색 알고리즘이기 때문에 국부 최소 문제를 근본적으로 해결하지는 못하고 있다. 다시 말해서, 육각 탐색에 기반한 블록 정합 알고리즘은 빠른 수렴에 기초하여 고속 움직임 추정이 가능한 반면에 그 동작 특성상 국부 최소로 수렴할 가능성이 상대적으로 높은 것이 단점이다.

제안된 방법은 이러한 단점을 최소화하기 위한 것으로, 우선, 마름모 패턴을 이용한 초기 탐색에 연이어 최적 국부 정합점의 위치에 따라 수직 편향 패턴과 수평 편향 패턴 중 어느 하나를 선택한 후 이렇게 정한 탐색 패턴에 차순위 국부 정합점의 위치에 따라 수직 편향 패턴과 수평 편향 패턴 중 어느 하나를 추가함으로써 적응적으로 탐색 패턴을 확장한다. 이후 이렇게 확장된 탐색 패턴에서 추출한 임시 전역 정합점을 대상으로 정합 검증 패턴을 이용하여 정합도의 비교우위를 재차 검증한다. 이 검증 결과에 따라 추가 탐색 과정을 계속적으로 진행할 것인지 또는 현 임시 전역 정합점을 최종 전역 정합점으로 확정할 것인지 여부를 결정하는 과정을 반복적으로 수행함으로써 움직임 보상 화질을 개선할 수 있다.

3.2 패턴 확장 및 정합 검증 과정

제안된 알고리즘은 크게 패턴 확장 과정과 정합 검증 과정으로 구성된다. 패턴 확장 과정에서는 차순위 국부 정합점을 이용하여 탐색 패턴을 확장하면서 임시 전역 정합점을 구한다. 정합 검증 과정에서는 확장된 탐색 패턴에서 추출한 임시 전역 정합점을 대상으로 정합 검증 패턴을 이용하여 정합도의 비교우위를 검증한 후에 그 결과에 따라 최종 전역 정합점으로 확정한다.

전체 탐색 과정을 요약하면 다음과 같다. 설명의 편의상 임의의 탐색 패턴 내에서 최소의 SSD 값을 갖는 탐색점을 '1 순위 정합점', 차순위로 작은 SSD 값을 갖는 탐색점을 '2 순위 정합점'이라 명한다.

○ 단계 1: 패턴 확장 과정

그림 1(c)의 초기 탐색 패턴을 미리 지정된 탐색 영역내의 중심점(0,0)에 위치하여 탐색 패턴내의 각 후보점들에 대하여 SSD 값을 계산한다.

계산된 SSD 값들 중에서 1 순위 정합점이 초기 탐색 패턴의 중심점에 위치할 경우, 현재 탐색 과정을 중지하고 단계 2의 정합 검증 과정으로 이동한다. 이 경우, 1 순위 정합점은 단계 2에서 임시 전역 정합점으로 사용된다. 그러나 1 순위 정합점이 초기 탐색 패턴의 중심점에 위치하지 않을 경우, 1 순위 정합점과 2 순위 정합점을 기준삼아 각각 독립적으로 육각 탐색 패턴을 형성하기 위해 HBP 또

는 VBP를 탐색 패턴으로 삼는다. 이때, 1 순위 또는 2 순위 정합점이 초기 탐색 패턴의 1 혹은 3의 위치에서 발생할 경우 HBP를 탐색 패턴으로 선정하고, 탐색점이 초기 탐색 패턴의 2 혹은 4의 위치에서 발생할 경우 VBP를 선정한다. 이를 통해 구한 이 두 육각 탐색 패턴의 교집합을 확장된 탐색 패턴으로 삼고 각 탐색 후보점들의 SSD 값을 계산하여 최소 SSD 값이 발생한 1 순위 정합점의 위치를 탐색한다.

1 순위 정합점이 HBP 또는 VBP의 내부점에 포함될 경우, 단계 2의 정합 검증 과정으로 이동한다.

그렇지 않을 경우, 1 순위 정합점과 2 순위 정합점을 기준삼아 각각 독립적으로 육각 탐색 패턴을 다시 형성하기 위해 HBP 또는 VBP를 탐색 패턴으로 삼는다. 이때, 최소 SSD 값을 갖는 위치가 정점들에서 발생할 경우, 탐색 패턴의 방향은 그대로 유지된다. 반면에 최소 SSD 값을 갖는 위치가 표면점들에서 발생할 경우, HBP에서 VBP로 또는 VBP에서 HBP로 탐색 패턴의 방향이 변경된다.

이를 통해 구한 이 두 육각 탐색 패턴의 교집합을 새로운 확장 탐색 패턴으로 삼고 각 탐색 후보점들의 SSD 값을 계산하는 과정을, 1 순위 정합점이 확장 탐색 패턴의 내부점에서 발견될 때까지 반복적으로 수행한다.

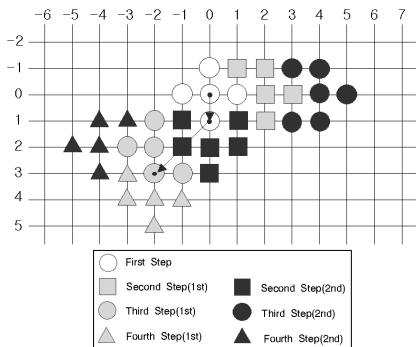


그림 3. 임시 전역 정합점 (-2,3)을 가정한 적응형 육각 탐색

그림 3은 확장된 적응형 육각 탐색을 사용한 블록 정합 과정을 예시한 것이다. 현재 탐색을 실행하는 블록의 실제 움직임 벡터를 $(mx, my) = (-2, 3)$ 로 가정한다면, 탐색 영역의 중심인 $(0, 0)$ 에서 시작한다. 5개의 탐색 후보점들 각각에 대하여 SSD를 계산한 후, 최소 SSD 값을 갖는 1 순위 정합점을 찾고 이어서 차순위로 작은 최소 SSD 값을 갖는 2 순위 정합점을 찾는다. 만약 1 순위 정합점이 $(0, 0)$ 에 위치한다면, 탐색 과정은 종료되고 최종 움직임 벡터는 $(0, 0)$ 이 된다. 반면에 $(0, 0)$ 을 제외한 다른 4개의 후보점들 중에서 1 순위와 2 순위 정합점이 발생되면 계속된 후속 탐색이 필요하다. 그림 3에서 회색의 도형은 1 순위 정합점의 위치에서 육각 탐색을 적용한 것이고, 흑색의 도형은 2 순위 정합점의 위치에서 적응형 육각 탐색을 적용한 것이다.

탐색 영역의 중심인 $(0, 0)$ 에서 시작하여 1 순위 정합점의 좌표가 $(1, 0)$ 이고 2 순위 정합점의 좌표가 $(0, 1)$ 일 때, 이 두 점의 좌표값을 시작으로 다시 확장된 육각 탐색을 전개하여 1 순위 정합점과 2 순위 정합점을 구하게 된다. 1 순위 정합점이 내부점에서 검출되면 그 좌표를 임시 전역 정합점으로 삼고, 그렇지 않다면 2 순위 정합점과 함께 다시 확장된 육각 패턴을 전개한다. 이때 2 순위 정합점이 내부점에서 검출되면 이를 제외시키고 다음 순위의 탐색점을 중심으로 육각 패턴을 전개한다. 이후, 1 순위 정합점의 좌표와 2 순위 정합점의 좌표를 중심으로 다시 확장된 육각 패턴을 전개할 때, 1 순위 정합점은 $(-1, 2)$ 이고, 2 순위 정합점은 $(3, 0)$ 이다. 이 두 정합점을 시작으로 다시 확장된 육각 탐색을 펼친 결과 $(-2, 3)$ 에서 1 순위 정합점이 검출되고, $(-3, 2)$ 에서 2 순위 정합점이 검출된다. 1 순위 정합점의 좌표가 내부점이므로 이것을 임시 전역 정합점으로 삼는다. 결론적으로 처음 $(0, 0)$ 에서 시작했을 때는 오른쪽 방향

으로 움직임이 추정되었다가 육각이 확장되면서 왼쪽 하단에서 최종 움직임이 발견된 것이다. 이 경우, 오른쪽 방향은 국부 최소점이 있는 위치로서 확장된 육각 패턴이 팽창하면서 국부 최소에 빠지지 않고 $(-2, 3)$ 에 있는 임시 전역 정합점을 탐색한 것을 볼 수 있다.

○ 단계 2: 정합 검증 과정

최종적으로 1 순위 정합점이 발생한 현 위치를 임시 전역 정합점으로 정한다. 이 임시 전역 정합점을 추출한 현재의 확장 탐색 패턴에서 내부점 또는 중심점을 제외한 차순위 정합점과 차차순위 정합점을 기준삼아 각각 독립적으로 패턴 확장 과정과 마찬가지로 HBP 또는 VBP를 탐색 패턴으로 정한다. 이를 통해 구한 이 두 육각 탐색 패턴의 교집합을 정합 검증 패턴으로 삼는다.

다음으로, 이 정합 검증 패턴에서 최소 SSD 값을 갖는 검증용 국부 정합점과 앞서 구한 임시 전역 정합점의 SSD 값을 비교한다. 그 비교 검증 결과에서 임시 전역 정합점의 SSD 값이 작을 경우, 현 임시 전역 정합점을 최종 전역 정합점으로 확정하고 그렇지 않을 경우 최종 전역 정합점이 발견될 때까지 패턴 확장 과정과 동일한 방식으로 추가 탐색 과정을 반복적으로 수행한다.

4. 시뮬레이션 결과 및 고찰

본 연구에서는 제안된 방식의 타당성을 평가하기 위해, 서로 다른 동적 특성을 갖는 복수의 동영상에 대상으로 기존의 방식과 제안된 방식에 대한 Pentium M 1.70GHz(512MB RAM)에서 컴퓨터 시뮬레이션을 수행하였다. 시험영상 시퀀스로는 352×224의 FlowerGarden(#40~#80)과 720×480의 Football(#0~#30)의 256 계조 휘도 영상을 이용하였다. 제안된 방법의 움직임 추정 블록의 크기는 16×16이고, 탐색 영역은 ±16이며 왜곡 평가 척도는 SSD를 사용하였다. 그림 4는 두 영상 시퀀스에 대한 각 방식의 PSNR(Peak Signal to Noise Ratio) 비교를 통한 성능 평가 결과를 나타낸 것이다.

표 1을 참고할 때, FlowerGarden과 Football에 대해 제안된 방법(Proposed Method)은 기존의 적응형 육각 탐색 방법(Adaptive Hexagon)[15]에 비해 각각 0.14dB와 1.01dB 만큼 PSNR 측면에서 평균적으로 우수하고, 확장된 육각 탐색 방법(Expanded Hexagon)에 비해 각각 0.05dB와 0.55dB 만큼 PSNR 측면에서 평균적으로 우수하다. 한편, 전역 탐색에 비해 각각 0.06dB와 1.10dB 만큼 평균적으로 저조합을 확인할 수 있다.

표 1. 각 방법의 평균 PSNR 비교 단위: [dB]

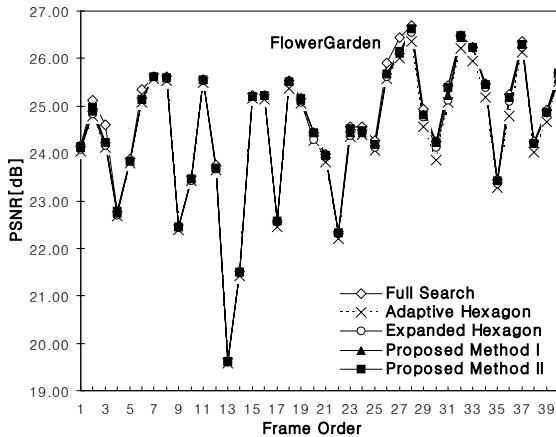
Algorithm / Test Image	Full	Adaptive Hexagon	Expanded Hexagon	Proposed Method
FlowerGarden (#40~#80)	24.58	24.38	24.47	24.52
Football (#0~#30)	24.04	21.93	22.39	22.94

표 2. 각 방법의 평균 연산시간 비교 단위: [msec]

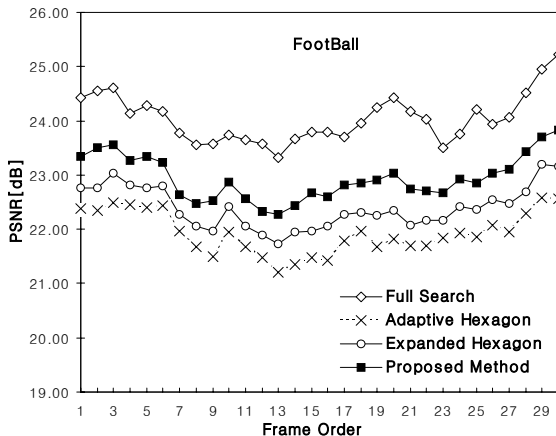
Algorithm / Test Image	Full	Adaptive Hexagon	Expanded Hexagon	Proposed Method
FlowerGarden (#40~#80)	1,212	8	14	24
Football (#0~#30)	6,106	74	143	212

상대적으로 움직임이 작은 FlowerGarden 시퀀스의 경우, 세가지 방법은 모두 전역 탐색에 근접하는 양호한 결과를 제공하는 반면에 움직임이 큰 Football 시퀀스의 경우, 국부 최소에 빠지는 빈도가 상대적으로 높은 것을 알 수 있다. 그러나 제안된 방법은 적응적으로 확장된 탐색 패턴과 정합 검증 패턴을 이용함으로써 기존의 적응형 육

각 탐색의 결점을 부분적으로 보완하고 있다. 그러나 제안된 방법은 표 2와 같이 FlowerGarden과 Football에 대해 기존의 적응형 육각 탐색[15]에 비해 약 2.9배 정도의 연산 시간이 소요되고, 확장형 육각 탐색에 비해 평균적으로 약 1.6배 정도의 연산시간이 소요되는 단점이 있다. 표 2는 각 탐색 방법의 평균 연산 시간을 비교한 것이다.



(a) FlowerGarden(#40~#80)



(b) Football(#0~#30)

그림 4. 영상 시퀀스들에 대한 각 방식의 PSNR 비교

5. 결론

제안된 방법은 적응형 육각 탐색의 문제점을 최소화하기 위해 적응형 육각 탐색의 최적 국부 정합점에 의해 형성된 기존의 탐색 패턴에 차순위 국부 정합점을 중심으로 새롭게 형성한 탐색점들을 추가하여 탐색 패턴을 적응적으로 확장한 후, 이렇게 확장된 탐색 패턴에서 추출한 임시 전역 정합점을 대상으로 정합 검증 패턴을 이용하여 검증하는 고속 움직임 탐색 알고리즘을 제안하였다. 제안된 방법에 따르면, 정합 검증 패턴을 적용한 검증 과정에서 추가적인 연산량 증가가 초래되지만 이를 상호타협(trade-off)적으로 보상할 수 있는 화질 측면에서의 성능 개선 효과를 얻을 수 있었다.

기존의 적응형 육각 탐색 방법은 움직임이 큰 영상 시퀀스에서 국부 최소화로 인한 성능 저하 폭이 큰 반면에 제안된 방법은 이러한 문제를 효과적으로 완화시키고 있음을 알 수 있다. 그러나, 제안된 알고리즘은 적응적인 블록 정합 알고리즘의 일종이기에 움직임 추정을 위한 연산 시간이 입력 영상에 의존적이다. 기존의 적응형 육각 탐색에 비해 보상 화질 측면에서 상대적으로 개선된 성능을 제공하고 있지만 탐색 시간의 편차 문제, 연산량 증가 문제 및 국부 최소 문제를 근본적으로 해결하지는 못하고 있다. 향

후, 이러한 문제들에 효과적으로 대응하기 위한 추가적인 연구가 필요하다.

참고 문헌

- [1] Borke Furht, Joshua Greenberg, Raymond Westwater, 'Motion Estimation Algorithms for Video Compression,' Kluwer Academic Publishers, 1997.
- [2] Guido M. Schuster, Aggelos K. Katsaggelos, 'Rate-Distortion Based Video Compression,' Kluwer Academic Publishers, 1997.
- [3] K. R. Rao, J. J. Hwang, 'Techniques and Standards for Image, Video, and Audio Coding,' Prentice Hall, 1996.
- [4] Iain E. G. Richardson, 'H.264 and MPEG-4 Video Compression,' Wiley, 2003.
- [5] Fernando Pereira and Touradj Ebrahimi, 'The MPEG-4 Book', Prentice Hall, 2002.
- [6] 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. on Circuits Syst. Video Technol., vol. 8, pp. 369-377, Aug. 1998.
- [7] Oscar. T.-C. Chen, "Motion Estimation Using a One-dimensional Gradient Descent Search", IEEE Trans. on Circuits syst. Video Technol., vol. 10, pp.608-616, Jun. 2000.
- [8] S. Panchanathan, E. Chan and X. Wang, "Fast Multiresolution Motion Estimation Scheme for a Wavelet Transform Video Coder," SPIE, vol. 2308, pp. 674-681, 1994.
- [9] 김성만, 이규원, 정학진, 박규태, "웨이브렛 변환 영역에서의 2단계 가변 블록 다해상도 움직임 추정", 한국통신학회 논문지, vol. 22, no. 7, pp. 1487-1504, 1997.
- [10] 광노윤, 황병원, "카메라 패닝 보상 기반 계층적 블록 정합 알고리즘", 정보처리학회논문지, 제6권, 제8호, pp. 2271- 2280, 1999. 8.
- [11] C. Zhu, X. Lin, and L. P. Chau, "Hexagon-based Search Pattern for Fast Block Motion Estimation", IEEE Trans. on Circuits Syst. Video Technol. vol. 12, pp. 349-355, May 2002.
- [12] Z. Guanfeng, L. Guizhong and S. Rui, "A Modified Hexagon-Based Search Algorithm for Block Motion Estimation", Proceedings of the 2003 International Conference on Neural Networks and Signal Processing, vol. 2, pp. 1205-1208, Dec. 2003.
- [13] C. Zhu, X. Lin, L. P. Chau, "An Enhanced Hexagonal Search Algorithm for Block Motion Estimation", Proceedings of the 2003 International Symposium on Circuits and Systems(ISCAS '03), vol. 2, pp. II-392-II-395, May 2003.
- [14] 박인영, 남현우, 위영철, 김하진, "십자와 육각패턴을 이용한 고속 블록 정합 동작 예측 기법", 정보처리학회 논문지B, 제10권, 제7호, pp. 811-814, 2003. 12.
- [15] 전병태, 김병천, "고속 움직임 추정을 위한 적응형 육각 탐색 방법", 한국통신학회 논문지, 제27권, 제7A호, pp. 828-835, 2004. 7.