

특집논문 (Special Paper)

방송공학회논문지 제23권 제5호, 2018년 9월 (JBE Vol. 23, No. 5, September 2018)

<https://doi.org/10.5909/JBE.2018.23.5.628>

ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

후보 선별과 최적후보 안정성을 이용한 고속 움직임 예측 알고리즘

김 종 남^{a)†}

Fast Motion Estimation Algorithm using Selection of Candidates and Stability of Optimal Candidates

Jong Nam Kim^{a)†}

요 약

본 논문에서는 비디오 부호화에서 중요한 고속 움직임 예측 알고리즘을 제안한다. 기존의 전영역 탐색 방법의 많은 계산량으로 인하여 예측화질향상과 연산량 감축을 위한 연구가 진행되어 왔으며, 본 논문에서는 전영역 탐색기반의 방법과 비교하여 예측화질은 거의 유지하면서 무의미한 계산량을 줄이는 방법을 제안한다. 제안하는 알고리즘은 각 후보 지점에 대하여 부분 블록 에러 합을 계산하고, 이에 따라 다음 단계에서 진행할 후보들을 선별하고, 최소 에러지점의 최적후보에 대해 단계별 안정성을 비교 판단하여 그 다음 단계의 진행 여부를 결정함으로써 최적의 움직임 벡터를 계산한다. 이를 통하여 전체의 최소블록매칭에러를 갖는 지점을 조기에 발견하고, 불필요한 후보들을 더 빨리 제거함으로써 불필요한 계산량을 줄이고 계산속도 향상을 얻을 수 있다. 또한 제안하는 알고리즘은 단독으로 사용할 뿐만 아니라 기존의 고속 알고리즘들과 결합하여 사용해도 예측화질은 거의 유지하면서 계산량을 대폭 줄일 수 있으며, 실험결과에서 이를 검증한다.

Abstract

In this paper, we propose a fast motion estimation algorithm which is important in video encoding. So many fast motion estimation algorithms have been published for improving prediction quality and computational reduction. In the paper, we propose an algorithm that reduces unnecessary computation, while almost keeping prediction quality compared with the full search algorithm. The proposed algorithm calculates the sum of partial block matching error for each candidate, selects the candidates for the next step, compares the stability of optimal candidates with minimum error, and finds optimal motion vectors by determining the progress of the next step. By doing that, we can find the minimum error point as soon as possible and obtain fast computational speed by reducing unnecessary computations. Additionally, the proposed algorithm can be used with conventional fast motion estimation algorithms and prove it in the experimental results.

Keyword : Motion estimation, full search, selection of candidates, stability of optimal candidates, partial distortion elimination

I. 서론

비디오 압축에서 움직임 예측 모듈은 비디오 데이터의 몇 가지 중복성 중에 시간적 중복성을 줄이는 기술이며, 이를 통하여 정지영상 압축보다 현저히 높은 압축률을 얻을 수 있다. 이러한 움직임 예측은 연속되는 비디오 프레임간의 유사한 영역을 찾는 것이며, 이웃하는 프레임에서 유사한 영역의 이동 변위를 움직임 벡터로 정의한다. 연속되는 비디오 프레임의 움직임을 예측하는 방법들 중에 블록매칭 움직임 예측 방법이 가장 널리 사용되며, 소요되는 계산량 대비 예측화질이 우수하여 동영상 압축 국제 표준에 널리 사용되어 왔다. 블록 매칭 움직임 예측 방법들에서 가장 기본적인 방법이 전영역 탐색 (full search) 방법인데, 이는 설정된 탐색 영역에서 모든 탐색 후보점에 대하여 블록매칭 에러 (block matching error)를 계산하고 그 중에 최소의 에러를 갖는 후보 탐색지점을 찾는 것이다. 이 방법은 예측화질과 구현복잡도 측면에서는 우수하나, 계산량이 많으며, 특히 탐색 영역이 커진다면 계산량이 심각한 문제가 되기도 한다. 이를 해결하기 위해 고속 움직임 예측에 대한 많은 연구가 수행되어 왔다^[1].

현재까지 블록기반의 움직임 예측 알고리즘들을 예측화질 측면에서 크게 두 가지로 나눌 수 있는데, 하나는 전영역 탐색 방법과 비교하여 예측 화질 저하를 갖는 것이고, 다른 하나는 예측 화질 저하를 갖지 않는 방법이다. 본 논문에서는 전자를 손실 움직임 예측이라 정의하고, 후자를 무손실 움직임 예측이라 정의한다. 손실 움직임 예측 방법은 화질저하를 감수하면서 계산량을 대량으로 감축하는 방식인데, 단일모달 에러 표면 가정기법(unimodal error surface assumption), 피라미드 계층적탐색 기법, 비트 해상도 조절 기법, 가변탐색영역 기법, 중간 멈춤기법, 단순화된 에러 비

교 방법 등을 포함한다^[2-8]. 반면에 무손실 움직임 예측 방법은 예측화질의 저하 없이 계산량만 줄이는 방법인데, 계산량의 감소가 이전의 손실 움직임 예측방법보다 현저히 떨어진다. 여기에는 기준 블록과 후보 블록의 블록합을 이용한 후보 제거 알고리즘(SEA: successive elimination algorithm) 및 그 변형 방법들^[9]과 부분 매칭 에러값을 이용한 후보 제거(PDE: partial distortion elimination) 알고리즘 및 그 응용방법 등이 있다^[10-12]. 손실 움직임 예측 방법들은 예측화질과 계산량의 감소를 향상하는 연구가 계속 수행되고 있으며, 무손실 움직임 예측 방법은 계산량의 감축을 위한 연구가 수행되고 있다.

본 논문에서는 예측화질이 전영역 탐색방법과 거의 또는 완전히 동일하면서 계산량을 현저히 줄이는 고속 움직임 예측 방법을 제안한다. 제안하는 방법은 위에서 언급한 PDE 방법을 기반으로 하며, 무의미한 탐색 후보지점을 조기에 제거하면서 움직임벡터를 더 일찍 계산하도록 한다. 이는 탐색 절차를 몇 단계로 나누는데, 먼저 탐색 영역내의 각 후보 지점에 대하여 부분 블록 에러 합을 계산하고, 이에 따라 다음 단계에서 진행할 후보들을 선별하고, 최소 에러 지점의 최적후보에 대해 단계별 안정성을 비교 판단하여 그 다음 단계의 진행 여부를 결정함으로써 최적의 움직임 벡터를 계산한다. 다음 단계에서 진행할 후보들을 선별할 때 그때까지 계산한 부분 블록 에러 합을 이용하여 적절한 문턱치를 설정하고 이보다 작은 값을 갖는 후보에 대해서만 우선적으로 다음단계에서 탐색을 계속하고, 나머지는 모든 단계를 마친 후에 처리하는데, 이를 통하여 최적 움직임 벡터로서 가능성이 높은 후보에 대해 먼저 검사함으로써 추후에 불필요한 계산을 더 많이 줄일 수 있는 것이다. 또한 단계별로 진행하면서 각 단계에서 최적 후보가 변경되지 않으면 이를 최종 움직임 벡터로 판단하고 나머지 탐색을 종료하게 된다. 이는 예측 화질의 손실을 초래할 수 있는데, 변경되지 않은 단계의 수를 조절함으로써 예측화질을 제어할 수 있다. 제안하는 방법은 각 단계에서 최적 후보가 변경되지 않는 수를 조절하여 손실 예측 화질과 무손실 예측 화질을 쉽게 설정할 수 있는 장점을 가진다. 또한 제안하는 알고리즘은 기존의 손실 움직임 예측 방법들과 결합하여 사용할 때 추가적인 계산을 하지 않기 때문에 이들 방법들과 결합하여 사용할 때, 예측화질 대비 많은 계산

a) 부경대학교 IT융합응용공학과(Dept. of IT Conv.&App. Engineering, Pukyong National University)

‡ Corresponding Author : 김종남(Jong Nam Kim)

E-mail: jongnam@pknu.ac.kr

Tel: +82-51-629-6259

ORCID:https://orcid.org/0000-0002-6332-4905

※ 이 논문의 연구결과 중 일부는 "IPIU 2018"에서 발표한 바 있음.

※ 본 연구는 2015-16년도 부경대학교 연구년 사업으로 지원됨.

※ The work was supported by the project of PKNU Reseach Year.

· Manuscript received June 30, 2018; Revised August 13, 2018; Accepted August 14, 2018.

량 감소를 얻을 수 있다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 기존의 관련 연구들을 소개하며, 3장에서는 본 논문의 제안 알고리즘을 기술한다. 4장에서는 제안 알고리즘 성능 평가를 위한 실험 결과 및 분석, 5장에서 결론으로 마무리를 할 것이다.

II. 관련연구

전술한 바와 같이, 블록 매칭 기반의 움직임 예측 방법 중에 전영역 탐색방법이 가장 기본적인 개념이며, 이를 통하여 수많은 방법들이 도출되었다. 전영역 탐색방법은 설정된 탐색영역내의 모든 탐색후보에 대해 블록매칭에러를 계산하여 최소의 에러값을 가진 후보를 찾는 방법인데, 에러값을 계산할 때 절대 오차의 합(SAD: sum of absolute difference)을 가장 많이 사용한다. $N \times N$ 의 화소 크기를 갖는 매칭 블록(matching block)에서 블록매칭에러합 $SAD(x,y)$ 는 식 (1)과 같다.

$$SAD(x,y) = \sum_{i=1}^N \sum_{j=1}^N |f_t(i,j) - f_{t-1}(i+x, j+y)| \quad (1)$$

식 (1)에서, $f_t(i,j)$ 는 t시간 영상프레임에서 (i, j)위치에서 화소값을 나타내고, $f_{t-1}(i+x, j+y)$ 는 t-1 시간 영상프레임에서 (i+x, j+y) 위치에서 화소값을 나타낸다.

현재까지 많은 고속 움직임 예측 방법들 중에서 손실 움직임 예측방법이 거의 대부분을 차지하며^[2-8], 무손실 움직임 예측 방법^[9-11]들은 몇 가지에 불과하다. 대표적인 무손실 예측 방법으로 연속 제거 알고리즘 (SEA: Successive Elimination Algorithm)방법^[9]과 부분 오차 제거(PDE: Partial Distortion Elimination) 방법들이 있는데, SEA 방법은 후보 지점의 블록합을 중복계산을 이용하여 고속으로 계산하고, 그 블록합이 특정 기준을 만족하지 못하면 그 탐색 후보지점은 제거됨으로써 예측화질의 저하 없이 계산감축을 얻는다^[10-12]. 이는 블록합을 계산할 때 탐색영역 내에서 탐색후보 전체에 대해서 수행해야 연산 효율이 높아지는데, 다른 손실 움직임 예측방법들과 결합하여 사용할 때 추가적인 계산 손실이 많을 수 있다. PDE방법에서는 계산하는

부분 매칭 블록의 에러의 중간 합이 그때까지의 최소 에러 위치의 전체 블록의 합보다 크다면 이는 움직임 벡터의 후보가 불가하다고 판단하고, 그 이후의 나머지 계산을 중단함으로써 불필요한 계산을 줄이는 것이다. 이 방법은 탐색 후보에 대해서만 부분계산을 수행할 수 있기 때문에 탐색 영역내의 후보 개수에 상관없으며, 이 때문에 기존의 고속 알고리즘들과 결합하여 사용 시 추가적인 계산 없이 불필요한 계산을 제거할 수 있다. PDE를 기반으로 한 응용 방법들은 블록 매칭에 참여하는 영상의 블록 부분을 분석하여 각 부분별로 계산 우선순위를 설정하고, 그 순서로 블록 매칭 에러를 계산함으로써 불가능한 후보 지점을 더 빨리 제거하도록 하였다^[10-12]. 현재까지 무손실 움직임 예측에 대한 연구결과들이 발표되어 왔지만, 여전히 계산 감축에 대한 성능향상의 요구가 계속되고 있다.

III. 제안 알고리즘

본 논문에서 제안하는 알고리즘은 예측화질에 있어서는 전영역 탐색방법의 결과와 거의 동일하면서 계산량을 효율적으로 감축하는 것이다. 전술한 대로, 전영역 탐색방법과 비교하여 예측화질의 손실이 발생하지 않으면서 계산량을 줄일 수 있는 방법으로 SEA와 PDE방법이 있다. 이 중에서 PDE 방법은 불가능한 후보를 제거하기 위해 부가적인 계산을 하지 않기 때문에 기존의 고속 움직임 예측 방법들과 연계하여 사용할 수 있으며, 계산량의 감축에 있어 우수한 방법이다. PDE 기반의 변경된 고속 움직임 예측 방법들은 블록 매칭에러를 계산함에 있어 블록 영역의 계산 순서를 결정하고, 그 선별된 순서대로 블록 매칭 에러를 구함으로써 고정된 순서로 매칭에러를 구하는 것 보다 불필요한 계산을 더 줄일 수 있도록 하였다. 또한 식 (3)에서 SAD값과 비교하지 않고 SAD 값에 일정 상수를 곱한 결과와 비교하여 계산량 감축을 많이 얻도록 하였으나 이는 예측화질의 저하를 초래할 수 있는 심각한 문제점을 가지고 있는 방법이다.

식 (2)와 (3)은 PDE 방법에서 사용되는 블록매칭에러인 SAD값의 부분비교를 나타낸 것이다. 여기서 $PSAD_k$ 는 매칭블록에서 k행까지의 누적된 부분 블록매칭에러의 값이

다.

$$PSAD_k = \sum_{i=1}^k \sum_{j=1}^N \left| f_t(i, j) - f_{t-1}(i+x, j+y) \right|, \quad (2)$$

where $k = 1, 2, \dots, N$

$$PSAD_k \leq SAD_{\min}, \quad \text{where } k = 1, 2, \dots, N \quad (3)$$

식 (2)에서, $f_t(i, j)$ 는 t 시간의 영상 (i, j) 위치에서 화소값을 나타내고, $f_{t-1}(i+x, j+y)$ 는 t-1 시간의 영상 (i+x, j+y)에서 화소값을 나타낸다. 여기서 $PSAD_k$ 는 매칭블록에서 k행까지의 누적 부분 블록 매칭 에러이다. N은 매칭 블록의 크기를 나타내며, SAD_{\min} 은 계산 시점에서 최소의 블록 매칭 에러인 SAD 값을 의미한다. $k < N$ 일 때, 부분 매칭에러 합이 그때까지의 최소 에러보다 크다면 $k+1$ 부터 N까지의 행에 대한 매칭에러 계산을 중단하고 다음 후보점으로 옮겨갈 수 있기 때문에 그 만큼의 계산량 절감을 얻을 수 있다. 식(3)에서 SAD_{\min} 을 가능한 일찍 찾을 수 있다면 더 많은 후보들이 조기에 제거될 것이다.

본 논문에서 제안하는 방법은 기존의 블록매칭에러값이 큰 영역을 유추하는 것이 아니라 단계별로 최소에러를 갖는 최적의 후보지점을 추가적인 계산 없이 가능한 일찍 찾아내어 불가능한 후보 벡터를 더 빨리 제거한다. 탐색영역 내에 존재하는 최소의 블록 에러 지점을 더 빨리 찾기 위하여 부분블록에러합의 결과와 적절한 조건을 이용하여 최적의 후보를 대상으로 전체 블록에러를 계산함으로써 탐색영역내의 최소에러 지점을 더 빨리 계산한다. 기존의 전영역 탐색 및 PDE, 그리고 PDE 변경 알고리즘들은 각 후보지점에 대하여 전체의 블록에러합을 계산을 한 후 다음 후보지점으로 이동하여 블록 에러합을 계산하는 구조를 가지고 있다. 이러한 방법은 탐색영역 내에서 블록매칭에러를 순차적으로 계산을 다 해보아야 최소에러 지점을 알 수 있게 되는 것이다. 또한 중간에 부분 블록 에러합을 갖는 위치가 변경이 되지 않는다면 k의 값이 N에 도달하기 전에 그 지점을 최종적인 움직임 벡터의 지점으로 간주하여 계산량을 대폭 줄일 수도 있다. 이때 연속적으로 변하지 않는 부분 블록에러 합을 단계수에 대한 문턱치를 조정하여 효율적으로 계산량을 조절할 수 있다.

본 논문에서는 탐색 영역전체에 대하여 부분블록에러의 합을 단계적으로 계산한다. 단계적으로 블록에러의 합을 계산하는 것은 그 블록의 행이나, 열, 또는 작은 서브 블록 별로 계산을 하는 것을 의미하며, 유효한 탐색 후보지점에 대해 작은 단위의 매칭 에러합을 계산한다. 한 지점에서 한 번에 블록매칭에러의 값을 계산하는 것이 아니라 매칭블록에서 작은 단위의 매칭영역 개수의 단계로 나누어서 실행을 하며, 각 단계는 탐색영역에 있는 전체 후보들을 대상으로 하는 것이다. 탐색 영역내에 있는 전체 후보지점에 대하여 불가능한 후보 벡터를 매 단계마다 제거하며, 제거되지 않고 남은 후보에 대하여 다음단계에서 같은 과정을 반복하게 된다. 이때, 제거되지 않고 남아서 다음 단계에서 계산해야 할 후보들 중에서 그 부분 블록 에러합의 크기에 따라 문턱값을 설정하고, 설정된 그 문턱값보다 작은 부분 블록 에러값을 갖는 후보지점에 대하여 다음 단계에서 계산을 한다면 효율적으로 최소에러의 후보지점을 더 빨리 찾아낼 수 있다. 그 문턱치는 식 (4)에서처럼 현 단계에서 최소값과 최대값 및 적절한 상수값을 이용하여 탐색영역별로 적응적으로 설정할 수 있다.

$$TH_{pk} = (\min(PSAD_k) + (SAD_k) * k / N) * \alpha \quad (4)$$

where $k = 1, 2, \dots, N$

$$TH_t = \beta \quad (5)$$

식 (4)에서, $\min(PSAD_k)$ 는 매칭블록에서 k행까지의 누적 부분 블록 매칭 에러값들 중에서 최소값을 의미하며, SAD_k 는 그 단계에서 최소부분블록의 에러값을 갖는 위치에서의 블록 매칭 에러값을 의미하고, α 는 특정상수를 의미한다. 이 식에서 보듯이, 문턱치 계산을 위하여 추가적으로 계산하는 항목이 없으며, 각 단계에서 절차적인 진행을 위하여 반드시 계산해야 하는 $\min(PSAD_k)$ 와 SAD_k 를 이용한다. 각 단계에서 최소에러의 값에 대한 위치가 변경이 되면 그 변경된 위치에 대하여 전체의 블록 에러값을 계산하고, 이전의 블록 에러값과 비교하여 대소의 여부에 따라 최소에러의 지점이 변경되는 것이다.

식 (5)는 각 단계별로 최소 지점이 변하지 않는 $\min(PSAD_k)$ 위치의 부동성을 문턱치로 설정한 것이다. 계산량

을 대폭 줄이고자 한다면 $\min(\text{PSAD}_k)$ 위치의 부동성을 나타내는 문턱치 TH_t 값을 작게 설정하면 가능한데, 이때 값을 너무 작게 가진다면 영상에 따라 화질 저하에 대한 위험성을 가질 수 있다.

그림 1은 제안 알고리즘의 진행절차를 도식화한 것이다. 그림 1에서 PSAD_k 은 k 단계에서 부분 블록 에러의 합을 나타내며, SAD_k 은 k 단계에서 최적의 후보지점에서 블록 에러의 합을 나타낸다. 그림 1에서 단계 6은 더 많은 계산량 감소를 위하여 손실 움직임 예측을 할 수 있는데, 문턱치 값을 매칭 블록 크기인 N 으로 설정하면 무손실 예측 방법으로서 예측화질의 저하 없이 계산량만 줄일 수 있다.

IV. 실험 결과 및 고찰

제안 알고리즘의 객관적인 성능 평가를 위해, 720×480

화소를 가진 “bus”, “bally”, “bicycle”, “flower garden”, “football”과 1920×1080 화소를 가진 “cactus”, 4096×2160 화소를 가진 “people on street”의 비디오 데이터를 이용하여 실험을 수행하였다. 이들 비디오 데이터는 표준적으로 사용하고 있는 영상으로서 영상의 공간 복잡도, 움직임의 크기, 움직임의 복잡도를 고려하여 작성되었다. 움직임 벡터를 계산하는 매칭 블록의 크기는 16×16 화소이며, 탐색 영역의 범위는 ± 15 화소, 비디오 프레임수는 95이다. 알고리즘의 성능평가를 위한 실험 결과는 움직임 벡터 계산을 위해 사용된 평균 계산량과 전영역 탐색방법에서의 계산시간과 각 방법별 계산시간의 비, 그리고 움직임 벡터의 정확도와 예측화질 평가를 위하여 PSNR(Peak Signal to Noise Ratio)로 나타내었다. 각 방법에 대한 계산량은 16×16 화소의 매칭블록의 크기에서 계산을 위해 사용된 행의 수로써 나타내었다. 또한, 식 (4)에서 PSAD_k 문턱치 계산을 위한 상수는 0.6을 사용하였다. 이의 값이 커지면 더 많은 후보들

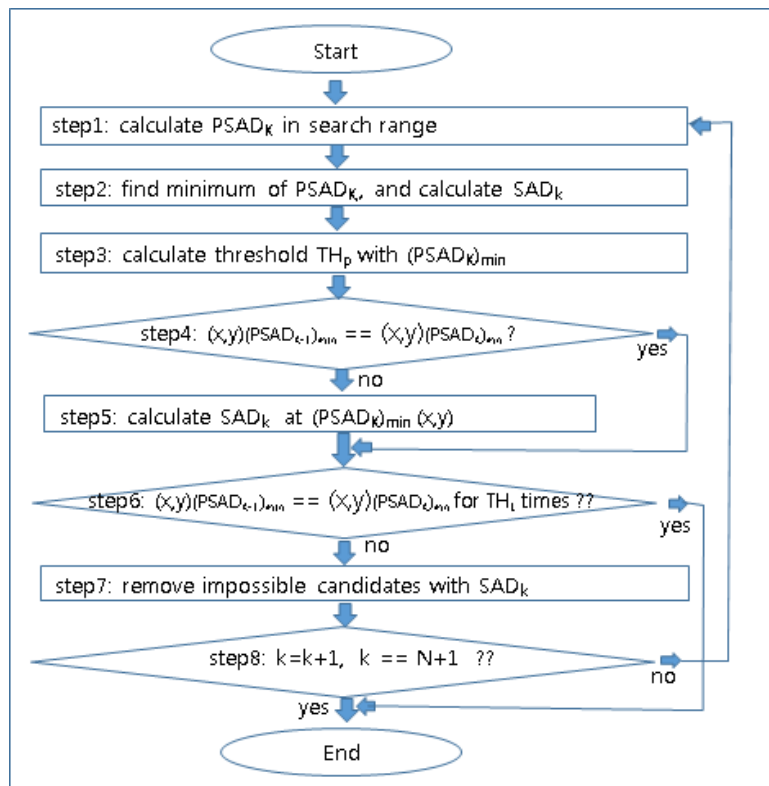


그림 1. 제안 알고리즘의 절차
Fig. 1. Procedure of the proposed algorithm

에 대하여 부분 블록 매칭 에러 계산을 하며, 값이 작아진다면 계산 대상자를 줄일 수 있다. 이의 값은 적절한 범위 내에서 제어되어야 하며, 이의 값이 너무 작거나 크다면 부분 블록 매칭 에러를 구하는 후보들이 더 많아질 수 있다. 또한 단계 6에서 $\min(\text{PSAD}_k)$ 의 부동성 문턱값은 경험치를 이용하여 4, 8, 16을 각각 사용하였다. 이의 문턱값이 16 미만인 경우는 손실 예측이 발생 할 수 있는 경우이며, 적절히 선택할 경우 예측화질 대비 계산량을 많이 줄일 수 있다.

본 논문에서 제안하는 방법은 무손실 예측 방법과 거의 동일한 예측화질을 추구하며, 계산량은 대폭 줄이도록 하였다. 각 방법들의 비교기준이 되는 무손실 움직임 예측 방법으로는 전영역 알고리즘 (FS)이 있으며, 고속의 무손실 움직임 예측방법으로는 PDE 알고리즘 (PDE) [11], Hada-

pard 변환을 이용한 고속 PDE 방법 (PDE HD)^[12], MSEA (Multilevel Successive Elimination Algorithm)^[9], CDMHS (Cross-Diamond Modified Hierarchical Search)^[6], ARTZS (Angle Restricted Test Zone Search)^[8] 등이 있다. CDMHS와 ARTZS 방법을 제외한 나머지 방법들은 무손실 움직임 예측 방법으로서 기존의 전영역 탐색방법과 비교하여 예측 화질의 저하가 전혀 없으며, 계산량만 줄이는 방법들이다. CDMHS와 ARTZS 방법은 본 논문에서 제안하는 방법이 기존의 고속 움직임 방법들과 연동하여 사용가능하며, 예측화질 대비 계산량을 줄이는 방법임을 보이기 위하여 실험방법에 포함시켰다.

표 1과 2는 계산 복잡도를 나타낸 것으로, 각 움직임 예측 방법에 대하여 각 비디오 시퀀스별로 움직임 벡터 계산

표 1. 알고리즘별 계산량(단위: 행)

Table 1. Computational amount of each algorithm for sequences (Unit:rows)

	bus	bally	bycl	flower	football	cactus	people
FS	16	16	16	16	16	16	16
PDE	9.9	7.5	9.5	6.6	7.5	10.7	9.2
PDE HD	9.0	6.5	8.3	5.7	6.7	9.9	8.2
MSEA	10.9	8.4	10.6	7.8	10.2	10.9	10.1
ARTZS	0.9	0.8	0.9	0.8	0.8	1.0	0.9
ARTZS+ Proposed $TH_i=8$	0.5	0.4	0.5	0.4	0.4	0.5	0.4
CDMHS	1.1	1.0	1.1	1.0	1.0	1.1	1.0
CDMHS+ Proposed $TH_i=8$	0.3	0.3	0.3	0.3	0.3	0.4	0.3
Proposed $TH_i=4$	6.6	5.0	6.1	4.2	5.0	7.2	6.2
Proposed $TH_i=8$	7.2	5.6	7.0	4.4	5.6	7.6	6.5
Proposed $TH_i=16$	7.6	5.9	7.4	4.5	5.8	8.1	6.9

표 2. 알고리즘별 상대적 계산 시간 비율 (단위: %)

Table 2. Rate of Computational time of each algorithm for sequences (Unit: %)

	bus	bally	bycl	flower	football	cactus	people
FS	100	100	100	100	100	100	100
PDE H.264	65.3	58.5	67.5	56.7	58.8	70.1	63.0
PDE HD	60.0	53.5	61.3	52.7	56.7	59.4	56.6
MSEA	69.9	65.4	71.6	61.8	66.8	73.7	69.1
ARTZ	18.1	15.9	19.6	16.5	15.5	20.1	17.4
ARTZ+ Proposed $TH_i=8$	13.3	11.7	14.3	12.0	11.5	15.9	13.7
CDMHS	20.3	19.4	22.6	18.5	20.3	21.4	19.0
CDMHS+ Proposed $TH_i=8$	15.4	15.1	16.6	14.4	15.3	16.8	15.0
Proposed $TH_i=4$	41.6	42.0	44.1	41.2	41.0	44.2	42.3
Proposed $TH_i=8$	46.2	45.6	48.0	44.4	45.6	49.6	46.7
Proposed $TH_i=16$	52.6	46.9	54.7	47.5	50.8	56.8	49.8

표 3. 알고리즘별 예측화질(단위: dB)

Table 3. Prediction quality of each algorithm for sequences (Unit: dB)

	bus	bally	bycl	flower	football	cactus	people
FS ~ MSEA	26.0	31.5	23.5	27.7	36.5	28.1	29.6
ARTZ	24.0	29.9	22.1	24.9	35.0	26.5	27.7
ARTZ+ Proposed $TH_t=8$	23.9	29.7	22.0	24.8	34.9	26.4	27.6
CDMHS	24.4	30.2	22.0	25.4	35.2	27.2	28.5
CDMHS+ Proposed $TH_t=8$	24.3	30.2	22.0	25.3	35.2	27.1	28.4
Proposed $TH_t=4$	25.8	31.4	23.4	27.6	36.5	27.9	29.2
Proposed $TH_t=8$	25.9	31.5	23.5	27.7	36.5	28.0	29.4
Proposed $TH_t=16$	26.0	31.5	23.5	27.7	36.5	28.1	29.6

을 위한 평균 연산량과 연산시간의 전영역 탐색 대비 비율을 나타내었고, 표 3은 움직임 벡터를 이용한 예측화질 결과를 나타내었다. 표 1은 영상 시퀀스별 각 알고리즘의 페이지완전 계산량을 나타낸 것인데, 표에서 보는바와 같이 전영역 탐색방법의 계산량은 블록의 전체행의 수를 사용하기 때문에 매칭블록 크기인 16을 나타낸다. 나머지 방법들은 고정된 계산량이 아니고 영상에 따라 가변적임을 알 수 있다. 표 2는 전영역 탐색 방법의 계산시간을 100%로 볼 때 각 방법별 계산시간을 비율로 나타내었다. 본 논문에서 제안하는 알고리즘은 TH_t 값에 따라 예측화질의 저하가 발생하기도 하며, TH_t 값이 16일 경우는 전영역 탐색 방법과 비교하여 예측화질 저하가 없는 무손실 예측화질 방법이 된다.

Complex PDE 와 PDE HD방법은 최소 에러 위치를 먼저 찾는 것이 아니라 불가능한 후보 벡터를 먼저 찾는 개념으로 접근하며, 이들은 블록의 부분 에러합이 큰 영역을 계산해내고 이의 순서대로 블록의 부분 에러합을 계산함으로써 가능한 일찍 블록의 부분 에러합이 큰 영역을 계산하게 하는 것이다. FS방법의 결과와 비교해 볼 때, 다른 고속의 무손실 예측방법들은 많은 연산량 감소가 있음을 알 수 있으며, 무손실 방법 중에서 제안 방법의 연산량 감소가 가장 많음을 알 수 있다. 또한 제안하는 방법의 TH_t 값에 대해 연산량 대비 예측화질의 성능이 우수함을 알 수 있다. 표 1에서 CDMHS은 손실방법의 하나인데, 다른 방법에 비하여 연산량 감소가 많음을 알 수 있으며, 이의 방법과 제안방법이 결합된 결과는 연산량 감소가 더욱 많음을 알 수 있다.

표 3은 모든 알고리즘의 예측화질 결과를 나타냈었는데, 비교 방법들은 전영역 탐색 방법에 비하여 예측화질의

저하가 발생하는 경우와 없는 경우로 나누어진다. 표 3에서 보듯이, FS, PDE H.264, Complex PDE, PDE HD, Proposed $TH_t=16$ 의 방법들은 화질저하가 FS방법에 비하여 없으며, 나머지는 손실이 발생함을 알 수 있다. CDMHS 방법과 ARTZS방법은 연산량 감소는 많으나 무손실 방법에 비하여 예측화질 저하가 현저히 발생하며, CDMHS+ Proposed $TH_t=8$ 방법과 ARTZS+Proposed $TH_t=8$ 방법은 CDMHS와 비교하여 예측화질저하가 극히 미미함을 알 수 있으며, 무손실 방법들과 Proposed $TH_t=4,8$ 방법들을 비교해 보면 예측화질의 저하가 미미함을 알 수 있다.

V. 결론

본 논문에서는 움직임 예측에서 전영역 탐색방법에 비하여 예측화질의 저하가 전혀 없거나 거의 없으면서 소요되는 계산량을 현저히 줄이는 방법을 제안하였다. 본 논문에서 제안한 방법은 탐색 영역의 후보지점 중에서 단계별 최소 에러를 갖는 최적의 후보를 찾고, 이를 통하여 얻는 최소의 블록매칭에러를 계산하고 단계별 부분블록매칭에러의 크기에 따라 다음 단계에서 우선 순위를 갖는 후보를 걸러내며, 단계별 부분블록매칭에러 지점이 변치 않는 횟수를 조사하여 그에 따라 최종 움직임벡터를 결정함으로써 불필요한 계산량을 현저히 줄일 수 있게 하였다. 본 논문의 제안 방법은 전영역 탐색 알고리즘과 비교하여 예측 화질의 저하가 전혀 없이 계산량을 최고 75%를 줄였으며, 미미한 화질저하를 감수한다면 더 많은 계산량을 줄일 수 있었다. 또한 다른 손실 예측 방법들과 결합하여 사용시 더욱 많은

계산량을 줄일 수 있으며, 결합된 기존의 방법들과 비교할 때 예측화질저하는 미미함을 알 수 있었다. 제안한 알고리즘은 MPEG-4 AVC 및 HEVC에서 사용이 가능하며, 이를 이용하는 실시간 비디오 압축 응용분야에 유용하게 사용될 수 있을 것이다.

참 고 문 헌 (References)

- [1] T. Tan, R. Weerakkody, and G. Sullivan, "Video quality evaluation methodology and verification testing of HEVC compression performance," *IEEE Transactions on Circuits System & Video Technology*, Vol. 26, No. 1, pp. 76-90, 2016.
- [2] T. Lee, Y. Chan, and W. Siu, "Adaptive search range by neighbouring depth intensity weighted sum for HEVC texture coding," *IEE Electron. Letters*, Vol. 52 No. 12, pp. 1018-1020, 2016.
- [3] H. Choi, J. Kim, S. Jung, "Fast Motion Estimation Algorithm using Importance of Search Range and Adaptive Matching Criterion," *The Journal of Korea Institute of Signal Processing and Systems*, Vol. 16, No. 4, pp. 129-133, 2015.
- [4] Z. Pan, j. Lei, Y. Zhang, X. Sun, and S. Kwong, "Fast motion estimation based on content property for low-complexity H.265/HEVC encoder," *IEEE Transactions on Broadcasting*, Vol. 63, No. 3, pp. 675-684, 2016.
- [5] P. Bhalge and S. Amdani, "Modified hexagonal search for motion estimation", *Proceeding of International Conference on Intelligent Computing and Control Systems*, pp. 94-96, 2017.
- [6] N. Alnajdawi, M. Alnajdawi, and S. Tedmori, "Employing a novel cross-diamond search in a modified hierarchical search motion estimation algorithm for video compression," *Elsevier Information Sciences*, Vol. 268, pp. 425-435, 2014.
- [7] A. Paramkusam, "Efficient motion estimation algorithm on the layers," *IEE Electron. Letters*, pp. 467-468, 2017.
- [8] N. Vayalil, M. Paul, and Y. Kong, "A novel angle-restricted test zone search algorithm for performance improvement of HEVC", *Proceeding of IEEE International Conference on Image Processing*, pp. 6-10, 2017.
- [9] X.Q. Gao, C.J. Duanmu, and C.R. Zou, "A Multilevel Successive Elimination Algorithm for Block Matching Motion Estimation," *IEEE Transactions on Image Processing*, Vol. 9, No. 3, pp. 501-504, 2000.
- [10] J. Kim, S. Byun, Y. Kim, and B. Ahn, "Fast Full Search Motion Estimation Algorithm Using Early Detection of Impossible Candidate Vectors," *IEEE Transactions on Signal Processing*, Vol. 50, No. 9, pp. 2355-2365, 2002.
- [11] H.264/AVC reference software, http://iphome.hhi.de/suehring/tml/download/old_jm/
- [12] S. Jin and H. Lee, "Fast Partial Distortion Elimination Algorithm based on Hadamard Probability Model," *IEE Electron. Letters*, Vol. 44, No. 1, pp. 17-19, 2008.

저 자 소 개



김 종 남

- 1997년 2월 : 광주과학기술원 정보통신공학과 졸업(공학석사)
- 2001년 8월 : 광주과학기술원 기전공학과 졸업(공학박사)
- 2001년 8월 ~ 2004년 2월 : KBS 연구원
- 2005년 12월 ~ 2006년 12월 : 미국 뉴욕시립대(CUNY) 방문 교수
- 2015년 7월 ~ 2016년 7월 : 미국 노스캐롤라이나주립대(NCSU) 방문 교수
- 2004년 3월 ~ 현재 : 부경대학 IT융합응용공학과 교수
- ORCID : <https://orcid.org/0000-0002-6332-4905>
- 주관심분야 : 비디오압축, 영상처리, 컴퓨터비전 등