JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지(J. Korea Inst. Inf. Commun. Eng.) Vol. 20, No. 8 : 1531~1536 Aug. 2016

단계별 최소에러를 통한 고속 움직임 예측 알고리즘

김종남*

Fast Motion Estimation Algorithm via Minimum Error for Each Step

Jong Nam Kim*

Department of IT Convergence and Application Engineering, Pukyong National University, Busan 48513, Korea

요 약

본 논문에서는 비디오 부호화에서 중요한 요소인 움직임 예측을 위한 고속 알고리즘을 제안한다. 기존의 고속 움직임 예측 방법들이 연구되어 왔지만 여전히 연산량 감축에 대한 문제점을 가지고 있다. 본 논문에서는 전영역 탐색 기반의 방법에 비하여 예측화질은 같게 유지하면서 불필요한 계산량을 현저히 줄이는 알고리즘을 제안한다. 제안하는 방법은 움직임 벡터를 찾기 위해 후보 벡터의 블록에러합을 계산해 갈 때 각 후보지점에서 한 번에 블록에러합을 전부 계산하는 것이 아니라 탐색 영역에 있는 모든 화소에 대해 몇 단계로 나누어 부분 블록에러합을 계산하고 이를 통하여 전체의 최소에러를 갖는 지점을 일찍 유추하여 불필요한 계산량을 줄임으로써 계산속도의 향상을 얻는다. 제안한 알고리즘은 전영역 탐색 알고리즘과 같은 예측화질을 갖는 기존의 고속 알고리즘과 비교하여 더 적은 계산량을 사용한다.

ABSTRACT

In this paper, we propose a fast motion estimation algorithm which is important in performance of video encoding. Even though so many fast algorithms for motion estimation have been published due to its tremendous computational amount of for full search algorithm, efforts for reducing computations in motion estimation still remain. In the paper, we propose an algorithm that reduces unnecessary computations only, while keeping prediction quality the same as that of the full search. The proposed algorithm does not calculate block matching error for each candidate at once to find motion vectors but divides the calculation procedure into several steps and calculates partial sum of block errors. By doing that, we can estimate the minimum error point early and get the enhancement of calculation speed by reducing unnecessary computations. The proposed algorithm uses smaller computations than conventional fast search algorithms with the same prediction quality as full search.

키워드 : 움직임 추정, 전역 탐색, 탐색영역, 매칭 에러, 부분블록에러합

Key word : Motion estimation, Full search, Search rang, Matching error, Partial sum of block error

Received 19 April 2016, **Revised** 21 April 2016, **Accepted** 25 May 2016

* **Corresponding Author** Jong Nam Kim (E-mail:jnkim1225@gmail.com, Tel:+82-51-629-6259)
Department of IT Convergence and Application Engineering, Pukyong National University, Busan 48513, Korea

Open Access http://dx.doi.org/10.6109/jkiice.2016.20.8.1531 print ISSN: 2234-4772 online ISSN: 2288-4165

ⓒThis is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(http://creativecommons.org/li-censes/by-nc/3.0/) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright Ⓒ The Korea Institute of Information and Communication Engineering.

Ⅰ. 서 론

비디오 부호화에서 움직임 예측은 비디오 데이터의 시간적 중복성을 줄이는 핵심 기술이며, 이는 연속되는 영상 데이터간의 유사한 영역을 찾는 것이다. 연속되는 두 프레임의 움직임을 예측하는 방법들 중에 블록 기반의 움직임 예측 방법은 계산량 대비 예측화질이 우수하여 현재까지의 동영상 압축 표준에 널리 사용되고 있다. 블록 기반 움직임 예측 방법에서 가장 기본적인 전영역 탐색 방법은 주어진 탐색 영역에서 모든 후보점에 대하여 매칭 에러를 계산하고 최소의 매칭 에러를 갖는 후보 지점을 찾는 것이다. 이 움직임 예측 방법은 구조가 간단하고 구현하기가 용이한 반면 그 계산량이 많기 때문에 이를 해결하기 위해 수많은 연구가 수행되어 왔다[1].

움직임 예측을 위한 수많은 고속 알고리즘들은 크게 두 그룹으로 나누어 질 수 있다. 하나는 전영역 탐색 방식에 비해 예측 화질의 손실을 갖는 것이고, 다른 하나는 예측 화질의 손실을 갖지 않는 방식이다[1]. 본 논문에서 이에 대한 정의를 한다면 전자는 손실 움직임예측이라 정의하고, 후자는 무손실 움직임예측이라 정의한다. 계산량의 감축이 많은 손실 움직임 예측 방법은 다시 몇 가지 개념으로 나누어 질 수 있다. 단일모달에러 표현 가정기법, 다해상도 계층적탐색, 낮은 비트 해상도, 가변 탐색법, 문턱치에 의한 중간 멈춤기법, 간소화된 에러 비교 방법, 복합방법 등이 이에 포함된다[1-6]. 그리고 무손실 움직임 예측 기법에는 기준 블록과 후보 블록의 블록합을 이용한 후보 제거 알고리즘(SEA: successive elimination algorithm) 및 그 변경 방법들[7], 그리고 부분 매칭 에러값을 이용한 후보 제거(PDE: partial distortion elimination) 알고리즘 및 그 응용방법 등이 있다[8-10]. 무손실 움직임 예측 방법의 문제점이 여전히 남아있는데, 이는 예측 화질은 전영역 탐색방법과 같지만 계산량의 감축이 많지 않다는 것이다.

본 논문에서는 예측화질은 전영역 탐색방법과 동일하면서 계산량을 현저히 줄이는 무손실 고속 움직임 예측 방법을 제안한다. 제안하는 방법은 PDE 방법을 기반으로 하며, 불가능한 후보 벡터를 제거하면서 움직임 벡터를 더 일찍 검출하도록 한다. 이는 각 후보점 각각에 대하여 블록매칭에러를 전부 계산하는 것이 아니고 이를 여러 단계로 나눈다. 주어진 탐색 영역에 있는 각 후보 화소에 대하여 각 단계별로 부분 블록 매칭에러를 계산하고 이의 결과에 따라 전체 블록 매칭에러 계산 여부를 결정한다. 이를 통하여 각 단계별 최소의 블록 에러 지점을 찾고, 불가능한 후보 벡터를 더 일찍 제거하여 계산 감축의 향상을 가져올 수 있다. 제안하는 방법은 전영역 탐색 움직임 예측방법의 예측화질과 동일하면서 이의 계산량 대비 현저히 감소된 계산량을 사용하며, 기존의 고속 무손실 움직임 예측 방법에 비해서도 더 적은 계산량을 사용한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 본 논문의 관련 연구로서 기존의 움직임 추정 알고리즘을 소개할 것이다. 3장에서는 본 논문의 제안 알고리즘을 소개하며, 4장에서는 본 논문의 알고리즘 성능 평가를 위한 실험 결과 및 분석이 기술될 것이다. 마지막 5장에서 결론으로 마무리를 할 것이다.

Ⅱ. 기존의 움직임 예측 알고리즘들

움직임 예측 방법에서 가장 기본적으로 사용하는 전영역 탐색(full search)방법이며, 블록매칭에러를 측정하는 많은 방법들이 제안되었지만 이들 가운데 절대 오차의 합(SAD: Sum of Absolute Difference)이 가장 많이 쓰인다. N×N의 크기를 갖는 매칭 블록(maching block)에서 매칭블록에러합 $SAD(x,y)$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 영상에서의 매칭블록 (i, j) 위치에서 화소값을 나타내고, $f_{t-1}(i+x, j+y)$는 $t-1$ 시간의 $(i+x, j+y)$ 위치에서 화소값을 나타낸다.

전영역 탐색의 문제점인 방대한 계산량을 줄이기 위하여 수많은 고속 움직임 예측 알고리즘들이 연구되었다. 그 중에서도 전 영역 탐색 방식에 비하여 예측 화질의 저하가 있는 손실 고속 움직임 예측알고리즘이 거의 대부분을 차지한다. TSS (Three Step Search), DS(Diamond Search), HXS(Hexgonal Search)등이 있다. 다해상도 탐색기법(Multi-resolution Motion Estimation), 문턱치값을 이용한 멈춤기법, 가변 탐색영역기법등 많은 방법들에 대한 연구결과가 발표되어 있다. 이들 방법은 계산 감축이 많지만 움직임이 많거나 복잡한 영상

에서는 예측화질의 저하가 심해지는 문제점을 가지고 있다[1-6].

기존의 전역 탐색방법과 비교하여 예측화질의 손실이 발생하지 않는 무손실 방법으로 대표적인 방법으로는 연속 제거 알고리즘 (SEA: Successive Elimination Algorithm)계열과 부분 오차 제거(PDE: Partial Distortion Elimination) 계열 알고리즘이 있다. SEA 계열 방법은 후보 지점의 블록합을 고속으로 계산하여 그 블록합이 특정 기준을 만족하지 못하면 그 지점은 블록매칭에러합(SAD)의 계산에서 제거됨으로써 예측화질의 저하 없이 계산감축을 얻는 방법이다[7]. PDE방법에서는 계산하는 매칭 블록의 에러의 중간 합이 그때까지의 최소 에러위치의 전체 블록의 합보다 크다면 그 이후의 나머지 계산을 중단하고 그 후보벡터는 불가능한 벡터라고 결정함으로써 불필요한 계산을 줄이는 것이다. 이후에 계산량 감축을 위한 방법들은 주로 블록의 매칭에 있어서 매칭 에러가 큰 영역을 유추하여 그 영역의 순서대로 블록 에러를 계산함으로써 예측화질의 저하 없이 계산 속도의 개선을 가져 올 수 있었다. PDE를 기반으로 한 무손실 고속 움직임 예측 방법들은 블록 매칭에 참여하는 블록의 부분을 영상 분석 방법을 이용하여 선별하고, 고정된 순서로 매칭 에러를 구하는 것이 아니라 선별된 순서로 블록 매칭을 함으로써 불필요한 계산을 더 빨리 제거하도록 하였다[8-10]. 즉, 블록 매칭내에서 어떤 영역을 블록 매칭 에러 계산에 먼저 참여 시키느냐에 따라 움직임 예측을 위한 계산을 더 줄일 수 있다는 것이다.

기존의 무손실 움직임 예측에 대한 연구에 대해 많은 연구결과들이 발표되어 계산량 감축을 가져왔지만, 여전히 무손실 기반의 움직임 예측에 있어 계산 속도에 대한 성능향상의 요구가 계속 제기되고 있다.

Ⅲ. 제안한 움직임 예측 알고리즘

본 논문에서 제안하는 알고리즘은 예측화질에 있어서는 전영역 탐색방법의 결과와 같으면서 불필요한 계산량을 감축하여 계산 속도를 향상시키는 것을 목표로 한다. 전술한 바와 같이, 전역 탐색방법과 비교하여 예측화질의 손실이 발생하지 않으면서 계산량을 감소할 수 있는 무손실 방법으로 대표적인 방법이 PDE방법이다. 또한 PDE 기반의 변경된 고속 움직임 예측 방법들은 추가 계산을 통하여 블록 매칭에 참여하는 영역의 계산의 순위를 정하고, 그 선별된 순서로 매칭 에러를 구함으로써 고정된 순서로 매칭에러를 구하는 것 보다 불필요한 계산을 더 빨리 제거하도록 하였다.

식 (2)와 (3)은 PDE에서 사용되는 블록매칭에러인 SAD값의 행별 부분비교를 나타낸 것이다. 여기서 $APSAD_k$는 매칭블록에서 k행까지의 누적된 부분 SAD 값이다.

$$APSAD_k = \sum_{i=1}^{k}\sum_{j=1}^{N}\left|\begin{matrix} f_t(i,j) - \\ f_{t-1}(i+x, j+y)\end{matrix}\right|, \quad \text{where} \quad k = 1, 2, \cdots, N \tag{2}$$

$$APSAD_k \le SAD_{\min}, \quad \text{where} \quad k = 1, 2, \cdots, N \tag{3}$$

식 (2)에서, $f_t(i,j)$는 t 시간의 매칭블록 (i, j) 위치에서 화소값을 나타내고, $f_{t-1}(i+x, j+y)$는 $t-1$ 시간의 $(i+x, j+y)$에서 화소값을 나타낸다. 여기서 $APSAD_k$는 매칭블록에서 k행까지의 누적 부분 블록매칭에러이다. N은 매칭 블록의 크기를 나타내며, SAD_{min}은 계산 시점에서 최소의 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.

본 논문에서 제안하는 방법은 기존의 블록매칭에러값이 큰 영역을 유추하는 것이 아니라 최소에러의 지점을 추가적인 계산 없이 가능한 일찍 찾아내는 데에 초점을 맞추고 있다. 이를 위해서는 움직임 탐색 동작 구조의 변경을 필요로 한다. 기존의 전영역 탐색 및 PDE, 그리고 PDE 변경 알고리즘들은 각 후보지점에 대하여 전체의 블록에러합을 계산을 한 후 다음 후보지지점으로 이동하여 블록 에러합을 계산하는 구조를 가지고 있다. 이러한 구조로는 전체 탐색 영역내에서 최소의 블록에러합을 갖는 위치를 유추하기가 불가능하다. 탐색 영역 내에서 계산을 다 해보아야 알 수 있게 되는 것이다. 이를 위하여 본 논문에서는 탐색 영역전체에 대하여 부분블록에러의 합을 순차적으로 계산한다. 순차적으로 블록에러의 합을 계산하는 것은 그 블록의 행이나,

열, 또는 작은 서브 블록별로 계산을 하는 것을 의미하며, 식 (2)를 이용하여 이런 작은 단위의 매칭에러합을 계산한다. 한 지점에서 한 번에 계산하는 것이 아니라 매칭블록에서 작은 단위의 매칭영역의 개수만큼을 단계로 나누어서 실행을 하며, 각 단계는 탐색영역에 있는 전체 후보들을 대상으로 하는 것이다. 식 (3)을 이용하여 탐색영역내에 있는 전체 후보지점에 대하여 불가능한 후보 벡터를 매 단계마다 제거 후에 남은 후보에 대하여 다음단계에서 식 (2)와 식 (3)을 이용하여 같은 과정을 반복하게 된다. 물론 첫 단계에서 최소의 부분 블록 에러의 지점에 대하여 전체의 블록 에러 값을 계산해야 하며, 이를 가지고 식 (3)을 이용할 수 있다. 그리고 각 단계에서 최소에러의 값에 대한 위치가 변경이 되면 그 변경된 위치에 대하여 전체의 블록에러값을 계산하고, 이전의 블록에러값과 비교하여 대소의 여부에 따라 최소에러의 지점이 변경되는 것이다. 그림 1은 제안 알고리즘의 진행을 의사코드로 나타낸 것이다.

그림 1에서의 제안 알고리즘은 영상의 블록크기가 N일 때 전체의 단계를 N으로 나누고 이를 반복하면서 각 단계에서의 부분블록에러합과 그 위치를 구하는 것으로 시작한다. 이전 단계에서 최소의 부분 블록에러합의 위치를 비교하여 새로운 지점에서 최소의 블록에러계산 여부를 결정하고 결정된 최소의 블록에러합을 이용하여 각 단계에서 불가능하다고 판단되는 후보점들을 식 (2)를 이용하여 제거한다. 알고리즘의 동작 절차로는 우선 1단계에서 전체 탐색 후보지점에 대하여 식 (2)를 이용하여 부분 블록 매칭에러($APSAD_1$) 계산을 하고 이를 바탕으로 최소의 부분 블록매칭에러의 지점에서 식 (2)의 k값을 N까지 계속하여 블록 매칭에러 (SAD)를 계산한다. 그 후 식 (3)을 이용하여 움직임 벡터가 될 수 없는 불가능한 후보 벡터를 제거한다. 제거되지 않은 후보 중에서 다음의 부분블록매칭에러를 계산하고, 이전단계에서 부분 블록매칭에러와 그 위치를 비교한다. 이 비교에서 동일한 위치이면 식 (3)을 이용하여 현 단계에서 불가능한 후보 벡터를 제거한다. 만약 이전 단계에서의 최소 부분블록 에러합과 위치가 다르다면 현 단계에서 전체 블록매칭에러를 계산하고 이 계산된 블록매칭에러와 이전단계에서의 블록매칭에러와 비교를 해서 이전 단계에서의 블록매칭에러가 더 작다면 최소 블록에러는 그대로 유지되며, 그 반대이면 현 단계에서의 최소블록에러가 새롭게 변경되는 것이다. 그 후 새롭게 변경된 최소블록에러를 가지고 그 단계에서 불가능한 후보 벡터를 제거하게 되는 것이며 남아있는 후보점들을 유효한 후보지점으로 업데이트 후에 N 단계까지 위의 과정을 반복하게 된다. 긍극적으로는 식 (3)의 우변항의 값을 최소화 함으로써 최소에러를 가능한 빨리 계산하며 이를 통하여 불가능한 후보벡터를 조기에 제거하게 되는 것이다.

```
For loop (i=1 ~ N)
  if ( i==1 )
    Calculate partial block matching errors for all the
      search points in the search range (APSAD1(x,y))
    Find the point of minimum error  from the calculated
      partial block matching errors => (MP(1))
    Calculate block matching error at MP (1) =>SAD(1)
    Remove impossible candidates with SAD(1) and PDE
      condition
  else
    Calculate partial block matching errors for remaining
      valid candidates in the search range(APSADi(x,y))
    Find the point of minimum error  from the calculated
      partial   block matching errors => (MP(i))
    if (MP(i) == MP(i-1) )
      SAD(i) =SAD(i-1)
    else
      Calculate block matching error at MP (i) =>SAD(i)
      if ( SAD(i) >SAD(i-1) )
        SAD(i) =SAD(i-1)
        MP(i) = MP(i-1)
      end if
    end if
    Remove impossible candidates with SAD(i) and PDE
      condition
  end if
  Update valid search points
  i=i+1
End For loop
```

Fig. 1 Pseudo code of the proposed algorithm

Ⅳ. 실험 및 고찰

제안 알고리즘의 객관적인 성능 평가를 위해, “foreman”, “car phone”, “trevor”, “akiyo”, “grandmother”의

비디오 데이터를 이용하여 실험을 수행하였다. 처음 두 개의 데이터인 “foreman” 과 “car phone”은 비교적 큰 움직임을 가지고 있으며, 마지막 두 개의 데이터인 “akiyo”, “grandmother”는 이전 두 시퀀스보다 비교적 작은 움직임을 갖는다. “trevor” 시퀀스는 이들 두 부류의 중간 정도 움직임을 갖는 것으로 분류할 수 있다. 움직임 벡터를 계산하는 매칭 블록의 크기는 16×16화소이며, 탐색 영역의 범위는 ±15 화소로 설정하였다. 영상의 해상도 크기는 QCIF (176×144 화소)이며, 비디오 프레임수는 100 프레임이다. 알고리즘의 성능평가를 위한 실험 결과는 움직임 벡터 계산을 위해 사용된 평균 계산량과 움직임 벡터의 정확도와 예측화질 평가를 위하여 PSNR(Peak Signal to Noise Ratio)로 나타내었다. 각 방법에 대한 움직임 벡터를 위한 계산량은 기존의 전영역 탐색방법의 계산량을 기준으로 했을 때 이의 상대적 비율로 나타내었다.

각 방법들의 비교기준이 되는 전영역 알고리즘 (FS) [1], 고속의 무손실 움직임 예측방법으로는 H.264에서 사용되는 PDE 알고리즘 (PDE H264) [9], 복잡도 기반의 고속 PDE 방법 (PDE Complex) [8], Hadamard 변환을 이용한 고속 PDE 방법 (PDE HD) [10], MSEA (Multilevel Successive Elimination Algorithm) [7]와 최근의 손실 예측 방법인 CDMHS [6]를 가지고 실험을 수행하였다. 성능비교의 초점이 무손실 고속 알고리즘에 집중되지만 손실 고속 알고리즘 [6]과 이에 본 알고리즘과의 결합성을 위하여 결과를 함께 비교하였다.

표 1과 2는 각 움직임 예측 방법에 대하여 각 비디오 시퀀스별로 움직임 벡터의 계산을 위한 평균 연산량과 움직임 벡터를 이용한 예측화질 결과를 나타내었다. 표 1은 영상 시퀀스별 각 알고리즘의 평균 계산량을 나타내었으며, 전영역 탐색방식의 계산량을 100%로 볼 때 나머지 알고리즘의 계산량을 상대적인 %로 나타낸 것이다.

최적의 움직임 벡터를 얼마나 빨리 찾고 불필요한 계산을 어떻게 줄이느냐에 따라 알고리즘별로 계산량이 달라진다. PDE complex [8]와 PDE Hadamard [10]는 최소에러위치를 먼저 찾는 것이 아니라 불가능한 후보 벡터를 먼저 찾는 개념으로 접근한다. 이들은 블록의 부분에러합이 큰 영역을 계산해내고 이의 순서대로 블록의 부분에러합을 계산함으로써 가능한 일찍 블록의 부분에러합이 큰 영역을 계산하게 하는 것이다.

Table. 1 Computational amount for sequences (%)

	foreman	carphone	trevor	akiyo	grand
FS	100.0	100.0	100.0	100.0	100.0
PDE H.264	20.1	19.5	15.1	8.6	19.5
PDE Comp.	15.8	16.2	12.8	7.3	16.5
PDE HT	15.9	16.3	12.9	7.7	17.2
MSEA	19.9	19.4	14.8	8.4	19.3
CDMHS	13.7	13.9	12.9	11.2	13.4
CDMHS +Proposd	2.9	3.1	2.6	2.1	2.7
Proposed	11.2	11.1	9.7	6.1	10.8

Table. 2 Prediction quality for sequences(dB)

	foreman	carphone	trevor	akiyo	grand
FS ~ MSEA	32.58	34.05	34.07	44.14	43.45
CDMHS	31.59	33.13	33.27	43.83	43.23
CDMHS +Proposd	31.59	33.13	33.27	43.83	43.23
Proposed	32.58	34.05	34.07	44.14	43.45

물론 제안하는 방법은 이들 방법들을 연속적으로 함께 사용할 수가 있으며 서로에게 부정적 영향을 미치지 않기 때문에 계산량의 감소를 더 얻을 수 있을 것이다. 전영역 탐색방법의 결과와 비교해 볼 때, 다른 고속의 무손실 예측방법들은 많은 연산량 감소가 있음을 알 수 있으며, 제안 방법의 연산량 감소가 가장 많음을 알 수 있다. 또한 최근에 발표된 CDMHS [6] 방법은 사용하는 계산량이 현격히 적음을 수 있는데, 이는 표 2에서 보면 특정 비디오에 대해서는 예측 화질이 떨어짐을 알 수 있다. CDMHS와 본 알고리즘을 결합한 결과를 보면 예측화질은 동일하면서 연산량만 현저히 감소시킴을 알 수 있다.

표 2의 예측화질 결과를 보면, 전영역 탐색 방법과 비교해 볼 때 다른 고속의 무손실 예측 방법들은 전영역 탐색 방법에 비하여 예측화질의 저하가 전혀 없음을 알 수 있다. 본 논문의 제안 방법은 전 영역 탐색방법과 동일한 예측화질을 가지면서 전영영 탐색 방법에 비하여 7~12%의 연산량을 가지며, 또한 H.264에서 사용되는 PDE 방법과 비교해서 상대적으로 20 ~ 50%, PDE Complex 방법과 비교해서는 10~30%의 상대적인 계산

감축을 얻었음을 알 수 있다. 수행시간의 관점에서는, 평균적으로 본 알고리즘이 전영역 탐색 방법보다 3 ~ 6배 더 빨랐으며, 이는 구현 최적화 및 구현된 언어에 따라 다소 차이가 날 수 있다. 또한 영상의 종류에 따라 사용되는 연산량은 조금씩 차이가 있지만, 전체적으로 실험에서 사용된 모든 영상에 대해서 훨씬 적은 계산량을 가짐을 알 수 있다.

Ⅴ. 결 론

본 논문에서는 비디오 부호화 모듈에서 중요한 영향을 미치는 움직임 예측의 방법을 제안하였다. 본 논문에서 제안한 방법은 탐색 영역의 후보지점 중에서 최소 에러를 가능한 일찍 찾는데 집중하며 이를 통하여 불가능한 후보벡터를 더 일찍 제거함으로써 예측화질의 저하 없이 불필요한 계산량만 줄일 수 있는 것이다. 본 논문의 제안방법은 전영역 탐색 알고리즘과 비교하여 예측 화질의 저하가 전혀 없이 계산량은 7%~12%정도이며, 무손실 고속 움직임 예측방법들 중에서도 가장 적은 계산량을 사용함을 알 수 있다. 제안한 알고리즘은 MPEG-4 AVC 및 HEVC와도 호환이 되며, 이를 이용하는 실시간 비디오 압축 응용분야에 유용하게 사용될 수 있을 것이다.

ACKNOWLEDGMENTS

This work was supported by a Research Grant of Pukyong National University(2015)

REFERENCES

[1] J. Kim and T. Choi, “Adaptive matching scan algorithm based on gradient magnitude for fast full search in motion estimation,” *IEEE Transactions on Consumer Electronics*, vol. 45, no. 3, pp. 762-772, Aug. 1999.

[2] V. Rao, P. Muralidhar, and R. Rao “Adaptive diamond step search motion estimation algorithm for H.264/AVC,” in *Proceeding of IEEE ICACCCT*, pp.1389-1392, 2014.

[3] X. Li, R. Wang, W. Wang, Z. Wang, and S. Dong, “Fast motion estimation methods for HEVC,” in *Proceeding of IEEE ISBMSB*, pp.1-4, 2014.

[4] N. Hamid and H. Sulaiman, “Adaptive diamond orthogonal search algorithm for motion estimation,” in *Proceeding of IEEE I4CT*, pp.498-501, 2015.

[5] X. Li, R. Wang, X. Cui, and W.Wang, “Context-adaptive fast motion estimation of HEVC,” in *Proceeding of IEEE ISCAS*, pp. 2784-2787, 2015.

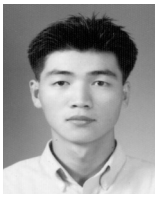
[6] N. Alnajdawi and S. Tedmori, “Employing a novel cross-diamond search in a modified hierarchical search motion estimation algorithm for video compression,” *Elsevier Information Sciences*, vol. 268, pp. 425-435, June 2014.

[7] T.G. Ahn, Y.H. Moon, and J.H. Kim, “Fast full-search motion estimation based on multilevel successive elimination algorithm,” *IEEE Transactions on Circuits and System for Video Technology*, vol. 14, no. 11, pp. 1265-1269, Oct. 2004.

[8] 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, Nov. 2002.

[9] HHI, H.264/AVC reference software [Internet]. Available: http://iphome.hhi.de/ suehring/tml/download/old_jm/

[10] S. Jin and H. Lee, “Fast partial distortion elimination algorithm based on Hadamard probability model,” *IEE Electronics Letters*, vol. 44, no. 1, pp. 17-19, Jan. 2008.



김종남(Jong Nam Kim)

1995년 2월 금오공과대학교 전자공학과 졸업(공학사)
1997년 2월 광주과학기술원 정보통신공학과 석사졸업(공학석사)
2001년 8월 광주과학기술원 기전공학과 박사졸업(공학박사)
2001년 7월 ~ 2004년 2월 한국방송(KBS) 연구원
2004년 3월 ~ 현재 부경대학교 교수
※관심분야 : 영상신호처리, 컴퓨터비젼