

다중 특징을 이용한 동작정보 측정

장석우*

Estimating Motion Information Using Multiple Features

Seok-Woo Jang*

요약

본 논문에서는 연속적인 레인지(range) 영상 자료로부터 동작 벡터를 추출하는 새로운 블록 정합 알고리즘을 제안한다. 본 논문에서 제안된 알고리즘은 단일 특징을 사용하지 않고 다중 특징인 명암값, 색상, 레인지 특징의 세 가지 특징을 통합한 정합 유사 함수를 정의하며, 엔트로피를 이용하여 각 특징의 기여도를 구한 후 이를 가중치의 형태로 정합 유사 함수에 적용한다. 그리고 제안된 알고리즘은 고정된 블록 템플릿을 사용하지 않고 가변적인 크기의 블록 템플릿을 사용한다. 제안한 블록 정합에서는 먼저 작은 정합 템플릿으로 블록 정합을 시작한다. 만일 정합 정도가 좋지 않으면 정합 템플릿의 크기를 조금 확장한 후 본 논문에서 정의한 정합기준이 만족하거나 사전에 정의된 최대 블록 크기에 도달할 때까지 블록정합을 반복한다. 실험에서는 본 논문에서 제안한 블록 정합 알고리즘과 기존의 다른 알고리즘의 성능을 비교분석하여 제안한 알고리즘의 우수함을 보인다.

Abstract

In this paper, we propose a new block matching algorithm that extracts motion vectors from consecutive range data. The proposed method defines a matching metric that integrates intensity, hue, and range. Our algorithm begins matching with a small matching template. If the matching degree is not good enough, we slightly expand the size of a matching template and then repeat the matching process until our matching criterion is satisfied or the predetermined maximum size has been reached. As the iteration proceeds, we adaptively adjust weights of the matching metric by considering the importance of each feature. In the experiments, we show that our block matching approach can work as a promising solution by comparing the proposed method with previously known method in terms of performance.

▶ Keyword : 레인지 데이터(range data), 블록 정합(block matching), 정합 척도(matching metric), 가중치 조절(weight adjustment), 동작벡터(motion vector)

• 제1저자 : 장석우
• 접수일 : 2005.03.02, 심사완료일 : 2005.05.10
* 한국건설기술연구원 건설CALIS연구센터 선임연구원

1. 서론

블록 정합은 비디오 압축, 동작 측정, 비디오 코딩, 비디오 회의, 이동 물체 감지 및 추적 등과 같은 많은 분야에서 널리 사용되어지고 있다[1]. 블록 정합에서 영상 시퀀스의 현재 프레임은 동일한 모양과 크기의 $N \times N$ 블록으로 분할된다. 그리고 현재 프레임의 분할된 각 블록에 대해 정해진 정합 척도에 따라 이전 프레임의 탐색 영역에서 가장 유사도가 높은 블록을 찾는다. 이 때 정합된 두 블록들의 상대적인 위치는 현재 프레임의 해당 블록의 동작벡터로 정의된다. 이런 동작벡터들의 집합이 동작필드(motion field)를 형성한다.

많은 기존의 블록 정합 방법은 계산 복잡도를 줄이기 위해서 대응 후보 블록을 찾기 위해 사용되는 탐색 영역을 정의하는데 초점을 맞춰 왔다. 예를 들면, 전역 탐색 블록 정합(FSBM: full search block matching) [2], 3단계 탐색에 의한 방법(TSS: three-step search) [3], 4단계 탐색에 의한 방법(FSS: four-step search) [4], 계층적인 1차원 탐색에 의한 방법[5] 등이 있다. 전역 탐색에 의한 방법은 탐색 영역내의 모든 위치에 대해 정합을 시도하여 최선의 정합 블록을 찾는다. 3단계 탐색에 의한 방법은 균일하게 할당된 탐색 형태를 조사한다. 4단계 탐색에 의한 방법은 중심 기반의 체크 포인트 패턴을 사용하여 빠른 블록 정합을 수행한다.

한편, 탐색 영역을 정의하는 방법과 더불어 정합 블록의 크기를 적당하게 선택하는 것도 블록정합의 정확도를 결정하는데 큰 역할을 수행한다. 일반적으로, 큰 크기의 블록들은 조밀하지는 않지만 강건적인 동작의 예측에 적당하고, 작은 크기의 블록들은 지역적인 예측에 적당하지만 잡음에 민감하다. 특히, 이동 물체 감지 및 추적과 같은 분야에서는 추적 작업이 영상 전체를 대상으로 하지 않고 영상 내의 예측되는 작은 탐색 영역내에서 수행된다. 따라서 이런 분야에서는 동작벡터를 빠르게 추출하기 보다는 정확하게 추출하는 것이 더 중요하다.

또한, 대부분의 블록 정합 알고리즘은 정합 정도를 계산할 때 대응하는 블록들의 명암값의 차이만을 고려한다. 이런 블록 정합 기법들은 연속적인 프레임 사이의 시·공간적

인 반복을 줄이는 것이 주된 관심사가 되는 비디오 코딩과 같은 분야에는 적합하다. 그러나 동작을 기반으로 영상의 질을 향상할 필요가 있는 보간(interpolation)이 필요한 화상회의와 같은 분야에서는 보다 정확한 블록 단위의 동작벡터가 필요하다. 예를 들어, 인접한 블록들은 이들의 내부 구조는 다르지만 유사한 명암값을 보일 수 있다. 이런 경우 명암값에 기반한 정합 척도는 대응블록을 잘못 선택할 수 있으므로 부정확한 동작벡터를 추출하게 된다.

본 논문에서는 이런 문제를 해결하기 위해 연속적인 레인지 데이터로부터 동작벡터를 추출하는 새로운 블록 정합 알고리즘을 제안한다. 레인지 데이터에 대한 설명은 다음 장에서 자세히 설명할 것이다. 본 논문에서 제안된 알고리즘은 단일 특징이 아니라 명암값, 색상, 레인지의 세 가지 다중 특징을 통합한 정합척도를 정의한다. 그리고 전체적인 정합정도는 개별적인 특징들의 정합 계산값의 가중화된 합으로 정의된다. 따라서 제안된 방법은 다중 특징을 효과적으로 사용하므로 기존의 방법보다 블록정합의 정확도를 높임으로써 보다 정확하게 동작벡터를 추출할 수 있다.

본 논문에서 제안된 알고리즘은 작은 정합 템플릿으로 블록 정합을 시작한다. 만일 정합 정도가 충분히 좋지 않으면 정합 템플릿의 크기를 조금 확장한 후 본 논문에서 정의한 정합기준이 만족하거나 사전에 정의된 최대 블록크기에 도달할 때까지 블록정합을 반복한다. 본 논문에서는 정합 과정이 진행됨에 따라 각 특징의 중요도를 고려하여 정합척도의 가중치를 적응적으로 조절한다. 정합척도의 가중치 조절은 탐색영역 내에서 각 특징의 엔트로피(entropy)를 사용하여 수행함으로써 대응블록 안의 특징들의 구분력에 따라 특징을 분별할 수 있다.

본 논문의 전체 구성은 다음과 같다. 1장에서는 연구 동기와 제안된 블록 정합의 개요를 설명하였다. 2장에서는 레인지 데이터에 대해 설명하고, 3장에서는 정합 척도로 사용하는 정합유사함수를 정의하는 방법 및 정합척도를 판단하는 평가함수를 기반으로 블록의 크기의 적합성을 결정하는 방법에 대해 설명하며, 4장에서는 엔트로피를 사용하여 각 특징의 가중치를 조절하는 방법에 대해 설명한다. 그리고 5장에서는 제안된 방법의 성능을 비교하는 실험 결과에 대해 설명하며, 6장에서는 결론을 기술한다.

II. 레인지 데이터

레인지 데이터(range data)는 연속적인 동작 분석에서 향상된 품질(quality)과 확장 가능한 자료 유용성(data usability) 때문에 동작 측정 및 이해, 3차원 제스처 인식, 물체의 이동궤적 추적 등의 분야에서 매우 유용하다(6). 보통, 레인지 영상에서 촬영된 공간 상의 지점까지의 거리는 구분적인 범위로 기록된다. 보통 레인지 데이터의 값의 범위는 0에서 255 사이의 값으로 양자화(quantization)된다.

예를 들어, 레인지 값 0는 카메라로부터 촬영되는 지점까지 2.5cm 떨어진 거리, 레인지 값 1은 2.5cm에서 5.0cm 사이의 거리, ... , 레인지 값 255는 637.5cm에서 640cm 사이의 거리를 의미할 수 있다. 다시 말해, 레인지 카메라로 촬영된 영상은 네 개의 입력자료, 즉 (r, g, b, d)로 표현될 수 있는데, 여기서 r, g, b는 칼라 정보를 의미하고, d는 카메라로부터 촬영되는 지점 사이의 거리를 의미한다.

일반적으로, 디스플레이를 목적으로 할 때 d는 명암값으로 종종 표현되는데, 픽셀이 점정색에 근접할수록 카메라와 촬영되는 지점 사이의 거리는 가깝다. 레인지 영상은 또한 시각적으로 구별하기 용이하도록 임의의 칼라를 이용하여

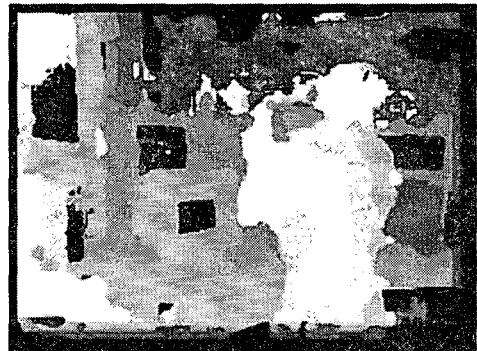
디스플레이 될 수 있다. 레인지 카메라로 획득된 데이터는 종종 3차원이 아니라 2.5 차원이라고 언급되기도 한다. 왜냐하면, 이런 유형의 장치를 마그네틱 공명 영상(magnetic resonance image), 즉 완전한 3차원 그리드(grid)로 촬영하는 다른 장치들과 구별하기 위해서인데, 레인지 영상은 단지 보이는 물체의 표면 깊이(depth) 정보만을 제공하기 때문이다(7). (그림 1)은 레인지 데이터의 한 예를 보여주고 있다. (그림 1)의 (a)는 레인지 카메라로 촬영한 칼라 영상을 보여주며, (그림 1)의 (b)는 입력받은 레인지 값을 명암값으로 디스플레이한 화면을 보여준다.

실세계에서 촬영된 영상 시퀀스는 카메라로부터의 서로 다른 깊이 정보를 가진 많은 영역들을 포함한다. 일반적인 카메라를 사용하여 촬영한 2차원 이미지에서 명암값이나 칼라 영상이 획득되지만 촬영된 지점까지의 깊이 정보는 포함되지 않는다. 한편, 레인지 영상은 촬영되는 지점까지의 거리 정보를 포함하기 때문에 동작 측정에 보다 유용하게 사용될 수 있다.

본 논문에서는 기존에 많이 사용되는 블록 정합 알고리즘의 정확도를 향상시키기 위해서 레인지 데이터를 사용한다. 본 논문에서 제안된 블록 정합 알고리즘은 세 가지의 다중 특징을 통합하여 사용하는 정합 척도를 정의할 때 레인지 특징을 이 중 하나로 포함한다. 그리고 명암값이나 색상 정보가 빈약한 영역에 대한 블록 정합 시에는 레인지 특징에 대한 가중치를 높게 조절하여 레인지 정보의 도움으로 블록 정합의 정확도를 향상하도록 시도한다. 레인지 특징에 대한 보다 자세한 사항은 다음 장에서 기술할 것이다.



(a) 입력 영상
(a) Input image



(b) 레인지 값
(b) Range value

그림 1. 레인지 데이터
Fig. 1 Range data

III. 정합 척도 정의

$N \times N$ 크기의 블록이 주어질 경우, 블록 단위의 동작 측정은 탐색 영역 안에서 정합 유사도가 가장 높은 블록을 찾는다. 블록 사이의 정합 척도로는 여러 가지 기법을 고려할 수 있다[8,9]. 본 논문에서는 기존의 블록 정합 방법들이 대부분 명암값이라는 하나의 특징만을 사용하여 블록 정합을 수행하므로 영상 안에 포함된 다른 유익한 특징들을 고려하지 않는다고 판단한다. 따라서 불충분한 특징을 사용하여 정합을 수행하므로 정합의 정확도가 저하되고 부정확한 정합으로 인해 잡음이 포함된 동작벡터를 추출할 확률이 높다.

본 논문에서는 이런 문제점들을 해결하기 위해서 영상 내의 여러 가지 특징을 동시에 고려하여 블록 정합을 수행한다. 본 논문에서 제안하는 블록 정합 알고리즘은 명암값 특징, 색상 특징, 레인지 특징의 세 가지 특징을 고려하는 알고리즘으로 블록 정합 시 각 특징의 기여도를 나타내는 가중치(weighting factor)를 자동으로 조절할 수 있도록 정합 유사 함수를 정의함으로써 영상 내의 특징을 보다 정확히 고려할 수 있다.

다중 특징을 사용하여 정합 척도를 정의할 때는 다음과 같은 두 가지 사항을 고려해야 한다. 첫 번째 이슈는 특징들의 크기를 정규화(normalize)하는 것이다. 만일 하나의 특징이 다른 특징들에 비해 넓은 범위의 값들을 가진다면 이 특징은 전체 유사도 합에 보다 큰 영향을 미칠 것이고

블록 정합의 최종적인 결정은 이 특징을 위주로 수행될 것이다. 이 문제를 해결하기 위해서는 정합 유사도를 계산하기 전에 각 특징에 스케일 기준(scale factor)을 적용하는 것이 필요하다. 두 번째 이슈는 각 특징의 중요도에 따라 가중치를 적용하는 방법이다. 정합하는 블록의 내부 컨텍스트에 따라

어떤 특징은 다른 특징들에 비해 보다 중요할 것이다. 만일 특징들의 상대적인 중요도에 대한 사전 지식을 사용할 수 있다면 각 특징은 그들의 중요도에 따라 가중화될 수 있을 것이다.

본 논문에서는 식 (1)과 같이 두 블록 사이의 정합 유사 함수 DBS (displaced block similarity)를 정의한다. 식 (1)에서 F_{inten} , F_{hue} , F_{range} 는 명암값, 색상, 레인지 특징값을 각각 나타낸다. n 은 블록 템플릿의 크기를 의미하고, (i, j) 는 영상의 (i, j) 에 위치하는 블록의 시작 좌표를 의미하며, (u, v) 는 기준 블록과 정합되는 블록 사이의 상대적인 변위를 나타낸다. I_t 와 I_{t-1} 은 현재 시점과 이전 시점 영상의 명암값을 의미하고, H_t 와 H_{t-1} 은 색상값을 의미하며, R_t 와 R_{t-1} 은 레인지 값을 의미한다. I_{max} , H_{max} , R_{max} 는 세 가지 특징값의 최대값을 의미한다. $\alpha(n)$, $\beta(n)$, $\gamma(n)$ 는 블록 템플릿의 크기가 n 일 때의 세 가지 특징에 대한 가중치를 나타낸다. 각 특징에 대한 가중치 조절에 대한 사항은 다음 장에서 자세히 기술할 것이다. 정합 유사 함수 DBS 는 각 특징들의 유사도의 가중화된 합으로 정의되며, 0에서 100 사이의 범위의 값을 가진다. $DBS(i; j; u, v; n)$ 를 최대화시키는 후보 블록이 최상의 정합 블록으로 판단되며, 두 블록 사이의 상대적인 변위 (u, v) 는 블록 $(i; j; n)$ 의 동작벡터로 정의된다.

$$DBS(i, j; u, v; n) = \alpha(n) \cdot F_{inten}(i, j; u, v; n) + \beta(n) \cdot F_{hue}(i, j; u, v; n) + \gamma(n) \cdot F_{range}(i, j; u, v; n) \dots\dots\dots (1)$$

$$F_{inten}(i, j; u, v; n) = \left(1 - \frac{1}{n^2} \sum_{y=0}^{n-1} \sum_{x=0}^{n-1} \left| \frac{I_t(i+x, j+y) - I_{t-1}(i+u+x, j+v+y)}{I_{max}} \right| \right) \times 100$$

$$F_{hue}(i, j; u, v; n) = \left(1 - \frac{1}{n^2} \sum_{y=0}^{n-1} \sum_{x=0}^{n-1} \left| \frac{H_t(i+x, j+y) - H_{t-1}(i+u+x, j+v+y)}{H_{max}} \right| \right) \times 100$$

$$F_{range}(i, j; u, v; n) = \left(1 - \frac{1}{n^2} \sum_{y=0}^{n-1} \sum_{x=0}^{n-1} \left| \frac{R_t(i+x, j+y) - R_{t-1}(i+u+x, j+v+y)}{R_{max}} \right| \right) \times 100$$

본 논문에서 제안하는 블록 정합 알고리즘은 구분력 있는 최대 정합 유사도를 추출할 때까지 블록의 크기를 확장 하면서 블록 사이의 정합을 반복적으로 수행한다. 이를 위해, 본 논문에서는 블록 템플릿의 크기의 적합성을 결정하기 위해 후보 블록들의 정합 정도를 조사하는 평가 함수 (evaluation function)를 새롭게 정의한다. 평가 함수는 다음과 같은 두 가지 평가 기준을 설정하여 사용한다.

첫째, 블록 크기의 확대 여부에 대한 기준을 설정한다. 즉, 정해진 탐색 영역에서 추출한 정합 유사도 분포 중 최대 정합 유사도를 가지는 블록이 주위의 후보 블록들의 정합 유사도보다 상대적으로 높다면 이 정합은 구분력(distinctiveness)이 있으며, 해당 블록의 크기는 적당하다고 판단한다. 다시 말해, 최대 정합 유사도 DBS_{max} 를 가지는 위치를 중심으로 인접 블록과의 DBS 의 차이를 비교하여 평가하는데, 본 논문에서는 DBS_{max} 를 가지는 블록과 인접 블록에서의 DBS 의 차이 중 최소값을 경사도로 정의하고, 경사도가 임계치 TH 이하일 경우에만 블록을 확대한다. 그러나, 최대 정합 유사도를 가지는 블록과 주위의 후보 블록들의 정합 유사도가 비슷하다면 이 정합은 특징값이 유사한 영역 안에서 블록 정합이 수행되어 정합 구분력이 낮고 블록 템플릿의 크기가 부적당하다고 판단한다. 따라서 정합 구분력을 높이기 위해 블록 템플릿의 크기를 조금 확대한 후 블록 정합을 반복한다.

둘째, 블록의 크기를 확대하면서 정합 유사도를 반복적으로 계산할 경우 블록 크기의 확대 지속 여부에 대한 기준을 설정한다. 본 논문에서는 블록의 크기를 확대하더라도 최대 정합 유사도의 구분력이 더 이상 향상되지 않을 때 블록 정합을 중지한다. 다시 말해, 블록의 크기를 확대함에 따라 반복적으로 구하는 정합 구분력의 차이를 구분력의 경사도

(gradient)로 정의하고, 이 경사도의 1차 미분값이 양수 (+)에서 음수(-)로 변하는 변곡점을 발견할 때까지 블록 정합을 반복한다. 위에서 언급한 두 가지 평가 기준을 바탕으로 본 논문에서 사용하는 평가 함수 $\Phi(i,j;n)$ 를 수학적으로 정형화하면 식 (2)와 같다.

식 (2)에서 $(i^*, j^*; n)$ 은 크기가 $n \times n$ 이고 (i, j) 에 위치한 블록이 최대 정합 유사도를 가질 때 이 블록의 중심 위치를 의미한다. $DT(i,j;n)$ 은 최대 정합 유사도를 가지는 블록의 구분력을 의미하는데, 이는 최대 정합 유사도를 가지는 블록과 인접 블록 사이의 정합 정도의 차이 중 최소값을 의미한다. (l, m) 은 최대 정합 유사도를 가지는 블록의 인접 블록의 위치를 나타내는 인덱스이다. $GD(i,j;n)$ 은 블록 템플릿의 크기에 대한 경사도를 의미하는데, 이는 $n \times n$ 인 블록 템플릿의 크기에서 계산한 구분력과 $(n-1) \times (n-1)$ 인 블록 템플릿의 크기에서 계산한 구분력의 차이로 정의된다. $TH(i,j;n)$ 은 구분력 $DT(i,j;n)$ 과 임계치 T_{pk} 와의 차이를 나타낸다. e_1 과 e_2 는 블록 정합 반복 시 평가 함수의 수렴 속도를 조절하는 상수이다.

본 논문에서 평가함수 $\Phi(i,j;n)$ 는 구분력 $DT(i,j;n)$ 가 임계치 T_{pk} 보다 크지 않고 구분력의 경사도 $GD(i,j;n)$ 가 양수일 경우에만 양의 값을 가지도록 설계되었다. 본 논문에서는 식 (2)에서 정의한 평가 함수를 이용하여 제안된 블록 정합을 수행하며, 평가 함수가 양의 값을 갖는 동안에만 블록 정합을 반복하면서 동작벡터를 구한다. 본 논문에서는 정합 유사도에 적용하는 임계치와 경사도를 결합하여 평가 함수를 정의함으로써 간단하게 블록 크기의 확대 여부 및 확대 지속 여부를 결정하면서 블록 정합을 반복하여 보다 정확하게 동작정보를 추출할 수 있다.

$$\Phi(i, j; n) = \max \left[\begin{matrix} e_1 + TH(i, j; n) \\ e_2 + GD(i, j; n) \end{matrix} \right] \times \frac{e_1 + TH(i, j; n)}{e_2 + GD(i, j; n)} \dots\dots\dots (2)$$

$$TH(i, j; n) = T_{PK} - DT(i, j; n)$$

$$GD(i, j; n) = DT(i, j; n) - DT(i, j; n-1)$$

$$DT(i, j; n) = \min_{-|s|, |ms|} [DBS(i^*, j^*; n) - DBS(i+l, j+m; n)]$$

IV. 가중치 설정

본 장에서는 세 가지 특징을 통합하여 정의되는 정합 유사 함수의 각 특징에 대한 가중치를 설정하는 방법에 대해 설명한다. 단일의 특징값이 아니라 여러 가지의 특징값을 통합하여 정합 유사 함수를 정의함으로써 정합의 구분력을 높일 수 있는 것은 좋은 방법이라 판단된다. 이 때 중요한 것은 서로 다른 특징값의 범위를 통일시키는 정규화(normalization) 과정과 각 특징값의 기여도를 나타내는 가중치를 정의 및 조절하는 방법일 것이다.

본 논문에서는 각 특징의 가중치를 보다 합리적으로 결정하기 위해서 탐색 영역 내의 해당 특징의 엔트로피(entropy)를 계산하여 사용한다. 엔트로피는 통계학(statistics)이나 코딩 이론(coding theory)에서 널리 사용되는 개념으로 확률 분포의 불확실성(uncertainty)을 얻기 위해 사용되며, 최적의 특징 선택을 하기 위한 분야에서 많이 사용된다 [10]. 즉, 균일한 분포를 가진 모집단일 경우에는 낮은 값의 엔트로피를 가지며, 비균일한 분포를 이루는 모집단일 경우에는 높은 값의 엔트로피를 가진다. 일반적으로, 모집단에 대한 엔트로피는 식 (3)과 같이 정의된다.

$$H = -\sum_{i=1}^n P_i \log P_i \dots\dots\dots (3)$$

식 (3)에서 P_i 는 i 번째 샘플이 발생할 확률을 의미한다. 이와 같은 엔트로피 개념을 이용하여 블록 정합을 수행할 탐색 영역에서 특징값들의 비균일도를 구한다. 본 논문에서는 식 (4)와 같이 특징의 가중치를 결정한다. 주어진 탐색 영역에서 불확실성을 증가시키는 특징은 반대의 효과를 발생시키는 특징보다 유용하다고 판단되는데, 이는 높은 불확실성을 가지는 특징은 탐색 영역에서 후보 블록이 다른 블록들보다 서로 다른 특징을 가질 확률이 높기 때문이다. 이 이론은 클래스 사이의 높은 비유사도를 보여주는 특징을 강조하는 것과 동일하다.

따라서 본 논문에서는 높은 값의 불확실성을 갖는 특징에 보다 높은 가중치를 할당하고, 낮은 값의 불확실성을 갖

는 특징에 보다 낮은 가중치를 할당한다. 본 논문에서는 이런 방법으로 식 (1)의 가중치를 정의한다. 가중치를 할당하는 작업이 현재 프레임의 각 블록에 대해 수행되기 때문에 해당되는 DBS가 보다 정확한 동작 벡터를 산출한다는 것을 기대할 수 있다. 본 논문에서는 각 특징의 엔트로피를 식 (4)와 같이 정규화된 엔트로피로 정의한다. 따라서 각 가중치는 0에서 1사이의 범위의 값을 가지도록 정규화되며, 이들 각각의 합은 1이 된다.

$$\alpha(n) = \frac{E_{inten}}{E_{inten} + E_{hue} + E_{range}}$$

$$\beta(n) = \frac{E_{hue}}{E_{inten} + E_{hue} + E_{range}} \dots\dots\dots (4)$$

$$\gamma(n) = \frac{E_{range}}{E_{inten} + E_{hue} + E_{range}}$$

식 (4)에서 E_{inten} , E_{hue} , E_{range} 는 명암값, 색상, 레인지 특징에 대한 엔트로피를 의미하며, 식 (5)와 같이 정의된다. 여기서, $P(\cdot)$ 는 주어진 탐색 영역에서 평가가 되는 특징값의 확률 밀도를 나타낸다. W 는 현재 블록이 조사될 탐색 영역을 나타낸다.

$$E_{inten} = -\sum_{i,j \in W} P(I(i,j)) \log P(I(i,j)) \dots (5)$$

$$E_{hue} = -\sum_{i,j \in W} P(H(i,j)) \log P(H(i,j))$$

$$E_{range} = -\sum_{i,j \in W} P(R(i,j)) \log P(R(i,j))$$

본 논문에서 정의한 정합 유사 함수는 블록 정합 시 사용되며 각 블록에 대한 정합을 시도할 탐색 영역 내에 존재하는 여러 가지 특징값들의 비균일도를 엔트로피로 정의한다. 그리고 이 엔트로피의 값에 따라 가중치를 설정하고, 정합 과정이 반복됨에 따라 적응적으로 가중치를 조절하므로 보다 정확한 정합 결과를 얻을 수 있다.

V. 실험 결과

본 장에서는 제안한 블록 정합 알고리즘에 대한 성능을 평가한다. 본 논문에서 사용한 레인지 입력 영상의 크기는 320×240이다. (그림 2)는 실험에서 사용한 테스트 영상 시퀀스 중 두 개의 인접 프레임을 보여주고 있다. 이 테스트 시퀀스에서 카메라는 고정되어 있으며, 영상 안의 사람은 주먹을 왼 손을 위쪽 방향으로 이동하고 있다.

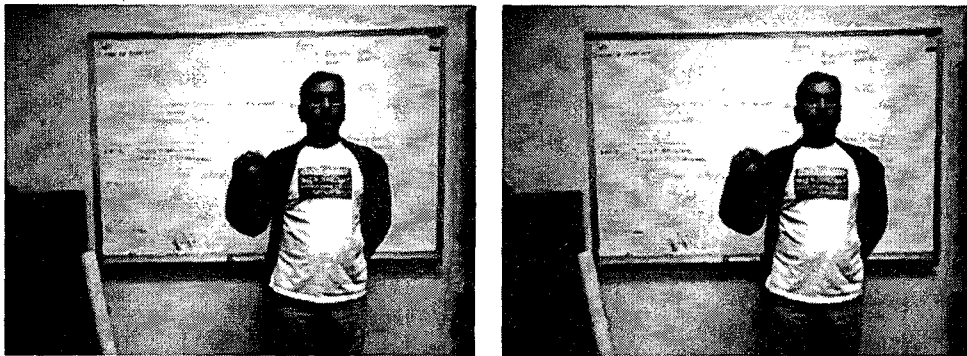
본 논문에서 제안한 블록 정합 방법과 기존의 블록 정합 방법들의 성능을 비교 및 분석하기 위해서 본 논문에서는 전역 탐색에 의한 방법(2), 3단계 탐색에 의한 방법(3), 4 단계 탐색에 의한 방법(4)을 구현하였다. (그림 3)은 이 네 가지 블록 정합 방법으로 (그림 2)의 (a)와 (b)의 입력 영상을 받아들여 추출한 각 동작 벡터를 보여준다.

(그림 3)에서 확인할 수 있듯이 모든 블록 정합 방법은 명암값의 차이가 분명한 영역에서 어느 정도 정확히 동작 벡터를 추출하는 것을 확인할 수 있다. 그러나 기존의 블록 정합 방법은 인접된 블록들의 명암값이 균일하게 분포하는 영역이나 에지나 라인 등이 존재하는 부분에서는 부정확한 블록 정합으로 인해 동작벡터를 잘못 추출하였다.

본 논문에서 제안한 방법은 이와 같은 영역에서 기존의 다른 방법들보다 좋은 결과를 보여주는 것을 확인할 수 있다. 다시 말해, 제안된 정합 알고리즘은 특징값이 유사한 영역에서 명암값 뿐만 아니라 색상 및 레인지 특징을 통합하여 활용하였으며, 각 특징에 대한 중요도를 고려하여 이를 정합 유사 함수의 가중치로 적용하였다. 또한, 고정된 블록 템플릿을 사용하는 대신 블록 템플릿의 크기를 확장하면서 최선의 정합 유사도를 추출했기 때문에 이런 영역에서도 다른 방법들보다 정확하게 동작벡터를 추출할 수 있었다. 본 논문에서 사용한 블록 정합 템플릿의 크기의 범위는 7×7에서 19×19까지이다.

(그림 4)는 제안한 블록 정합 방법에서 영상 전체의 블록들에 적용했던 블록 템플릿의 크기를 보여준다. (그림 4)의 x 축은 영상에서 블록의 수평 위치를 나타내는 인덱스를 의미하고, y 축은 블록의 수직 위치를 나타내는 인덱스를 나타낸다. (그림 4)에서 0은 $(N+0) \times (N+0)$ 크기의 블록 템플릿, 1은 $(N+1) \times (N+1)$ 크기의 블록 템플릿을 나타내며, 나머지도 이런 방식의 크기를 나타낸다. 그림에서 확인할 수 있듯이 인접한 블록의 명암값이 유사한 영역에서 블록 템플릿의 크기가 확장되었음을 확인할 수 있다.

(그림 5), (그림 6), (그림 7)은 각 블록 정합 시 사용되었던 가중치 α , β , γ 를 나타낸다. 이들은 각 특징의 구분력(distinctiveness)에 따라 서로 다른 값을 나타낸다. 가중치를 표현한 그림을 통해 어떤 특징이 보다 유용한지(informative)를 확인할 수 있다.



(a) 입력 영상 t
(a) Input image t

(b) 입력 영상 t+Δt
(b) Input image t+Δt

그림 2. 레인지 입력 영상
Fig. 2 Range input images

VI. 결론

본 논문에서는 블록 단위의 동작벡터를 보다 정확하게 추출하기 위해서 단일 특징으로 많이 사용하는 명암값 특징만을 사용하지 않고, 색상 특징과 레인지 특징도 통합하여 사용하였다. 레인지 특징은 레인지 정보를 출력해 주는 레인지 카메라를 이용하여 획득하였다. 또한, 고정적인 블록 템플릿을 사용하지 않고 해당되는 탐색 영역 내의 특징값들의 구분력을 구하여 블록 템플릿의 크기를 적응적으로 조절하여 블록 정합의 정확도를 높였다.

블록 정합 시 다중의 특징인 명암값, 색상, 레인지 특징

을 사용하기 위해 이들을 통합할 수 있는 정합 척도를 정의하였다. 그리고 정합 척도에 각 특징의 기여도를 효과적으로 반영시키기 위해 엔트로피 개념을 이용하여 각 특징의 비균일도를 조사하여 이를 정합 척도의 가중치로 활용하였다.

제안된 알고리즘에서 블록 템플릿을 적응적으로 확장하여 블록 정합의 정확도를 향상하기 위해 평가 함수를 새롭게 정의하여 사용하였다. 본 논문에서는 평가 함수를 이용하여 먼저 작은 정합 템플릿으로 블록 정합을 시작한다. 만일 정합 정도가 충분히 좋지 않으면 정합 템플릿의 크기를 조금 확장한 후 본 논문에서 정의한 정합기준이 만족하거나 사전에 정의된 최대 블록 크기에 도달할 때까지 블록정합을 반복한다.

향후에는 보다 다양한 영상을 통해 실험할 예정이며, 2차원의 동작 벡터를 3차원의 동작 벡터 추출로 확장할 예정이다.

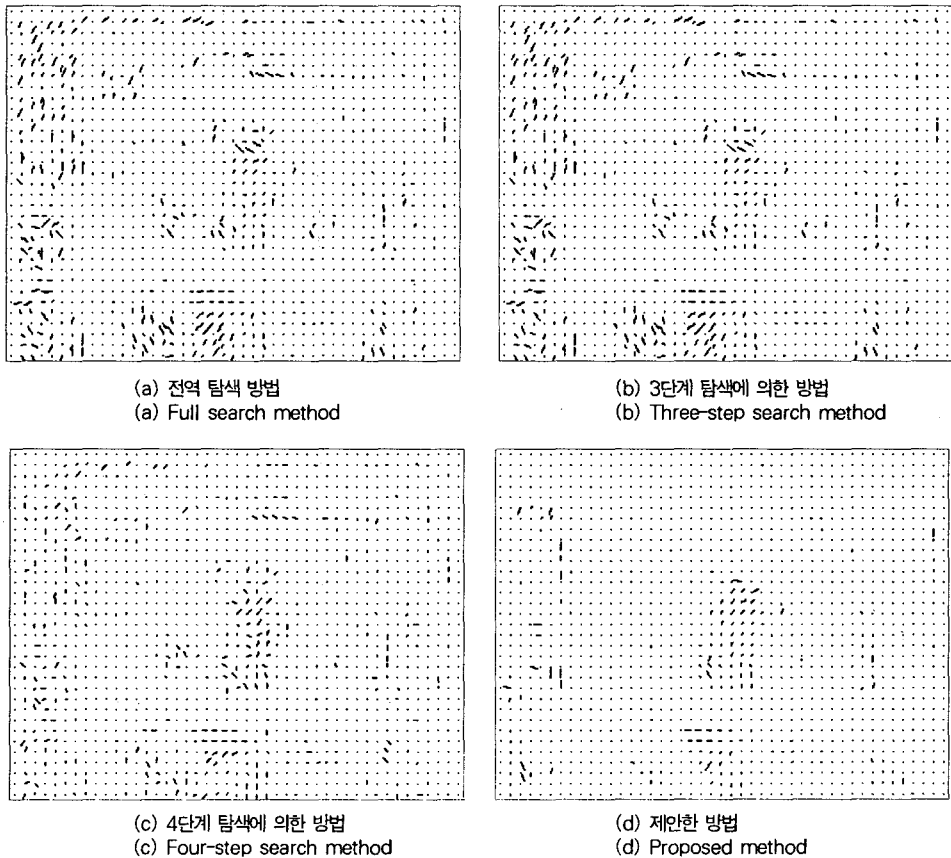


그림 3. 추출된 동작 벡터
Fig. 3 Estimated motion vectors

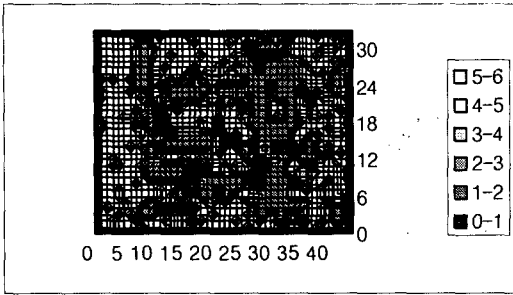


그림 4. 블록 템플릿의 크기
Fig. 4 Sizes of block templates

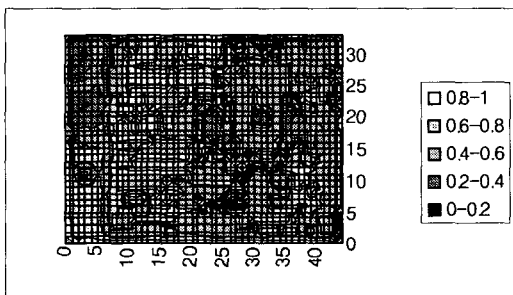


그림 5. 명암값 특징(a)
Fig. 5 Intensity feature(a)

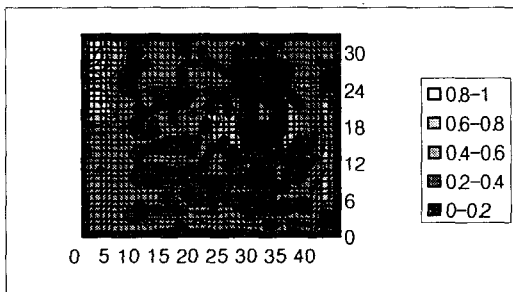


그림 6. 색상 특징(b)
Fig. 6 Hue feature(b)

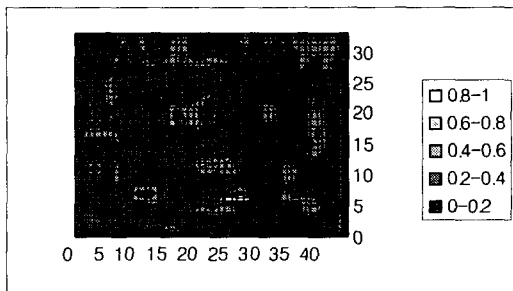


그림 7. 레인지 특징(v)
Fig. 7 Range feature(v)

참고문헌

- [1] 이희만, 서정만, "組細 단계를 이용한 光流검출 알고리즘," 한국컴퓨터정보학회논문지, Vol. 10, No. 1, pp. 223-229, 2005.
- [2] B. Liu and A. Zaccarin, "New Fast Algorithms for the Estimation of Block Motion Vectors," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 3, No. 2, pp. 438-441, 1994.
- [3] R. Li, B. Zeng, and M. L. Liou, "A New Three-Step Search Algorithm for Block Motion Estimation," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 4, No. 4, pp. 438-441, 1994.
- [4] Lai-Man Po and Wing-Chung Ma, "A Novel Four-Step Algorithm for Fast Block Motion Estimation," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 6, No. 3, pp. 313-317, 1996.
- [5] Seok-Woo Jang, Kyu-Jung Kim, and Hyung-II Choi, "Accurate Estimation of Motion Vectors Using Active Block Matching," Proceedings of the International Conference on Rough Sets and Current Trends in Computing, Banff, Canada, pp. 527-531, 2000.
- [6] Leonid V. Tsap and Min C. Shin, "Improving Quality and Speed of Range Data for Communication," IEEE International Workshop on Cues in Communication, pp. 348-353, 2001.
- [7] Min C. Shin, Leonid V. Tsap, and D. B. Goldgof, "Gesture Recognition Using Bezier Curves for Visualization Navigation from Registered 3-D Data," Pattern Recognition, Vol. 37, Issue 5, pp. 1011-1024, 2004.
- [8] Gerard Medioni, *Emerging Topics in Computer Vision*, Prentice Hall, pp. 122-145, 2004.

- [9] 임인선, "인터넷상의 동영상에서의 물체 특징 점 탐지 및 추적." 한국컴퓨터정보학회논문지, Vol. 10, No. 1, pp. 149-156, 2005.
- [10] Prasanna K. Sahoo and Gurdial Arora, "A Thresholding Method Based on Two-Dimensional Renyi's Entropy." Pattern Recognition, Vol. 37, Issue 6, pp. 1149-1161, 2004.

저자 소개



장 석 우

2000년 8월 숭실대학교 대학원
컴퓨터학과 졸업 (공학박사)
2003년 10월 - 현재 한국전설기술
연구원 선임연구원
〈관심분야〉 컴퓨터 비전, 영상 처리,
동작 이해, 비디오 색인 및
검색