

적분영상을 사용한 특징점 필터링

배병조*, 박종승**
 인천대학교 컴퓨터공학부
 e-mail:wings810@incheon.ac.kr

Feature Point Filtering Using Integral Image

Byeong-Jo Bae*, Jong-Seung Park**
 Dept. of Computer Science and Engineering, University of Incheon

요 약

두 영상에서 특징점을 추적하기 위하여 특징점을 중심으로 $N \times N$ 크기 윈도우 패치의 SSD 값을 비교하는 방법을 사용한다. 그러나 영상에서 다수의 특징점이 추출되어 매칭을 시도하는 경우 많은 처리 시간을 필요로 한다. 처리 시간의 증가는 실시간 처리를 어렵게 만든다. 이러한 문제를 해결하기 위하여 적분 영상(integral image)을 사용하여 매칭 가능성이 높은 특징점을 필터링하여 SSD 매칭의 처리 시간을 단축시키는 방법을 제안한다. 본 논문에서 제안한 적분 영상을 사용한 특징점 필터링을 적용한 방법과 기존의 SSD 매칭 방법을 비교 실험하여 특징점 매칭의 처리 시간을 절감하는 결과를 얻을 수 있었다.

1. 서론

영상에서의 특징점을 비교하여 매칭하는 과정에서 특징점은 크게 두 가지로 구분될 수 있다. 먼저 서술자를 사용하는 크기 및 회전 변화에 강건한 SIFT[1], SURF[2] 등의 방식과 특징점 서술자를 사용한 방법 보다는 강건하지 않은 영상에서의 코너(corner)점을 사용하는 FAST 코너 검출 방식[3]이 있다.

특징점 서술자를 사용하는 특징점 매칭 방식은 카메라 시점의 변화와 회전의 정도가 비교적 큰 영상에서 대상이 되는 특징을 찾는 것에는 유리하나 특징점 추출 및 특징점 매칭을 위한 계산량이 많아 처리시간이 늘어가는 단점을 가지고 있다.

영상에서의 코너(corner)점을 사용하는 특징점 매칭 방식은 영상의 움직임이 크지 않은 두 영상에서 특징점이 위치하는 이웃 픽셀들과의 조합을 통한 비교에서 오차가 가장 작은 특징점을 찾아내는 방법이 사용된다. 이때 특징점을 중심으로 $N \times N$ 크기의 윈도우 패치를 비교하여 특징점을 매칭하는 SSD(Sum of Squared Differences) 매칭 방법을 사용한다.

[4]에서는 증강현실의 구현을 위하여 Map을 사용하는 데, Map을 구성하는 Keyframe에서 추출된 특징점의 개수는 Map의 크기가 커질수록 증가한다. 그리고 등록된 Keyframe의 특징점과 입력되는 현재영상에서의 추출된 특징점을 지속적으로 매칭하게 된다. 이 경우 Map을 구성

하는 특징점이 많이 존재하기 때문에 현재영상에서 추출된 특징점과의 SSD 매칭을 시도하는 경우에 처리 시간이 증가된다. 또한 특징점 개수와 더불어 특징점의 윈도우 패치의 크기가 증가하면 추가적인 처리 시간의 증가는 불가피하다[5]. 이러한 처리 시간의 증가를 막고자 적분 영상을 사용한 필터링 기법을 제안한다.

본 논문은 2절 적분 영상에 대한 설명, 3절 적분 영상 필터링을 사용한 특징점 매칭, 4절 실험 결과에 대하여 설명, 5절 결론을 맺는 순으로 구성된다.

2. 적분영상

적분영상은 원본 영상의 픽셀 밝기 값의 합으로 나타낸 이미지를 말한다. 식 (1)을 통하여 적분영상에서의 픽셀 값을 계산할 수 있다.

$$I_{\Sigma}(X) = \sum_{i=0}^x \sum_{j=0}^y I(i, j) \quad (1)$$

1	2	3	4	5	6	1	3	6	10	15	21
7	8	9	10	11	12	8	18	30	44	60	78
13	14	15	16	17	18	21	45	72	102	135	171
19	20	21	22	23	24	40	84	132	184	240	300
25	26	27	28	29	30	65	135	210	290	375	465
31	32	33	34	35	36	96	198	306	420	540	666

그림 1. (좌)일반 영상, (우)적분 영상 예

* 이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2010-0010950)

(그림 1)은 6×6 크기의 영상을 예를 들어 나타낸 것이다. 좌측 그림은 일반 영상이 가지는 픽셀 밝기 값을 나타내고 있으며 우측 그림은 이를 사용하여 생성된 적분 영상의 예를 나타내고 있다. 본 논문에서 사용되는 적분 영상 필터링을 위하여 픽셀 값의 제곱 값을 사용한 적분 영상을 생성한다.

	0	1	2	3	4	5
0	1 ^A	3	6	10	15	21 ^B
1	8	18	30	44	60	78
2	21	45	72	102	135	171
3	40	84	132	184	240	300
4	65	135	210	290	375	465
5	96 ^C	198	306	420	540	666 ^D

그림 2. 적분 영상의 윈도우 패치 계산

(그림 2)는 적분 영상을 사용하여 녹색 사각 영역으로 나타낸 특징점 윈도우 패치의 밝기 값을 구하는 예를 보이고 있다. 식 (1)을 통하여 생성된 적분영상은 어떠한 크기의 윈도우 패치 영역도 식 (2)와 같이 네 번의 메모리 접근을 통하여 구할 수 있다.

$$\Sigma = D - C - B + A \quad (2)$$

이러한 특성을 사용하여 윈도우 패치 크기의 증가에도 동일한 연산 속도를 가질 수 있다.

3. 적분영상 필터링을 사용한 특징점 매칭

특징점 매칭 방법은 SSD 매칭 방법을 사용한다. SSD 값의 계산은 식(3)을 통해 이루어진다.

$$\sum_{(x,y) \in W} (r(x,y) - t(x,y))^2 \quad (3)$$

식 (3)의 r 은 등록된 특징점의 윈도우 패치, t 는 현재 영상에서 추출된 특징점의 윈도우 패치를 의미한다. W 는 윈도우 패치 영역을 뜻하며 x,y 는 윈도우 영역에서의 위치를 나타낸다. 이 방법은 동일한 수의 특징점 매칭에서도 윈도우 패치의 크기에 따라 메모리 접근 횟수가 증가하여 계산 시간은 증가하게 된다.

특징점 매칭을 위한 윈도우 패치 영역의 제곱합의 차이가 적은 특징점은 매칭될 가능성이 높은 특징점을 의미한다. 특징점 윈도우 패치의 각 픽셀 밝기값의 제곱합의 차이가 적은 식 (4)와 같이 나타낼 수 있다.

$$f(r,t) = \sum_{(x,y) \in W} r(x,y)^2 - \sum_{(x,y) \in W} t(x,y)^2 \quad (4)$$

$f(r,t)$ 값의 계산은 생성된 적분영상을 사용하여 빠르게 계산된다. 영상에서 추출된 특징점 각각의 윈도우 패치와 특징점 매칭을 위한 윈도우 패치 사이를 식 (4)를 사용하

여 얻어진 값의 차이가 임계값 보다 작은 특징점만을 선택하여 SSD 매칭을 시도한다. (그림 3)은 적분영상을 사용한 특징점 필터링 과정의 이해를 돕기 위하여 나타낸 것으로 위의 그림은 기존의 SSD 매칭이고 아래의 그림은 제안된 방법을 사용한 SSD 매칭이다. 그리고 좌측의 그림은 현재영상에서 추출된 특징점이며, 우측의 그림은 기존에 등록된 특징점 집합을 나타내고 있다. 그림에 표기된 수치는 식 (4)를 통하여 계산된 $f(r,t)$ 값을 의미한다.

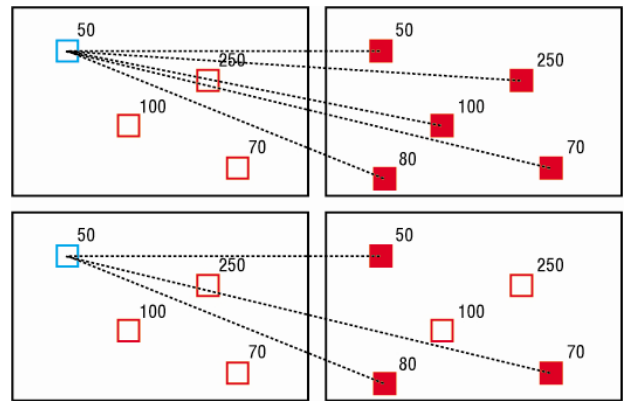


그림 3. 적분 영상 필터링을 사용한 특징점 매칭 : 기존의 SSD 매칭(위), 제안된 방법(아래)

(그림 3)의 위에 나타난 기존의 SSD 매칭 과정에서 하나의 특징점은 등록된 특징점 집합 전부와 SSD 매칭을 시도한다. 그러나 아래 그림의 제안된 방법은 윈도우 패치의 $f(r,t)$ 값의 차이가 일정 범위 내에 있는 특징점들과 비교를 시도하는 제안된 방법을 나타내고 있다. 이 과정을 통하여 특징점 매칭에 사용되는 특징점의 개수를 줄일 수 있어 기존의 SSD 매칭 방법에서 특징점이 매칭될 확률이 낮은 특징점과의 불필요한 매칭과정을 줄일 수 있다. (그림 4)를 통하여 적분 영상을 필터링을 통한 특징점 매칭 과정을 단계별로 설명한다.

- 단계 1. 픽셀 제곱 값을 사용하여 적분 영상 생성
 - 단계 2. 비교할 특징점 윈도우 패치의 $f(r,t)$ 값 계산
 - 단계 3. $f(r,t)$ 값의 차이 < 임계값 → 단계 3
 $f(r,t)$ 값의 차이 > 임계값 → 단계 2
 - 단계 3. (단계 3)에서 통과된 특징점은 SSD 매칭 시도
 - 단계 4. 현재 영상에서 추출된 특징점 집합에서 SSD 값이 가장 작은 값을 매칭된 특징점으로 선택
 - 단계 5. 현재영상에서 추출된 특징점에 대하여 (단계 2~4)를 반복

그림 4. 적분 영상 필터링을 사용한 특징점 매칭 과정

4. 실험 결과

본 논문에서 제안된 적분 영상을 사용한 특징점 필터링을 적용한 SSD 매칭 방법의 성능 평가를 위하여 5개의

실험 영상 세트를 사용하였다.



그림 5. 실험 영상

(그림 5)는 실험에 사용된 실험 영상 세트 중 하나이다. 좌측 그림은 기준이 되는 패턴 영상으로 추출된 특징점은 윈도우 패치를 빨간색 사각형으로 표기하였다. 우측 그림은 입력되는 영상에서 추출된 특징점으로 파란색 사각형으로 표기하였다.

실험은 좌측 그림의 패턴 영상에서 추출된 특징점과 우측 그림의 입력 영상에서 추출된 특징점과의 매칭을 시도하도록 하였다.

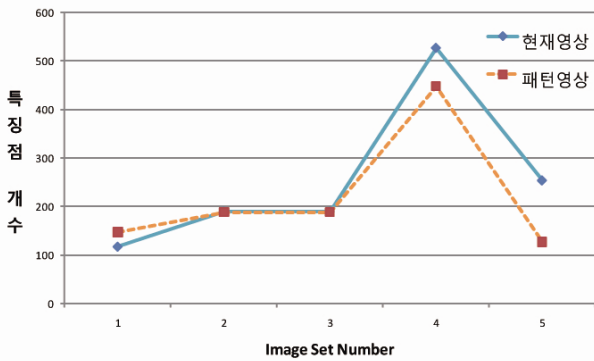


그림 6. 추출된 특징점 개수

(그림 6)은 실험 영상에서 추출된 특징점의 개수를 나타내고 있다. 4, 5번 실험 세트의 경우 패턴이 되는 영상 보다 입력되는 현재 영상의 추출된 특징점이 많다.

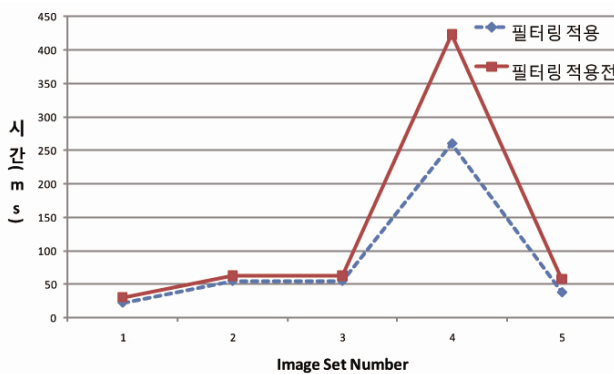


그림 7. 특징점 매칭 시간

(그림 7)은 특징점 매칭 시간의 실험 결과를 나타내고 있다. 대체적으로 필터링 적용전의 SSD 매칭은 필터링을 적용하기 전의 SSD 특징점 매칭보다 처리시간이 감소한 것을 확인 할 수 있다. 특히, 4, 5번 실험 세트의 경우와 같이 패턴 영상에서 추출된 특징점의 개수 보다 현재 영상에서 추출된 특징점이 많은 경우에서 처리 시간 단축이 두드러졌다. 이는 현재 영상에서 패턴의 특징점과는 다른 특징점들이 많이 추출되어 필터링을 통하여 일치할 가능성이 낮은 특징점들을 제하고 SSD 매칭을 시도하였기 때문으로 분석할 수 있다.

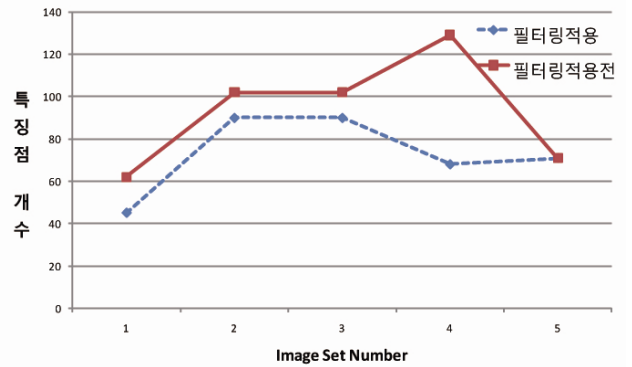


그림 8. 매칭된 특징점 개수

(그림 8)은 매칭된 특징점의 개수를 나타내고 있다. 필터링을 적용한 후 매칭된 특징점의 개수가 줄어드는 것을 확인 할 수 있다. 이때 정확한 매칭이 아닌 이상점 (outlier)을 포함한 결과이다. 이러한 매칭되는 특징점 개수의 저하는 필터링 임계값의 조절을 통하여 완화 시킬 수 있다.



그림 9. 매칭 결과(#set5):
(좌)필터링적용전, (우)필터링적용

5. 결론

본 논문에서는 적분 영상을 사용한 특징점 매칭의 경우 SSD 매칭 방법을 사용하는 경우 특징점의 개수의 증가 및 윈도우 패치 영역이 증가하는 경우 처리 시간의 증가가 불가피하다. 이 때문에 특징점 매칭 처리 시간을 줄이

기 위하여 매칭을 위한 특징점 집합에서 매칭의 가능성이 높은 특징점을 선택하여 특징점 매칭을 시도함으로써 전체 특징점 매칭 시간을 줄이도록 한다. 매칭에 사용되는 특징점의 개수를 줄이기 위하여 본 논문에서는 특징점 윈도우 패치의 픽셀 제공값을 사용하여 적분 영상을 생성하여 적분 영상 값의 차이를 사용하는 특징점 필터링 기법을 제안하였다. 이 방법을 적분 영상의 장점으로서 네 번의 메모리 접근을 통하여 특징점 윈도우 패치의 제공값의 차를 빠르게 구할 수 있다. 실험을 통하여 특징점 필터링을 적용하여 매칭에 사용되는 특징점 개수를 줄이고 이를 통하여 SSD 매칭 방법을 사용한 특징점 매칭의 처리 시간을 절감할 수 있음을 확인 하였다.

참고문헌

- [1] D. Lowe, "Distinctive image features from scale-invariant keypoints", *Int. Journal of Computer Vision*, Volume 60, Issue 2, pp. 91-110, 2004.
- [2] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded Up Robust Features", *ECCV*, pp. 404-417, 2006.
- [3] E. Rosten and T. Drummond. "Machine learning for high-speed corner detection", *9th European Conference on Computer Vision (ECCV'06)*, Volume 3951/2006, pp. 430-443, 2006.
- [4] G. Klein, D. Murray, "Parallel Tracking and Mapping for Small AR Workspaces", *6th IEEE and ACM International Symposium on in Mixed and Augmented Reality(ISMAR)*, pp. 225-234, 2007.
- [5] D. Keysers, T. Deselaers, T. M. Breuel, "Optimal Geometric Matching for Patch-Based Object Detection", *ELCVIA 6(1)* pp.44-54, 2007.