전방위 영상 생성을 위한 SIFT 연산량 감소

*문원준 *이재은 *서영호 *김동욱

*광운대학교 전자재료공학과

wonjun94@kw.ac.kr wodms6364@kw.ac.kr yhseo@kw.ac.kr dwkim@kw.ac.kr

Reducing the SIFT Calculation for Omni-directional Image Generation

*Moon, Won-Jun *Lee, Jae-Eun *Seo, Young-Ho *Kim, Dong-Wook

*Department of Electronic Materials Engineering, Kwangwoon University

요약

본 논문에서는 시장규모가 점진적으로 커지고 있는 VR의 콘텐츠 제작에 있어서 전방위 영상 생성을 위해 촬영한 영상들을 스티칭하는 데 소요되는 연산량을 감소시키는 방법을 제안한다. 제안하는 방법은 널리 사용되는 SIFT의 알고리즘에서 가장 많은 연산을 차지하는 가우시안 피라미드 생성 부분을 파라미터를 조절하여 감소시키고, 이를 통해 얻은 점들을 스티칭하는데 사용하여 영상을 합성하는 방법이다. 영상에 대한 평가는 변환에 의한 점들의 평균오차를 계산하여 유효성을 판단한다.

1. 서론

가상현실(Virtual Reality, VR)에 대한 시장규모가 확대됨에 따라 콘텐츠 시장 역시 대규모로 성장하고 있다[1]. VR 콘텐츠는 360도의 정보를 모두 포함해야 하는데, 이를 만족하는 영상을 전방위 영상이라고 한다. 전방위 영상을 만들기 위해서는 스티칭(stitching), 즉 획득한 영상을 붙이는 작업을 수행해야 하는데, 스티칭을 위해서는 영상에서 동일한 부분을 찾아 정합하는 과정이 필요하다.

스티칭에는 주로 SIFT 알고리즘이 사용되고 있는데, 이를 통해 특징점을 추출하고 각 영상간의 특징점을 비교하여 정합 여부를 판단하여 스티칭을 수행한다[2,3].

SIFT에서는 영상에 분산을 키워가면서 가우시안 필터를 반복 적용하여 피라미드를 만드는 과정이 포함되는데, 이 연산에 많은 시간이 소요되기 때문에 성능은 떨어지지만 속도를 향상시킨 SURF와 같은 다른 알고리즘도 연구되었다[4,5].

본 논문에서는 연산시간 문제를 해결하기 위해 가우시안 필터에서의 초기 분산과 피라미드의 한 옥타브 내에서의 레이어 수를 조절한다. 이에 따라 SIFT 알고리즘에서의 연산량은 감소되지만, 스티칭의 결과는 근사하게 나오는 방법을 제안한다.

2. 연산량 감소 방법

2.1. 필터의 특성

본 논문에서는 SIFT 알고리즘을 사용하는데, SIFT는 크게 특징점을 찾는 과정과 특징점의 특징을 추출하는 과정으로 나뉜다. 특징점을 찾는 과정은 영상을 가우시안 필터로 반복 처리하게 된다.

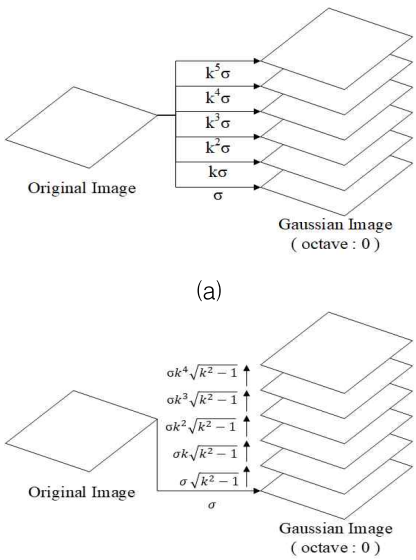


그림 1. 가우시안 피라미드 구성 방법: (a) 기본적인 방법, (b) 캐스케이드 특성을 통한 계산량 감소 방법.

그림 1(a)은 원본 영상에 분산 값을 증가시키어 가우시안 필터링을 수행하여 피라미드를 만드는 과정인데, 그림 1(b)와 같이 가우시안 필터의 캐스케이드 특성을 이용하여 원본 영상이 아닌 이전에 처리된 영상에 누적으로 필터링하는 방식을 통해 연산량을 감소시킬 수 있다[6].

영상의 해상도가 4,000×3,000인 영상에 대해 8 옥타브까지 SIFT 특징점 추출을 위한 필터링 연산을 했을 때, 사용되는 MAC(multiplier-accumulator)의 연산 횟수가 (a)방식은 7.4×10^{10}번의 연산이 필요하고, (b)는 3.2×10^{10}번의 연산이 필요하여 (a)의 연산량이 약 2.3배 많았다.

표 1. 분산 값에 따른 필터사이즈 (레이어는 6개로 고정)

Sigma	Basic method						Cascade method					
	I0	I1	I2	I3	I4	I5	I0	I1	I2	I3	I4	I5
0.6	7	7	9	11	13	17	7	5	7	7	9	11
0.8	7	9	11	15	17	21	7	7	7	9	11	13
1	9	11	15	17	21	27	9	7	9	11	13	17
1.2	11	13	17	21	25	31	11	9	11	13	17	21
1.4	13	15	19	23	29	37	13	11	13	15	19	23
1.6	15	17	21	27	33	43	15	11	13	17	21	27
1.8	15	19	25	31	37	47	15	13	15	19	23	29
2	17	21	27	33	41	53	17	13	17	21	27	33

표 2. 레이어 개수에 따른 필터사이즈 (분산은 1.6 고정)

Layer	Basic method					Cascade method				
	L4	L5	L6	L7	L8	L4	L5	L6	L7	L8
I0	15	15	15	15	15	15	15	15	15	15
I1	27	19	17	17	17	23	15	11	9	9
I2	53	27	21	19	19	45	19	13	11	9
I3	103	37	27	23	21	91	27	17	13	11
I4	▪	53	33	27	23	▪	37	21	15	13
I5	▪	▪	43	31	27	▪	▪	27	17	15
I6	▪	▪	▪	37	31	▪	▪	▪	21	15
I7	▪	▪	▪	▪	35	▪	▪	▪	▪	19

2.2 파라미터 변화

가우시안필터는 분산에 따라 필터의 사이즈가 달라지는데, SIFT에서의 분산은 초기로 설정해주는 분산 값과, 레이어(*Ln*)의 개수가 영향을 미친다. 분산 값과 레이어 개수에 따른 필터의 사이즈는 표 1, 2 와 같다.

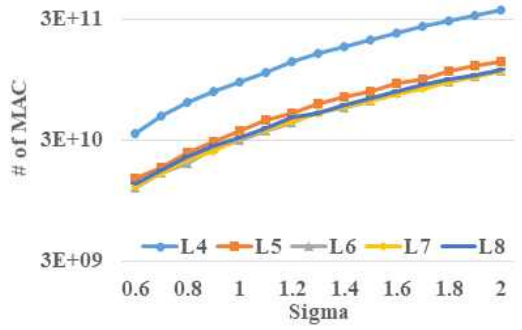
3. 실험 및 결과

3.1 실험환경 및 조건

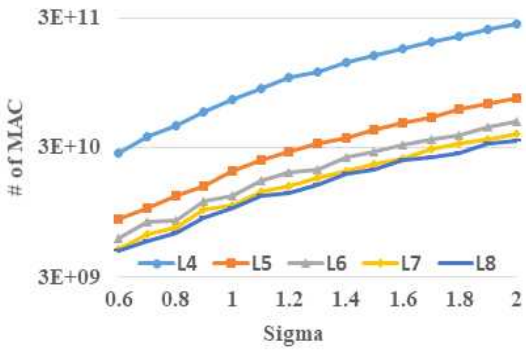
본 실험에서는 겹치는 영역이 존재하는 2개의 영상을 한 세트로 하여 총 10세트에 대해 분산과 레이어의 개수를 조절해가며 스티칭을 수행하였다. 생성된 영상에 대한 평가는 SIFT 알고리즘을 통해 매칭이 된다고 정해진 점과 RANSAC을 통한 호모그래피(homography)를 구하여 실제 변환을 통해 얻어지는 점의 유클리드 거리를 계산하여 평균을 내는 방식으로 오차 값을 계산하였다.

3.2 연산량의 감소

초기 분산은 0.6~2, 레이어는 4~8의 범위에 대해서 각 파라미터의 변화에 따른 그림 1(a)와 (b) 방식에 대한 MAC의 연산수를 그림 2에서 나타내었다. 영상의 해상도는 2.1절과 마찬가지로 4,000×3,000에 대해 수행하였다.



(a)



(b)

그림 2. 분산과 레이어의 개수에 따른 MAC연산 횟수: (a) 기본적인 피라미드, (b) 캐스케이드 특성을 이용한 피라미드.

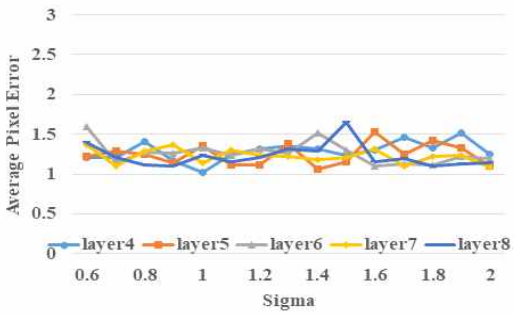


그림 3. 분산과 레이어의 개수에 따른 스티칭 오차

3.3 초기 분산과 레이어의 개수에 따른 결과 비교

그림 3에서는 파라미터에 상관없이 화소의 평균 제곱 오차가 1~1.5픽셀 사이에 위치하는 것을 확인할 수 있는데, 이 오차는 사람이 구분할 수 없을 정도로 미세하다.

따라서 초기 분산과 레이어 선택에 따른 스티칭 결과는 비슷하다고 볼 수 있으므로, 2.2절에서의 가장 적은 MAC의 수를 가지는 값이 가장 효율적이다.

가장 적은 MAC의 수를 가지는 8개의 레이어를 가질 때, 분산에 따른 스티칭 결과를 그림 4에서 나타내었다.

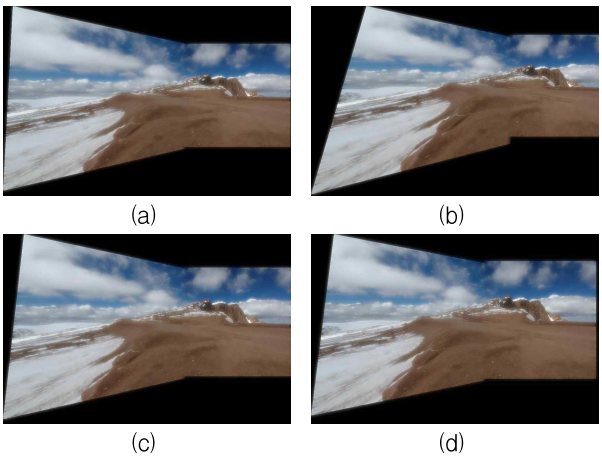


그림 4. 레이어 8개일 때의 분산의 변화에 따른 스티칭 결과 (a) 0.6, (b) 1.0, (c) 1.6, (d) 2.0

4. 결론

본 논문에서는 전방위 영상 생성과정에서의 연산량 감소를 위해 초기 분산과 레이어의 개수를 선정하는 방법을 제시하였다. 가장 연산량이 적은 초기분산 0.6과 레이어 8개에 대해 비교했을 때, 전방위 영상의 결과는 비슷하지만 계산상에서 약 42.9배의 이득을 얻는다. 두 장의 영상에 대해서만 결과를 비교한 것이기에, 실제 여러 장의 영상을 합성하여 만드는 전방위 영상에서는 더 큰 효율을 가질 것이라 기대한다.

감사의 글

이 논문은 2016 년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (NRF-2016R1D1A1B03930691)

참고문헌

[1] Institute for Information & Communications Technology Promotion: “Technology Development Trend and Market Forecast of VR/AR,” Weekly Technology Trends, Vol. 1803, pp. 48-55, July, 2017.

[2] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints,” International Journal of Computer Vision, Vol. 60, No. 2, pp. 91-110, Jan. 2004.

[3] D. G. Lowe, “Automatic Panoramic Image Stitching using Invariant Features,” International Journal of Computer Vision, Vol. 74, No. 1, pp. 59-73, Aug. 2007.

[4] F. C. Huang, S. Y. Huang, J. W. Ker Y. C. Chen “High-Performance SIFT Hardware Accelerator for Real-Time Image Feature Extraction,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 22, No. 3, pp. 340-351, Mar. 2012.

[5] H. Bay, T. Tuytelaars, L. V. Gool “SURF: Speeded Up Robust Features,” European Conference on Computer Vision, pp. 404-417, 2006.

[6] R. Hess, “An Open-Source SIFT Library,” ACM Multimedia, pp. 1493-1496, Oct. 2010.