

일반논문 (Regular Paper)

방송공학회논문지 제23권 제4호, 2018년 7월 (JBE Vol. 23, No. 4, July 2018)

<https://doi.org/10.5909/JBE.2018.23.4.559>

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

## 영상 스티칭 관점에서 SIFT 특징점 추출시간 감소를 위한 파라미터 분석

문 원 준<sup>a)</sup>, 서 영 호<sup>b)</sup>, 김 동 육<sup>a)†</sup>

### Parameter Analysis for Time Reduction in Extracting SIFT Keypoints in the Aspect of Image Stitching

Won-Jun Moon<sup>a)</sup>, Young-Ho Seo<sup>b)</sup>, and Dong-Wook Kim<sup>a)†</sup>

#### 요 약

최근 가상현실(VR, Virtual Reality) 등 가장 많은 분야에서 가장 활발히 응용되고 있는 영상매체 중 하나가 전방위 영상 또는 파노라마 영상이다. 이 영상은 다양한 방법으로 획득된 영상들을 스티칭하여 생성하는데, 그 과정에서 스티칭에 필요한 특징점을 추출하는데 가장 많은 시간이 소요된다. 이에 본 논문은 현재 가장 널리 사용되고 있는 SIFT 특징점을 추출하는 연산시간을 감소하는 것에 목적을 두고 SIFT 특징점을 추출에 관여하는 파라미터들을 분석한다. 본 논문에서 고려하는 파라미터는 가우시안 필터링에 사용되는 가우시안 커널의 초기 표준편차, 국소극점을 추출하기 위한 가우시안 차영상군의 수, 그리고 유타브 수의 세 가지이다. SIFT 알고리즘으로는 이 알고리즘을 제안한 Lowe 방식과 컨볼루션 캐스케이드(convolution cascade) 방식인 Hess 방식을 고려한다. 먼저 각 파라미터 값이 연산시간에 미치는 영향을 분석하고, 실제 스티칭 실험을 수행하여 각 파라미터가 스티칭 성능에 미치는 영향을 분석한다. 마지막으로 두 분석결과를 토대로 성능저하 없이 연산시간을 최소로 하는 파라미터 값을 추출한다.

#### Abstract

Recently, one of the most actively applied image media in the most fields such as virtual reality (VR) is omni-directional or panorama image. This image is generated by stitching images obtained by various methods. In this process, it takes the most time to extract keypoints necessary for stitching. In this paper, we analyze the parameters involved in the extraction of SIFT keypoints with the aim of reducing the computation time for extracting the most widely used SIFT keypoints. The parameters considered in this paper are the initial standard deviation of the Gaussian kernel used for Gaussian filtering, the number of gaussian difference image sets for extracting local extrema, and the number of octaves. As the SIFT algorithm, the Lowe scheme, the originally proposed one, and the Hess scheme which is a convolution cascade scheme, are considered. First, the effect of each parameter value on the computation time is analyzed, and the effect of each parameter on the stitching performance is analyzed by performing actual stitching experiments. Finally, based on the results of the two analyses, we extract parameter value set that minimize computation time without degrading.

Keyword : omni-directional image, image stitching, SIFT keypoints, SIFT parameters, Gaussian pyramid

Copyright © 2018 Korean Institute of Broadcast and Media Engineers. All rights reserved.

"This is an Open-Access article distributed under the terms of the Creative Commons BY-NC-ND (<http://creativecommons.org/licenses/by-nc-nd/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited and not altered."

## I. 서 론

최근 디지털 영상미디어에서 가장 활발히 응용되는 미디어 중 하나가 전방위(Omnidirectional) 영상이다<sup>[1]</sup>. 전방위 영상은 다수의 2차원 카메라로 획득한 영상을 서로 연결(스티칭, stitching)하여 얻는다. 여기에 사용하는 영상은 어안(魚眼, fisheyes)렌즈를 사용하는 카메라로 촬영한 영상이거나 6장 이상의 영상을 동시에 촬영할 수 있는 리그(rig)를 사용한 영상이다.

스티칭 과정은 인접한 두 영상을 연결하는 과정으로, 두 영상의 특징점을 정합(matching)하고, 그 결과로 얻어지는 호모그래피(homography)를 사용하여 두 영상의 각 화소들의 위치를 보정하고 색상을 보정하여 연결된 영상을 얻는다. 스티칭에서 가장 많은 시간이 소요되는 과정은 특징 점을 추출하는 과정인데, 현재 이 특징점을 추출하는 알고리즘으로 SIFT(Scale Invariant Feature Transform)<sup>[2][3]</sup>를 가장 많이 사용하고 있다. SIFT는 가우시안(Gaussian) 필터링을 수행한 영상들의 차이영상들로부터 극점들을 찾고, 그 극점들 중 특징점으로 사용하기 어려운 점들을 제외한 나머지 점들을 특징점으로 한다. VGA 영상 한 장에 대해 Intel Core 2 Duo 2.09 GHz CPU를 사용하는 PC에서 소프트웨어로 SIFT 특징점을 추출하는데 약 3초의 시간이 소요되며, 이 중 72.85%가 가우시안 필터링을 수행하는 과정에서 소요된다<sup>[4]</sup>.

이에 SIFT 또는 그에 버금가는 특징점을 추출하는 시간을 단축시키는 연구가 진행되어 있는데, 그 중 Grabner 등은 집적영상(integral image) 기법을 이용하여 SIFT 특징 점 연산시간을 줄였다<sup>[5]</sup>. 또한 PCA(Principal Component

Analysis) 기법을 사용하여 연산시간은 다소 늘어나지만 특정점 정합에서 시간을 단축시킬 수 있는 방법도 발표되었 다<sup>[6]</sup>. 또한 SURF(Speeded-Up Robust Features)라는 이름으로 Hessian 필터를 단순화하고 기술자를 생성함에 있어 Haar 웨이블릿 필터를 사용하여 복잡도를 감소시키는 방법도 제안되었다<sup>[7]</sup>. Rob Hess는 캐스캐이드(cascade) 필터링 방법을 가우시안 파라미드 생성에 적용하여 연산시간을 단축시켰다<sup>[8]</sup>. 이 방법은 원래 SIFT와 조금 다른 특징점들이 도출되지만 스티칭을 수행하는 데에는 별 문제가 없다. 이 외 SIFT 전체를 ASIC 하드웨어로 구현하거나<sup>[4]</sup> 특징점을 추출하는 부분만을 FPGA로 구현한 방법<sup>[9]</sup>도 발표되었으며, 하드웨어/소프트웨어 융합설계(co-design) 방법도 발표되었다<sup>[10]</sup>. 또한 GPU(Graphic Processing Unit)을 이용하여 고속화한 방법도 연구되었다<sup>[11][12]</sup>.

영상에서 특징점은 다양하게 정의할 수 있으나, SIFT의 특징점이 다양한 응용분야에서 가장 좋은 효율을 보이는 것으로 알려져 있다<sup>[13]</sup>. SIFT은 기본적으로 가우시안 저대역통과(low-pass) 필터를 적용하여 주위의 화소보다 크거나 작은 극한값을 갖는 화소들을 찾는 것이다. 이 과정은 원 영상의 해상도를 확대하거나 축소하면서 지속적으로 적용한다. 따라서 SIFT 특징점 추출과정은 가우시안 필터의 크기, 한 해상도에서 추출하는 가우시안 필터링 결과영상의 개수, 영상의 해상도를 얼마까지 축소하면서 적용할 것인가 등 다양한 파라미터를 갖고 있다. 본 논문에서는 SIFT 특징점들의 사용분야를 영상 스티칭에 국한하고, 이들 파라미터들이 특징점 추출에 미치는 영향과 추출된 특징점들을 분석한다. 이를 토대로 전방위 영상생성을 위한 스티칭을 타겟으로 하여 특징점 추출시간을 단축하는 파라미터 세트를 제안하고, 그 결과 특징점들이 스티칭을 문제없이 수행함을 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 스티칭 과정과 그에 필요한 특징점을 추출하는 SIFT 알고리즘을 원래의 방법<sup>[2]</sup>과 캐스캐이드 필터링을 수행하는 방법<sup>[8]</sup>을 비교하여 설명한다. 3장에서는 SIFT 알고리즘의 파라미터들이 연산시간과 추출되는 특징점의 개수에 미치는 영향을 분석한다. 4장에서는 실제 스티칭 실험을 토대로 각 파라미터가 스티칭 성능에 미치는 영향을 분석하고, 그 결과를 토대로 성능저하 없이 연산시간을 최소로 하는 파라미터 세트를 추출한다. 마지막으로 5장에서 본 논문의 결론을 맺는다.

a) 광운대학교 전자재료공학과(Department of Electronic Materials Engineering, Kwangwoon University)

b) 광운대학교 인체니움학부대학(Ingenium College of Liberal Arts, Kwangwoon University)

‡ Corresponding Author : 김동욱(Dong-Wook Kim)  
E-mail: dwkim@kw.ac.kr  
Tel: +82-2-940-5167

ORCID:https://orcid.org/0000-0002-4668-743X

※ This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2016R1D1A1B03930691).

※ The present Research has been conducted by the Research Grant of Kwangwoon University in 2018.

· Manuscript received June 18, 2018; Revised July 5, 2018; Accepted July 5, 2018.

## II. 스티칭과 SIFT 특징점 추출

본 논문에서는 SIFT 특징점들을 영상의 스티칭에 사용한다는 것을 전제하고 있다. 따라서 영상 스티칭에 대해 먼저 설명한다.

### 1. 두 영상의 스티칭 과정

그림 1은 두 영상을 스티칭하는 과정을 도식화한 것이다. 여기서는 각 영상의 특징점을 정합하는 방법을 가정한 것이다. 따라서 가장 먼저 각 영상에서 같은 성질의 특징점을 찾고, 두 영상의 특징점을 서로 일치하는 특징점들 쌍을 찾는다. 이 쌍들 중 가장 일치도가 높은 네 쌍으로 두 영상간 기하학적 상관관계를 나타내는 호모그래피 행렬을 구한다. 호모그래피 행렬이 구해지면 한 영상을 기준으로 다른 영상의 화소 위치를 기하학적으로 변환하여 두 영상의 화소들을 일치시키고, 마지막으로 두 영상의 컬러정보를 일치시키기 위한 블렌딩(blending)과정을 거치면 스티칭이 완성된다.

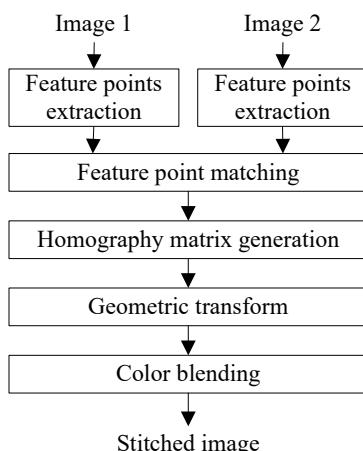


그림 1. 두 영상의 스티칭 과정

Fig. 1. Stitching procedure for two images

호모그래피 행렬로 스티칭되는 영상을 기하학적으로 변환한 결과에 대해서 식 (1)과 같이 변환된 점과 정합된 기준 영상의 점간의 평균 거리오차  $d_{error}$ 를 계산할 수 있으며, 이 값이 작을수록 특징점이 잘 추출되었다고 판단하기도 한다.

$$d_{error} = \frac{1}{N} \sum_{i=1}^N \sqrt{(x_{Ti} - x_{Mi})^2 + (y_{Ti} - y_{Mi})^2} \quad (1)$$

여기서  $N$ 은 정합된 특징점 쌍의 수이고,  $(x_{Ti}, y_{Ti})$ 와  $(x_{Mi}, y_{Mi})$ 는 각각 호모그래피 행렬로 변환된  $i$ 번째 화소와 그에 대응하는 기준영상의 화소의 좌표를 각각 나타낸다.

### 2. SIFT 특징점 추출

본 연구에서는 그림 1의 스티칭에 사용되는 특징점으로 SIFT 알고리즘으로 추출된 특징점을 사용한다. 그림 2에 SIFT 특징점을 추출하는 과정을 흐름도로 나타내었다. SIFT는 특정 해상도의 영상으로 특징점을 추출하는 단위를 옥타브(octave)라고 하며, 한 옥타브에서 그 상위 옥타브로 진행할 때 현재 영상을 가로와 세로를 1/2로(크기를 1/4로) 축소(down-sampling)하여 사용한다. SIFT의 초기 영상으로는 필요에 따라 원 영상을 가로와 세로로 2배 확장(up-sampling)하여 사용할 수도 있다.

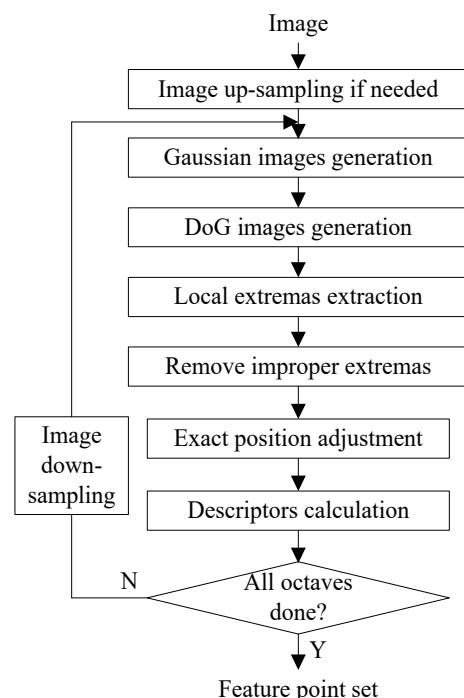


그림 2. SIFT 특징점 추출 과정

Fig. 2. Procedure to extract SIFT key points

SIFT의 옥타브 수는 초기 영상을 확대하지 않는 경우 입력영상의 해상도가  $(w, h)$  일 때  $\log_2 \min(w, h) - 2 > 0$ 를 만족하는 수만큼 가능하지만, 사용자가 임의로 정할 수 있다. 그림 3(a)에 SIFT의 국부극점(local extrema)을 구하는 방법을 도식화하였는데<sup>[2]</sup>, 이 그림은 그림 2에서 가우시안 영상생성(Gaussian images generation)부터 국부극점 추출(Local extrema extraction)까지를 포함한다. 먼저 옥타브 당  $s + 3$  개의 가우시안 영상을 생성하는데, 여기에는 식 (2)의 가우시안 커널(kernel)을 이산화(discretization)한 필터를 사용하여 저대역통과 필터링을 수행한다. 이 때  $s$ 는 DoG (Difference of Gaussian) 영상군에서 극점을 추출하는 단위영상군의 개수를 나타낸다.

$$K_G(x, y; \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (2)$$

그림 3에서 보듯이 상위 가우시안 영상에 사용되는 커널의 표준편차를  $\sigma, k\sigma, k^2\sigma, \dots$  ( $k = 2^{1/s}$ )로 증가시키는데, 따라서 상위 가우시안 영상일수록 더 강한 저주파 필터링이 수행된다. 한 옥타브의 가우시안 영상군에 대하여 인

접한 두 가우시안 영상의 차영상을 구하여 DoG 영상군을 형성하는데, 옥타브 당  $s + 2$  개의 DoG 영상이 생성된다. 이 영상군에서 특정 영상의 특정 화소에 대해 주위의  $3 \times 3 \times 3$  화소를 비교하여 그 화소가 최대 또는 최솟값을 가질 때 그 화소가 국부극점이 된다(그림 3에서 DoG의 맨 하위 세 영상 내부에 흰색으로 표시한 부분 참조. 여기서 대상화소를 회색으로 표시하였다). 따라서 국부극점은 해당 DoG 영상과 그 상위 및 하위의 세 DoG 영상이 그 극점을 추출하는 단위 DoG 영상군(국소극점 추출 단위 DOG 영상군)이 되며, 각 옥타브에서  $s$  개의 국소극점 추출 단위 DoG 영상군이 생성된다.

추출된 극점을 중 영상의 에지(edge)에 너무 근접하거나 극값자체가 작은 화소들은 부적절하다고 판단하고 제거한다(그림 2의 Remove improper extrema). 또한 나머지 극점들에 대해서는 화소단위보다 더 정확한 위치를 파악하게 되고(그림 2의 Exact position adjustment), 각 극점에 대해서 방향성에 대한 값으로 128차원의 기술자(descriptor)를 생성한다(그림 2의 Descriptors calculation). 특정 옥타브에서 그 상위 옥타브로 진행할 때는 표준편차가  $k^s\sigma = 2\sigma$ 인 커널로 필터링한 영상을 가로, 세로로 1/2씩 축소하여 사용한다.

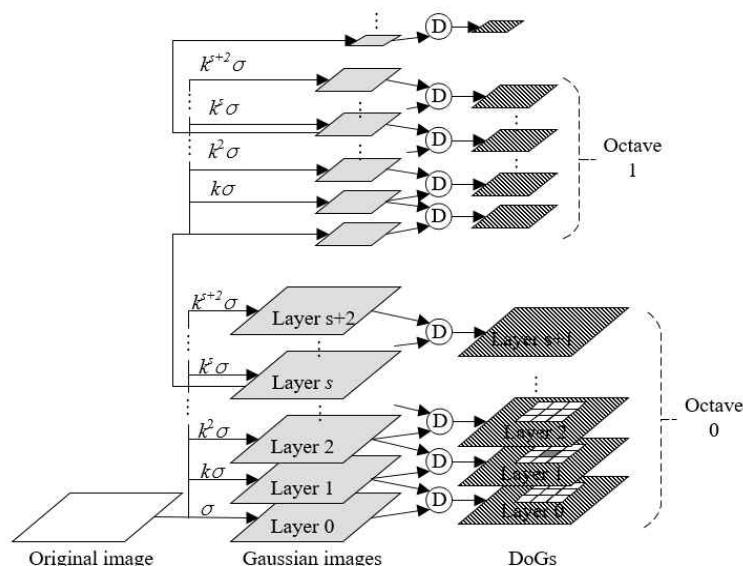


그림 3. SIFT 극점 추출  
Fig. 3. SIFT extrema extraction

### 3. 특징점 추출시간 축소 방법의 예: Hess 방법

표 1에 그림 3의 방법으로 국소극점을 추출하여 그림 2의 SIFT 특징점을 추출하는 방법의 단계별 소요시간을 비율로 나타내었다<sup>[4]</sup>. 이 데이터는 PC에서 소프트웨어로 구현한 것을 수행한 결과로, 사용하는 PC의 사양에 따라 절대적인 소요시간이 달라지므로 상대적인 비율을 보였다. 이 표에 보인 바와 같이, SIFT 단계 중 가우시안 영상을 생성하는 단계가 72.85%로 전체의 약 3/4의 많은 시간이 소요된다. 이것은 가우시안 커널을 적용한 컨볼루션(convolution) 연산을 수행하여야하기 때문이다.

표 1. SIFT 단계별 소요시간 비교

Table 1. Comparison of the time used for SIFT steps

Steps in algorithm	Relative time
Gaussian images generation	72.85%
DoG images generation	0.14%
Local extrema extraction	5.71%
Remove improper extrema and exact position adjustment	2.85%
descriptors calculation	18.57%

그림 4는 가우시안 영상 생성시간을 단축시키기 위해 제안된 한 방법<sup>[8]</sup>을 보였다. 이 방법은 식 (3)의 캐스케이드(cascade)식을 이용하여 식 (4)의 캐스케이드 방식으로 필터링을 수행하는 방법을 제안하였다. 식 (4)에서  $G_\sigma(I)$ 는 영상  $I$ 를 표준편차가  $\sigma$ 인 가우시안 커널로 필터링한다는 것을 의미한다.

$$(k^n\sigma)^2 = (k^{n-1}\sigma)^2 + (k^{n-1}\sigma\sqrt{k^2-1})^2 \quad (3)$$

$$G_{k^n\sigma}(I) = G_{k^{n-1}\sigma\sqrt{k^2-1}} [G_{k^{n-1}\sigma}(I)] \quad (4)$$

즉, 그림 3의 방법에서는 한 옥타브의 모든 가우시안 영상은 그전 옥타브에서 가져온 영상에 표준편차를  $k$  ( $k = s^{1/s}$ )배씩 증가하면서 필터링을 수행하였으나, 그림 4의 방법은 바로 전에 필터링된 가우시안 영상에 표준편차를  $\sqrt{k^2-1}$  배 증가시킨 커널을 적용하여 필터링한다. 이 두 방법에서 사용하는 이산 가우시안 필터의 텁(tap)수를

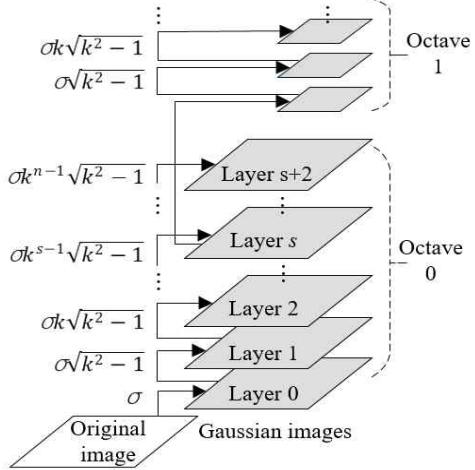


그림 4. Hess의 가우시안 영상 생성 방법

Fig. 4. Hess's scheme to generate Gaussian images

표 2에 보였다. 이 표의 값은 초기  $\sigma$ 를 1.6으로 하였고,  $s$ 를 3으로 한 결과이다. 표에서 보듯이 Hess 방식이 훨씬 적은 텁수를 사용함을 알 수 있고, 실제 컨볼루션 연산에서 MAC(Multiplication and Accumulation) 수가 Hess 방식이 Lowe 방식의 약 43%만 수행하면 된다.

표 2. 가우시안 커널 텁 수

Table 2. Numbers of Gaussian kernel taps

variance	Lowe scheme <sup>[2]</sup>	Hess scheme <sup>[4]</sup>
$\sigma$	$15 \times 15$	$15 \times 15$
$k\sigma$	$17 \times 17$	$11 \times 11$
$k^2\sigma$	$21 \times 21$	$13 \times 13$
$k^3\sigma$	$27 \times 27$	$17 \times 17$
$k^4\sigma$	$33 \times 33$	$21 \times 21$
$k^5\sigma$	$43 \times 43$	$27 \times 27$

한 가지 주의할 점은 두 방식이 한 옥타브에서 상위 옥타브로 진행될 때 약간의 차이를 보인다는 것이다. 두 방식 모두 특정 옥타브에서  $k^s\sigma = 2\sigma$ 로 필터링한 영상을 축소하여 상위 옥타브의 가우시안 영상을 생성하는 것과 하위 옥타브에서 축소되어온 영상을 이미 표준편차가  $\sigma$ 인 커널로 필터링한 것으로 간주하는 것은 같다. 그러나 그 다음 가우시안 영상을 생성하기 위하여 Lowe 방식은 표준편차가  $k\sigma$ 인 커널로 필터링 하는 반면, Hess 방식은 표준편차

가  $\sigma\sqrt{k^2-1}$ 인 커널로 필터링하기 때문에 옥타브 1의 두 번째 가우시안 영상부터는 서로 다른 가우시안 영상을 생성해낸다. 그림 5에  $s=3$ ,  $\sigma=1.6$  일 때 두 방식으로 옥타브 0과 옥타브 1에서 추출한 SIFT 특징점들(그림에서 노란색 점들)을 보이고 있는데, 옥타브 0에서 추출한 특징점들은

거의 동일하나(연산 오차로 한 두 개 정도 달라짐), 옥타브 1에서 추출한 특징점들에서는 많은 차이를 보이는 것을 확인할 수 있다. 참고로 한 특징점이 한 개의 화소이므로 영상 내에서 확인이 어려워 그림 5의 특징점들은 특징점 화소를 기준으로  $5\times 5$  화소를 표시한 것이다.

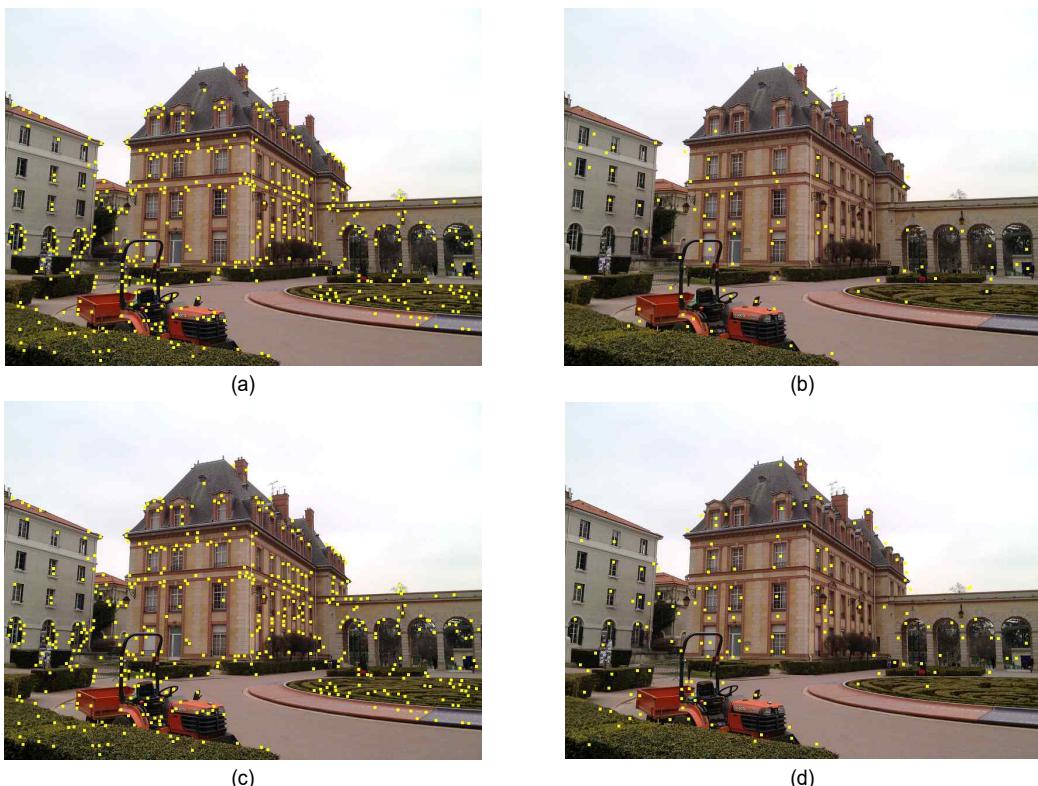


그림 5. 추출한 특징점들의 예: Lowe 방식; (a) 옥타브 0, (b) 옥타브 1, Hess 방식; (c) 옥타브 0, (d) 옥타브 1

Fig. 5. Examples of extracted keypoints: Lowe scheme; (a) octave 0, (b) octave 1, Hess scheme; (c) octave 0, (d) octave 1



그림 6. 두 영상을 스티칭한 결과의 예; (a) Lowe 방식, (b) Hess 방식

Fig. 6. Examples of stitched results of two images; (a) Lowe scheme (b) Hess scheme

그림 6은 Lowe 방식과 Hess 방식으로 각각 추출한 특징 점들로 그림 1의 스티칭 과정을 수행한 결과의 예를 보이고 있다. 그림 5에서 보인 바와 같이, 두 방식에서 추출된 특징 점들은 다르지만 두 방법 모두 문제없이 스티칭을 잘 수행하는 것을 볼 수 있다. 두 방식에서 식 (1)의 평균거리오차  $d_{error}$ 는 각각 1.205와 1.130이었다.

### III. SIFT 파라미터에 따른 특징점 추출시간 및 개수 분석

본 장에서는 SIFT 특징점 추출과정 중 가우시안 영상을 생성하는 단계에서 사용하는 파라미터들이 특징점 추출시간과 추출된 특징점 수에 미치는 영향을 분석한다. 여기서는 가우시안 커널의 초기 표준편차 값( $\sigma$ ), 극소극점 추출을 위한 단위 DoG 영상군의 수( $s$ ), 그리고 옥타브 수를 고려한다.

#### 1. 가우시안 커널의 초기 표준편차( $\sigma$ )

일련의 SIFT 가우시안 영상군(이것을 가우시안 피라미드(pyramid)라 부르기도 함)에서 한 옥타브 당  $s+3$ 개의 가우시안 영상을 생성하며, 여기에 사용되는 커널은 표준 편차가  $\sigma$ 부터  $k$ 배씩 증가시켜  $k^{s+2}\sigma$ 까지 사용한다. 따라서 초기  $\sigma$  값이 정해지면 사용되는 커널들의 크기가 정해지며, Lowe<sup>[2]</sup>는  $\sigma = 1.6$ 이 가장 좋은 성능을 보인다고 주장하였다, 그 근거로는 회전 등의 기하학적 변형에도 같은 특징점들이 추출되는 반복성을 제시하였다<sup>[2]</sup>. 그러나 응용 분

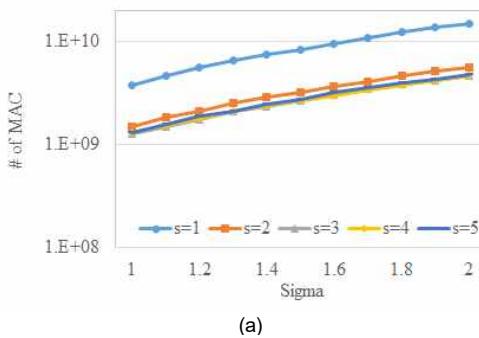
야가 본 논문에서 다루는 영상 스티칭의 경우에는 특징점 추출의 반복성에 직접적인 의미를 부여하기는 어렵다.

가우시안 필터링은 커널의 크기, 즉 표준편차의 크기가 클수록 강한 저주파통과 필터링을 수행하며, 표준편차가 클수록 이산화한 텁 수는 증가한다. 표 3은 Lowe 방식<sup>[2]</sup>과 Hess 방식<sup>[4]</sup>에 대해  $s = 3$ 인 경우 초기  $\sigma$ 가 0.6에서 2까지 변할 때 한 옥타브의 각 레이어(Li)에서 사용되는 커널의 텁 수를 보이고 있다. 여기서 각 수는 한쪽 방향의 텁 수를 나타내는 것으로, 모든 커널은 정방형 2차원이다. 표에서 보듯이 초기  $\sigma$ 가 커짐에 따라 텁 수가 크게 증가하며, 영상에 대해서는 2차원 커널이 사용되므로 연산량의 증가는 가중된다. 그림 7에 가우시안 필터링에 필요한 연산량을 MAC 수로 나타내었는데, 이 값들은 해상도 816×612 영상 20개에 대하여 옥타브 수를 6으로 하여 구한 영상 당 평균 값들이다. 종축이 log-scale이므로 초기  $\sigma$ 가 증가함에 따라 연산량이 기하급수적으로 증가함을 알 수 있다.

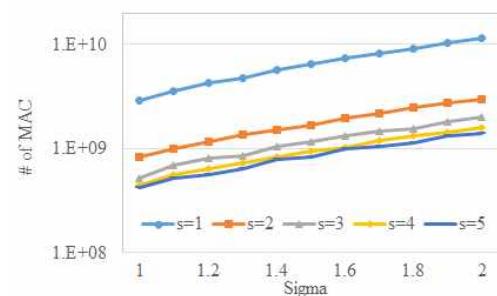
표 3. 초기  $\sigma$ 에 따른 가우시안 커널 텁 수( $s = 3$ )

Table 3. Numbers of Gaussian kernel taps for the initial  $\sigma$  ( $s = 3$ )

$\sigma$	Lowe scheme[2]						Hess scheme[4]					
	L1	L2	L3	L4	L5	L6	L1	L2	L3	L4	L5	L6
1.0	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	27	35	13	11	13	15	19	25
1.6	15	17	21	29	35	45	15	11	13	17	21	27
1.8	15	18	25	31	37	47	15	13	15	19	23	29
2.0	17	21	27	33	41	53	17	13	17	21	27	33



(a)

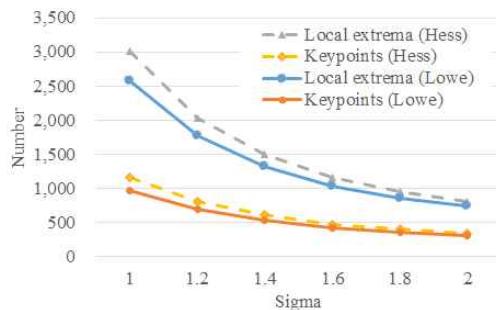


(b)

그림 7. 가우시안 커널의 초기  $\sigma$ 와  $s$ 에 따른 가우시안 필터링 연산량; (a) Lowe 방식, (b) Hess 방식

Fig. 7. Amount of calculation for the initial  $\sigma$  of the Gaussian kernel and  $s$ ; (a) Lowe scheme (b) Hess scheme

초기 커널의 크기가 달라지면 필터링 강도가 달라짐에 따라 추출된 특징점의 위치뿐만 아니라 개수도 달라진다. 그림 8은 초기  $\sigma$  값을 달리하였을 때 추출되는 국소극점 수와 부적절한 극점들을 제거한 최종의 특징점 수를 보이고 있다. 이 그림의 값들은 그림 7과 동일한 영상에 대해 동일한 조건으로 실험한 결과이다. 그림에서 보는 바와 같이, 초기  $\sigma$  값이 커짐에 따라 추출된 국소극점과 특징점 수 모두 급격히 줄어드는 것을 볼 수 있다.

그림 8. 초기  $\sigma$ 에 따른 특징점 수Fig. 8. Numbers of keypoints for initial  $\sigma$ 

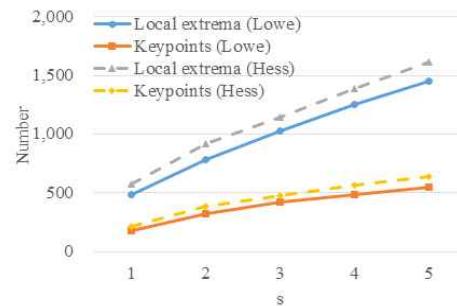
## 2. 국소극점 추출 단위 DoG 영상군의 개수( $s$ )

가우시안 피라미드를 생성할 때 옥타브 당 가우시안 영상 수는  $s + 3$ 개이므로 국소극점을 추출하는 단위 DoG 영상군의 개수인  $s$ 도 가우시안 피라미드 생성에 깊이 관여한다. 또한 가우시안 커널의 크기가  $s$ 에 의해  $\sigma$ ,  $k\sigma$ ,  $k^2\sigma$ , ... ( $k = 2^{1/s}$ )로 결정된다. 따라서  $s$ 가 증가하면 각 옥타브의 가우시안 영상 수는 선형적으로 증가하지만,  $k$ 는 감소하기 때문에 가우시안 영상을 생성하기 위한 커널의 크기는 감소한다. 연산량의 관점에서 영상의 증가는 선형적인 반면, 커널의 크기 감소는  $s+3$ 개의 영상 모든 화소에 영향을 주기 때문에 커널의 크기에 더 지배적이다.

$s$ 의 변화에 따른 연산량 변화의 실험결과를 그림 7(초기  $\sigma$ 의 변화에 대한 그래프에서  $s$ 의 변화를 매개변수로 사용)에 보이고 있다. Lowe 방식과 Hess의 방식 모두에서 보듯이  $s$ 가 증가하면 연산량이 감소하는 것을 볼 수 있으며, 감소율은 Hess 방식이 더 컸다.

$s$ 의 변화에 대해 추출된 국소극점 수와 특징점 수에 대한 실험도 진행하였는데, 그림 9에 그 결과를 보이고 있다. 이

실험은 그림 7과 동일한 영상들을 사용하였으며,  $\sigma$ 는 1.6으로, 옥타브 수는 6으로 하였다.  $s$ 의 증가는 국소극점을 추출하는 단위 DoG 영상군 자체가 증가하는 것이어서 국소극점과 특징점 모두 거의 선형적으로 증가함을 볼 수 있다.

그림 9.  $s$ 에 따른 추출된 국소극점 수와 특징점 수Fig. 9. Numbers of the extracted local extrema and key points for  $s$ 

## 3. 옥타브 수

가우시안 피라미드에서 특정 영상크기로 생성한 가우시안 영상들을 한 옥타브라고 부르며, 그 옥타브의 가우시안 영상 중 표준편차를  $k^s\sigma$ 로 필터링 한 영상의 가로와 세로를 각각 1/2로 줄인 영상을 대상으로 다음 옥타브의 가우시안 영상군을 생성한다. 여기서  $k^s\sigma$  영상을 선택하는 것은  $k = 2^{1/s}$ 이므로  $k^s\sigma = 2\sigma$ 로 필터링한 영상을 1/4로 축소함으로서 축소된 영상이  $\sigma$ 로 필터링된 결과로 간주하고 그 다음 과정을 진행한다. 따라서 주어진  $s$ 에서  $(i+1)$ 번째 옥타브를 계산하는 시간은  $i$ 번째 옥타브를 수행하는 시간의 약 1/4( $\sigma$ 로 필터링하는 과정이 빠지므로)이 소요된다.

옥타브가 증가하면 영상이 1/4씩 축소되므로 추출된 국소극점과 특징점도 1/4로 줄어들 것이라 예상할 수 있다. 그림 10에 옥타브별로 추출된 국소극점과 특징점 수를 보였는데, 이 실험 또한 그림 7과 동일한 영상들을 사용하였다. 그림 10(a)는 그림 8의 모든  $\sigma$ 와 그림 9의 모든  $s$ 에 대해 실험한 결과를 전체 국소극점, 특징점 중에 각각 옥타브별 비율로 나타낸 것이고, (b)는  $\sigma = 1.6$ ,  $s = 3$ 의 경우에 대해 실제 추출된 개수의 예인데, 두 그래프가 거의 동일한 경향성을 보이고 있다. 예측한 바와 같이, 추출된 국소극점과 특징점 모두 옥타브가 증가할수록 약 1/4씩 줄어드는 것(Lowe 방식의 경우 조금 많이, Hess의 방식은 조금 적게)을 확인할 수 있다.

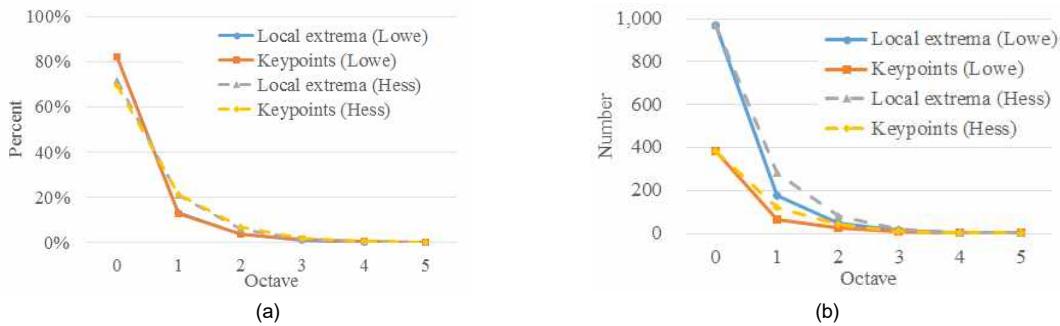


그림 10. 옥타브별 국소극점 및 특징점 수: (a) 비율, (b) 실제 개수의 예 ( $\sigma = 1.6$ ,  $s = 3$ )

Fig. 10. Number of local extrema and keypoints according to the octave; (a) ratios, (b) example of real numbers ( $\sigma = 1.6$ ,  $s = 3$ )

표 4.  $s$ 에 따른 호모그래피 특징점 선정 확률: (a) Lowe 방식, (b) Hess 방식

Table 4. Probability to be selected as the homography keypoints according to  $s$ ; (a) Lowe scheme, (b) Hess scheme

$\sigma$	$s$	DoG group to extract local extrema				
		1	2	3	4	5
1.0	1	X	X	X	X	X
	2	0.53	0.47	X	X	X
	3	0.32	0.38	0.3	X	X
	4	0.22	0.29	0.26	0.23	X
	5	0.11	0.18	0.36	0.27	0.08
1.2	1	X	X	X	X	X
	2	0.45	0.55	X	X	X
	3	0.24	0.51	0.25	X	X
	4	0.19	0.32	0.27	0.22	X
	5	0.1	0.31	0.22	0.19	0.18
1.4	1	X	X	X	X	X
	2	0.49	0.51	X	X	X
	3	0.29	0.39	0.32	X	X
	4	0.24	0.34	0.22	0.2	X
	5	0.18	0.26	0.23	0.16	0.17
1.6	1	X	X	X	X	X
	2	0.47	0.53	X	X	X
	3	0.31	0.43	0.26	X	X
	4	0.22	0.35	0.27	0.16	X
	5	0.08	0.25	0.23	0.28	0.16
1.8	1	X	X	X	X	X
	2	0.5	0.5	X	X	X
	3	0.37	0.38	0.25	X	X
	4	0.2	0.34	0.3	0.16	X
	5	0.16	0.23	0.33	0.17	0.11
2.0	1	X	X	X	X	X
	2	0.53	0.47	X	X	X
	3	0.24	0.42	0.34	X	X
	4	0.25	0.36	0.24	0.15	X
	5	0.15	0.29	0.25	0.19	0.12
Avg	1	X	X	X	X	X
	2	0.5	0.51	X	X	X
	3	0.3	0.42	0.29	X	X
	4	0.22	0.33	0.26	0.19	X
	5	0.13	0.25	0.27	0.21	0.14

(a)

$\sigma$	$s$	DoG group to extract local extrema				
		1	2	3	4	5
1.0	1	X	X	X	X	X
	2	0.63	0.37	X	X	X
	3	0.44	0.31	0.25	X	X
	4	0.37	0.37	0.18	0.08	X
	5	0.45	0.19	0.16	0.13	0.07
1.2	1	X	X	X	X	X
	2	0.57	0.43	X	X	X
	3	0.45	0.31	0.24	X	X
	4	0.32	0.29	0.25	0.14	X
	5	0.27	0.27	0.19	0.18	0.09
1.4	1	X	X	X	X	X
	2	0.6	0.4	X	X	X
	3	0.4	0.41	0.19	X	X
	4	0.31	0.21	0.25	0.23	X
	5	0.23	0.31	0.2	0.12	0.14
1.6	1	X	X	X	X	X
	2	0.65	0.35	X	X	X
	3	0.43	0.29	0.28	X	X
	4	0.29	0.3	0.27	0.14	X
	5	0.15	0.33	0.2	0.16	0.16
1.8	1	X	X	X	X	X
	2	0.6	0.4	X	X	X
	3	0.48	0.26	0.26	X	X
	4	0.34	0.3	0.28	0.08	X
	5	0.16	0.31	0.23	0.17	0.13
2.0	1	X	X	X	X	X
	2	0.69	0.31	X	X	X
	3	0.38	0.34	0.28	X	X
	4	0.51	0.21	0.13	0.15	X
	5	0.22	0.28	0.24	0.17	0.09
Avg	1	X	X	X	X	X
	2	0.62	0.38	X	X	X
	3	0.43	0.32	0.25	X	X
	4	0.36	0.28	0.23	0.14	X
	5	0.25	0.28	0.2	0.16	0.11

(b)

## IV. 스티칭 실험 및 분석

본 장에서는 앞 장에서 설명한 세 개의 가우시안 피라미드 파라미터가 스티칭에 미치는 영향을 실험적으로 분석한다.

표 5. 옥타브별 호모그래피 특징점 선정 확률; (a) Lowe 방식, (b) Hess 방식

Table 5. Probability to be selected as the homography keypoints for each octave; (a) Lowe scheme, (b) Hess scheme

$\sigma$	s	Octave					
		0	1	2	3	4	5
1.0	1	0.11	0	0	0	0	0
	2	0.13	0.05	0	0	0	0
	3	0.25	0.08	0.02	0.01	0	0
	4	0.18	0.07	0.01	0.01	0	0
	5	0.07	0.01	0	0	0	0
	Sum	0.74	0.21	0.03	0.02	0	0
1.2	1	0.1	0	0	0	0	0
	2	0.24	0.07	0	0	0	0
	3	0.17	0.04	0	0.01	0	0
	4	0.1	0.08	0	0.01	0	0
	5	0.14	0.03	0.01	0	0	0
	Sum	0.75	0.22	0.01	0.02	0	0
1.4	1	0.18	0	0	0	0	0
	2	0.18	0.08	0	0	0	0
	3	0.18	0.04	0.01	0	0	0
	4	0.12	0.04	0	0	0	0
	5	0.14	0.01	0.01	0.01	0	0
	Sum	0.8	0.17	0.02	0.01	0	0
1.6	1	0.08	0	0	0	0	0
	2	0.18	0.05	0.01	0.01	0	0
	3	0.19	0.03	0.01	0	0	0
	4	0.24	0.03	0.01	0	0	0
	5	0.11	0.03	0.01	0.01	0	0
	Sum	0.8	0.14	0.04	0.02	0	0
1.8	1	0.16	0	0	0	0	0
	2	0.21	0	0.02	0	0	0
	3	0.19	0.11	0.02	0.01	0	0
	4	0.15	0.02	0	0	0	0
	5	0.06	0.04	0	0.01	0	0
	Sum	0.77	0.17	0.04	0.02	0	0
2.0	1	0.15	0	0	0	0	0
	2	0.21	0.07	0.01	0	0	0
	3	0.15	0.08	0.01	0.01	0	0
	4	0.15	0.04	0	0	0	0
	5	0.08	0.04	0	0	0	0
	Sum	0.74	0.23	0.02	0.01	0	0
Avg	1	0.13	0	0	0	0	0
	2	0.192	0.053	0.007	0.002	0	0
	3	0.188	0.063	0.012	0.007	0	0
	4	0.157	0.047	0.003	0.003	0	0
	5	0.1	0.027	0.005	0.005	0	0
	Sum	0.767	0.19	0.027	0.017	0	0

(a)

### 1. 호모그래피 행렬 생성을 위한 네 특징점

그림 1에서 보인 바와 같이 특징점이 결정되면 각 특징점의 기술자를 만들고, 각 영상의 기술자들을 비교하여 호모그래피 행렬을 생성한다. 이 호모그래피에는 두 영상을 스

$\sigma$	s	Octave					
		0	1	2	3	4	5
1.0	1	0.38	0.07	0	0	0	0
	2	0.15	0.04	0	0	0	0
	3	0.09	0.07	0	0	0	0
	4	0.09	0.04	0	0	0	0
	5	0.05	0.02	0	0	0	0
	Sum	0.76	0.24	0	0	0	0
1.2	1	0.2	0.06	0.01	0	0	0
	2	0.23	0.04	0	0	0	0
	3	0.14	0.05	0	0	0	0
	4	0.16	0.02	0	0	0	0
	5	0.06	0.03	0	0	0	0
	Sum	0.79	0.2	0.01	0	0	0
1.4	1	0.18	0.05	0	0	0	0
	2	0.18	0.12	0.01	0	0	0
	3	0.1	0.08	0.02	0	0	0
	4	0.11	0.01	0	0	0	0
	5	0.07	0.07	0	0	0	0
	Sum	0.64	0.33	0.03	0	0	0
1.6	1	0.07	0.08	0	0	0	0
	2	0.23	0.09	0.01	0	0	0
	3	0.11	0.07	0.02	0	0	0
	4	0.13	0.03	0	0	0	0
	5	0.09	0.07	0	0	0	0
	Sum	0.63	0.34	0.03	0	0	0
1.8	1	0.12	0.04	0	0	0	0
	2	0.25	0.06	0	0	0	0
	3	0.14	0.08	0.01	0	0	0
	4	0.11	0.06	0	0	0	0
	5	0.12	0.01	0	0	0	0
	Sum	0.74	0.25	0.01	0	0	0
2.0	1	0.18	0.04	0	0	0	0
	2	0.17	0.1	0.01	0	0	0
	3	0.16	0.07	0.01	0	0	0
	4	0.14	0.03	0	0	0	0
	5	0.05	0.03	0.01	0	0	0
	Sum	0.7	0.27	0.03	0	0	0
Avg	1	0.188	0.057	0.002	0	0	0
	2	0.202	0.075	0.005	0	0	0
	3	0.123	0.07	0.01	0	0	0
	4	0.123	0.032	0	0	0	0
	5	0.073	0.038	0.002	0	0	0
	Sum	0.709	0.272	0.019	0	0	0

(b)

티칭할 때 스티칭되는 부분의 기하학적 변형을 정의한 것이다. 여기에는 4개의 특징점이 사용되는데, 이 4개의 특징점을 찾는 방법은 무작위로 4개의 특징점을 선택하여 식 1의 평균 거리오차  $d_{error}$ 를 계산하며, 일정 회수를 반복하여 이  $d_{error}$ 가 가장 작은 4개의 특징점(이를 호모그래피 특징점이라 칭함)을 최종 선정한다.

호모그래피 행렬은 스티칭 후 영상왜곡에 큰 영향을 미치므로 선정된 호모그래피 특징점들은 매우 중요하다. 따라서 먼저 특징점들 중 어떤 경우에 추출된 특징점이 호모그래피 특징점으로 선정되는지를 실험하였다. 이 실험은 그림 7과 동일한 영상들로 실험하였다. 각  $\sigma$ 에 대해 국소극점 추출 DoG 영상군 별 호모그래피 특징점을 추출할 확률을 계산하여 표 4에 나타내었다. 이 표는 그림 7과 동일한 영상들을 사용하였으며, 옥타브는 6개로 동일하게 하고 각  $\sigma$ 에 대해  $s$ 를 1부터 5까지 각각 실험한 것이다. 즉, 특정  $\sigma$ 에서  $s = m$ 의 경우는 국소극점 추출 DoG 영상군이  $m$  (1에서  $m$ 까지)개 존재한다. 따라서 표에서 ‘x’는 해당사항이 없음을 표시한다. 또한 표 맨 아래에 모든  $\sigma$ 에 대한 평균 확률(Avg)을 보였다. 표에서 보듯이 호모그래피 특징점으로 선정될 확률은  $s$ 에 대해서는 뚜렷한 경향성을 보이지 않는 것을 알 수 있다.

표 5는 동일한 영상에 대해  $s$ 를 5로 하여 각 옥타브별 호모그래피 특징점으로 선정될 확률을 계산한 것이다. 이 표에는 각  $\sigma$ 에 한 옥타브별 합계(Sum)와 전체 옥타브별 평균(Avg)을 같이 보였다. 두 방법 모두 모든  $\sigma$ 에 대해 옥타브 1이 2/3 이상의 확률을 보였고, 옥타브 2 이상에서 호모그래피 특징점이 추출될 확률은 매우 낮았다.

## 2. 스티칭

이상의 실험을 토대로 실제 영상을 스티칭하는 실험을 진행하였다. 스티칭 실험에 사용한 영상들을 표 6에 보이고 있는데,  $4,000 \times 3,000$  영상은 Gopro사의 Hero4 Black 카메라 6대를 장착한 F360 Explorer Mount로 자체 촬영한 것이고, 그 외의 영상은 인터넷을 통해 구하였으며<sup>[14-22]</sup>, 그 중 일부는 해상도를 조절하여 사용하였다.

4-1절의 표 5를 보면 2 이상의 옥타브는 거의 의미가 없으나, 스티칭 결과로 확인하기 위해 먼저 옥타브에 대한 실험을 수행하였다. 표 7에서 보였듯이 실험은 옥타브 0에서 옥타브 3(옥타브 0, 1, 2, 3을 모두 포함)까지 수행하였으며, 표의 숫자는 식 1의 평균거리오차  $d_{error}$ 이다. 표에서 보듯이 두 방식 모두 모든 옥타브에서 평균거리오차가 큰 차이



그림 11. 옥타브에 따른 스티칭 결과의 예. Lowe 방식; (a) 옥타브 0, (b) 옥타브 3, Hess 방식; (c) 옥타브 0, (d) 옥타브 3  
 Fig. 11. Example of stitching results according to octave. Lowe scheme; (a) octave 0, (b) octave 3, Hess scheme; (c) octave 0, (d) octave 3

를 보이지 않았으며, 그림 11에 보인 스티칭 결과 영상에서 도 차이를 거의 찾아볼 수 없었다. 따라서 이후의 실험은 모두 옥타브를 0(하나의 옥타브)으로 하였다.

표 6. 스티칭 실험에 사용한 영상들

Table 6. Images used in the stitching experiments

Resolution	Origin	# of sets
4,000×3,000	self	10
816×612	[14~17]	10
568×758	[18]	2
750×500	[14]	2
517×374	[3]	1
457×316	[19]	1

표 7. 옥타브에 대한 스티칭 실험 결과  $d_{error}$ Table 7. Resulting  $d_{error}$  from stitching experiments for octave

Octave	Lowe scheme	Hess scheme
0	0.840	0.834
0,1	0.862	0.834
0,1,2	0.884	0.875
0,1,2,3	0.879	0.860

다음은  $\sigma$ (초기 표준편차)와  $s$ 에 대한 스티칭 결과인데, 표 8에 표 7과 같이 각 경우의 평균거리오차  $d_{error}$  값을

표 8.  $\sigma$ 와  $s$ 에 대한 스티칭 실험 결과  $d_{error}$ ; (a) Lowe 방식, (b) Hess 방식Table 8. Resulting  $d_{error}$  from stitching experiments for  $\sigma$  and  $s$ ; (a) Lowe scheme, (b) Hess scheme

$\sigma$	$s$					
	1	2	3	4	5	Avg
1.0	0.711	0.808	0.902	0.862	0.866	0.830
1.2	0.752	0.848	0.816	0.796	0.805	0.803
1.4	0.792	0.855	0.778	0.817	0.819	0.812
1.6	0.800	0.849	0.878	0.737	0.776	0.808
1.8	0.827	0.892	0.828	0.817	0.809	0.834
2.0	0.820	0.896	0.862	0.833	0.795	0.841
Avg	0.784	0.858	0.844	0.810	0.812	

(a)

$\sigma$	$s$					
	1	2	3	4	5	Avg
1.0	0.799	0.904	0.941	0.911	0.902	0.891
1.2	0.840	0.956	0.843	0.863	0.839	0.868
1.4	0.795	0.841	0.862	0.908	0.888	0.859
1.6	0.885	0.811	0.860	0.817	0.859	0.846
1.8	0.980	1.022	0.910	0.869	0.854	0.927
2.0	1.005	0.865	0.930	0.887	0.867	0.910
Avg	0.884	0.900	0.891	0.876	0.868	

(b)

나열하였다.  $\sigma$ 는 1.0에서 2.0까지,  $s$ 는 1에서 5까지의 모든 경우에 대해 실험하였다. 먼저 Lowe 방식(a)과 Hess 방식(b)를 비교하면 Lowe 방식에서  $d_{error}$ 가 조금 작게 나타났으나 그 차이가 크지는 않았다. Lowe 방식과 Hess 방식 각각에서  $s$ 의 변화에 대한 결과는 각  $\sigma$ 에 대한 열의 값들이며,  $\sigma$ 의 변화에 대한 결과는 각  $s$ 에 대한 행의 값들이다. 표에서 보듯이  $\sigma$ 나  $s$ 에 대한 뚜렷한 경향성을 보이지 않으며, 이것은 맨 오른쪽 행과 맨 아래 열의 평균값(Avg)으로도 확인할 수 있다. 또한 표의 모든 값을 자체가 큰 차이를 보이지 않는 것을 알 수 있다.  $d_{error}$ 가 스티칭 성능의 절대적인 평가기준은 아니더라도 객관적인 자료임에는 틀림없으므로 이 값이 비슷하다는 것은 스티칭 성능이 비슷하다는 것으로 간주할 수 있다.

그림 12에 Lowe 방식과 Hess 방식으로 스티칭한 결과 영상의 예를 보이고 있는데, 가장 작은  $\sigma$ 와 가장 큰  $\sigma$ 에 대해 가장 작은  $s$ 와 가장 큰  $s$ 로 스티칭한 결과를 각각 보이고 있다. 이 그림들에서 보듯이 유판으로는 각 결과의 차이를 거의 찾을 수 없으며, 다른 영상들에서도 동일한 결과를 보였다.

3장과 4장의 실험결과를 비교하면, 스티칭 성능에는 별 차이가 없으면서 가장 연산량이 적은  $\sigma = 1.0$ ,  $s = 5$ , 옥타브 0의 경우와  $\sigma = 2.0$ ,  $s = 1$ , 옥타브 6의 가장 많은 경우, 그리고 가장 많이 사용하는  $\sigma = 1.6$ ,  $s = 3$ , 옥타브 3경우의 파라미터 값들과 연산량을 표 9에 보였는데, 연산량으로는 그림 7과 같이 연산에 필요한 MAC 수로 나타내었다. 앞에서도 언급한 바와 같이, Hess 방식이 Lowe 방식보다 연산량이 적음을 확인할 수 있다. 표에서 나타난 바와 같이, 최대 연산량이 필요한 파라미터 조합에 대비 최소 연산량의 파라미터 조합은 Lowe 방식의 경우 6.7% 정도, Hess 방식의 경우는 2.8%의 연산만 필요로 한다. Lowe 방식의 최대 연산량에 대한 Hess 방식의 최소 연산량은 2.1%에 불과하다.

표 9. 세 경우의 연산량 비교

Table 9. Comparison of the amount of calculation for the three cases

$\sigma$	$s$	Octave	Lowe scheme		Hess scheme	
			# of MACs	Ratio	# of MACs	Ratio
1.0	5	0	970,818,048	0.066	315,615,744	0.028
1.6	3	3	3,065,328,888	0.207	1,309,164,696	0.115
2.0	1	6	14,808,386,900	1.000	11,415,852,300	1.000



그림 12.  $\sigma$ 와  $s$ 에 대한 스티칭 실험 결과의 예. Lowe 방식; (a)  $\sigma = 1.0$ ,  $s = 1$ , (b)  $\sigma = 1.0$ ,  $s = 5$ , (c)  $\sigma = 2.0$ ,  $s = 1$ , (d)  $\sigma = 2.0$ ,  $s = 5$ , Hess 방식; (e)  $\sigma = 1.0$ ,  $s = 1$ , (f)  $\sigma = 1.0$ ,  $s = 5$ , (g)  $\sigma = 2.0$ ,  $s = 1$ , (h)  $\sigma = 2.0$ ,  $s = 5$

Fig. 12. Examples of stitching results for  $\sigma$  and  $s$ . Lowe scheme; (a)  $\sigma = 1.0$ ,  $s = 1$ , (b)  $\sigma = 1.0$ ,  $s = 5$ , (c)  $\sigma = 2.0$ ,  $s = 1$ , (d)  $\sigma = 2.0$ ,  $s = 5$ , Hess scheme; (e)  $\sigma = 1.0$ ,  $s = 1$ , (f)  $\sigma = 1.0$ ,  $s = 5$ , (g)  $\sigma = 2.0$ ,  $s = 1$ , (h)  $\sigma = 2.0$ ,  $s = 5$

## V. 결 론

본 논문에서는 VR영상 등을 생성할 때 필요한 영상의 스티칭 측면에서 SIFT 특징점을 추출하는데 관여하는 파라

미터들을 분석하고, 이들이 연산량(연산시간)과 스티칭 성능에 미치는 영향을 분석하였다. 여기에 고려한 파라미터들은 가우시안 커널의 초기 표준편차의 크기, 국소극점을 추출하기 위한 가우시안 차영상군의 수, 그리고 옥타브 수

이었다. 먼저 각 파라미터가 연산량에 미치는 영향을 분석하였고, 스티칭을 위한 호모그래피 행렬을 형성하는 네 특징점에 기여하는 정도와 스티칭에 각 파라미터가 얼마나 영향을 미치는지를 실제 스티칭 실험을 통해 분석하였다.

실험결과에 대한 분석 결과 먼저 연산량에 대해서는 가우시안 커널의 초기 표준편차의 크기의 증가는 기하급수적인 영향을 미치는 것으로 나타났고, 국소극점을 추출하기 위한 가우시안 차영상군의 수는 증가할수록 오히려 연산량을 감소시키는 것으로 확인되었으며, 옥타브 수의 증가는 이전 옥타브의 약 1/4씩만 연산량을 증가시켰다. 또한 옥타브 3 이상에서는 생성되는 특징점 수가 매우 적었으며, 호모그래피 행렬을 형성하는 특징점에도 거의 기여하지 못하였다. 또한 실제 스티칭 실험에서는 가우시안 커널의 초기 표준편차의 크기과 국소극점을 추출하기 위한 가우시안 차영상군의 수 모두 스티칭 성능에 거의 영향을 미치지 않는 것으로 나타났다.

결론적으로, 스티칭을 목적으로 하는 경우 가능하면 작은 초기 가우시안 커널을 사용하고 가우시안 차영상군의 수를 최대로 하며, 옥타브는 하나만 생성하여 연산시간을 최소로 하여도 스티칭 성능에는 거의 영향을 미치지 않는다는 것을 확인하였다. 또한 본 논문에서 분석한 내용들은 향후 영상 스티칭에 있어서 필요에 따라 성능저하 없이 연산시간을 단축하는 여러 가지 방법에서 많은 도움이 될 것으로 사료된다.

### 참 고 문 헌 (References)

- [1] Institute for Information & communications Technology Promotion, Technology Development Trend and Market Forecast of VR/AR, Weekly Technology Trends, Vol.1803, July, 2017.
- [2] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, Vol.60, No.2, pp.91-10, Jan. 2004, <https://doi.org/10.1023/B:VISI.0000029664.99615.94>.
- [3] M. Brown, and D. G. Lowe, "Automatic Panoramic Image Stitching using Invariant Features," International Journal of Computer Vision, Vol.74, No.1, pp.59-73, Aug. 2007, <https://doi.org/10.1007/s11263-006-0002-3>.
- [4] F. C. Huang, S. Y. Huang, J. W. Ker, and 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, doi: 10.1109/TCSVT.2011.2162760.
- [5] M. Grabner, H. Grabner, and H. Bischof, "Fast approximated SIFT," Asian Conference on Computer Vision, pp.918-927, 2006, [https://doi.org/10.1007/11612032\\_92](https://doi.org/10.1007/11612032_92).
- [6] Y. Ke, and R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," IEEE CVPR2004, Washington DC, USA, pp.506-513, July, 2004, doi: 10.1109/CVPR.2004.1315206.
- [7] H. Bay, T. Tuytelaars, and L. V. Gool "SURF: Speeded Up Robust Features," European Conference on Computer Vision, Graz, Austria, pp.404-417, 2006, [https://doi.org/10.1007/11744023\\_32](https://doi.org/10.1007/11744023_32).
- [8] R. Hess, "An Open-Source SIFT Library," ACM Multimedia, Firenze, Italy, pp.1493-1496, Oct. 2010.
- [9] J. Qiu, T. Huang, and T. Ikenaga, "A FPGA-based dual-pixel processing pipelined hardware accelerator for feature point detection in SIFT," 5th International Joint Conference on INC, IMS and IDC, Seoul, South Korea, pp.1668-1674, Nov. 2009, doi: 10.1109/NCM.2009.38.
- [10] H. D. Chati, F. Muhlauer, T. Braum, C. Bobda, and K. Berns, "Hardware/software co-design of a key point detector on FPGA," 15th Annual IEEE Symposium on Field-Programmable Custom Computing Machines, Napa, USA, pp.355-356, 2007, doi: 10.1109/FCCM.2007.61.
- [11] M. Lalonde, D. Bryns, L. Gargon, N. Teasdale, and D. Laurendeau, "Real-time eye blink detection with GPU-based SIFT tracking," 4th Canadian Conference on Computer and Robot Vision, Montreal, Canada, pp.481-487, 2007, doi: 10.1109/CRV.2007.54.
- [12] G. Hsu, C. Lin, and J. Wu, "Real-time 3-D object recognition using scale invariant feature transform and stereo vision," 4th International Conference on Autonomous Robots Agents, Wellington, New Zealand, pp.239-244, 2009, doi: 10.1109/ICARA.2009.4803919.
- [13] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.27, No.10, pp.1615-1630, Oct. 2004, doi: 10.1109/TPAMI.2005.188.
- [14] <http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/15463-f10/www/proj4/www/junjieli/>
- [15] [http://www.cs.cmu.edu/afs/andrew/scs/cs/15-463\\_f07/proj4/www/lisachan/](http://www.cs.cmu.edu/afs/andrew/scs/cs/15-463_f07/proj4/www/lisachan/)
- [16] <http://hugin.sourceforge.net/tutorials/two-photos/en.shtml>
- [17] <https://sites.google.com/a/umich.edu/eecs442-winter2015/home-work/image-stitching>
- [18] <http://www.cad.zju.edu.cn/home/gfzhang/training/Panorama/Panorama.htm>
- [19] <https://courses.engr.illinois.edu/cs498dwh/fa2010/lectures/Lecture%2020%20Photo%20Stitching.pdf>

---

### 저자 소개

---



#### 문 원 준

- 2018년 2월 : 광운대학교 전자재료공학과 공학사
- 2018년 3월 ~ 현재 : 광운대학교 전자재료공학과 공학석사
- ORCID : <https://orcid.org/0000-0002-9620-9524>
- 주관심분야 : Virtual Reality, 워터마킹, 2D 영상 신호처리



#### 서 영 호

- 1999년 2월 : 광운대학교 전자재료공학과 공학사
- 2001년 2월 : 광운대학교 일반대학원 공학석사
- 2004년 8월 : 광운대학교 일반대학원 공학박사
- 2005년 9월 ~ 2008년 2월 : 한성대학교 조교수
- 2008년 3월 ~ 현재 : 광운대학교 인제니어링학부대학 정교수
- ORCID : <http://orcid.org/0000-0003-1046-395X>
- 주관심분야: 실감미디어, 2D/3D 영상 신호처리, 디지털 홀로그램, SoC 설계



#### 김 동 육

- 1983년 2월 : 한양대학교 전자공학과 공학사
- 1985년 2월 : 한양대학교 공학석사
- 1991년 9월 : Georgia 공과대학 전기공학과 공학박사
- 1992년 3월 ~ 현재 : 광운대학교 전자재료공학과 정교수
- ORCID : <http://orcid.org/0000-0002-4668-743X>
- 주관심분야 : 3D 영상처리, 디지털 홀로그램, 디지털 VLSI Testability, VLSI CAD, DSP설계, Wireless Communication