

논문 2012-49SP-2-10

# 관심영역 기반 와핑을 이용한 3D 동영상 안정화 기법

(ROI-Based 3D Video Stabilization Using Warping)

이 태 환\*, 송 병 철\*\*

(Tae Hwan Lee and Byung Cheol Song)

## 요 약

가정용 캠코더의 보급으로 인해서 손떨림이나 카메라 흔들림을 보상하기 위해 다양한 동영상 안정화 기법들이 연구되고 있다. 동영상 안정화 기법은 초기에 2차원 움직임만을 고려하였지만, 최근에는 3차원 움직임까지 고려하여서 더 좋은 성능을 얻을 수 있게 되었다. 이러한 기법들 중 가장 좋은 성능을 보이는 것으로 알려진 기법이 바로 content preserving warping을 이용한 기법인데 이것은 뛰어난 성능을 보이지만 방대한 연산량이 단점이다. 그래서, 우리는 ROI 측면에서 종래 기술 대비 동등한 화질을 보이면서도 연산량이 적은 full frame warping을 제안한다. 먼저, 목표로 하는 깊이 정보를 바탕으로 관심 영역을 설정하고, 설정된 관심 영역을 기반으로 full frame warping을 수행한다.

## Abstract

As the portable camcorder becomes popular, various video stabilization algorithms for de-shaking of camera motion have been developed. In the past, most video stabilization algorithms were based on 2-dimensional camera motion, but recent algorithms show much better performance by considering 3-dimensional camera motion. Among the previous video stabilization algorithms, 3D video stabilization algorithm using content-preserving warps is known as the state-of-the-art owing to its superior performance. But, the major demerit of the algorithm is its high computational complexity. So, we present a computationally light full-frame warping algorithm based on ROI (region-of-interest) while providing comparable visual quality to the state-of-the-art in terms of ROI. First, a proper ROI with a target depth is chosen for each frame, and full-frame warping based on the selected ROI is applied.

**Keywords:** video stabilization, 3D motion, segmentation, content preserving warping, full frame warping

## I. 서 론

불과 십 년 전만 하더라도 디지털 카메라를 통해서 고화질의 영상을 얻기가 쉽지 않았으나, 최근에는 누구나 쉽게 고화질의 영상을 얻을 수 있게 되었다. 심지어 최근에는 캠코더뿐만 아니라 핸드폰과 같은 휴대용 기기로도 고화질의 영상을 얻을 수 있게 되었다. 이러한 기술의 발달로 인해서 편리함 측면에서는 크게 좋아졌지만 그와 함께 다른 문제들도 생기게 되었다. 그 중 하

나가 바로 카메라의 흔들림 문제이다. 최근에는 대부분의 카메라들이 한 손에 잡힐 수 있을 정도로 작아졌기 때문에 촬영 시 카메라를 잡고 있는 손의 조그만 흔들림에도 카메라가 심하게 흔들리게 된다. 이러한 결과로 영상의 흔들림이 심해지게 되고 심지어 촬영된 영상을 제대로 알아 볼 수 없을 정도로 흔들리는 경우도 발생하게 된다.

흔들림을 제거하기 위해서는 먼저 각 프레임의 움직임을 추정한 후, 원하는 카메라 경로에 맞춰서 프레임의 motion을 바꿔줘야 한다. 일반적으로 원하는 카메라 경로를 설정할 때는 시간축으로 저대역 필터 (low-pass filter)를 이용해서 안정화시켜주게 되는데 그 이유는 원래의 카메라 움직임에서 크게 벗어나지 않으면서 자연

\* 학생회원, \*\* 평생회원 인하대학교 전자공학부 (School of Electronic Engineering, Inha University)

※ 이 논문은 인하대학교의 지원에 의하여 연구되었으며, 삼성전자(주)의 지원에 의하여 수행되었음.

접수일자: 2011년6월21일, 수정완료일: 2011년11월30일

스러운 움직임이 나타낼 수 있기 때문이다. 만약 원래의 카메라 움직임에서 크게 벗어나게 된다면 영상에서의 왜곡이 발생할 가능성이 커지기 때문에 원하는 경로를 설정하는데 한계를 가지고 있다.

이러한 카메라의 경로를 추정하는데 있어서 초기에는 블록 정합을 이용한 방법들이 주로 연구되었다.<sup>[1~2]</sup> 하지만 블록 정합 방법은 가려진 영역 (occlusion)이 존재하거나 평탄한 영역에 대해서 잘못 찾을 확률이 크기 때문에 성능에 한계를 지닐 수밖에 없다. 그래서, 이러한 한계를 극복하기 위해서 특징점 정합 (feature point matching)을 이용한 방법이 연구되었는데 영상에서 신뢰할 수 있는 특징점들을 추출하여 정합하기 때문에 조금 더 정확한 결과를 낼 수 있게 되었다.<sup>[3~4]</sup> 그러나, 이러한 움직임 추정들은 기본적으로 2차원 움직임을 이용하기 때문에 실제 카메라의 3차원 움직임을 표현하는데 있어서는 여전히 한계가 있다. 그래서 최근에는 카메라의 3차원 움직임을 추정하여서 안정화시켜주는 기법들이 주로 연구되고 있다.<sup>[5~7]</sup>

최근 기법들 중 가장 뛰어난 성능을 보이는 것으로 Liu 등이 제안한 내용 보존 와핑 (content preserving warping)<sup>[5]</sup>을 이용한 안정화 기법이 있다. Liu의 알고리즘은 총 3단계로 이루어져 있는데, 첫 번째는 SfM (structure from motion)을 통한 움직임 추정 단계, 두 번째는 원하는 카메라 경로를 설정해 주는 단계, 그리고 마지막으로 경로에 맞춰서 영상을 와핑 (warping)시켜주는 단계로 나뉘지게 된다. 와핑 과정도 두 단계로 나뉘지게 되는데 먼저 homography 행렬을 이용한 전 프레임 (full frame) 와핑 과정을 거친 후에 이를 기반으로 내용 보존 와핑을 수행하게 된다. 내용 기반 와핑의 성능은 매우 뛰어나지만 연산량 문제 때문에 실시간 구현에 어려움이 있다.

본 논문에서는 실시간 구현을 위해 관심 영역 (region-of-interest; ROI)기반의 전 프레임 와핑을 이용하는 방법을 제안하고자 한다. 각 프레임 내 ROI를 설정하고 그 ROI 영역에 대해서만 특징점을 추출하여 와핑을 수행함으로써 시각적으로 상당히 안정된 영상이 얻을 수 있다.

## II. 제안 기법

### 1. 기존 방법

그림 1과 같이 논문 [5]에서 제안된 Liu의 방법은 총

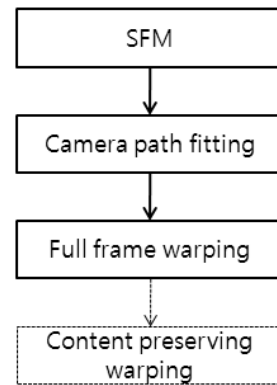


그림 1. 내용 보존 와핑 기반 안정화 기법의 순서도.  
Fig. 1. Flowchart of content preserving warping algorithm.

4 단계로 이루어져 있다. 본 절에서는 편의상 전 프레임 와핑(full frame warping) 단계까지만 소개한다.

#### 가. Structure from Motion (SfM)

카메라의 3차원 움직임을 측정하기 위해서 SfM 단계를 거치게 된다. 여기서 우리는 voodoo tracker라는 프로그램을 통해서 이 과정을 수행하게 되는데, 이것은 웹사이트를 통해서 무료로 구할 수 있는 대표적인 SfM 프로그램이다. SfM 단계를 거치게 되면 카메라의 외부 인수와 내부 인수를 얻게 되는데 이것을 통해서 카메라의 3차원 위치( $C_p$ ) 성분과 3차원 회전( $C_r$ ) 성분을 알 수 있게 된다. 이 외에도 각 프레임에 존재하는 특징점의 2차원 좌표와 3차원 위치 정보를 얻을 수 있게 된다.

#### 나. 카메라 경로 설정 (camera path fitting)

카메라 경로 설정 방법은 상호 작용을 통한 카메라 경로 설정 (interactive camera path creation), 저대역 필터링 (low-pass filtering), 경로 보정 (path fitting) 등 3개로 나뉠 수 있다.

첫 번째로 상호 작용을 통한 방법은 사용자가 카메라의 경로를 직접 설정하는 방식이다. 이 방법은 영상을 보고 사용자가 직접 판단해서 흔들림을 얼마나 제거할 것인지 결정하게 된다. 두 번째로 저대역 필터링 방법은 원래의 카메라 경로에 대해서 시간축으로 저대역 필터링해주는 방식이다. 필터링은 위치 성분인  $C_p(t)$ 와 회전 성분인  $C_r(t)$ 에 대해서 각각 해주게 되는데,  $C_p(t)$ 는 각 x, y, z성분에 대해서 전형적인 저대역 필터링을 해주면 된다.  $C_r(t)$ 는 비선형적인 값이기 때문에 논문 [13]에 있는 방법을 이용해서 필터링해주게 된다.

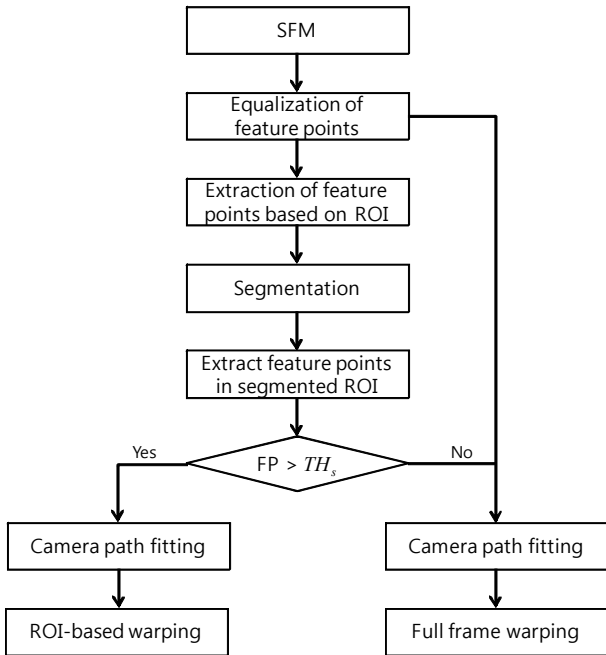


그림 2. 제안한 기법.

Fig. 2. The proposed algorithm.

이렇게 해서 필터링된  $C'_p(t)$ ,  $C'_r(t)$ 을 통해서 우리가 원하는 카메라 경로를 얻을 수 있게 된다. 마지막 세 번째 경로 보정 방법은 저대역 필터링 방법과 마찬가지로 위치 성분과 회전 성분을 각각 따로 처리하게 된다. 먼저,  $C_p(t)$ 의 경로는 상수, 선형, 비선형 이렇게 3가지로 설정할 수 있다. 상수 방식은 각  $x, y, z$  성분을 일정하게 설정해서 구하고 선형 및 비선형 방식은 최소자승법 (least square method)을 이용해서 구할 수 있다. 이렇게 해서 얻어진  $C'_p(t)$ 를 이용해서 새로운 특징점  $P'(t)$ 를 얻게 되는데, 이렇게 구해진 특징점 정보를 이용해서  $C'_r$ 를 구할 수 있게 된다.  $C'_r$ 는 식 (1)을 최소화함으로써 얻을 수 있다.

$$\sum_{1 \leq t \leq T} \sum_{p_k \in P(t)} \|p'_k - p_k\|_2^2 \quad (1)$$

식 (1)은 비선형 최소자승법 (non-linear least square minimization)을 이용해서 풀 수 있는데, 본 논문에서는 논문 [16]의 Levenberg-Marquardt 알고리즘을 채택했다.

#### 다. 전 프레임 와핑(Full Frame Warping)

카메라 경로 설정 과정을 통해서 새로운  $C'_p$ ,  $C'_r$ 을 구하게 되면 새로운 특징점  $P'$ 를 얻을 수 있게 된다. 이제  $P$ 와  $P'$ 와의 위치 관계를 통해서 호모그래피 행렬

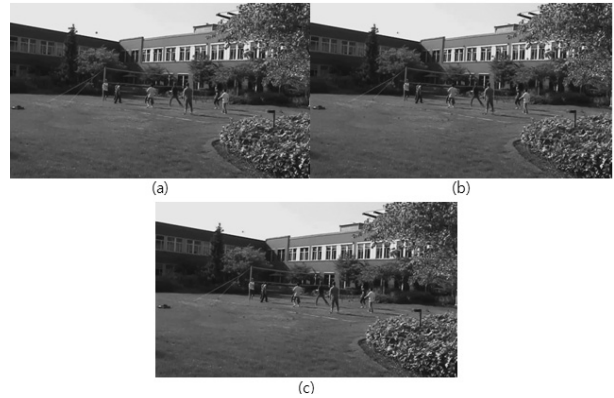


그림 3. (a) 입력 영상의 특징점, (b) 최대 개수를 늘렸을 때의 특징점, (c) 최소 간격을 설정했을 때의 특징점.

Fig. 3. (a) The feature points of an input frame, (b) the feature points after increasing the maximum number of feature points, (c) the feature points when the minimum distance between feature points is set.

(homography matrix)을 구할 수 있는데 이 때 최소자승법을 이용해서 구하게 된다. 일반적으로 움직임 모델은 투영 변환(perspective transformation)을 사용하게 되는데 이렇게 구해진 행렬을 통해서 영상을 와핑하는 것을 전 프레임 와핑이라고 한다. Liu의 방법은 이 과정 이후 내용 보존 와핑을 수행한다. 자세한 내용은 논문 [5]를 참조하면 된다.

## 2. 제안하는 기법

우리는 ROI를 찾아내고 그 ROI영역 내 특징점들을 이용하여 전 프레임 와핑 하는 방법을 제안한다. 제안하는 방법은 그림 2와 같은 순서로 진행된다. 먼저 SfM 과정을 통해 얻어진 특징점들의 분포를 평활화하는 작업을 수행하게 된다. 두 번째로, ROI 박스 추출 및 ROI 박스 내 특징점들을 추출한다. 이를 기반으로 ROI를 분할한다. 마지막으로 ROI 내 특징점 개수에 따라서 두 가지 모드를 결정하게 된다. 각 모드 별로 적절한 카메라 경로 설정을 거쳐 와핑이 수행된다.

### 가. ROI 설정

입력 영상의 특징점들은 SfM 과정을 통해서 선택되는데 여기서는 voodoo tracker를 통해 얻은 결과를 이용한다. 우리는 이 특징점들을 이용해서 ROI 영역을 설정하는 방법을 제안하고자 한다. 먼저, 입력 영상에 voodoo tracker를 적용해서 특징점들을 추출한다. 예를

들어 SfM을 통해 추출된 특징점들의 분포를 살펴보면, 그림 3(a)에서와 같이 주어진 영상의 오른쪽에 특징점들이 많이 모여 있는 반면에 왼쪽에는 특징점들이 상대적으로 적은 것을 확인할 수 있다. 그래서, 우리는 그림 3(b)와 같이 충분히 많은 수의 특징점들을 추출하고 식 (2)와 같이 특징점들 간 최소 간격을 설정하여서 고른 분포를 얻을 수 있도록 하였다.

$$P^{uni} = [p_k \in F(t) | Dx(p_k, p_s) \geq D_{max}, Dy(p_k, p_s) \geq D_{max}] \quad (2)$$

$$1 \leq s \leq k-1$$

여기서,  $Dx$ ,  $Dy$ 는 각각 x, y 방향의 거리를 나타내고,  $D_{max}$ 는 20으로 설정하였다. 그림 3(c)에서 보는 바와 같이 최소 간격을 설정함으로써 특징점들이 기존 방법과 개수는 비슷하면서 좀 더 고르게 분포되는 것을 확인할 수 있다. 이제  $P^{uni}$ 를 이용해서 ROI를 설정하게 되는데, 첫 번째로 특징점의 깊이 정보를 이용해서 특징점들을 걸러내고, 그 후에 걸러진 특징점들을 이용해서 ROI를 설정하게 된다. 먼저, 식 (3)을 통해 입력 영상의 중앙부에 위치한 특징점들을 선택한다. 식 (3)에서 X, Y는 각각 영상의 가로, 세로 크기를 나타낸다.

$$P^{center} = [p_k \in P^{uni}(t) | \frac{Y}{4} \leq p_k(x,y) \leq \frac{3Y}{4}, \frac{X}{4} \leq p_k(x,y) \leq \frac{3X}{4}] \quad (3)$$

여기서  $p_k$ 는 k번째 특징점을 의미한다. 이렇게 추출된  $P^{center}$ 를 사용하여 특징점들이 가장 밀집해 있는 깊이 영역을 식 (4)를 통해 구한다.

$$P^{ROI} = [p_k \in P^{center}(t) | depth(p_k) \geq Depth_{max}(t)] \quad (4)$$

여기서  $Depth_{max}$ 는 다음과 같이 구해진다.

$$Depth_{max}(t) = \max \sum_k \delta_k^z(t) \quad (5)$$

식 (5)에서

$$\delta_k^z(t) = \begin{cases} 1, & \text{if } depth(p_k) \geq z, depth(p_k) \leq z+100 \\ 0, & \text{otherwise} \end{cases} \quad (6)$$

여기서, z의 최소 단위는 10으로 설정하였다. 이러한 방식으로  $P^{ROI}$ 를 추출할 때  $Depth_{max}$ 보다 멀리 있는 특징점들만 추출하는 이유는 일반적으로 멀리 있는 물체일수록 프레임간의 움직임 차이가 적고 가까운 물체일수록 움직임 차이가 크기 때문에 멀리 있는 물체에서 울렁거

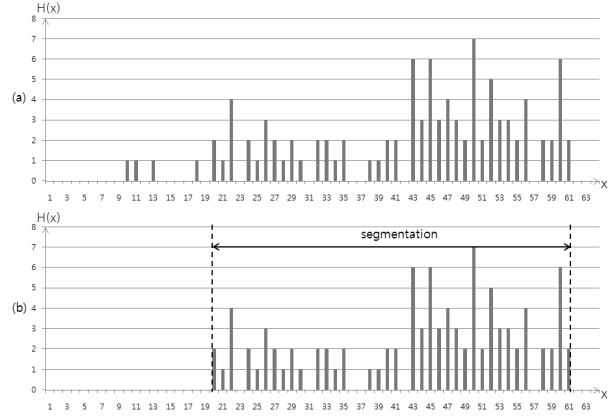


그림 4. (a) 특징점들의 히스토그램 (x방향), (b) 불필요한 특징점들을 제거하고 난 뒤의 히스토그램.  
Fig. 4. (a) The histogram of feature points (x axis), (b) the histogram after removing outlier feature points.

림이 눈에 띄는 경우가 발생하기 때문이다. 그래서, 우리는 이러한 영역을 ROI 영역으로 설정하고 이 영역에 있는 특징점들만을 가지고 전 프레임 와핑을 수행하게 된다. 실제로 거리가 가까운 물체에 존재하는 울렁거림은 우리 눈에 잘 보이지 않기 때문에 이러한 방식을 통해서 우리는 시각적으로 더 좋은 영상을 얻을 수 있게 된다.

다음 과정은 추출된 특징점 집합  $P^{ROI}$ 를 이용해서 최종적인 ROI를 설정하는 것이다. 이 때, 모든 과정은 격자 (grid) 단위로 이루어진다. 먼저, 관심 영역을 설정하기 전에 불필요한 특징점들을 제거하는 작업을 진행하게 된다. 이 과정은 특징점의 히스토그램 (histogram)을 이용해서 제거하게 되는데 먼저 x 방향에 대해서 실시하고 마찬가지로 y 방향에 대해서 실시하게 된다. 즉, 각 격자에 속해 있는 특징점의 개수를 측정한 후 그림 4(a)와 같이 x방향으로 히스토그램을 구하게 된다. 그런 다음 식 (7)을 통해 일차적으로 outlier 특징점들을 제거하고, 식 (8) 혹은 식 (9) 조건을 만족하는 특징점들을 이차적으로 제거한다.

$$H(x) = \begin{cases} 0, & \text{if } H(x-1) + H(x) + H(x+1) \leq TH_p \\ H(x), & \text{otherwise} \end{cases} \quad (7)$$

$$Px \leq TH, H(s) = 0, \quad x \leq s \leq x+6$$

$$Px = \sum_{k=0}^x H(k) \quad (8)$$

$$Px' \leq TH, H(s) = 0, \quad x-6 \leq s \leq x$$

$$Px' = \sum_{k=x}^{x_{end}} H(k) \quad (9)$$

식 (8)의 조건을 적용할 때는 첫 번째 격자에서부터 시작하면서 오른쪽 방향으로 진행하게 되고, 식 (9)의 조건을 적용할 때는 반대로 마지막 격자에서부터 시작하면서 왼쪽 방향으로 진행하게 된다. 여기서 문턱값  $TH_p$ 와  $TH$ 는 각 프레임의 특징점의 개수를 각각 100과 10으로 나눈 값을 의미하며,  $H(x)$ 는 같은 x축 격자 위치에 있는 특징점들의 히스토그램을 나타낸다. 이러한 과정을 거치고 나면 최종적인 ROI를 설정하게 되는데 그림 4(b)와 같이 0이 아닌 히스토그램 값을 모두 포함하여서 설정하게 된다. 마찬가지로 y방향에 대해서도 ROI구간을 설정한다.

#### 나. ROI 기반 전 프레임 와핑

다음 과정은 ROI 내 특징점들을 기반으로 와핑을 수행하는 것이다. 이 때, 잡음에 강인하게 하기 위해 본 논문에서는 식 (10)과 같이 20 프레임 이상 지속되는 특징점들만을 사용하게 된다.

$$P^{seg} = [p_k \in P^{ROI}(t) | p_k \in segmentation(t)] \quad (10)$$

이제  $P^{seg}$ 를 이용해서 카메라 경로 설정 과정을 거친 후에 전 프레임 와핑을 실시하게 된다.

일반적으로 한 프레임 내에 깊이 차이가 큰 영역이 존재하면 이러한 영역에서 서로 다른 움직임이 발생하기 쉽다. 이러한 경우에는 전 프레임 와핑을 통해서 하나의 움직임으로 나타내기가 쉽지 않기 때문에 우리가 원하는 만큼의 결과가 나오지 않을 수 있다. 반면, 제안한 방법에 따라  $P^{seg}$ 를 기반으로 와핑을 수행하게 되면 그림 5와 같이 관심 영역 측면에서 시각적으로 나온 영상을 얻을 수 있게 된다. 하지만 제안한 방법을 통해 얻은  $P^{seg}$ 의 개수가 부족하다면 전 프레임 와핑이 제대로 되지 않을 확률이 높아지게 된다. 그래서 이러한 경우



그림 5. 관심 영역에서의 특징점.  
Fig. 5. Feature points in a segmented ROI.

를 예방하기 위해  $P^{seg}$ 가 임계값 이하인 경우에는 제안한 방법을 사용하지 않고  $P^{uni}$ 를 이용하여서 전 프레임 와핑을 수행한다.

### III. 실험

제안한 기법의 성능을 검증하기 위해 그림 6과 같이 총 5개의 영상 시퀀스들에 대해서 실험을 진행하였다. 본 논문에서는 테스트영상 1, 2, 3, 4, 5라고 표현하였다. 이 영상들은 [5]에서 사용되었으며, <http://pages.cs.wisc.edu/~fliu/project/3dstab.htm>에서 내려받을 수 있다. 본 실험에서는 voodoo tracker의 KLT기반 SfM을 사용하였다.

그림 7은 제안 기법의 결과를 보인다. 입력 영상은 [5]에서 사용된 영상 중 건물 (ROI) 방향으로 걸으면서 심하게 카메라가 흔들린 시퀀스이다. 붉은 색의 절대 좌표를 기준으로 볼 때 건물의 흔들림이 완벽하게 보정됨을 알 수 있다. 그림 7의 입력 영상을 보면 건물의 창문 부분의 흔들림이 결과 영상에서 보면 사라진 것을 확인할 수 있다.

표 1에서는 정량적 비교 결과를 보인다. 먼저 각 특징점들의 움직임을 최소자승법을 이용해서 직선으로 표현한 후 이 직선과 특징점 움직임 간의 차이를 MSE (Mean Squared Error)로 구하였다. 본 논문에서는 카메라의 경로를 선형으로 가정하였기 때문에 이상적인 경우라면 특징점들의 움직임이 완벽한 직선이 될 것이다. 따라서 직선과의 차이가 적을수록 이상적인 움직임에



그림 6. 실험에 사용된 영상.  
Fig. 6. Test sequences.



그림 7. (상) 입력 영상 (하) 결과 영상.  
Fig. 7. (Top) input sequences (bottom) stabilized results.

표 1. 정량적인 비교. 수치는 ROI 영역 내 MSE값임.  
Table 1. Quantitative comparison of the results. The numbers indicate MSE values in ROI.

	기존 기법 [5]	제안 기법	ROI 내 특징점 개수
테스트 영상 1	1001	<b>927</b>	51개
테스트 영상 2	3791	<b>3606</b>	178개
테스트 영상 3	6167	<b>5770</b>	207개
테스트 영상 5	<b>4414</b>	4689	185개

표 2. 특징점의 개수 비교.  
Table 2. Comparison of the number of feature points.

	기존 방법 [5]에서의 특징점 수	$P^{uni}$	$P^{seg}$
테스트 영상 1	347	273	174
테스트 영상 2	501	350	238
테스트 영상 3	497	266	253
테스트 영상 4	359	146	N/A
테스트 영상 5	592	362	267

가깝다고 생각할 수 있다. 정확한 수치를 비교하기 위해 각 시퀀스의 첫 프레임부터 60번째 프레임까지  $P^{seg}$  중에서 지속되는 특징점들만 사용하였다. 표 1의 정량적 결과를 살펴보면 테스트 영상 1에서는 제안한 방법이 큰 효과를 발휘하는 것을 확인할 수 있다. 그 이유는 영상 1의 특징점들 간 깊이차가 크고 영상에서 폐색 영역이 존재하기 때문에 제안한 방법을 이용하게 되면 큰 효과를 발휘하는 것이다. 표 1의 결과를 통해서 제안한 방법이 ROI 내에서 더 좋은 결과를 보이지만 ROI를 제외한 영역은 MSE값이 비슷하거나 조금 더 높아지게 된다. 하지만 이러한 영역은 영상의 가장자리이거나 가까운 영역으로 비교적 빠르게 움직이면서 사라지기 때문에 우리 눈에 미치는 영향은 굉장히 적다. 한편, 표 2는 제안한 기법의 특징점 수가 기존 기법<sup>[5]</sup> 대비 훨씬 작음을 확인할 수 있다. 이는 제안한 기법이 연산량 측면에서 매우 유리함을 의미한다.

#### IV. 결 론

본 논문에서는 ROI 관점에서 와핑을 수행하는 방식의 동영상 안정화 기법을 제안하였다. 각 프레임 내 ROI를 설정하고 그 ROI 영역에 대해서만 특징점을 추출하여 와핑을 수행함으로써 시각적으로 상당히 안정된 영상이 얻을 수 있다. 모의실험 결과를 통해 제안한

기법이 종래 기법 대비 연산량을 대폭 줄이면서, ROI 영역에서의 화질은 유사하거나 오히려 나음을 확인하였다.

#### 참 고 문 헌

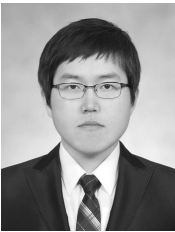
- [1] K. Ratakonda, "Real-time digital video stabilization for multi-media applications," *IEEE International Symposium on Circuits and Systems*, pp. 69 - 72, 1998.
- [2] Y. M. Yeh, H. C. Chiang, and S. J. Wang, "A digital camcorder image stabilizer based on gray coded bit-plane block matching," *IPPR Conf. Computer Vision, Graphics and Image Processing, Taipei, Taiwan*, pp.244-251, 2000.
- [3] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H. Y. Shum, "Full-frame video stabilization with motion inpainting," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1150 - 1163, 2006.
- [4] S. Battiato, G. Gallo, G. Puglisi, and S. Scellato, "SIFT features tracking for video stabilization", *ICIAP*, pp. 825-830, 2007.
- [5] F. Liu, M. Gleicher, H. Jin, and A. Agarwala, "Content-preserving warps for 3D video stabilization," *ACM SIGGRAPH, New York, NY, USA*, pp. 1 - 9, 2009.
- [6] J. M. Wang, H. P. Chou, S. W. Chen, and C. S. Fuh, "Video stabilization for a hand-held camera based on 3d motion model," *IEEE International Conference on Image Processing*, pp. 3477 - 3480, 2009.
- [7] G. Zhang, W. Hua, X. Qin, Y. Shao, and H. Bao. "Video stabilization based on a 3D perspective camera model," *The Visual Computer*.
- [8] M. Alexa, D. Cohen-Or, and D. Levin, "As-rigid-as-possible shape interpolation," *ACM SIGGRAPH, New Orleans, USA*, pp. 157-164, 2000.
- [9] T. Igarashi, T. Moscovich, and J. F. Hughes, "As-rigid-as-possible shape manipulation," *ACM Trans. Graph.*, pp.1134-1141, 2005.
- [10] R. Gal, O. Sorkine, and D. Cohen-Or, "Feature-aware texturing," *Eurographics Symposium on Rendering*, pp. 297-303, 2006.
- [11] S. Schaefer, T. McPhail, and J. Warren, "Image deformation using moving least squares," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 533 - 540, 2006.
- [12] S. Avidan et al., "Seam carving for content-aware image resizing," *ACM Trans.*

- Graphics*, vol. 26, no. 3, 2007.
- [13] J. Lee and S. Y. Shin, "General construction of time-domain filters for orientation data," *IEEE Trans. Visualization and Computer Graphics*, vol. 8, no. 2, pp. 119-128, 2002.
- [14] R. I. Hartley and A. Zisserman, "Multiple view geometry in computer vision," *Cambridge University Press*, 2000.
- [15] P. S. Heckbert, "Fundamentals of texture mapping and image warping", *Master thesis, Computer Science Division, UC Berkeley*, 1989.
- [16] J. More, "The Levenberg-Marquardt algorithm, implementation, and theory", *Springer-Verlag, numerical analysis edition*, 1977.
- [17] B. Triggs, P. McLauchlan, R. I. Hartley, and A. Fitzgibbon, "Bundle adjustment—a modern synthesis." *Vision Algorithms: Theory and Practice*, pp. 298 - 373, 1999.
- [18] T. Jebara, A. Azarbajejani, and A. Pentland, "3D structure from 2D motion", *IEEE Signal Processing Magazine*, vol. 16, no. 3, pp. 66-84, 1999.

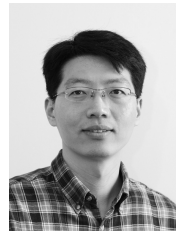
---

 저 자 소 개
 

---



이 태 환(학생회원)  
 2010년 인하대학교 전자공학과 학사 졸업.  
 2011년~현재 인하대학교 전자공학과 석사 과정.  
 <주관심분야 : 영상처리, 동영상 잡음제거, 비디오 안정화기법, 3차원 영상처리>



송 병 철(평생회원)  
 1994년 한국과학기술원 전기 및 전자공학과 졸업 (학사).  
 1996년 한국과학기술원 전기 및 전자공학과 졸업 (석사).  
 2001년 한국과학기술원 전기 및 전자공학과 졸업 (공학박사).  
 2001년~2008년 삼성전자 디지털미디어연구소 책임연구원  
 2008년~현재 인하대학교 전자공학부 교수  
 <주관심분야: 영상 신호처리, 영상시스템/SoC>