

논문 2011-48SP-4-6

동일 평면상의 자연 특징점 검출 및 추적을 이용한 증강현실 시스템

(Augmented Reality System using Planar Natural Feature Detection
and Its Tracking)

이 아 현*, 이 재 영*, 이 석 한**, 최 종 수***

(Ahyun Lee, Jae-Young Lee, Seok-Han Lee, and Jong-Soo Choi)

요 약

일반적으로 사용되는 마커 기반의 증강현실 시스템은 카메라 입력영상 내에 마커가 항상 존재해야 한다는 제한 때문에 사용자의 접근에 불편을 준다. 때문에 최근 배경 영상에서 취득할 수 있는 객체를 자연 마커로 생성한 시스템이나 배경 영상의 특징을 이용해 기하학적 지도를 작성하여 가상의 객체 정합에 이용한 증강현실 시스템들이 관심을 끌고 있다. 본 논문에서는 카메라 입력 영상에서 동일 평면상에 존재하는 특징들을 검출하고, 이를 추적함으로써 카메라 위치 정보를 추정하는 증강현실 시스템을 제안한다. 또한 특징점 추적에 사용된 추적 방법은 카메라에서 취득한 영상 밖으로 특징점이 벗어날 경우 더 이상 추적할 수 없는 문제점을 가지고 있어, 이를 보완하기 위해 새로운 특징점을 재검출하여 객체의 정합을 유지하는 방법도 제시한다. 제안된 방법은 미리 지정된 마커를 사용하지 않기 때문에 사용자의 접근이 편리하고, 특정한 형태의 마커를 사용하지 않는 다른 시스템보다 비교적 간단하게 구현할 수 있어 다양한 모바일 환경에서 유용하게 이용될 수 있다.

Abstract

Typically, vision-based AR systems operate on the basis of prior knowledge of the environment such as a square marker. The traditional marker-based AR system has a limitation that the marker has to be located in the sensing range. Therefore, there have been considerable research efforts for the techniques known as real-time camera tracking, in which the system attempts to add unknown 3D features to its feature map, and these then provide registration even when the reference map is out of the sensing range. In this paper, we describe a real-time camera tracking framework specifically designed to track a monocular camera in a desktop workspace. Basic idea of the proposed scheme is that a real-time camera tracking is achieved on the basis of a plane tracking algorithm. Also we suggest a method for re-detecting features to maintain registration of virtual objects. The proposed method can cope with the problem that the features cannot be tracked, when they go out of the sensing range. The main advantage of the proposed system are not only low computational cost but also convenient. It can be applicable to an augmented reality system for mobile computing environment.

Keywords : augmented reality, markerless AR, object tracking, corner detecting

I. 서 론

증강현실(AR: Augmented Reality)은 가상과 실사의

영상들이 정합(registration)되어 사용자에게 새로운 형태의 컴퓨팅 환경을 제공한다. 현실 세계를 가상 세계로 보완해주는 개념인 증강현실은 컴퓨터 그래픽으로 만들어진 가상 환경과 연계되어 구현된다. 하지만 기본적으로 현실 환경을 주축으로 구현되며, 컴퓨터 그래픽은 현실 환경에 필요한 정보를 추가로 제공하는 역할을 한다. 사용자가 보는 실사 영상과 3차원 가상 영상의 융합을 통해서 현실 환경과 가상의 컴퓨터 그래픽의 구

* 학생회원, ** 정회원, *** 평생회원-교신저자, 중앙대학교 첨단영상대학원 영상학과

(Graduate School of Advanced Imaging Science, Multimedia, and Film, Chung-Ang University)

※ 이 논문은 2단계 BK21 사업 지원과제임.

접수일자: 2010년11월11일, 수정완료일: 2011년2월23일

분이 모호해짐으로써 현실 환경의 확장을 가져다준다. 무엇보다 직관적 인터페이스(perceptual interface)와 유비쿼터스 컴퓨팅(ubiquitous computing) 등에 대한 응용 가능성으로 다양한 형태의 증강현실 기반 프로그램들이 연구되고 있다^[1~4]. 최근 스마트폰 기반으로 출시되는 다양한 프로그램들은 GPS(Global Positioning System)센서, 전자 나침반등을 사용하여 2차원 정보를 실사 영상에 정합하는 형태이다. 위와 같은 접근 방법은 간단하게 증강현실 시스템을 구성할 수 있다는 장점이 있지만, 3차원 객체를 정합을 위한 카메라 위치 정보를 계산할 수 없기 때문에 실질적인 의미인 현실 환경의 확장이라는 증강현실 시스템 구현에는 어려움이 있다. 이에 반해 별도의 다른 입력장치 없이 카메라를 이용하는 방법은 카메라에서 취득한 영상만을 이용하여 시스템을 구성한다. 카메라를 사용한 시스템은 영상으로부터의 3차원 기하학 정보 취득 기술을 기반으로 구현된다. 이를 위해서는 배경 영상의 2차원 공간과 실제 좌표계의 3차원 공간 사이의 사영 관계를 이용하여 카메라 위치 정보를 취득하기 위한 정확한 계산 과정이 필수적이다. 계산된 카메라 위치 정보를 통해 3차원 객체를 정합시킴으로써 실질적 의미의 증강현실 시스템을 구성할 수 있다.

일반적으로 카메라를 이용한 시스템은 취득한 영상에서 미리 지정된 특정한 형태의 마커(marker) 정보를 추출하고, 마커의 위치 정보를 통해서 카메라 위치 정보를 계산한다. 마커 기반의 시스템은 무엇보다 간단하게 증강현실 시스템을 구현할 수 있다는 장점이 있다. 하지만 카메라 취득 영상 내에 마커가 존재하지 않거나 마커의 일부가 가려짐 등으로 인해 마커 정보 추출이 불가능한 경우 시스템 구동이 되지 않는 단점을 가지고 있다. 또한 시스템 구동을 위해서는 특정한 형태의 마커가 항상 요구되어 사용자 접근이 불편하다는 단점을 가지고 있다. 이러한 이유로 최근 특정한 마커 형태를 사용하지 않고 배경 영상에서 취득할 수 있는 객체를 자연마커로 생성하여 마커로 사용한 시스템이나, 배경 영상에서 검출한 특징점을 통해 객체의 위치 인식 및 지도를 작성한 증강현실 시스템들이 관심을 받고 있다^[5~7]. 무엇보다 위와 같은 시스템들의 장점은 미리 지정된 특정한 형태의 마커를 사용하지 않기 때문에 사용자의 접근이 편리하여 보다 자연스러운 의미의 증강현실 시스템을 구성할 수 있다는 것이다.

본 논문에서는 위와 같은 연구들과 마찬가지로 마커

기반의 증강 현실 시스템의 문제점을 보완하기 위해서, 특정한 형태의 마커를 사용하지 않고 카메라에서 취득한 영상만을 가지고 특징을 검출하여 카메라의 위치 정보를 계산하는 방법을 제안한다. 제안하는 기법은 배경 영상 내 동일 평면상의 특징점을 검출하고, 광류(optical flow) 방법으로 특징점을 추적함으로써 수행된다. 최초 검출한 특징점과 이후 추적된 특징점의 관계를 계산하여 3차원 객체 정합에 필요한 카메라 위치 정보를 계산할 수 있다. 제안된 방법은 미리 지정된 마커를 사용하지 않기 때문에 사용자의 접근이 편리하고 특정한 형태의 마커를 사용하지 않는 다른 시스템보다 비교적 간단하게 구현할 수 있어 다양한 모바일 환경에서 유용하게 이용될 수 있을 것으로 예상된다.

본 논문의 구성은 다음과 같다. II장에서는 제안된 기하학 정보 추정 기법에 대해서 설명한다. 그리고 III장에서 제안된 기법에 대한 실험 결과를 보이고, IV장에서 결론 및 향후 연구 계획을 논한다.

II. 제안된 기법의 개요

본 논문에서 제안하는 기법에 대한 간략한 개념도를 그림 1에 보인다. 사용자는 모바일 기기를 사용하여 가상의 3차원 객체를 정합시킬 영역을 편의에 따라 설정하여 자연마커로 생성한다. 해당 영역에서 특징점 검출 및 추적을 이용하여 가상의 3차원 객체를 그림과 같이 정합할 수 있다. 제안된 방법의 수행과정은 그림 2에 보인다. 가상의 3차원 객체를 카메라로부터 취득된 영상에 정합하기위해, 먼저 취득된 영상으로부터 특징점을 검출한다. 검출된 2차원 특징점의 좌표를 3차원 공간상의 좌표로 변환하고, 이후 카메라에서 취득된 연속된 프레임마다 최초 프레임에서 검출된 2차원 좌표의 특징점을 추적한다. 최초 프레임에서 검출되어 변환한 특징점 좌표와 추적된 특징점 좌표 간의 상관관계를 이용하여 카메라 위치 정보를 계산하고, 이를 통해 3차원 객체의 정합을 한다. 특징점 검출 및 3차원 좌표로 변환은 시스템의 초기화 단계에 해당하고, 이것은 그림 3에서 보듯이 객체가 정합되는 위치를 결정하게 된다. 하지만 카메라 입력 영상이 2차원 정보만을 가지고 있기 때문에 객체의 3차원 관계를 정의할 수 없다. 따라서 초기화 단계는 해당 프레임에서 카메라가 배경 영상의 수직 방향에 위치함을 가정하고, 검출된 모든 특징점이 동일한 평면상에 위치한다는 전제하에 수행된다.



그림 1. 제안된 기법의 개념도
Fig. 1. Overview of the proposed technique.

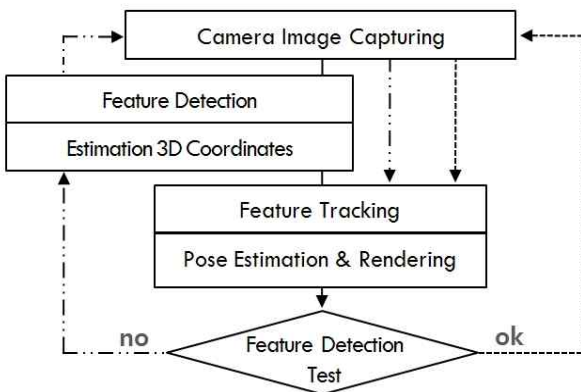


그림 2. 제안된 기법의 블록도
Fig. 2. A Block diagram of the proposed system.

이후 연속된 프레임에서 수행하면서 초기화 단계에서 검출한 특징점을 추적하고, 특징점의 추적 여부에 따라 재검출을 수행하여 객체의 정합을 유지한다.

1. 특징점 검출 및 추적

취득한 영상 내 검출한 특징은 매 프레임마다 동일한 배경 영상 내 특징으로 검출되어야 하기 때문에, 일반적으로 밝기 값의 변화가 급격하게 변화하여 큰 미분 계수를 갖는 코너점을 검출하는 것이 유리하다. 제안하는 알고리즘에서는 코너점과 같은 추적하기 좋은 특징점을 검출하기 위해 Shi-Tomasi^[8] 검출 방법을 사용한다. 일반적으로 가장 널리 사용되는 해리스 코너 검출 방법(Harris corner detection)은 영상 밝기 값의 2차 미분 값을 이용한다. 영상 내 각각의 화소에서 2차 미분으로 구성된 헤시안 행렬(hessian matrix)의 고유값(eigenvalue)이 모두 큰 값을 가질 경우 해당 화소를 코너라고 간주한다. 2차 미분 영상은 균일한 기울기(gradient)에서 값을 가지지 않기 때문에 코너 검출에 유용하다. 반면에 Shi-Tomasi 코너 검출 방법은 헤시

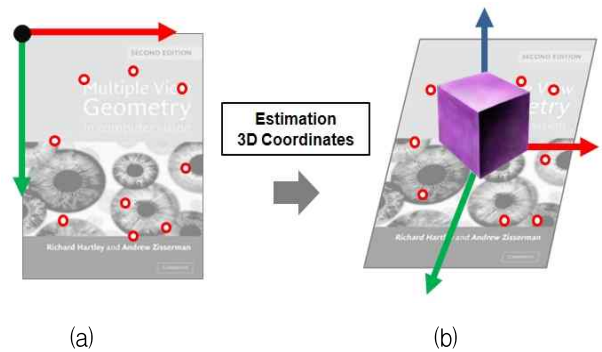


그림 3. 제안된 기법의 초기화 단계 (a) 2차원 특징점 검출 (b) (a)의 특징점을 3차원 좌표로 변환
Fig. 3. Initial process of the proposed method (a) 2D feature detection (b) transformation features from (a) to 3D coordinates.

안 행렬의 2개의 고유 값에서 작은 고유 값의 크기가 특정 임계치보다 큰 경우 해당 화소를 추적하기 좋은 대상으로 판별한다.

검출한 특징점의 기하학적인 측정치를 추출하기 위해서는 정수형 화소 좌표 대신에 실수형의 좌표를 사용해야 한다. 일반적으로 영상에서 뾰족한 정점(peak)을 찾을 때, 이 점이 화소의 정중앙에 위치하는 경우는 드물다. 이를 위해 서브 픽셀 코너 검출방법(sub-pixel corner detection)^[9]을 사용한다. 포물선과 같은 곡선을 이용한 적합(fitting)을 수행함으로써 화소들 사이에 존재하는 정점의 좌표를 찾는다.

검출한 특징점을 추적하는 방법은 LKT(Lukas-Kanade Tracker)^[10]을 사용한다. 이 방법은 미리 지정된 화소들 위에 씌워진 작은 윈도우로부터 얻은 지역 정보만을 이용하고, 추적할 점을 미리 지정하여 수행하는 희소 광류 방식(sparse optical flow)이다. LKT는 밝기 항상성(brightness constancy), 시간 지속성(temporal persistence), 공간 일관성(spatial coherence)의 3가지 가정에 기초를 둔다. 밝기 항상성은 연속된 특정 화소들의 밝기 값이 다른 시간의 프레임에도 일정하다는 가정이다. 따라서 식 (1)에서 시간축 t에 대한 편미분 값이 0이다. 이를 통해 동일한 밝기 값을 가진 영역을 추적함으로써 연속된 프레임 간의 속도의 양을 계산할 수 있다. I_x, I_y, I_t 는 각각 x, y, 시간축의 편미분 결과이고, u, v는 x, y축의 좌표 변화량을 의미한다. 공간 일관성은 시간에 따라 이동 변화량을 갖는 특정 화소의 주변 화소들이 일관된 좌표의 변화량을 가질 것이라는 가정이다. 식 (1)에서 하나의 함수로 두 개의 변수

(u, v) 를 계산할 수 없기에 근방 25개의 화소들이 동일한 변화량을 갖는다는 가정으로 식 (2)과 같이 변수 u, v 의 값을 최소자승법을 통해 계산할 수 있다.

$$\begin{aligned} \frac{\partial I}{\partial x} \left(\frac{\partial x}{\partial t} \right) + \frac{\partial I}{\partial y} \left(\frac{\partial y}{\partial t} \right) + \frac{\partial I}{\partial t} &= 0 \\ \frac{\partial I}{\partial x} = I_x, \frac{\partial I}{\partial y} = I_y, \frac{\partial I}{\partial t} = I_t, \frac{\partial x}{\partial t} = u, \frac{\partial y}{\partial t} = v \end{aligned} \quad (1)$$

$$\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ I_x(p_2) & I_y(p_2) \\ \dots & \dots \\ I_x(p_{25}) & I_y(p_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix} \quad (2)$$

LKT는 시간 변화량에 비해서 좌표 변화량이 크지 않다는 시간 지속성을 가정하여 작은 지역 윈도우 영역을 사용하기 때문에 연산속도가 빠르다. 하지만 작은 지역 윈도우 크기보다 큰 움직임이 발생하였을 경우에는 움직임을 계산하지 못하는 단점이 있다. 이러한 문제점을 해결하기 위해 가우시안 영상 피라미드(gaussian image pyramid)를 이용한 방법이 사용된다. 본 영상으로부터 가우시안 영상 피라미드를 구성하고, 크기가 작은 계층에서 추적을 시작하여 점점 크기가 큰 계층으로 갈수록 추적 변화량을 누적하여, 제한된 크기의 윈도우를 사용하여도 급격한 좌표의 변화량을 검출할 수 있다^[11~12].

2. 카메라 위치 정보 계산

3차원 객체 정합은 2차원 좌표 값에 대한 3차원 공간으로의 역사영(back-projection)을 통하여 계산된다. 제안된 시스템은 취득한 영상에서 검출한 특징점의 2차원 좌표를 3차원 공간의 좌표로 변환하는 초기화 단계가 선행된다. 카메라 입력영상에서 검출한 좌표 값은 2차원이기에, 이를 3차원 공간으로 변환할 때 깊이 값(z축)이 모두 0이라는 가정을 한다. 만약 검출된 특징 화소들이 동일 평면에 위치하지 않을 경우 오차가 생길 수 있다는 문제점을 갖는다. 이러한 이유 때문에 제안하는 방법에서는 책상 위와 같이 비교적 평평한 배경영상에서 수행된다는 전제가 필요하다. 이후 연속된 프레임에서 취득한 특징점의 2차원 좌표와 초기화 단계에서 취득한 3차원 공간의 좌표의 관계를 통해서 카메라의 위치 정보를 계산한다. 그림 4와 같이 3차원 좌표인 $\mathbf{M} = (X, Y, Z)$ 와 2차원 영상의 좌표인 $\mathbf{m} = (x, y)$ 의 동차 좌표계를 $\tilde{\mathbf{M}} = [X Y Z 1]^T$ 과 $\tilde{\mathbf{m}} = [x y Z 1]^T$ 이라고 할

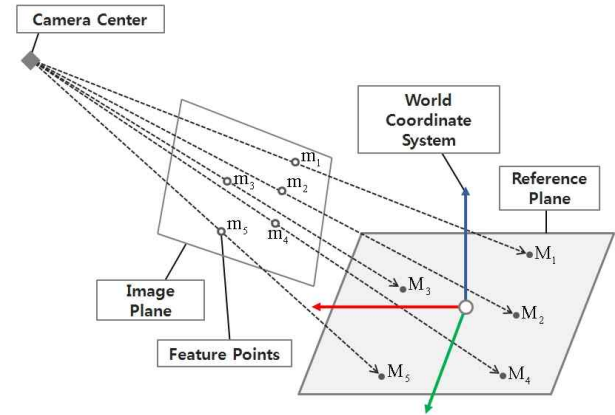


그림 4. 배경과 카메라 영상 사이의 기하학적 관계

Fig. 4. Geometric relationship between features and an image taken by camera.

때, 이들 사이의 사영관계는 3×4 카메라 행렬 $\tilde{\mathbf{P}}$ 에 의해서 식 (3)과 같이 정의 된다.

$$\tilde{\mathbf{m}} = \lambda \tilde{\mathbf{P}} \tilde{\mathbf{M}} = \lambda \mathbf{K} [\mathbf{R} | \mathbf{t}] \tilde{\mathbf{M}} = \lambda \mathbf{K} [\mathbf{r}_1 \mathbf{r}_2 \mathbf{r}_3 | \mathbf{t}] \tilde{\mathbf{M}} \quad (3)$$

λ 는 사영 행렬 $\tilde{\mathbf{P}}$ 에 대한 스케일 변수이며, \mathbf{R} 은 카메라의 회전 변위에 의한 3×3 행렬이다. 또한 \mathbf{r} 은 행렬 \mathbf{R} 의 i 번째 열(column)을 나타내며, $\mathbf{t} = [t_x \ t_y \ t_z]^T$ 는 카메라의 이동을 의미하는 3×1 이동 벡터(translation vector)이다. 또한 3×3 행렬 \mathbf{K} 는 카메라의 내부 파라미터(intrinsic parameter)를 원소로 갖는 카메라 교정 행렬을 나타내는 정칙 행렬(non-singular matrix)이며, 식 (4)로 정의된다. 식 (4)의 행렬에서 f_x , f_y 는 영상의 각 좌표 축 방향으로의 스케일 값을 의미하며, s 는 영상에 대한 비틀림 파라미터(skew parameter)이다. 그리고 (x_0, y_0) 는 영상의 주점(principal point)을 나타낸다. 식 (4)의 카메라 행렬을 구하기 위해서는 일반적으로 별도의 카메라 교정이 필요하다.

$$\mathbf{K} = \begin{bmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (4)$$

본 논문에서는 Zhang^[13]이 제안한 카메라 교정 기법을 이용하여 카메라 행렬을 구한다. Zhang의 카메라 교정 기법은 무한 평면상의 절대점(absolute points)이 갖는 특징 중 하나인 등거리 변환(isometric transformation)에 대한 불변성(invariance)을 이용하여 영상으로 사영되는 IAC(Image of Absolute Conic) ω 를 계산한 다음, $\omega^{-1} = \mathbf{K} \mathbf{K}^T$ 의 관계로부터 카메라의 내부 파라미터 행렬을 구한다. 따라서 Zhang의 기법의 구현을

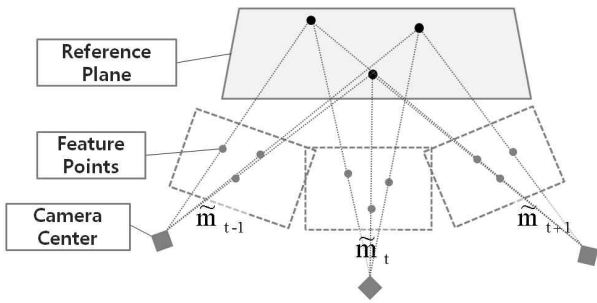


그림 5. 연속된 배경 영상 사이의 사영관계
Fig. 5. Projective geometry of input image sequences.

위해서는 서로 다른 방향 및 위치를 갖는 동일 평면에 대한 3장 이상의 영상이 필요하다^[14~16].

식 (5)과 같이 매 프레임마다 \tilde{M} 과 \tilde{m} 의 값을 입력하여 카메라 매트릭스 \tilde{P} 를 계산한다. 여기서 \tilde{P} 는 미리 지정된 카메라 내부 파라미터 K 행렬을 제외한 $[R | t]$ 행렬을 구함으로써 연산된다. \tilde{m} 은 매 프레임 카메라에서 취득한 연속된 2차원 영상에서 추적되는 특징점의 좌표 값이고, \tilde{M} 은 초기화 단계에서 \tilde{m} 을 검출과 동시에 3차원 공간상으로 변환한 좌표 값이다. \tilde{M} 을 구하기 위해서는 먼저 식 (6)과 같이 \tilde{S} 를 계산한다. 좌표계의 원점이 영상의 좌측 상단에 위치하는 2차원 좌표계와 달리 3차원 좌표계는 영상의 중심에 좌표계의 원점이 존재하고, y축의 증가방향은 반대이다. 초기화 단계에서의 특징점 검출 영역을 이미지 전체 크기의 1/2로 설정하였다. 따라서 g_w 는 특징점 검출 영역의 가로 크기의 1/2이고, g_h 는 검출 영역의 세로 크기의 1/2이다. \tilde{S} 를 통해 식 (7)처럼 3차원 좌표계로 변환한 \tilde{M} 을 구한다. \tilde{M} 은 모든 연속된 프레임에서 구하는 것이 아니라 초기화 단계에서만 수행되고, 이후 그림 5과 같이 이후 연속된 프레임에서 변화하는 \tilde{m} 을 통해서 카메라 행렬 \tilde{P} 를 계산할 수 있다. s_{nx} 과 s_{ny} 는 각각 \tilde{S} 행렬의 1행과 2행의 인자를 의미한다.

$$\begin{bmatrix} m_{1x} & m_{2x} & \dots & m_{nx} \\ m_{1y} & m_{2y} & \dots & m_{ny} \\ 1 & 1 & \dots & 1 \end{bmatrix} = [R | t] \begin{bmatrix} M_{1x} & M_{2x} & \dots & M_{nx} \\ M_{1y} & M_{2y} & \dots & M_{ny} \\ 0 & 0 & \dots & 0 \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (5)$$

$$\tilde{S} = \begin{bmatrix} m_{1x} - g_w & m_{2x} - g_w & \dots & m_{nx} - g_w \\ -(m_{1y} - g_h) & -(m_{2y} - g_h) & \dots & -(m_{ny} - g_h) \end{bmatrix} \quad (6)$$

$$\tilde{M} = \begin{bmatrix} s_{1x} & s_{2x} & \dots & s_{nx} \\ s_{1y} & s_{2y} & \dots & s_{ny} \\ 0 & 0 & \dots & 0 \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (7)$$

3. 특징점 재검출

특징점 추적에 사용된 LKT는 그림 6과 같이 추적된 특징점이 취득된 영상 프레임 밖으로 특징점이 벗어날 경우 더 이상 추적할 수 없다는 단점이 있다. 이러한 문제점을 보완하기 위해 그림 7과 같이 추가로 특징점을 검출하고, 이를 카메라 위치 정보 계산에 사용한다. 재검출된 2차원 특징점은 초기화 단계에서 사용된 특징점의 2차원 좌표가 3차원 좌표로 변환되는 방법과 유사하게 3차원 좌표로 변환한다. 하지만 초기화 단계에서 수

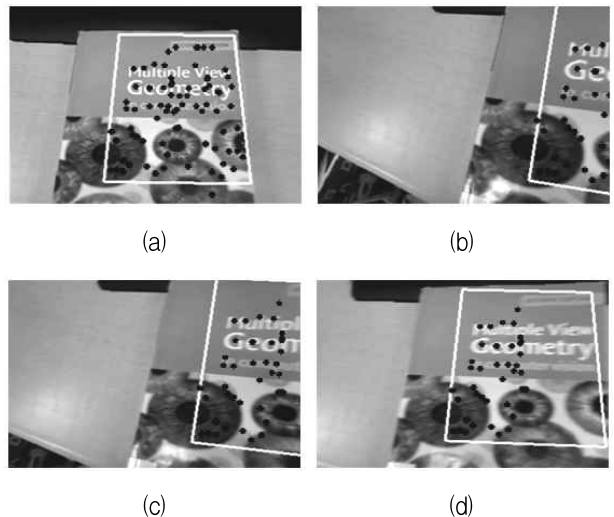


그림 6. 기존 방법에서 추적된 특징점이 취득된 영상 프레임을 벗어날 경우 (a) 특징점 검출 (b) 카메라 이동으로 인한 특징점 소실 (c)(d) 소실된 특징점

Fig. 6. When tracked features go out of the sensing range in traditional system (a) camera movements (b) lost features because of camera movements (c)(d) lost features.

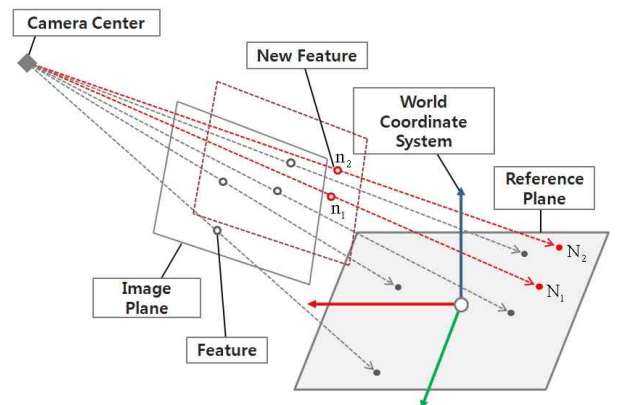


그림 7. 재검출된 배경의 특징점과 카메라 영상의 기하학적 관계

Fig. 7. Geometric relationship between re-detected features and input image.

행했던 변환 방법과는 조금은 다르다. 검출된 특징점이 동일 평면상에 놓여있다고 가정하고 수행했던 초기화 단계와 달리, 이후 연속된 프레임에서는 재검출된 특징점의 3차원 좌표의 깊이 값을 0이라고 가정할 수 없다. 따라서 먼저 재검출을 수행한 프레임에서 이미 추적하고 있던 특징점인 2차원 좌표 $\tilde{\mathbf{m}}$ 와 변환한 3차원 좌표 $\tilde{\mathbf{M}}$ 간의 관계인 $\tilde{\mathbf{H}}$ 행렬을 식 (8)과 같이 구한다. $\tilde{\mathbf{H}}$ 행렬은 2차원 공간상의 사영변화행렬(perspective transformation matrix)로써, 검출된 특징점이 동일 평면상이라는 가정 때문에 이와 같은 관계가 성립 된다. (8)에서 구한 $\tilde{\mathbf{H}}$ 행렬을 식 (9)에 적용함으로써, 새로 검출된 2차원 특징점 좌표 $\tilde{\mathbf{n}}$ 의 3차원 좌표 $\tilde{\mathbf{N}}$ 을 얻을 수 있다.

$$\tilde{\mathbf{H}} \begin{bmatrix} m_{1x} & m_{2x} & \dots & m_{nx} \\ m_{1y} & m_{2y} & \dots & m_{ny} \\ 1 & 1 & \dots & 1 \end{bmatrix} = \begin{bmatrix} M_{1x} & M_{2x} & \dots & M_{nx} \\ M_{1y} & M_{2y} & \dots & M_{ny} \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (8)$$

$$\tilde{\mathbf{H}} \begin{bmatrix} n_{1x} & n_{2x} & \dots & n_{nx} \\ n_{1y} & n_{2y} & \dots & n_{ny} \\ 1 & 1 & \dots & 1 \end{bmatrix} = \begin{bmatrix} N_{1x} & N_{2x} & \dots & N_{nx} \\ N_{1y} & N_{2y} & \dots & N_{ny} \\ 1 & 1 & \dots & 1 \end{bmatrix} \quad (9)$$

III. 실험 결과

모든 실험은 Intel(R) Core(TM)2 Quad CPU 2.66GHz, 4GB이고, 320×240 해상도와 초당 30프레임으로 작동되는 USB 카메라를 이용하여 수행되었다. 카메라에서 취득한 영상에서 자연 마커로 생성할 영역을 설정하기 위해 그림 8(a)에서 보듯이 흰색의 가이드라인을 통해 해당 영역의 특징점만을 검출한다. 검출된 동일 평면상의 특징점을 3차원 공간상의 좌표로 변환할 때, 해당 좌표의 깊이 값(z축)을 0으로 지정하면 특징점이 검출된 평면은 카메라가 바라보는 방향에서 수직으로 위치한다. 따라서 가상의 3차원 객체는 배경 영상에 수직방향으로 정합된다. 그림 8(b)의 흰색 선은 가이드라인이고, 점은 검출된 특징점이다. 최초 프레임에서 검출한 특징점을 이후 연속된 프레임에서 추적한 결과는 그림 8(c)에서 보인다. 점은 LKT를 통해 추적된 특징점들의 결과이고, 흰색 선은 추적된 특징점 좌표 결과에 따라 최초 특징점이 검출된 프레임에서의 가이드라인이 변화된 결과이다. 그림 9에서는 각기 다른 평면에서 자연마커를 생성하여 3차원 객체를 정합시킨 결과를 확인할 수 있다. 그림 9(a), (b)와 같이 책 표면과 같은

평면의 객체를 사용하거나, 그림 9(c), (d)와 같은 비교적 평평한 배경 영상을 사용하여도 안정적인 객체 정합 결과를 확인할 수 있다. 그림 10(a)와 같이 검출한 특징점이 카메라 이미지 영역을 벗어날 경우, 앞서 설명한 것처럼 LKT는 해당 특징점을 더 이상 추적할 수 없다. 이러한 문제점을 해결하기 위해 새로운 특징점을 재검출하여 객체 정합을 유지할 수 있다. 그림 10(b)는 재검출 방법을 사용하지 않은 객체 정합 결과이다. 소실된 특징점으로 인해 불안정한 객체 정합 결과를 확인할 수 있다. 그림 10(c)는 제안된 재검출 방법을 사용한 객체 정합 결과이다. 기존의 방법보다 안정적으로 3차원 객체 정합이 이루어지는 것을 볼 수 있다.

제안된 방법의 성능 평가를 위해 특정한 형태의 마커 기반인 시스템인 ARToolKit^[17]과 카메라 위치 정보를 비교하였다. 그림 11은 그래프의 가로축인 프레임 변화에 따른 ARToolKit의 카메라 위치 추정 방법과 제안된 카메라 위치 추정 방법의 결과를 x, y, z 값으로 각각 비교한 수치를 보인다. 마커 기반의 안정적인 시스템으로 잘 알려진 ARToolKit과 비교하여 제안된 시스템의 카메라 위치 추정 결과가 유사한 성능을 갖는 것으로 확인되었다.

제안된 시스템의 수행속도는 표 1에서 보인다. 수행속도 측정은 300프레임을 가지고 각각 과정의 측정 결과 평균을 내었다. 특징점 검출과 특징점 추적의 수행속도는 사용되는 특징점 개수에 따라 민감하게 영향을 받기 때문에, 특정 개수에 해당되는 수행 속도로 측정하였다. 초기화 단계에서 수행되는 특징점 검출 시간은 최초 한번만 수행되고, 이를 제외한 이후 연속된 프레임에서 수행되는 다른 과정인 특징점 추적, 카메라 위치 정보 연산과 렌더링 수행 시간은 10ms이 넘지 않는다. 이는 초당 30프레임 이상의 실시간 시스템에 이용될 수 있음을 보여준다.

표 1. 시스템 수행 속도

Table 1. System processing speed(ms).

구분	초기화 단계	매 프레임 수행되는 과정		
	특징점 검출	특징점 추적	카메라 위치 정보 연산	렌더링
ms	77.21 (71 points)	6.59 (71 points)	0.65	1.02

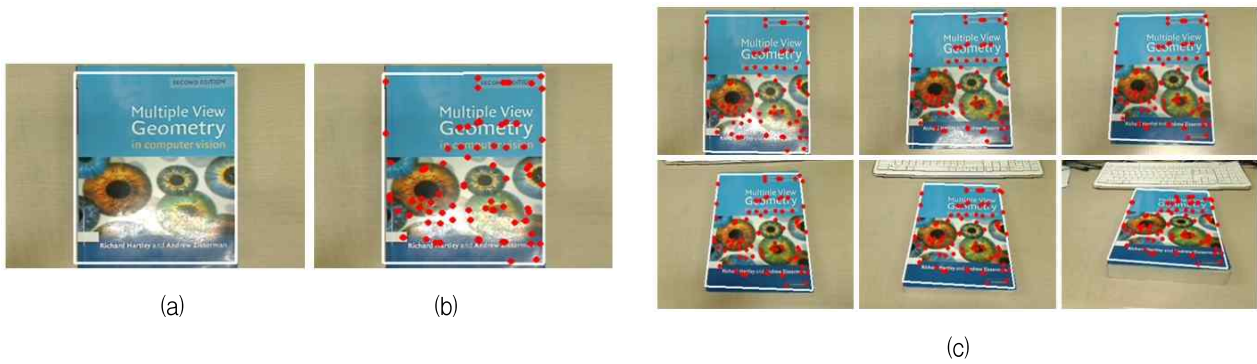


그림 8. 특징점 검출 및 추적
 (a) 검출 영역 (b) 특징점 검출 (c) 특징점 추적
 Fig. 8. Results of feature detection and tracking.
 (a) detection area (b) feature detection (c) feature tracking

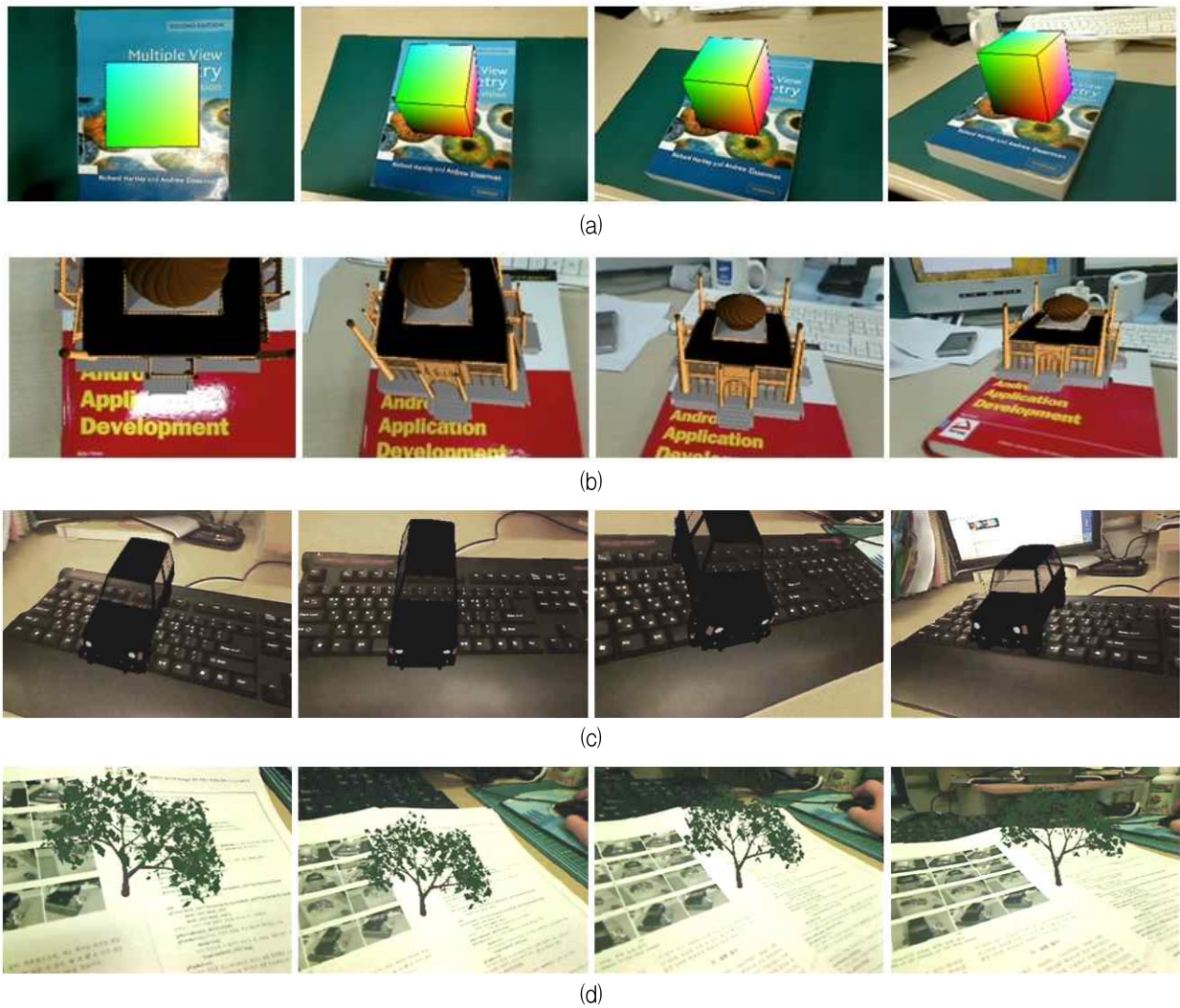


그림 9. (a)~(d) 각각 다른 평면에서의 3차원 객체 정합 결과
 Fig. 9. (a)~(d) Rendering results of 3D objects on the other plane.

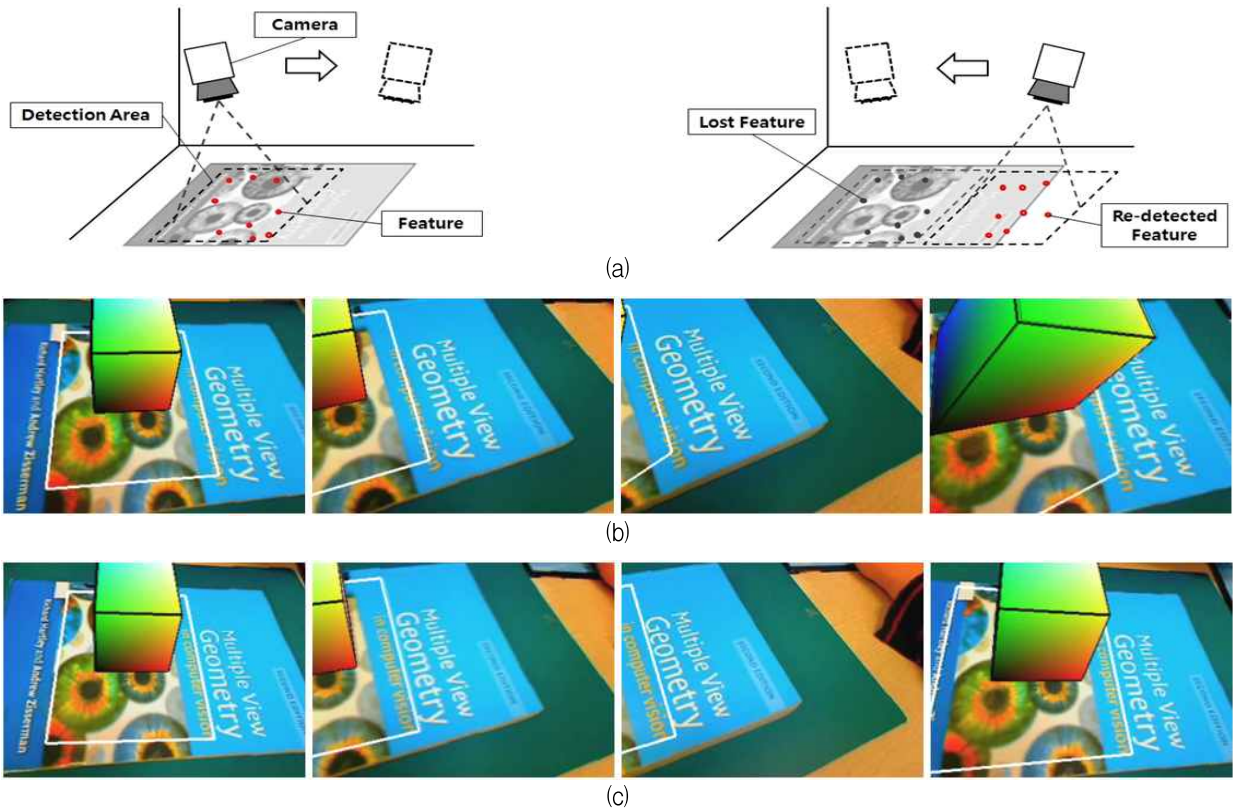


그림 10. 특징점 재검출을 통한 객체 정합 (a) 카메라 이동 및 특징점 검출 (b) 제안된 방법을 사용하지 않은 객체 정합 결과 (c) 제안된 방법의 객체 정합 결과

Fig. 10. Rendering results using with features re-detection (a) camera movements and feature detection (b) rendering results of the method which does not use the proposed method (c) rendering results of using the proposed method.

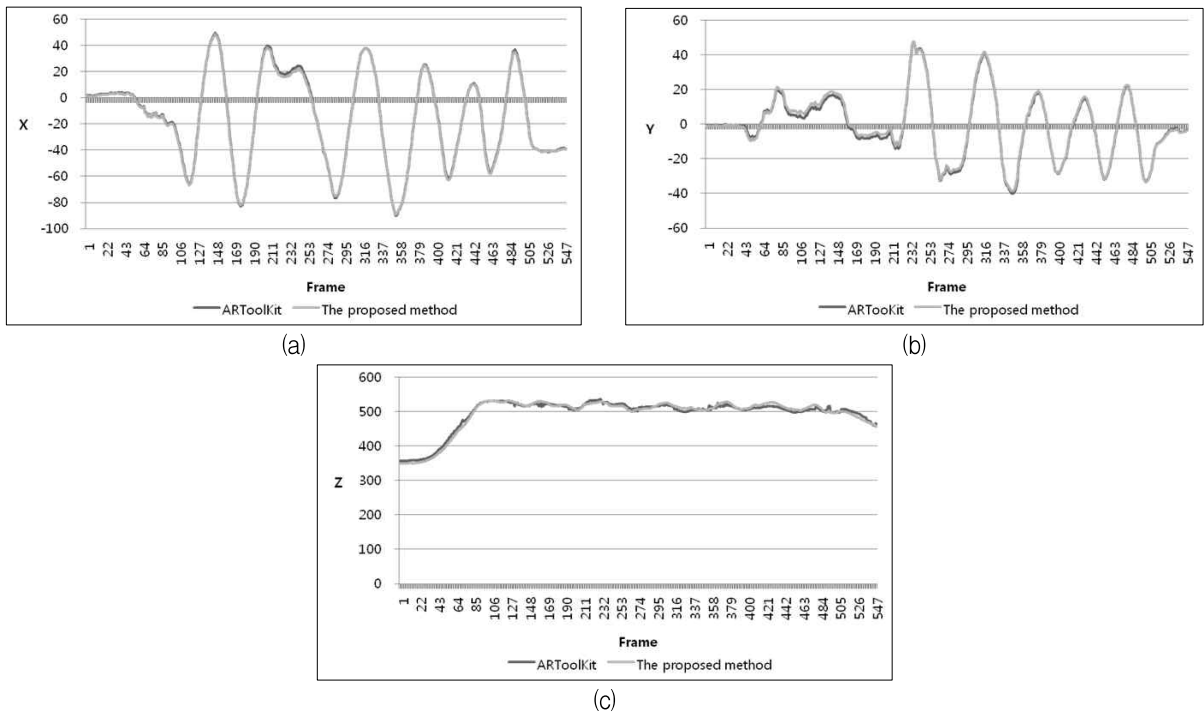


그림 11. ARToolKit과 제안된 방법의 카메라 위치 추정 결과 비교 (a) tx (b) ty (c) tz

Fig. 11. Comparisons of camera pose estimation results in the ARToolKit and the proposed method (a) tx (b) ty (c) tz.

IV. 결론 및 향후 연구 계획

본 논문에서는 미리 지정된 특정한 형태의 마커를 사용하지 않고도, 비교적 간단히 구현할 수 있는 증강현실 시스템을 제안하였다. 이를 위해 카메라에서 취득된 영상에서 특징점을 검출하고 이를 3차원 공간상의 좌표로 변화하였다.

이후 연속된 프레임에서 추적된 특징점의 2차원 좌표와 초기화 프레임에서 얻은 3차원 좌표간의 관계를 통해서 가상의 3차원 객체 정합을 수행하였다. 또한 추적된 특징점을 더 이상 추적할 수 없을 경우 특징점 재검출을 수행하여, 이를 통해 가상의 3차원 객체 정합을 유지하였다. 제안된 기법은 기존의 특정한 형태의 마커 기반 증강현실 시스템보다 사용자의 접근이 편리하다는 장점을 가지고 있다. 향후 진행될 연구에서는 배경영상에서 검출된 특징점이 모두 동일 평면에 위치해 있다는 가정을 벗어나, 초기화 단계에서 동일 평면상의 특징점만을 분류하는 방법과 SIFT(Scale Invariant Feature Transform)^[18]와 같은 강건한 특징점 검출 방법이 포함될 예정이다^[19].

참 고 문 헌

- [1] S.H. Lee, S.K. Lee and J.S. Choi, "Real-time camera tracking using a particle filter and multiple feature trackers," Games Innovations Conference, 2009. ICE-GIC 2009. International IEEE Consumer Electronics Society's, pp. 29-36, Aug. 2009.
- [2] Peter Keitler, "Mobile Augmented Reality based 3D Snapshots," in Proc. IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 199-200, 2009.
- [3] R. Azuma, Y. Baillet, R. Behringer, S. Feiner, S. Julier and B. Macintyre, "Recent Advances in Augmented Reality," IEEE Computer Graphic and Applications, pp. 34-47, 2001.
- [4] 박형민, 이재영, 이석한, 최중수, "반 깜박임 선택을 이용한 응시 상호작용과 착용형 AR 시스템," 전자공학회논문지, 제46권, 제5호, 91-100 쪽, 2009년 9월.
- [5] A. Davison, I. Reid, N. Morton and O. Stasse, "Monoslam: Real-time single camera slam," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 29, no.6, pp. 1052-1067, Jun. 2007.
- [6] G. Klein and D. Murray, "Parallel tracking and mapping for small AR workspaces," in Proc. IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 225-234, Sep. 2007.
- [7] D. Wanger, T. Langlotz and D. Schmalstieg, "Robust and Unobtrusive Marker Tracking on Mobile Phones," in Proc. IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 225-234, Sep. 2008.
- [8] J. Shi and C. Tomasi, "Good features to track," IEEE Proc. Conf. Computer Vision and Pattern Recognition, pp. 293-600, Jun. 1994.
- [9] Dazhi Chen, Guangjun Zhang, "A New Sub-Pixel Detector for X-Corners in Camera Calibration Targets," International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, 2005.
- [10] J.-Y. Bouguet, "Pyramidal Implementation of the Lucas Kanade Feature Tracker," Intel Corporation, Microprocessor Research Labs, 2000.
- [11] J. Barron, N. Thacker, "Computing 2D and 3D Optical Flow," Tina-Vision, 2005.
- [12] G. Bradski, A. Kaehler, "Learning OpenCV: Computer Vision with the OpenCV Library," O'REILLY, 2008.
- [13] Z. Zhang, "Flexible New Technique for Camera Calibration," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 19, no. 7, pp. 1330-1334, Nov. 2000.
- [14] 이석한, 최중수, "단일 카메라를 이용한 보행자의 높이 및 위치 추정 기법." 전자공학회논문지, 제45권, 제3호, 20-31쪽, 2008년 8월.
- [15] R. Hartley and A. Zisserman, "Multiple View Geometry in Computer Vision," Cambridge Univ. Press, 2003.
- [16] O. Faugeras, "Three-Dimensional Computer Vision," The MIT Press, 1993.
- [17] H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, K. Tachibana, "Virtual Object Manipulation on a Table-Top AR Environment," Augmented Reality, 2000. (ISAR 2000). in Proc. IEEE and ACM International Symposium on, pp. 111-119, 2000.
- [18] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91-110, 2004.
- [19] T. Lee and T. Höllerer, "Multithreaded Hybrid Feature Tracking for Markerless Augmented

Reality,” IEEE Trans. Visualization and Computer Graphics, vol. 15, no. 3, pp.355-368, May/June. 2009.

— 저 자 소 개 —



이 아 현(학생회원)
2008년 광운대학교 미디어 영상학부 학사 졸업.
2009년~현재 중앙대학교 첨단영상대학원 영상학과 석사과정.

<주관심분야 : 증강현실, 컴퓨터 비전>



이 재 영(학생회원)
2005년 세명대학교 전기공학과 학사 졸업.
2007년 중앙대학교 첨단영상대학원 영상학과 석사 졸업.
2007년~현재 중앙대학교 첨단영상대학원 영상학과 박사과정.

<주관심분야 : 영상처리, 증강현실, 컴퓨터 비전>



이 석 한(정회원)
1999년 중앙대학교 전자공학과 학사 졸업.
2001년 중앙대학교 첨단영상대학원 영상학과 석사 졸업.
2001년~2004년 LG전자 주임연구원.

2009년 중앙대학교 첨단영상대학원 영상학과 박사 졸업.

2009년~2010년 중앙대학교 첨단영상대학원 박사 후 과정.

2010년~현재 중앙대학교 첨단영상대학원 BK21 연구 교수

<주관심분야 : 실시간 카메라 추적, 증강현실, 3차원 컴퓨터 비전>



최 종 수(평생회원)-교신저자
1975년 인하대학교 전기공학과 학사 졸업.
1977년 서울대학교 대학원 전자공학과 석사 졸업.
1981년 Keio University(일본) 전기공학과 박사 졸업.

1981년~1981년 Aloka Co. Ltd 연구소 연구원.

1981년~1999년 중앙대학교 전자공학과 교수.

1999년~현재 중앙대학교 첨단영상대학원 영상학과 교수.

<주관심분야 : 영상신호처리, 컴퓨터 비전, 증강현실, 영상합성, 영상통신>