

## 스마트폰 카메라를 이용한 증강현실 시스템

이관욱 장성은 김만배  
 강원대학교 IT대학 컴퓨터정보통신공학과  
 lkw1340@kangwon.ac.kr

### Augmented Reality System Using Smart Phone Camera

Lee, Kwan-Wook Seung Eun Jang and Kim, Man-Bae  
 Dept. of Computer and Communications Engineering  
 IT College, Kangwon National University

#### 요약

최근 스마트폰의 급속한 성장에 이어 보급률 또한 급증하고 있다. 또한 아이폰 및 안드로이드 등 스마트폰 시대를 이끌어가는 대표적인 OS마다 다양한 API를 공개하고 있어 더욱 질 좋고 다양한 스마트폰 어플리케이션이 연구되어지고 있다. 본 논문에서는 스마트폰의 모노 카메라를 이용하여 입체영상을 제작하고, 여기에 증강현실 콘텐츠를 합성하는 스마트폰 입체 증강현실 시스템을 제안한다.

#### 1. 서론

최근 애플, 구글 등을 앞세워 스마트폰이 큰 이슈가 된 이후, 현재까지 스마트폰 사용자 수는 폭발적으로 증가하고 있다. 이에 따라 스마트폰의 어플리케이션에 대한 연구도 크게 각광받고 있다. 또한 스마트폰의 카메라 프레임에 대한 접근이 용이해지게 되면서 다양하고 실용적인 스마트폰 증강현실 시스템이 연구되어졌다. 증강현실은 가상현실의 한 분야로, 현실세계에 가상의 정보를 합성하는 것을 말한다. 대부분의 경우 위치기반서비스(Local Based Service)를 이용한 위치 정보나 사물에 대한 정보를 제공, 혹은 게임 등의 분야에서 많이 연구되어지고 있다[1][2][3]. 본 논문에서는 이러한 추세에 맞춰, 스마트폰 입체 증강현실 시스템을 제안한다. 먼저 스마트폰의 카메라를 수평으로 이동하며 N장의 영상을 획득한다. 다음으로 획득한 영상 간의 주각(Principal Orientation)의 차이 및 이동벡터(Translation Vector)를 구한 후, 이를 이용하여 입체영상으로 제작하기에 가장 적합한 두 장의 영상을 선택한다. 선택된 두 장으로 입체영상을 제작한 후, CG 콘텐츠와 합성하여 최종적으로 스마트폰 입체 증강현실 시스템을 구축한다.

본 논문의 구성은 다음과 같다. 2절에서는 제안하는 시스템의 구조에 대하여 상세히 설명한다. 3절에서는 입체 콘텐츠의 제작 방법에 대하여 설명하고, 4절에서는 CG 콘텐츠의 제작 방법에 대하여 설명한다. 5절에서는 실험 결과를 보여주고, 6절에서 결론을 맺는다.

#### 2. 제안하는 시스템의 구조

그림 1은 실험에 사용된 IOS의 구조를 나타낸다. 최하위에 하드웨어 및 드라이버 등 커널 레벨의 라이브러리를 제공하는 Core OS 계층이 있고, 그 상위 계층인 Core Service 계층에서는 저수준 혹은 소켓 등의 라이브러리 서비스를 제공한다. 세 번째 Media 계층에서는 비디오, 오디오, OpenGL 등의 멀티미디어 라이브러리를 제공하며, 최상위인 Cocoa Touch 계층에서는 실질적으로 개발자들이 개발할 수 있는 환경을 제공해준다. 이 계층에서는 배열이나 문자열 등 로우레벨의 데이터와 관련된 클래스를 제공하는 Foundation Framework, 각종 컨트롤 및 화면에 대한 구성을 담당하는 UIKit Framework 등 두 가지의 프레임워크를 제공한다.

크를 제공하는데, 개발자들은 이 라이브러리를 가져다 사용함으로써 어느 정도 로우 레벨로의 접근이 가능해지고 또한 전체적인 시스템 및 UI를 구축할 수 있게 된다.

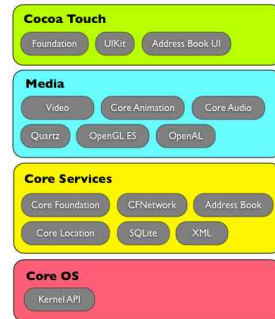


그림 1. IOS 시스템 구조

다음의 그림 2는 입체 증강현실 시스템의 전체 흐름을 보여준다.

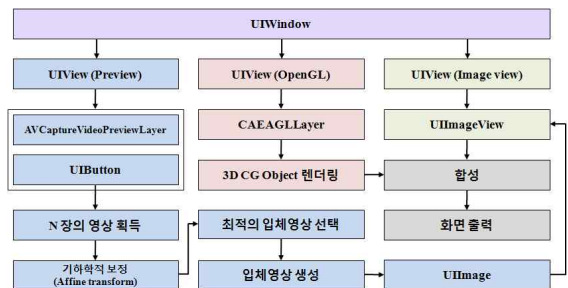


그림 2. 시스템의 전체 흐름도

첫 번째 UIWebView는 카메라의 프리뷰를 담당한다. 카메라의 프레임을 획득하기 위해, AVCaptureVideoDataOutputSampleBufferDelegate를 사용한다. 여기에 스마트폰 카메라의 프리뷰를 보여주기 위한 레이어인 AVCaptureVideoPreviewLayer, 그리고 프레임의 획득이 끝난 후 최적의 입체영상 선택 알고리즘을 실행하기 위한 UIButton이 추가된다. 두 번째 UIWebView에서는 OpenGL ES를 이용하여 CG 콘텐츠를 화면에 렌더링하는 역할을 한다. UIWebView는 제작된 입체영상을 화면에 보여준다. 버

튼 터치에서부터 입체영상을 생성하는 역할은 첫 번째 UIView에서 모두 처리되어지는데, 그 결과로 제작된 UIImage를 전송받아 최종적인 출력을 하게 된다. 이 동작은 UIImageView를 생성하여 추가한 후, 그 위에서 이루어진다. 여기에 또한 CG 콘텐츠가 합성되어, 입체 증강현실 콘텐츠를 시청할 수 있게 된다.

### 3. 입체 콘텐츠 제작

입체 콘텐츠를 제작하기 위해, 스마트폰의 카메라로 촬영한 N장의 영상을 이용하여 입체영상으로 제작하기에 가장 적합한 두 장의 영상을 선택하는 방법을 사용한다. 먼저 최적의 입체영상을 선택하기 위한 두 가지의 변수를 구한다. 이를 위해 에지의 성분을 구해야 하는데, Sobel 연산자 등을 이용하게 되면 복잡성이 증가하여 스마트폰 상에 적용하기에 적합하지 않다. 따라서 다음의 식 (1)을 이용하여, 픽셀 주변의 상하 및 좌우의 차이를 이용하여 에지를 계산한다.

$$\begin{aligned} E_{hor} &= P(i, j-1) - P(i, j+1) \\ E_{ver} &= P(i-1, j) - P(i+1, j) \end{aligned} \quad (1)$$

다음으로 수평 및 수직 에지 성분을 이용하여 다음의 식 (2)와 같이 에지의 방향성을 구한다.

$$\theta = \tan^{-1} \left( \frac{E_{ver}}{E_{hor}} \right) \quad (2)$$

각 에지 픽셀의 방향성을 구한 후, 히스토그램을 이용하여 누적값을 계산한다. 그리고 히스토그램의 누적값 중 가장 큰 값을 가진 방향이 주각으로 사용되어진다. 또한 주각을  $\theta^P$ , 각 영상간의 주각의 차이인  $\Delta\theta$ 를 다음의 식 (3)과 같이 구할 수 있다.

$$\Delta\theta = \theta_i^P - \theta_j^P \quad (3)$$

다음으로 이동벡터를 계산하는데, 프로젝션 기반 상관관계(Correlation) 계산 방법을 사용한다[4][5]. 그러나 스마트폰의 성능을 고려하여, 다음의 식 (4)와 같이 수평 및 수직 프로젝션 데이터를 계산할 때 간격(Interval)을 두어 처리한다. 여기서 간격은  $Intv$ 라 한다.

$$P_V(i) = \sum_{j=1, j=j+Intv}^N Y(i, j) \quad (4)$$

이를 이용하여 다음의 식 (5)와 같이 상관관계를 계산한다. 이 때 역시 간격을 두어 실행 속도를 높인다.

$$C_Y(w) = \sum_{i=1, i=i+Intv}^{M-2S+1} |P_Y^i(i+w-1) - P_Y^i(i+S)| \quad (5)$$

마지막으로 상관관계 값 중 가장 최소가 되는 값을  $w_V^{MIN}$ ,  $w_H^{MIN}$ 이라 했을 때, 다음의 식 (6)과 같이 이동벡터  $\vec{T}$ 를 구할 수 있다.

$$\vec{T}_x = S + 1 - w_H^{MIN} \quad (6)$$

이렇게 구해진  $\Delta\theta$ 와  $\vec{T}$ 를 이용하여 입체영상으로 제작하기에 가장 적합한 두 장을 선택한다. 이는 다음의 식 (7)과 같은 비용함수를 생성하여, 가장 적은 비용을 가진 두 장의 영상이 최적의 영상으로 선택되어진다.

$$f(i, j) = \alpha \cdot \frac{|\theta_j - \theta_i|}{\theta_{max}} + \beta \cdot \frac{|T_x| + |T_y|}{(S_x + S_y)} \quad (7)$$

### 4. CG 콘텐츠 합성

입체영상이 제작되면, 여기에 합성할 CG 콘텐츠를 제작한다. 이는 카메라 프리뷰 및 입체영상을 보여주는 뷰와는 따로 동작하게 되는데, 입체영상이 출력되면 함께 제작되어, 입체영상 위에 출력되어진다. 이 때 가려짐 현상이 발생하기 때문에, 뷰 자체에 투명 속성을 부여함으로써 렌더링 된 콘텐츠 이외의 부분은 배경의 입체영상이 보이도록 제작한다. CG 콘텐츠는 구체(Sphere)를 렌더링하게 되는데, 크기 및 위치를 랜덤하게 하여 제작함으로써 깊이감을 준다. 또한 구체를 상하로 움직임으로써, 거품이 화면에 떠 있는 것과 같은 효과를 주게 된다.

### 5. 실험 결과

실험은 IOS 4.3 버전 상에서 이루어지며, 장비는 애플사의 iPod touch 4세대를 이용하였다. 우선 카메라를 수평으로 움직이며 총 5장의 프레임은 획득하였다. 이는 다음의 그림 3에서 보여준다.



그림 3. 카메라를 수평이동하며 촬영한 5 장의 영상

그림 3은 제작된 입체영상 및 합성된 CG 콘텐츠를 보여준다. 입체영상은 Anaglyph 방식으로 제작되었고, 이 때 선택되어진 영상은 2, 3번째 프레임이다. 또한  $\Delta\theta$ 는  $0.001^\circ$ ,  $\vec{T}$ 는 (32, 5)이다. 프레임 획득 후, 버튼을 터치할 때부터 입체 콘텐츠를 시청할 때 까지는 약 4초가 소요되었다.



그림 4. 합성된 CG 콘텐츠 및 제작된 입체영상

### 6. 결론

본 논문에서는 스마트폰의 카메라를 이용하여 입체영상을 제작하고, 여기에 CG 콘텐츠를 합성한 스마트폰 입체 증강현실 시스템을 제안하였다. 또한 입체영상을 제작할 때에는, 스마트폰의 성능을 고려하여 알고리즘을 고안하였다. 그 결과, 콘텐츠의 품질은 저하되었지만 제작 속도가 개선되는 효과를 얻었다.

### 감사의 글

본 연구는 교육과학기술부와 한국산업기술진흥원의 지역혁신인력양성사업 및 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2011-(C1090-1111-0003)).

### 참고문헌

- [1] Anders Henrysson, Mark Billinghurst, Mark Ollila, "Face to Face Collaborative AR on Mobile Phones", ISMAR05, pp. 80-90, 2005.
- [2] Gun A. Lee, Ungyeon Yang, Yongwan Kim, Dongsik Jo, Ki-Hong Kim, Jae Ha Kim, Jin Sung Choi, "Frees-Set-Go Interaction Method for Handheld Mobile Augmented Reality Environments", VRST 2009, pp. 143-146, 2009.
- [3] RenCheng Sun, FengJing Shao, ShuJing Li, Yi Sui, "The MagicARPhone: An AR-based Game on Phone Offering Personalized Enjoyment", WCSE 2009, pp. 264-266, 2009.
- [4] S. Tian, N. Wang, P. Zhao, C. Wang, "Aims at Moving Objects' Improvement Based on Gray Projection of Algorithm of the Electronic Image Stabilization", 3rd International Congress on Image and Signal Processing, pp. 2483-2487, 2010.
- [5] Yin Bin, Duan Hui-chuan, "Image Stabilization by Combining Gray-Scale Projection and Block Matching Algorithm", IT in Medicine & Education, pp. 1262-1266, 2009.