

Flying Cake: PDA를 이용한 실감형 게임

박안진, 양종열, 정기철
숭실대학교, 정보과학대학, 미디어학과
e-mail : {anjn,yjyhorse,kcjung}@ssu.ac.kr

Flying Cake: Augmented Game using PDA

Anjin Park, Jong-Yeol Yang, Keechul Jung
School of Media, College of Information Science, Soongsil University

요 약

언제, 어디서, 누구나 대용량 네트워크를 사용할 수 있는 유비쿼터스(ubiquitous) 시대가 다가오면서, 카메라가 장착되어 있고 무선 통신이 가능한 PDA, 웨어러블(wearable) 컴퓨터와 같은 휴대용 장치가 가까운 미래에는 일상의 한 부분이 될 것이다. 이런 상황을 반영하듯, 휴대용 장치를 이용하여 실감형 게임(augmented game)을 수행하는 다양한 연구가 진행되어 왔다. 기존의 실감형 게임들은 전통적인 'backpack' 시스템을 이용하거나 패턴마커(pattern marker)를 이용하였다. 'backpack' 시스템은 비싸고, 거추장스러우며, 사용하기 불편한 단점을 가지고 있으며, 패턴마커를 이용하면 미리 정한 장소에서만 게임을 즐길 수 있는 단점을 가지고 있다. 본 논문에서는 *Flying Cake*이라는 실감형 게임을 소개하며, 거추장스러운 장비 대신, 가볍고 휴대 가능한 PDA를 이용하여 게임을 수행하며, 실제 세계에서 가상의 물체를 접목할 위치를 지정하기 위해, 미리 정해져 있는 패턴마커 대신 얼굴 영역을 이용한다. *Flying Cake*은 연산자원이 부족한 PDA에서 실시간으로 얼굴 영역을 찾기 위해 스킨칼라(skin-color) 모델과 CAMShift 알고리즘을 이용하며, PDA상에서 카메라를 이용하여 실제 세계와 가상의 물체 사이의 상호작용을 제공하는 새로운 게임 패러다임(paradigm)을 통해 사용자에게 새로운 즐거움을 제공한다.

1. 서론

언제, 어디서, 누구나 대용량 네트워크를 사용할 수 있는 유비쿼터스(ubiquitous) 시대가 다가오면서, 카메라가 장착되어 있고 무선 통신이 가능한 PDA, 웨어러블(wearable) 컴퓨터와 같은 휴대용 장치가 가까운 미래에는 일상의 한 부분이 될 것이다.

이런 시대를 반영하듯 최근 휴대용 장치를 이용하여 실감형 게임(augmented game)을 수행하는 다양한 연구가 진행되어 왔다[1-7]. Szalavari[1] 등은 실제 환경에서 두 명 이상이 가상의 마작 패를 이용하여 게임을 진행하는 협동적인 실감형 게임 *마작*을 소개하였다. Ohshima[2] 등은 두 사람이 하나의 실제 환경인 테이블과 가상의 펙(puck)을 공유하는 실감형 게임 *air-hockey*를 소개하였다. Starnier[3] 등은 네트워크를 이용하여 여러 사람이 같이 즐길 수 있는 실감형 게임 환경을 제안하였으며, 제스처(gesture)를 통해 사용자 사이에 공격과 수비를 할 수 있는 *쿵후* 게임에 적용

하였다. Piekarski와 Thomas[4]는 일인용 슈팅 PC 게임인 *Quake*를 옥외에서 플레이할 수 있는 실감형 게임으로 확장한 최초의 게임으로, 플레이어가 실제 세계에서 이동할 수 있도록 하며 동시에 컴퓨터가 만들어 내는 그래픽 유령과 물체들을 경험하고 제거할 수 있도록 한다. Cheok[5] 등은 실제 환경인 도시 전체를 게임 공간으로 확장한 *Game-City*를 소개하였으며, 웨어러블 기기와 증강 현실을 함께 유비쿼터스 환경에 적용하여 개발하였다. 이렇게 개발된 *Game-City* 환경에서 사용자가 돌아다니며, 가상의 쿠키를 모우는 실감형 게임 *팍맨(pacman)*을 적용하였다[6].

위에서 언급한 게임들은 실제 세계에 가상의 물체를 정합하기 위해 HMD, 카메라, 추가적인 지원장비(노트북 등)로 구성된 전통적인 'backpack' 시스템을 이용하였다. 그러나 이 시스템은 비싸고, 거추장스러우며, 사용하기 매우 불편한 단점을 가지고 있다. 게다가 HMD는 가로 시각 범위가 34도 밖에 되지 않

때문에¹, 사용자는 시각적으로 많은 불편함을 느낀다 [2]. 이런 불편함을 해결하기 위해 거추장스러운 시스템 대신 PDA를 이용하여 수행된 실감형 게임이 소개되었다[7]. 그러나 이 게임은 실제 세계에서 가상의 물체를 정합할 위치를 지정하기 위해 패턴마커(pattern marker)를 이용하였으며, 패턴마커가 미리 놓여져 있는 장소에서만 게임이 가능한 단점을 가지고 있다.

본 논문에서는 *Flying Cake*이라는 실감형 게임을 소개한다. 이 게임은 케익을 던져서 적의 얼굴 영역에 접촉된 가상의 캐릭터를 공격하는 게임으로써, 기존의 거추장스러운 'backpack' 시스템 대신 가볍고, 휴대 가능하며, 네트워크와 쉽게 연동할 수 있는 PDA를 이용하였으며, 가상의 캐릭터를 접촉할 위치가 미리 정해져 있으며, 항상 휴대하고 있어야 게임이 가능한 수동적인 패턴마커 대신, 언제 어디서나 게임을 즐길 수 있는 얼굴 영역을 이용하였다. 우리는 얼굴 영역 추출을 위해 스킨칼라 모델[10]과 CAMShift 알고리즘[11]을 이용하였으며, 이 방법은 부족한 연산 자원을 가진 PDA에서 효과적인 수행 시간을 보인다.

본 논문의 구성은 다음과 같다. 제 2장에서 *Flying Cake*의 전체적인 시스템 설계에 대해 기술하고, 제 3장에서 입력 영상에 가상의 캐릭터를 접촉할 위치를 계산하기 위한 얼굴 추출 방법을 기술하며, 제 4장에서 3차원 공간에서 케익을 던지고, 공격의 성공여부를 확인하기 위해 필요한 3차원 정보에 대해 기술한다. 마지막으로 제 5장에서 결론 및 향후 연구 방향을 기술한다.

2. Flying Cake

실감 현실(augmented reality) 시스템은 다음의 3가지 특성을 가지고 있어야 한다[8]. 첫째, 실제 환경에서 가상의 물체가 접촉되어야 한다. 둘째, 실시간 상호작용이 있어야 한다. 셋째, 가상의 물체는 3차원상에 정합되어야 한다. 우리 게임은 얼굴 추출 결과를 바탕으로 PDA 화면에 보여지는 실제 환경에 가상의 캐릭터를 접촉하며, 날아오는 케익을 피하기 위해 얼굴을 움직이므로써 실시간 상호작용을 수행하고, 빠른 수행 시간을 위해 가상 캐릭터의 좌우방향(yaw), 상하방향(pitch), 회전(roll)을 고려하지 않는다는 조건하에 3차원상의 정합을 2차원상의 접촉으로 생각하여 게임을 수행한다.

그림 1은 *Flying Cake*의 시스템 구성을 보여주며, 싱글(single) 모드와 듀얼(dual) 모드로 진행된다. 그림 1(a)는 싱글 모드로, 하드웨어(hardware)는 카메라가 장착된 PDA만을 필요로 한다. 카메라는 상대방(적)을 향하며, PDA 화면에는 가상 캐릭터가 접촉된 상대방이 보인다. 그림 1(b)는 듀얼 모드이며, 하드웨어는 카메라가 장착된 PDA, 무선 통신 모듈 그리고 서버로 구성된다. 카메라는 사용자 자신에게 향하며, PDA 화면에는 무선 랜을 통해 수신된 상대방의 카메라를 통해 입력된 영상이 보인다.

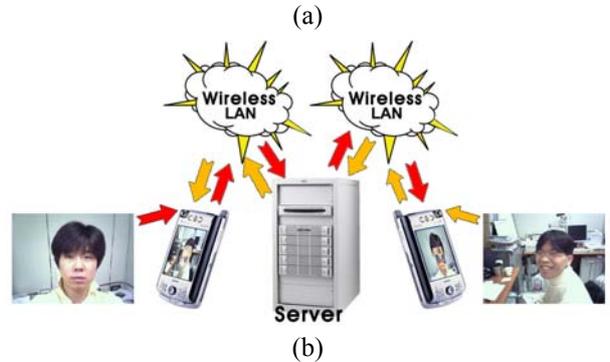
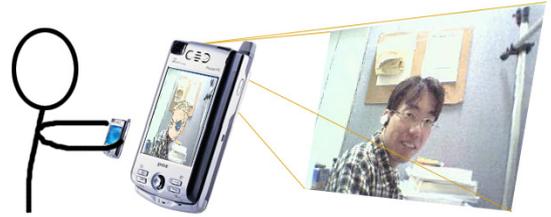


그림 1. *Flying Cake*의 시스템 구성: (a) 싱글 모드, (b) 듀얼 모드.

그림 2는 PDA 화면을 보여준다. 심벌(symbol) C는 얼굴 영역에 가상의 캐릭터가 접촉되어 있는 상대방을 보여주는 부분이다. 심벌 D는 바람의 방향이며, 케익이 날아간 X축 방향에 영향을 준다. 심벌 F는 내 가상 캐릭터이며, 내 캐릭터가 점점 붉은색으로 변함으로써 적의 케익이 가까워짐을 알려 준다. 심벌 G는 공격을 수행할 때 케익의 출발 위치이며, 심벌 H는 케익을 던지기 위한 파워게이지(power gauge)이며, 케익이 날아간 깊이 정보에 영향을 준다. 그림 3은 가상 캐릭터의 예를 보여준다. 그림 3에서 보는 바와 같이 적의 공격으로 에너지가 줄어들면, 가상 캐릭터는 점점 더 많은 상처를 가지게 되며, 그림 2에서 적의 캐릭터가 내 가상 캐릭터보다 더 상처를 가지는 건, 적과 내 에너지의 차이 때문이다.



- A: Enemy Energy
- B: My Energy
- C: Captured Image of Enemy with a Bruised Virtual Character.
- D: Direction of Wind
- E: Items
- F: My Virtual Character with a Black Eye
- G: Start Position of the Cake
- H: Power Gauge

그림 2. 게임이 진행중인 PDA 화면.



그림 3. 가상 캐릭터의 예.

¹ 사람의 가로 시각 범위는 약 100 도 정도이다.

3. 얼굴 추출(Face Detection)

이 게임은 PDA 카메라를 통해 입력된 영상의 얼굴 영역에 가상의 캐릭터를 접목시킨다. PDA에서 사용하는 대부분의 CPU는 실수연산 구성요소가 없는 정수형 CPU를 사용하며, 실수연산이 많은 비전 시스템이나 영상 처리를 처리하는데 오랜 수행시간이 소요된다[9]. 그래서 우리는 연산자원이 부족한 PDA에서 실시간으로 얼굴 추출을 수행하기 위해, 스킨칼라 모델[10]과 CAMShift 알고리즘[11]을 이용한다.

피부색과 조명에 민감하지 않는 스킨칼라 모델은, 여러 샘플에서 얻어진 사람의 피부색 영역 각각의 RGB 칼라 데이터를 조명의 영향을 덜 받는 채색 공간으로 변환하면 가우시안(gaussian) 분포로 근사화되고, 이를 이용하여 얻을 수 있다[10].

일반적으로, 각 픽셀의 칼라 값은 R,G,B의 조합으로 표현되며, 밝기 값은 $I = R + G + B$ 로 표현된다. 스킨칼라는 밝기 값에 민감하므로, 각 픽셀을 채색 공간으로 변형해야 하며, 다음과 같은 정규화된 RGB 모델(normalized RGB)을 이용한다.

$$r = R/I, g = G/I, b = B/I \quad (1)$$

여기서 r과 g값만을 이용하고, 채색 공간 (r, g) 에서 다양한 조명, 다양한 피부색을 가진 사람들의 스킨 칼라 분포는 서로 비슷한 가우시안 분포를 가지며, 다음과 같이 2D 가우시안 분포 $G(x, y)$ 로 표현된다.

$$G(x, y) = \frac{1}{2\pi\sigma_r\sigma_g} e^{-\frac{1}{2}\left[\left(\frac{r(x,y)-m_r}{\sigma_r}\right)^2 + \left(\frac{g(x,y)-m_g}{\sigma_g}\right)^2\right]} \quad (2)$$

$r(x, y)$, $g(x, y)$ 는 x, y 좌표에서의 r, g 값이며, m_r, m_g 는 미리 정한 스킨 칼라 영역의 r, g 평균을 나타내며, σ_r, σ_g 는 분산을 나타낸다. 본 실험에서는 m_r, m_g 를 각각 0.463705, 0.267740, σ_r, σ_g 는 각각 0.002883, 0.001239값을 사용하였고, 입력 영상의 (r, g)를 모델링 된 가우시안 분포(식(2))를 이용하여 스킨 칼라와 비-스킨 칼라를 구분한다.

가우시안 분포를 이용하여 얼굴 영역을 추출하면 입력 영상 전체를 수행해야 하기 때문에 오랜 수행시간이 소요되는 단점이 있다. 우리는 이런 단점을 해결하기 위해 CAMShift 알고리즘을 이용하였다(그림 4).

■ Set up the initial locations ($mean_{x_0}, mean_{y_0}$) and sizes ($\lambda_{x_0}, \lambda_{y_0}$) of search window W

■ Do

Generate the detected face image within W using skin-color model.

Based on the mean shift vector, derive the new location ($mean_{x_t}, mean_{y_t}$) and size($\lambda_{x_t}, \lambda_{y_t}$) of a face region.

Modify W according to the derived values.

Increment the iteration number t .

While($\|mean_{x_t} - mean_{x_{(t-1)}}\| > \epsilon_x$ or $\|mean_{y_t} - mean_{y_{(t-1)}}\| > \epsilon_y$)

그림 4. 얼굴 추출을 위한 CAMShift 알고리즘.

시작 단계에서 초기 위치($mean_{x_0}, mean_{y_0}$)와 크기(λ_{x_0} ,

λ_{y_0})를 결정하고, 연속된 일련의 단계에서 2차원 모멘트(moment)를 계산하여 얼굴 영역의 크기와 위치를 구하며, 검색창의 크기를 얼굴 영역의 크기에 비례해서 수정한다. 그리고 미리 정한 임계값 보다 검색창의 크기가 변하지 않을 때까지 반복 수행한다.

그림 5는 CAMShift 알고리즘을 이용하여 검출된 얼굴 영역 결과 영상이다. 그림 5(a,b)는 입력 영상이며, 5(c,d)는 얼굴 영역은 흰색, 비-얼굴 영역은 검은색으로 표시된 결과 영상이다. 입력 영상 5(b)는 두 개의 얼굴 영역을 가지지만, 결과 영상 5(d)를 보면 하나의 얼굴 영역만을 가진다. 이것은 미리 정한 임계값 보다 검색창의 변화가 작기 때문에 더 이상의 수행이 되지 않아 생긴 결과이며, 이전 프레임에서 결정된 얼굴 영역의 위치(그림 5(c))가 다음 영역의 초기값으로 들어가기 때문에 왼쪽 얼굴이 검출된다. 결과적으로, CAMShift 알고리즘은 입력 영상 전역 탐색에 관한 문제를 해결하며, 영상에서 얼굴 영역이 많이 차지하지 않을 때 계산량을 줄일 수 있다.

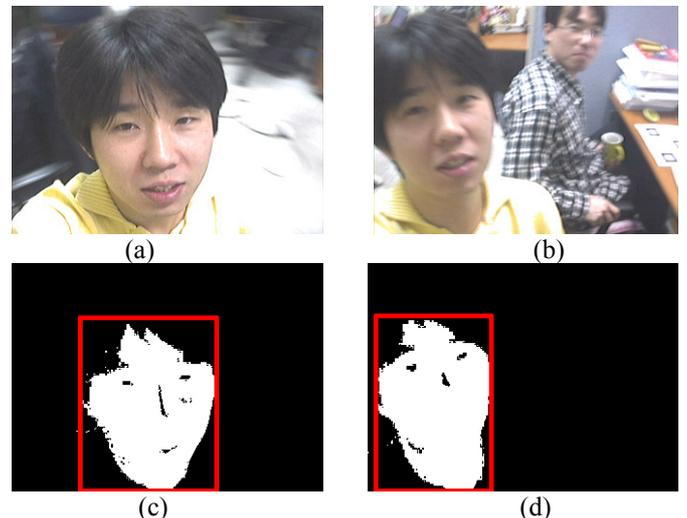


그림 5. 얼굴 추출 결과:(a),(b) 입력 영상, (c),(d) 추출된 결과 영상.

4. 3차원 정보

가상의 캐릭터를 3차원 상에 정합하기 위해서는 2차원 입력 영상에 대한 3차원 정보가 필요하다. 우리는 PDA에서 게임의 빠른 수행을 위해 3차원 정합에서 가상 캐릭터의 좌우방향, 상하방향, 회전을 고려하지 않는다. 이런 가정 하에 접목된 가상 캐릭터는 PDA 화면에서 2차원으로 보이기 때문에 (그림 6), 실제 환경에서 가상 캐릭터를 접목할 때 얼굴 추출의 결과인 폭, 높이, 중심값만을 이용하여 2차원적으로 접목하여 게임을 수행한다. 그리고, 공격을 수행할 때 케익은 포물선을 그리며 날아가며, 사용자에게 의해 입력 받은 x, y 좌표와 파워 값에 의해 날아가 위치가 결정된다. 3차원 공간에 날아오는 케익의 공격성공 여부를 확인하기 위해서는 PDA에서 입력 받은 그림 7은 2차원 입력 영상을 3차원 좌표 계로 변환한 영상을 보여주며, 그림 7에서 우리는 깊이 정보가 얼굴 영역의 크기와 비례함을 알 수 있다. 이 가정을 기반으로 우리는

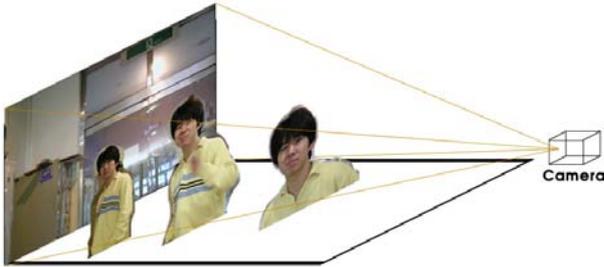
얼굴 영역의 크기를 이용하여 대략적인 깊이 정보를 구한다. 우리는 PDA상에서 복잡한 계산을 피하고 빠른 수행을 위해 얼굴 영역의 크기에 비례해서 깊이 정보를 5단계로 구분(그림 8)하고, 얼굴 영역의 크기와 파워게이지를 이용해서 공격의 성공과 실패를 구분한다. 그림 8은 파워게이지에 의해 날아가는 케이크의 범위를 보여주며, 얼굴 영역의 크기와 파워게이지의 상관관계를 보여준다.



(a) (b)
그림 6. 가상 캐릭터가 접촉된 입력 영상.



(a)



(b)

그림 7. 2 차원 입력 영상을 3 차 좌표계로 변환한 영상: (a) 2 차원 입력 영상, (b) 3 차원으로 변환된 영상.

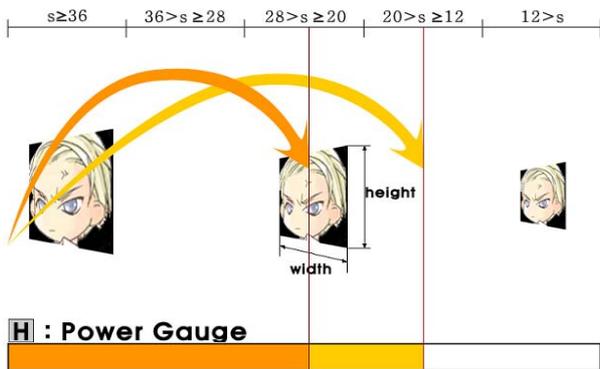


그림 8. 발사대의 파워게이지에 의해 케이크가 날아가는 깊이 범위.

5. 결론

기존의 실감형 게임은 비싸고, 거추장스러우며, 사용하기 불편한 'backpack' 시스템과 미리 지정되어 있

는 장소에서만 가상의 오브젝트 생성이 가능한 패턴 마커를 이용하였다. 우리는 실감형 게임 *Flying Cake*을 제안하였으며, 거추장스러운 장비 대신 PDA를 이용하였고, 실제 세계에서 가상의 물체를 접촉할 위치를 미리 지정하는 패턴마커 대신 얼굴 영역을 이용하였다. *Flying Cake*은 적은 연산 자원을 가진 PDA상에서 실시간으로 수행되며, 가상의 오브젝트와 실제 세계 사용자 사이에서의 상호작용을 제공하는 새로운 게임 패러다임을 제공함으로써 사용자에게 새로운 즐거움을 제공한다.

그러나, 현재 게임에서 싱글 모드에서 적의 에너지를 알 수 없다. 적의 에너지를 알기 위해 PDA의 연산 자원에 적합한 얼굴 인식, GPS를 이용한 싱글과 듀얼 모드의 결합과 같은 연구가 진행중이며, 3차원 가상 캐릭터의 접촉에서 얼굴 영역의 좌우방향, 상하방향, 회전에 대한 연구도 진행중이다.

참고문헌

- [1] Z. Szalavari, E. Eckstein and M. Gervautz, "Collaborative Gaming in Augmented Reality," Proceedings of ACM Symposium on Virtual Reality Software and Technology, pp. 195-204, Nov. 1998.
- [2] T. Ohshima, K. Satoh, H. Yamamoto and H. Tamura, "AR2Hockey: A Case Study of Collaborative Augmented Reality," Proceedings of IEEE Virtual Reality Annual International Symposiums, pp. 268-275, Mar. 1998.
- [3] T. Starner, B. Leibe, B. Singletary and J. Pair, "MIND-WARPING: Towards Creating a Compelling Collaborative Augmented Reality Game," Proceedings of International Conference on Intelligent User Interface, pp. 256-260, Jan. 2000.
- [4] W. Piekarski and B. Thomas, "ARQuake: The Outdoor Augmented Gaming System," Communications of the ACM, Vol. 45, No. 1, pp. 36-38, 2002.
- [5] A. Cheok, F. Wan, X. Yang, W. Weihua, L. Huang, M. Billinghurst and H. Kato, "Game-City: A Ubiquitous Large Area Multi-Interface Mixed Reality Game Space for Wearable Computers," Proceedings of the International Symposiums on Wearable Computers, pp. 156-157, Oct. 2002.
- [6] A. Cheok, H. K. Goh, W. Liu, F. Farbiz, S. Fong, S. Teo, Y. Li and X. Yang, "Human Pacman: a Mobile, Wide-area Entertainment System based on Physical, Social, and Ubiquitous Computing," Personal and Ubiquitous Computing, Vol. 8, Issue 2, pp. 71-81, May 2004.
- [7] D. Waqner, T. Pintaric and F. Ledermann, "Towards Massively Multi-User Augmented Reality on Handheld Devices," Proceedings of the International Conference on Pervasive Computing, to be published, May 2005.
- [8] R. Azuma, Y. Baillet, R. Behringer, S. Feiner, S. Julier and B. MaxIntyre, "Recent Advances in Augmented Reality," Computer and Graphics, Nov. 2001.
- [9] A. Park and K. Jung, "PDA-based Text Extraction System using Client/Server Architecture," Lecture Notes in Computer Vision, Vol. 3157, pp. 833-842, 2004.
- [10] J. Yang and A. Waibel, "A Real-Time Face Tracker," Proceedings of IEEE Workshop Applications of Computer Vision, pp. 2-4, Dec. 1996.
- [11] G. Bradski and V. Pisaresvsky, "Intel's Computer Vision Library: Applications in Calibration, Stereo, Segmentation, Tracking, Gesture, Face and Object Recognition," Proceedings of IEEE Conference of Computer Vision and Pattern Recognition, Vol. 2, pp. 796-797, 2000.