전경 이미지 분리와 마우스 트래킹 프로그램을 이용한 사용자 관심 영역 유도

이면재

백석대학교정보통신학부

davidlee@bu.ac.kr

Extraction of user interest area using foreground image separation and
mouse tracking program

MyounJae-Lee

Division of Information&Communication, Baekseok University

요 약

영상 처리에서 전경 이미지 추출은 움직이는 대상이나 객체를 인식하려는 경우에 주로 응용된다. 게임에서 전경 이미지에 포함되는 객체들은 주로 캐릭터와 NPC(Non Player Character), 아이템 등이 될 수 있다. 이 객체들은 플레이어들의 이동, 공격, 방어, 수집의 대상이 되는 객체들로 플레이어들의 주요 관심 대상이 될 수 있다. 본 연구는 이러한 배경에서, 플레이어들의 관심 영역을 추출하기 위한 연구이다. 이를 위해, 첫째, 전경 이미지를 추출한다. 둘째, 추출한 전경 이미지를 일정시간 누적시켜서 히트맵(Heat Map) 이미지를 결과 이미지로 보여준다. 마지막으로 마우스 트래킹 프로그램을 이용하여 마우스 이동 영역을 검출하고 히트맵 이미지와 비교함으로써 플레이어의 관심 영역을 유도할 수 있다.

ABSTRACT

The location of the objects that make up a game can be an element of immersion for players. repeatedly appearing at the same position, the fun may be reduced, and as the play time elapses, the players will feel the game's fun as they appear in a larger area than at the beginning of the game play. This paper is a study to find out the location of objects according to the passage of time and to see how players controlled these objects. First, foreground images are extracted and accumulated using OpenCV programming language. The accumulated result is displayed as a heat map image. Second, the mouse movement area is detected using the mouse tracking program and compared with the heat map image, so that the screen area in which the player is interested can be known.

Keywords : User Interest Area(사용자관심영역), Extraction of Foreground Image(전경 이미지 추출), Mouse Tracking(마우스 트래킹), Heat map(히트맵), Fun Element(재미요소)

Received: Oct. 01. 2017   Revised: Oct. 18. 2017

Accepted: Oct. 20. 2017

Corresponding Author: MyounJae-Lee(소속)

E-mail: davidlee@bu.ac.kr

© The Korea Game Society. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

ISSN: 1598-4540 / eISSN: 2287-8211

1. 서론

대부분의 게임에서 플레이어들은 새롭게 등장하는 게임 오브젝트들, 즉 NPC(Non Player Character)나 적을 공격하거나 전략 전술에 도움이 될만한 아이템을 획득하기 위해 마우스를 이동하거나, 스코어를 확인하면서 게임을 플레이한다. 그리고 동료들이 게임 맵의 어떠한 위치에 있는지 상황을 파악하기 위해 미니맵을 보기도 한다. 이러한 플레이어들의 모든 행동들은 플레이어들의 시선을 필요로 한다. 플레이어들은 게임 객체들의 특성과 등장 위치에 따라 시선을 이동하고 해당 객체에 적합한 조작을 행함으로써 게임을 플레이한다.

이러한 게임 객체들은 플레이어가 제어 가능한 여부와 게임 객체가 포함된 레이어 레벨에 따라서 전경(foreground) 이미지의 구성 요소가 되거나 배경(background) 이미지의 구성 요소가 된다. 배경 이미지의 구성 요소에는 게임 진행의 밑 바탕 이미지로서 길, 도로, 나무 등으로 구성되며 게임 씬을 구성하는 처음 레이어로 가장 먼저 화면에 디스플레이 되며 게임 시작 초기부터 일정한 패턴으로 반복 재생되는 특징을 갖고 있다. 후에 전경 이미지의 구성 요소들인 NPC, 미니맵, 생명치, 채팅창 등이 배경 이미지 위에 그려진다. 이 전경 이미지의 구성 요소들은 게임을 플레이하는 도중에 화면에 등장하는 객체들이다. 플레이어들은 게임 씬에서 새롭게 등장하는 전경 이미지의 구성 요소들을 제어해서 게임을 이겨야 하기 때문에 전경 이미지의 구성 요소들이 등장하는 화면 영역에 집중과 주의를 둔다. 이 영역들이 플레이어들의 주요 관심 영역이 된다.

본 논문은 플레이어들의 관심 영역이 되는 전경 이미지를 구성하고 있는 화면 영역에 관한 연구이다. 이 영역의 크기가 너무 작은 경우 플레이어들은 작은 영역에서 전경 이미지를 구성하고 있는 객체들을 제어하기 때문에 반복적인 플레이로 인해 지겨움을 느낄 수 있다. 이와 비교하여 전경 이미지를 구성하는 요소들이 게임 시작부터 너무 넓은

영역에 등장하는 경우 게임의 집중도가 저하될 수 있으며 초보자들에게 게임 진입 장벽이 어렵게 느껴질 수 있는 요인이 될 수 있다.

본 논문은 플레이어들의 관심 영역을 알아보기 위한 연구이다. 이를 위하여, 첫째 시간에 따른 전경 이미지의 구성 요소들의 등장 영역을 히트맵(Heat Map)으로 표현한다. 둘째, 히트맵으로 표현된 화면 영역과 플레이어의 마우스 조작과의 연관성을 알아보기 위해, 실시간으로 마우스 트랙킹 프로그램을 이용하여 마우스 조작 화면 위치를 분석한다. 마우스를 조작한다는 것은 플레이어들의 시선과 관심이 이동된다는 것을 뜻하기 때문에 마우스 트랙킹 프로그램을 사용한다.

이를 위하여 본 논문에서는 첫째, OpenCV(Open Source Computer Vision) 프로그램을 사용하여 배경 이미지 모델을 생성하고 생성된 배경 이미지 모델을 학습한다. 둘째, 학습된 이미지를 바탕으로 입력으로 들어온 프레임들에서 전경 이미지를 추출하고 추출된 이미지를 누적한다. 셋째, 누적된 전경 이미지를 히트맵으로 변환한다. 이러한 히트맵 이미지는 전경 이미지를 구성하고 있는 객체들이 등장하는 화면 영역을 알 수 있도록 도와준다.

마지막으로 마우스 트랙킹 프로그램을 사용하여 시간에 따른 마우스 이동 경로를 파악한다. 이는 전경 이미지가 차지하는 화면 영역과 마우스 이동 경로를 비교함으로써, 플레이어의 조작 대상 즉 관심 대상과 실제 전경 이미지의 구성 요소가 등장하는 화면 영역을 파악하는데 도움을 준다. 본 연구에 사용된 게임은 라이엇 게임즈에서 제작한 리그오브레전드이다[1].

본 논문의 구성은 다음과 같다. 2장에서는 관심 영역과 OpenCV의 이용 분야에 대한 관련 연구를 살펴보고 3장에서는 게임 화면의 전경 이미지를 히트맵으로 변환하는 과정과 마우스 트랙킹 프로그램을 설명하고 구현 결과를 설명한다. 4장에서는 결론 및 추후 연구방향에 대해 논한다.

2. 관련 연구

관심 영역은 다양한 분야에서 활용되고 있다. 관심 영역 기법을 자동 노출 조절 알고리즘에 적용하여 도로 상황 분석 성능을 높인 연구[2], 이동형 단말기에서의 축구경기 시청을 위한 해상도 및 관심 영역 크기에 관한 사용자 만족도 조사에 관한 연구[3], 관심 영역 구분을 통한 감시영상시스템 구현에 관한 연구[4], 얼굴검출에 기반한 강인한 객체 추적 시스템[5], 관심 영역을 결정하고 불필요한 부분을 제거하여 차선인식을 위한 계산 시간을 줄이기 위한 연구[6], 살펴본 연구에서와 같이 관심 영역은 실시간으로 입력으로 들어온 비디오 영상에서 객체를 판별하는 목적과, 콘텐츠 기반의 영상 축소 및 확대, 영상 분할, 감시 시스템등 다양한 부분에 주로 사용되어져 왔다[7].

게임 화면은 게임의 진행 상황을 보여주는 게임 진행정보, 소유하고 있는 리소스와 유닛을 보여주는 리소스 정보, 선택된 개체의 정보를 나타내는 개체 설명 정보, 맵 정보, 메뉴 정보로 구성되어 있다. [Fig. 1]은 리그오브레전드 게임화면을 보여준다. [Fig. 1]의 게임 화면에서 보듯이 왼쪽 상단에는 선택한 캐릭터의 정보가 나오고 왼쪽 하단에는 채팅, 자신의 캐릭터 정보가 있으며 오른쪽 상단은 게임시간과 스코어가 제시되고 오른쪽 하단은 미니맵과 아군들의 상태가 표시된다.

[Fig. 1] League of Legend Game Screen Shot

[Fig. 2]는 본 연구에서 사용된 마우스 트래킹 프로그램인 IOGraph[8]를 설치하고 실행한 후 3시간동안 포토샵 프로그램을 실행하여 마우스를 트래

킹한 결과[8]를 보여준다. 화면 중간 영역은 이미지를 편집하기 위해 마우스를 클릭하고 이동하였기 때문에 다소 마우스 트래킹 흔적이 많은 것을 볼 수 있으며 화면 상단은 메뉴를 선택하는데 마우스가 움직인 것으로 보여진다.

[Fig. 2] Mouse Tracking Example

[Fig. 3]은 포토샵 프로그램을 실행시킨 예제[9]이다. 포토샵을 실행시킨 후 화면의 레이아웃 구성 형태에 따라 마우스 트래킹한 결과가 달라질 수 있으나 일반적으로 화면 상단은 메뉴가 화면 중간은 편집할 오브젝트가 위치하고 있다.

배경 이미지를 분리하여 누적하고 이를 히트맵으로 변환하는 것은 OpenCV[10]를 이용하여 구현한다. OpenCV는 컴퓨터 비전과 이미지 프로세싱 프로그램 개발을 위한 오픈 소스 라이브러리로, MS 윈도우, 리눅스 등에서 사용할 수 있다. 히트맵은 열(heat)과 맵(map)이 결합된 단어로 다양한 정보를 열 정보 형태의 그래픽으로 보여주는 것이다. 히트맵은 사람이 많이 모이는 영역이나, 자동차가 많이 운행되는 지역 등을 열정보로 변환시켜

서 주어서 자주 등장하는 영역일수록 빨간색을 띄며 적게 등장할수록 파란색을 띄게 된다. [Fig. 4][11]는 축구장에서 축구 선수들이 이동하는 영역을 히트맵으로 표현한 것이다. 이와 같이 히트맵은 출몰 정도나 빈도를 비주얼 효과로 표현해 주는데 유용하다.

[Fig. 4] Heat Map example

3. 구 현

3.1 구현 환경

본 논문의 실험은 마우스를 트래킹하기 위한 프로그램 실행과 배경 이미지를 누적하여 히트맵으로 변환하는 부분으로 구성된다. 이 실험에 사용된 PC의 CPU는 Intel I5-3570이고 메인 메모리는 8GB, SSD는 120GB이다. 개발에 사용된 통합 환경은 Microsoft Visual Studio 2015이고, 마우스 트래킹에 사용된 프로그램은 IOGraphica사의 IOGraph로 버전 1.1이다. IOGraph는 마우스가 움직이는 과정을 트래킹하여 아트로 변환시켜주는 소프트웨어로서 Anatoly Zenkov가 개발한 것이다. 영상 재생 시간은 30분 49초이며, 용량은 140Mb, 초당 프레임 개수는 24이며 총 프레임 개수는 43068 프레임이다.

3.2 구현 알고리즘

구현 알고리즘은 크게 3단계로 구성된다. 첫 번째는 배경 이미지 모델(Background

Model(BGMode))을 생성하고 배경 이미지를 학습하는 단계이다. 이때 모든 프레임들은 그레이 이미지로 변환된 이미지로 학습된다. BG모델은 전경 이미지와 배경 이미지를 분리하기 위한 모델로서 학습에 필요한 프레임 개수를 입력으로 받을 때까지는 전경 이미지와 배경 이미지를 학습하고 그렇지 않는 경우에는 학습된 배경 이미지를 바탕으로 배경 이미지와 전경 이미지를 분리한다.

두번째는 학습이 종료된 BG 모델을 사용하여 전경 이미지를 추출하고 이를 누적하는 단계이다. 세 번째는 누적된 전경 이미지를 히트맵으로 변환하는 단계이다.

첫 번째 단계에서는 cvCreateGaussianBGMode() 함수를 이용하여 BG 모델을 생성하고 cvUpdateBGStatModel() 함수를 호출하여 배경 이미지를 학습시킨다. 학습에 필요한 프레임 개수가 많을수록 정확한 배경 이미지를 분리할 수 있지만 계산 시간이 많이 소요되므로 50으로 설정한다. [Fig. 5]는 첫 번째 단계의 결과물로서 200번째 프레임 이미지([Fig. 5] (a))와 이를 그레이 스케일로 변환한 이미지([Fig. 5] (b))를 각각 보여준다.

(a) Input Frame

(b) Grayscale Image

[Fig. 5] 1st Step

두 번째 단계에서는 학습을 마친 BG 모델을 중심으로 `cvUpdateBGStatModel()` 함수를 호출하여 전경 이미지와 배경 이미지를 분리한다. 이후에 추출된 전경 이미지에서 불필요한 부분을 제거하기 위하여 침식 연산과 팽창 연산을 수행하고 이 결과를 다듬는 연산을 수행한다. 침식 연산(`cvErode()`)과 팽창 연산(`cvDilate()`)은 각각 흰색 부분이 있는 부분을 강조하고 검은색 부분에 있는 부분을 강조하는 역할을 수행한다. 즉 경계의 모호함을 없애는데 사용된다. 이후에 `cvSmooth()` 연산을 수행한다. [Fig. 6] (a)는 학습이 종료된 BG Model의 전경 이미지를 보여준다. 새로운 프레임이 들어오면 이 이미지를 기준으로 배경 이미지와 전경 이미지를 분리한다. [Fig. 6] (b)은 [Fig. 6] (a)의 학습된 BG 모델로부터 입력 프레임의 전경 이미지를 추출하여 침식 연산, 팽창 연산, 그리고 스무딩 연산을 수행한 결과이다. 200번째 프레임 자체에서 플레이어의 이동이나 NPC등과 같은 전경 이미지를 구성하는 요소들이 프레임에 많지 않아서 학습된 BG 모델의 전경 이미지([Fig. 6] (a))와 큰 차이를 보이지 않는다. 게임을 플레이 한지 약 8초 밖에 경과되지 않아서 이러한 결과를 보인 것이다.

(a) Foreground Image in BG Model

(b) Erode, Dilate, Smooth Operation

[Fig. 6] 2st Step

세 번째 단계는 두 번째 단계에서의 결과 이미지, 즉 전경 이미지에 대해 현재까지 누적된 전경 이미지를 `cvAdd()` 연산으로 합한 이미지를 생성한다. 그리고 합한 이미지 결과에 컬러맵을 적용하여 히트맵으로 변환한다. [Fig. 7]은 200번째 프레임까지의 누적된 전경 이미지([Fig. 7] (a))와 이 이미지를 히트맵으로 변환한 결과 이미지([Fig. 7] (b))를 보여준다. [Table 1]은 구현된 알고리즘을 보여준다.

(a) Accumulated Foreground Image

(b) Accumulated Heat Map Image
[Fig. 7] 3st Step

[Table 1] The proposed algorithm

- 1) BG model is created and background image is learned.
- 2) The foreground image and the background image are separated around the learned BG model
- 3) The foreground image is processed by erosion, expansion, and smoothing to refine the foreground image generated by the BG model.
- 4) Accumulates the current foreground image and the accumulated foreground image.
- 5) Apply a color map to the accumulated result image to show it as a heat map.

3.3 구현 결과

실험에 사용된 게임은 리그오브레전드이다. 리그오브레전드 게임은 라이엇게임즈에서 개발하여 서비스를 제공한다. 게임 장르는 온라인 RTS(Real Time Strategy)에 속한다. 게임 플레이를 시작하고 나서 5분 10분 20분 30분이 경과 되었을 때 마우스 트래킹 프로그램을 실행한 결과와 전경 이미지를 추출하여 이를 히트맵으로 변환한 구현 결과를 비교한다. 두 방법 모두 게임 플레이를 시작한 이후부터 해당 시간까지의 결과를 누적한 결과를 보여준다. 히트맵을 실행한 결과 화면은 전경 이미지가 많이 추출된 영역일수록 빨간색을 띄며 파란색을 띤 영역일수록 전경 이미지 추출이 적게 된 영역이다. 검은색은 전경 이미지가 출현되지 않는 영역이다.

[Fig. 8]는 게임 플레이를 시작한 이후부터 5분이

경과된후 마우스 트래킹 프로그램([Fig. 8] (a))과 전경 이미지에 대한 누적 결과를 추출하여 히트맵으로 변환한 결과([Fig. 8] (b))이다. 마우스 트래킹을 실행한 결과의 경우, 캐릭터를 선택하기 위해 화면 중앙 영역에서 마우스를 많이 이동하고, 왼쪽 상단의 경우 선택한 대상 캐릭터의 정보를 보기 위해 마우스를 많이 이동한 것을 알 수 있다.

히트맵 변환 이미지의 경우에서도 전투를 하기 위해 플레이어 캐릭터들을 이동하거나 NPC들이 화면 중앙 영역에서 주로 등장하여서 화면 중앙 영역이 빨간색을 띄고 있으며 왼쪽 상단은 배경 이미지 위에 캐릭터의 정보가 계속 디스플레이 되고 있기 때문에 빨간색을 띄고 있는 영역이 일부 있다.

(a) Mouse Tracking

(b) Heat Map
[Fig. 8] 5 Minute

[Fig. 9]는 게임 플레이가 시작된 후 10분이 경과된 누적된 결과를 보여준다. 마우스를 트래킹한 실험 결과의 경우 [Fig. 8] (a) 보다 화면 중앙 영역의 마우스 이동이 많아져서 다소 많은 마우스

이동 라인이 표시되었고 오른쪽 상단 영역의 스코어와 오른쪽 하단 영역에 미니맵 영역에 마우스가 많이 이동되었다. 히트맵으로 변환한 이미지의 경우([Fig. 9] (b)) [Fig. 8] (a)에서와 비슷하게 화면 중앙 영역이 빨간색을 띄고 있다.

(a) Mouse Tracking

(b) Heat Map

[Fig. 9] 10 minute

[Fig. 10]은 게임 플레이가 시작된 후 20분이 경과된 누적 결과를 보여준다. [Fig. 10] (a)에서는 [Fig. 9] (a)와 [Fig. 9] (b)에서와 같이 화면 중앙 영역에서 마우스 이동이 많았음을 알 수 있다. 히트맵의 경우 화면 중앙 영역이 다른 화면 영역에 비해 현저하게 빨간색을 띄고 있으며 화면 왼쪽 상단 영역이 많이 이동된 것을 알 수 있다. 전경 이미지의 구성 요소가 많이 나타나는 영역에 마우스도 많이 움직였다는 것을 의미하며 이는 플레이어의 시선도 해당 영역에 집중하였음을 뜻한다.

[Fig. 11]는 게임 플레이가 시작된 후 30분이 경과된 누적 결과를 보여준다. 마우스 트래킹 프로그램의 실행결과([Fig. 11] (a))에서는 화면 중앙영역에서 플레이어가 캐릭터를 선택하여 전략을 구사하는 이유로 마우스 이동이 가장 많음을 확인할 수 있다. 히트맵의 실행 결과([Fig. 11] (b))에서도 전경 이미지를 구성하는 객체들이 화면 중앙에 많이 차지하고 있음을 알 수 있다.

(a) Mouse Tracking

(b) Heat Map

[Fig. 10] 20 Minute

(a) Mouse Tracking

(b) Heat Map
[Fig. 11] 30 Minute

4. 결론 및 추후 연구 방향

본 연구에서는 시간이 경과됨에 따라서 전경 이미지를 구성하는 객체들이 등장하는 화면 영역을 알아보고 이 영역이 플레이어가 마우스를 조작하는 영역과의 일치성을 알아보기 위한 연구이다. 이러한 일치성은 플레이어들이 전경 이미지를 구성하는 객체에 집중하고 있으며 화면 집중 영역을 파악하는데 도움을 준다.

연구를 위하여 OpenCV 프로그램을 이용하여 전경 이미지를 추출하는 프로그램을 구현하였다. 마우스 트래킹 프로그램은 프리웨어를 사용하였다. 연구에 사용된 게임은 리그오브레전드 게임이다.

연구 결과, 시간이 지날수록 전경 이미지를 구성하는 객체들은 처음에는 화면 중앙 영역에 많이 분포하다가 시간이 경과되면서 화면 중앙을 중심으로 등장하는 범위가 커짐을 확인할 수 있었다. 마우스 트래킹 프로그램의 경우에도 화면 중앙 영역에서 가장 많은 조작을 행했다. 이는 화면 중앙 영역이 플레이어의 관심을 가장 많이 끄는 화면 영역이라는 것을 알 수 있다. 시간이 지남에 따라 전경 이미지를 구성하는 객체들의 등장 영역은 화면 중앙 영역을 중심으로 커지는 패턴을 보이고 있다. 이는 플레이 시간이 지남에 따라 발생할 수 있는 단순 반복성을 감소시킬 수 있도록 플레이어의 공격 영역이나 방어 영역을 확대시킨 게임 기

획자의 의도로 판단된다.

추후 본 연구는 보다 많은 게임들에 대해 이러한 연구를 진행할 예정할 예정이다.

ACKNOWLEDGMENTS

This research is supported by 2014 Baekseok University Fund

REFERENCES

- [1] www.leagueoflegends.co.kr.
- [2] Yongsu Jeon, Heejin Park, Youngsub Yoon, Yunju Baek, "Design and Implementation of Multi Exposure Smart Vehicular Camera Applying Auto Exposure Control", The Journal of The Korean Institute of Communication Science, 42(1), pp.181-192 (12 pages), 2017.1,
- [3] Changick Kim, Jaeseung Ko, Ilkoo Ahn, Jaeho Lee, Kiwon Seo, Jae Hoon Kwon, Young Hun Joo, Yun Je Oh, "Studying the Viewers' Acceptability on the Image Resolutions and Assessing the ROI-Based Scheme for Mobile Displays", JOURNAL OF BROADCAST ENGINEERING, 11(3), pp.336-348, 2006.9.
- [4] Mi Ae Ko, Young Mo Kim, Kang Sik Koh, "Multimedia : Effective Compression of the Surveillance Video with Region of Interest", The KIPS Transactions : Part B, Vol.10 No.1, pp.85-102, 2003.
- [5] Min Seok Kwak, "Robust Object Tracking System Based on Face Detection", KIPS Tr. Software and Data Eng, Vol.6, No.1 pp.9~14, 2017.
- [6] Chanho Lee, and Dajun Ding. "An Adaptive Road ROI Determination Algorithm for Lane Detection", Journal of the Institute of Electronics Engineers of Korea, Vol.51, No.1, pp.116-125, 2014.
- [7] WonJun Kim, Changik Kim, "Area of

Interest Detection Technology and Application Examples", The Magazine of the IEEE, 39(2), pp.20-28, 2012.2.

[8] <http://iographica.com/>

[9] <http://rollstory.tistory.com>

[10] <http://www.opencv.org>

[11] <http://cfile4.uf.tistory.com/image>

이 먼 재(Lee, MyounJae)

약 력 : 2009.03- 백석대학교 정보통신학부 교수

관심분야 : 게임 엔진, 기능성 게임