

OCR(Optical Character Recognition)을 활용한 실시간 정보 제공 애플리케이션 - 게임에서의 활용방안을 중심으로 -

윤희창, 이승혁, 유승호*

부경대학교 컴퓨터인공지능공학부

{yhc1228, leesh7807}@pukyong.ac.kr, shyoo@pknu.ac.kr*

Real-time user helper application with OCR(Optical Character Recognition) on game

Yun Hui Chang, Lee Seung Hyeok, Seungho Yoo*

Pukyong University, Division of Computer Engineering and Artificial Intelligence

요약

본 논문은 사용자의 화면을 캡처하여 OCR을 통해 텍스트를 분리하는 것으로 사용자에게 편의를 제공하는 기술을 제안한다. 그리고 해당 애플리케이션을 개발하여 게임에 적용시켜 예시를 보인다. Tesseract와 OpenCV를 활용하여 사용자의 화면에서 문자열을 추출하였고, WebSocket을 사용하여 OCR 프로그램, 웹 페이지, 웹 서버간의 실시간 통신 환경을 구축하였다. 부족한 OCR성능을 해결하기 위해 문자열 매칭 알고리즘을 사용하였다. 웹 서버는 OCR 프로그램에게서 받은 정보를 문자열 매칭 알고리즘을 사용하여 추출한 문자열을 적절한 정보와 매칭시키고, 이를 통해 필요한 정보를 웹 페이지로 실시간으로 사용자에게 제공한다. 또한 Channel Layer를 사용하여 한 화면에 대한 정보 분석 결과를 여러 사용자에게 공유할 수 있도록 하였다.

I. 서론

게임과 동영상과 같이 텍스트 정보가 이미지의 형태로 전달되는 매체에서 텍스트 정보를 추출하여 활용하고자 한다. 사용자는 컴퓨터를 사용하면서 수많은 정보를 접하게 되고, 그중 텍스트 정보는 단연코 많은 부분을 차지한다. 그런데 공급자는 여러 이유로 텍스트 정보를 다른 형태로 전달하는 경우가 많고, 사용자가 이를 활용하기 위해서는 일일이 변환하는 과정을 거쳐야 한다. 이러한 이유로 사용자가 쉽게 텍스트 정보를 활용할 수 있게 돕는 애플리케이션을 개발하기로 했고, 이것을 많은 사람이 즐기는 게임에 적용하기로 했다. 특히, 여러 게임 중 '로스트아크'에 표시되는 정보 중 지역 정보를 활용하여 사용자에게 유용한 정보를 제공하는 애플리케이션을 만들기로 하였다. 지역의 현재 정보가 항상 화면에 표시되고, 숨겨진 요소가 존재하기에 텍스트 정보를 활용한다는 개념에 적합하다고 생각했다. 본 논문에서는 저자들이 제안하는 실시간 정보 제공 애플리케이션의 구현 방안 및 적용 기술 및 그 결과를 설명한다.

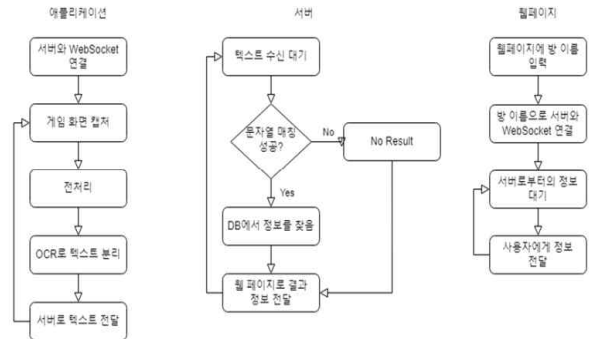
II. 본론

1. 애플리케이션구성

애플리케이션은 Optical Character Recognition (OCR) 프로그램과 서버 그리고 결과를 보여주는 웹페이지로 이루어져 있다. OCR프로그램은 사용자의 컴퓨터에서 게임의 화면을 주기적으로 캡처하고 이미지에서 텍스트를 분리해낸다. 분리해낸 텍스트를 주기적으로 서버에 전송한다. 서버는 텍스트 정보로부터 사용자에게 전달해야 할 정보를 식별한다. 이렇게 선택된 정보를 실시간으로 사용자의 웹페이지에 보여준다.

1.1 OCR 프로그램

사용자 화면의 특정 영역을 주기적으로 캡처하기 위해 Pillow 라이브러리를 사용하였고, 캡처한 이미지에서 텍스트를 분리해내기 위해 OCR 기술을 제공하는 오픈소스 Tesseract를 사용하였다. 사용자는 정보를 받아볼 방의 이름을 서버로 전달할 후 분리해낸 텍스트를 전달한다.



[그림1] 애플리케이션 흐름도

1.2 서버

서버는 Python 기반의 Django 웹 프레임워크를 통해 구현하였다. 애플리케이션을 구성하는 많은 부분이 Python 기반으로 구성되었기 때문에 Django를 선택하였다. 프로그램과 서버 사이의 통신은 실시간 통신에 유리한 WebSocket 프로토콜을 사용하였다. 사용자에게서 전달받는 텍스트에는 왜곡이 일부 섞여 있는데, 이를 건어내기 위해 유사도를 활용한 문자열 매칭 기능을 서버에 구현하였다. 서버는 매칭한 문자열을 바탕으로 DB에서 정보를 찾아 사용자가 사용 중인 방으로 전달한다.

1.3 웹 페이지

웹 페이지는 방 이름을 입력받으며, 입력받은 이름의 방으로 접속해 서버에서 전달되어 오는 실시간 정보를 볼 수 있다.

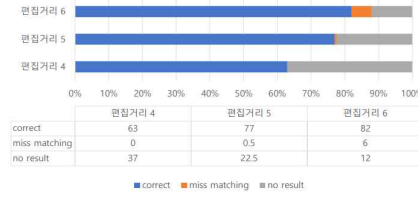
2. 화면으로부터의 정보 획득 방법

2.1 이미지 캡처

애플리케이션에서 화면에 표시된 정보를 획득하기 위하여 Pillow 라이브러리를 사용해 주기적으로 화면을 캡처하였으며, 애플리케이션 종류에 유연하게 대응하기 위하여 캡처 영역을 마우스로 지정할 수 있도록 구현



[그림 2] 화면 캡처 영역 지정



[그림 3] 편집 거리별 정확도



[그림 4] 애플리케이션 흐름

하였다. 캡처 영역 지정은 [그림 2]와 같이 실행 중인 애플리케이션 위에 박스를 그리는 것처럼 보이게 하여 사용자의 편의성을 도모하였다. 영역을 설정한 후에, 애플리케이션은 캡처를 1초에 한 번 실행하고 OCR을 통한 정보 추출 과정을 거쳐 서버로 정보를 전달한다.

2.2 전처리 및 OCR을 통한 글자 인식

이미지에서 텍스트를 분리하기 위해 Tesseract OCR을 이용하였다. Tesseract는 자체적으로 한국어 인식을 위한 모델을 가지고 있으나, 배경이 계속해서 변하는 게임의 특성상, Tesseract의 텍스트 인식률이 높지 않아 이를 높이기 위하여 OpenCV를 활용하여 전처리 과정을 추가하였다. 여러 이미지 전처리 기능들을 사용하였으나 게임상의 많은 시각적 효과 때문에 모음에 점이 추가된 것 같이 인식되는 현상으로 ‘ㅡ’가 ‘ㅏ’나 ‘ㅑ’로 인식되기도 하였다. 여러 시도 끝에 ‘로스트아크’ 게임에 최적화하는 방향을 선택함으로써 문제를 해결하기로 하였고, 게임 내에 표출되는 글자의 색 범위를 활용하여 실제로 사용할 수 있을 정도로 인식률을 높였다.

3. 지역 매칭 알고리즘

OCR 프로그램에서 전달받은 문자열에는 왜곡이 있어 정보 매칭의 정확도가 매우 떨어진다. 이를 보완하기 위하여 정규식 및 레벤슈타인 거리 알고리즘을 활용하였다.

3.1 레벤슈타인 거리

레벤슈타인 거리 알고리즘은 두 개의 문자열이 같아지기 위한 최소 수정 연산횟수를 구하는 알고리즘이다[1]. 여기서 수정 연산이란 추가와 대체, 삭제를 의미한다. 두 문자열 간에 수정 연산이 생길 때마다 연산 비용 1을 부여하여 총 연산 비용을 문자열 간의 거리를 의미하는 편집 거리로 계산하였다. 본 애플리케이션에서는 기준 안에 들어오는 지역 중 편집 거리가 가장 짧은 지역을 선택하여 보여주며, 최적의 기준 편집 거리를 파악하기 위해 184초의 게임 플레이 영상을 1초 주기로 캡처하여 테스트해 보았고, 그 결과는 [그림 3]과 같이 나타내었다. 기준 편집 거리를 크게 설정하면 No Result의 비율이 감소하지만 잘못된 매칭에 대한 위험부담이 증가하고, 반대로 작게 설정하면 잘못된 매칭은 없어지지만, 결과가 나오지 않는 경우가 많아진다. 여러 편집 거리에 대한 비교를 통하여 잘못된 정보를 최대한 제공하지 않는 선에서 가장 좋은 성능을 보여준 5를 기준 편집 거리로 선택하였다.

3.2 매칭 인식률을 높이기 위한 추가적인 기법 적용

문자열 매칭시 ‘ㅎ’과 ‘ㅇ’, ‘ㅂ’과 ‘ㅍ’ 같이 단순 자음 하나의 오타자만으로 편집 거리가 늘어나는 경우가 많았다. 이를 완화하기 위해 Python의 jamo 라이브러리를 사용하여 자음과 모음을 나누어 더 세밀한 문자열 유사도 비교를 수행하여 정확도를 높였다.

Tesseract가 문자를 인식하는 과정에서 불필요한 띄어쓰기를 만들거나 한글을 특수문자로 인식하는 현상이 발생하였다. 이러한 문제를 해결하기 위하여 정규식을 사용하여 한글과 숫자만을 입력으로 받아들이도록 구현하여 인식 정확도가 더욱 향상됐다.

4. 실시간 반응 웹페이지

지역 매칭 결과를 주기적으로 사용자에게 받아서 사용자에게 실시간으로 정보를 전달하는 웹 애플리케이션을 구현하고자 한다. 실시간 웹 애플리케이션 구현을 위해 Django, WebSocket,, Redis를 사용하였다.

4.1 웹의 구조 및 통신 방식 선정

입장할 방 이름을 입력받는 페이지, 그리고 방에 입장하여 실시간으로 로그 및 정보를 띄워주는 room페이지로 총 2개의 페이지를 제작하였다. 사용자가 웹페이지로 들어가게 되면 방 이름을 입력받는 페이지[그림 4]로 다이렉트되고 사용자가 방 이름을 입력하면 URL을 기반으로 해당 방에 해당하는 페이지를 보여주며, 방 이름을 WebSocket connection의 group 이름으로 하여, 같은 방이름을 사용하는 사용자들과 연결한다.

4.2 비동기 통신 및 Channel Layer

웹상에서 비동기 통신을 하는 방법은 여러 가지이다. 그 중 AJAX(Asynchronous Java script), WebSocket등이 대표적이다. 관련 연구[2]에 따르면 AJAX 비동기 통신 기술과 WebSocket은 5배에서 10배가량의 지연시간 차이가 난다. 웹서버, OCR 프로그램, 사용자 간의 실시간 통신이 필요하므로 빠른 속도의 WebSocket을 사용하였으며, WebSocket connection에 다중 사용자 간의 통신을 구현하기 위해 Channel Layer를 사용하였다. Channel layer를 사용할 때 각 연결에 group 이름을 지정하여 같은 group 이름을 가진 연결끼리 서로 통신할 수 있게 하였다. 이를 통해 웹 서버와 사용자의 1대1 연결만이 아니라 웹 서버와 다중 사용자 간의 연결을 가능하게 하였다.

III. 결론

본 논문에서 OCR 기술을 통해 텍스트를 분류해 내고 이를 통해 사용자가 편리하게 게임을 할 수 있도록 돕는 애플리케이션을 구현하였으며, 구현 결과는 [그림 4]와 같이 나타내었다. 이미지에서 텍스트를 분리해 낸다는 특징 아래, 플랫폼에 제약 없이 다양한 방향으로 확장할 수 있다는 점이 강점이며, 이러한 특징을 기반으로 데이터를 보충하여 여러 게임으로 확장이 가능하고, 기능을 추가함에 따라 여러 추가적인 가치를 제공할 수 있다. 그리고, OCR 기술의 정확도를 올리고 데이터를 결과와 매칭시킨다면, 훨씬 더 강력한 애플리케이션이 될 수 있을 것으로 기대한다.

ACKNOWLEDGEMENT

이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. RS-2023-00213336).

참고 문헌

[1] Levenshtein, "Binary codes capable of correcting deletions, insertions, and reversals," Soviet Physics Doklady, Vol.10, pp.707-710, 1966
 [2] Seung-Jae Oh, "Comparative Analysis of Asynchronous Communication Technology in Web Data Communication" Journal of Multimedia Services Convergent with Art, Humanities, and Sociology pp.419-426, 2017