

Finger Gesture와 Voice Recognition을 활용한 IoT 통합 제어 웹 플랫폼

강진형, *김한주, †김동호
skyful51g@gmail.com, *hanju98@seoultech.ac.kr †dongho.kim@seoultech.ac.kr

IoT Multi Control Platform by Finger Gesture and Voice Recognition

Jinhyeong Kang, *Hanju Kim, †Dong Ho Kim
Seoul National University of Science and Technology

요약

증강현실로 날씨, 뉴스 요약 등이 제공되거나 AI 비서 기능을 제공하는 스마트 미러(smart mirror)가 개발되고 있다. 본 작품에서는 IoT 통합제어, 뉴스 요약 및 날씨 정보 제공 등의 서비스를 하나의 웹 플랫폼으로 구축하고 이를 손가락 제스처 및 음성 명령으로 제어하는 것을 제안하고 구현하였다. 본 작품에서는 음성 인식을 통해 IoT 서비스를 직관적으로 이용할 수 있게끔 설계하여 사용자의 편의성을 높였으며, 디바이스를 직접 터치하는 방식이 아닌 finger gesture로 제어하는 방식을 채택해, 디바이스 유지 보수 및 위생 문제를 해결하였다. 단순 IoT 통합 제어 기능뿐만 아니라 다양한 콘텐츠 및 기능을 제공함으로써 통합 플랫폼의 기능을 수행할 수 있도록 하였다. 뉴스 홈페이지에서 Crawling한 뉴스를 text rank 알고리즘을 이용, 자동으로 요약하는 기능과, 사용자의 IP를 기반으로 위도와 경도를 추론, 해당 지역의 일기 예보 정보를 표현해 주는 등 단순 IoT 제어 플랫폼이 아닌, 통합 플랫폼의 기능을 다하도록 설계하였다. 이처럼 다양한 정보를 압축해서 사용자가 편하게 볼 수 있도록 제공하며, 직관적인 two track 제어 방식을 채택, 사용 대상의 편의성을 증대시켜 본 프로젝트는 기존 프로젝트보다 사용자에게 더 나은 사용 경험을 제공할 것이다.

1. 작품의 제작 동기

최근 AI 비서 기능과 증강현실 서비스를 제공하는 스마트 미러(smart mirror)가 개발되어 날씨, 뉴스 요약, 개인 일정 관리 등 다양한 기능을 제공하고 있다. 스마트 미러에는 사물 인터넷(IoT: Internet of Things) 기능을 내장하여 인터넷에 연결하는 무선 통신을 포함한다.

최근 가전 제품과 실내의 가구들에 센서 및 모터를 장착시키고 기존 기능을 새롭게 통합시켜 새로운 가치를 창출 하는 방향으로 IoT 기술을 사용한다. 이 때 IoT 제어를 위해 하나의 플랫폼에서 여러 IoT를 제어하는 Control Center Platform이 필요하다. 이 Control Center Platform에서 가장 중요한 것은 플랫폼이 정보를 표현하는 방식과 IoT 기능을 제어하는 방식이다.

기존 IoT 기기제어를 위한 플랫폼은 보통 핸드폰 app등을 이용한 손가락 touch 방식으로 개발되는 탓에, 사용법에 익숙치 못한 노약자들이 어려움을 겪고 있다. 따라서 더 직관적인 인터페이스와 제어 방법을 설계하여 디지털 소외계층이 쉽게 사용할 수 있도록 구현할 필요가 있다. 본 프로젝트에서는 이와 같은 어려움을 해결하기 위해 버튼 터치 방식 대신 사용자의 목소리를 인식하여 정해진 기능을 수행하도록 구현하였다.

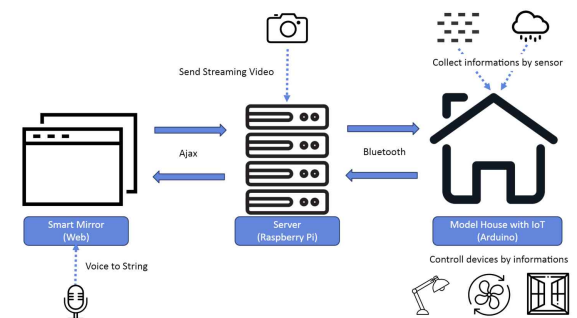
또한 Mirror라는 하드웨어에서 구동되는 플랫폼인 만큼 손가락으로 직접 터치하여 제어할 경우, 자국이 남아 하드웨어 유지 보수에 어려움을 겪는다. 특히 코로나 19로 인해 다수가 사용하는 장치에 접촉하는 것에 대한 거부감과 위생 문제에 대한 경각심이 늘어나고 있다. 이와 같은 어려움을 해결하기 위해 디바이스에 직접 터치할 필요 없이 거울에서 finger gesture를 취하면 디바이스가 이를 인식하고 그에 따른 적절한 동작을 수행하도록 구현하였다. 이처럼 손가락 제스처를 인식하여 플랫폼을 제어하는 방식, 목소리를 인식하여 플랫폼을 제어하는 방식. 총 두

가지 종류의 제어 방법을 병행하여 사용하므로 기존의 방식보다 더 직관적인 방식으로 동작을 제어할 수 있게 될 것이다.

또한, 단순히 IoT를 제어하는 플랫폼의 기능만 수행하는 것이 아니라, text rank 알고리즘을 이용한 뉴스 요약 기능, 위치에 기반을 둔 날씨 정보 표현, IoT 센서간 연동을 통한 실내 가구 동작 자동화 등 복합 플랫폼으로써 기능을 수행할 수 있도록 본 프로젝트를 설계하였다.

2. 작품의 설계 및 구현

가. 전체 시스템 구조도



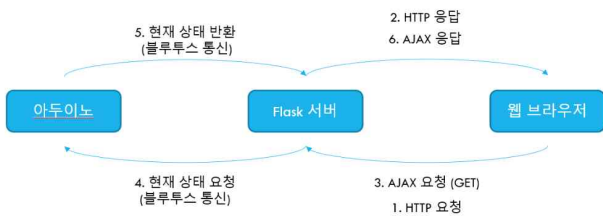
[그림 1] 전체 구조도

본 시스템의 전체 구조는 [그림 1]과 같다. 하드웨어는 크게 세 가지 부분으로 나뉜다. 1. UI/UX를 표출하기 위한 모니터, 2. 웹 페이지를 로드하고 IoT와 웹캠, 마이크 간 통신을 담당하는 라즈베리파이, 3. 각종 센서와 제어부를 아두이노로 제어한 모델하우스(아두이노)이다.

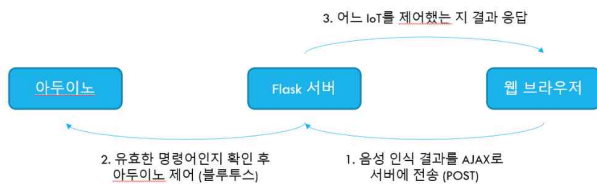
나. ajax 통신

기본적인 http 프로토콜에서 사용자가 request를 보내고 Server에서 응답을 보내면 이어졌던 연결이 끊긴다. 따라서 웹사이트의 내용을 갱신할 경우 전체 http 프로토콜을 다시 반복하여 전체 웹페이지를 받아야 한다.

하지만 IoT 각 모듈이 작동할 때마다 상태를 표시하기 위해 웹페이지를 갱신하게 되면, 성능에 큰 영향을 끼친다. 웹페이지 일부분을 업데이트 하기 위해 페이지 전체를 다시 로드하는데, 이는 엄청난 자원낭비이다. 이를 방지하고자 ajax 통신을 사용하였다. ajax 통신은 전체 html의 갱신이 아닌 일부만 갱신할 수 있도록 XML HttpRequest 객체를 이용한다. Json 이나 XML 형태로 서버에서 필요한 부분의 데이터만 받아 갱신하기 때문에 HTML 전체를 리로드(reload) 하는 기존의 방식보다 효율적이다. IoT 모듈과 웹 플랫폼 사이 통신 프로토콜은 크게 처음 시스템을 구동할 때와 IoT 상태가 업데이트 되었을 때 2가지로 나뉜다.



[그림 2] 시스템 구동 시 IoT 통신 프로토콜



[그림 3] 명령어에 따른 IoT 통신 프로토콜

처음 시스템을 구동할 경우 IoT 초기값을 받아오기 위한 동작은 그림 2와 같다. 처음 웹플랫폼을 서버에 요청할 때, 서버는 블루투스 통신을 이용해 IoT 센서의 각 현재 상태를 요청한다. 요청받은 아두이노 모듈은 각 센서의 현재 값을 읽어와 반환하고 서버에서는 html에 이 정보들을 넣어 동적으로 표출한다.

이후 각 명령어에 따라서 IoT 그림3과 같다. 웹 플랫폼에서 인식한 명령어를 서버에 post 방식으로 전송한다. flask 서버에서는 일련의 과정(라 항목 참고)을 거쳐 유효한 명령어인지를 확인한다. 유효한 명령어인지를 검증했으면 블루투스 통신을 통해 아두이노 모듈에 전송하고 아두이노는 각 명령에 1:1 대응된 동작을 실행한다.

다. 손가락 제스처 제어

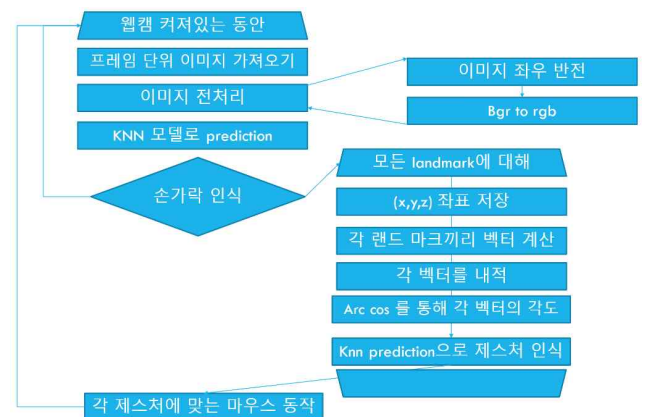
1) knn 모델 학습



[그림 4] mediapipe에서 사용하는 손가락 joint 위치[4]

손가락 제스처를 인식하기 위해 KNN를 분류 사용했다. 사용한 데이터 셋은 손가락으로 1,2,3을 가리키는 3종류의 직철 촬영한 사진을 각각 300장 사용하였다. 총 900장의 사진을 Media Pipe를 이용해 [그림4]와 같이 21개의 JOINT 좌표를 찾은 후 N번 joint와 n+1번 joint의 벡터를 계산하여 총 20개의 손가락 마디 벡터를 생성했다. 이후 각 벡터의 내적을 계산하여 이를 특징으로 갖는 Ground Truth를 생성한 뒤 숫자 1을 가리키는 사진은 1, 숫자 2를 가리키는 사진은 2, 숫자 3을 가리키는 사진을 3으로 labeling하여 knn 모델을 생성했다.

2) 손가락 제스처 제어



[그림 5] Finger Gesture Recognition 동작도

라즈베리파이에는 연결된 웹캠을 통해, 계속해서 손을 찾으려고 한다. 손을 찾을 경우, [그림 4]와 같이 한 손에 대해 21개의 JOINT의 위치 값을 반환한다. 21개의 joint 좌표를 이용하여 n번 joint와 n+1번 joint 사이의 벡터 총 20개의 손가락 마디 벡터를 생성한다. 생성한 벡터를 서로 내적시켜 각 벡터끼리의 각도를 구한 후 이를 미리 학습시켜놓았던 knn모델에 prediction하여 기 학습된 제스처인지 여부를 인식한다. 이때 손가락 제스처의 라벨이 1, 2, 3으로 반환될 경우 사이드바의 하이퍼링크를 작동시켜 해당 페이지로 이동한다.

라. Jaccard Coefficient를 활용한 음성 인식

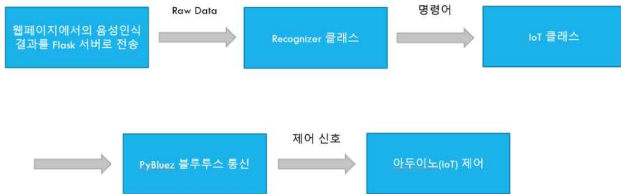
$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

[그림 6] Jaccard Coefficient

음성인식은 Javascript 라이브러리인 Anyang을 사용했다. Anyang 라이브러리에서 제공하는 함수를 통해 마이크 입력을 받는다. 받은 음성 정보를 텍스트로 추출하여 반환한다. 이 때, 일부 음성의 경우 발음이 애매모호하기 때문에 저조한 인식률을 보여주는 데, 이를 보완하고자 Intersection Over Union방법론을 사용했다. Intersection Over Union은 집합간의 유사도를 정량적으로 나타내는 방법이다. [그림3]과 같이 분모에는 집합간의 공통 부분, 분자에는 전체 집합 크기를 반영하여 집합 간의 유사도를 측정한다.

Jaccard Index를 text 포맷으로 적용한 방법은 다음과 같다. anyang을 이용하여 음성을 인식한 뒤 문자열 형태로 반환한다. 문장을

음절 단위로 분해한 뒤 list에 넣는다. 이를 set 자료형으로 바꾼 뒤 미리 등록해놓은 명령어 음절 set과 합집합과 교집합을 계산하여 각 명령어에 대한 jaccard index를 계산한다. 이후 Threshold 값을 넘는 jaccard index만 해당 명령어로 인식하여 명령어를 수행하게 된다. 이와 같은 방식으로 음성 인식 성능 저하를 해결하였다.



[그림 7] Voice Recognition 동작도

이후 최종 반환된 명령어에 따라 제어 신호를 PyBluez 블루투스 통신으로 아두이노에 전달, 아두이노는 수신한 신호에 맞는 동작 수행한다.(켜기, 끄기) 그 직후 아두이노는 현재 연결된 기기의 동작 여부를 블루투스 통신으로 서버에 전달해 웹 페이지에 기기 연결 상태를 표시할 수 있도록 설계하였다.

마. IoT 센서 연동을 통한 실내 가구 동작 자동화 및 상태 표시

실내 공기질 센서, 미세먼지 센서, 빔방울 센서를 통해 실내 가구 동작을 자동화한다. 3개의 센서는 실내 공기질 센서, 미세먼지 센서, 빔방울 센서 순으로 우선순위를 가지고 동작한다. 실내 공기질 센서가 활성화되면 즉시 창문을 개방하고 환풍기를 가동하며 센서가 비활성화되기 전까지 사용자가 창문과 음성 명령으로 환풍기를 조작하는 것을 막는다. 미세먼지 센서와 빔방울 센서는 동일한 우선순위를 가지며 두 센서 중 하나라도 활성화되면 창문을 닫고 사용자가 음성 명령으로 창문을 조작하는 것을 막는다. 각 센서의 활성화 여부는 각 센서에 표시등을 연결하고 플랫폼 UI/UX에서도 실시간으로 표시하게끔 반영하여 사용자가 쉽게 센서 활성화 여부를 인식할 수 있다.

바. TextRank알고리즘과 Web Crawling을 이용한 뉴스 요약

파이프 라인은 다음과 같다. Beautiful Soup에서 뉴스 사이트 메인 페이지를 Crawling 해오면 html을 parsing하여 각 html tag 별로 정보를 저장한다. 뉴스 제목, 사진, 하이퍼링크, 본문 내용을 dictionary 형태로 저장한다. 각 본문 내용을 text rank 알고리즘을 이용하여 1줄로 요약하고 웹페이지 상에 표현한다.

textrank 알고리즘은 문장 요약을 위해 널리 사용되는 알고리즘이다. 작동 원리는 다음과 같다. 핵심 단어를 선택하기 위해 단어간의 co-occurrence graph를 생성한다. 이후 핵심 문장을 생성하기 위해 문장 간 유사도를 기반으로 sentence similarity graph를 생성한다. 이후 그래프의 각 마디의 랭킹을 계산하고, 랭킹이 높은 순서대로 키워드와 핵심문장이 된다.

크롤링해 온 본문의 내용을 문장 단위로 분리한다. 분리한 문장을 형태소 단위로 나눈 후 품사 태깅을 통해 명사를 추출한다. 후에 TF-IDF 모델을 생성한다. TF-IDF는 특정 문장 내에서 단어의 중요도를 나타내는 통계적 수치이다. TF-IDF 모델링을 수행 한 후 그래프를 생성하기 위해 Sentence - Term Matrix를 생성한다. 전치행렬을 구한 뒤 서로 곱해서 Correlation Matrix를 생성한다.

$$TR(V_i) = (1 - d) + d * \sum_{V_j \in In(V_i)} \frac{W_{ji}}{\sum_{V_k \in Out(V_j)} W_{jk}} TR(V_j)$$

[그림 8] TR 계산식

이를 인접 매트릭스처럼 바라본다면, 문장과 문장, 단어와 단어 사이 가중치 그래프로 나타낼 수 있다. 마지막으로 이 그래프에 [그림] Text Rank 알고리즘을 적용한다. 알고리즘을 통해 Ranking값이 높은 순으로 정렬한 뒤 요약된 문장을 출력한다.

사. 사용자 위치 정보를 기반으로한 날씨 예측 정보 표현

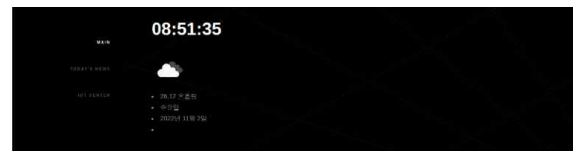
위치 정보를 알아내기 위해 사용자의 IP 주소를 먼저 알아낸다. IP 주소를 알아내기 위해 파이썬의 socket 모듈을 사용한다. socket 모듈을 이용하여 네트워크 연결을 만들고, 네트워크를 통해 메시지를 보낸다. 이를 이용하여 해당 코드가 실행되고 있는 시스템의 호스트 이름(IP)을 반환한다. 이를 ip2location을 이용, 해당 ip주소의 위도 경도 값을 반환한 후 openweather에서 제공하는 일기예보 정보를 가져온다.

3. 작품의 구현결과

가. UI/UX

웹페이지는 1. 시간 및 기상 예보 섹션, 2. 뉴스 섹션, 3. IoT 통합 제어 섹션 총 3가지 섹션과 사이드바로 이루어져있다.

1) 시간 및 기상 예보 섹션



[그림 9] 시간 및 기상예보 섹션 화면

현재 위치를 기반으로 하여 시간과 날씨, 날씨를 출력한다. 날씨 code에 맞춰 미리 설정해놓은 날씨 아이콘이 반영되어 직관적으로 알 수 있다.

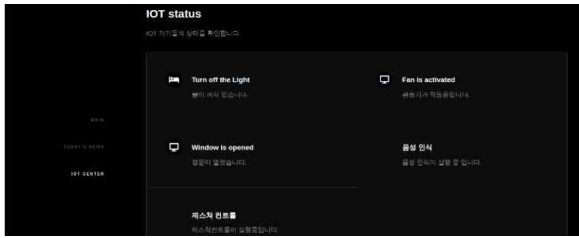
2). 뉴스 섹션



[그림 10] 뉴스 요약 섹션 화면

크롤링해온 본문 내용을 TextRank알고리즘에 기반해 요약을 진행한다. 헤드라인 밑에 요약 내용을 출력한다.

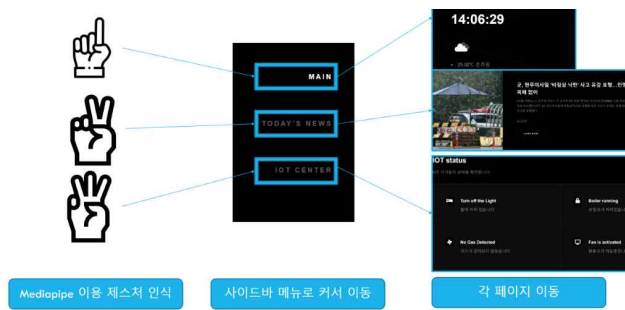
3). IoT 통합 제어 섹션



[그림 11] IoT 통합 제어 섹션 화면

IoT 제어를 하여 각 센서의 상태가 변하거나, 동작을 했을 때 직관적으로 인식할 수 있도록 아이콘의 색깔이나 문구가 변하도록 동적기능을 추가했다.

4). 사이드바



[그림 12] Sidebar workflow

웹 페이지 좌측에 Side Bar를 구현하여 각 섹션으로의 이동이 자유롭게 했다. 해당 Side Bar의 하이퍼링크를 이용하여 손가락 제스처와 일대일 대응을 시켜놓았다. 따라서, 손가락 제스처를 인식하여 각 섹션을 자유롭게 넘나들 수 있다.

나. 모델하우스



[그림 13] 모델 하우스

실제 집 내부를 가정하고 원목으로 모델하우스 형태를 제작하였다. 이후 아두이노 센서를 사용하여 IoT 센서부와 가구 동작부를 구현했다. 센서는 미세먼지 감지 센서, 빛방울 센서, 공기질 센서 3종류를 사용했다. 또한 모터를 이용하여 창문과 환풍기의 동작을 구현했다.

다. Smart Mirror



[그림 14] Smart Mirror

가정에서 쉽게 이용할 수 있게 Mirror 형태의 하드웨어를 제작했다. 시중에서 판매하는 편광필름과 Dell P2317HT 23인치 모니터를 사용해 디스플레이를 볼 수 있으면서 거울의 역할을 동시에 수행할 수 있도록 제작했다. 라즈베리 파이와 연결하여 웹 플랫폼 동작부의 기능을 수행할 수 있다.

웹 서버로 작동하는 라즈베리파이의 사양은 1.2GHz Quad-Core ARM Cortex-A53 / 802.11 b/g/n Wireless LAN + Bluetooth 4.1 (Bluetooth Classic and LE) / 1GB RAM / 3ModelB이다.

4. 작품의 기대효과

IoT 센서의 기능과 이를 제어하는 하드웨어의 성능이 향상되면서 이를 어떤 방식으로 제어하고, 어떤 방식으로 표현할지가 중요해졌다. 본 작품은 손가락 제스처를 인식하여 플랫폼을 제어하는 방식, 목소리를 인식하여 플랫폼을 제어하는 방식. 총 두가지 종류의 제어 방법을 병행하여 사용하므로 기존의 방식보다 더 직관적인 방식으로 동작을 제어할 수 있다.

직접 디바이스에 터치하는 방식이 아닌 손가락 제스처 인식과 음성 명령 제어 방식으로 동작한다. 따라서, 디바이스 위생 문제와 유지 보수 문제를 모두 해결할 수 있다.

또한 단순히 IoT를 제어하는 플랫폼의 기능만 수행하는 것이 아닌, text rank 알고리즘을 이용한 뉴스 요약 기능, 위치에 기반을 둔 날씨 정보 표현, IoT 센서간 연동을 통한 실내 가구 동작 자동화 등 복합 플랫폼으로써 기능하여 더 나은 사용자 경험을 제공할 것이다.

Acknowledgement

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 정보통신·방송 연구개발사업의 일환으로 수행하였음. [IITP-2021-0-01816, 메타버스 자율트윈 핵심기술 연구]

5. 참고 문헌

[1] Mallick, Chirantana, et al. "Graph-based text summarization using modified TextRank." *Soft computing in data analytics*. Springer, Singapore, 2019. 137-146.
 [2] Garrett, Jesse James. "Ajax: A new approach to web applications." (2005).
 [3] Zhang, Fan, et al. "Mediapipe hands: On-device real-time hand tracking." *arXiv preprint arXiv:2006.10214* (2020).
 [4]<https://googlec.github.io/mediapipe/solutins/hands.html>