

영상처리와 딥러닝을 이용한 식단 관리 및 추천 애플리케이션 설계 및 구현

김현지, 이주영, 정연희
 인하대학교 정보통신공학과
 gusw11414@gmail.com, leeee9860@gmail.com, 12181833@inha.edu

Diet Management and Recommended Applications Using Image Processing and Deep Learning

Hyeon-Ji Kim, Ju-Young Lee, Yeon-Hee Jeong
 Dept. of Information and Communication, Inha University

요 약

‘100 세 시대’라는 말이 오가는 시대가 도래했다. 길어진 수명에 맞춰 건강한 식단 및 식습관에 대한 관심 또한 증가하였다. 그러나 혼자서 식단에 대한 정보를 찾고, 기록하고 영양성분을 관리하는 것에는 한계가 있다. 이에 본 팀은 손쉽게 찍어 올린 식단의 사진만으로 사용자의 식단을 분석하고 관리해주며 식습관을 교정해주는 애플리케이션을 설계 및 구현하여 제안한다.

1. 서론

1.1 연구배경 및 필요성

최근 헬스케어에 대한 대중의 관심이 높게 형성되고 있다. 그에 맞춰 건강한 식단에 대한 관심도 증가하였다. 그리고 헬스케어에 대한 유형 중 식단 관리를 개인 스스로하고자 하는 움직임을 SNS 검색어들의 인기 순위에서도 열 볼 수 있다. 그러나 개인이 식단 하나하나에 대한 정보를 찾고, 기록하고 계산하는 것에는 한계가 있다.

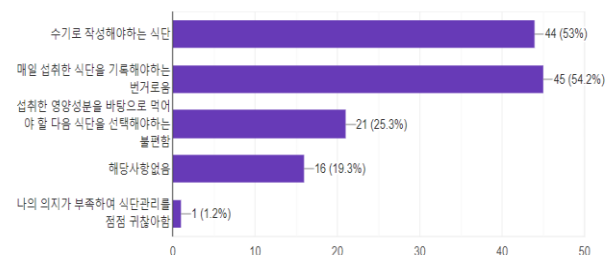
현재 식단 관리 애플리케이션은 구글 스토어 기준 25 만여개, 앱스토어 기준 10 만여개가 등록되어있다. 그러나 이를 식단 관리 애플리케이션의 총 다운로드 수와 비교하여 보면 굉장히 적은 수임을 알 수 있다. 즉, 이는 역설적으로 식단 관리 애플리케이션을 사용하는 사용자가 하나의 애플리케이션에 정착하지 못하고, 여러 애플리케이션을 단기적으로 사용해보고 있음을 설명한다.

식단 관리 애플리케이션의 핵심은 사용의 편의에서 오는 장기적인 앱 사용 및 식단 관리이다. 그러나 현재 식단 관리 애플리케이션 시장은 app 의 디자인에만 차별화가 있을 뿐 식단 관리의 편의성에 우열을 가릴 수 없을 만큼 식단 관리 기능이 평준화되어 있다. 실제 식단 관리 애플리케이션을 사용한 이용자들을 토대로 설문조사를 진행하였고, 그 결과인 그림 1 을 보게 되면 매일 식단을 수기로 작성하는 것에 대한 불편함, 섭취한 영양성분을 토대로 다음 식단을

고려하는 것에 대한 한계점이 있는 것을 볼 수 있다.

식단 관리 애플리케이션을 사용 할때의 불편함

응답 83개



(그림 1) 설문조사 결과

1.2 연구 방법 및 내용

이에 대한 해결 방안으로 영상처리와 딥러닝을 이용한 식단 관리 및 추천 애플리케이션을 고안해냈다. 본 논문에서 설계한 영상 및 추천 애플리케이션은 수기로 영양성분을 등록할 필요없이 사용자가 섭취한 식단 사진을 업로드하게 되면 학습시킨 가중치를 활용하여 음식을 라벨링하고 이를 바탕으로 영양성분 DB 에 접근하여 자동으로 식단을 분석해준다. 또한 사용자가 이전에 섭취한 영양성분을 기반으로 콘텐츠 기반 필터링을 통해 다음에 섭취하면 좋을 영양성분이 있는 음식을 추천해준다. 때문에 영양성분이 불균형한 식단을 섭취하는 등의 잘못된 식습관을 교정해주는 효과를 기대해 볼 수 있다. 그리고 커뮤니티 기능, 섭취 영양소 데이터

분석 지표 제공 등의 기능을 식단 관리에 대한 목표 의식 및 사용자의 애플리케이션 활용도 증진에 목적을 두고 추가로 구현했다.

1. 관련 연구

2.1 OPENCV

OPENCV(Open Source Computer Vision)는 오픈소스로 개발된 컴퓨터 비전 라이브러리이다. 이를 활용하여 YOLO 로 학습시킨 food detection 가중치를 linux 환경이 아닌 Python 환경에서 구동되도록 구성하였고 앱 이용자가 업로드한 사진 품질에 따른 오차 범위를 줄이기 위한 전처리 과정에 활용하였다. Unsharp mask filtering, histogram-equalization 등을 적용하고 confidence 를 비교하여 음식 사진에 최적화된 전처리 과정을 선택하였다.

최종적으로 전처리에는 가장 높은 개선 효과를 보인 histogram-equalization 을 적용하게 되었다. 자세한 구현 과정은 본 논문의 3.2 에서 기술하였다.

2.2 YOLO

YOLO(You Only Look Once: Unified, Real-Time Object Detection)는 딥러닝 기반 FAST 객체 탐색 기법이다. YOLO 는 Real-time 에서 빠르고 높은 탐지 능력을 갖고 있으며 loss 조절로 인하여 배경에서의 false positive 를 다른 이미지 학습 방법보다 영향을 조금 받는 특성이 있어서 food image 를 학습시키는 방법으로 선택했다.

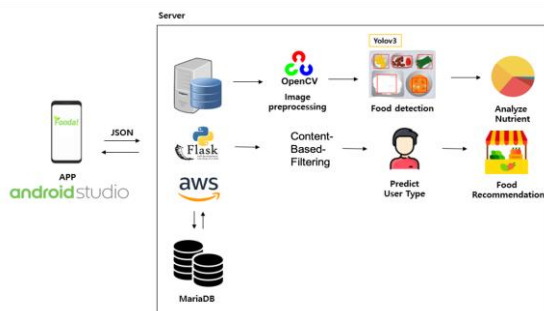
2.3 콘텐츠 기반 필터링

Clustering analysis, Artificial Neural Network, TF-IDF 등의 기술을 활용하여 콘텐츠의 내용을 분석하는 방법이다. 여러 항목에서 프로파일을 구성하는 것이 어려워, 다양한 형식의 항목을 추천해야하는 경우 어려움이 있다. 그러나 협업 필터링에서 발생하는 Cold-start 문제를 쉽게 해결할 수 있다는 장점이 있다.

2. 설계 및 구현

3.1 시스템 구조도

그림 2 는 시스템의 구성을 도식화한 것이다.



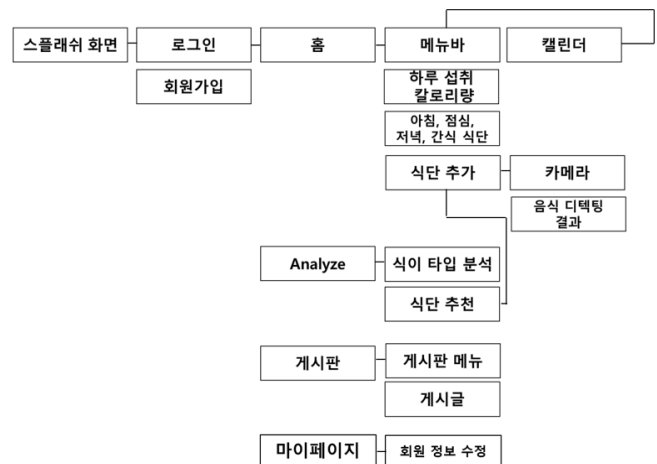
(그림 2) 시스템 도식화

Android Studio 를 통해 구현된 Frontend 는 사용자가 필요한 정보를 요청하며, Amazon AWS Server 는 Rest API 로 구현된 Flask 를 통해 자원을 제공한다. Database 는 MariaDB 를 사용하였다.

Food Detect 를 위해 Front-end 에서 이미지를 보내면, opencv 를 통해 이미지를 가공한 뒤, yolov3 를 통해 음식을 감지한다. 이후 감지한 음식을 바탕으로 DB 에서 음식의 영양분을 찾아 client 에 전송한다.

Food Recommendation 을 위해 DB 에 누적된 사용자의 식단을 바탕으로 사용자에게 부족한 영양분을 충당할 수 있는 식단을 추천해 준다.

그림 3 은 본 논문의 어플리케이션 핵심 기능을 나타낸 시스템 구조도이다.



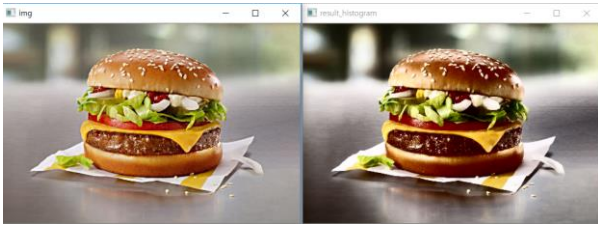
(그림 3) 시스템 구조도

어플리케이션의 구동 과정은 3.6 에서 자세하게 설명한다.

3.2 OPENCV 를 이용한 Image 전처리 구현

앞서 말했듯 다양한 필터들을 적용해 보았으나 효과가 미미하거나 side-effect 가 발생하여 오히려 error 를 범하는 것을 확인하였고, 최종적으로 histogram-equalization 을 적용하여 빛의 강도에 대한 Contrast 를 조정해주는 것으로 전처리 구현을 결정했다. color 이미지의 histogram-equalization 은 HSV 기반 histogram-equalization 과 YCrCb 기반 histogram-equalization 이 있다. HSV 를 기반한 histogram-equalization 을 진행한 결과 RGB 의 색상을 HSV 로 변환하고 명도(V) 성분을 histogram equalization 시키고 RGB 로 다시 변환하는 과정에서 이전 이미지의 색감을 손상시키는 것을 확인하였다. HSV 기반 histogram equalization 을 사용하게 되면 이미지의 색감에 따라 민감하게 반응하는 food detection 에 큰 오차를 생성할 수 있다. 따라서 HSV 기반이 아닌 YCrCb 기반 histogram equalization 을 적용하여 색감 변화 없이 밝기의 범위만 조절하도록

전처리를 구현하였다. 아래의 그림 4 를 통해 전처리 전과 후의 차이를 확인할 수 있다.



(그림 4) Histogram Equalization 적용

3.3 YOLO 를 이용한 food detection 구현

UEC FOOD 100 이라는 데이터셋을 기반으로 Yolo 로 이미지 학습을 수행하였다. UEC FOOD 100 은 총 100 가지 class 로 구성하였으며 총 945MB 의 이미지로 구성되어있다. 이때 컨볼루션은 Imagenet 에서 사전 훈련된 컨볼루션을 사용하였다. 학습시킨 결과 2D 이미지 및 webcam 에서도 높은 detection 룰로 음식을 detection 하는 것을 확인할 수 있었다. 그림 5 는 이를 음식을 detection 한 결과 사진이다.



(그림 5) food detection 결과 이미지

3.4 콘텐츠 기반 필터링 식단 추천시스템 설계

<표 1> Rating Table

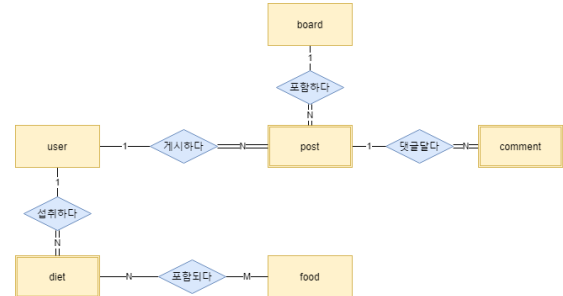
userid	foodid	rating	timestamp
1	1	4.0	964982703
1	3	4.0	964981247
1	6	4.0	964982224
1	47	5.0	964983815
1	50	5.0	964982931
1	70	3.0	964982400
1	101	5.0	964980868
1	110	4.0	964982176
1	151	5.0	964984041
1	157	5.0	964984100
1	163	5.0	964983650
1	216	5.0	964981208
1	223	3.0	964980985
1	231	5.0	964981179
1	235	4.0	964980908
1	260	5.0	964981680

이는 크게 3 단계로 나뉘서 설계하였다. 첫째, 유저가 섭취한 음식의 영양소 분석. 둘째, 영양소에 따른 rating table 구현. 셋째, Rating table 에 기반한 음식 추천. 세 가지 과정을 거쳤다. Pearson 유사도 및 kdd 를 활용하여 상관계수를 구하고, 이를 통해 추천해준다. 이러한 과정을 거치면 다음과 같은 결과 값이 출력된다. 다음 사진은 이전 식단에 지방이 많이 포함된 음식을 먹었던 사람에게 다음 식단을 추천해준 결과 값이다.

[(4.833342600369542, 'bibimbap'),
(1.6251238503032275, 'chicken-n-egg-on-rice')]

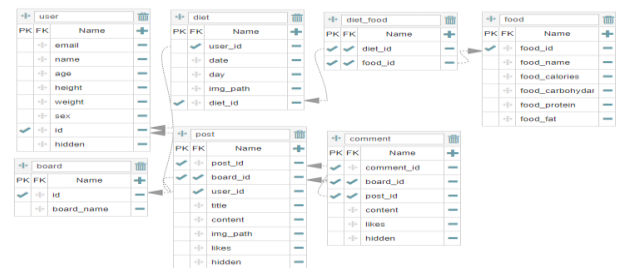
3.5 DB 설계

그림 6 은 본 논문의 어플리케이션 구현을 위해 사용한 데이터베이스의 관계를 나타낸 관계도이다.



(그림 6) Entity 관계도

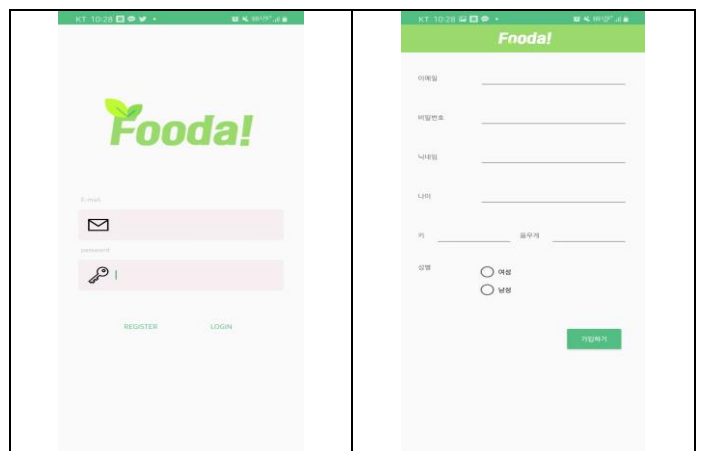
본 시스템은 총 6 개의 Entity 를 가지고 있으며, 이 Conceptual Design 을 바탕으로 Logical Design 으로 바꾼 것은 그림 7 와 같다.



(그림 7) Logical Design

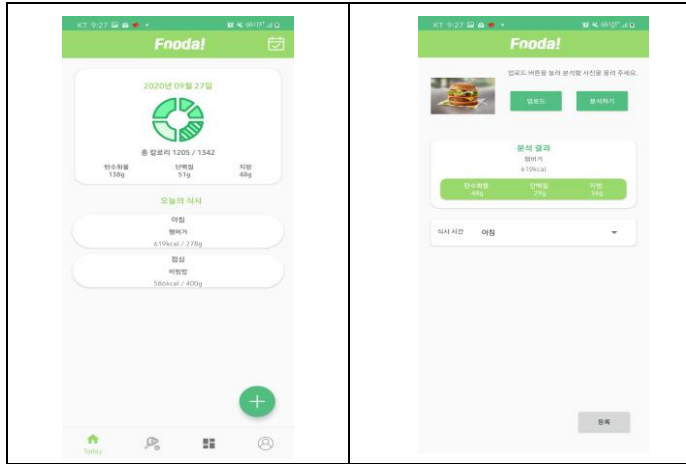
DB Table 을 구현하기 위해서 AQueryTool 을 사용하였다. 총 7 개의 테이블이 있으며, 이는 User, Diet, Diet_food, food, board, post, comment 이다. 각 테이블의 hidden 필드는 삭제 시에 데이터를 보존한다.

3.6 구현



(그림 8) 로그인, 회원가입 페이지

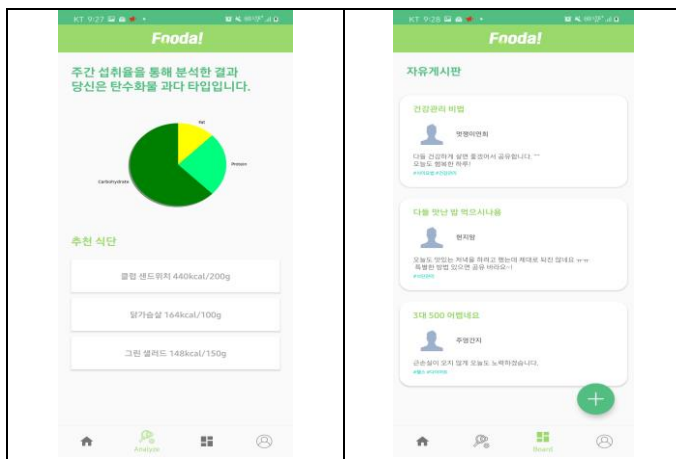
그림 8 은 로그인, 회원가입 페이지이다. 인증 토큰은 sha256 encrypt 를 통해 id, password 를 더해 encrypt 한 형태로 사용하였다. 사용자의 password 정보 또한 sha256 으로 encrypt 하여 개인 정보를 보호한다.



(그림 9) 메인 페이지, 식단 추가 페이지

메인 페이지에서 사용자가 해당 날짜에 섭취한 총 칼로리와, 기초대사량을 안내한다. 또한 3 대 영양소 섭취 현황을 알려주며, 해당 날짜의 식사 내역을 보여준다. 이를 바탕으로 사용자는 해당 일자에 섭취 가능한 잔여 칼로리를 확인한다. 우측 상단의 캘린더 버튼을 통해 날짜를 이동하며 정보 확인이 가능하다.

우측 아래의 플로팅 버튼을 클릭하면 그림 9 의 오른쪽 화면이 뜨게 된다. 업로드 버튼을 클릭하여 사진을 올리고, 분석하기를 눌러 해당 음식의 분석 결과를 확인할 수 있다. 이는 yolo 를 통해 음식 이름을 추론하고, DB 에 누적된 영양 성분이 반환된 값이다. 이를 바탕으로 사용자는 자신이 섭취한 음식에 대한 세부 정보를 확인할 수 있으며, 식사 시간을 선택하여 DB 의 Diet table 에 등록한다.



(그림 10) 추천 및 커뮤니티 페이지

추천 페이지에서 사용자가 1 주일 간 섭취한 영양 성분을 분석하여 그래프로 나타낸다. 이는 Matplot 을 사용하여 그래프로 그린 것이다. 주간 섭취 내역을 바탕으로 사용자의 식습관을 정의하고, 추천 식단을 안내한다. 커뮤니티 페이지에서 사용자 간의 상호 소통이 가능한 게시글을 작성할 수 있다.

3. 결론

본 논문에서는 영상처리와 딥러닝을 이용한 식단 관리 및 추천 애플리케이션을 설계 및 구현하였다. 본 논문의 어플리케이션은 기존의 식단관리 어플리케이션과 달리 식단을 수기로 작성해야하는 번거로움이 없으며 식단사진을 업로드하면 식단을 자동으로 분석해주는 편리함을 지니고 있다. 또한 섭취한 영양성분에 맞춰서 다음 식단을 추천해주고 사용자의 식습관을 분석하여 제공해준다. 그리고 추가로 사용자들끼리의 커뮤니케이션 기능도 추가하여 식단과 관련된 의견을 자유롭게 나눌 수 있게 구현하였다. 이처럼 식단 관리와 관련된 기능들을 이용자가 사용하기 편리하게 제공함으로써 지속적인 식단 관리에 도움이 되는 애플리케이션을 기대 할 수 있다. 추가로 질병군에 도움이 되는 DB 를 추가하게 되면 건강상에 문제가 있는 환자나 노인계층에 대한 자가 식단 관리 기능을 지닌 애플리케이션으로서 활용 또한 기대해 볼 수 있다. 이는 향후 연구 과제로 남긴다.

참고문헌

- [1] Maiyaporn Phanich "Food Recommendation System Using Clustering Analysis for Diabetic Patients" IEEE, 2013
- [2] M. Kim and H. Kim, "A study on diet behaviors and related factors in dieting college students," J. The East Asian Soc. Dietary Life, vol. 18, no. 1, pp. 135-148, Dec. 2008.
- [3] Joseph Redmon and Santosh Divvala and Ross Girshick and Ali Farhadi "You Only Look Once: Unified, Real-Time Object Detection" arXiv:1506.02640v5 [cs.CV] 9 May 2016

[본 논문은 과학기술정보통신부 정보통신창의인재양성 사업의 지원을 통해 수행한 ICT 멘토링 프로젝트 결과물입니다.]