



최종 보고서

머신러닝을 활용한 객체인식 레시피 추천 시스템 (냉장고를 부탁해)

2019.6.3

한국외국어대학교 융복합소프트웨어학과

1팀(NUDGE)



문서 정보

구 분	소 속	성 명	날 짜	서 명
작성자	한국외국어대학교	이민기	2019. 06.03	
	한국외국어대학교	윤형로	2019. 06.03	
	한국외국어대학교	김영주	2019. 06.03	
	한국외국어대학교	황선기	2019. 06.03	
	한국외국어대학교	표승수	2019. 06.03	
	한국외국어대학교	정기욱	2019. 06.03	
검토자	한국외국어대학교	윤형로	2019. 06.03	
승인자	한국외국어대학교	홍진표		

개정 이력

버전	작성자	개정일자	개정 내역	승인자
상세설계 보완 2.0	이민기	2019.05.27	최종 보고서 초안 작성	
	윤형로			
	김영주			
	황선기			
	표승수			
	정기욱			
최종보고서 1.0	이민기	2019.06.03	최종 보고서	
	윤형로			
	김영주			
	황선기			
	표승수			
	정기욱			

목차

1. 개요	4
1.1 서비스 정의	4
1.2 문서의 목적	4
1.3 범위	4
1.4 관련 문서	5
2. 시스템 소개 및 구성	6
2.1 개발 동향	6
2.2 개발 동기 및 필요성	6
2.3 시스템 구성요소	7
2.3.1 시스템 구성도	7
2.3.2 시스템 소개	7
3. 사용한 소프트웨어	8
4. 머신러닝 모델	10
4.1 머신 러닝	10
4.1.1 Create ML Frame Work를 사용한 데이터 모델 구축	10
4.1.2 새로운 모델이 필요한 이유 및 구축 과정	12
4.1.3 사용자 선택 유도	13
4.1.4 Object Detection	14
5. APPLICATION 인터페이스 및 서비스 설명	18
5.1 재료 스캔	18
5.2 오늘의 레시피	18
5.3 요리 선택 및 레시피 순서	18
6. 가상 시나리오	20
6.1 시나리오	20
6.2 SEQUENCE 다이어그램	21
7. 실제 적용 방안 및 기대 효과	22
8. 프로젝트 일정	23

1. 개요

본 장에서는 머신러닝과 스마트폰 카메라를 이용한 냉장고 식재료 인식 및 레시피 추천 시스템인 '냉장고를 부탁해'에 대한 상세 설계의 총괄 개요를 제공한다.

1.1 서비스 정의

"냉장고를 부탁해"는 머신러닝 기법과 실시간 객체 인식을 기반으로, 스마트폰으로 실시간 촬영한 재료를 자동으로 판별하여 그에 맞는 레시피를 추천한다.

1.2 문서의 목적

- 본 문서는 머신러닝 기반 객체인식 레시피 추천 어플리케이션 개발 프로젝트 중 다양한 요구에 유연하게 대응하기 위한 시스템 상세설계를 명세하고 있다.
- 본 문서는 사용자, 기획팀, 프로젝트 관리자를 대상으로 한다.
- 본 문서를 바탕으로 고객의 요구사항을 명확하게 도출하여 향후 개발 과정에서 이를 반영하는데 그 목적이 있다. 따라서 본 문서는 요구정의서에서 요구한 상세 기술에 있어서 분석하여 명세한다.

1.3 범위

"냉장고를 부탁해" 서비스는 사용자가 스마트폰으로 냉장고 속 재료를 촬영하면 실시간으로 해당 재료의 이름을 파악할 수 있도록 한다. 이를 바탕으로, 비교 알고리즘을 통해 해당 재료로 만들 수 있는 추천 레시피를 사용자에게 제공한다. 또한, 재료 부족으로 아깝게 요리하지 못하는 추천 레시피를 소개하고, 부족한 재료를 어플리케이션 내에서 구매할 수 있도록 별도의 서비스를 제공한다.

본 프로젝트에서는 개발 진행에서 다음과 같은 범위를 둔다.

- 1) 카메라가 인식한 객체 구별: 냉장고에서 인식한 객체들은 ML모델 내에 있는 데이터와 비교하여 정확도가 95% 이상일 경우 재료 리스트에 추가한다.
- 2) 재료 리스트 추가: 냉장고 안쪽에 있는 재료들을 인식하기 위해 카메라를 잠시 옆에 두고 재료를 꺼낸다. 그 후 카메라로 재료들을 인식하고 기존의 재료 리스트에 추가한다.

상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스

- 3) 레시피 추천 : 인식한 재료가 사용되는 레시피를 추천한다. 레시피는 미리 저장해둔 DB에서 참조한다.
- 4) 추가 재료 추천 : 몇 가지 재료를 추가하였을 시 새로운 요리가 가능할 경우, 해당 레시피를 분석하여 추가 재료 구매를 권유한다.

1.4 관련 문서

저자	문서 제목
김근영	ios 앱 개발을 위한 Swift 4
최은만	새로 쓴 소프트웨어 공학
Ramez Elmasri	데이터베이스 시스템 제 6판
최은만	UML로 배우는 시스템 분석 설계
오렐리앙 제롱	핸즈온 머신러닝(텐서플로를 활용한 머신러닝)

2. 시스템 소개 및 구성

2.1 개발 동향

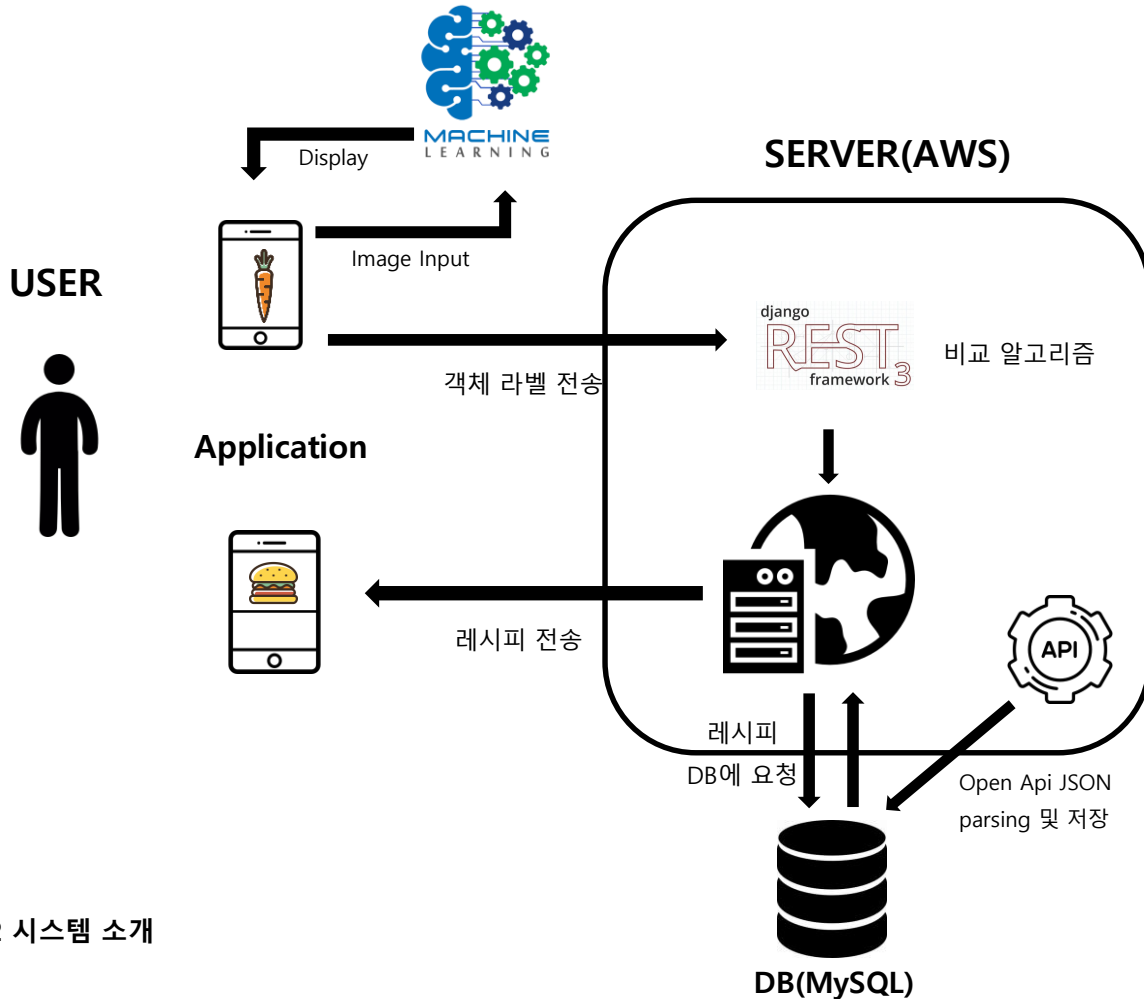
최근 "냉장고를 부탁해", "오늘 뭐 먹지" 등의 요리 예능이 많은 인기를 끌면서, 자취생부터 가정 주부까지 다양한 연령대의 사람들이 요리에 많은 관심을 보이고 있다. 또한, 이러한 예능에서 "집에서 쉽게 할 수 있는 요리"를 많이 선보이면서 사람들은 집에 있는 재료로 손쉽게 할 수 있는 요리 레시피를 많이 찾아보고 있다. 그로 인해 "만개의 레시피", "해먹남녀" 등의 레시피 관련 어플리케이션이 많은 인기를 끌고 있는 상황이다. 하지만 이들은 정해진 재료가 갖추어져야 요리를 할 수 있다는 한계점이 있다. 재료를 제대로 갖추지 못하거나 요리 실력이 부족한 사용자, 특히 자취생들은 수많은 레시피들이 존재함에도 불구하고 제대로 된 요리를 하지 못하는 경우가 많다. 따라서 본 프로젝트는 이러한 사용자를 타겟으로 하여, 머신러닝 기법을 이용한 객체 인식을 통해서 사용자가 가진 재료 내에서 더 편리하게 레시피를 추천할 수 있는 서비스를 제공하고자 한다.

2.2 개발 동기 및 필요성

최근 "쉽게 할 수 있는 요리"에 대한 관심이 급증하면서 요리 레시피 관련 어플리케이션도 많은 인기를 끌고 있다. 하지만 대부분의 어플리케이션은 단순히 여러 레시피를 보여주거나 인기 레시피를 추천하는 정도에만 머물러 있다. 또한, 해당 레시피의 재료가 없다면 다른 가능한 레시피를 일일이 찾아봐야 하는 불편함도 존재한다. 따라서 본 프로젝트는 최근 유행하는 머신러닝 기법을 이용하여 냉장고 속 재료를 촬영하기만 하면, 자동으로 재료를 인식하고 이에 맞게 레시피를 추천하는 서비스를 제공하고자 한다. 또한, 부족한 재료의 구매를 즉각적으로 배송 업체와 연결하여, 사용자들이 부족한 재료를 손쉽게 구매할 수 있도록 한다. 최종적으로, 이러한 기능을 통해서 사용자가 더욱 편리하고 효율적으로 요리할 수 있도록 도움을 주고자 한다.

2.3 시스템 구성요소

2.3.1 시스템 구성도



2.3.2 시스템 소개

USER	'냉장고를 부탁해' 서비스를 받는 대상이다. 스마트폰 카메라를 통해 인식한 객체의 라벨을 서버로 전송한다
Application	: 스마트폰에서 운용되는 Application으로 각종 구성요소와 Interface를 담당하고 있다. 카메라를 이용하여 재료를 찍고, Delegate Multi-Thread는 머신러닝으로 재료를 인식하고 디스플레이해준다. 인식된 재료를 서버에 객체와 라벨로 보내준다. 그리고 레시피를 서버로부터 받는다.
Server	: 비교 알고리즘 기반으로 냉장고에 있는 재료에 적합한 레시피를 DB에 요청하고 받는다. 또한 식품안전처에서 제공받은 JSON 타입 Open Api 를 parsing하여 DB에 저장시킨다.

DB(MySQL)	Server에서 운용되는 Database로 Server와 정보를 주고 받을 수 있다. 테이블은 레시피 테이블, 마켓테이블, 유저테이블
-----------	--

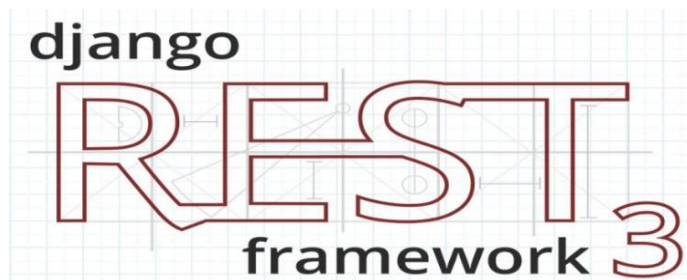
3. 사용한 소프트웨어

- Mysql



표준 데이터베이스 질의 언어인 구조화 질의 언어(SQL: Structured Query Language)를 사용하는 공개 소스의 관계형 데이터베이스 관리 시스템(RDBMS). 매우 빠르고, 유연하며, 사용하기 쉬운 특징이 있다. 다중 사용자, 다중 스레드(thread)를 지원하고, C, C++, 에펠(Eiffel), 자바, 펄, PHP, 파이썬(Python) 스크립트 등을 위한 응용 프로그램 인터페이스(API)를 제공한다. 유닉스나 리눅스, 윈도우 운영 체제 등에서 사용할 수 있다. 본 프로젝트에서는 레시피 API의 정보를 저장하는 역할을 한다.

- Django Rest Framework



프레임 워크는 소프트웨어의 구체적인 부분에 해당하는 설계와 구현을 재사용이 가능한 형태로 클래스들을 제공하는 것으로 쉽게 말하면 프로그래밍에서 응용 프로그램 표준 구조를 구현하는 클래스와 라이브러리의 모음이다. Django는 Python 으로 만들어진 무료 오픈 소스 웹 어플리케이션 프레임워크이다. 특징으로는 구현에 빠르고 웹 개발과 관련된 기능이 많으며 보안에 뛰어나고 확장성이 좋아 다양한 분야에 사용된다. Django REST 프레임 워크는 웹 API를 구축하기 위한 강력하고 유연한 툴 킷입니다. 본 프로젝트에서는 Django REST를 이용하여 API서버를 제작한다.

- AWS



아마존닷컴이 제공하는 각종 원격 컴퓨팅 서비스(웹 서비스)이다. 아마존 웹 서비스는 다른 웹 사이트나 클라이언트측 응용 프로그램에 대해 온라인 서비스를 제공하고 있다. 이러한 서비스의 상당수는 최종 사용자에게 직접 공개되는 것이 아니고, 다른 개발자가 사용 가능한 기능을 제공하는 플랫폼을 제공하는 PaaS이다. 아마존 웹 서비스의 각종 서비스는 REST 프로토콜 및 SOAP 프로토콜을 통해 접근, 이용 및 관리가 가능하다. 비용은 실제 사용량에 따라 결정되며, 일부 서비스의 경우 미리 고정된 금액을 지불하는 형태도 있다. 본 프로젝트에서는 비용 문제로 인해 테스트 서버로만 사용하였다.

- Create ML Framework



프레임워크를 사용하면 머신 러닝 모델을 손쉽게 빌드할 수 있다. 머신 러닝에 관한 전문적인 지식이 없어도 된다. Swift 덕분에 익숙하고 사용하기 쉬워진 Create ML은 Xcode10의 플레이그라운드에서 통합되어 모델 생성 워크플로우를 실시간으로 볼 수 있다. 단 몇 줄의 Swift 코드를 추가하여 Vision 및 자연 언어 기술을 활용하고 회귀, 이미지 분류, 단어 태그 지정, 문장 분류 등의 다양한 작업을 위해 Apple 생태계에 최적화된 모델을 만들 수 있다. 또한, Mac을 사용하여 전용 서버 없이도 사용자 설정 데이터로 Apple의 모델을 학습시킬 수 있다.

본 프로젝트에서는 객체를 인식하는 모델 분류기 머신러닝 모델을 만들기 위해 사용하였다.

- Turicreate



Turi Create는 맞춤형 기계 학습 모델 개발을 간소화한다. 권장 사항, 객체 탐지, 이미지 분류, 이미지 유사성 또는 활동 분류를 앱에 추가하기 위해 기계 학습 전문가 일 필요는 없다.

상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스

알고리즘 대신에 태스크에 집중하였고 내장된 스트리밍 시각화를 사용한 데이터 탐색을 한다. 텍스트, 이미지, 오디오, 비디오 및 센서 데이터 지원하고 단일 시스템에서 대규모 데이터 세트 작업이 가능하다. iOS, macOS, watchOS 및 tvOS 앱에서 사용할 수 있도록 코어 ML로 모델 내보낼 수 있다. 본 프로젝트에서는 이미지 사진을 이용한 머신러닝 모델 학습을 하였다.

4. 머신러닝 모델

4.1 머신 러닝

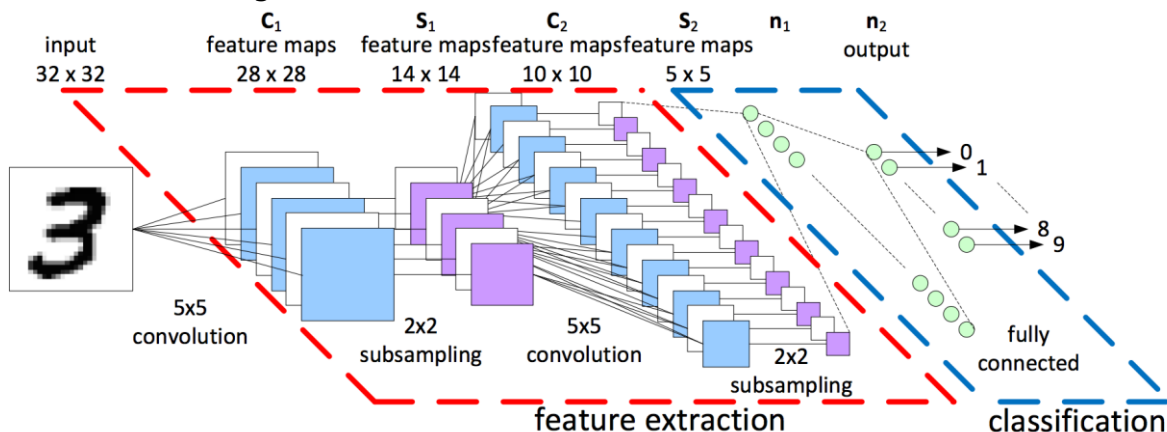
4.1.1 Create ML Frame Work를 사용한 데이터 모델 구축

Real time Object Classification : Mac OS에서 제공되는 Pre-Trained Model(scenePrint, Apple에서는 Pre-trained Model로 부르지 않고 Feature Extractor라 부른다.

왜냐하면 일반적인 pre-trained 모델은 이미지를 특징 벡터로 전환한 후 classification에서 사용하기 때문이다. (Apple uses the phrase "feature extractor" instead of the more common "pre-trained model". This is because the common layers of the pre-trained model convert an image into a feature vector that is good for classification tasks.)

출처: <https://www.learnopencv.com/how-to-train-a-deep-learning-based-image-classifier-in-macos/>

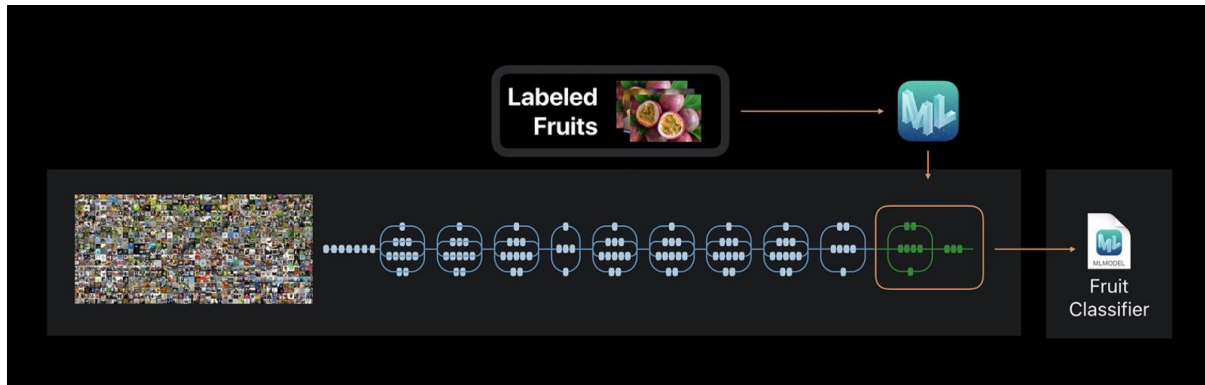
1억장 이상의 트레이닝 된 이미지 데이터가 있으며 이미지 데이터는 모두 특징 벡터로 전환되어 있어 transfer learning을 하는데 유리하다.)



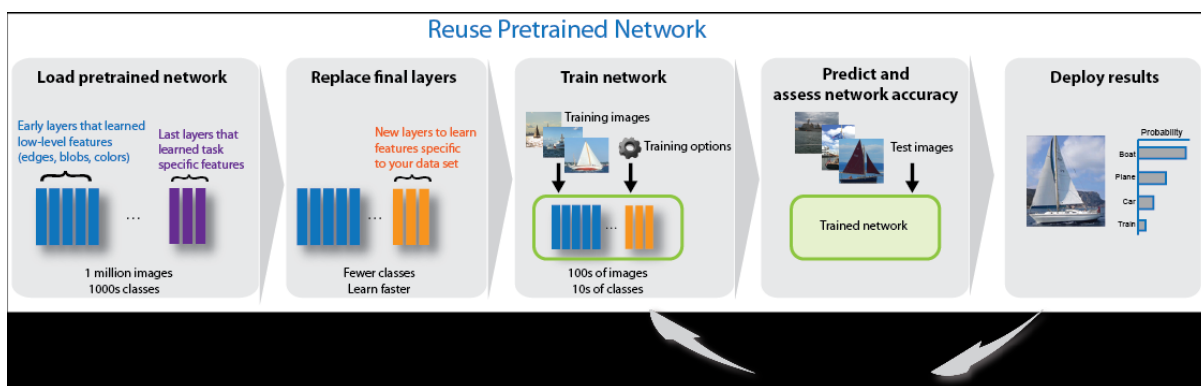
출처 : https://www.kernix.com/blog/a-toy-convolutional-neural-network-for-image-classification-with-keras_p14

즉, pre-trained 모델을 사용하는 것은 앞서 학습된 모델의 feature extraction을 이용해 마지막 layer를 학습시키는 것이다.

상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스



출처 : <https://medium.com/eliiza-ai/wwdc-2018-apple-announces-create-ml-976c30a80192>



(Network에서 Pre-trained 모델을 사용해서 새로운 모델을 학습시키는 과정)

(출처 : <https://www.mathworks.com/help/deeplearning/examples/transfer-learning-using-alexnet.html?sessionid=108d986e091575f5d4b0d64853a8>)

→위의 그림에서는 네트워크에서 pre-trained 모델을 로드 받은 뒤,
사용자가 입력한 데이터 셋을 추가적으로 학습시키는 것을 설명하고 있다.

▼ Model Evaluation Parameters

Name	Type	Flexibility	Description
▼ Inputs			
image	Image (Color 299 x 299)	299... x 299...	Input image to be classified
▼ Outputs			
classLabelProbs	Dictionary (String → Double)		Probability of each category
classLabel	String		Most likely image category

(실제로 만든 ML모델을 사용해서 이미지를 인식할 때 input과 output)

ML 모델을 사용하여 이미지를 인식 할 때 인풋(input)인 이미지(image)를 이용하여 연산한 결과 (output) 는 String(물체의 이름) 과 Double(정확도)를 반환해준다.

상세 설계서: 머신러닝을 활용한 객체 인식 레시피 추천 '냉장고를 부탁해' 서비스



김치

물체의 이름과 정확도 반환결과

4.1.2 새로운 모델이 필요한 이유 및 구축 과정

ResNet-50, Inception v3는 식재료만을 인식하는 ML모델이 아니고 또한 음식 및 식재료를 인식한다고 해도 한국 식재료에 맞는 식재료를 인식하지 못하는 어려움이 있다. 따라서 한국 식재료에 맞는 새로운 모델을 구축할 필요가 있다.

ML 모델 구축은 Apple사에서 제공하는 Create ML Framework를 사용할 예정이며 ML 모델을 학습시키는 과정은 다음과 같은 순서로 이루어진다.

- ① 현재 문제를 인식하고 그에 따른 모델을 선정함.

냉장고에 많이 들어가는 식재료를 기준으로 16가지의 식재료를 선정하고 이를 모델로 선정하였다.

- ② 선정된 모델에 따른 데이터 셋을 준비함.

배경화면의 변수가 가장 적고 모델의 카테고리를 각 대표할 수 있는 이미지 데이터를 준비함.

- ③ 다양한 환경에서 인식할 수 있는 이미지가 필요하므로 Data Augmentation 기법을 적용함.

카메라 해상도, 화질, 냉장고 환경을 다각적으로 고려하여 Data Augmentation 기법을 적용시켜 준다. 적용시킬 기법은 Blur(흐릿함), Exposure(빛 노출), Noise(이미지 노이즈)를 선정하였음. 왜냐하면 냉장고 화면은 어둡거나, 냉장고 조명에 의해 밝은 경우가 많고 카메라 해상도와 촬영 환경에 따라 흐리게 나오는 경우가 많기 때문이다.

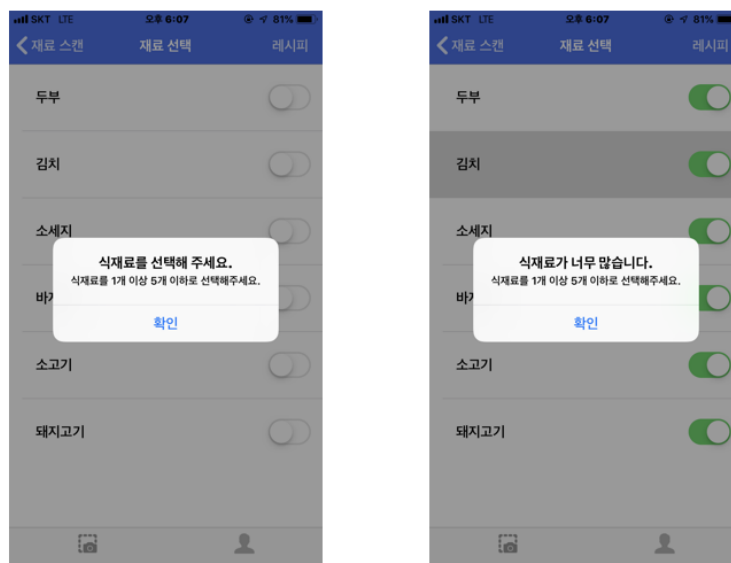
④ Training

약 27,000 장의 이미지를 학습시켰고 학습 시간은 1시간 30분가량이 소요됨.

⑤ Evaluation

80:20의 비율로 준비한 데이터셋에서 20에 해당하는 평가 데이터를 넣어 줌으로써 평가를 실시함.

4.1.3 사용자 선택 유도



- 위에서 만들어준 switchCounter를 이용해 0일 때 그리고 6개 이상일 때 경고창이 나오도록 해 사용자가 1~5개 사이의 재료를 선택하게 만든다.
- 사용자가 1~5개의 재료를 선택했다면 다음 화면으로 전환한다.
- 화면을 전환하면서 compactMap이라는 swift library함수를 사용해 사용자에게 의해 선택된 재료만 새로운 배열에 추가하여 그 배열을 전달한다.

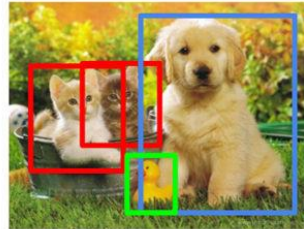
4.1.4 Object Detection

Classification



CAT

Object Detection



CAT, DOG, DUCK

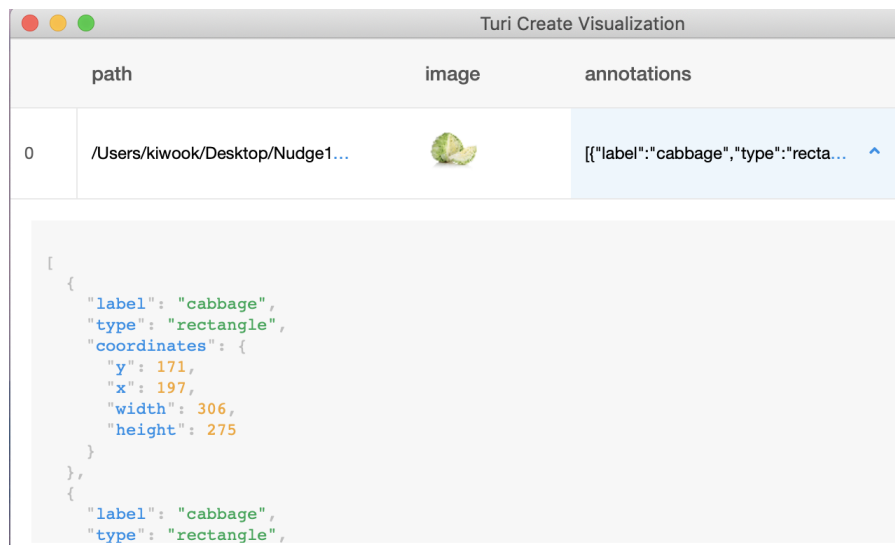
출처 : <https://mikail0205.github.io/computer%20science/2018/05/20/CS231n-Computer-Vision/>

Object Detection이란?

→ 물체를 무엇인지 분류한 뒤 물체의 위치까지 파악해주는 머신 러닝 기술.

① 데이터 준비

일반적인 이미지 데이터에 네모 박스(Bounding Box Annotation)을 그린 뒤 이미지와 연결되는 JSON Annotation을 준비한다.



(실제 bounding box annotaion을 적용한 이미지)

② 사용한 신경망(YOLO)

Darknet은 신경망 프레임워크(neural network framework)로써 DNN(deep neural network)들을

상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스

학습시키고 실행시킬 수 있는 틀(framework)이다. 이 Darknet을 통해 만들어진 YOLO 신경망에 이미지와 바운딩 박스를 결합시켜 학습을 시켜주었다.

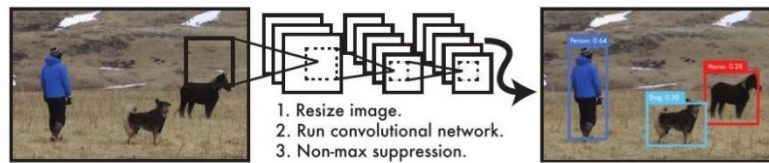
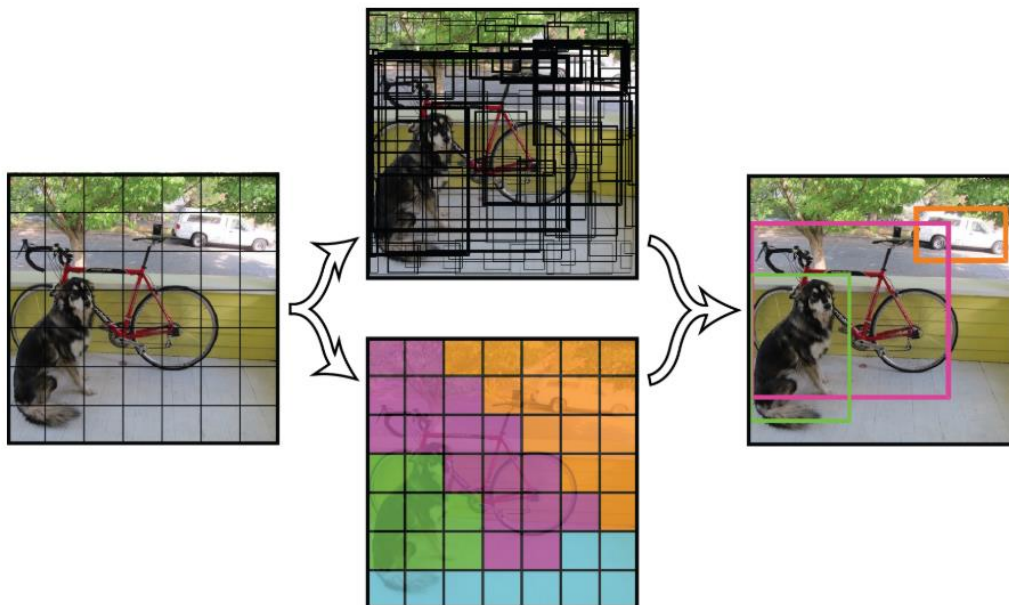


Figure 1: The YOLO Detection System. Processing images with YOLO is simple and straightforward. Our system (1) resizes the input image to 448×448 , (2) runs a single convolutional network on the image, and (3) thresholds the resulting detections by the model's confidence.

(출처 : <https://curt-park.github.io/images/yolo/Figure1.JPG>)



(출처 : <https://github.com/pjreddie/darknet/wiki/YOLO:-Real-Time-Object-Detection>)

JSON과 결합한 이미지는 YOLO신경망을 통과하면서 이미지를 여러 지역으로 분류하고 그 지역에 위치한 바운딩 박스의 확률을 계산(Prediction)하여 가장 높은 수치의 지역에 바운딩 박스를 그려주는 기법이다. 즉, YOLO신경망은 각 Grid(격자무늬) 별로 예측을 하여 가장 높은 값을 얻는다. (This network divides the image into regions and predicts bounding boxes and probabilities for each region.)

③ 실제로 사용한 데이터 및 카테고리 & 장비

실제 학습 시킨 데이터는 약 800장 (10가지 카테고리 : 양배추, 된장, 스팸, 달걀, 고추장, 김치, 우유, 당근, 토마토, 양파) 이며 예측을 위한 Test 데이터 200장을 사용하여 총 1000장의 데이터를 학습 시킴. 실제 GPU로도 학습시킬 수 있었지만 CPU (i5 8세대) 를 사용하여 학습을 진행하였음. 또한, Max iteration의 수치(최대 반복)를 1000으로 설정하였으며 Data Augmentation을 해주지 않고 학습 시킴. (원래 Max iterations의 Default는 6000이었지만 시간관계상 1000회만 실시함.)

```
Run: MLClassifier x
Finished parsing file /Users/kiwook/Desktop/Nudge10TrainingDataset/annotations.csv
Parsing completed. Parsed 804 lines in 0.008804 secs.
Materializing SFrame
Using 'image' as feature column
Using 'annotations' as annotations column
Setting 'batch_size' to 32
Using CPU to create model
+-----+-----+-----+
| Iteration | Loss      | Elapsed Time |
+-----+-----+-----+
| 1          | 11.131    | 8.8           |
| 3          | 11.097    | 25.8          |
```

CPU를 사용한 학습을 실시

```
Run: MLClassifier x
+-----+-----+-----+
| 983      | 2.374     | 8007.3        |
| 985      | 2.377     | 8023.4        |
| 987      | 2.373     | 8039.5        |
| 989      | 2.404     | 8055.7        |
| 991      | 2.355     | 8072.1        |
| 993      | 2.317     | 8088.3        |
| 995      | 2.310     | 8104.5        |
| 997      | 2.310     | 8120.6        |
| 999      | 2.418     | 8137.1        |
| 1000     | 2.424     | 8145.8        |
+-----+-----+-----+
Finished parsing file /Users/kiwook/Desktop/Nudge10TestDataset/annotations.csv
Parsing completed. Parsed 100 lines in 0.037787 secs.
```

1000회의 반복을 마친 후 약 8100초의 시간이 경과됨. 약 2시간 20분이 걸림. 원래의 디폴트 iteration을 사용했다면 약 6배의 시간이 더 걸릴 것으로 예측됨.



상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스

④ 학습을 완료한 mlmodel과 classifier와 비교

Machine Learning Model

Name NudgeOD02
Type Pipeline
Size 64 MB
Author unknown
Description Object detector (darknet-yolo) with non-maximum suppression created by Turi Create (version 5.5)
License unknown

Model Class

NudgeOD02
Automatically generated Swift model class

Model Evaluation Parameters

Name	Type	Flexibility	Description
▼ Inputs			
image	Image (Color 416 x 416)		Input image
iouThreshold	Double		(optional) IOU Threshold override (default: 0.45)
confidenceThres...	Double		(optional) Confidence Threshold override (defa...
▼ Outputs			
confidence	MultiArray (Double 0 x 10)	0... x 10	Boxes x Class confidence (see user-defined m...
coordinates	MultiArray (Double 0 x 4)	0... x 4	Boxes x [x, y, width, height] (relative to image...

10가지 카테고리를 학습시킨 Object Detection 모델, Output이 정확도(Confidence)와 [x, y, width, height] 즉, 위치정보를 포함한 값이 나온다.

Machine Learning Model

Name NudgeMLModel07
Type Image Classifier
Size 263 KB
Author 정기욱
Description A model trained using CreateML
License Unspecified

Model Class

NudgeMLModel07
Automatically generated Swift model class

Model Evaluation Parameters

Name	Type	Flexibility	Description
▼ Inputs			
image	Image (Color 299 x 299)	299... x 299...	Input image to be classified
▼ Outputs			
classLabelProbs	Dictionary (String → Double)		Probability of each category
classLabel	String		Most likely image category

Classifier Model, Output이 Label(물체 이름) 과 Confidence(물체 정확도) 가 나온다.

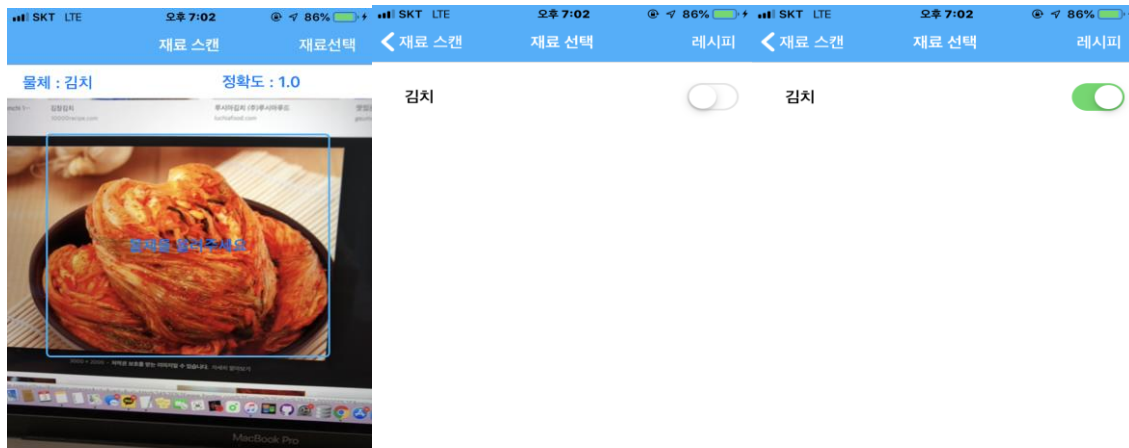
그러므로, Classifier와 Detection Model의 공통점은 물체의 이름 (Label)과 정확도

상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스

(Confidence)를 결과물(Output) 으로 얻는 것이다. 반면, 차이점은 Detection은 예측된 물체의 위치값을 같이 보내주는 것이다.

5. Application 인터페이스 및 서비스 설명

5.1 재료 스캔

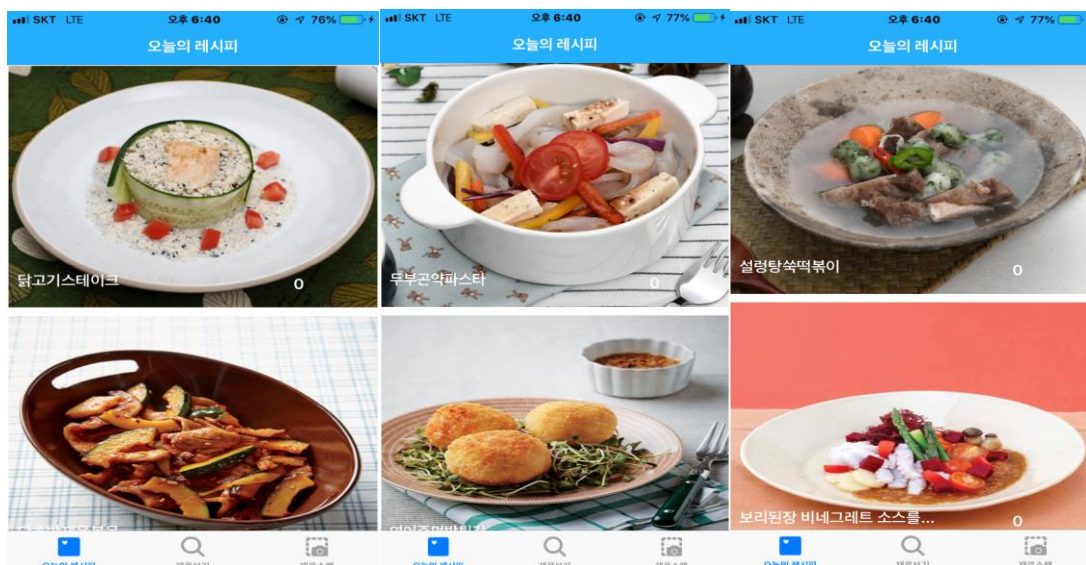


김치



인식하고 싶은 재료를 모바일 폰 카메라로 객체 탐색을 한다. 탐색된 재료는 사용자가 선택을 할 수 있고 선택하지 않을 수 있다. 구현상 1~5개사이로 재료를 선택해야 하는 제한 사항이 있다.

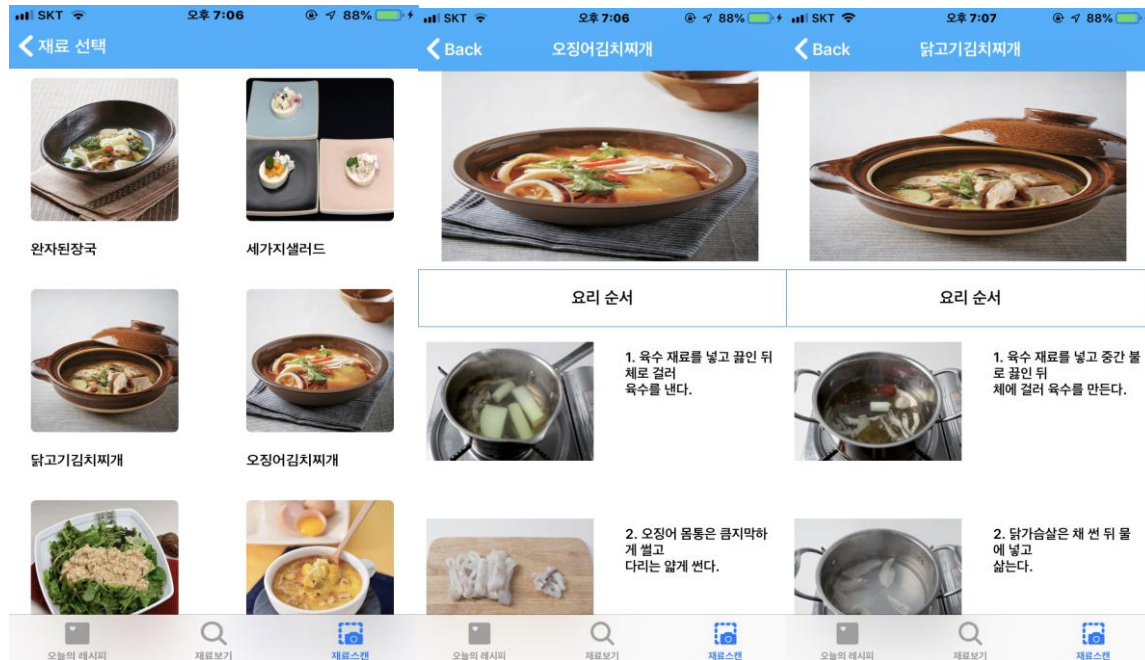
5.2 오늘의 레시피



오늘의 레시피는 매일 10개의 새로운 레시피를 랜덤으로 첫 화면에 보여준다. 사용자는 오늘의 레시피를 보며 만들고 싶은 요리의 레시피를 확인할 수 있다.

5.3 요리 선택 및 레시피 순서

상세 설계서: 머신러닝을 활용한 객체인식 레시피 추천 '냉장고를 부탁해' 서비스



인식한 재료로 만들 수 있는 음식들을 보여준다. 요리를 선택하면 사진과 함께 조리순서를 보여준다.

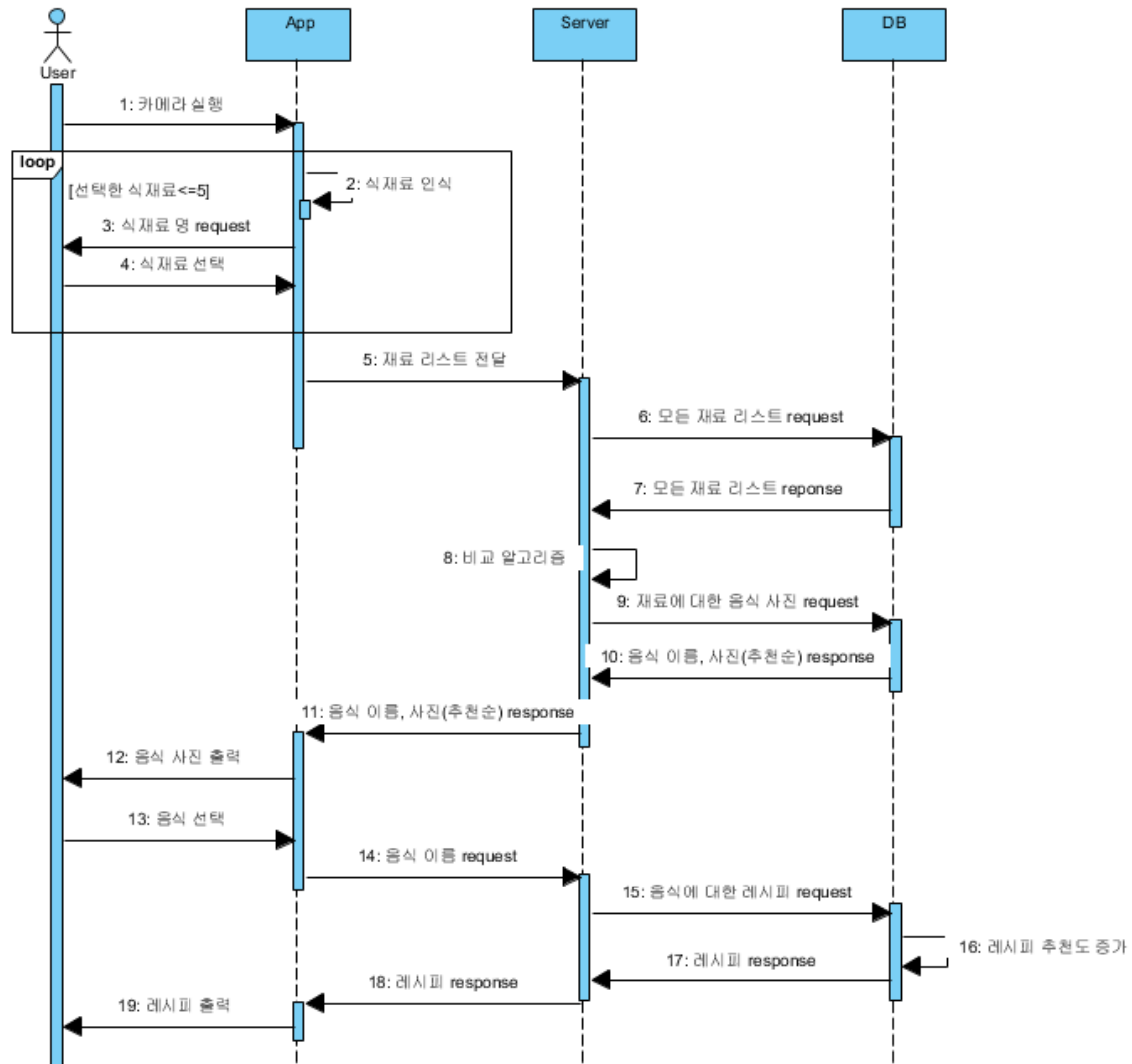
6.가상 시나리오

6.1 시나리오

민기는 학교 앞에서 혼자 지내고 있는 자취생이다. 그의 냉장고에는 넣어둔지 오래된 재료들이 가득 쌓여있다. 그는 이 재료들을 이용해서 할 수 있는 요리가 없을까 고민했다. 그래서 "냉장고를 부탁해" 어플리케이션을 이용하기로 했다. 그의 냉장고에는 재료들이 많이 있었지만 어떻게 사용해야할지 잘 모르기 때문에, 사진의 객체 인식을 통해 재료를 판별하는 "냉장고를 부탁해"가 사용하기 매우 적합하였다. 그는 먼저 어플리케이션에 로그인을 하였다. 그리고 촬영을 위해서 어플리케이션의 카메라 접근을 허용하였다. 이 후, 스마트폰을 이용해 냉장고 속 재료를 촬영하였다. "냉장고를 부탁해"는 객체 인식을 통해 민기의 냉장고 재료를 인식하고, 그에 맞는 레시피를 추천하였다. 민기는 그 중에서 가장 마음에 드는 "돼지고기 김치찌개"를 선택해서, 냉장고 속 재료를 이용해 요리하였다. 그리고 "계란"을 추가로 구매하면 "장조림"을 할 수 있다는 어플리케이션의 추천을 받는 민기는 화면 아래에 있는 링크를 통해 "계란"을 즉시에서 구매할 수 있었다.



6.2 Sequence 다이어그램



7. 실제적용방안 및 기대효과

- 증가하는 '혼밥 족'을 타겟으로 하는 건강한 혼밥 레시피 제공
-

요즘 혼자 사는 가구가 증가하면서 '혼밥족'들이 증가하고 있다. 2016년, 인구수가 5000만을 넘으면서, 전체 1911만 1000가구 가운데 1인 가구 수는 520만3000가구로 집계되었다. 이는 전체 가구 수 중에 27.2%를 차지하는 것으로 2인 가구 비중을 제쳤다. 당초에 2020년에 1인 가구 수가 전체 가구 가운데 1위로 올라설 것으로 전망했지만, 5년 앞서 현실화된 것이다. 이처럼 1인 가구수의 증가속도는 예상보다 빨라지고 있고, 이에 따라 우리나라의 배달 음식의 시장도 폭발적으로 급성장하였다. 2018년 배달 음식의 시장 규모는 20조원 이상으로 집계되고 있다. 또한 배달하지 않고, 집에서 직접 해먹는 인구 수도 급격하게 증가하였다. 이에 따라 건강한 혼밥에 대한 관심도 증가하였다.

해당 시스템은 이러한 트렌드적 가치관과 미래 유망 사업성을 기반으로 하고 있다. 사용자는 스마트 폰에서 애플리케이션을 설치한 후, 스마트 폰의 카메라를 이용해 사용하고 싶은 재료를 촬영하게 되고, 이렇게 모인 재료를 토대로 만들어 먹을 수 있는 건강한 메뉴를 제공해주는 서비스를 제공한다. 이를 통해 사용자는 건강한 혼밥을 실천할 수 있고, 남은 재료를 적절히 활용하여 낭비되는 재료를 줄일 수 있을 것이다.



8. 프로젝트 일정

구분	구현 내용		4월				5월				6월			
			1주	2주	3주	4주	1주	2주	3주	4주	1주	2주	3주	4주
계획	아이디어 구상 및 기초 설계													
사전 준비	데이터 수집 및 정제	데이터 수집												
		데이터 정제												
설계	DB설계 및 서버 구축	DB설계												
		서버 구 축												
개발	어플리케이션 개발	UI제작												
		프로그램 연동												
테스트	서버 연동 및 테스트	서버 연 동 및 프 로그램 디버깅												
종료	작품전시회 준비 및 최종보고서 작성													