DeepCalorieCam: An iOS App for Dish Detection and Calorie Estimation

Ryosuke Tanno    Ege Takumi    Keiji Yanai
The University of Electro-Communications, Tokyo, Japan

1. Overview

- A CNN-based real-time dish detection and calorie estimation system, DeepCalorieCam, running on iOS.

“DeepCalorieCam” running on iPhone 7 Plus

Predict Dish Category

Predict Food Calorie

2. Flow of Implementation

① Training CNN Models

Calorie Estimation CNN

Predict Dish Category

Predict Food Calorie

Food Detection

Multi-task CNN

518 kcal

spaghetti

② Converting Trained Models into Core ML

Core ML Tools

python

③ Deploying Core ML Models on iOS Devices

Core ML model

Core ML

DeepCalorieCam

3. Implementation Details

Training

- First, we trained the dish detector by fine-tuning YOLOv2[1] with UECFOOD-100. (with Keras)
- Next, we trained the food calorie estimation CNN by fine-tuning of Inception-v3 with food calorie estimation with 15 categories dataset[2]. (with Keras)
- After that, we converted the trained Keras models into CoreML models, and deployed it.

Q&A

- Why YOLOv2 and Inception-v3?
  - There are three major networks for object detection, Faster R-CNN, YOLO and SSD. Both YOLO and SSD showed better performance when compared to Faster-RCNN.
  - Now we are left out with two major options: YOLO and SSD. On the other hand, the CoreML team is constantly releasing new updates. So, I should wisely choose a network with simple layers (only convolutions) and not fancy operations such as Deconvolutions, Dilated convolutions and Depth-wise convolutions. - YOLO (v2) is much simpler than SSD. - Inception-v3 is also the same reason as above.

4. Demo Movie

YOLOv2 Results on Pascal VOC 2007

5. Project Page

- Project HP with online demos:
  - http://foodcam.mobi/deepcaloriecam/

References