eYSIP2017

Vegetable Identification Using Transfer Learning

Sanket Shanbhag
Supriya Suresh
Saurav and Naveen

Duration of Internship: 22/05/2017 – 07/07/2017

2017, e-Yantra Publication

Vegetable Identification Using Transfer Learning

Abstract

This project aims to create a system for automatically logging farm produce from the green house at e-Yantra. An image is captured of the produce and this image is fed to a neural network based on the Inception-v3 model by Google. This system then identifies the vegetable and classifies it into different classes it has been trained on. Also, a re-training system is implemented which automatically trains any new images captured and thereby further increasing the accuracy of the system over time.

Completion status

The system has been successfully trained for 18 classes (vegetables) achieving accuracy of 99.5% on a separate test set. Also a system has been implemented for auto-training any new images captured on the system every 4 days.

Hardware parts

- List of hardware
- Detail of each hardware: Datasheet, page 5, Vendor link,
- Connection diagram

Software used

- Python 3.5.2

- Tensorflow 1.2.0
- PrettyTensor 0.7.4
- Installation Steps:
  - Open a terminal in the project base folder and type:
    `pip3 install -r requirements.txt`

Assembly of hardware

Circuit diagram and Steps of assembly of hardware with pictures for each step

Circuit Diagram

Circuit schematic, simplified circuit diagram , block diagram of system

Step 1

Steps for assembling part 1

Step 2

Steps for assembling part 2

Step 3

Steps for assembling part 3

Software and Code

The complete code is available here. It is divided into various folders each with a self contained module of the project.

Downloading and formatting data

The folder `download_data` contains python scripts and shell scripts for downloading and formatting data from a list of URLS or from ImageNet. Information on using these scripts can be found in the wiki page.

Transfer Learning

The code for transfer learning can be found in the `transfer_on_inception_v3` folder. The `transferveg.py` file contains the code for running the model. We remove the final layer of the Inception-v3 model and add 3 fully connected layers of 4096, 2048, and 1024 nodes with a dropout layer after the 2048-node layer. This helps us achieve an accuracy of 99.5%

Module Integration

The `ghfarm.py` script contained in the `module_integration` folder will run on the raspberry pi on the weighing machine. It interfaces with the server to predict the crop and sends data once the image is taken.

Auto-Training

The `AutoTrain` folder contains code for running a server to accept images and scripts for adding cron tasks to run the autotraining code every 4 days.

Use and Demo

Final Setup Image

User Instruction for demonstration

Youtube Link of demonstration video

Future Work

What can be done to take this work ahead in future as projects.

Bug report and Challenges

Any issues in code and hardware.

Any failure or challenges faced during project

Bibliography

[1] Ad Kamerman and Leo Monteban, *WaveLAN-II: A High-Performance Wireless LAN for the Unlicensed band*, 1997.