University of Hong Kong

COMP4801 Final year project

Detailed Project Plan

InnoIris

Chiu Yat Man, Lam Fong Pui, Lam Tim Po, Wong Yuk Kwan

Department of Computer Science

Supervised by

Dr. Chui Chun Kit

Department of Computer Science

September 29, 2019

Contents

1 Introduction

1.1 Background

The Tam Wing Gan Innovation Wing (Inno Wing) will serve as an iconic landmark of the Faculty of Engineering in The University of Hong Kong in 2020 [1]. It provides makerspace area for students to do creative explorations and practical hands-on experiments. While students are working in the makerspace, some tools are available for them to use. Since it is hard to maintain the tool status with limited manpower. InnoIris as an automated tool management system can take over traditional human power and paperwork in makerspace area in Inno Wing.

1.2 Objective

InnoIris is an automated tool management system for the Innovation Wing. Taking the advantages of the unmanned store, InnoIris helps students to have better working experience when they are using the tools in makerspace area.

InnoIris achieves three major objectives.

First, InnoIris reduces human error by automating the tool checkout and return the Inno Wing makerspace area. A centralized resource center will be deployed to do the borrow and return process in one place.

Second, InnoIris gives high protection for school properties since only registered students can enter Inno Wing makerspace area and borrow authorized tools.

Third, InnoIris provides a centralized resource center. Student can borrow and return tools in Inno Wing makerspace area instead of different laboratory.

InnoIris will also serve as a computer vision community. The public can send their dataset to InnoIris to enhance its ability or make use of of it. Since a real-life application will be developed based on the models, InnoIris will also show the potential and how one can make use of the models, along with other software development technology to facilitate our daily lives.

1.3 Related Work

1.3.1 Amazon Go

Amazon Go is the most famous and successful unmanned stores in the world [2]. It is the first unmanned store opened to the public in January 2018 [3]. It combines technology with computer vision, deep learning algorithm, and sensor fusion to do all the checkout process. When a customer wants to buy sometimes from Amazon Go, they just need to enter the store by using the app and then pick the products. They can leave the store directly when the shopping is done. Amazon team uses the camera, sensors and RFID readers to collect the data, and Just Walk Out Technology [4] created by the team can detect the products picked up by the customer through those data.

1.3.2 CKC18

A.S. Watson Group launched an unmanned concept store, CKC18, at Central Hong Kong in 2018 [5]. CKC18 makes use of RFID technology to identify each product. Every product contains an RFID tag and the checkout counters embed the sensor. When a customer places a shopping basket in the checkout counter, the RFID sensor will scan the tags pasted on every product. The system will calculate cart status and the total price accordingly.

2 Execution

To make it easier to manage the project, the project will be divided into two stages. The first stage will be focusing on detecting tools. The second stage will be the application of the first stage.

2.1 Stage 1

2.1.1 Objective

In Stage 1, the goal is to build a machine learning model which can detect specific tools with high accuracy. Hammers, scissors and screwdriver are the earliest target tools. With the camera sending a video stream to the system, if a specific tool exists inside the video, the system will detect it with its name and display a rectangle to bound the tool for demonstration.

2.1.2 User Journey

A user passes a video stream to the machine learning model through an API(Application Programming Interface). Then, the API will return the names and coordinates of identified tools within the scene. For the demonstration, the coordinates will be visualized as a rectangle labeling the corresponding tools in the scene.

2.1.3 Methodology

TensorFlow, the most popular among machine learning frameworks [6], will be used to train the model for detecting the specific tools in Inno Wing makerspace area. Working together with researchers, Google had played an important role in the development of TensorFlow. Due to its high popularity, there are tutorials, either in Chinese or English, regarding the usage of TensorFlow. We also expect the community for TensorFlow is large enough for us to seek help from when encountering obstacles.

To facilitate the development of InnoIris, machine learning model will be trained with a pre-trained model instead of training from scratch to shorten development time. There are a number of pre-trained models which can be found in the official TensorFlow GitHub repository [7]. In that repository, there are two major types of models, R-CNN (Region-Based

Convolutional Neural Network) and SSD (Single Shot MultiBox Detector). R-CNN and SSD are both methods of bounding objects in an image with different underlying principles. In general, SSD runs faster but with a lower accuracy while R-CNN has a lower speed but higher accuracy. Each pre-trained model is either R-CNN or SSD, together with a feature extractor. One of the documentation there [8] lists out the speeds as well as mAPs (mean Average Precision) of different models. mAP is a measure of the success rate of spotting out an object in an image and is generally regarded as a metric of the accuracy of a model in different object detection contexts, for instance, Open Images Challenge [9]. Machine learning model will be trained on top of different pre-trained models with our custom dataset.

The model with the highest mAP in 50,000 training steps among all models which will be used for the later development of the whole project and we plan to use the model with video stream which can be considered as a series of images at a constant rate, speed of detection is the secondary concern. As a result, the speeds of each model will be tested before selecting a model for our use. A detailed calculation method will be developed after having more intensive and in-depth literature review to determine the exact weightings for improving accuracy and the processing speed.

Regarding the dataset, current pre-trained models are usually trained with either COCO dataset from Microsoft or Open Images Dataset from Google. They cover several kinds of objects in their datasets. As a result, the pre-trained models should have the ability to recognize the items covered in the datasets. However, we decided to train with our dataset on top of those pre-trained models because of the following two reasons. Firstly, the machine learning model needs to be very specific to the usage in DreamLab. Therefore, Only Inno Wing makerspace tools can be included the tools. Secondly, some tools might not be covered in the datasets, for example, hammers are not covered in COCO dataset. Hence, the pre-trained models have to be trained to adopt COCO dataset with images of the hammer.

The dataset will be composed of images of different kinds of items. It contains the images from different sources such as Google search, COCO Dataset, and Open Images Dataset. Also, The tool images in Inno Wing makerspace area will be included in the dataset. Around 180 images with 1280x720 resolution will be prepared for each kind of tool. LabelImg [10] will be used to annotate all images and mark the coordinates of the target items. The annotation will be saved as a .xml file in VOC format by LabelImg. A prepared script will convert it into a .csv file and then to TFRecord format which is readable to TensorFlow for the training process. For images from COCO dataset and Open Images dataset, image annotations can be downloaded from their websites. However, they are not in VOC format and we will carry out the conversion from their formats to TFRecord. Also, image augmentation also will be done before training. The purpose of using Image augmentation is to prevent overfitting in machine learning and have a better performance in image classification [11]. To do the Image augmentation, the conversion for a set of input images into the new images with different effects, such as rotation, affine, fliplr, is necessary.

Apart from building the model, to smoothen the development of Stage 2, API Driven Development will be adopted. There will be mainly two types of API. They will be written in Python which TensorFlow is friendly with. The first type of API will be a RESTFul API powered by web service. We will create an API for users or other programs to send an image to and then the API will call backend TensorFlow inference service and return the coordinates and the types of objects spotted in the image uploaded. For the second type of API, the machine learning model will be packed as an engine. It can be used as a module to other Python programs, to be specific, the programs that are going to be developed in Stage 2. It is expected to be used to take in the video stream and continuously output inference result which is the detection of objects.

2.1.4 Challenge

The major challenge is the accuracy of the deliverable machine learning model. we do not know if a successful model with high accuracy can be made. Though the hyperparameters, which are the weightings to help the model judges whether or not the pixels on images belong to an object, might not be able to be adjusted.

There are various approaches we will adopt for the challenge. Increasing the number of samples in model training and downsampling (lower the resolution of the sample images) can improve the accuracy of the model. Also, changing the brightness and contrast of samples might help the model to adapt to different images. In addition to the measures to be done to the sample images, two models can be trained to recognize tools in different environments. After evaluation, the combined results from the two models will be the final judgment.

2.2 Stage 2

2.2.1 Objective

Inspired by Amazon Go's automated checkout idea, the advantage of automating borrowing and returning of tools via computer vision models is developed in stage 1. Apart from the automated borrowing, inventory management is also digitized, meaning any overdue borrows will be notified to both the students and administrative staff.

2.2.2 User Journey

The flow is composed of 3 stages: pre-borrow, borrowing, post-borrow.

During the pre-borrow stage, registered user logins with a QR in a mobile app. The student scans the QR code on the QR reader to finish the login process. InnoIris can detect the student in front of the QR code and associated the person in the camera with the authenticated user. The position tracking remains until the student leaves the Inno Wing makerspace area.

In the borrowing stage, based on live footage from the camera, the system utilizes the tool recognition model to know what tool the user is interacting with. Once the students have taken the tools and leave the makerspace, the system will register the borrowing and put this confirmed record in the database. During the post-borrow stage, students can check the app for return due date. The mobile app will also notify students if there exist overdue tools. To return the tools, students have to log in the app at the zone again. At the return collection counter (which is next to the QR code login zone), the students are asked to show the tool to the camera and put it into a return box. The returning is recorded and the inventory management is done by the system.

2.2.3 Methodology

As shown above, 3 major components have to be implemented, backend system, SQL database, and mobile application.

The backend system mainly receives the camera footage from IP cameras as input. Occasionally, it will receive a login request from students. To take live stream data, Kafka message engine will be served as the main bus between the backend system and the data source. Kafka is a high-throughput, low-latency, open-source platform for handling real-time data feeds[12]. Thus, Kafha is a feasible technology to centralize all data input into the same place.

All trained computer vision models are saved locally in the backend server. The backend system evaluates the video stream received against the models to detect any persons and tools taken.

When users login in front of the QR code reader, the camera footage on the wall will be used by the backend system to detect the students and identify the coordinates of students within the makerspace area. The information will be mapped to the most recent login request. As long as the students are still captured by the cameras, the system will keep track of the movement.

When tools are taken by students, the cameras on the tool boards will see the movement of detected tools. The backend system evaluates the video against the models to validate what tools are being borrowed. The corresponding tool information will be pushed to students via the "shopping cart" page of the mobile app. After the users left the zone, the transaction will be confirmed.

For returning, logged-in students are asked to show the tools to the camera and put them into a return box. After the video footage are evaluated to confirm what tools are shown, the backend system will mark down that the students have returned the tool.

A SQL database is used to store all transactions (borrowing and returning) and login records, users information. A crown job that goes through all overdue borrowing will be done every day. Notification emails are sent to users' HKU email address if there is overdue borrowing. Both the backend system and the SQL database will be hosted on a VM via Google Cloud.

The cross-platform mobile application contains 3 major components, QR page, cart page, and record page. First, QR code page displays a QR code which contains student information to be scanned for entering Inno Wing makerspace area in the pre-borrow stage. Second, the cart page shows what tools are seen as taken by the system. Third, the record page shows the due date of tools and highlight any overdue borrowing.

2.2.4 Challenge

The online pre-trained person detection model is used instead of developing a brand new model. The accuracy cannot be ensured if it is used in Inno Wing makerspace area because of the ambient setting, for example, surrounding lighting and camera positioning. With this difficulty foreseen, if the accuracy is indeed lower than needed after fine-tuning, video data preprocessing approach would be adopted to enhance the data quality for better prediction result.

3 Budget Plan

Item Name	Price (on 30 Sep, 2019)
Google Cloud & Firebase	HKD 4800 (12 months)
IP Camera * 4	HKD 2500

4 Schedule and Milestones

September 2019	● Data Mining for Tool Images ● Literature Review ● Detailed Project Plan ● Building Project Webpage
29 September 2019	**Deliverables of Phase 1 (Inception)**
October 2019 - January 2020	● Build Machine Learning Model ● Image Preprocessing ● Person Object Detection System ● RESTful API Design & Implementation ● Database Design ● Enhance ML Model for more kinds of tools ● Literature Review
13-17 January 2020	**First Presentation**
February 2020	● Preliminary Implementation ● Hardware Setup ● System Integration

	● System Enhancement ● Detailed Interim Report
2 February 2020	**Deliverables of Phase 2 (Elaboration)**
March 2020 - April 2020	● Internal Demo ● Finalized Tested Implementation ● Final Report
19 April 2020	**Deliverables of Phase 3 (Construction)**
20-24 April 2020	**Final Presentation**
May 2020	● Project Exhibition
5 May 2020	**Project Exhibition**

5 Reference

[1] “Tam Wing Fan Innovation Wing,” Innovation Wing. [Online]. Available: http://innowing.hk/. [Accessed: 29-Sep-2019].

[2] “Amazon Go” *Amazon*, [Online]. Available: https://www.amazon.com/b?ie=UTF8&node=16008589011. [Accessed: 29-Sep-2019].

[3] R. Williams, “Amazon Go: what it's like to go shopping with no queues, no cashier and no till,” *inews*, 26-Sep-2019. [Online]. Available: https://inews.co.uk/news/technology/amazon-go-shop-seattle-supermarket-no-till-637302. [Accessed: 29-Sep-2019].

[4] M. Tillman, “What is Amazon Go, where is it, and how does it work?,” *Pocket-lint*, 18-Feb-2019. [Online]. Available: https://www.pocket-lint.com/phones/news/amazon/139650-what-is-amazon-go-where-is-it-and-how-does-it-work. [Accessed: 29-Sep-2019].

[5] “Grand Opening of CKC18 at Cheung Kong Center,” *A.S. Watson Group | A member of CK Hutchison Holdings*, 31-May-2018. [Online]. Available: https://www.aswatson.com/grand-opening-of-ckc18-at-cheung-kong-center-bringing-brand-new-shopping-experience-to-customers. [Accessed: 29-Sep-2019].

[6] S. Shetty, S. Shetty, V. Davis, A. Joshi, R. Gall, Sunith ShettyData Science, and Bhagyashree R, “Why TensorFlow always tops machine learning and artificial intelligence tool surveys,” *Packt Hub*, 20-Sep-2018. [Online]. Available: https://hub.packtpub.com/tensorflow-always-tops-machine-learning-artificial-intelligence-tool-surveys. [Accessed: 29-Sep-2019].

[7] Tensorflow, “tensorflow/models,” *GitHub*, 16-Jul-2019. [Online]. Available: https://github.com/tensorflow/models/tree/master/research/object_detection. [Accessed: 29-Sep-2019].

[8] Tensorflow, “tensorflow/models,” *GitHub*. [Online]. Available: https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md. [Accessed: 29-Sep-2019].

[9] “Open Images Challenge object detection evaluation,” *Googleapis*. [Online]. Available: https://storage.googleapis.com/openimages/web/evaluation.html#instance_segmentation_eval. [Accessed: 29-Sep-2019].

[10] Tzutalin. “Tzutalin/LabelImg.” *GitHub*, 28 Aug. 2019, https://github.com/tzutalin/labelImg.

[11] J. Wang and L. Perez, "The Effectiveness of Data Augmentation in Image Classification

using Deep Learning", 13-Dec-2017. [Online]. Available: https://arxiv.org/pdf/1712.04621.pdf. [Accessed: 29-Sep-2019].

[12] Apache, “apache/kafka,” *GitHub*. [Online]. Available: https://github.com/apache/kafka. [Accessed: 29-Sep-2019].