2019 Trimester 2

**MTRN4230**

**Robotics**

**Group Assignment - Computer Vision (Decoration)**

Jonathan Loong z5014751

Sub-Report Workflow for Computer Vision - Decoration

**1. Preliminary Research**

**i. Quirkle Block ML Research/Ideas**

The major task for the Computer Vision Engineer (Decoration), was to reliably execute object recognition on Quirkle blocks. In order to achieve this, the development of a Quirkle O**bject Detection** program was required. This can be broken up into two parts, requiring ‘image classification’ and ‘object localisation.’ In the context of the CAKE Proof of Concept:

* **Image Classification** – prediction of which Quirkle block are needed and assigning a class ‘labels’ to them (eg: Criss cross, circle).
* **Object Localisation** – the customer image will contain an arrangement of blocks. The coordinates of the centre of each block is to be identified (and sent to robot arm)
* **Object Detection** – combination of both in real time (on conveyor)

Using MATLAB, this program can be developed in a variety of ways:

1. Transfer Learning. Fine-tuning a previously trained neural network and adjusting it for the Quirkle training database.

Advantages:

* Faster to train

Disadvantages:

* If using existing networks like googleNet, the Quirkle blocks may not work well

2. R-CNN. Region-based Convolutional Neural Network. Locates many regions of interest for object classification.

Advantages:

* Allows for a selective search of objects of interest
* Relatively simple method for object localisation and classification

Disadvantages:

* Slow. Generates around 2000 proposed regions in any input image

3. YOLO (You Only Look Once). Predicts and labels bounding boxes in a single evaluation. Simple structure in a single network.

Advantages:

* Possible to use in real time (up to 45 fps)
* Pretrained using the imageNet dataset

Disadvantages:

* Multiple/overlapping bounding boxes are a possibility
* Lower accuracy in relation to object localisation (might be issue with final decoration placement). YoloV2 may have some better performance/accuracy

**ii. Proposed ML Algorithm – YOLOv2**

Based on prior research and the capabilities of the MATLAB Deep Learning Toolbox, it was decided to develop the Quirkle Object Detection program using YOLOv2 utilizing transfer learning. YOLOv2 contains a feature network (pre-trained CNN) & detection network (smaller CNN specific to YOLO).

**iii. Outline Decoration Detection Pipeline**

Here is a flowchart to present the development of a Quirkle object detection program using both the cameras in the robot cell and the conveyor camera. In particular, the interaction/relationships between the two cameras and the rest of the robot cell was considered.

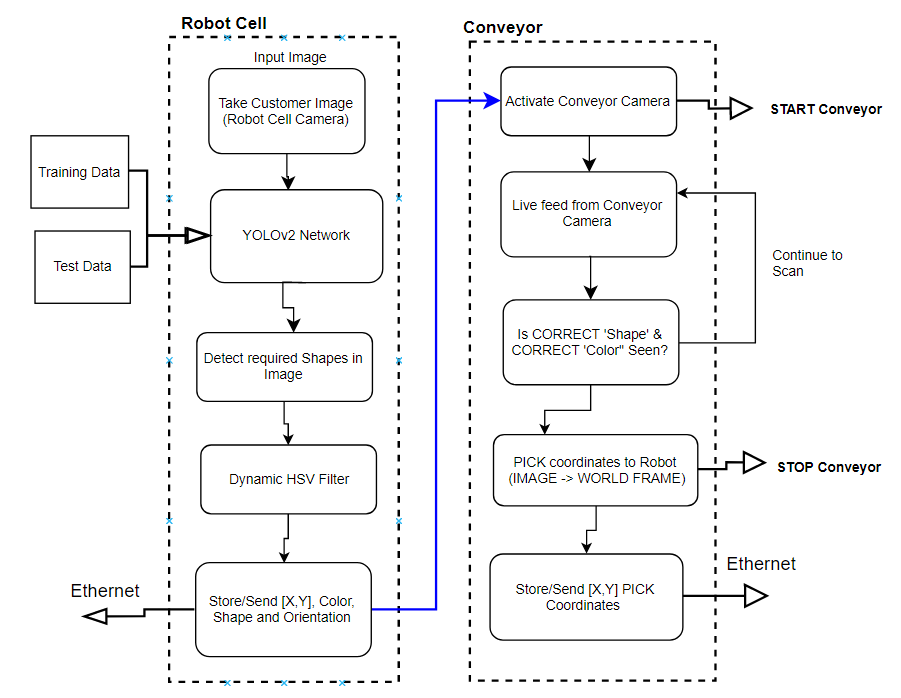


Figure 1: ML/CV Pipeline

**2. Calibration of Robot Cell and Conveyor Camera**

**Intrinsic**

The intrinsic parameters for the robot cell camera are concerned with the focal length and distortion properties of the camera. Calibration required performing a projective transformation to relate the camera frame to the image plane.

1. Measure width of one square side of the checkerboard pattern (length = 25mm x 25mm )
2. Obtain 10-20 images at the distance between camera and OOI (height from robot camera to table top)
3. Place checker board at angles less than 45 degrees relative to camera plane
4. Different orientations relative to the camera
5. Using Single Camera Calibration Toolbox in MATLAB

After the initial round of calibration, the average re-projection error was found to be 0.11 pixels. This error is the distance in pixels for which the detected points differs from the re-projected points. After the removal of the two outliers, the error was reduced to 0.09 pixels.

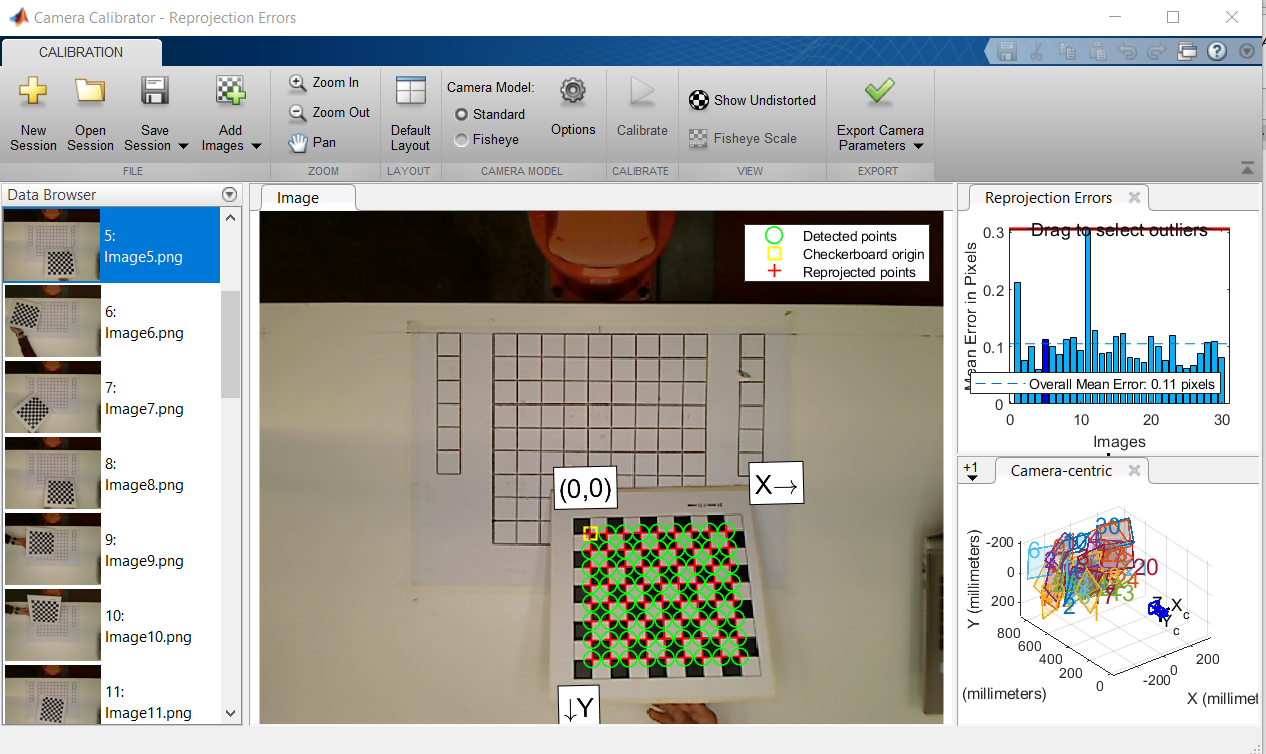


Figure 2: Initial Intrinsic Calibration

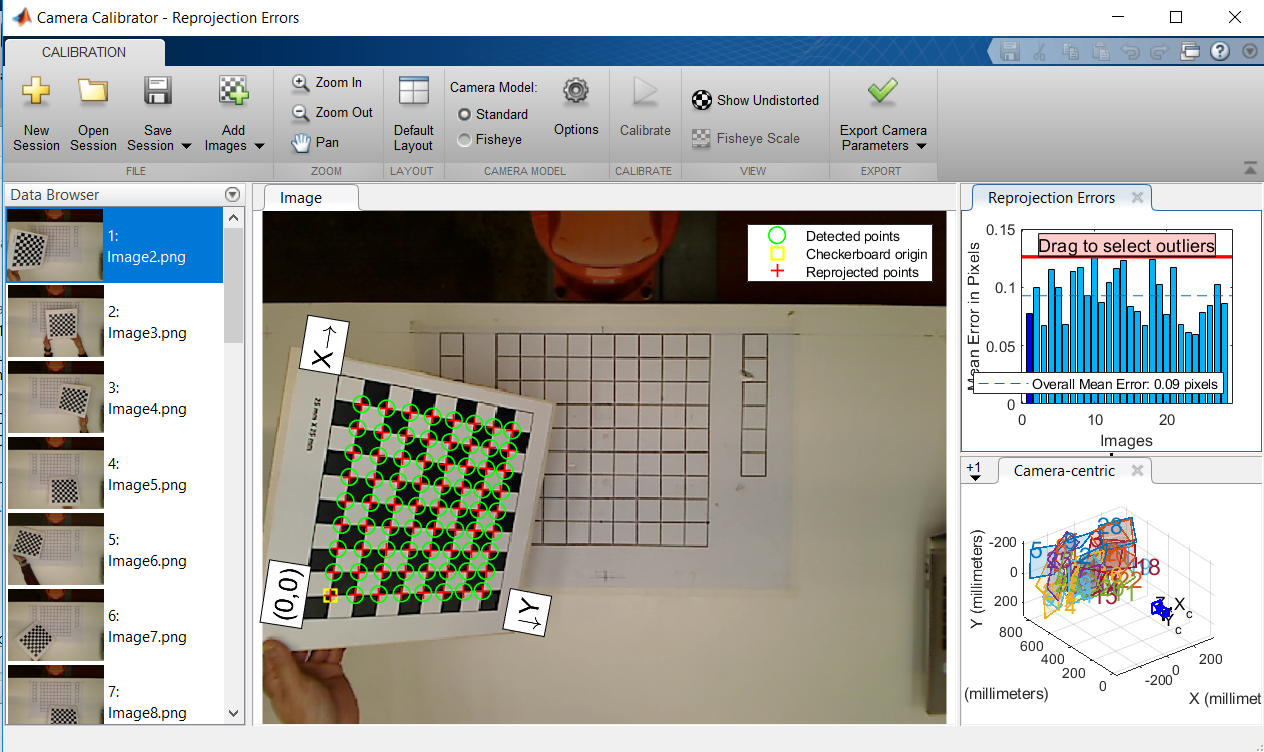


Figure 3: Final Intrinsic Calibration

The intrinsic matrix K (Calibration matrix) was found to be:

Kint =

**Extrinsic**

Using the following points as control reference points, the pixel values for each (u,v) were estimated using the following 3D points noted as (Xw,Yw,1) as listed in Table 1:

Table 1: World and Camera Coordinates for Ref Points

|  |  |
| --- | --- |
| **3D Point (World Frame)** | **2D Point (Camera Frame)** |
| T2 (175, -520, 149) | (7,289) |
| T1 (175, 0, 149) | (800,289) |
| T3 (175, 520, 149) | (1597,289) |
| T4 (548.6, 0, 149) | (800,855) |
| T5 (548.6,520,1409) | (1597,855) |

The correspondence (homography) between these reference points in the image plane (u,v) and the world coordinate frame (X,Y,Z) enables the determination of the extrinsic matrix for the robot cell camera.

Translation vector – transformation from world coordinate to camera coordinate

TranslationVector =

Rotation vector – position of camera in world coordinate frame

RotationMatrix =

**3. Creating Quirkle Detector Network**

A combination of ‘Ink Writing’ (Letters) and ‘Chocolate Blocks’ (Colored Quirkle blocks) make up the customer’s sample pattern. Focus for the decoration part was on the reliable detection of the correct Quirkle block and the required position and orientation.

**3a. Training Data**

To create the Quirkle block detector, training images were required for Machine Learning. These blocks will be required to be successfully detected at both the robot cell and the conveyor. Using the ‘Image Labeler’ App, a ROI (bounding box Region of Interest) was created for the variety of blocks. These created the database for the ‘ground truth’ images to be passed into the Machine Learning network (see Figure 4).

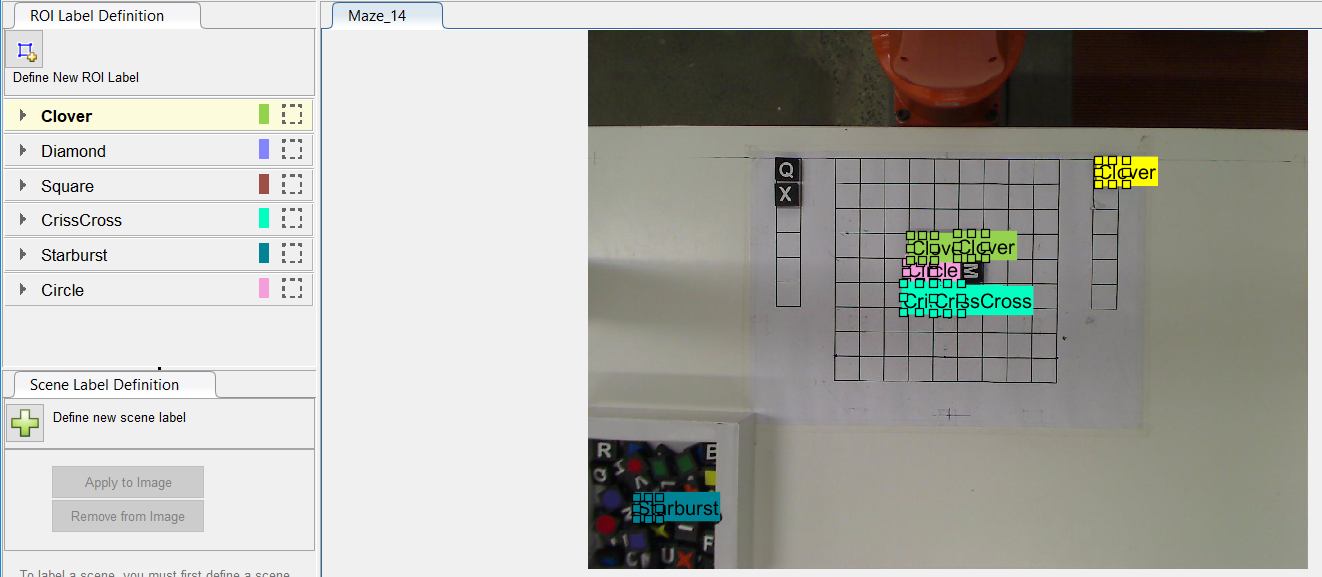


Figure 4: Labelling Ground Truth Image

**3b. Creating the YOLOv2 Network**

As detailed in Section 1, a YOLOv2 Detector was tested for the prototype cake decoration system. This is a single stage object detection network as outlined in Figure 5:

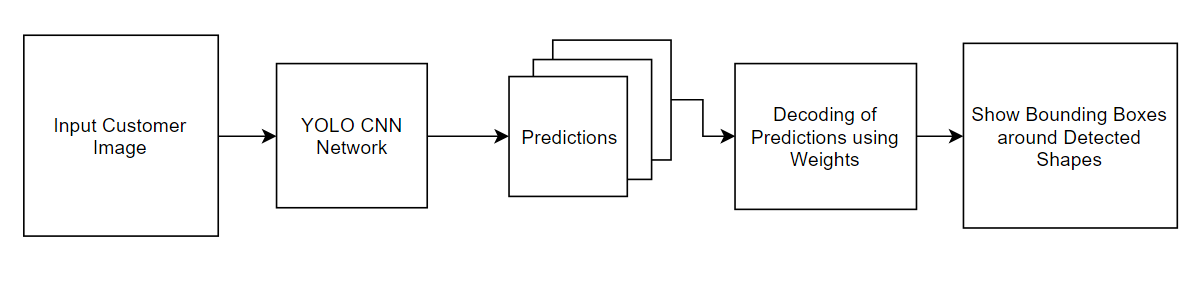


Figure 5: YOLOv2 Flowchart

This type of object detector uses the concept of anchor boxes to classify objects. These were utilized to identify which Quirkle block was found in the image. For each bounding box, the following was predicted:

* Dimensions of the bounding box
* Confidence of having an object in the box
* Probabilities of which class the object belongs to

To speed up the training process, the concept of transfer learning was used. For the feature extraction part of the network, a ResNet-50 network was used. The detection network consisted of YOLOv2 specific layers.

**i) Sample Training Data**

Labelled ground truth images of Quirkle blocks taken from the Conveyor and Robot Cell cameras at 1600 x 1200 resolution (for consistency) (see Figure 6).

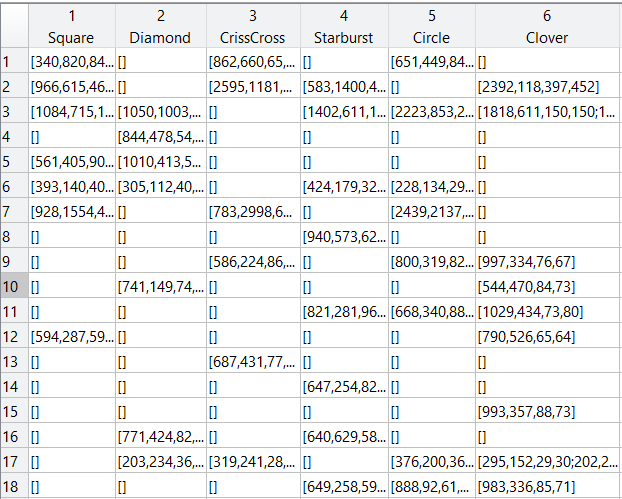


Figure 6: Sample Training Data

**ii) Training Loss**

The purpose of the Quirkle Detector is to reliably and quickly detect and classify which classes/patterns of Quirkle blocks are present in the current image frame. How accurate the model is in training stage.

Figure 7: Graph of Training Loss

**iii) Evaluating YOLOv2 Quirkle Detector**

Sample test image

Figure 8: Result of YOLOv2 on Test Image

Precision

Figure 9: Graph of Network Precision over Time

**4. Image Processing on Customer Image**

After the YOLOv2 network, the Quirkle Detector is to reliably and quickly perform Image Localisation and Image Classification

**4a Localisation and Classification for Blocks**

d

**4b. Pose for Blocks**

The next step was deriving the orientation and locations for the required blocks.

Location = HSV thresholding and centroids (how many of each color are in image)

Which shape, which color? Correlate localisation using the bounding box from the ML network and computer vision

**5. Sending Info to Robot Arm**

Sending world coordinates of blocks (PLACE coordinates) and orientation to robot arm (Ethernet)

**6. Detecting Blocks on Conveyor**

Dynamic object recognition in real-time for blocks incoming from conveyor. YoloV2 Detection network. Send PLACE Coordinates to robot arm