

1708165 presentation 2

by Adam Bedner

Submission date: 20-May-2022 07:01PM (UTC+0100)

Submission ID: 180991591

File name: 13694_Adam_Bedner_1708165_presentation_2_179632_52054555.pdf (1,003.23K)

Word count: 1269

Character count: 5841



colegsirgâr

Presentation 2

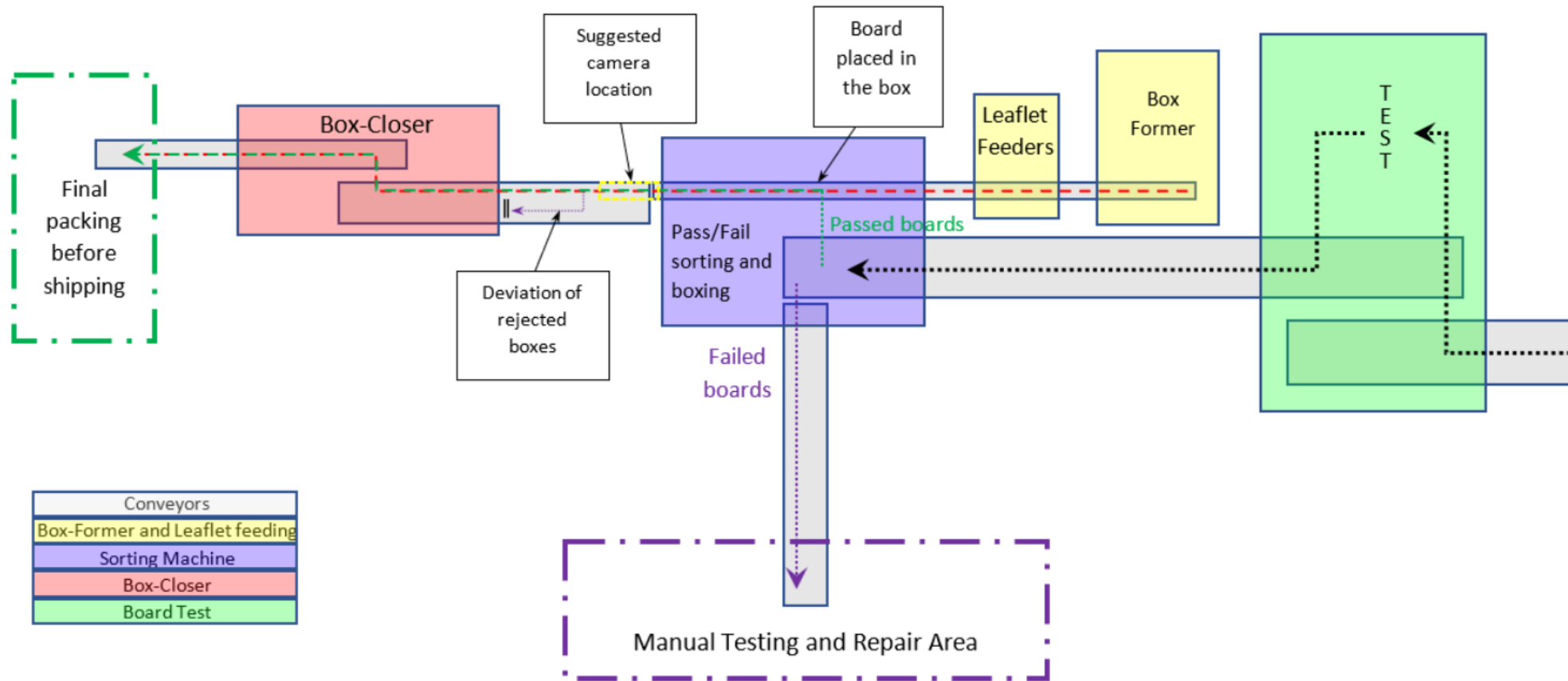
Project Design, Implementation and Evaluation

Name: Adam Bedner

Student Number: 1708165

Date: 20/05/2022

Line Overview

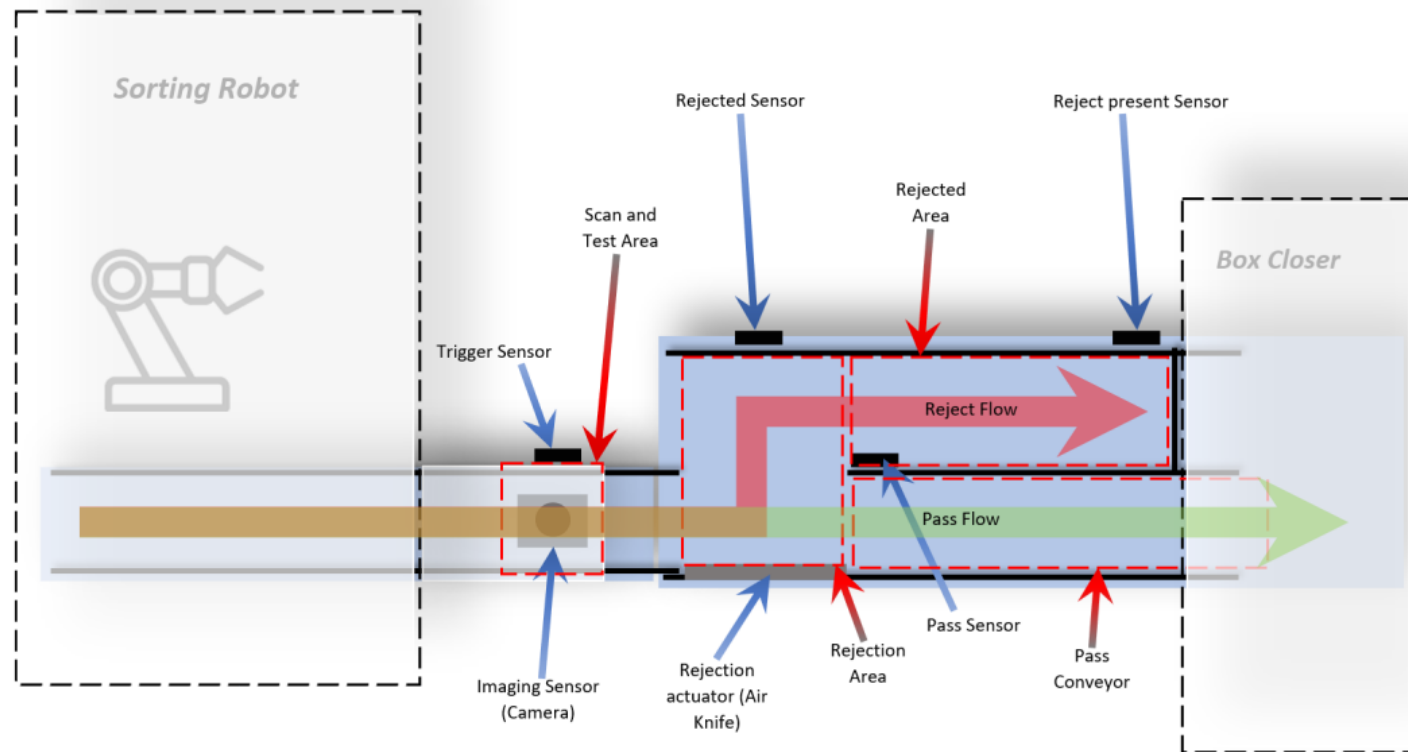


Original Functional Design Specifications

Spec #	Specification	Description
1	Misplaced product recognition	It must recognise a product that has been misplaced in such a way that it is sticking out of the edges of the box (in any direction).
2	Compatibility with different models.	The ability to work with more than one model. Those vary slightly in the component type and position. The overall size of the product remains the same across all models. It is not required for the system to reject wrong products; different systems are responsible for that.
3	Cooldown time <3s	In order for the system to keep up with the line, it must scan every 3 seconds or less. Or 1200 scans an hour.
4	Jam condition detection	The system will be capable of detecting a jam up product boxes (abnormal stop of ABC)
5	Misfire condition detection	Detection of an actuator misfire, causing a reject to carry on to ABC. If a condition is detected, a stop signal will be sent to ABC.
6	IIOT compatibility	Network connectivity for IIOT. It must be capable of transmitting data about the passes and fails, as well as sending the images of rejected products.
7	Minimal machine modification for installation	The system must not require extensive machine modifications for installation.
8	No interference with existing safety	The system must not interfere with the existing safety system of the machine. Those are Cat 4 Safety Circuits.
9	Cat 4 safety compliant	The system must be compliant with Cat 4 safety to match the safety rating of the existing machine.

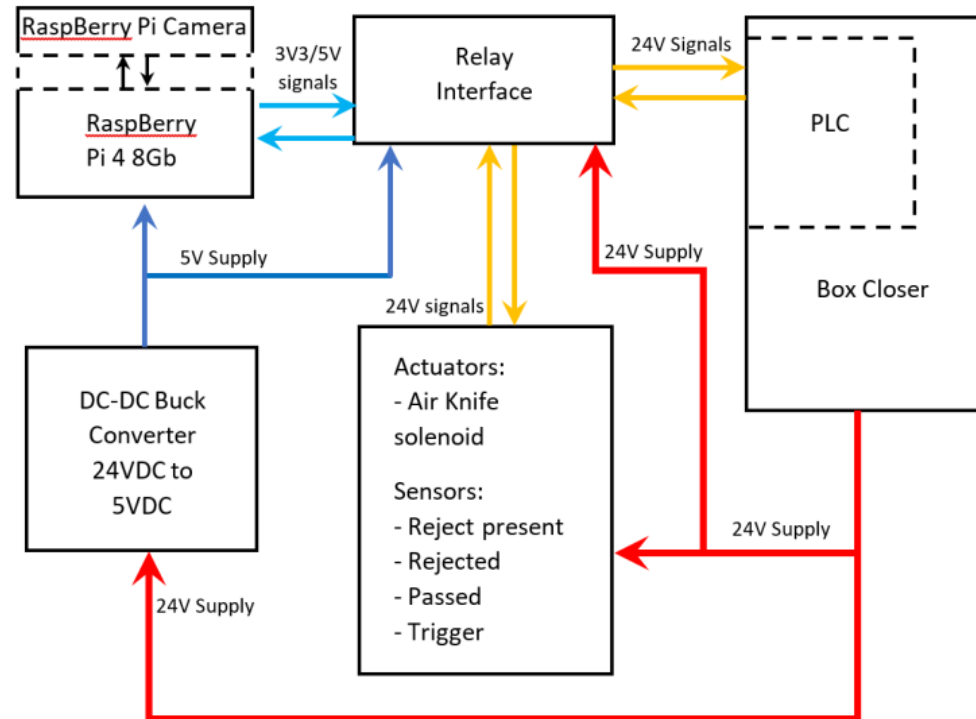
Modified Functional Design Specifications

Spec #	Specification	Description	Change	Reason for Change
1	Misplaced product recognition	It must recognise a product that has been misplaced in such a way that it is sticking out of the edges of the box (in any direction).	None	N/A
2	Compatibility with different models.	The ability to work with more than one model. Those vary slightly in the component type and position. The overall size of the product remains the same across all models. It is not required for the system to reject wrong products; different systems are responsible for that.	None	N/A
3	Cooldown time <3s	In order for the system to keep up with the line, it must scan every 3 seconds or less. Or 1200 scans an hour.	None	N/A
4	Jam condition detection	The system will be capable of detecting a jam up product boxes (abnormal stop of ABC (Automated Box Closer))	None	N/A
5	Misfire condition detection	Detection of an actuator misfire, causing a reject to carry on to ABC. If a condition is detected, a stop signal will be sent to ABC.	None	N/A
6	IIOT compatibility	Network connectivity for IIOT. It must be capable of transmitting data about the passes and fails, as well as sending the images of rejected products.	Specification no longer required	Customer no longer requires IIOT on the system.
7	Minimal machine modification for installation	The system must not require extensive machine modifications for installation.	None	N/A
8	No interference with existing safety	The system must not interfere with the existing safety system of the machine. Those are Cat 4 Safety Circuits.	None	N/A
9	Cat 4 safety compliant	The system must be compliant with Cat 4 safety to match the safety rating of the existing machine.	None	N/A

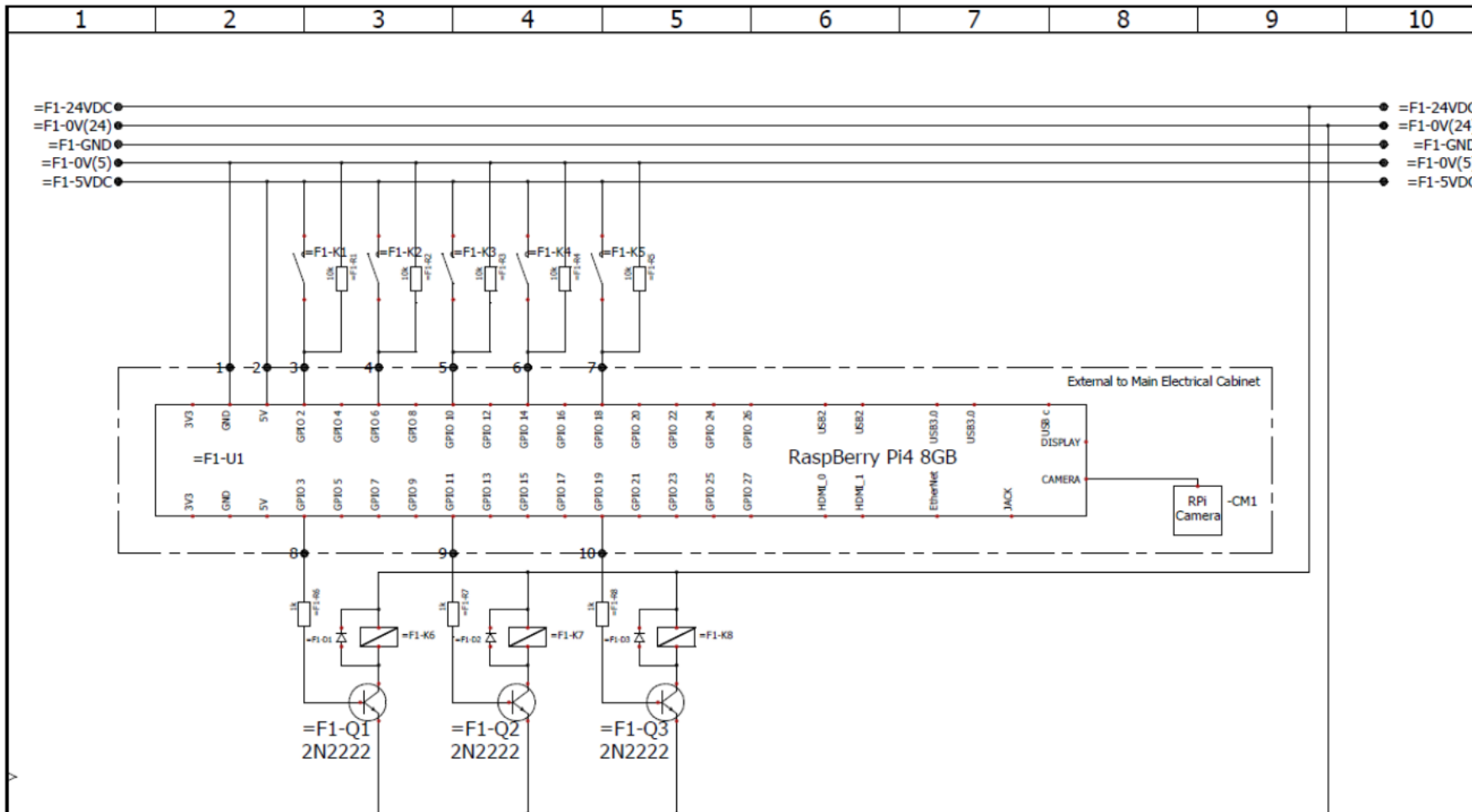


System Overview

Electrical Block Diagram



Part of Electrical Wiring



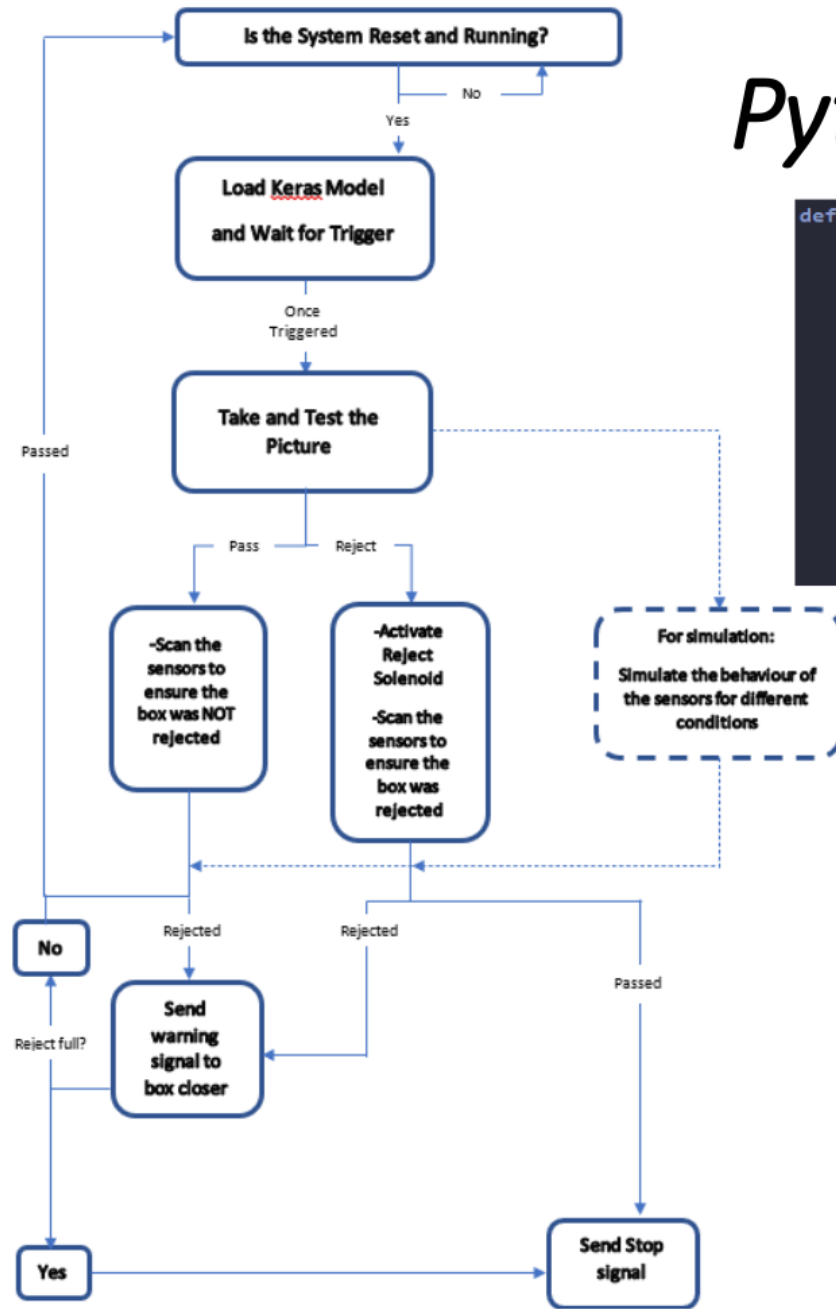
Adam Bedner	Electrical Drawing	0	30/04/2022	Adam		REVISION
						0
						SCHEM
		REV.	DATE	NAME	CHANGES	03
CONTRACT:	LOCATION: +L1 Main Electrical Cabinet	User data 1			User data 2	

Python Code Design

```
def PassScan ():
    c.MBit.BCall = True
    Behaviour = threading.Thread(target=B.Behavior_Selection)
    Behaviour.start()
    print("PassScan Started", c.Sensor.Passed, c.MBit.BCall,)
    sleep(0.1)
    while c.Sensor.Passed == True:
        print("in pass while loop", c.MBit.PassScan)
        sleep(0.1)
        c.MBit.PassScan += 1
        if c.MBit.PassScan == 30 or c.Sensor.Passed == False:
            print("PassScann finished at:", c.MBit.PassScan)
            c.MBit.BCall = False
            break
```

```
import tensorflow.keras
import cv2
import numpy as np
import time
import defs
import classes as c
from time import sleep
# from picamera import PiCamera
```

pycache
 Th B
 Th classes
 Th CSS
 Th defs
 keras_model_pi.h5
 pass
 reject



```
def PassDecisionMaker ():
    if c.Sensor.CloserStop == True:
        print("Closer Stopped!")
        c.MBit.PassScan = 0
        c.MBit.ErrorCount += 1
    elif c.MBit.BError == True:
        sleep(0.25)
        print(
            """
            Program stopped due to Behavior_Selection Error!
            """
        )
        c.MBit.SystemRunning = False
    elif c.MBit.PassScan == 0:
        c.MBit.ErrorCount += 1
        #c.Actuator.CloserStop = True
        print(
            """
            Good box not in closer!!
            """
        )
    elif 5 < c.MBit.PassScan < 25:
        print(
            """
            Box passed successfully
            """
        )
        c.MBit.Pass += 1
        c.MBit.PassScan = 0
    elif c.MBit.PassScan >= 25:
        if c.Sensor.CloserStop == False:
            c.Actuator.CloserStop = True
            print(
                """
                Closer Error: Input to closer jammed and closer not stopped!!
                Closer stopped!
                Call technician
                """
            )
            c.MBit.ErrorCount += 1
            c.MBit.PassScan = 0

        if 0 < c.MBit.RejectScan < 25:
            c.Actuator.CloserWarning = True
            c.MBit.RejectScan = 0
            print(
                """
                Good Box Rejected!!
                Investigate and place on closer input if ok
                Call technician if fault repeats
                """
            )
        )
        c.MBit.ErrorCount += 1
    elif c.MBit.RejectScan >= 25:
        c.Actuator.CloserStop = True
        c.MBit.RejectScan = 0
        print(
            """
            Reject shute full!!
            Empty shute before restarting the machine
            """
        )
        c.MBit.ErrorCount += 1
```

Keras Model Training

Training

Train Model

Advanced

Epochs: 50

Batch Size: 16

Learning Rate: 0.001

Reset Defaults

Under the hood

Class 1

1000 Image Samples

Webcam Upload

Class 2

1000 Image Samples

Webcam Upload

Add a class

Preview

Export Model

Input ON Webcam

Output

Class 1

Class 2 100%

Confusion Matrix

Class 1	0	150
Class 2	0	150
	Class 1	Class 2

scaleCount 150 0

Accuracy per class

CLASS	ACCURACY	# SAMPLES
Class 1	0.00	150
Class 2	1.00	150

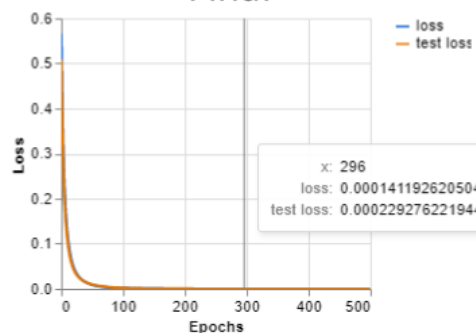
Keras Model Training

Loss

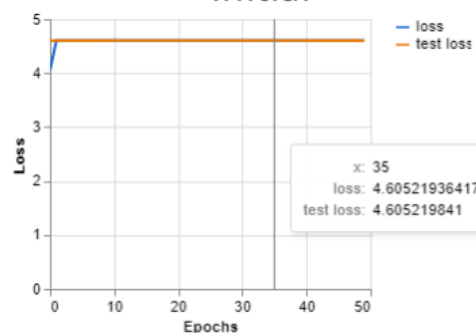
Loss is a measure for evaluating how well a model has learned to predict the right classifications for a given set of samples. If the model's predictions are perfect, the loss is zero; otherwise, the loss is greater than zero.

To get an intuitive sense of what this measures, imagine you have two models: A and B. Model A predicts the right classification for a sample but is only 60% confident of that prediction. Model B also predicts the right classification for the same sample but is 90% confident of that prediction. Both models have the same accuracy, but model B has a lower loss value.

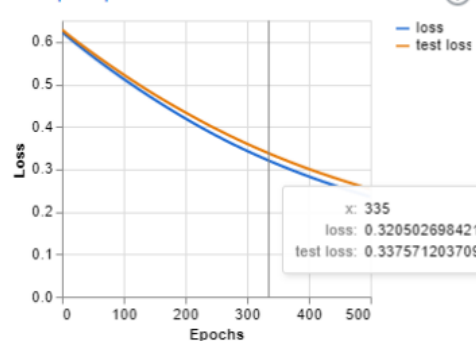
Loss per epoch Final



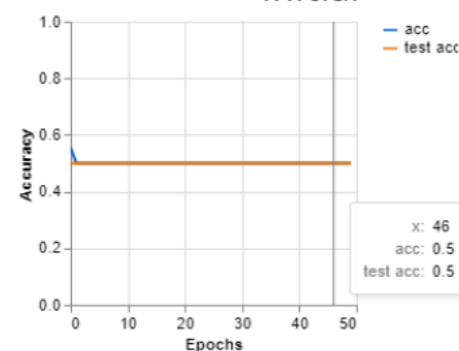
Loss per epoch Initial



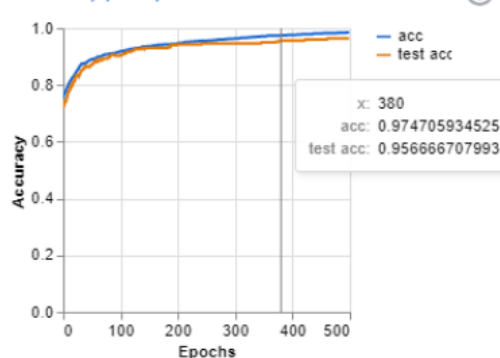
Loss per epoch



Accuracy per epoch Initial



Accuracy per epoch

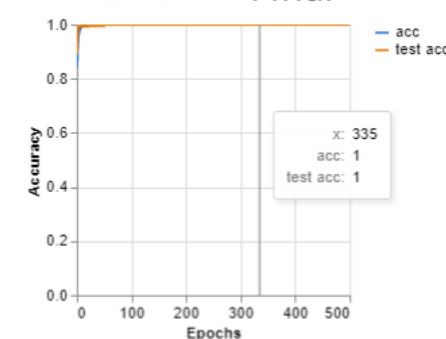


Accuracy

Accuracy is the percentage of classifications that a model gets right during training. If your model classifies 70 samples right out of 100, the accuracy is $70 / 100 = 0.7$.

If the model's prediction is perfect, the accuracy is one; otherwise, the accuracy is lower than one.

Accuracy per epoch Final



Keras Model Training

Training Parameters and Results

	Parameters			Results					
	Epochs	Batch Size	Learning Rate	Accuracy per class		Accuracy per epoch		Loss per epoch	
				Class 1	Class 2	Acc	Test acc	Loss	Test loss
Set 1	50	16	1x10 ⁻³	0.00	1.00	0.5	0.5	4.605	4.605
Set 2	500	32	1x10 ⁻¹⁷	1.00	0.93	0.974	0.956	0.320	0.337
Set 3	500	16	1x10 ⁻⁵	1.00	1.00	1	1	1.4x10 ⁻⁴	2.2x10 ⁻⁴

Training Parameters and Confusion Matrix Comparison

			Prediction					
			Set 1		Set 2		Set 3	
			Cls 1	Cls 2	Cls 1	Cls 2	Cls 1	Cls 2
Class	Set 1	Cls 1	0	150				
		Cls 2	0	150				
	Set 2	Cls 1			150	0		
		Cls 2			11	139		
	Set 3	Cls 1					150	0
		Cls 2					0	150

Simulation and testing

Testing table of System Simulation				
Test #	Specification	Testing method	Expected Result	Actual Results
1	Misplaced product recognition	Pass and Reject pictures fed into the keras model from local HDD using code. Camera code must be commented out.	Reject of the misplaced products regardless of the severity of misplacement.	For pass boxes: - Pass prediction >0.8 - Reject prediction < 0.2 For reject boxes: - Pass prediction < 0.2 - reject prediction > 0.8
2	Compatibility with different models.	The photos used must be different from the ones used for model training	Same result as for test #1 for every model.	For pass boxes: - Pass prediction >0.8 - Reject prediction < 0.2 For reject boxes: - Pass prediction < 0.2 - reject prediction > 0.8
3	Cooldown time <3s	Monitor the test times during simulations. This will be displayed after each test in the terminal	The test times should be below 5s (except jam condition due to additional 3s IO scan time)	Standard running cycle time < 4s With exception for jammed conditions where cycle time < 7s
4	Jam condition detection	Follow the instructions on the simulation terminal, entering the correct behaviour number for the test	IO scan times out after 3s alerting of the jam and stopping the box closer.	In both cases (pass and reject) the PassScan module was activated and timed out at 3s; stopping the box closed.
5	Misfire condition detection		IO scan detects a trigger from the wrong sensor and stops the box closer	PassScan module activated, detecting a box going past; stopping the box closer.

Test examples

Test #1 & 2

The output from the terminal when loading a reject box:

```
waiting for trigger
Enter 0 to load a pass board image
Enter 1 to load a reject board image
```

Pass prediction is: 0.00019944199

Reject prediction is: 0.9998006

The output from the terminal when loading a pass box:

```
waiting for trigger
Enter 0 to load a pass board image
Enter 1 to load a reject board image
```

Enter selection to trigger test: 0

Pass prediction is: 0.997331

Reject prediction is: 0.0026690152

Test #4

Reject box:

```
Box failed test
behaviours:
1. Box rejected
2. Box rejected and shute full after rejection
3. box not rejected
4. box closer not running
5. box closer jammed
```

enter behaviour's number: 5

PassScan finished at: 30

```
Closer Error: Input to closer jammed and closer not stopped!!
Closer stopped!
```

Pass box:

```
Box passed test
behaviours:
1. Box rejected
2. Box rejected and shute full after rejection
3. box not rejected
4. box closer not running
5. box closer jammed
```

enter behaviour's number: 5

PassScan finished at: 30

```
Closer Error: Input to closer jammed and closer not stopped!!
Closer stopped!
```

Test #3

```
test completed in: 4.19
test completed in: 2.29
test completed in: 2.24
test completed in: 3.63
test completed in: 2.62
test completed in: 6.5
test completed in: 5.42
test completed in: 6.69
```

Test #5

```
Box failed test
behaviours:
1. Box rejected
2. Box rejected and shute full after rejection
3. box not rejected
4. box closer not running
5. box closer jammed
```

enter behaviour's number: 3

PassScan finished at: 7

```
Actuator misfire!
Closer Stopped.
```

The project's total cost

- Initially predicted cost: £200
- Final cost: £2080.16
- Final cost variation due to not considering the price some of the components as well as the cost of man-hours required for the project.

Entry No.	Item Description	Price/Unit	Qty	Price
1	Aestetically damaged Pi4 (Scrap)	£0.00	1	£0.00
2	Aestetically damaged HQ Camera (Scrap)	£0.00	1	£0.00
3	LAN Camera Extension Kit	£52.00	1	£52.00
4	5V 3A Dual USB Step-Down converter	£8.20	1	£8.20
5	Low profile cooling fan	£9.00	1	£9.00
6	Tripod Swivel Ball Adapter	£2.00	1	£2.00
7	Mounting Plate for HQ Camera	£5.00	1	£5.00
8	Phoenix NO relay	£6.17	8	£49.36
9	10k Resistors	£0.16	5	£0.80
10	1k Resistors	£0.16	3	£0.48
11	PN2222ATA NPN Transistor	£0.21	3	£0.62
12	Curve B 1A 2Pole MCB (siemens)	£47.75	1	£47.75
13	Curve B 3A 2Pole MCB (siemens)	£47.75	1	£47.75
14	Omron PhotoElectric Sensor	£92.30	4	£369.20
15	Labour (Excluding instalation and commisioning)	£12.00	100	£1,200.00
16	Instalation and commisioning (Estimated on hours required)	£12.00	24	£288.00
17				£0.00
18				£0.00
19	Total Cost			£2,080.16

Evaluation and the wider issues

- The system works as intended in reference to FDS, at least is Simulation
 - The final cost is as lot higher than initially predicted
 - Compatibility and Stability issues due to the nature of microcontrollers
 - Cyber-Security on open-source based systems
-
- Microcontrollers such as Pi are a cheaper alternative for machine automation compared to PLCs, but require more time and experience to work with them.

1708165 presentation 2

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

59/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

GRADING FORM: NEW CSG-RUBRIC

ADAM BEDNER

59

1. ENGAGEMENT WITH LITERATURE

 Section 3: Use of technical literature in final design solution: Some evidence presented 3/5

3

2. KNOWLEDGE AND UNDERSTANDING

 Section 1: Introduction: Good recap of project background aims and objectives. Section 2: FDS: Modifications explained for completion of the project Section 3: Advanced knowledge and critical

19

understanding demonstrated in the design of the project: This is clearly evident for the software aspects of the project design and hardware interface shown as components, but no PCB design shown. All sections: Response to questioning: Reasonable answers given but lacking in succinctness. 19/30

3. COGNITIVE AND INTELLECTUAL

 Section 5: Testing and analysis of result: Good evidence presented for simulation only. Section 6: Critical evaluation of the project outcomes and comparison with the FDS: A good critical analysis and an honest evaluation. Section 8: Overall conclusions and recommendations: Good overall conclusions but an admission that this was not the best solution 18/30 **18**

4. PRACTICAL APPLICATION

 Section 3: Formulation of innovative and creative design solution to the project problem: Excellent software solutions but hardware interface is lacking in professionalism. Section 4: Deployment method, use of material, tools, equipment and techniques in the construction of the project: Good methods used for software but hardware was not designed nor build Section 4: Quality of the final product or system: Good only for software Section 5: Use of test equipment and presentation of results of the project: Testing was done through simulation only. Section 7: Management techniques and planning, comparison with initial plan: Poor evidence of management, planning and comparison with original plan 13/25 **13**

5. TRANSFERABLE SKILLS

 All sections: Quality of the presentation slides: Good quality slides All sections: Quality of the delivery style: Not the best style, nor the best delivery. 6/10

6