DEVELOPMENT OF AN ANN-BASED DEFECT DETECTION SYSTEM FOR PROCESS QUALITY OPTIMISATION IN BOTTLING INDUSTRY

BY

EMUOWHOCHERE OGHENEVWEGBA THANKGOD

(13CM015542)

SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING COVENANT UNIVERSITY, OTA OGUN STATE.

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE AWARD OF THE MASTERS OF ENGINEERING DEGREE M.Eng (Hons.) IN MECHANICAL ENGINEERING.

SUPERVISOR: DR. ENESI Y. SALAWU

2022

- **DECLARATION**

I, Emuowhochere Oghenevwegba Thankgod (13CM015542), declare that this project on "Development of an Artificial Neural Network Based Defect Detection System for Process Quality Optimisation in Bottling Industry", is an original work done by me under the supervision of Dr. Enesi Y. Salawu, the department of Mechanical Engineering covenant university, Ota

EMUOWHOCHERE OGHENEVWEGBA. T …….

Signature and Date

- **CERTIFICATION**

This is to certify that this project work on "Development of an Artificial Neural Network Based Defect Detection System for Process Quality Optimisation in Bottling Industry" was carried out Emuowhochere ThankGod Oghenevwegba; with the Matriculation Number: 13CM015542.

Dr. Enesi Y. Salawu: …….

(Supervisor) Signature and Date

Prof. Joshua O. Okeniyi: ...…….……

(Head of Department) Signature and Date

- **DEDICATION**

This research project is dedicated first to “THE ALMIGHTY GOD” for His enabling strength He bestowed on me in completing this work. Secondly to myself Emuowhochere ThankGod Oghenevwegba

- **ACKNOWLEDGEMNT**

I am most grateful to God Almighty, the sole provider of knowledge, wisdom, love, mercy and grace for his protections throughout the period of the programme. I sincerely appreciate my parents, Mr. and Mrs. Akpoviri Benson, for the unquantifiable love and financial assistance during this period. I also want to appreciate my supervisor, Dr. Enesi Y. Salawu who offered timely criticism and corrections that led me through the various stages of this project. May God bless you all in Jesus' name, Amen.

- **CONTENT** **TABLE OF CONTENTS**

Pages

DECLARATION **i**

CERTIFICATION **ii**

DEDICATION **iii**

ACKNOWLEDGEMNT **iv**

TABLE OF CONTENTS **v**

TABLE OF PLATES **vii**

TABLE OF FIGURES **x**

LIST OF TABLES **xi**

ABSTRACT **xii**

1 1

1.1 1

1.2 4

1.3 6

1.3.1 Aim 7

1.3.2 Objectives 7

1.4. 7

1.5. 8

1.6. 9

2 11

2.1 11

2.1.1 Cons 13

2.2 13

2.2.1 Mechanical Industries 13

2.2.2 Electrical Industries 16

3 18

3.1. 18

3.1.1. Python Programming languages and ANN Libraries 18

3.1.2. 18

3.1.3. 19

3.1.4. 19

3.1.5. 19

3.1.6. 20

3.1.7. 20

3.2. 20

3.2.1. 20

3.2.2. 34

4 40

4.1. 40

4.2. 43

4.2.1. 43

4.2.2. 45

4.3. 49

4.3.1. 49

4.3.2. 50

4.4. 52

4.4.1. 52

4.4.2. 53

4.5. 54

4.5.1. 54

5 56

5.1. 56

5.2. 56

REFERENCES **58**

● LIST OF PLATES

Plates	Title of Plates	Pages
1. 1	Sophia the robot (Sophia the AI Robot, n.d.)	2
1. 2	Perceptron (Team, 2020)	3
1. 3	Feed Forward Neural Network (Team, 2020)	3
1. 4	Multi-layer Perceptron (Team, 2020)	4
1. 5	Convolutional Neural Network (Team, 2020)	4
1. 6	Raspberry pi (Bisson, 2021)	8
1. 7	Bottling Inspection sensors (S. Sridevi, 2018)	9
2. 1	The Evolution of Robotics in Engineering. *(Gao, Wanyama, Singh, Gadhrri, & Schmidt, From Industry 4.0 to Robotics 4.0-A Conceptual Framework for Collaborative and Intelligent Robotic Systems, 2020)*	14
2. 2	AI Robotic Manufacturing *(.R, 2018)*.	14
2. 3	Autonomous Vehicle Network (Mark .K., n.d.)	15
2. 4	Smart home illustration (Smart Home IoT Market: Best Products for Smart Home., n.d.)	17
2. 5	Car Interior lighting using (Stefan K., 2020)	17
3. 1	Bottle sample	18
3. 2	Raspberry pi 3 A+	19
3. 3	Raspberry pi Camera v2	20
3. 4	Female-Female Jumper wire (female-female-jumper-wires, n.d.)	20
3. 5	Image dataset for training	22
3. 6	Image dataset for training	22
3. 7	Image dataset for training	22
3. 8	Image dataset for training	22
3. 9	Image dataset for training	24
3. 10	Importing necessary libraries	24
3. 11	The class of the dataloader	25
3. 12	Python function for Saving datasets and label to csv file	26

3. 13 The __len__() method 27
3. 14 the __getitem__() method 27
3. 15 Image dataset for training 27
3. 16 Image representation of cross correlation process in CNN(*Convolutional Neural Networks Tutorial in PyTorch – Adventures in Machine Learning*, 2018) 28
3. 17 cross correlation process output (*Convolutional Neural Networks Tutorial in PyTorch – Adventures in Machine Learning*, 2018) 29
3. 18 Image representation of cross correlation process in CNN for multiple channels(*Convolutional Neural Networks Tutorial in PyTorch – Adventures in Machine Learning*, 2018) 29
3. 19 Image representation of CNN (*Convolutional Neural Networks Tutorial in PyTorch – Adventures in Machine Learning*, 2018) 30
3. 20 Importing the required libraries 31
3. 21 The class implementing the torch nn module 31
3. 22 method for feed forward 32
3. 23 Dataset Loading code 33
3. 24 Train Function 33
3. 25 Test function 34
3. 26 Importing required libraries 35
3. 27 Variable 35
3. 28 The CNN class 36
3. 29 Live Stream DataLoader 37
3. 30 Predict Function 37
3. 31 Socket Function 38
3. 32 Client-side code on Raspberry pi 39
3. 33 Orthographic representation of Raspberry pi Case 40

- **LIST OF FIGURES**

Figure	Title of Figure	Pages
4. 1	Chart of bottle quantity received and quantity used for production	42
4. 2	Quantity of images received and quantity outputted	42
4. 3	Normal Distribution of the above data	43
4. 4	The cumulative probability	43
4. 5	Training Result for 100- pixel dataset	45
4. 6	100 pixels image dataset Accuracy per training cycle	45
4. 7	100 pixels image dataset Loss per training cycle	46
4. 8	Training Result for 300- pixel dataset	47
4. 9	300 pixels image dataset Accuracy per training cycle	48
4. 10	300 pixels image dataset Loss per training cycle	48
4. 11	Train Accuracy Comparison	49
4. 12	Training Accuracy Comparison per training cycle	49
4. 13	100-pixel dataset predictions	50
4. 14	100px dataset wrong predictions	51
4. 15	300-pixel datasets prediction result	52
4. 16	300-pixel dataset wrong predictions	52

- **LIST OF TABLES**

Table	**Title of Table**	**Pages**
4. 1	300 pixels image dataset live prediction result	53
4. 2	100 pixels image dataset live prediction result	54
4. 3	300-pixels dataset live test result after upgrade	55

LIST OF EQUATIONS

Eqn	**Title of Equations**	**Pages**
1,2	300 pixels image dataset live prediction result	28
3	100 pixels image dataset live prediction result	30

● ABSTRACT

The quality of a manufactured product is a major factor that contributes to the growth, productivity and sustainability of a manufacturing firm. This is the major reason a lot of resources both human and machines in the area of quality control and quality assurance. In bottling industries, (that is firms that make use of bottles as raw material, and firms that produce bottles as end product), quality control in the production floor is also vital part of the manufacturing process and here in Africa, human labour are mostly used for the quality control process and also to save cost. The use of these human factors in tracking possible defected in the production line has led to loss of material due to fatigue, which in turn affects the productivity of the manufacturing firm. The aim of this study is to develop an CNN based defect detection system that is capable of detecting possible defected bottles in the production line. The Convolutional Neural Network model was developed with; bottles with bad labels, bottles with bad caps and, bottle that met the quality requirement. These are the defects the model is expected to detect on the production line. The materials selected for this development of this system were; a server computer, raspberry pi 3A+, raspberry camera, and the CNN trained model. The raspberry pi serves as the client side of the setup which takes photos of bottles in the production line and sends them to the server running the CNN model. The training model of the CNN model was divided into two part which were; training of the CNN model with 100-pixels dataset and training of the CNN model with 300-pixel dataset. From the result obtained, the prediction accuracy of the trained model with 100-pixel dataset was 83.333% and that of the 300-pixel dataset was 90.81654%. Running the CNN model on the raspberry pi solo leads to an average response time of 15 minutes per prediction and this is not ideal for an industrial space. Running the raspberry pi as a client-side image collector reduced the response time of the setup to an average response time of 8 sec. This response time is ideal and better than the human factor response time.

Key words: ANN, CNN, Artificial Intelligence, Defects, bottle, Manufacturing, Industries