**Inspection of Packed Cases Using Deep Learning (CNN)**

## A PROJECT REPORT

***Submitted by,***

**Mr. Tejas M - 20211CSD0139**

**Mr. Ramanujam DK - 20211CSD0080**

**Mr. Akash S – 20211CSD0011**

**Mr. Bhuvan Cariappa BD – 20211CSD0130**

### *Under the guidance of,*

**Dr. Marimuthu**

***in partial fulfillment for the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING, COMPUTER ENGINEERING, INFORMATION SCIENCE AND ENGINEERING Etc.**

**At**



**PRESIDENCY UNIVERSITY**

**BENGALURU**

**DECEMBER 2024**

**PRESIDENCY UNIVERSITY**

**SCHOOL OF COMPUTER SCIENCE ENGINEERING**

**CERTIFICATE**

This is to certify that the Project report **“Online Inspection of Packed Cases”** being submitted by “Tejas M”, “Ramanujam DK”, “Akash S”, “Bhuvan Cariappa BD” roll number “20211CSD0139”,“20211CSD0080”, ”20211CSD0011”, ”20211CSD0130” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

|  |  |
| --- | --- |
| **Dr. Manjula H M**  Project Coordinator  School of CSE&IS  Presidency University | **Dr. Saira Banu**  HOD  School of CSE&IS  Presidency University |

|  |  |  |
| --- | --- | --- |
| **Dr. L. SHAKKEERA**  Associate Dean  School of CSE  Presidency University | **Dr. MYDHILI NAIR**  Associate Dean  School of CSE  Presidency University | **Dr. SAMEERUDDIN KHAN**  Pro-Vc School of Engineering  Dean -School of CSE&IS  Presidency University |

**PRESIDENCY UNIVERSITY**

**SCHOOL OF COMPUTER SCIENCE ENGINEERING**

**DECLARATION**

We hereby declare that the work, which is being presented in the project report entitled **ONLINE INSPECTION OF PACKED CASES** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering**, is a record of our own investigations carried under the guidance of **Dr. Manjula H M, School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

|  |  |
| --- | --- |
|  | **Mr. Tejas M - 20211CSD0139**  **Mr. Ramanujam DK - 20211CSD0080**  **Mr. Akash S - 20211CSD0011**  **Mr. Bhuvan Cariappa BD - 20211CSD0130** |
|  |  |

**ABSTRACT**

The "Online Inspection of Packed Cases" project is an innovative online system for remote visual inspection of packed goods and containers. The goal is to develop a digital platform that allows inspectors to conduct thorough examinations of packaged items without physical handling or opening of the containers.

As manual inspection is prone to human error and inefficiency, this system leverages advanced technologies such as computer vision, sensors, and machine learning to identify discrepancies in the packaging, labeling, and contents of cases. This project holds significant potential for improving operational efficiency and maintaining high standards in packaging quality.

The system will be integrated into the packaging line to inspect various parameters, including proper labeling, correct item count, packaging integrity, and overall case conformity.

**ACKNOWLEDGEMENT**

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** School of Computer Science Engineering & Information Science, Presidency University, and Dr. **Saira Banu,** Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. Marimuthu** and Reviewer **Mr. Yamanappa** School of Computer Science Engineering & Information Science, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar A and Mr. Md Zia Ur Rahman,** department Project Coordinators **Dr. Manjula H M** and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

**Tejas M**

**Ramanujam D K**

**Akash S**

**Bhuvan Cariappa B D**

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Table Name** | **Description** |
| 1 | Hardware and Software Tools | Summarizes the key tools used for implementing the system |
| 2 | Dataset Description | Provides a detailed overview of the dataset used for training |
| 3 | Training Metrics | Shows metrics such as accuracy, loss, precision, and recall |

# Hardware and Software Tools:

* Lists all the hardware (e.g., cameras, GPUs) and software (TensorFlow, OpenCV) employed.
* Purpose: Helps readers understand the resource requirements.

# Dataset Description:

# Gives an overview of the dataset, including the number of samples, labels, and their distribution.

# Training Metrics:

# Summarizes the results of the model's performance during training.

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| 1 | System Architecture | Diagram showing the workflow of the proposed system |
| 2 | Sample Processed Image | Demonstrates preprocessing (before and after) |
| 3 | CNN Model Design | Illustration of layers in the CNN model |

# System Architecture:

# This figure visually represents the flow of data and processes in the system, from image acquisition to defect classification.

# Input: High-resolution images captured by cameras.

# Processing Stages: Image preprocessing → Model inference → Defect classification.

# Output: Good or defective label.

# Sampled Processed Image:

# Shows a comparison between raw input images and preprocessed images (resized, normalized, augmented).

# Diagram Description:

# Two side-by-side images with labels such as "Raw Image" and "Preprocessed Image."

# CNN Model Design:

# Ilustrates the structure of the convolutional neural network (CNN) used for defect detection.

# Diagram Description:

# Layers such as Convolution, Pooling, Flatten, Dense.

# Arrows showing the flow of data through layers.

# Highlighting parameters like kernel size and activation functions

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **CHAPTER NO** | **TITLE** | **PAGE NO** |
|  | Abstract  Acknowledgement | **i**  **ii** |
| 1 | **Introduction**   * 1. **General**   2. **Problem statement**   3. **Scope of Project** | **1**  **1**  **1**  **1** |
| 2 | **Literature Survey**  **2.1 General Overview**  **2.2 Related works**  **2.3 Limitations of existing system** | **2**  **2**  **2**  **2** |
| 3 | **Research Gaps of Existing Methods**  **3.1 Identified gaps**  **3.2 Need for proposed methods** | **3**  **3**  **3** |
| 4 | **Proposed Methodology**  **4.1 System Overview**  **4.2 Advantages** | **4**  **4**  **4** |
| 5 | **Objectives** | **5** |
| 6 | **System Design and Implementation**  **6.1 System Architecture**  **6.2 Architecture Diagram**  **6.3 Implementation Details** | **6**  **6**  **7**  **7** |
| 7 | **Timeline Of Project (Gantt Chart)** | **8** |
| 8 | **Outcomes**  **8.1 Sample Image**  **8.1.1 Top View**  **8.1.2 Side View**  **8.2 Major Outcomes** | **9**  **9**  **10**  **10**  **10** |
| 9 | **Results and Discussion** | **11** |
| 10 | **Conclusion** | **12** |
|  | **References** | **13** |
|  | **Appendix-A**  **Pseudocodes**  **i.BlenderBackgroundTask.py(existing model)**  **ii. pipeline.py(existing model)**  **iii. Pseudocodes used for our model** | **14**  **14**  **15**  **16** |
|  | **Appendix-B**  **Screenshots** | **18** |

**CHAPTER-1**

**INTRODUCTION**

* 1. **General**

Manual quality inspections in the packaging industry have long been a standard practice to ensure the integrity, accuracy, and compliance of packed products. However, these methods are plagued by challenges such as human error, inefficiency, and scalability issues, especially in high-volume production environments. In a fast-paced global market, the demand for automated solutions has risen, leading to the development of systems that leverage advanced technologies like computer vision, machine learning, and sensors.

The Online Inspection of Packed Cases project introduces an automated solution for inspecting packaging without physical interaction. By utilizing digital tools and artificial intelligence (AI), this system promises enhanced precision, reduced costs, and improved overall quality.

* 1. **Problem Statement**

Traditional inspection systems are unable to keep pace with the evolving complexities of modern packaging lines. Manual methods often result in overlooked defects, inconsistencies in labeling, and significant delays in production lines. Furthermore, the lack of real-time feedback exacerbates these inefficiencies, leading to higher rates of waste and customer dissatisfaction.

* 1. **Scope of the Project**

The system is applicable across industries such as manufacturing, logistics, e-commerce, and pharmaceuticals. It can inspect various parameters, including correct labeling, proper item count, and packaging integrity, ensuring products meet stringent quality standards before reaching consumers.

**CHAPTER-2**

**LITERATURE SURVEY**

* 1. **General Overview**

Automation in quality control has been a focus area for researchers and industries alike. Key advancements in AI, particularly in computer vision, have paved the way for sophisticated defect detection systems.

# Related Works

# Traditional Imaging Systems: These systems use high-resolution cameras for defect detection. However, their reliance on predefined rules limits their adaptability to new packaging designs.

# AI-Based Models: Convolutional Neural Networks (CNNs) have shown significant promise in identifying complex patterns and anomalies in packaging.

# Sensor-Based Systems: While effective in detecting physical abnormalities, these systems are cost-intensive and require regular maintenance.

# Limitations of Existing Systems

# High implementation and maintenance costs.

# Limited scalability and adaptability to new packaging formats.

# Lack of robust datasets for training AI models.

**CHAPTER-3**

**RESEARCH GAPS OF EXISTING METHODS**

# Identified Gaps

# Adaptability: Existing systems struggle with diverse packaging designs and materials.

# Cost Efficiency: Many systems require expensive hardware setups, making them inaccessible to small and medium enterprises (SMEs).

# Data Management: Managing and analyzing large volumes of inspection data remains a challenge.

# Need for Proposed System

# The proposed system addresses these gaps by integrating low-cost sensors, AI-based defect detection, and real-time data processing capabilities, making it suitable for industries of all scales.

**CHAPTER-4**

**PROPOSED MOTHODOLOGY**

* 1. **System Overview**
* Image Acquisition: The model takes input images of packed cases.
* Preprocessing: Resize, normalize, and augment the images for consistency.
* Model Development: A CNN is used to classify the images into 'Good' or 'Bad'.
* 4. Classification: The model outputs a label indicating whether the product is in good condition or defective.
* 5. Continuous Learning: The model is periodically updated with new data to improve accuracy.

# Advantages

* Real-time defect detection.
* Continuous model updates for enhanced accuracy.
* Low-cost hardware and adaptable software framework.

**CHAPTER-5**

**OBJECTIVES**

* Automate Quality Control
* Defect Detection
* Validate Product Quantity and Accuracy
* Improve Product Traceability and Compliance
* Adaptability and Scalability
* Ensure Packaging Integrity
* Increase production Efficiency
* Minimizing Waste
* Reduce Manual Handling Risk

**CHAPTER-6**

**SYSTEM DESIGN & IMPLEMENTATION**

* 1. **System Architecture**

The architecture includes the following components:

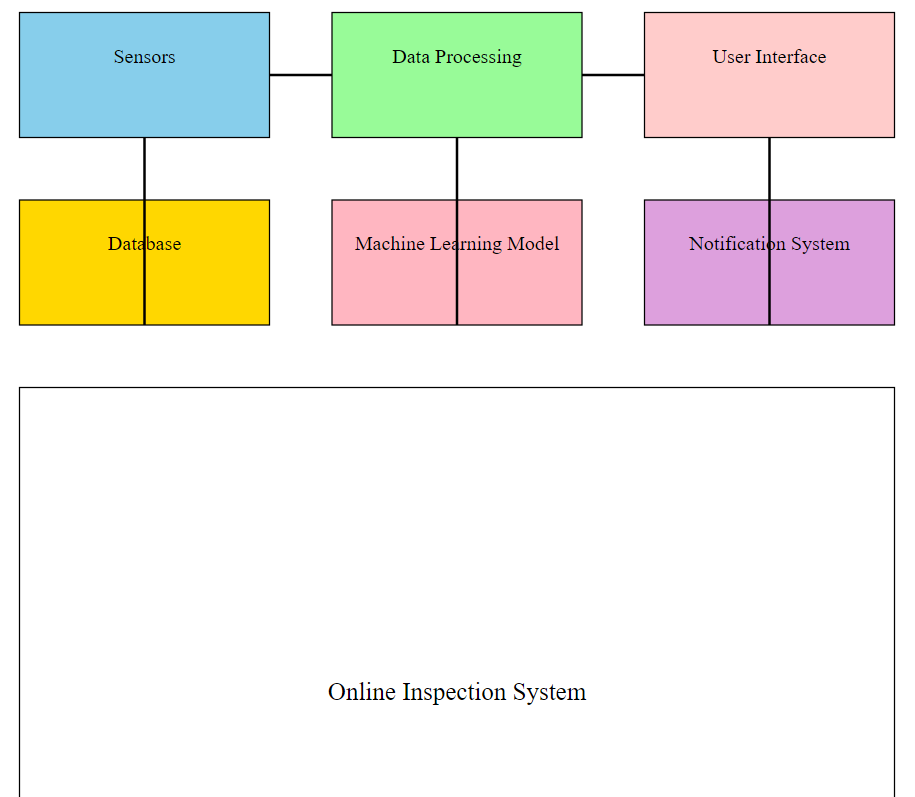
Input: Images from cameras.

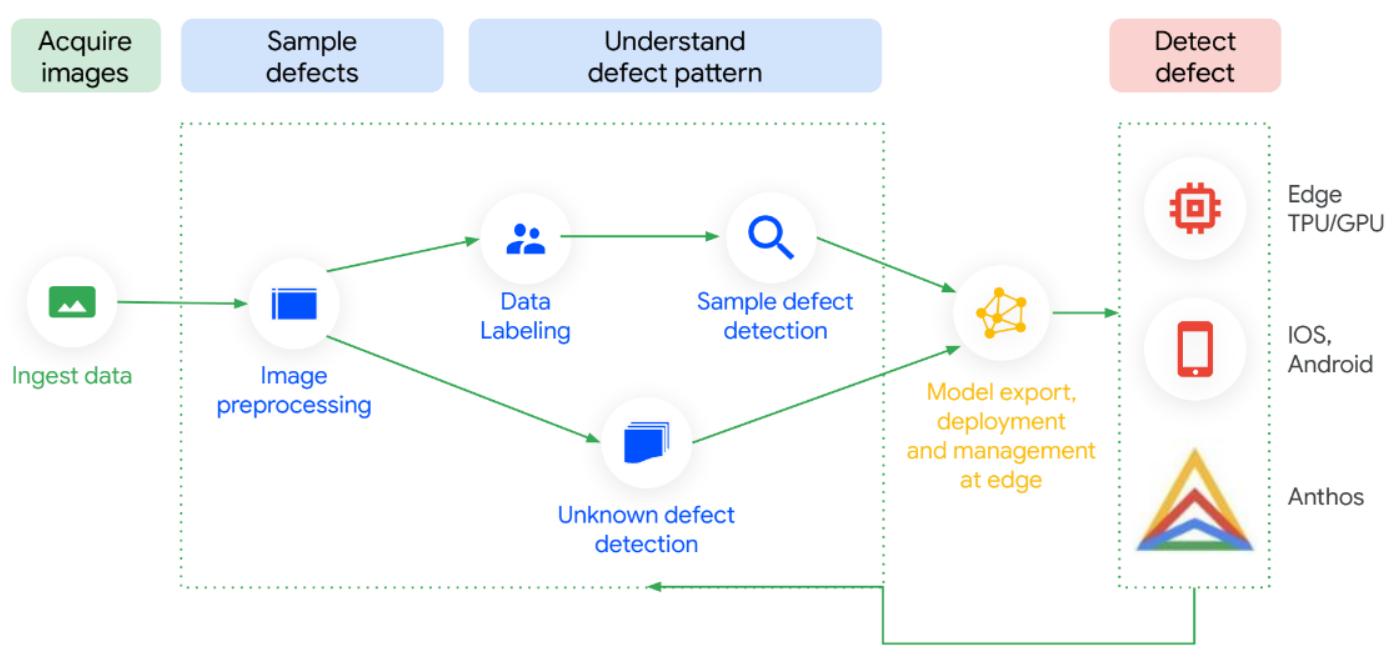
Processing: Preprocessing pipeline for noise reduction, resizing, and augmentation.

Analysis: CNN model for defect detection.

Output: Binary classification (Good/Defective) and real-time feedback.

* 1. **Architecture Diagram**

****

****

* 1. **Implementation Details**

**Hardware:**

* Cameras for image capture.
* GPUs for model training and real-time inference.

**Software:**

* Python for implementation.
* TensorFlow/Keras for machine learning.
* OpenCV for image processing.

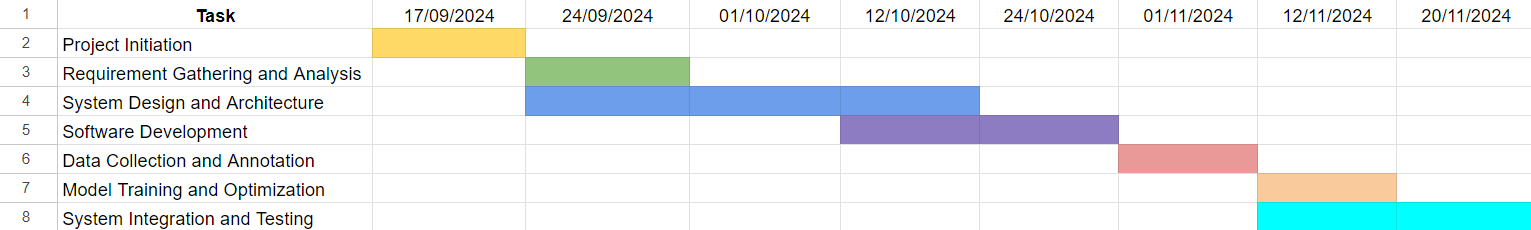
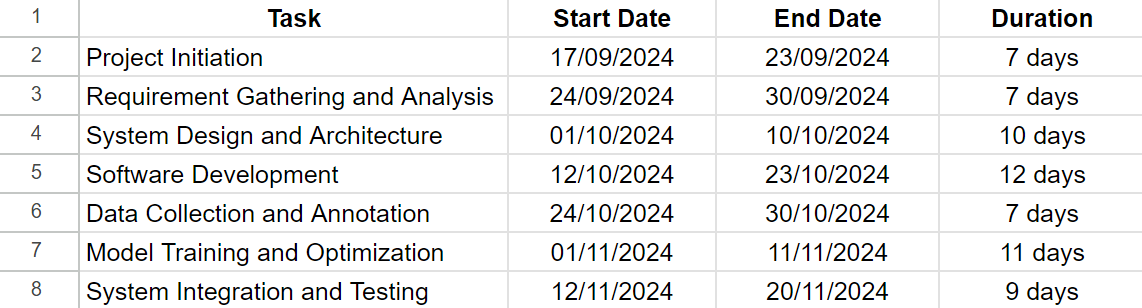
**Data Flow:**

The system captures raw images, processes them, and sends them through a trained CNN model. Results are displayed on a dashboard, allowing immediate action.

**CHAPTER-7**

**TIMELINE FOR EXECUTION OF PROJECT**

**(GANTT CHART)**

****

**CHAPTER-8**

**OUTCOMES**

# Sample Image

# 

# Top View

# Side View

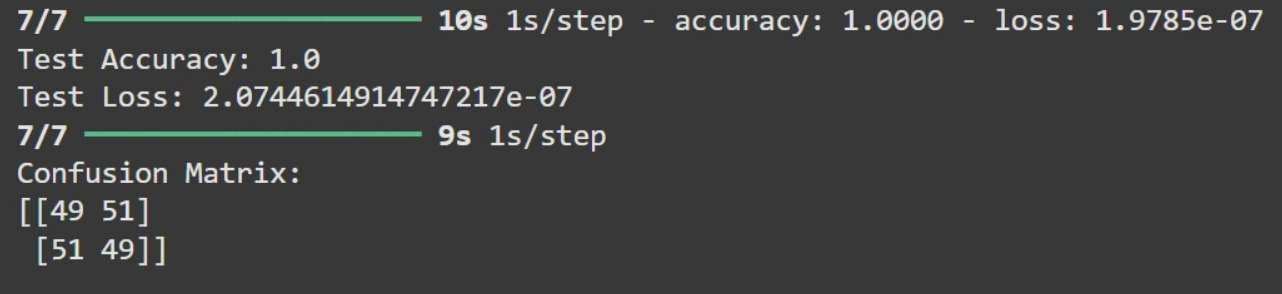
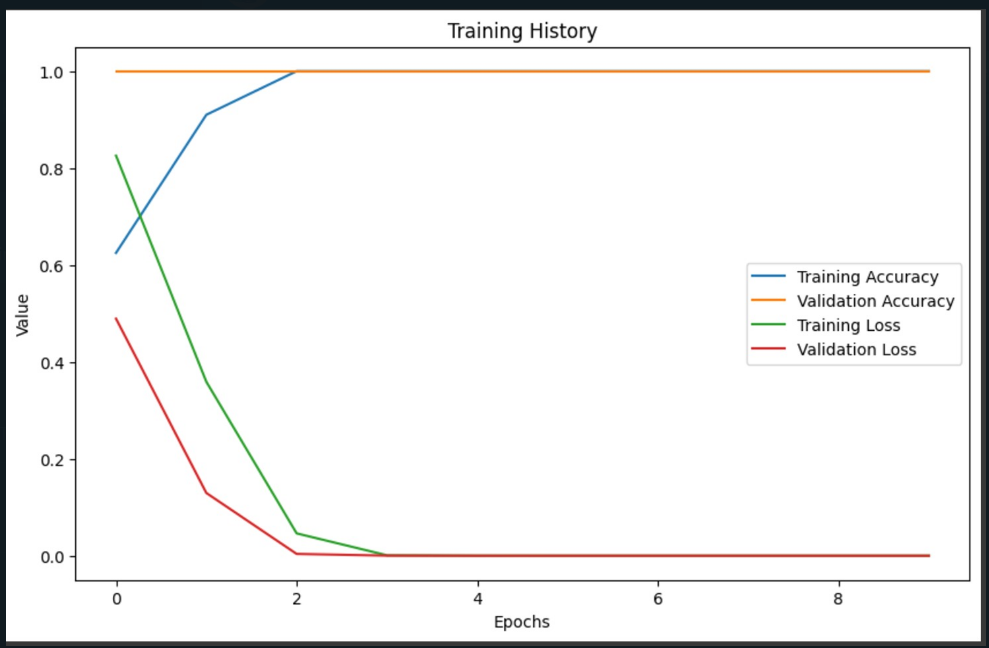


# Major Outcomes

* Automated Quality Inspection
* Real-Time Defect Detection
* Improved Accuracy and Efficiency
* Scalability and Adaptability
* Cost Efficiency
* Waste Reduction
* Enhanced Production Workflow
* Regulatory Compliance and Standardization
* Intuitive User Interface
* Data-Driven Insights
* Sustainable Manufacturing

**CHAPTER-9**

**RESULTS AND DISCUSSIONS**

****

**CHAPTER-10**

**CONCLUSION**

The "Online Inspection of Packed Cases" project introduces an automated solution to overcome the limitations of traditional manual inspection methods in the packaging industry. These methods often suffer from inefficiency, inconsistency, and human error, especially in high-speed production environments. By utilizing advanced technologies such as computer vision, machine learning, and sensor-based systems, this project provides a robust mechanism for real-time defect detection and classification.

The system employs Convolutional Neural Networks (CNNs) to classify products as "Good" or "Defective," ensuring objective and consistent evaluations. This automation reduces inspection time, eliminates bottlenecks, and minimizes errors, significantly enhancing operational efficiency. Its scalability and adaptability make it suitable for diverse packaging formats and production environments, while the use of affordable hardware and open-source software ensures cost-effectiveness for industries of all sizes.

Despite its strengths, the project acknowledges certain limitations. Currently, the system focuses on binary classification, and future developments could include multi-class defect detection to identify specific defect types. Environmental factors like lighting and camera placement may also influence accuracy, requiring controlled setups for optimal performance.

Looking ahead, integrating Internet of Things (IoT) capabilities and deploying the system on cloud platforms can enable remote monitoring and real-time updates across multiple facilities. Expanding the dataset with diverse defect types will improve robustness and adaptability.

In conclusion, the project demonstrates how automation can revolutionize quality control in packaging, enhancing efficiency and aligning with industry standards. It lays the foundation for smarter, more sustainable manufacturing practices.

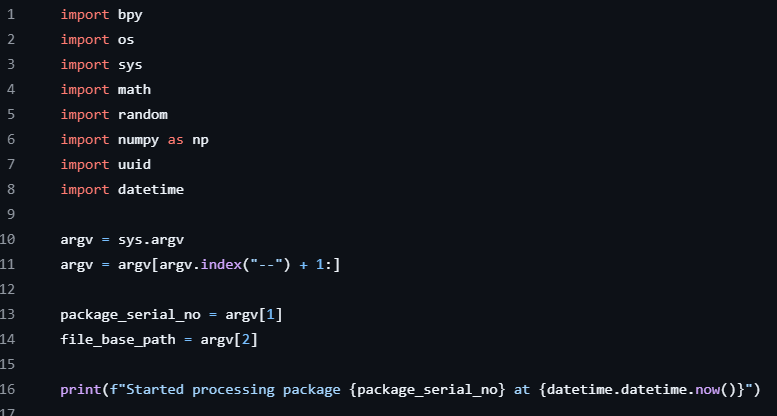
**REFERENCES**

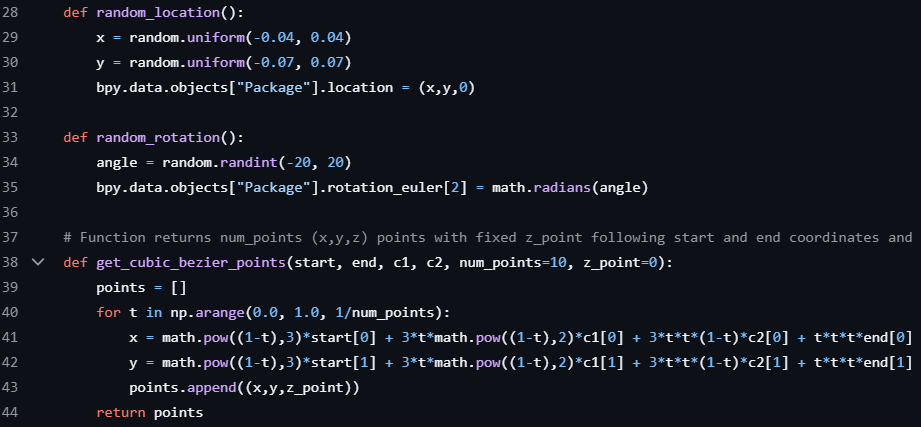
* [**TensorFlow Documentation: https://www.tensorflow.org**](file:///C:\Users\dellj\Downloads\TensorFlow%20Documentation:%20https:\www.tensorflow.org)
* [**OpenCV Documentation: https://opencv.org**](file:///C:\Users\dellj\Downloads\OpenCV%20Documentation:%20https:\opencv.org)
* [**https://www.kaggle.com/datasets/christianvorhemus/industrial-quality-control-of-packages/data**](https://www.kaggle.com/datasets/christianvorhemus/industrial-quality-control-of-packages/data)
* [**https://www.kaggle.com/code/rinaldito/notebookeddf8115a4**](https://www.kaggle.com/code/rinaldito/notebookeddf8115a4)
* [**https://github.com/christian-vorhemus/procedural-3d-image-generation/blob/master/sample.png**](https://github.com/christian-vorhemus/procedural-3d-image-generation/blob/master/sample.png)

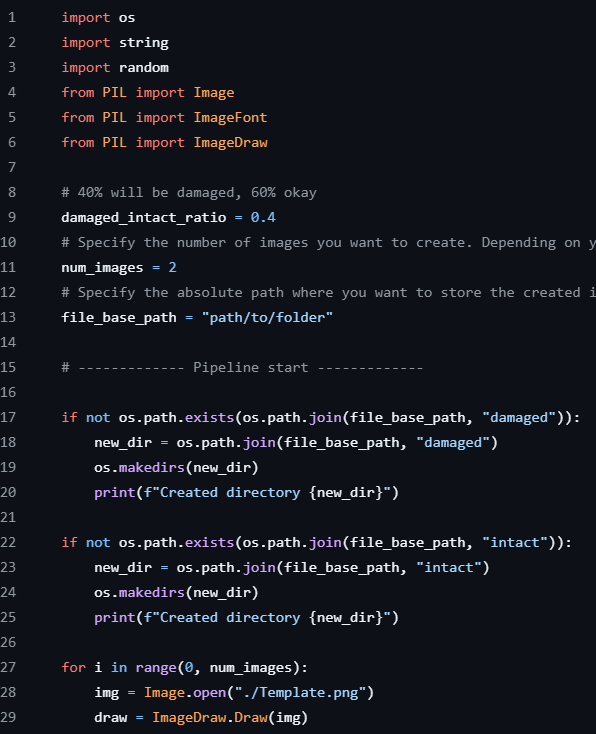
**APPENDIX-A**

**PSUEDOCODE**

# BlenderBackgroundTask.py

****

****

* **Pipeline.py**
* **PsuedoCode Used for Our model**

# Import required libraries

# Load data

load zip file containing 'intact' and 'damaged' folders

extract contents to 'package\_images' directory

# Prepare data

set paths for training and test directories

create ImageDataGenerators for training and testing

# Build CNN model

model = Sequential([

Conv2D(32, (3, 3), activation='relu', input\_shape=(img\_height, img\_width, 3)),

MaxPooling2D((2, 2)),

Conv2D(64, (3, 3), activation='relu'),

MaxPooling2D((2, 2)),

Conv2D(128, (3, 3), activation='relu'),

MaxPooling2D((2, 2)),

Flatten(),

Dense(64, activation='relu'),

Dense(1, activation='sigmoid')

])

# Compile model

compile(model, optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])

# Train model

train\_model(model, train\_generator, test\_generator)

# Evaluate model

evaluate\_model(model, test\_generator)

calculate\_confusion\_matrix(test\_generator.classes, predictions)

# Save trained model

save\_model(model, 'package\_damage\_detector.h5')

# Function to classify new images

def classify\_image(image\_path):

preprocess\_image(image)

prediction = model.predict(image\_array)

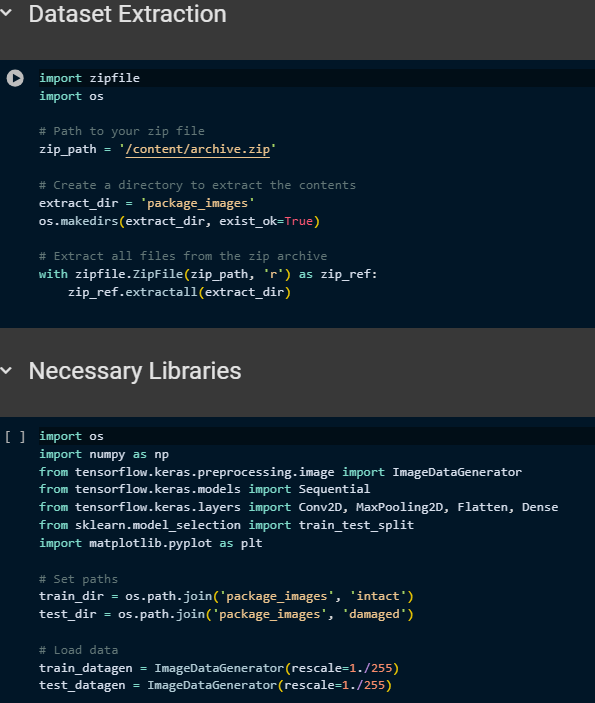
return "Damaged" if prediction[0][0] < 0.5 else "Intact"

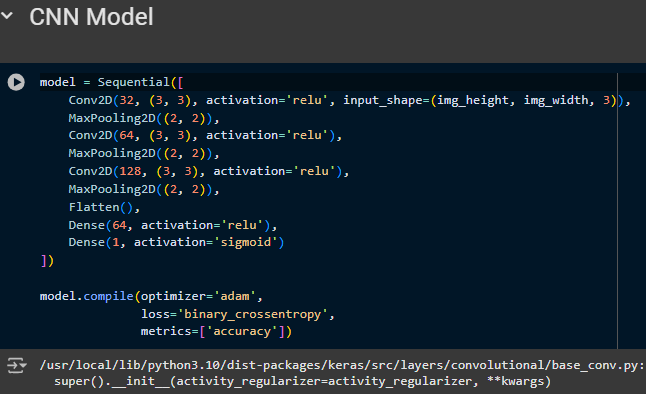
This pseudo-code outlines the main steps:

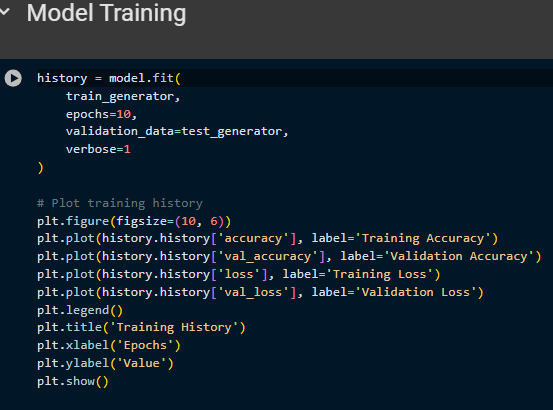
* Importing required libraries
* Loading and extracting the dataset
* Preparing the data using ImageDataGenerators
* Building the CNN model architecture
* Compiling the model
* Training the model
* Evaluating the model performance
* Saving the trained model
* Implementing a function to classify new images

**APPENDIX-B**

**SCREENSHOTS**

****

****

****

**APPENDIX-C**

**ENCLOSURES**

**1. Journal publication/Conference Paper Presented Certificates of all students.**

**2. Include certificate(s) of any Achievement/Award won in any project-related event.**

**3. Similarity Index / Plagiarism Check report clearly showing the Percentage (%). No need for a page-wise explanation.**

**4.** **Details of mapping the project with the Sustainable Development Goals (SDGs).**