COLLEGE CODE 1128

COLLEGE NAME TJS ENGINEERING COLLEGE

DEPARTMENT ELECTRONICS AND COMMUNICATION ENGINEERING

STUDENT NM-ID aut112823ECE001,aut112823ECE004,aut112823ECE005
aut112823ECE006,aut112823ECE011

ROLL NO 112823106001,112823106004,112823106005,112823106006,
112823106011

DATE 12.05.2025

TECHNOLOGY-PROJECT NAME

SUBMITTED BY,

Your Name and team member names.

DEEPAK.K
DENZIL ROSHAN RAJ.G
DHAKSHANA MOORTHY .A
ADARAM GANGA PRASAD
HARIHARAN .K

Phase 5: Project Demonstration & Documentation

Title

Quality Control in Manufacturing Using AI and Computer Vision

Abstract

The project aimed to automate and optimize quality control in manufacturing using AI-based computer vision systems. This final phase showcases a complete and deployable system with a CNN model for defect detection, real-time inference capabilities, an interactive GUI, and integrated cloud analytics. The document covers system demonstration, full documentation, testing metrics, final code, and stakeholder feedback. The AI model identifies defects such as scratches, dents, and misalignments in under 300ms with over 95% accuracy. Screenshots, performance benchmarks, and future recommendations are included.

1. Project Demonstration

Overview:

The final demonstration validates the real-time performance of the quality control system under simulated factory conditions.

Demonstration Details:

- Live Detection: Real-time video processing showing AI detecting product defects accurately.- GUI Walkthrough: Navigating the tabbed interface with defect history, metrics charts, and CSV export.- Cloud Dashboard: Display of defect logs on Firebase with filter and summary options.

- Security Check: Demonstration of role-based login and encrypted data logs.
- Performance Display: Monitoring latency and model performance live under varied lighting.

Outcome:

The system demonstrated consistent detection performance, fast GUI responsiveness, and real-time analytics integration.

2. Project Documentation

Overview:

Complete technical documentation was created to ensure system understanding and continuity.

Documentation Sections:

- Architecture Diagrams: Detailing CNN pipeline, GUI flow, and cloud integration.
- Codebase Overview: With explanations for:
  - CNN Model Training
  - Real-Time Detection Logic
  - GUI Application
  - Cloud Sync Modules
- User Manual: Instructions for quality supervisors to interact with the GUI and dashboard.
- Admin Manual: Maintenance procedures, cloud sync settings, and security protocols.- Testing Reports: Accuracy graphs, latency metrics, load testing summaries.

Outcome:

Comprehensive documentation supports seamless deployment and future upgrades.

3. Feedback and Final Adjustments

Overview:

Feedback from stakeholders and faculty was collected and used for fine-tuning.

Steps Taken:

- Feedback Collection: Via demo sessions and direct observation of user interaction.
- Final Fixes:
  - GUI thread lag resolved with better multiprocessing.
  - Enhanced defect categorization via fine-tuned model.

Outcome:

System is now optimized for real-world deployment in production environment

4. Final Project Report Submission

Overview:

Summarizes all phases from model development to cloud deployment.

Included Sections:

- Executive Summary

- Phase-by-Phase Analysis: Data collection, model training, GUI and cloud integration.

- Key Challenges & Solutions

- System Capabilities: Real-time AI inference, 95.2% defect classification accuracy, secure logging.

Outcome:

Comprehensive final report prepared, showcasing readiness for deployment and scalability.

5. Project Handover and Future Works

Next Steps & Recommendations:

- Predictive Maintenance: Expand to include anomaly prediction.

- Edge Deployment: Optimize for embedded systems and Raspberry Pi-based inference.- Multilingual Interface: GUI expansion for multilingual operator support.

- Continuous Learning: Automate retraining with new defect data.

Outcome:

System handed over for production use with clear future roadmap.

6. Source Code Snapshots

TensorflowPlugin.py - C:\Users\Administrator\AppData\Local\Programs\Python\Python36\Lib\site-packages\nuitka\plugins\standa...

File Edit Format Run Options Window Help

```
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(150, 150, 3)),
    MaxPooling2D(2, 2),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D(2, 2),
    Dropout(0.3),
    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(3, activation='softmax')
])
```