Master Thesis Proposal

Study Programme: [MEng] Engineering and Sustainable Technology Management- Focus on Industry 4.0: Automation, Robotics & 3D Manufacturing

Name Surname :- Likith Jayaram Gowda

Matriculation Number: 3121844

Intake: Summer semester 2024

Master Thesis Topic:

Computer Vision and Deep Learning for Real-Time Quality Inspection in Manufacturing

Structure:

Name of First Supervisor:

Name of Professorial Supervisor:

(In case the first Supervisor is a scientific worker)

Signature:

Date, Candidate´s Signature

Date, SRH Supervisor´s Signature

Date, SRH/External Supervisor´s Signature

(In case first Supervisor is a scientific worker)

TABLE OF CONTENTS

INTRODUCTION

1. Background and Context

Modern manufacturing increasingly depends on high-precision, real-time quality inspection to maintain competitiveness and reduce waste. Traditional manual inspection suffers from subjectivity, fatigue, and low throughput (Yadav et al., 2024). The emergence of Industry 4.0 has accelerated the integration of computer vision (CV) and deep learning (DL) into production systems for defect detection, enabling continuous monitoring and data-driven decision-making (Zhang et al., 2025). Recent advances in convolutional neural networks (CNNs), such as ResNet, YOLO, and SSD, have made it possible to automatically localize and classify defects with high accuracy (Li et al., 2025; Kumar & Patel, 2024).

At the same time, the proliferation of edge devices like Raspberry Pi 4 and NVIDIA Jetson Nano provides affordable computational resources that can execute optimized DL models directly on the shop floor (Deepa et al., 2024). These developments open the path for lightweight, low-latency, and energy-efficient inspection systems, which are essential for small and medium-sized enterprises (SMEs) striving to implement intelligent manufacturing solutions (Kim et al., 2024).

2. Problem Statement

Despite impressive progress in deep learning–based inspection, several practical challenges persist.

- High-performing CNN architectures usually require large labeled datasets and expensive GPUs, which limits adoption in SMEs (Kumar & Patel, 2024).
- Real-time inspection demands low inference latency, yet most existing models are computationally heavy and unsuitable for edge deployment (Yadav et al., 2024).
- Many industrial datasets lack domain generalization, making models sensitive to lighting, vibration, and texture variations (Zhang et al., 2025).
- Although edge computing can reduce latency, optimized, lightweight frameworks that maintain accuracy while minimizing energy consumption remain scarce (Deepa et al., 2024).

Therefore, the real-world problem faced by professionals is the absence of a unified, low-cost, and resource-efficient CV framework that can deliver real-time, accurate defect detection under industrial constraints.

3. Relevance and Importance of the Research

This research directly addresses the current demand for energy-aware, low-cost automation in manufacturing lines (Deepa et al., 2024). It contributes a practical blueprint for deploying DL-based inspection systems on affordable edge hardware—critical for SMEs pursuing digital transformation without heavy infrastructure investment.

From a scientific standpoint, it enriches ongoing work in transfer learning optimization (Kumar & Patel, 2024) and temporal feature stabilization (Yadav et al., 2024) by demonstrating how these concepts can be operationalized in real-time, on-device scenarios. The study's findings will be relevant to manufacturing engineers, automation researchers, and sustainability advocates seeking to balance accuracy, cost, and energy efficiency. Given global trends toward net-zero production and smart factories, the work's timeliness is unquestionable (Zhang et al., 2025).

4. Scientific Problem for the Research

The scientific problem centers on designing and validating a transfer-learning-based deep neural network that maintains high detection accuracy while operating within the computational and power limits of edge devices.

The theoretical contribution lies in combining concepts from deep learning generalization, model compression, and edge computing optimization to create a lightweight architecture capable of learning effectively from limited data (Kumar & Patel, 2024; Deepa et al., 2024). By unifying transfer learning, quantization, and real-time inference, the research seeks to provide an empirically tested model that demonstrates how accuracy, latency, and energy metrics can be jointly optimized.

5. General Objective of the Research

To develop and evaluate a lightweight computer vision framework based on transfer learning and edge deployment for real-time defect detection in manufacturing environments.

6. Specific Objectives of the Research

- To design and implement a CNN-based defect detection pipeline using transfer learning on pre-trained models such as ResNet50 or EfficientNet-B0.
- To optimize the trained models through quantization and pruning for efficient edge deployment (TensorFlow Lite / ONNX Runtime).
- To compare the model's performance across accuracy, latency, and energy consumption metrics between desktop and edge hardware.
- To simulate real-time inspection using benchmark datasets (e.g., NEU Surface Defect, DAGM 2007) and live video streams.
- To evaluate the framework's cost-effectiveness and scalability for SMEs aiming for digital transformation.

7. Research Questions

- How effectively can transfer learning adapt pre-trained CNNs to detect manufacturing defects with limited domain-specific data?

- What level of inference speed and energy efficiency can be achieved on low-power edge devices without compromising accuracy?
- Which optimization techniques (quantization, pruning, input-resolution tuning) provide the best trade-off between performance and latency?
- Can the proposed framework serve as a reproducible and scalable solution for SMEs adopting Industry 4.0 technologies?

LITERATURE REVIEW

1. Key Concepts, Theories, and Studies

The evolution of computer vision (CV) and deep learning (DL) has profoundly transformed industrial quality inspection. The foundational concept is that visual data captured from cameras or sensors can be analyzed algorithmically to detect anomalies or surface defects automatically (Yadav et al., 2024). Early machine vision systems relied on rule-based image processing—such as edge detection, thresholding, and texture analysis—using methods like Sobel, Canny, or GLCM (Zhang et al., 2025). These systems, though simple, were sensitive to variations in lighting and orientation, limiting their robustness in real-world production environments.

The shift toward deep learning–driven computer vision introduced convolutional neural networks (CNNs), which automatically learn hierarchical features from raw images without manual feature engineering (Kumar & Patel, 2024). CNN-based architectures such as ResNet, Inception, and EfficientNet demonstrated state-of-the-art performance in surface-defect classification and localization tasks. Transfer learning emerged as a critical theoretical approach, enabling pre-trained models on large datasets (e.g., ImageNet) to be fine-tuned for industrial applications with limited labeled data (Kumar & Patel, 2024).

Meanwhile, object detection frameworks such as YOLO, SSD, and Faster R-CNN became essential for localizing multiple defect regions within a single image. Li, Wang, and Zhou (2025) advanced this further by integrating CV with deep learning–based decision-making for automated repair guidance, bridging the gap between detection and corrective action.

Another key theoretical underpinning is edge computing—processing data locally on embedded devices instead of cloud servers. Studies such as Deepa et al. (2024) demonstrate how lightweight CNNs and TensorFlow Lite deployments on Raspberry Pi or Jetson Nano achieve real-time detection while lowering latency and energy consumption. This directly supports sustainable Industry 4.0 practices by reducing both computational cost and carbon footprint.

Complementing these purely neural approaches, hybrid systems combining symbolic reasoning with neural networks (Kim et al., 2024) have been proposed to enhance interpretability and reliability, especially in safety-critical manufacturing environments. Collectively, these studies highlight a growing ecosystem of AI-driven quality inspection systems that balance accuracy, speed, and efficiency.

2. Key Debates and Controversies

Despite these advances, there is ongoing debate around model complexity versus deployment feasibility. Deep, high-capacity models such as ResNet101 or DenseNet provide strong accuracy but are computationally expensive, making them unsuitable for real-time, on-edge applications (Yadav et al., 2024; Deepa et al., 2024). Conversely, lightweight architectures like MobileNet or SqueezeNet offer faster inference but may sacrifice precision in detecting fine-grained defects (Zhang et al., 2025). Balancing these trade-offs remains an active research challenge.

Another controversy concerns data dependency and generalization. CNNs excel when trained on large, diverse datasets; however, industrial defect datasets are typically small and highly domain-specific, making transfer learning both an opportunity and a limitation (Kumar & Patel, 2024). Questions persist about how well pre-trained models from natural images transfer to manufacturing textures, metallic surfaces, and reflective materials (Li et al., 2025).

There is also discussion about sustainability versus performance. While edge deployment reduces network latency and energy consumption, continuous on-device processing can still introduce heat and energy overheads. Researchers argue for energy-aware architectures that adapt inference dynamically to production conditions (Deepa et al., 2024). Lastly, a methodological debate persists over whether hybrid neuro-symbolic systems (Kim et al., 2024) or pure deep learning architectures are more effective for industrial inspection tasks, especially when interpretability and compliance are critical.

3. Gaps in Existing Knowledge

- While transfer learning has shown promise, there is limited empirical evidence comparing how specific pre-trained backbones (e.g., ResNet50, EfficientNet-B0, MobileNetV3) perform under edge-device constraints (Kumar & Patel, 2024; Deepa et al., 2024). Most studies report results on high-end GPUs rather than real-time embedded systems.
- Existing literature lacks standardized evaluation metrics for cross-comparing accuracy, latency, and energy efficiency, which are all critical for real-time industrial inspection (Yadav et al., 2024). Moreover, few frameworks attempt to jointly optimize these three metrics—most focus on improving accuracy alone.
- While several studies demonstrate defect detection, very few integrate temporal reasoning or stabilization mechanisms to handle vibration, lighting fluctuations, or object motion common in manufacturing lines (Yadav et al., 2024).
- Despite promising demonstrations of edge deployment (Deepa et al., 2024), there remains a gap in open, reproducible frameworks that SMEs can easily adopt without requiring expensive hardware or complex setup.

- There is an absence of research quantifying the energy and carbon efficiency gains of real-time vision inspection systems under realistic factory conditions—an increasingly relevant aspect given global sustainability mandates.

RESEARCH DESIGN AND METHODS

Research Design

This research adopts a quantitative and experimental design, aimed at developing and validating a lightweight computer vision framework for real-time defect detection in manufacturing environments. The study involves original data experimentation using open-source defect image datasets and performance benchmarking across hardware platforms.

The approach is deductive, grounded in prior theories of transfer learning, model optimization, and edge computing. The research begins with established frameworks and hypotheses derived from previous studies (Kumar & Patel, 2024; Deepa et al., 2024), testing how pre-trained CNN models perform when optimized for resource-constrained environments.

The design follows a descriptive–correlational–experimental pattern:

- Descriptive, in defining key variables and performance metrics such as accuracy, latency, and energy efficiency.
- Correlational, in analyzing the relationship between model complexity, inference speed, and power consumption.
- Experimental, in training, optimizing, and deploying CNNs under different configurations on both desktop and edge devices.

- **Methodological Approach**:

  The study uses a computational modeling approach supported by quantitative analysis of real-time performance metrics. The proposed methodology combines machine learning experiments with hardware-based performance measurements to ensure replicability and objectivity (Yadav et al., 2024).

- **Access and Data Sources**:

  All datasets are publicly accessible from standard industrial CV benchmarks such as:

  - NEU Surface Defect Database (six defect types)
  - DAGM 2007 Dataset (fabric and texture defects)
  - Kolektor Surface Defect Dataset (KSDD2) (small metallic surface imperfections)

  These datasets represent various material textures and surface patterns common in manufacturing. Since they are open-access, no additional permissions are required.

- **Case Selection:**

The study includes three CNN architectures (ResNet50, EfficientNet-B0, and MobileNetV3) as experimental cases. Each model will be tested on two datasets (NEU and DAGM), creating six experimental combinations.

Selection is purposive, based on their known suitability for transfer learning and edge deployment (Kumar & Patel, 2024; Deepa et al., 2024).

- **Timeframe**:

  The estimated timeframe is 24 weeks (six months):

  - Weeks 1–4: Literature review and dataset preparation.
  - Weeks 5–10: Model training and fine-tuning.
  - Weeks 11–14: Model optimization (quantization, pruning).
  - Weeks 15–20: Edge deployment and performance testing.
  - Weeks 21–24: Analysis, validation, and report compilation.

Indicators and Data Structure

- **Indicators / Parameters**:

  The key indicators to be measured include:

  - Accuracy (F1-score, Precision, Recall) – model performance.
  - Latency (ms/frame) – inference speed for real-time inspection.
  - Frames per second (FPS) – operational throughput.
  - Energy consumption (Wh/frame) – energy efficiency metric (Deepa et al., 2024).
  - Model size (MB) – memory footprint relevant to deployment feasibility.

- **Data Recording**:

  All training logs, performance outputs, and hardware readings will be recorded using Python scripts and TensorBoard logs. Energy consumption data will be collected through power-monitoring utilities (e.g., Jetson Stats or USB power meters).

- **Data Coding and Categorization:**

  Since the study uses quantitative data, coding follows deductive categorization based on predefined metrics rather than emergent categories. Data will be coded numerically for computational analysis (accuracy, latency, energy, etc.).

- **Data Analysis Approach:**

The analysis will be primarily quantitative, using statistical tools such as mean, standard deviation, and correlation analysis. Comparative performance graphs will be generated using Matplotlib and Pandas.

Additionally, a multi-criteria evaluation will be applied to rank model–deployment combinations using normalized weight scores (accuracy, latency, and energy trade-offs).

Methods and Sources

Tools and Software

- Programming Environment: Python 3.10, TensorFlow 2.x, PyTorch 2.x.
- Hardware:
  - Development: Laptop workstation with NVIDIA GPU (training phase).
  - Deployment: Raspberry Pi 5 and NVIDIA Jetson Nano (inference phase).
- Optimization Frameworks: TensorFlow Lite, ONNX Runtime, and NVIDIA TensorRT.
- Visualization and Analysis: TensorBoard, Matplotlib, and Streamlit for GUI-based monitoring.

Procedure

1. Dataset preprocessing: normalization, augmentation (rotation, brightness, flipping).
2. Model training: fine-tuning pre-trained CNNs using transfer learning.
3. Model optimization: post-training quantization and pruning for edge deployment.
4. Deployment testing: running inference on edge hardware under real-time simulated conditions using recorded conveyor footage (Li et al., 2025).
5. Performance evaluation: benchmarking accuracy, latency, and energy efficiency.

Data collection Timeline:

- Training Data Collection: 2–3 weeks (open datasets).
- Model Training and Validation: 4–5 weeks.
- Edge Testing and Analysis: 4 weeks.

Analysis Methods:

- Quantitative comparison of models via accuracy–latency–energy trade-off.

- Regression and correlation analysis between model parameters and performance indicators.
- Visualization of trade-offs and performance through charts and dashboards.

Practical Considerations

Limitations

- Hardware constraints: limited RAM or compute power on edge devices may require aggressive quantization, risking minor accuracy loss.
- Dataset bias: public datasets may not fully represent industrial variability, potentially affecting generalization (Zhang et al., 2025).
- Lighting and environmental factors: simulated lighting variations will be used to mimic real-world conditions (Yadav et al., 2024).

Mitigation Strategies

- Employ data augmentation to improve generalization.
- Use hybrid precision quantization (INT8/FP16) to balance performance and speed.
- Test under controlled artificial lighting and motion blur to simulate factory environments.

Ethical Considerations

No human subjects or sensitive data are involved. Ethical compliance includes using only open-source datasets and transparent reporting of all experiments. Hardware testing will follow safety guidelines for device thermal management during continuous inference (Deepa et al., 2024).

IMPLICATIONS AND CONTRIBUTIONS TO KNOWLEDGE

Practical Implications

- The proposed research directly supports the digital transformation of manufacturing by offering a lightweight, cost-effective, and scalable computer vision framework deployable on edge devices such as Raspberry Pi or Jetson Nano. The practical impact lies in enabling small and medium-sized enterprises (SMEs)—which often lack access to high-end GPUs or cloud-based systems—to adopt real-time defect detection as part of their production lines (Deepa et al., 2024).
- By demonstrating that transfer learning and model optimization techniques (quantization and pruning) can preserve accuracy while drastically reducing inference

latency and energy use, the research will improve the efficiency, sustainability, and responsiveness of automated quality control systems (Kumar & Patel, 2024). This can lead to reduced material waste, fewer production halts, and enhanced product traceability—key components of sustainable manufacturing (Yadav et al., 2024).

- Furthermore, this work aligns with the Industry 4.0 vision by integrating machine learning, edge computing, and intelligent quality inspection into a single, deployable solution (Zhang et al., 2025). The outcomes could inform policy and training initiatives within industrial learning factories, where low-cost CV systems can serve as prototypes for workforce upskilling in AI-based automation.
- In summary, the framework contributes a practical blueprint for developing energy-aware, real-time defect detection systems that can be implemented in educational, research, and industrial contexts with minimal resources. It demonstrates how sustainable automation can be achieved without reliance on cloud infrastructure or expensive hardware.

Theoretical Implications

- From a theoretical standpoint, the research strengthens the growing body of work on transfer learning and edge-optimized deep learning architectures in manufacturing. It empirically tests the generalizability of pre-trained CNN backbones (ResNet50, EfficientNet-B0, MobileNetV3) when applied to industrial defect detection tasks—a topic often discussed but rarely validated under real-time, low-power conditions (Kumar & Patel, 2024; Deepa et al., 2024).
- This study also contributes to the multi-objective optimization theory within computer vision by quantifying trade-offs between accuracy, latency, and energy consumption—parameters that are typically evaluated in isolation (Yadav et al., 2024). The integration of these performance metrics into a single analytical framework can inform future research on resource-constrained AI systems, fostering a new paradigm for sustainable intelligent manufacturing.
- Moreover, by embedding temporal stability and energy-awareness into the model evaluation process, this research helps bridge the gap between traditional accuracy-centered computer vision models and emerging eco-efficient AI paradigms (Li et al., 2025). The results will provide a theoretical foundation for subsequent studies exploring adaptive, context-aware defect detection systems that can dynamically balance speed and precision in response to production demands.
- In essence, this project aims to advance the theoretical understanding of how deep learning models can be optimized for edge environments, strengthening the intersection of machine vision, sustainable computing, and Industry 4.0 research frameworks (Zhang et al., 2025).

STRENGTHS AND WEAKNESSES OF THE RESEARCH PROPOSAL

Strengths

1. The research is highly relevant to current industrial needs, addressing pressing issues of cost, latency, and sustainability in manufacturing quality inspection.
2. It offers a measurable and reproducible methodology that combines empirical experimentation with theoretical grounding in transfer learning and model optimization.
3. It bridges the gap between research and implementation, contributing to both applied engineering and AI research communities.
4. The inclusion of energy and carbon efficiency metrics adds a sustainability dimension often missing from prior defect detection studies.

Weaknesses

1. The research relies primarily on open-source datasets, which may not fully capture the variability and complexity of industrial conditions (Zhang et al., 2025).
2. Real-world validation within a production environment may be constrained by accessibility and safety regulations.
3. Edge devices have limited processing capabilities, which may restrict the deployment of advanced architectures without further compression.
4. External factors such as lighting variation or vibration could impact inference stability, despite controlled testing (Yadav et al., 2024).

Nevertheless, these limitations also represent opportunities for future research—including adaptive domain transfer, sensor fusion, and collaborative cloud–edge inspection systems. The project's overall strength lies in its balance between innovation, practicality, and sustainability, positioning it as a valuable contribution to the evolution of intelligent, energy-efficient manufacturing ecosystems.

REFERENCES

- Deepa, D., Bhattacharjee, K., Yoganand, S., & Balaji, N. (2024). *Cost-effective machine vision system for real-time defect detection and energy efficiency in learning factories.* IEEE International Conference on Cyber-Physical Production Systems. IEEE.
- Kumar, R., & Patel, A. (2024). *Automated quality inspection in manufacturing with computer vision systems using transfer learning on ResNet50.* IEEE International Conference on Automation and Manufacturing Systems. IEEE.
- Li, Z., Wang, X., & Zhou, Q. (2025). *Integrating computer vision and deep learning for automated object diagnosis and repair.* In Proceedings of the IEEE International Conference for Emerging Technology (INCET). https://doi.org/10.1109/INCET64471.2025.11140111
- Yadav, B., Rao, D. D., Mandiga, Y., Gill, N. S., & Gulia, P. (2024). *The role of computer vision in automating defect detection in manufacturing.* In Proceedings of the IEEE ICTBIG Conference (pp. 1–8). https://doi.org/10.1109/ICTBIG64922.2024.10911417
- Zhang, X., Tsang, W. M., Yamazaki, K., & Singh, A. S. (2025). *Quality inspection and defect recognition technology for intelligent manufacturing based on machine vision.* In Proceedings of the IEEE International Conference on Digital Analysis and Processing (DAPIC). https://doi.org/10.1109/DAPIC66097.2025.00068
- Kim, H., Su, J., & Huang, D. (2024). *Computer vision for automated quality inspection in manufacturing.* In Proceedings of the IEEE International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI). https://doi.org/10.1109/ACCAI61061.2024.10601894