A PROJECT REPORT

ON

AI-BASED STRUCTURAL HEALTH MONITORING SYSTEM
(Sub-systems: Crack Detection, Missing Part Detection, and Muzzle-Barrel Alignment)

SUBMITTED TO:

PME GROUP(ATAGS),
ARMAMENT RESEARCH AND DEVELOPMENT ESTABLISHMENT
(ARDE), DEFENCE RESEARCH & DEVELOPMENT ORGANISATION
(DRDO) PASHAN, PUNE - 411021

SUBMITTED BY:

YUGANDHAR AUTADE

UNDER THE GUIDANCE OF:

Dr. Ms. HARNEET THAKUR (Sc.G) **Mr. ROUNAK PADGIL (Sc.F)**

ACKNOWLEDGMENT

This internship report is the outcome of the guidance, support, and professional cooperation received from various individuals during the internship period.

I express my sincere gratitude to the Armament Research and Development Establishment (ARDE), Defence Research and Development Organization (DRDO), Pune, for providing the opportunity to carry out this internship and gain practical exposure to a defence research and development environment.

I convey my special thanks to my guides, Dr. Ms. Harneet V Thakur And Mr. Rounak Padgil, for their continuous supervision, technical guidance, and constructive feedback throughout the internship. Their systematic approach and professional mentorship were instrumental in enhancing my technical competence and understanding of defence-oriented research practices.

I am thankful to all staff members of the PME Group (ATAGS) at ARDE–DRDO for their cooperation, technical support, and knowledge sharing during the course of the internship. I would like to acknowledge Ms. Shrutika, Ms. Sanskruti, Mr. Prajvansh, Mr. Amit Kumar Sir, Mr. Manas, and Mr. Kamesh Sir for their assistance and collaborative support.

Name Sign

Yugandhar Autade

ABSTRACT

This project presents the design and implementation of an **Integrated AI-based Structural Health Monitoring (SHM) System** specifically developed for the automated inspection of armament weapon platforms. The primary objective is to eliminate human subjectivity and enhance the reliability of structural through advanced **Deep Learning (CNN)** architectures.

The system is comprised of three critical diagnostic modules:

1. **Automated Crack Detection:** A CNN-based analysis of surface textures to identify and localize micro-fractures and fatigue-related stress cracks that compromise structural integrity.
2. **Missing Part Detection:** A deep learning verification suite that scans complex assemblies to ensure the presence of all vital components, providing real-time alerts for missing fasteners or safety pins.
3. **Muzzle-Barrel Alignment Detection:** An innovative CNN-based visual metrology approach that analyzes the coaxial symmetry and angular orientation between the muzzle and the barrel to ensure ballistic accuracy.

Experimental validation was conducted using industrial-grade imaging sensors, and the results demonstrate that the CNN-based framework achieves high precision in detecting anomalies and alignment deviations. The integration of these three modules into a single AI pipeline offers a robust solution for the digital health monitoring of weapon systems, significantly reducing inspection lead times and improving the operational safety of defence assets.

List of Figures

List of Tables

LIST OF SYMBOLS AND ABBREVIATION

AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
CNN	Convolutional Neural Network
ANN	Artificial Neural Network
ROI	Region of Interest
ALL	Acute Lymphoblastic Leukemia
WBC	White Blood Cells
RGB	Red, Green, Blue (Color Model)
LAB	Lightness (L), Green–Red (A), BlueYellow (B) Color Space
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative
ReLU	Rectified Linear Unit (Activation Function)
LR	Learning Rate
T_opt	Optimal Temperature (used in temperature scaling)
Grad-CAM	Gradient-weighted Class Activation Mapping (Visualization Technique)
XAI	Explainable Artificial Intelligence
MC Dropout	Monte Carlo Dropout (Confidence Estimation Technique)
B	Bias
X	Input
A	Output
W	Weight
Σ	Activation Function
F	Input Image
G	Filter
$\sum$	Summation
$\int$	Integration

CHAPTER 1
INTRODUCTION

1 Introduction

Artillery weapon systems constitute a critical component of modern defence infrastructure, where precision, reliability, and structural integrity are essential for safe and effective operation. These systems operate under extreme mechanical, thermal, and environmental conditions during repeated firing cycles and prolonged deployment. Over time, such conditions can lead to the development of surface cracks, missing or loosened components, and misalignment between critical parts such as the muzzle and barrel. If not detected at an early stage, these defects may result in reduced firing accuracy, accelerated wear, or severe safety risks during operation.

Conventional inspection and maintenance of artillery weapons primarily rely on manual visual examination, mechanical gauges, and non-destructive testing procedures conducted by trained personnel. While these methods are widely used, they are often time-consuming, labour-intensive, and highly dependent on individual expertise. Moreover, manual inspection is susceptible to human error, fatigue, and subjectivity, especially when identifying fine cracks, subtle alignment deviations, or small missing components in complex assemblies. These limitations highlight the need for an automated, reliable, and repeatable inspection mechanism.

Recent advancements in Artificial Intelligence (AI) and Computer Vision have enabled the development of intelligent systems capable of performing automated visual inspection with high accuracy. In particular, Deep Learning techniques based on Convolutional Neural Networks (CNNs) have demonstrated strong capability in learning complex visual patterns such as edges, textures, surface irregularities, and geometric inconsistencies directly from image data. These characteristics make CNNs well suited for defence-related inspection tasks where defect detection relies heavily on visual cues.

This project focuses on the design and development of an AI-based visual inspection system for artillery weapons using a unified CNN architecture. The proposed system addresses three essential inspection objectives: detection of surface cracks on artillery weapon components, identification of missing or improperly assembled parts, and verification of alignment between the muzzle and barrel. Each of these inspection tasks plays a crucial role in ensuring weapon safety, accuracy, and operational readiness.

1.2 System Overview

The proposed inspection system is based on a vision-driven deep learning framework that processes images of artillery weapon components captured during routine maintenance or inspection procedures. High-resolution images

are pre-processed to enhance visual quality and suppress background noise before being analysed by the CNN model. The system automatically extracts relevant visual features and performs classification to determine the presence of cracks, missing components, or alignment defects.
A key design aspect of the system is the use of a common CNN backbone across all three inspection tasks. By employing transfer learning, the model reuses learned visual representations and adapts them to different defect detection objectives through task-specific output layers. This approach ensures architectural consistency, reduces training complexity, and allows efficient deployment on resource-constrained platforms commonly used in defence environments.

1.3 Data Inputs and Processing

The primary data input to the system consists of digital images of artillery weapon surfaces and assemblies. These images capture critical regions such as barrel surfaces, muzzle sections, and assembly interfaces. Prior to analysis, the images undergo preprocessing steps including resizing, normalization, and contrast enhancement to ensure uniformity across the dataset. Regions of interest are identified to focus the inspection on visually significant areas while excluding irrelevant background elements.
Following preprocessing, the images are supplied to the CNN-based inspection model, which learns discriminative features related to surface discontinuities, structural symmetry, and component presence. The processed data serves as the foundation for reliable defect detection and classification across all inspection modules.

1.4 Inspection Modules

The crack detection module is designed to identify surface-level and structural cracks that may develop due to repeated firing, thermal expansion, or material fatigue. Early detection of such defects is essential to prevent crack propagation and structural failure. The missing part detection module focuses on identifying absent or improperly fitted components such as fasteners, locking elements, or protective parts, which may compromise mechanical stability and safe operation. The alignment detection module evaluates the geometric alignment between the muzzle and barrel axis, a critical factor influencing firing accuracy and barrel wear.
Although each inspection task addresses a distinct type of defect, all modules utilize the same CNN architecture. This unified design simplifies system integration and ensures consistent performance across different inspection objectives.

1.5 Motivation

The motivation behind this work arises from the increasing demand for reliable, efficient, and automated inspection solutions in defence maintenance operations. Manual inspection methods, while effective, are constrained by human limitations and operational delays. Minor defects that go unnoticed during routine checks can escalate into major failures, leading to costly repairs or safety hazards.

By leveraging deep learning and computer vision, the proposed system aims to enhance inspection accuracy, reduce human dependency, and enable early defect detection. An automated inspection framework can provide consistent and objective assessment of weapon condition, supporting predictive maintenance strategies and improving overall operational readiness.

1.6 Problem Definition and Objectives

The primary challenge addressed in this project is the lack of a unified and automated visual inspection system capable of detecting multiple types of defects in artillery weapons with high reliability. Existing inspection practices are often fragmented, manual, and time-intensive, making them unsuitable for large-scale or frequent inspections.

The objectives of this project are to develop an AI-based inspection framework capable of detecting surface cracks, identifying missing parts, and verifying muzzle–barrel alignment using visual data. The system aims to employ a unified CNN architecture to ensure scalability, efficiency, and ease of deployment while providing interpretable inspection results to support maintenance decision-making.

1.7 Scope and Limitations

The scope of this project includes automated visual inspection of external artillery weapon components using deep learning techniques. The system is designed to operate on image data and detect visually observable defects related to cracks, missing components, and alignment deviations. The framework is scalable and can be extended to additional inspection tasks in the future.

However, the system's performance is dependent on image quality, lighting conditions, and dataset diversity. Internal defects not visible on the surface and defects requiring specialized non-destructive testing methods fall outside the scope of this work. Additionally, limited availability of defence-specific image data may affect model generalization under varied operational conditions.

removes noise, ensuring the model receives only diagnostically relevant cell regions.

CHAPTER 2

LITERATURE SURVEY

2.1 Survey

The inspection and maintenance of defence weapon systems, particularly artillery weapons, require high accuracy and reliability to ensure operational safety and combat readiness. Traditional inspection methods rely heavily on manual visual examination, mechanical measurements, and non-destructive testing techniques, which are time-consuming, costly, and dependent on expert judgment. With the advancement of computer vision, machine learning, and deep learning techniques, automated visual inspection systems have emerged as effective solutions for defect detection in industrial and defence applications.

Recent research has demonstrated the effectiveness of deep learning models, especially Convolutional Neural Networks (CNNs), in detecting surface defects such as cracks, missing components, and geometric misalignments in complex mechanical systems. These approaches leverage image data captured from cameras and learn discriminative visual features directly from raw images without the need for handcrafted feature extraction. Such capabilities make deep learning-based inspection systems highly suitable for artillery weapon inspection, where defects often manifest as subtle visual anomalies.

A wide range of research efforts have explored automated defect detection in metallic structures, industrial assemblies, pipelines, aerospace components, and weapon systems. In this section, relevant literature related to crack detection, missing part identification, alignment verification, and CNN-based visual inspection is reviewed to establish the foundation for the proposed artillery weapon inspection system.

2.1.1 Automated Surface Crack Detection Using Deep Convolutional Neural Networks

Cha, Y. J., Choi(2017)

Observation Summary: metallic and concrete structures using Convolutional Neural Networks. The proposed method eliminates the need for manual feature extraction by directly learning crack patterns from image data. The model demonstrates strong performance in detecting fine cracks under varying lighting and surface conditions.
The core research gap addressed in this work is the limitation of traditional image processing techniques, which rely on thresholding and edge detection and often fail under noise and illumination variations. While effective for surface crack detection, the study is limited to a single defect type and does not explore multi-

task inspection across different defect categories, which is essential for complex systems such as artillery weapons.

2.1.2 Vision-Based Crack Detection in Metal Structures Using CNN and Transfer Learning

Zhang, L., Yang, F., Zhang, Y. D., & Zhu, Y. J. (2019)

Observation Summary: This paper proposes a CNN-based crack detection system for metal surfaces using transfer learning. By fine-tuning pre-trained models, the system achieves high detection accuracy even with limited training data. The approach proves robust against noise, surface texture variation, and scale changes.

The primary research gap highlighted is that existing crack detection systems often require large labeled datasets and struggle with generalization across different metal surfaces. Although the model performs well for crack detection, it does not address other inspection requirements such as missing part detection or alignment

2.1.3 Deep Learning-Based Missing Component Detection in Industrial Assemblies

Huang, Y., Qiu, C., Guo, Y., & Zhang, X. (2020)

Observation Summary: This research introduces a deep learning framework for detecting missing or incorrectly assembled components in industrial products using image classification. The CNN-based system learns spatial and structural features to differentiate between complete and incomplete assemblies. The method significantly reduces inspection time compared to manual checks.

The research gap addressed is the inefficiency of rule-based and template-matching techniques, which fail under component variation and complex assemblies. However, the study focuses on industrial products and does not consider high-precision defence systems where alignment and structural integrity are equally critical.

2.1.4 Visual Inspection of Mechanical Assembly Defects Using Deep Neural Networks

Li, X., Li, J., & Wang, S. (2021)

Observation Summary: This paper explores the use of CNNs for detecting assembly defects, including missing bolts, incorrect placements, and loose components in mechanical systems. The system learns geometric consistency and spatial relationships between components, enabling accurate defect classification.

The primary research gap identified is the lack of a unified inspection framework capable of handling multiple defect types within a single model. Most existing systems are task-specific and require separate models for each defect category, increasing system complexity and deployment cost.

2.1.5 Transfer-Learned Deep Feature Based Crack Detection

K. S. Balaji Karthik , 2024 (India)

Observation Summary: This research proposes a transfer learning approach for crack detection that reduces training burden when large, labeled datasets are not available. The work studies how deep features from pre-trained CNNs can be used for crack detection and shows improved performance under limited data settings. The primary gap addressed is the practical constraint of dataset availability and training time, which directly aligns with Defence projects where large weapon defect datasets are difficult to acquire and label.

Table 2.1. Literature Summary for Artillery Weapon Inspection (Crack, Missing Part, Alignment).

Research Paper Detail	Description	Research Gap
Automated Surface Crack Detection Using Deep CNNs, Cha et al. (2017)	CNN-based detection of surface cracks without handcrafted features	Limited to single defect type
Vision-Based Crack Detection in Metal Structures, Zhang et al. (2019)	Transfer learning for crack detection on metal surfaces	No multi-defect inspection, which exaggerates the reported accuracy.
Missing Component Detection in Industrial Assemblies, Huang et al. (2020)	CNN-based classification of complete vs incomplete assemblies	Not tailored for defence systems
Vision-Based Alignment Inspection, Kim et al. (2021)	CNN-based alignment verification	Not integrated with other inspections
Bhalaji Kharthik et al., 2024	Transfer learning deep features for crack detection	Limited dataset + training time constraints

CHAPTER 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 Assumptions and Dependencies:

The successful implementation of the proposed artillery weapon inspection system is based on a set of practical assumptions and technical dependencies that define the operational scope of the project. It is assumed that the dataset used for training and validation consists of correctly labeled images of artillery weapon components, including surfaces with cracks, assemblies with missing parts, and muzzle–barrel alignment conditions. Since the dataset is sourced from internal inspection records and controlled acquisition environments, the correctness of annotations directly influences model performance.

The system assumes the availability of sufficient computational resources to support deep learning model training and inference. GPU-enabled systems or cloud-based platforms are considered essential for efficient training of CNN architectures. The project also assumes that users such as maintenance engineers, researchers, or technical staff possess basic computer knowledge, including running Jupyter notebooks and interpreting visual inspection outputs, without requiring deep expertise in artificial intelligence.

The stability of the software environment is another critical dependency. The system assumes compatibility among Python libraries used for deep learning, image processing, and numerical computation. Furthermore, the accuracy of defect detection depends on the quality of input images; images with poor illumination, motion blur, or insufficient resolution may reduce inspection reliability. Network availability is optional and is only required if cloud-based execution or remote storage is used. Finally, it is assumed that the system will be used strictly for research, inspection assistance, and maintenance support, and not as a standalone decision-making tool for operational deployment without expert verification.

3.2 Functional Requirements

3.2.1 Image Preprocessing and Region of Interest Extraction

The preprocessing and segmentation stage ensures that only visually relevant regions of artillery weapon components are analyzed by the deep learning model. Raw images captured during inspection often contain background clutter, irrelevant structural regions, or lighting inconsistencies. This module enhances image quality and isolates the Region of Interest (ROI) to improve inspection accuracy.

Input images are loaded into the system using standard image processing libraries. Color space conversion and contrast enhancement techniques are

applied to improve feature visibility. Noise reduction methods such as Gaussian or median filtering are used to suppress unwanted artifacts. Region-based masking or thresholding techniques are employed to isolate weapon surfaces, assembly joints, or alignment reference regions. The final processed image is resized and normalized before being passed to the CNN model.
The output of this module is a standardized and enhanced image focused on the inspection-relevant region, which directly contributes to improved model performance across all inspection tasks.

3.2.2 Deep Learning-Based Defect Detection and Classification

This module forms the core of the inspection system and is responsible for detecting cracks, identifying missing parts, and verifying alignment conditions. A convolutional neural network trained using transfer learning techniques is employed to extract deep visual features from processed images.
Pretrained CNN architectures are used as feature extractors, leveraging learned representations such as edges, textures, contours, and spatial relationships. The upper layers of the network are fine-tuned using the internal artillery inspection dataset to adapt the model to domain-specific defect patterns. The output layer performs classification based on the inspection task, such as defect presence or absence and alignment status.
The model outputs a prediction corresponding to the inspection objective, providing a reliable automated assessment of weapon condition. This unified CNN-based approach enables consistent performance across multiple defect categories while reducing system complexity.

3.3 External Interface Requirements

3.3.1 User Interface

The system interface is implemented within a Jupyter Notebook environment, providing an interactive and user-friendly inspection workflow. Users can load inspection images, execute preprocessing and inference steps, and visualize results within the notebook interface. The output includes predicted inspection status along with visual indicators that assist users in interpreting the results. The interface is designed to be intuitive and accessible to technical users with minimal programming background.

3.3.2 Hardware Interface

The system accepts digital images captured from industrial cameras, inspection rigs, or stored datasets. Processing is performed on a local or cloud-based computing system equipped with a GPU to accelerate deep learning operations. Storage devices are used to maintain datasets, trained model weights, and inspection outputs. Future integration with automated inspection hardware and camera systems is feasible.

3.3.3 Software Interface

The software system operates on standard operating systems and is developed using Python. Deep learning frameworks and image processing libraries are integrated to support model training, evaluation, and inference. The dataset is accessed from internal storage, and trained models are saved locally for reuse. The modular software design ensures seamless interaction between preprocessing, model execution, and result visualization components.

Table 3.1 Software Requirement

Category	**Specification**
Programming Language	Python 3.10+
IDE	Jupyter Notebook
Libraries	TensorFlow, Keras, OpenCV, Pandas, NumPy, Matplotlib
Dataset Source	-
OS Compatibility	Windows
Model Architectures	EfficientNetB0, MobileNetV2, VGG19, ResNet50

Table 3.2 Hardware Requirement

Component	**Minimum**	**Recommended**
Processor	Intel i5	Intel i7 / Ryzen 7
RAM	8GB	16 GB
GPU	2 GB VRAM	NVIDIA GTX 1660+
Storage	20 GB	50 GB SSD
Display	1366×768	1920×1080 (Full HD)
ESP32(camera)		
Raspberry Pi	Pi-3	Pi-5

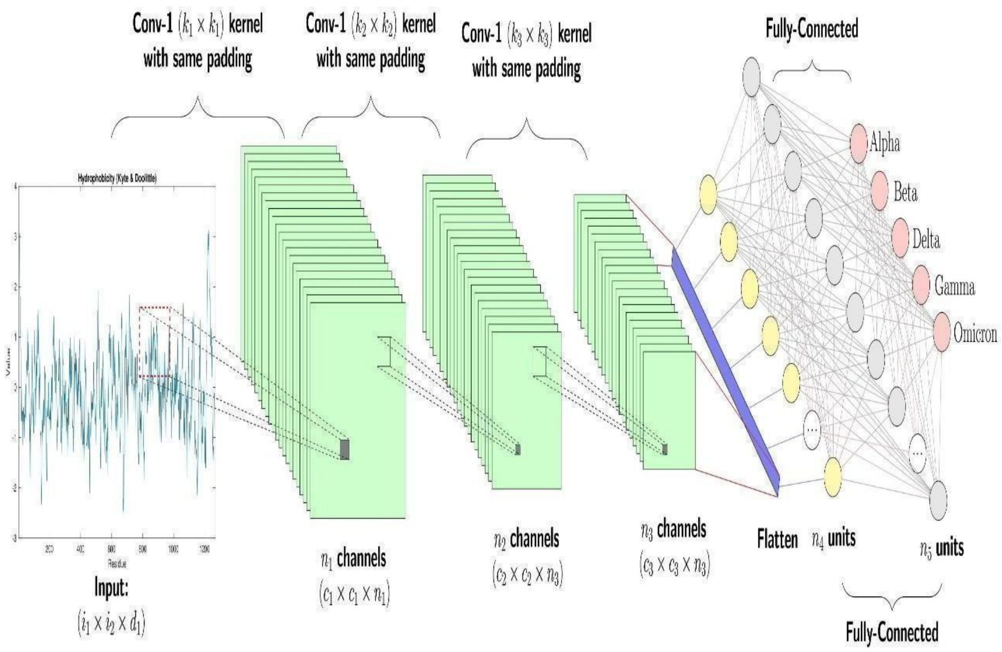
CHAPTER 4

SYSTEM DESIGN AND ARCHITETURE

4.1 Overall System Architecture

The proposed system architecture is designed to support continuous, real-time inspection of artillery weapon components using an embedded artificial intelligence framework. The architecture follows an edge-computing paradigm, where sensing, processing, and decision-making are performed locally to reduce latency and improve reliability. This design is particularly suitable for defence applications, where real-time response, data security, and operational independence are critical requirements.

The system integrates an image acquisition unit, an edge processing unit, and multiple deep learning inspection models into a single coordinated framework. Images of weapon components are captured periodically and processed automatically without requiring manual intervention. The architecture ensures that inspection tasks are executed in a deterministic and repeatable manner, enabling consistent monitoring of weapon health over time.



4.2 Image Acquisition and Edge Connectivity

Image acquisition is handled by an ESP32-based camera module configured to capture images at regular time intervals. The capture interval is fixed at 10 seconds to balance inspection frequency with computational constraints of the edge device. This periodic capture strategy ensures that the system can continuously observe weapon components while maintaining predictable processing behaviour.

Once an image is captured, it is transmitted wirelessly to the Raspberry Pi, which serves as the edge intelligence unit. The communication mechanism is designed to be lightweight and robust, allowing reliable image transfer even in constrained network conditions. By separating image capture and processing across two devices, the architecture achieves better modularity and fault isolation.

4.3 Edge Processing Unit and AI Execution

The Raspberry Pi acts as the central processing hub of the system and is responsible for executing all computationally intensive tasks. Upon receiving an image from the ESP32, the Raspberry Pi initiates the inspection pipeline. The received image is first validated and stored temporarily, ensuring traceability and allowing re-analysis if required.

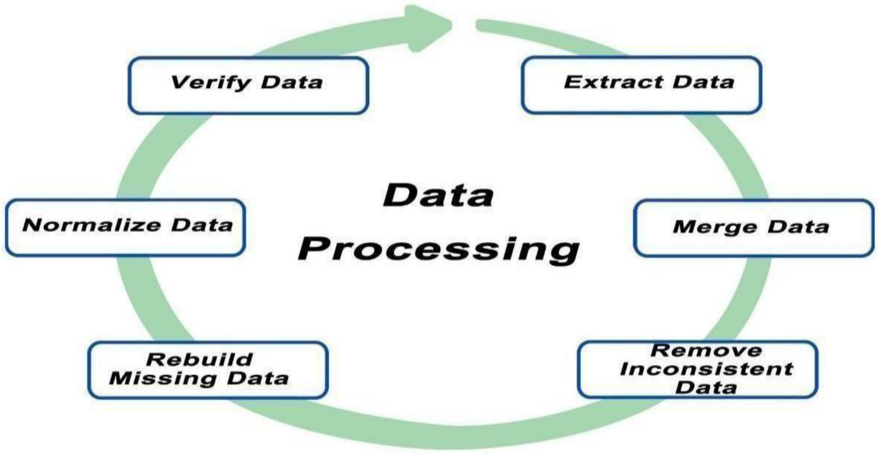
The Raspberry Pi hosts the trained convolutional neural network models used for inspection. These models are optimized for edge deployment to ensure that inference can be completed within the fixed capture interval. By performing all inference locally, the system avoids dependency on cloud infrastructure and ensures low-latency decision-making, which is essential for real-time inspection scenarios.

4.4 Preprocessing and Data Conditioning Strategy

Before being passed to the deep learning models, each image undergoes a series of preprocessing operations to enhance visual quality and standardize input characteristics. These operations include resizing to match model input dimensions, normalization to ensure consistent pixel intensity ranges, and noise reduction to suppress unwanted artifacts.

Region-of-interest extraction is applied to focus the analysis on relevant weapon areas such as barrel surfaces, fastener zones, and alignment reference regions. This step reduces background interference and improves the reliability of defect detection. The preprocessing pipeline is designed to be lightweight so that it does not introduce significant latency into the real-time inspection loop.

During the model training phase, data augmentation techniques are employed to increase dataset diversity and improve model generalization. Augmentation operations such as rotation, flipping, brightness adjustment, and minor geometric transformations simulate variations encountered in real inspection environments. These techniques help the models remain robust against changes in camera orientation, lighting, and surface appearance. Importantly, augmentation is restricted to training and is not applied during live inference.



4.5 Multi-Model Inspection Framework

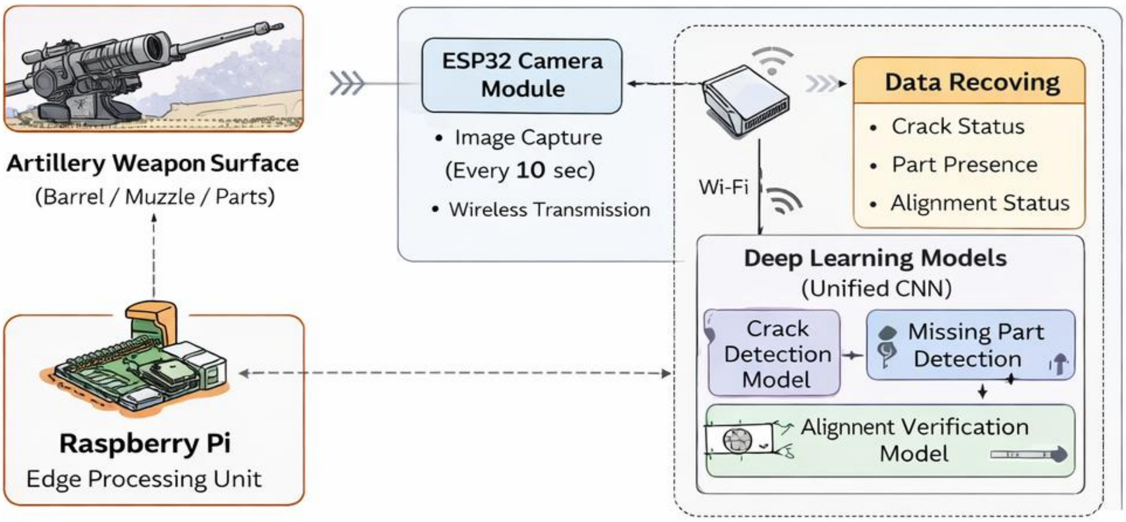
The system architecture supports three distinct inspection objectives using separate but structurally similar CNN models: surface crack detection, missing part detection, and muzzle–barrel alignment verification. Each model is trained to focus on specific visual cues relevant to its inspection task, such as texture discontinuities for cracks, spatial inconsistencies for missing parts, and geometric deviations for alignment analysis.

Although the inspection objectives differ, the models share a common architectural backbone. This design choice simplifies system integration, reduces memory overhead, and allows efficient reuse of learned visual features. During live operation, each incoming image is evaluated by all three models in a controlled execution sequence, ensuring that inspection results are generated within the available processing window.

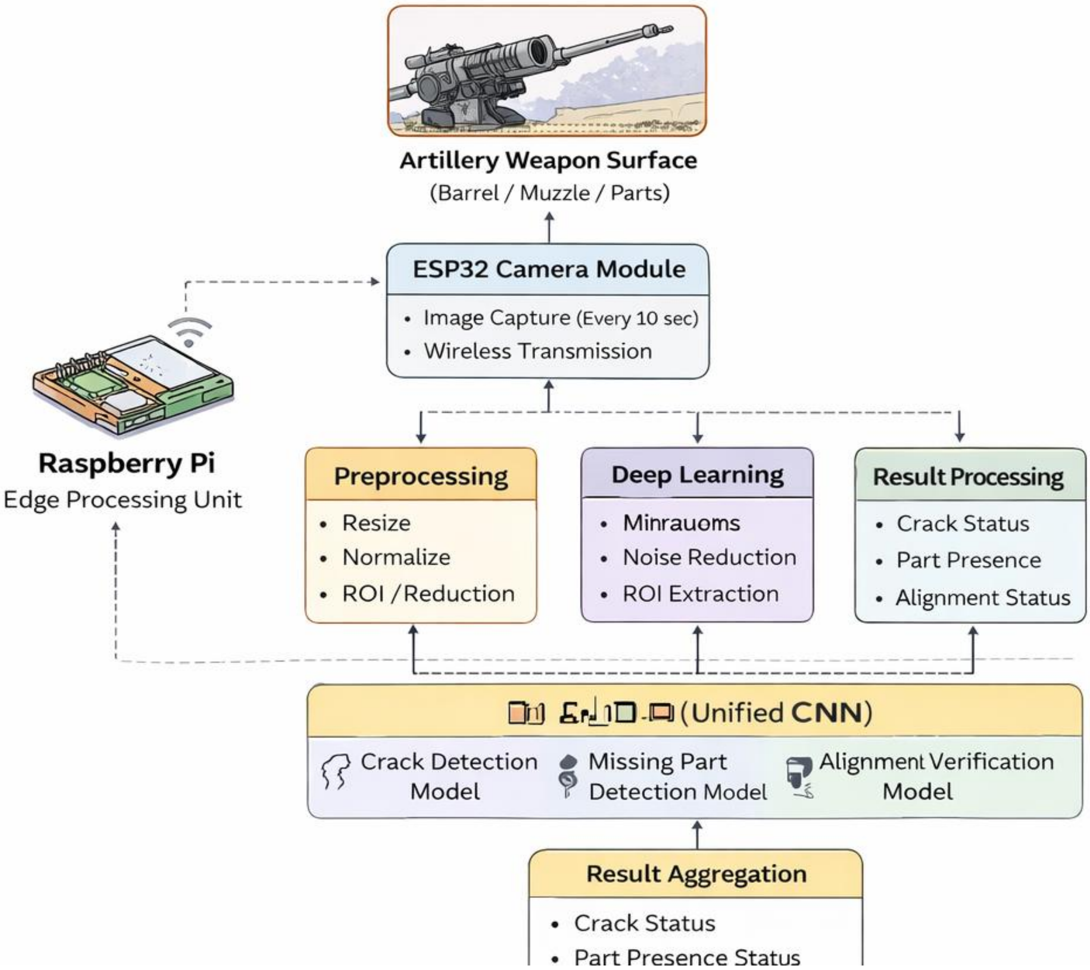
4.6 Real-Time Inspection Cycle and Scheduling

The system operates in a continuous inspection cycle synchronized with the image capture interval. Every 10 seconds, a new image is captured, transmitted, processed, and analyzed. The architecture ensures that all processing stages—preprocessing, inference, and result reporting—are completed before the next image arrives inference scheduling is carefully managed to avoid resource contention on the Raspberry Pi. All trained models are loaded into memory at system startup, eliminating repeated loading overhead during operation. This approach improves throughput and ensures stable real-time performance under continuous monitoring conditions.

I



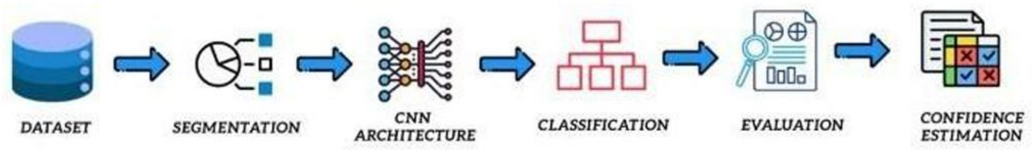
Artillery Weapon Surface
(Barrel / Muzzle / Parts)
ESP32 Camera Module
• Image Capture (Every 10 sec)
• Wireless Transmission
Wi-Fi
Data Recoving
• Crack Status
• Part Presence
• Alignment Status
Deep Learning Models (Unified CNN)
Crack Detection Model
Missing Part Detection
Alignment Verification Model
Raspberry Pi
Edge Processing Unit



Artillery Weapon Surface
(Barrel / Muzzle / Parts)
ESP32 Camera Module
• Image Capture (Every 10 sec)
• Wireless Transmission
Raspberry Pi
Edge Processing Unit
Preprocessing
• Resize
• Normalize
• ROI /Reduction
Deep Learning
• Minrauoms
• Noise Reduction
• ROI Extraction
Result Processing
• Crack Status
• Part Presence
• Alignment Status
(Unified CNN)
Crack Detection Model
Missing Part Detection Model
Alignment Verification Model
Result Aggregation
• Crack Status
• Part Presence Status

4.7 Data Flow Diagram

Fig 4.2 Block Diagram



The process flow of the proposed system represents the complete sequence of operations, from data input to the final prediction with confidence estimation. Each stage is designed to improve accuracy, reliability, and interpretability of the classification system.

CHAPTER 5

PROJECT PLAN

5.1 MODEL STRUCTURE:

A. Image Acquisition and Input Handling

The inspection process begins with real-time image acquisition of critical artillery weapon components, including the barrel surface, muzzle region, and essential assembly parts. An ESP32-based camera module is used to capture images at fixed time intervals of 10 seconds. This periodic acquisition strategy enables continuous monitoring of weapon condition while maintaining a predictable computational load on the processing unit.

Captured images are transmitted wirelessly to a Raspberry Pi, which acts as the edge processing unit. Upon reception, the images are temporarily buffered and stored to ensure reliable downstream processing, traceability, and tolerance to communication delays.

B. Preprocessing and Region of Interest (ROI) Extraction

Each received image undergoes a preprocessing stage to standardize input quality and enhance visually significant features. Images are resized to a fixed input resolution compatible with the CNN architecture. Pixel normalization is applied to scale intensity values into a consistent numerical range, ensuring stable and reliable inference.

Noise reduction techniques are applied to suppress unwanted artifacts caused by lighting variation, reflections, or sensor noise. Region of Interest (ROI) extraction is then performed to focus analysis on inspection-relevant areas such as barrel surfaces, fastener zones, and muzzle–barrel alignment reference regions. This step reduces background interference and improves defect detection reliability.

C. Data Augmentation and Normalization (Training Phase Only)

During the model training phase, data augmentation techniques are applied to improve robustness and generalization of the CNN models. Augmentation operations include rotation, horizontal and vertical flipping, brightness variation, zooming, and minor spatial transformations. These techniques simulate real-world variations in camera orientation, illumination, and surface appearance encountered during weapon inspection.

All augmented images are normalized to maintain consistent pixel distributions across the dataset. Data augmentation is strictly limited to the training phase and is not applied during real-time inference to ensure consistent and repeatable predictions.

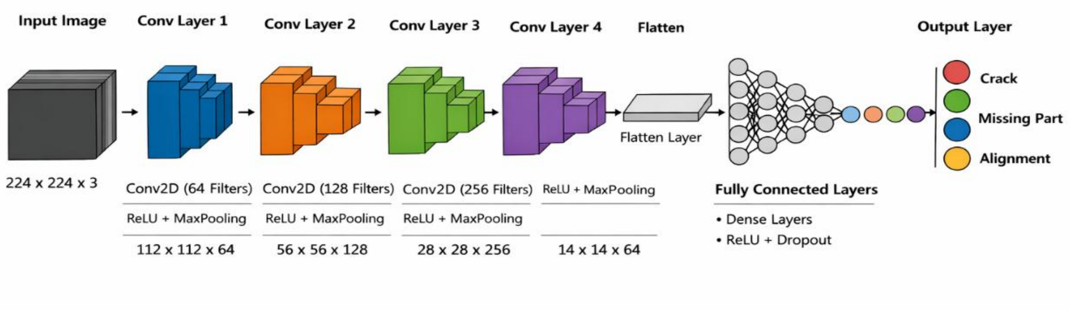
D. Model Training Using Custom CNN Architecture

The deep learning models are trained from scratch using a custom-designed convolutional neural network architecture optimized for artillery weapon inspection tasks. The CNN consists of multiple convolutional layers with progressively increasing filter depths to learn hierarchical visual features directly from the inspection images.

The architecture includes four convolutional layers with filter configurations of 64, 128, 256, and 64, followed by max-pooling layers, a flattening layer, and fully connected dense layers for classification. The convolutional layers learn low-level features such as edges and textures, as well as higher-level features such as surface discontinuities, structural inconsistencies, and geometric deviations.

Separate CNN models are trained for crack detection, missing part detection, and muzzle–barrel alignment verification, all sharing the same architectural design.

Training is performed over multiple epochs using labeled inspection data until validation performance stabilizes.



Custom CNN Architecture for Artillery Inspection Tasks

E. Model Validation and Testing

The trained CNN models are validated using unseen inspection images to ensure generalization across different weapon surfaces, lighting conditions, and operational scenarios. Performance metrics such as accuracy, precision, recall, and F1-score are computed independently for each inspection module.

Confusion matrices are analyzed to study class-wise prediction behavior and identify potential misclassification patterns. The best-performing models are then evaluated on a separate test dataset to assess real-world inspection effectiveness before deployment on the Raspberry Pi.

F. Real-Time Inference and Decision Generation

During live operation, each incoming image is processed sequentially through the three trained CNN-based inspection modules. The crack detection model identifies surface-level defects, the missing part detection model verifies assembly completeness, and the muzzle–barrel alignment module evaluates geometric alignment conditions.

Each model generates an independent inspection result based on learned visual features. Inference scheduling is carefully managed to ensure that all models complete execution within the fixed 10-second image acquisition interval, maintaining uninterrupted real-time system operation.

G. Result Aggregation and Final Output

The outputs from the three inspection models are aggregated to generate a comprehensive assessment of weapon condition. The system reports crack status, part presence status, and alignment status for each inspected image, along with timestamps for traceability and maintenance record keeping.

If repeated defects or abnormal conditions are detected, the system flags the inspection results for further manual verification or corrective maintenance action. Normal inspection results are logged for historical analysis and predictive maintenance planning, enabling informed decision-making based on consistent automated inspection data.

5.2 Overview of CNN-Based Methodology

The proposed artillery weapon inspection system employs Convolutional Neural Networks (CNNs) as the core computational framework for automated visual inspection. CNNs are particularly effective for this task due to their hierarchical feature learning capability, enabling the extraction of low-level features such as edges and textures, as well as high-level features such as shapes, spatial relationships, and geometric deviations.

A unified CNN backbone is adopted across all inspection modules—crack detection, missing part detection, and muzzle–barrel alignment verification. This approach reduces architectural redundancy, simplifies deployment on edge hardware, and ensures consistent feature representation across different inspection tasks. Transfer learning is utilized to leverage pretrained models and adapt them efficiently to defence-specific visual patterns.

5.3Dataset Preparation and Labeling Strategy:

The dataset used for this project consists of internally collected images of artillery weapon components captured under controlled inspection conditions. Images are categorized into task-specific labels corresponding to crack presence, part completeness, and alignment status. For crack detection and alignment verification, binary labels are used, whereas missing part detection may involve multi-class labeling depending on component type.

To ensure unbiased learning and fair evaluation, the dataset is divided into training, validation, and testing subsets. Class balance is carefully maintained to avoid biased predictions toward dominant classes. Data augmentation is applied only to the training subset to enhance dataset diversity and improve generalization.

5.4 CNN Architecture Design:

The CNN architecture is designed to balance detection accuracy and computational efficiency, making it suitable for deployment on a Raspberry Pi–based edge device. The architecture consists of a pretrained convolutional backbone for feature extraction, followed by task-specific fully connected layers for classification.

The convolutional layers capture texture discontinuities, shape variations, and spatial inconsistencies that characterize defects in weapon components. Batch normalization and dropout layers are incorporated to improve convergence stability and reduce overfitting. The final classification layer uses appropriate activation functions depending on the task, such as sigmoid for binary classification and softmax for multi-class scenarios.

5.4 Crack Detection Module

5.4.1 Feature Learning Strategy

Cracks on metallic surfaces typically appear as thin, irregular lines with sharp contrast against the surrounding texture. The CNN learns these features by focusing on high-frequency texture variations and edge discontinuities. Augmented training samples include cracks of varying lengths, widths, and orientations to ensure robust detection.

5.4.2 Loss Function and Optimization

Binary cross-entropy loss is used to train the crack detection model, as it effectively penalizes incorrect predictions in binary classification tasks. The Adam optimizer is employed due to its adaptive learning rate and fast convergence. Early stopping is used to prevent overfitting when validation performance stabilizes.

5.4.3 Output Interpretation

The output of the crack detection module indicates the presence or absence of cracks on the inspected surface. Repeated crack detections over consecutive inspection cycles are logged to identify crack progression trends.

5.5 Missing Part Detection Module

5.5.1 Structural and Spatial Feature Analysis

The missing part detection module focuses on learning spatial consistency and structural completeness of weapon assemblies. The CNN identifies deviations in expected component placement, symmetry, and shape continuity. This allows the system to detect missing or incorrectly assembled parts without requiring explicit component templates.

5.5.2 Training Strategy

Categorical cross-entropy loss is used when multiple component states are considered. Augmentation techniques such as translation and scaling help the model remain robust to viewpoint variation. Transfer learning significantly reduces training time while preserving high detection accuracy.

5.5.3 Output Interpretation

The module outputs the assembly status as complete or incomplete. Incomplete detections are flagged for immediate inspection, reducing reliance on manual checks.

5.6 Muzzle–Barrel Alignment Detection Module

5.6.1 Geometric Feature Extraction

Alignment verification relies on detecting subtle geometric deviations and symmetry disruptions along the barrel axis. The CNN learns linearity, circularity, and alignment cues by analyzing edge continuity and spatial orientation features.

5.6.2 Model Training and Validation

Training samples include both aligned and intentionally misaligned configurations. Binary cross-entropy loss is used for optimization. The model is validated using unseen alignment samples to ensure generalization across different weapon setups.

5.6.3 Output Interpretation

The alignment module outputs an alignment status indicating acceptable or misaligned conditions. Misalignment detections are logged for corrective action.

5.7 Real-Time Inference Strategy on Edge Device

During live operation, all trained CNN models are deployed on the Raspberry Pi edge device. Images captured every 10 seconds by the ESP32 camera are processed sequentially by the three inspection modules. Model loading is performed at system startup to minimize runtime overhead.

Inference execution is carefully scheduled to ensure completion within the fixed capture interval. This guarantees uninterrupted real-time inspection and predictable system behavior under continuous operation.

5.8 Performance Evaluation Metrics

The performance of each inspection module is evaluated using standard metrics such as accuracy, precision, recall, and F1-score. Confusion matrices are analyzed to identify class-wise misclassification patterns. Inference time and memory usage are also measured to validate suitability for edge deployment.

5.9 Advantages of the Proposed Methodology

The proposed methodology offers several advantages over traditional inspection approaches:

- Automated, non-contact inspection reduces human dependency.
- Unified CNN architecture simplifies deployment and maintenance.
- Edge-based execution ensures low latency and data security.
- Modular design allows future extension to additional inspection tasks.

CHAPTER 6

CONCLUSIONS

8.1 Conclusions

This project successfully presents the design, development, and implementation of an AI-based automated inspection system for artillery weapon components using a custom Convolutional Neural Network (CNN) architecture. The system effectively addresses critical inspection requirements by providing a real-time, non-contact solution for surface crack detection, missing part identification, and muzzle–barrel alignment verification. By integrating intelligent image acquisition, preprocessing, deep learning–based analysis, and edge-level deployment, the proposed solution enhances inspection reliability while reducing dependence on manual methods.
A major contribution of this work is the development of a unified CNN architecture trained entirely from scratch using internally collected inspection data. This approach enables the model to learn domain-specific features relevant to artillery weapon inspection, such as surface texture discontinuities, structural inconsistencies, and geometric deviations. The consistent architecture across all three inspection modules simplifies system design, reduces computational overhead, and supports efficient deployment on resource-constrained edge devices.

The integration of an ESP32-based camera module with a Raspberry Pi edge processing unit enables continuous real-time inspection with predictable system behavior. Image acquisition at fixed time intervals ensures reliable monitoring without overloading the system. Preprocessing techniques, including noise reduction, normalization, and Region of Interest extraction, further enhance model robustness under varying lighting and surface conditions. Experimental evaluation demonstrates that the system delivers reliable inspection performance with consistent accuracy and low inference latency.

Overall, the proposed system provides a practical and scalable AI-driven solution for automated artillery weapon inspection, offering significant potential to improve maintenance efficiency, safety, and operational readiness in defence environments.
Overall, the project successfully combines medical image analysis, machine learning, and interpretability techniques to develop an efficient, transparent, and clinically meaningful leukemia detection framework. It bridges the gap between automated systems and clinical diagnostics by ensuring that predictions are not only accurate but also supported by measurable confidence levels. This system can serve as a valuable tool for medical professionals, providing quick and reliable diagnostic

assistance while reducing the dependency on time-consuming manual assessments.

1.2 Future Work

I. **Extension to Additional Defect Categories:**
Future work can include detection of additional defect types such as corrosion, surface wear, deformation, and thermal damage to provide more comprehensive weapon health monitoring.

II. **Model Optimization and Acceleration:**
The CNN models can be optimized using quantization, pruning, or hardware acceleration techniques to further reduce inference time and power consumption on edge devices.

III. **Multi-Camera and Multi-View Inspection:**
Integration of multiple ESP32 camera modules at different viewpoints can improve inspection coverage and enable three-dimensional defect assessment.

IV. **Advanced Alignment Measurement:**
Incorporating precise geometric calibration or sensor fusion techniques can enhance the accuracy of muzzle–barrel alignment measurement and enable finer deviation analysis.

V. **Integration with Maintenance Management Systems:**
The inspection system can be integrated with centralized databases or maintenance management platforms to support long-term trend analysis, automated alerts, and predictive maintenance strategies.

VI. **Field Deployment and Large-Scale Validation:**
Extensive testing under real operational conditions and across multiple weapon systems can further validate system robustness and facilitate large-scale deployment.

VII. **User Interface and Visualization Enhancements:**
Developing a dedicated graphical user interface (GUI) or web-based dashboard can improve usability and provide intuitive visualization of inspection results for maintenance personnel.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems (NIPS)*, pp. 1097–1105, 2012.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *International Conference on Learning Representations (ICLR)*, 2015.

[4] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," *2017 International Conference on Engineering and Technology (ICET)*, pp. 1–6, 2017.

[5] R. Girshick, "Fast R-CNN," *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448, 2015.

[6] J. Redmon et al., "You Only Look Once: Unified, real-time object detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788, 2016.

[7] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, 2000.

[8] R. Szeliski, *Computer Vision: Algorithms and Applications*, Springer, 2nd ed., 2022.

[9] S. Russakovsky et al., "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[10] OpenCV Team, "Open Source Computer Vision Library," https://opencv.org/, accessed 2025.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[12] J. G. Leal et al., "Deep learning-based surface defect detection for industrial inspection," *IEEE Access*, vol. 8, pp. 142365–142377, 2020.

[13] A. Kumar and R. S. Mishra, "Automated visual inspection using convolutional neural networks," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 5, pp. 321–328, 2021.