National College of Ireland

National College of Ireland

Project Submission Sheet

Student Name:	Zuu Zuu Kyaw Shwe
Student ID:	24106585
Programme:	MSCDAD_C **Year:** 2024 - 2025
Module:	Domain Applications
Lecturer:	Vikas Sahni
Submission Due Date:	4th Apr 2025
Project Title:	Domain Applications Project
Word Count:	3106

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
<u>ALL</u> internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature: Zuu

Date: 4th Apr 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS:

1. Please attach a completed copy of this sheet to each project (including multiple copies).
2. Projects should be submitted to your Programme Coordinator.
3. **You must ensure that you retain a HARD COPY of ALL projects**, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. **Late submissions will incur penalties.**
5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

AI Acknowledgement Supplement

Domain Applications

Project

Your Name/Student Number	Course	Date
Zuu Zuu Kyaw Shwe/24106585	Domain Applications	4th Apr 2025

This section is a supplement to the main assignment, to be used if AI was used in any capacity in the creation of your assignment; if you have queries about how to do this, please contact your lecturer. For an example of how to fill these sections out, please click here.

AI Acknowledgment

This section acknowledges the AI tools that were utilized in the process of completing this assignment.

Tool Name	Brief Description	Link to tool
Chatgpt	Clarify concepts and provide additional insights on complex topics	https://chatgpt.com/

Description of AI Usage

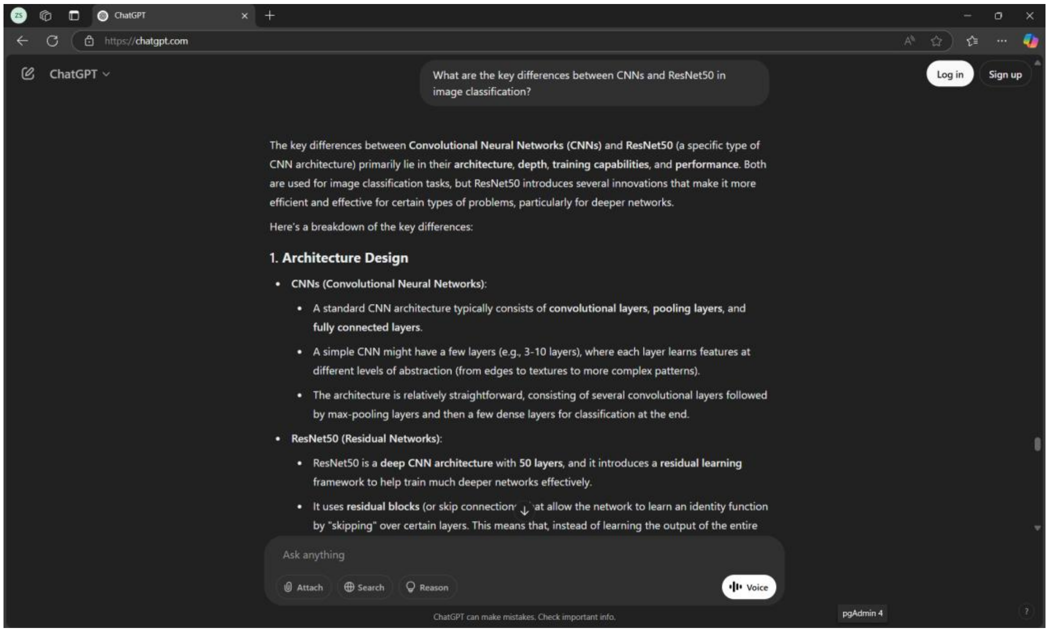
This section provides a more detailed description of how the AI tools were used in the assignment. It includes information about the prompts given to the AI tool, the responses received, and how these responses were utilized or modified in the assignment. **One table should be used for each tool used**.

ChatGPT	
For some of the questions and concepts explored, I sought help from OpenAI's ChatGPT, which helped provide additional understanding.	
What are the key differences between CNNs and ResNet50 in image classification?	1. Architecture Design **CNNs (Convolutional Neural Networks):** • **A standard CNN architecture typically consists of convolutional layers, pooling layers, and fully connected layers.** **ResNet50 (Residual Networks)**: • ResNet50 is a **deep CNN architecture** with **50 layers**, and it introduces a **residual learning** framework to help train much deeper networks effectively.

Evidence of AI Usage

This section includes evidence of significant prompts and responses used or generated through the AI tool. It should provide a clear understanding of the extent to which the AI tool was used in the assignment. Evidence may be attached via screenshots or text.

Additional Evidence: ChatGPT



Domain Application of Deep Learning for Casting Defect Detection

Zuu Zuu Kyaw Shwe
School of Computing (MSCDAD-C)
National College of Ireland
Dublin, Ireland
x24106585@student.ncirl.ie

Abstract

This project focuses on automating defect detection in casting products using image classification techniques with deep learning. A custom CNN was created with a ResNet50 and the ability to classify images of casting parts into acceptable or defective categories. The model was trained and evaluated with a public dataset and achieved strong accuracy as well as generalization performance. The solution aims to replace manual inspection processes with real-time, reliable, and cost-effective AI-driven quality control in industrial manufacturing.

Index Terms

Casting Defects, Image Classification, Convolution Neural Network, ResNet50

I. Introduction

Casting is the most essential manufacturing process for generating metal parts, which involves pouring molten metal into a mold and allowing it to cool. Although casting is one of the most widely used techniques for manufacturing parts in different industries, it can result in numerous defects, including; shrinkage, cracking, inclusions, and porosity, all of which may affect the structure as well as the functionality of the final product. Such flaws must be diagnosed to ensure that the established quality criteria are met in order to reduce economic damages stemming from faulty products. Traditionally, visual inspection, which is time-consuming and subject to immense human error and variability, has been the most common method of finding defects. With growing automation and precision manufacturing, finding defects in components has to be done faster and with greater accuracy.

The combination of computer vision and deep learning techniques has made automated defect detection possible to increase the accuracy, dependablity, and scope of quality control in manufacturing. With the advancement of technology, the performance of image classification tasks and defect detection tasks has improved remarkably by the use of deep learning models, particularly Convolutional Neural Networks (CNNs). This project aims at creating a casting defect detection system through automation using a deep learning model based on the pre-trained ResNet50 and a Custom CNN model designed for feature extraction enhancement.

The intent of this project is to build a classification model aimed at efficiently identifying defects in casting components. To exploit feature extraction and domain-specific learning, the deep learning model is constructed using Custom CNN with ResNet50. The effectiveness of the model is assessed through important metrics including accuracy, confusion matrix, and classification report. Furthermore, the model is trained and the untrained images are later evaluated to measure the real-world applicability of the model.

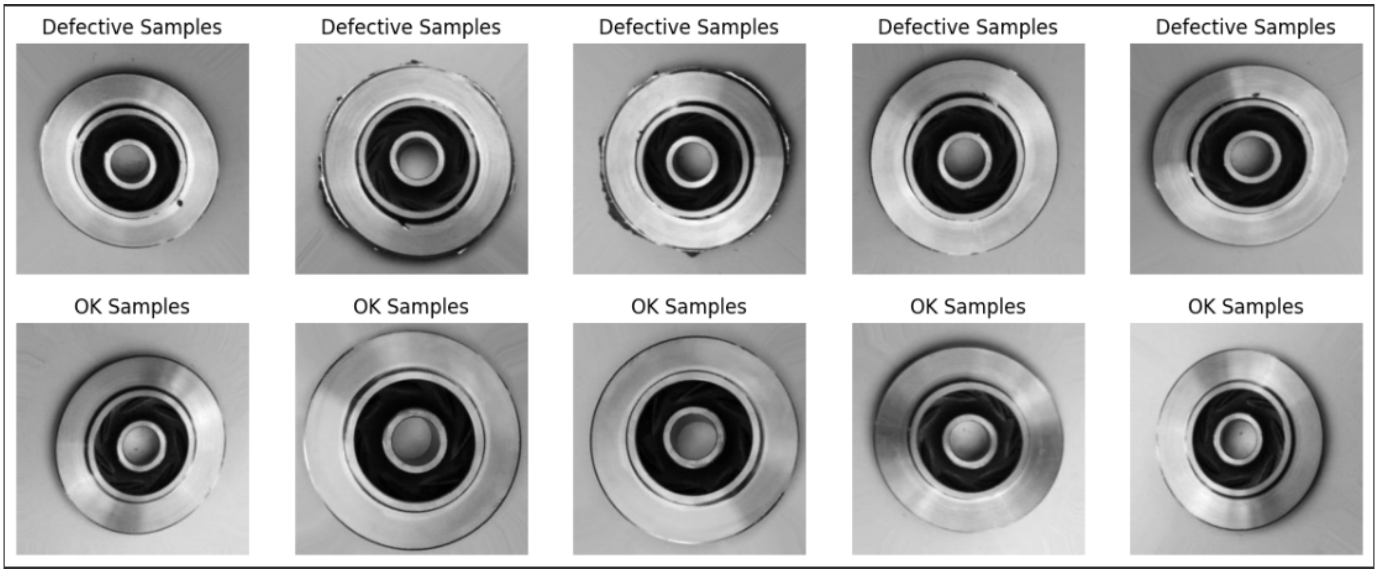


Fig. 1: Sample Casting Images

II. Goals & Business Value

The project's primary goal is to implement an automated deep learning system for casting defect detection to improve the standards of quality control in manufacturing. Inspection through manual methods is subject to inconsistencies, time-consuming and prone to human error. This project's objective is to enhance defect classification accuracy and scalability by utilizing a deep learning approach that employs a combination of ResNet50 and Custom CNN. The model will accurately classify casting products into defected and non-defected categories with the utmost precision to ensure only quality components advance to subsequent production stages.

From a business perspective, the automated defect detection system has numerous advantages. To start, it alleviates labor expenditures and human participation in quality control, shifting workers to more sophisticated responsiblities. This also allows for greater consistency in defect identification and decreases the chances of products failing to meet standards, which could be financially damaging due to unnecessary recalls and reputational losses. Moreover, identifying defects at an earlier stage enables manufacturers to identify problems in the casting process, allowing them to make the necessary changes to enhance production quality and efficiency.

This project's scalability is another important feature. Because the model is developed based on a real-world industrial dataset, it can be smoothly adapted into existing manufacturing systems with slight changes. The model, once deployed, would be capable of real-time monitoring, conducting high-speed inspections of casting products, thus increasing outputs and alleviating production bottlenecks. The ability to continuously improve the model through additional training with new defect patterns further strengthens its long-term value in manufacturing environments.

In addition to the possible benefits in manufacturing, this project also relate to the overarching trends concerning smart manufactring and Industry 4.0. Defect detection using AI will enable manufacturers to partially automate production lines, enhancing product traceability and reliability. Ultimately, the goal is to create a comprehensive AI-based quality control system that optimizes operation, minimizes waste, and increases customer satisfaction.

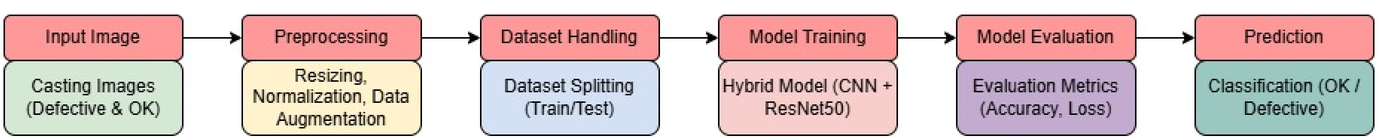


Fig. 2: Proposed Methodology

III. Literature Review

An ensemble learning approach integrating ResNet50 and a custom CNN has been developed to enhance classification performance. Evaluation of several pre-trained models highlighted the optimal performance of ResNet50, which attained an accuracy of 98.18% using a domain-specific CNN in conjunction with it. This method mitigates overfitting and enhances model strength by utilizing the feature extraction capabilties of ResNet50's pre-trained networks and a domain-specific CNN that adds defect-specific feature learning. This architecture captures the balance between generalization provided by ResNet50 and specialization offered by the Custom CNN, thus making it better suited to detect subtle variations in defects. Using the Adam optimizer, a batch size of 32, and 30 epochs of training led to optimal levels of accuracy for casting defect detection **[1]**.

Another study combines MLP classifiers with pre-trained CNNs for uncertainty-aware deep learning in casting defect identifcation. The power of transfer learning is noted as models like ResNet50 and VGG16 utilize advanced feature extraction for improved classification. Many datasets containing labeled samples for casting defects are rare, so employing pre-trained models allows the network to leverage models that have already been trained on extensive datasets like ImageNet. This drastically enhances the efficiency of extracting features, especially for subtle defect patterns, without incurring much computation, time, and training costs **[2]**.

A hybrid model of CNN and Random Forest was also developed with four convolutional and max pooling layers for hierarchical feature extraction followed by classification using Random Forest algorithms. This model achieved over 95% precision, recall, and F1-scores, achieving 98% accuracy on average across ten common defect types. This research illustrates the efficacy of using CNN-based feature extraction and ensemble learning for industry quality control, applicable to the approach utilized within this project **[3]**.

Beyond the scope of CNNs, deep learning-based object detection techniques like Faster R-CNN and MobileNet-SSD have also been explored. These algorithms are aimed at achieving a balance between speed and accuracy which makes them appropriate for classifying industrial defects. Although real-time defect detection continues to be a challenge, there is new hope with advances under compatible pre-trained models such as ResNet50 together with custom CNN models designed for quality control in casting manufacturing **[4]**.

Other approaches focus on Convolutional Autoencoders (CAEs) for data augmentation, addressing dataset imbalance challenges by generating synthetic data to improve defect classification accuracy. The use of CAEs along with CNN models has proven useful in improving overall defect detection efficiency, which is relevant in improving the reliability of classification in quality control for casting **[5]**.

Based on these study, our project adopts a similar approach by integrating ResNet50 with a Custom CNN. We chose this method because it demonstrated strong performance in defect classification by combining general feature extraction from ResNet50 with targeted defect recognition through a domain-specific CNN. This aligns with our objective to build a model that generalizes well across casting images while capturing specific defect patterns unique to our dataset. The success of this method in previous work supports our implementation of a dual-model architecture for enhanced accuracy and robustness.

TABLE I: Summary of Various Defect Detection Approaches.

Ref	Methodology	Parameters Measured	Result Achieved
[1]	ResNet50 (pre-trained) + Custom CNN	Casting defect classification	Achieved 98.18% accuracy; ensemble approach improved feature learning and reduced overfitting
[2]	CNN + MLP Classifier	Feature extraction with uncertainty quantification	Improved classification reliability with transfer learning; leveraged ResNet50 and VGG16
[3]	CNN + Random Forest	Casting defect classification	Achieved 98% accuracy; high precision and recall across 10 defect types
[4]	Faster R-CNN, MobileNet-SSD	Real-time defect detection	Balanced speed and accuracy; suitable for industrial use
[5]	Convolutional Autoencoders (CAEs) + CNN	Data augmentation for defect classification	Addressed class imbalance; improved classification accuracy through synthetic data generation

IV. Ethical Concerns

The implementation of defect detection systems in casting manufacturing using deep learning technologies raises ethical issues regarding workforce displacement, data bias, model transparency, and accountability. Greater precision and efficiency from automation come at the expense of traditional roles, which face obsolescence like that of a manual inspector. Workers are also required to be educated and trained to operate with AI-powered quality control systems so that they are not excluded from opportunities in Industry 4.0.

Another ethical challenge bias on model defect detection. The model bias is likely to occur when the dataset used for training is unbalanced towards non-defective samples, which means that the algorithm is likely to ignore defective products. This poses a significant challenge within the industrial world where undetected product failures lead to catastrophic financial implications and safety issues. With the enhancement of reliability and fairness, our project aims to improve the representation of defects in the training data by applying data augmentation methods and implementing active learning techniques.

Additionally, model interpretability and decision accountability remain critical concern. Deep learning models operate as "black boxes," meaning they make defect classification decisions based on complex, non-transparent internal computations. In manufacturing environments, stakeholders must understand the rationale behind defect classification decisions, especially when rejecting products incurs financial implications.

Privacy concerns may also arise in data collection and sharing. Industrial datasets used for defect detection often contain proprietary manufacturing details. Ensuring data security, compliance with industry regulations, and restricting unauthorized access is crucial when implementing AI-based defect detection systems.

Finally, ethical considerations in AI deployment extend beyond technology to its broader societal impact. Like with any other implementer of automation, an organization will need frameworks that ensure ethical bound on governance, responsibility, and fairness does not get breached, thus encouraging transparency in decision making. There is a need to dictate a deep learning adoption policy in manufacturing for some balance between technological evolution and human-centered practices to industrial.

V. Implementation of Techniques

In this project, we applied deep learning methods to classify casting products into either defective or acceptable castings. The implementation was done on a public dataset which contains 7,348 grayscale images of submersible pump impellers, labeled either as defective (def_front) or acceptable (ok_front). The dataset was already split into 6,633 images for training and 715 for testing. As part of the data preparation, we resized the images from 300×300 pixels to 224×224 pixels so that they would be compatible with input requirements of ResNet50. Furthermore, pixel values were normalized to ensure stable training, with intensity values scaled between 0 and 1 and standardized to a mean of 0.5 and a standard deviation of 0.5.

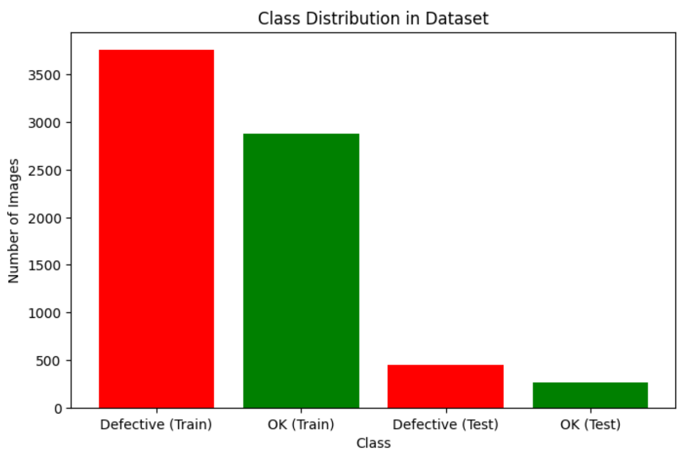


Fig. 3: Class Distribution in Dataset

To increase model generalization and prevent overfitting, we employed data augmentation techniques like random horizontal flips and slight rotations (±10°) on the training images. These features aided the model to learn to ignore rotation invariant features of rotation defects. The dataset was loaded conveniently through PyTorch's ImageFolder and DataLoader, which enabled structural automatic image labeling. The DataLoader facilitated the implementation of mini-batches, where the dataset was divided into smaller groups of 32 images per iteration. These mini-batches were shuffled to enhance model generalization.

For the model architecture, we adopted an ensemble strategy by integrating ResNet50 with a custom CNN. We utilized ResNet50's feature extraction capabilites because its early layers capture important visual features like edges and textures. However, since ResNet50 was originally trained for a 1,000-class classification problem, we replaced its final fully conneted (FC) layer with a custom classifier suited for our binary classification task. The modified FC layer included a fully connected (Linear) layer reducing the 2048-dimensional feature space to 256, followed by a ReLU activation function, a Dropout layer (p=0.4) for regularization, and a final Linear layer that outputs predictions for two classes (Defective and OK).

In complementing the feature extraction of ResNet50, we constructed a custom CNN consisting of three convolutional layers with ReLU activations and max pooling on top. This structure ensured further downsampling of feature maps, while also enriching the representations that were extracted. The final output of the custom CNN was reshaped to align with ResNet50's input expectations, enabling a seamless integration between the two networks. All these models were integrated into a single Ensemble Model in which the custom CNN processed the input image first and then sent it to ResNet50 for classification.

The model was trained with supervised learning using a cross-entropy loss function, which is effective for both binary and multi-class classification problems. To update the gradients, the Adam optimizer was chosen because of its efficacy in dealing with different gradient scales, especially when fine-tuning pre-trained networks. The learning rate was 0.0001 (1e-4) to ensure smooth integration of new represntations from ResNet50 without drastically altering its weights. The complete training pipeline was run in a GPU-parallelized setting with CUDA, which further accelerated the speed of backpropagation and parameter updating.

The entire training process took 10 epochs, tracking loss and accuracy for both the training and validation sets. Results were visualized using curves of loss and accuracy to evaluate convergence, and a confusion matrix along with classification metrics were calculated to evaluate performance. The final step involved testing the model against a previously unseen test dataset of images, resulting in a striking test accuracy of 99.72%,confirming the model's viability for real-time application in industrial settings.

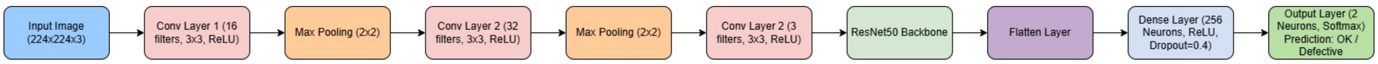


Fig. 4: Proposed CNN-ResNet50 Model Architecture

VI. Findings and Business Value Interpretation

The final model achieved exceptional performance, attaining a maximum test accuracy of 99.72% and demonstrating reliable generalization across unseen test data. Below is a summary of training and test performance across 10 epochs:

TABLE II: Training and Testing Performance Across Epochs

Epoch	Train Loss	Train Acc (%)	Test Loss	Test Acc (%)
1	0.1991	91.66	0.0764	97.20
2	0.0540	98.48	0.0273	98.88
3	0.0367	98.73	0.0157	99.30
4	0.0254	99.13	0.0265	99.16
5	0.0290	99.05	0.0103	99.44
6	0.0226	99.34	0.0137	99.44
7	0.0163	99.47	0.0086	99.86
8	0.0126	99.61	0.0182	99.30
9	0.0161	99.61	0.0151	99.44
10	0.0235	99.23	0.0212	99.72

Evaluation using the test set yielded the following classification report:

TABLE III: Classification Report

Class	Precision	Recall	F1-Score	Support
Defect	1.0000	0.9956	0.9978	453
OK	0.9924	1.0000	0.9962	262
Accuracy		0.9972		715
Macro Avg	0.9962	0.9978	0.9970	715
Weighted Avg	0.9972	0.9972	0.9972	715

The confusion matrix further confirmed minimal misclassifications: only two defective parts were incorrectly labeled as OK, and none of the OK parts were misclassified.

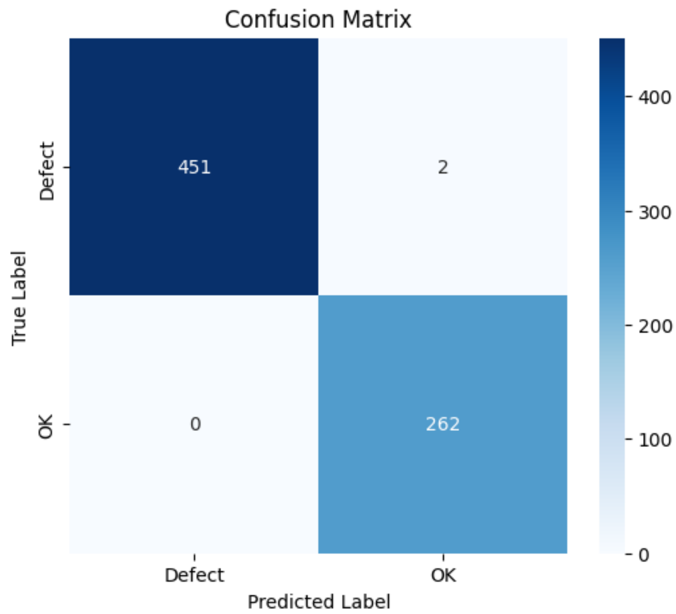


Fig. 5: Confusion Matrix

The learning curves demonstrated effective convergence — both training and validation losses steadily decreased and remained low, while training and validation accuracy increased consistently without significant divergence. This indicates that the model learned meaningful patterns rather than overfitting to the training data.

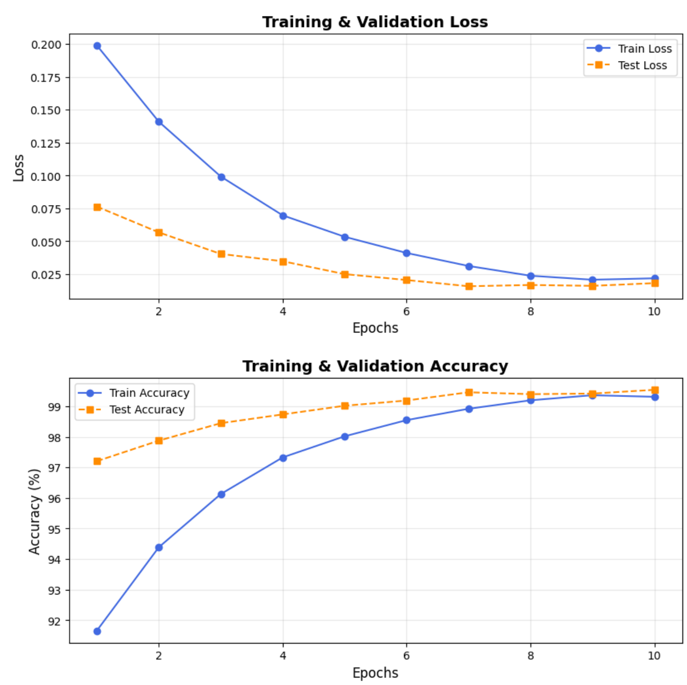


Fig. 6: Training & Validation Loss & Training & Validation Accuracy

In order to confirm the model's performance, we evaluated it with some sample images from the test set. The model correctly classified both defective and non-defective casting parts, with every prediction having a confidence level above 99%. These findings verify that the model performs well on unseen data and is able to accurately classify defective and OK components, proving its efficiency in practical quality inspection applications.

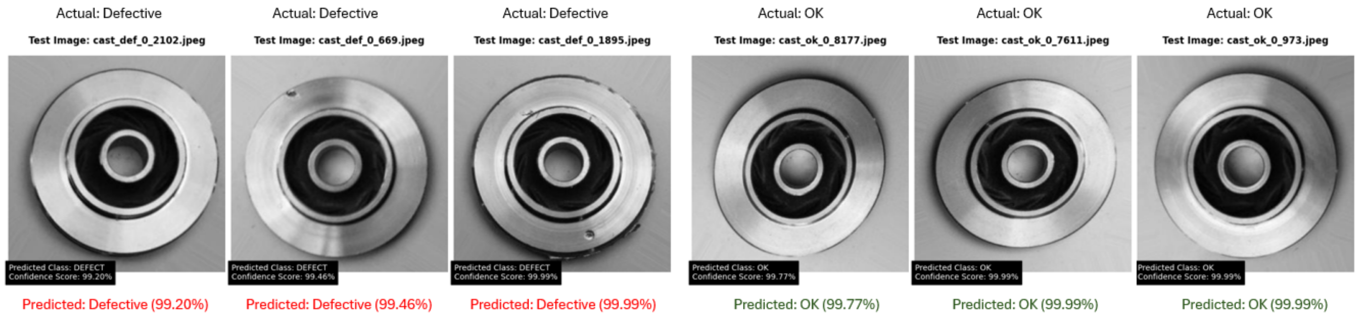


Fig. 7: Classification Results

From a business perspective, there is considerable advantage for the manufacturers with the use of the AI based defect classification system. The automatic inspection of defects does not require human oversight, which helps not only in cutting wage costs but also increases the dependability of inspection. Because of the model's reasoning ability, it can be mounted on production lines where images of components recently manufactured are taken and classified in real time. All captured defective components are marked for removal. This guarantees that products sent to clients have passed through stringent checks to reduce claims under warranty which inflates expenditrues and impacts reputation as a damaging blow.

VII. Conclusions & Future Work

The project has confirmed that deep learning-based automated quality inspection systems are feasible for industrial settings. It is capable because the transfer learning from ResNet50 used alongside a custom domain-specific CNN provided a superior feature extraction model for defect detection. The model accomplished 99.72% accuracy in efficient computational defect detection, which makes it more suitable for implementation in casting product manufacturing plants. This economically enhances inspection expenditures, operational efficiency, and parts reliability, thus increasing satisfaction and profit for manufacturers.

Compared to human inspection, these AI-powered inspection methods provide greatly improved accuracy, instantaneous information processing, and lower costs. Results illustrate that deep learning can be applied to decision-making processes related to inspections in various sectors while modernizing smart automation in processes and improving the standards of quality in the manufacturing sector.

Beyond classification accuracy, the explainability of model decisions is essential for trust and adoption in industrial settings. We did not apply Grad-CAM style visualization techniques but it could highlight regions in an image that corresponded with a model's particular decision. Automated Defect Recognition (ADR) systems would need to verify that the AI is actually implementing true defect recognition and not simply processing background data.

To enhance its utility for quality control teams, modifying the model from binary to multi-class classification would enable it to differentiate between various types of casting defects. Also, the use of active learning would support successful system post-deployment evolution through retraining with newly annotated data. Finally, obtaining a more comprehensive and diverse dataset across different manufacturing settings would enhance model reliability and improve performance and usefulness across multiple industries.

References

[1] R. Gupta, V. Anand, S. Gupta, and D. Koundal, "Deep learning model for defect analysis in industry using casting images," Expert Systems With Applications, vol. 232, p. 120758, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417423012605?via%3Dihub

[2] M. Habibpour, H. Gharoun, A. Tajally, A. Shamsi, H. Asgharnezhad, A. Khosravi, and S. Nahavandi, "An Uncertainty-Aware Deep Learning Framework for Defect Detection in Casting Products," arXiv preprint arXiv:2107.11643, 2021. [Online]. Available: https://arxiv.org/pdf/2107.11643

[3] D. Banerjee, "Advancements in Casting Defect Classification: A Comprehensive Evaluation of Deep Learning Models," Proceedings of the Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI 2024), 2024, pp. 1102-1107. [Online]. Available: https://ieeexplore.ieee.org/document/10696543

[4] A. Saberironaghi, J. Ren, and M. El-Gindy, "Defect Detection Methods for Industrial Products Using Deep Learning Techniques: A Review," Algorithms, vol. 16, no. 95, pp. 1–30, 2023. [Online]. Available: https://www.mdpi.com/1999-4893/16/2/95

[5] S. Oh, J. Cha, D. Kim, and J. Jeong, "Quality inspection of casting product using CAE and CNN," 4th International Conference on Imaging, Signal Processing and Communications (ICISPC), pp. 34-40, 2020. [Online]. Available:https://ieeexplore.ieee.org/document/9444031