I.J. Intelligent Systems and Applications, **2024, 4, 39-54**
Published Online on August 8, 2024 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2024.04.03

Product Defect Detection Using Deep Learning

Venkatesh Khemlapure
Department of Information Technology, Rajarambapu Institute of Technology, Shivaji University, Sakharale, MS-415414, India
E-mail: venkateshkhemlapure13@gmail.com
ORCID iD: https://orcid.org/0009-0009-6484-4601

Ashwini Patil*
Department of Information Technology, Rajarambapu Institute of Technology, Shivaji University, Sakharale, MS-415414, India
E-mail: ashwini.patil@ritindia.edu
ORCID iD: https://orcid.org/0000-0003-3387-5543
*Corresponding author

Nikita Chavan
Department of Information Technology, Rajarambapu Institute of Technology, Shivaji University, Sakharale, MS-415414, India
E-mail: nikitachavan1032002@gmail.com
ORCID iD: https://orcid.org/0009-0002-4197-3741

Nisha Mali
Department of Information Technology, Rajarambapu Institute of Technology, Shivaji University, Sakharale, MS-415414, India
E-mail: nishamali0781@gmail.com
ORCID iD: https://orcid.org/0009-0005-3275-3810

Received: 06 February 2024; Revised: 15 March 2024; Accepted: 07 May 2024; Published: 08 August 2024

Abstract: To maximize production efficiency, product quality control is paying more attention to the quick and reliable automated quality visual inspection. Product defect detection is a critical part of the inspection process. Manual defect detection has a lot of flaws that can be overcome using a deep learning approach. In this paper we have proposed and implemented the deep learning models to detect defects in the manufactured product. Two types of classification, i.e., binary and multiclass classification, is done using CNN, AlexNet, and YOLO algorithms. For the binary classification which is just used to check whether there is a defect in the product, we have proposed three different architectures of CNN, out of which the third CNN model gave 99.44% and 97.49% for training and testing, respectively. We also tested the AlexNet model and got accuracy of 97.6%. And for the multiclass classification that is used for identification of type(s) of defects, the YOLOv8 model is proposed and implemented, which gives better results by attaining a remarkable accuracy of 98.7% for multiclass classification. We also designed and developed the Android Application, which is used on the field for defect detection in the manufacturing industry.

Index Terms: Defect Detection, CNN, YOLO, Alexnet, Mobile Application, Binary, Multiclass, Classification.

1. Introduction

Defect Detection of products is a very important inspection task in manufacturing industries before delivery of items. Corrective action is taken based on the types of defects. Defect detection is a critical part of the inspection process, which will identify whether to accept or reject the item produced.

When using the manual or traditional method, human workers often check the product for flaws in order to detect production defects.

Each piece may be examined under various lighting circumstances during the inspection process, or faults may be checked for using specialized instruments and tools, including micro-scopes, gauges, and micro-meters. Sometimes it is very difficult to identify defects because some defects are not easily visible to the naked eye.

Workers use their judgement and skills during the inspection process to identify defects and classify them according to their findings.

The procedures for testing for defects are typically carried out at several points during the manufacturing process, such as following each stage of production or prior to the finished product being shipped to the client.

The Information about defects, such as nature of defect, position, and magnitude, is frequently recorded manually by inspection personnel. This process is very time-consuming and an investment in labor costs and also carries the risk of detecting human error.

In this paper, we have proposed and implemented a system for product defect detection in the manufacturing industry using artificial intelligence. This system aims to automatically identify and classify defects in products during the inspection process in the manufacturing industry.

The system will examine product photos and detect any irregularities that might point to flaws using a variety of AI approaches, including computer vision, machine learning, and deep learning algorithms.

The goal of the implemented system is to automate fault identification using AI, which will decrease the need for human interaction and increase the process's efficiency and accuracy.

We also implemented the mobile application, which is used for analyzing images of products captured during various stages of the manufacturing process. The deep learning algorithms used in the system will be able to identify subtle differences in the images that may indicate defects such as cracks, scratches, or other imperfections. Once the system detects a defect, it alerts the production team, who can then take appropriate action to rectify the issue.

2. Motivation and Background

The process of locating and fixing manufacturing process flaws presents a number of difficulties. One of the primary problems is the reliance on manual inspection, which can be time-consuming, subjective, and prone to human error. Human inspectors may overlook. A further issue with product fault detection is the absence of real-time feedback and monitoring systems. Without real-time monitoring and feedback, manufacturers could find it difficult to pinpoint the underlying causes of flaws and swiftly take corrective measures, which would prolong production inefficiencies and lower customer satisfaction.

Using the automatic product defect detection system, manufacturers may lower costs, increase productivity, and guarantee that only high-quality items are supplied to customers by automating fault detection.

This project is motivated by our collaboration with Mukund Trans-Gears, an esteemed manufacturing industry in Kupwad MIDC, Sangli. Mukund, Trans-Gears is a leading manufacturer, exporter, and supplier of various industrial equipment, including helical gears, planetary gearboxes, worm gearboxes, industrial sprockets, and a range of gears and spare parts.

Our proposed system addresses a significant problem faced by mechanical manufacturing industries, like Mukund Trans Gears and many more. By developing a system that utilizes image analysis techniques to classify defective and non-defective pieces, we aim to provide a reliable and efficient solution to enhance the quality control process.

3. Literature Review

Senthikumar, M., Palanisamy, V., & Jaya, J. (2014) [1] used the iterative thresholding technique for detection of defects on metal surface. The authors used the color metal surface as the input image of the product. They did the binarization of the colored image using the iterative thresholding method. The binarization differentiates the detected region of the surface from the non-defected part.

Liu, J., et al. (2017) [2] achieved 96% accuracy for classification and identification of defects. The authors use a BP neural network to analyze the four different types of stainless-steel resistance spot welding specimens, including failed weld, stick weld, faulty weld with gas pore, and good weld.

Wang, T., et al. (2018) [3] used the Convolutional Neural Network (CNN) for defect detection on the DAGM datasets which consists of images of 6 classes, and each class has 1000, 150 non-defective, and 150 defective images respectively. The author achieved the accuracy of 99.8% which is quite higher than the previous outperforming model, i.e., 12-class CNN network. The proposed CNN architecture consists of two parts: global frame classification, which classifies images into the correct classes and sub-frame detection, which decides whether a sample is defective.

Lin, Z., et al. (2019) [4] combined the ResNet network with the HOG-PCA algorithm and detected the defects of the workpiece based on the Faster RCNN algorithm. The authors used the dataset of German Pattern Recognition Association (GAPR). The implemented model helps to detect the defects of surface of different workpieces.

Yang, J., et al. (2020) [5], carried out a survey of the research status of product defect-detection technology in industry. Authors also point out the challenges in this. They compared tradition methods and deep learning techniques for defect detection.

The author discovered that the main areas of interest for both academic and commercial research are 3D object detection, high precision, high positioning, fast detection, small targets, complicated backdrops, identification of obscured objects, and object connections.

Dong, X., et al. (2020) [6] proposed a CNNs model for locating and classify the defects in industrial components.

The authors generated a large set of realistic abnormal images with pixel-level annotation data using the image fusion technique. The z-normalization is also applied to get a synthetic image set. These two sets were trained by U-Net models, which are CNN models trained using image segmentation.

Boikov, A., et al. (2021) [7] proposed and implemented the deep learning model for steel defect detection. The authors found that the deep neural network has good results in the classification and segmentation of surface defect detection in steel object images. The Unet and Xception are two models of deep learning that were trained during experimentation. The performance of the model was verified in a real-world environment.

Wen, Q., et al. (2021) [8] applied the deep learning approach to find the defects in the aerial images of insulation. The two models were proposed and implemented by the authors, i.e., Exact R-CNN and CME-CNN. The result is compared with five different modules, i.e., HOG+SVM, Haar+AdaBoost, Faster R-CNN, YoloV3, and YoloV4. The results found better than the compared ones.

Zhao, W., et al. (2021) [9]. The deformable network is combined with multiscale feature fusion to propose the new defect detection algorithm, which is tested over the NEU-DET dataset. The processed images are affected by uncertainty, so the author suggested using soft computing technology to improve the image quality. Also, the detection time and accuracy have scope to improve.

Bhatt, P. M. et al. (2021) [10] performed the image-based surface defect detection using deep learning. The authors found that image-based surface detection using deep learning is an emerging field widely used in industry automation. This paper is a survey of three different techniques based on defect detection context, learning techniques, and defect localization and classification methods, respectively.

Schmedemann, O., et al. (2022) [11] used the deep learning approach for the detection of defects on PCBs. The authors used a multi-scale feature pyramid network to enhance tiny defect detection through context information inclusion and refined the complete intersection over union loss function to precisely encapsulate tiny defects. The publicly available YOLOv5 deep learning model, by modifying its FPN, is used to achieve higher accuracy.

Lim, J., et al. (2023) [12] performed the AI-based defect detection technique for industrial surface detection. The supervised machine learning algorithm needs a large amount of data for training, which is difficult to gather from industry. In this paper, the author presents the Procedural pipeline for generating the training data based on the physically rendered object under inspection.

The system will involve the integration of a camera-based image capture system, a deep learning-based image classification model, and a database for storing and retrieving the results. The system will be designed to automatically capture images of the manufactured pieces and classify them as either defective or non-defective based on predetermined criteria.

Singh, S. A., & Desai, K. A. (2023) [13], the authors report their work on the use of the least amount of computing power and training datasets for the detection of surface defects using a pre-trained Convolutional Neural Network (CNN), ResNet-101. The feature extracted by ResNet-101are combined with multi-class Support Vector Machine (SVM) as a classifier to detect defective images.

Dwivedi, D., et al. (2024) [14], in this research authors used an attention-based Vision transformer (ViT) model to perform multi-class image classification in order to identify surface flaws from images of solar panels and wind turbine blades. The ViT model has effectively classified the damages in solar panels and wind turbine blades with an accuracy of 98.66% and 97.33% and MCC scores of 0.9829 and 0.9635, respectively.

4. Proposed Work

We provide a novel solution to the problem of identifying faulty parts in the mechanical manufacturing sector. Our approach creates a dependable and effective quality control system by combining the capabilities of computer vision and machine learning approaches.

Using image classification algorithms to automatically identify and classify the collected images as either faulty or non-defective forms the basis of our methodology. Convolutional neural networks (CNNs), one of the most advanced deep learning models, allow us to successfully extract complex characteristics and patterns from the images, enabling precise classification.

Our approach's novel features include creating a personalized dataset, applying cutting-edge deep learning models, and integrating it seamlessly with the infrastructure already in place in the business.

We are making sure that our system delivers high accuracy and reliability in identifying defective items through rigorous evaluation and performance metrics, allowing the industry to improve product quality and reduce waste.

The component that we have selected for this project is a circular hollow disc, which is a part of an automobile.

The disc has a plain surface, which can have some kinds of defects, like:

- Dents
- Scratches
- Rust
- Extra burs

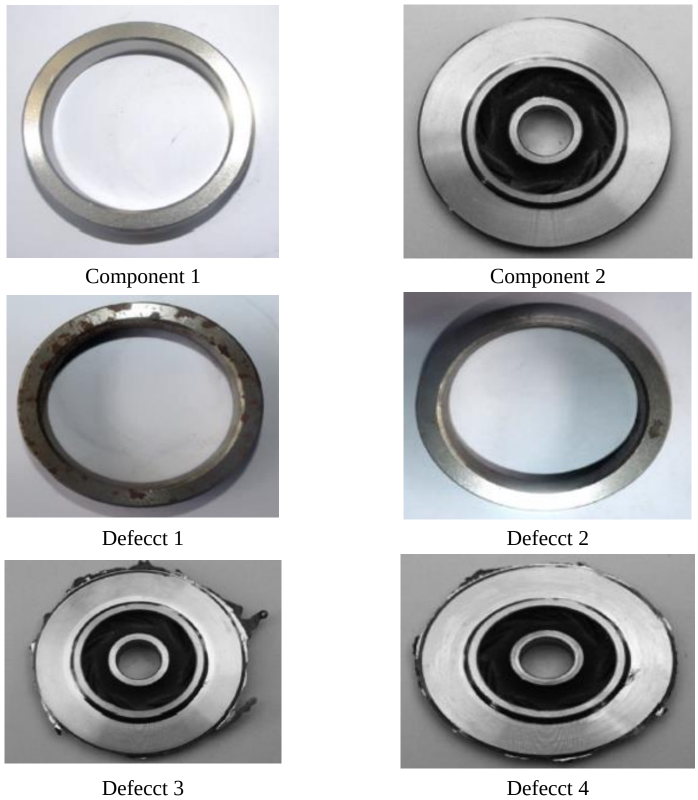

Fig.1. Sample of defective and non-defective piece

4.1. Architecture of Proposed Work

Our main aim was to design and develop the deep learning algorithm, i.e., CNN, for training. We also implemented a mobile application in which the trained deep learning model is integrated.

Fig. 2 shows the CNN architecture of the proposed system.

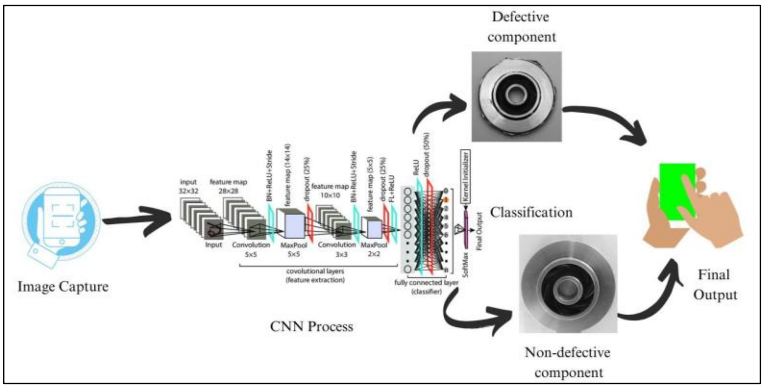

Fig.2. CNN Architecture of the proposed system

In the proposed system, we have used the deep learning model, i.e., the convolutional neural network (CNN), for classifying defective and non-defective products. The image of the manufactured compound is captured using the mobile. That image is classified into defective or non-defective classes. The captured image is tested by a trained deep learning model.

In a product defect detection system, a CNN model can be employed to automatically identify defects in images of product.

The fig. 3 shows YOLO (You Only Look Once) v8 architecture is designed for object detection tasks, particularly in real-time scenarios. We proposed the YOLO v8 (You Only Look Once) model for multiclass classification.

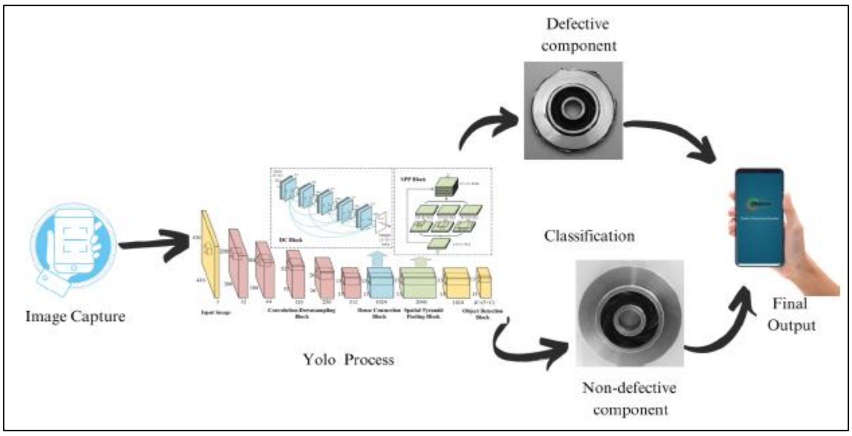

Fig.3. YOLOv8 System Architecture of the proposed system

4.2. Objective of the Proposed System

Following are the objectives of the proposed system:

- To generate the dataset to identify defect trends and patterns and take proactive steps to prevent future defects.
- To design and develop the system for defect identification using deep learning.
- Train and test the deep learning model for defect identification on an available dataset.
- Development of a mobile app for defect detection with the integration of a deep learning model
- Deploy the system to improve the overall efficiency of the manufacturing process by automating the fault-finding process for safety measurements at a low cost.

4.3. Work Flow of System

The fig. 4 shows working of the proposed system for CNN,

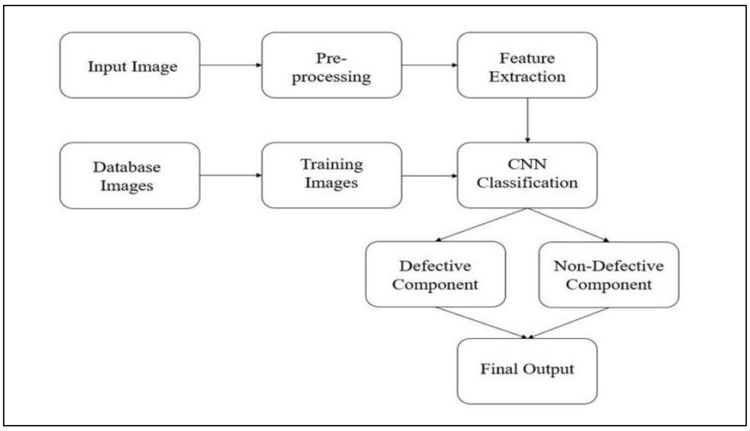

Fig.4. Work Flow of proposed system for binary classification using the CNN model

The dataset of different components is created, and the CNN model is trained over the created dataset. The input image captured using a mobile application is tested using a trained CNN model to find whether the input image is defective or non-defective.

The fig. 5 shows working of the proposed system for YOLO,

4.4. Modules in Proposed System

The proposed system consists of following modules

- Image capture and Image preprocessing
- Defect classification
- Defect Analysis
- Notification
- Report generation

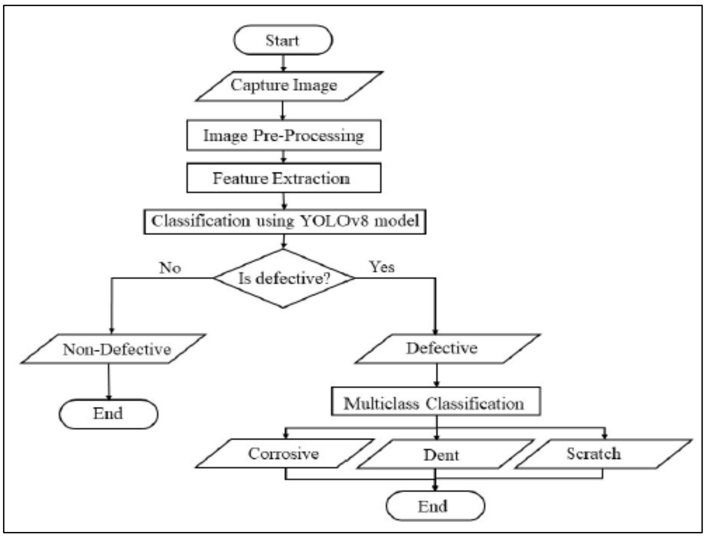

Fig.5. Work flow of proposed system for multiclass classification using the YOLO model

4.5. Hardware and Software Requirement

Table 1 and 2 shows the hardware and software requirements respectively.

Table 1. Hardware requirement

Camera	Minimum of 48 megapixels with additional features such as autofocus, adjustable exposure, and image stabilization
Processing Unit	The device's processing unit, typically the CPU and GPU, performs image processing and deep learning computations.
Computer	Intel Core i5 or equivalent processor, 8GB of RAM, and a large storage capacity, such as a 1TB hard drive or a solid-state drive (SSD).
Mobile Device	At least a quad-core processor and 4GB of RAM

Table 2. Software requirement

Operating system	Windows 11
Programming language	Python 3.11.3
AI Framework/libraries	TensorFlow, Keras, PyTorch, CNN
Database	Firebase 9.22.1
Android Studio	Android Studio Flamingo | 2022.2.1

Fig.6. Dataset used

4.6. Dataset Used

For implementation, a dataset comprising 5036 images, with 3328 representing defective pieces and 1708 non-defective pieces, was collected. The collected dataset is divided into training and testing sets, with standardized image dimensions of 640x640 pixels.

For binary classification, images were labelled as defective or non-defective, while for multiclass classification, defective pieces were further categorized into corrosive, dent, and scratch types.

5. Implementation Methodology

- Deep learning Models

To generate precise insights and forecasts, deep learning models are able to identify intricate patterns in images, text, sounds, and other types of data. Leveraging deep learning models to revolutionize product quality control, we employ three prominent models: Convolutional Neural Networks (CNN), AlexNet, and You Only Look Once (YOLO).

CNNs are foundational for image-based tasks, and in our case, three distinct CNN architectures have been proposed for binary defect classification. AlexNet, a renowned deep neural network, is implemented to broaden the spectrum of model comparisons. For the complex task of multiclass classification, we turn to YOLOv8, a state-of-the-art real-time object detection model. These models collectively form the backbone of our defect detection framework, each contributing its unique strengths to address the nuances of our manufacturing quality control challenges.

In our project, Convolutional Neural Networks (CNNs) play a vital role in binary defect classification. CNNs excel in image-based tasks, capturing spatial patterns crucial for defect detection.

Adding diversity, AlexNet contributes to binary defect classification by providing a different perspective on feature learning. As a deep neural network, AlexNet excels at capturing intricate details within images.

For multiclass defect classification, we employ You Only Look Once (YOLOv8), a real-time object detection model. YOLOv8 efficiently identifies multiple defect types within an image, aligning with our goal of comprehensive defect identification. This underscores YOLOv8's applicability in addressing the various challenges of identifying and classifying defects in manufactured products. Together, these models form a robust framework, collectively advancing the automation of defect detection in the manufacturing industry.

The implementation is divided into two sections,

i) Binary Classification
ii) Multiclass Classification

5.1. Binary Classification

In binary classification, the input image of the product is classified into two classes, i.e., defective or non-defective. For binary classification, we have proposed and implemented three different architectures of the CNN and AlexNet models.

The details of the architecture are given below.

- CNN Architecture 1

In architecture 1, the CNN model features three convolutional layers (32, 64, and 128 filters) with 'valid' padding and ReLU activation, followed by three max-pooling layers and a flattening layer. Three fully connected layers (of 128, 64, and 1 neuron) follow the flattening layer. This model is trained over 10 epochs. The model achieves a notable accuracy of 63.99%, highlighting its effectiveness in image classification tasks.

- CNN Architecture 2

In order to improve the accuracy of above model, we added batch normalization layers between each MaxPooling and Convolutional layers. Stable convergence is encouraged throughout training with this addition. In order to reduce overfitting, dropout layers with a 0.1 rate were strategically positioned after the first two fully connected layers. These improvements, which we describe in our study, resulted in a significant increase in accuracy, reaching an astounding 93.15% accuracy after 10 epochs. This improved model is a great example of how batch normalization and dropout layers strengthen the convolutional neural network and make a significant advancement in the field of image classification techniques.

- CNN architecture 3

We improved our convolutional neural network (CNN) with four ascending convolutional layers (32, 64, 128, and 256 filters) and ReLU activation in an effort to achieve higher accuracy. The second stride of complementary MaxPooling layers maximizes spatial downsampling. In order to provide stability, the model incorporates batch normalisation between the convolutional and max-pooling layers. Additionally, dropout layers (0.2) are used to reduce overfitting between the final two dense layers. The architecture demonstrated the effectiveness of these improvements by achieving an exceptional 97.49% accuracy after 15 epochs of training. Our research outlines a sophisticated CNN design that achieves a significant improvement in image classification accuracy through well-planned architectural complexity.

- AlexNet

We experimented with an innovative binary classification approach, building upon the foundational structure of AlexNet. Retaining the essential layers from the original model, we have customized the final layers to align with binary classification objectives. The modified architecture comprises convolutional and pooling layers, a flatten layer, and adapted fully connected layers. Stochastic Gradient Descent (SGD) optimizes training, employing a mini-batch size of 32 over 20 epochs. L2 regularization prevents overfitting, and GPU acceleration accelerates the process. This exploration represents a unique avenue for binary classification, offering an alternative perspective within the realm of deep learning methodologies.

The Model summary of CNN architecture is shown on following table,

Table 3. Architecture details of CNN

Architecture 1	**Architecture 2**	**Architecture 3**
Input: 640×640×1	**Input: 640×640×1**	**Input: 640×640×1**
Conv2D: 2×2 size, 32 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 64 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 128 filters MaxPooling2D: 2×2 size Flatten: 1D vector Dense: 128 neurons Dense: 64 neurons Dense: 1 neuron	Conv2D: 2×2 size, 32 filters BatchNormalization: 32 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 64 filters BatchNormalisation: 64 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 128 filters BatchNormalisation: 128 filters MaxPooling2D: 2×2 size Flatten: 1D vector Dense: 128 neurons Droupout: 0.1 Dense: 64 neurons Droupout: 0.1 Dense: 1 neuron	Conv2D: 2×2 size, 32 filters BatchNormalization: 32 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 64 filters BatchNormalisation: 64 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 128 filters BatchNormalisation: 128 filters MaxPooling2D: 2×2 size Conv2D: 2×2 size, 256 filters BatchNormalisation: 256 filters MaxPooling2D: 2×2 size Flatten: 1D vector Dense: 128 neurons Droupout: 0.2 Dense: 64 neurons Droupout: 0.2 Dense: 1 neuron
No. of Epochs: 10 Accuracy: 63.99%	No. of Epochs: 10 Accuracy: 93.15%	No. of Epochs: 15 Accuracy: 97.49%

5.2. Multiclass Classification

In this, the input image of the project is classified into different defective classes, like dents, scratches, rust, extra burs etc., or non-defective classes.

The YOLO v8 (You Only Look Once) model for multiclass classification.

- YOLO v8 (You Only Look Once)

For reliable feature extraction, the YOLOv8 architecture uses a sequence of convolutional layers with skip connections and a spatial pyramid pooling module. It is effective for real-time object recognition jobs because it predicts both class probabilities and object bounding boxes simultaneously with a single forward pass.

The YOLOv8 architecture was tailored for multiclass defect detection, specifically classifying defective pieces into three categories: corrosive, dent, and scratch. The implementation involved configuring the network to have three output channels, each corresponding to one of the defect classes. The dataset, comprised of annotated images of defective pieces, was pre-processed, and the model was trained using stochastic gradient descent (SGD) optimization.

Training parameters included a specific batch size, a number of epochs (i.e., 150), and a learning rate of 0.2 tailored for the dataset size and complexity. The resulting YOLOv8 model exhibited a capability for accurate multiclass defect detection, demonstrating its adaptability and effectiveness in industrial quality control applications

5.3. Potential Challenges in Real-world Application

In the real-world application of Product defect detection in manufacturing industry using deep learning, several challenges may arise, particularly in dealing with varying lighting conditions and angles in product images.

To address these challenges, our system incorporates robust preprocessing techniques to normalize and enhance image quality, making the model more resilient to fluctuations in lighting. Additionally, data augmentation methods are employed during the training phase to expose the model to a diverse set of angles, ensuring its ability to generalize effectively across different orientations of the products. Regular updates and fine-tuning of the model based on real-world data are essential to adapt to dynamic manufacturing environments. By continuously optimizing the system's performance through these strategies, we aim to enhance its reliability and effectiveness in detecting defects under a range of challenging conditions in practical manufacturing settings.

6. Result and Discussion

6.1. Binary Classification

A. Result of CNN Architecture 1

During experimentation, the implemented architecture, featuring three convolutional layers followed by max-pooling and fully connected layers, achieved a training accuracy of 79.53% and a validation accuracy of 63.99% after 10 epochs. Fig. 7 shows the result of CNN architecture 1.

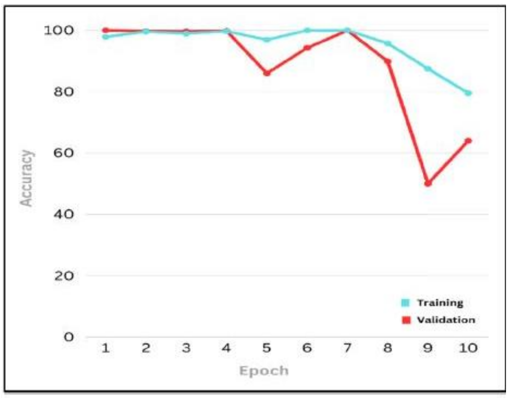

Fig.7. Result of CNN architecture 1

While this accuracy is indicative of effective feature extraction, potential enhancements could involve adjusting the convolutional layer depth or exploring different activation functions. Further investigations into hyperparameter tuning may yield improvements.

B. Result of CNN Architecture 2

In this architecture, we added batch normalization between the convolutional and MaxPooling layers and dropout layers after the first two dense layers, which exhibited a substantial accuracy boost to both training and validation (testing) accuracy of 96.38% and 93.15%, respectively, over 10 epochs. These hyperparameter adjustments significantly improved the model's ability to generalize and prevent overfitting. Batch Normalization enhanced stability during training, while dropout layers added a regularisation effect, collectively contributing to the noteworthy performance increase. The results emphasize the efficacy of these modifications in refining the model for enhanced accuracy and robustness.

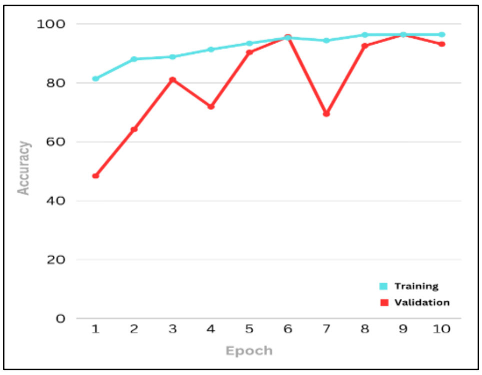

Fig.8. Result of CNN architecture 2

The graph in fig. 8 depicts the training and validation accuracy of our CNN model, showcasing a significant surge in performance. The illustrated trend reveals a notable increase in both training and validation accuracy.

C. Result of Architecture 3

In contrast to previous architectures, the latest model, featuring four convolutional layers, batch normalization, and increased dropout (0.2), showcased remarkable progress with a striking training and testing accuracy of 99.44% and 97.49%**,** respectively, over 15 epochs.

The graph shown in fig. 9 illustrates the training and validation accuracy of the CNN architecture 3.

This heightened accuracy demonstrates the potency of deeper architectures and increased regularization. Furthermore, when practically tested with 1644 product images from the test dataset (948 defective and 696 non-defective), the model accurately classified them into their respective categories. This suggests that the expanded architecture not only excelled in the training phase but also exhibited promising real-world application potential, underscoring its effectiveness for product.

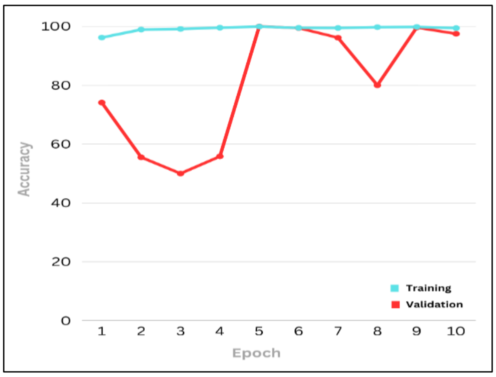

Fig.9. Result of CNN architecture 3

These outstanding accuracy metrics underscore the robust performance achieved by the model.

D. Defect Detection in Binary Classification

Fig. 10 depicts the identification of defects by a trained CNN model. The model performs the classification of the input image and identifies whether the input image of any product is defective or not.

Fig.10. Samples of defect detection of product

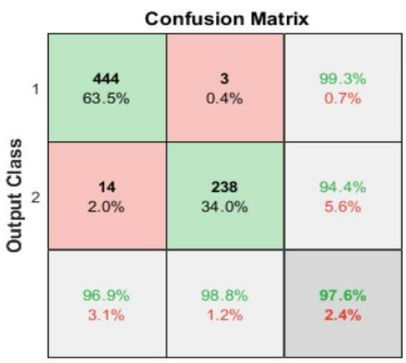

Fig.11. Confusion matrix of AlexNet model

These are the results produced by the CNN model, which shows accurate detection of defects and classification of product images as defective or non-defective

E. Result of AlexNet Model

Upon testing our modified AlexNet model on images of defective and non-defective pieces using MATLAB, it yielded accurate results, achieving an impressive accuracy of 97.6%. This outcome underscores the model's proficiency in correctly classifying defective and non-defective instances, showcasing its reliability.

This confusion matrix shown in fig. 11 indicates that the model is accurate at classifying defective and non-defective pieces. The model correctly classifies 99.3% of the defective pieces and 98.8% of the non-defective pieces.

6.2. Multiclass Classification

YOLO v8

The YOLOv8 model, tested on our product images through the command line interface, demonstrated remarkable accuracy in defect detection. Among the 1644 images from testing dataset, including 948 defective pieces with various types and complexities of defects, the model consistently provided precise classifications. Notably, it effectively handled instances with multiple types of defects on a single piece, as well as those with small, isolated defects. The model presented clear and annotated output images, featuring bounding boxes around detected defects and annotations specifying the defect type. In instances of non-defective pieces, the model accurately returned clear images with no annotations. This robust performance underscores the effectiveness of YOLOv8 in real-world defect detection scenarios, showcasing its versatility and reliability in product quality assessment.

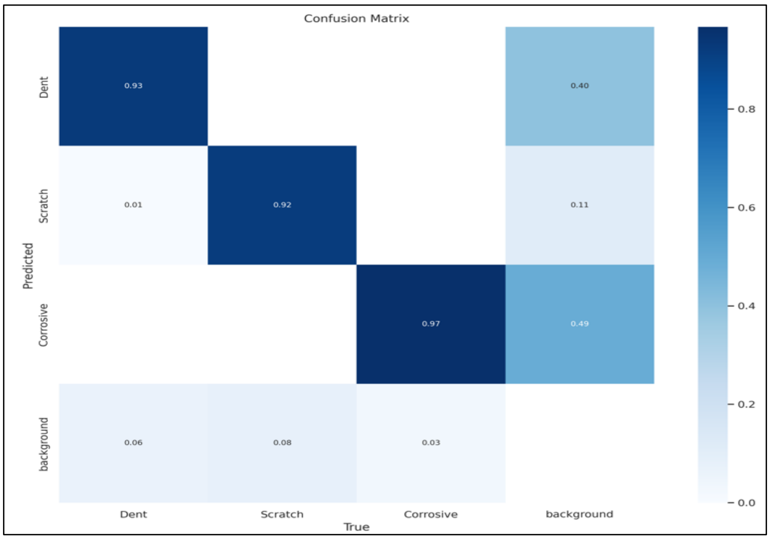

Fig.12. Confusion matrix of Yolov8 model

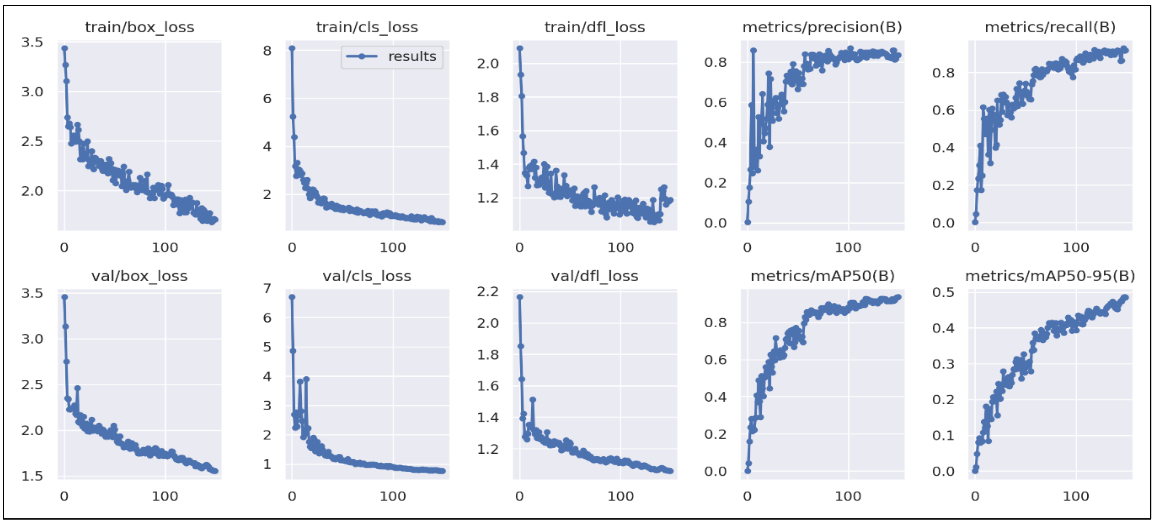

Fig.13. Graphs of Yolov8 model

This confusion matrix in fig. 12 shows the accuracy of the YOLOv8 model. The confusion matrix shows that the model is most accurate at detecting dents on products (0.93), followed by scratches (0.97) and corrosive products (0.97).

This set of graphs showing the training and validation loss and metrics for a YOLOv8 model. The graphs show that the model's performance improves over time, with the training loss decreasing and the validation metrics increasing.

Fig.14. Defect detection and classification by YOLO v8 model

The fig. 14 shows the results produced by the Yolo v8 model which reflect the accurate detection and classification of defects on an image of product.

6.3. Android Application

In the field of defect detection within manufacturing processes, we have developed a user-friendly Android application tailored for the use of operators. This application facilitates seamless interaction between operators and the system, offering a comprehensive suite of features encompassing login, registration, and authentication.

Within the application's framework, operators initiate the defect detection process by activating the device's camera upon selecting the "Start Detection Process" option. This action initiates the capture of the product image, which is subsequently relayed to a YOLO (You Only Look Once) model hosted in the cloud. The YOLO model processes the received image and transmits the results back to the Android application. Finally, operators can access and review the output within the Android application, thereby enhancing the defect detection process in manufacturing operations.

Android App Snaps

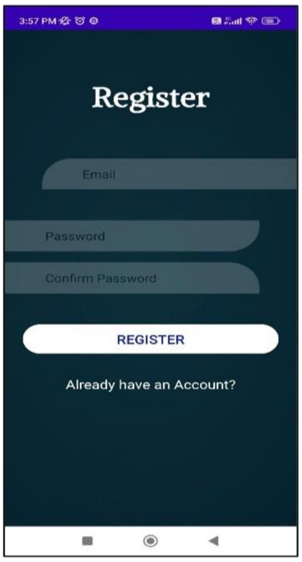

Fig.15. Registration page

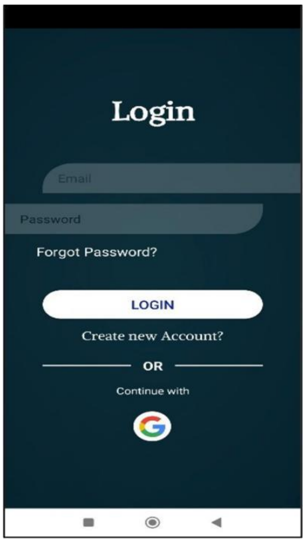

Fig.16. Login page

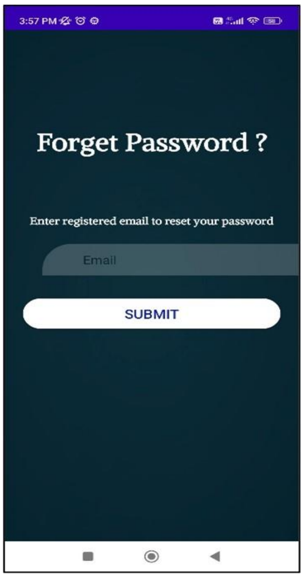

Fig.17. Reset password

Fig.18. Home page

Fig.19. Start detection

Fig.20. View result

The findings of our project have significant implications for industrial settings. By automating defect detection using deep learning techniques, manufacturing industries can greatly enhance their quality control processes. The ability to accurately identify defects in real-time allows for immediate corrective action, reducing the likelihood of faulty products reaching customers and minimizing potential damages to brand reputation.

Additionally, the scalability of our solution is notable, as it can be applied across various manufacturing sectors and adapted to different types of products and defects. However, potential scalability issues such as the computational resources required for training and deploying deep learning models, as well as the need for ongoing maintenance and updates to accommodate changes in manufacturing processes, should be carefully considered. Addressing these challenges through optimization, real-time enhancements, robustness improvements, edge computing, and continuous monitoring ensures sustained effectiveness and efficiency in industrial applications. Overall, our project offers a promising solution to improve efficiency and product quality in industrial settings while addressing the challenges of scalability to ensure long-term viability and effectiveness.

7. Conclusion and Future Work

The proposed approach makes it possible to detect defects in manufacturing companies and also helps to classify the objects as per their defect class. In this paper, we discussed the implementation of deep learning models like CNN, AlexNet, and YOLOv8, which help improve the accuracy of defect detection. The three different architectures of CNN, along with AlexNet, are used for binary classification. The YOLOv8 deep learning model is used for multiclass classification.

Our exploration into binary classification encompassed three distinct CNN architectures and an innovative AlexNet model. Architecture 1 demonstrated proficient feature extraction, achieving a noteworthy accuracy of 63.99%. Architecture 2, enhanced with batch normalisation and dropout layers, notably improved accuracy to 93.15%, underscoring the impact of these modifications. Architecture 3, characterised by increased layer depth and regularisation, showcased exceptional accuracy at 97.49%. The AlexNet model, customised for binary classification, achieved an impressive accuracy of 97.6%, affirming its efficacy.

In the domain of multiclass classification, our implementation of the YOLOv8 model for defect detection exhibited robust performance. With a testing dataset comprising 948 defective pieces, YOLOv8 accurately classified various complexities of defects, showcasing versatility in real-world scenarios. The model adeptly handled instances with multiple types of defects on a single piece and demonstrated precision even with small, isolated defects. Notably, in non-defective pieces, YOLOv8 returned clear images with no annotations, exemplifying its discriminative power.

Additionally, it is noteworthy that the YOLOv8 model achieved an exceptional accuracy of 98.7%, surpassing the performance of all other models attempted. This outstanding accuracy underscores the model's efficiency in defect detection and positions it as a top-performing solution in our evaluation.

Furthermore, our Android application we integrated with YOLOv8 for real-time defect detection, offers a user-friendly interface for manufacturing operators. This cloud-based solution seamlessly captures and processes product images, providing timely and accurate defect classifications. Overall, our comprehensive exploration into binary and multiclass defect detection, coupled with the practical implementation in an Android application, signifies a substantial contribution to advancing quality control practices in manufacturing environments.

The proposed deep learning model Android application can potentially be used in a real-world manufacturing factory. The more types of defects on different materials as be tested with proposed deep learning model. For this we need to prepare the dataset of defective and non-defective materials of specified type.

Data Availability

The image data set which is used to implement this project can be made available from the corresponding author upon request.

Acknowledgments

This project is sponsored by Mukund Trans Gears. We sincerely thank Mukund Trans Gears, Miraj (MH) for their steadfast assistance during the project's execution. Their ongoing support, financial aid, and insightful advice have been crucial to the accomplishment of our work.

References

[1] Senthikumar, M., V. Palanisamy, and J. Jaya. "Metal surface defect detection using iterative thresholding technique." In *Second International Conference on Current Trends In Engineering and Technology-ICCTET 2014*, pp. 561-564. IEEE, 2014.

[2] Liu, Jing, Guocheng Xu, Lei Ren, Zhihui Qian, and Luquan Ren. "Defect intelligent identification in resistance spot welding ultrasonic detection based on wavelet packet and neural network." *The International Journal of Advanced Manufacturing Technology* 90 (2017): 2581-2588.

[3] Wang, Tian, Yang Chen, Meina Qiao, and Hichem Snoussi. "A fast and robust convolutional neural network-based defect detection model in product quality control." *The International Journal of Advanced Manufacturing Technology* 94 (2018): 3465-3471.

[4] Lin, Zhongkang, Zhiqiang Guo, and Jie Yang. "Research on texture defect detection based on faster-RCNN and feature fusion." In *Proceedings of the 2019 11th International Conference on Machine Learning and Computing*, pp. 429-433. 2019.

[5] Yang, Jing, Shaobo Li, Zheng Wang, Hao Dong, Jun Wang, and Shihao Tang. "Using deep learning to detect defects in manufacturing: a comprehensive survey and current challenges." *Materials* 13, no. 24 (2020): 5755.

[6] Dong, Xinghui, Christopher J. Taylor, and Tim F. Cootes. "Defect detection and classification by training a generic convolutional neural network encoder." *IEEE Transactions on Signal Processing* 68 (2020): 6055-6069.

[7] Boikov, Aleksei, Vladimir Payor, Roman Savelev, and Alexandr Kolesnikov. "Synthetic data generation for steel defect detection and classification using deep learning." *Symmetry* 13, no. 7 (2021): 1176.

[8] Wen, Qiaodi, Ziqi Luo, Ruitao Chen, Yifan Yang, and Guofa Li. "Deep learning approaches on defect detection in high resolution aerial images of insulators." *Sensors* 21, no. 4 (2021): 1033.

[9] Zhao, Weidong, Feng Chen, Hancheng Huang, Dan Li, and Wei Cheng. "A new steel defect detection algorithm based on deep learning." *Computational Intelligence and Neuroscience* 2021 (2021): 1-13.

[10] Bhatt, Prahar M., Rishi K. Malhan, Pradeep Rajendran, Brual C. Shah, Shantanu Thakar, Yeo Jung Yoon, and Satyandra K. Gupta. "Image-based surface defect detection using deep learning: A review." *Journal of Computing and Information Science in Engineering* 21, no. 4 (2021): 040801.

[11] Schmedemann, Ole, Melvin Baaß, Daniel Schoepflin, and Thorsten Schüppstuhl. "Procedural synthetic training data generation for AI-based defect detection in industrial surface inspection." *Procedia CIRP* 107 (2022): 1101-1106.

[12] Lim, JiaYou, JunYi Lim, Vishnu Monn Baskaran, and Xin Wang. "A deep context learning based PCB defect detection model with anomalous trend alarming system." *Results in Engineering* 17 (2023): 100968.

[13] Singh, Swarit Anand, and K. A. Desai. "Automated surface defect detection framework using machine vision and convolutional neural networks." *Journal of Intelligent Manufacturing* 34, no. 4 (2023): 1995-2011.

[14] Dwivedi, Divyanshi, K. Victor Sam Moses Babu, Pradeep Kumar Yemula, Pratyush Chakraborty, and Mayukha Pal. "Identification of surface defects on solar pv panels and wind turbine blades using attention based deep learning model." *Engineering Applications of Artificial Intelligence* 131 (2024): 107836.

Authors' Profiles

Mr. Venkatesh Khemlapure is currently pursuing his B.Tech in Computer Science and Information Technology from the Department of Information Technology at Rajarambapu Institute of Technology, Rajaramnagar, Sakharale, MS, India.

Prof. Ashwini B. Patil is M. Tech. in Computer Science and Technology. Currently, she is working as Assistant Professor in the Department of Information Technology at Rajarambapu Institute of Technology, Rajaramnagar, Sakharale, MS, India. She has 20 years of academic experience. She is having life membership of ISTE. Her area of interest is Networking, Machine Learning, and Programming. She has published total 32 research papers in National / International Journal and Conferences. She has also published book entitled "Commence Web Development with PHP and MySQL", ISBN Number: 978-93-81476-13-0, Aruta Publishers, in academic year 2014-15.

Miss Nikita Chavan is currently pursuing her B.Tech in Computer Science and Information Technology from the Department of Information Technology at Rajarambapu Institute of Technology, Rajaramnagar, Sakharale, MS, India.

Miss Nisha Mali is currently pursuing her B.Tech in Computer Science and Information Technology from the Department of Information Technology at Rajarambapu Institute of Technology, Rajaramnagar, Sakharale, MS, India.

How to cite this paper: Venkatesh Khemlapure, Ashwini Patil, Nikita Chavan, Nisha Mali, "Product Defect Detection Using Deep Learning", International Journal of Intelligent Systems and Applications(IJISA), Vol.16, No.4, pp.39-54, 2024. DOI:10.5815/ijisa.2024.04.03