7th Bangladesh Conference on Industrial Engineering and Operations Management
December 21-22, 2024
Publisher: IEOM Society International, USA DOI: 10.46254/BA07.20240233
Published: December 21, 2024

Improving Die-Casting Part Classification using Transfer Learning with Deep Convolutional Neural Networks

Al Sani
Department of Industrial and Production Engineering
Jashore University of Science & Technology
Jashore, Bangladesh
alsani.ipe18@gmail.com

Khalilur Rahman
Department of Computer Science & Engineering
Sylhet Engineering College
Sylhet, Bangladesh
khalil.cse18@gmail.com

Md Nuruzzaman
Department of Computer Science & Engineering
Hajee Mohammad Danesh Science & Technology University
Dinajpur, Bangladesh
mdnkcse14@gmail.com

Md Mahfuzur Rahman
Department of Industrial and Production Engineering
Jashore University of Science & Technology
Jashore, Bangladesh
mrahman.ipe@just.edu.bd

Abstract

Product quality is a crucial factor in the manufacturing process today, as it determines the company's competitive advantages and the consumer's requirements. The problem arises from traditional techniques and quality control methods becoming less effective in the current environments, hence the increasing demand for advanced technologies. This study aims to design and evaluate the effectiveness of the deep learning (DL) technique in identifying defects in die-casting parts. The dataset has been preprocessed, and data augmentation techniques are then employed to mitigate the risk of model overfitting and improve generalization capability. Five pretrained DL models—AlexNet, DenseNet, EfficientNet, SqueezeNet, and WideResNet—were compared with a proposed custom Convolutional Neural Network (CNN) model. The evaluation was conducted using performance metrics, including precision, recall, accuracy, and F1-score. The proposed custom CNN model achieved the highest accuracy of 98.08%, outperforming all other models, with the next best accuracy being 97.70% from the SqueezeNet model. The results show that the proposed DL-based approach, especially the custom CNN model, can greatly improve the quality control process and lower the manufacturing of faulty goods.

Keywords
Die-casting, Classification, Transfer Learning, Convolutional Neural Network, Deep Learning

1. Introduction
The primary goal of any manufacturing company is to maintain profitability and competitiveness in the global market by eliminating defects and rejections in the casting process (Gupta et al. 2023). Casting is a highly adaptable process employed in various applications, including industrial machinery such as motors, generators, and compressors, as well as domestic items such as stoves, furniture, and kitchen appliances. Die-casting is essential in producing complex metal components, particularly in aerospace, automotive, electronics, and modern manufacturing (Duan et al. 2023).

Defective products can result in weak and unreliable constructions, which pose substantial risks. All casting products require a comprehensive inspection before shipment to satisfy customer expectations and maintain quality standards (Oh et al. 2020). Casting companies face significant risks of revenue loss and potential order cancellations if defective products are delivered. Such issues can damage their reputation and strain customer relationships, leading to long-term negative impacts on business performance. Traditionally, defect detection in die-casting has depended on manual inspection techniques, which are costly, time-consuming, and vulnerable to human mistakes (Yousef & Sata 2023). With the automation and digitalization of industrial processes, there is a growing demand for more efficient and precise defect detection systems. Transfer learning has been demonstrated to be an effective solution for addressing the challenge of limited data by enabling the adaptation of pre-trained models to specific tasks. DL methods can significantly improve the speed and accuracy of defect detection by automating the inspection process, resulting in enhanced product quality and reduced production costs. This research aims to improve the detection of defects in die-casting, thereby facilitating the optimization of manufacturing processes, the reduction of waste, and ensuring higher output quality.

1.1 Objectives
This research focuses on designing and developing a deep learning-based custom CNN model for defect classification in die-casting products. Then, we analyze the model's performance using hyperparameter optimization techniques. Subsequently, we evaluate multiple state-of-the-art transfer learning techniques using the Die-casting dataset and compare their performance with our proposed custom CNN model. Consequently, it seeks to contribute to advancements in die-casting product inspection technologies by enhancing the efficiency and accuracy of defect detection processes, ultimately improving overall quality assurance in the die-casting industry.

2. Literature Review
Researchers are advancing die-casting part inspection by integrating artificial intelligence to address critical challenges. They aim to develop real-time autonomous defect recognition systems that ensure high production efficiency and maintain quality standards. Casting products have been identified and classified using various methods.

Wang & Jing (2024) introduced a deep-learning method to detect defects in cast iron parts, overcoming the limitations of traditional polishing techniques. By creating and augmenting a specialized dataset, the model incorporated a coordinate attention mechanism and a bidirectional weighted feature pyramid network (BiFPN) to improve accuracy. Dong et al. (2020) proposed a framework for automated industrial inspection using small datasets, leveraging a pre-trained CNN encoder for enhanced initialization and fine-tuning. Tested on aerospace weld X-rays and public datasets, it proved effective in defect detection and classification. Pranav et al. (2023) highlight the challenge of limited data for training models on defective products and evaluate algorithm performance with reduced training sample sizes. EfficientNetB7, due to its compound scaling, achieves higher accuracy in classifying casting products, even with small training sets. Lal et al. (2023) tackled surface defect detection in steel manufacturing by developing a lightweight neural network that outperforms pre-trained models in accuracy and inference time. Using depth-wise separable convolutions, global average pooling, and augmentations, their custom model achieved 81.87% accuracy and 12 ms inference time, surpassing ResNet and vision transformers.

Using the CNN technique, Chigateri et al. (2023) developed a neural network to detect and classify casting defects in submersible pump impeller castings. The model achieves a classification accuracy of 94%. Mery (2021) introduced a training strategy for aluminum casting defect detection using deep object detection methods, employing low defect-free X-ray images with simulated defects. Eight state-of-the-art detection methods were evaluated, with YOLOv5s demonstrating a high-performance average precision of 90% and an F1-score of 91%. Hu & Wang (2022) suggested a proficient CNN model designed to identify casting defects in radiography images. The model was trained only using image-level labels. Their innovative training approach includes a unique object-level attention mechanism, which improves the detection of local contrast defects. Dongling et al. (2022) presented an adaptive update template defect improvement approach utilizing a Gaussian model to identify surface defects on Si3N4 ceramic bearing rollers. The method successfully improves contrast, removes noise, and accurately locates defects, with an average detection time of 0.84 seconds and a detection accuracy of 96.2%. Habibpour et al. (2021) investigated the ability of four pre-trained CNN models (VGG16, DenseNet121, Inception-ResNetV2, and ResNet50) to extract significant features from casting product photos. The classifier's performance varied depending on the characteristics retrieved by CNN convolutional layers, with VGG16 being recognized as the most effective for identifying defects. Zhao & Wu (2022) devised the RCNN-DC algorithm, employing a recursive attention model for casting defect detection and classification. With a focus on accuracy, especially for products with similar contours, RCNN-DC achieved a remarkable 96.67% classification accuracy, surpassing traditional models and popular networks like AlexNet and ResNet-50. Jiang et al. (2021) developed an improved CNN-based model to classify faulty castings in X-ray images. A new attention-guided data augmentation approach is introduced to increase the size of the training dataset by generating additional pictures. They show that their suggested mutual-channel loss and the data augmentation network (MC-DAN) efficiently create additional training data and capture distinctive minor characteristics.

3. Methods

The methodology begins with data collection, data preprocessing (contrast adjustment, resizing, and noise removal), and data augmentation techniques to improve the dataset. The dataset is divided into subsets of 10% for testing, 10% for validation, and 80% for training. Several models, including WideResNet (Zagoruyko & Komodakis, 2017), EfficientNet (Tan & Le, 2019), SqueezeNet (Iandola et al., 2016), AlexNet (Krizhevsky et al., 2017), and DenseNet (Huang et al., 2017). After determining which model performs the best, a custom, CNN model is chosen. An ablation study analyzes the impact of factors like the loss function, optimizer, learning rate, and epochs. The final results are evaluated using precision, F1 score, accuracy, and recall and compared with other works. The systematic workflow is demonstrated in Figure 1.

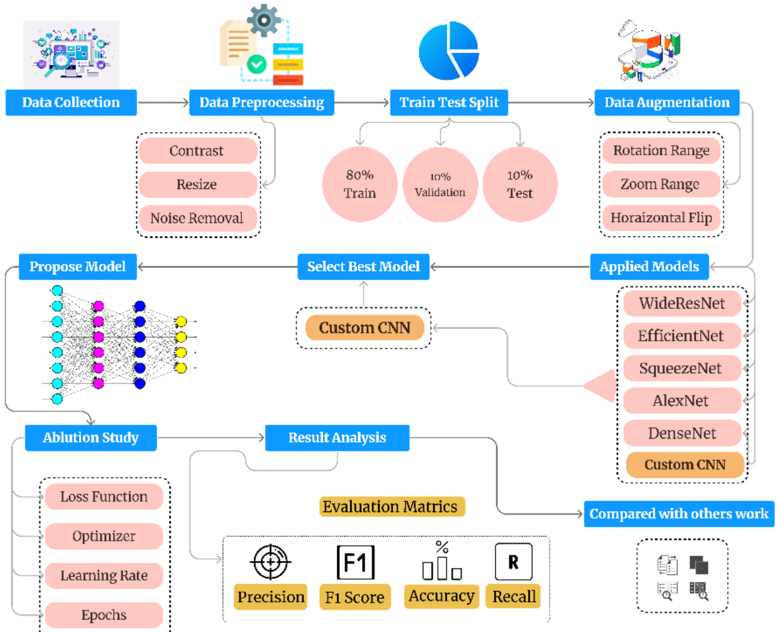

Figure 1. Systematic workflow to analyze datasets, preprocessing, selecting models, and evaluating them to improve die-casting part classification systems.

3.1 Data Collection and Analysis

The dataset used for this research consists of images of die-casting parts collected for quality inspection. The dataset consists of 1300 images collected from Pilot Technocast (2024). It comprises high-resolution JPEG images of cast submersible pump impellers, captured from a top-view perspective using a Canon EOS 1300D digital single-lens reflex camera. The dataset comprises two categories: defective and non-defective casting images, illustrated in Figure 2, each with a pixel resolution of 512×512 and presented in grayscale. Table 1 summarizes the dataset, detailing the total number of images, image dimensions, data type, and number of classes. A thorough dataset analysis was conducted to identify key patterns and features that distinguish defective from non-defective castings, aiming to improve quality inspection algorithms' performance. Table 2 summarizes the dataset distribution after implementing the train-test split. A random seed was employed during the splitting procedure to ensure uniformity and reproducibility of the results.

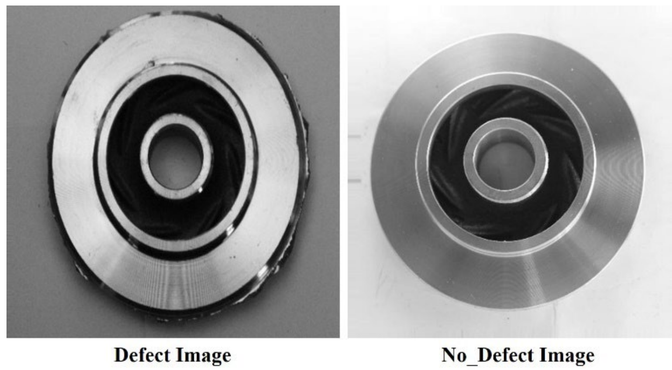

Figure 2. Sample images of casting products from the quality inspection dataset.

Table 1. Dataset Description of collected die-casting images

Name	Description
Total Number of Images	1300
Dimension	512×512
Dataset Type	Imbalance
Total Number of Classes	2
Defect	781
No Defect	519

Table 2. Dataset Description after applying the Train-Test Split method

Name	Description
Train set percentage	80%
Test set percentage	10%
Validation set percentage	10%
Number of Train set Images	1040
Number of Test Set Images	130
Number of Validation set Images	130
Total Number of Images	1300

3.2 Applied Model

Five state-of-the-art DL models were selected for training: AlexNet, EfficientNet, DenseNet, WideResNet, and SqueezeNet. The selection of these models was determined by their architecture's suitability for image classification tasks and demonstrated efficacy in similar applications. Our research concentrates on the classification of die-casting images by applying DL techniques, particularly emphasizing the transfer learning techniques.

3.2.1 AlexNet

In 2012, Krizhevsky et al. (2017) developed AlexNet, which comprises 60 million parameters and half a million neurons. The architecture contains two fully interconnected layers, five convolutional layers, and some of which are followed by max-pooling layers. The architecture ends with a thousand-class SoftMax layer. AlexNet processes an input image with dimensions of 227×227×3, where the spatial dimensions are represented by 227×227, and the "3" denotes the three-color channels such as red, green, and blue. Binary cross-entropy was the loss function that was chosen for the AlexNet model. The learning rate was set at 0.0001, and Adam was selected as the optimizer. The epoch count has been set at 40. Figure 3 illustrates the efficiency of the AlexNet model.

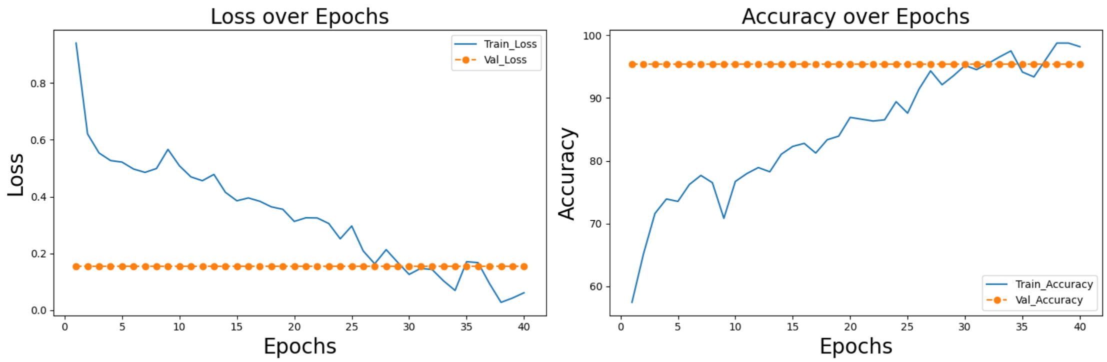

Figure 3. Accuracy and Loss over Epochs for the Alexnet Model during Training and Validation.

3.2.2 EfficientNet

EfficientNet is a CNN architecture designed to optimize computational efficiency and provide cutting-edge performance. It presents a compound scaling technique based on Tan & Le (2019) suggested balancing the depth, breadth, and resolution of the network. This exacting scaling method ensures a harmonic trade-off between precision and model size. This meticulous scaling approach ensures a balanced trade-off between accuracy and model size. The fundamental concept of EfficientNet is to scale the network in a manner that optimizes performance without unnecessarily increasing computational demands, thereby ensuring that it is both resource-efficient and effective. The loss function for the EfficientNet model was binary cross-entropy. Adam elected to serve as the optimizer, and the learning rate was 0.0001. The number of epochs has been set at 40. The model performance of EfficientNet is depicted in Figure 4.

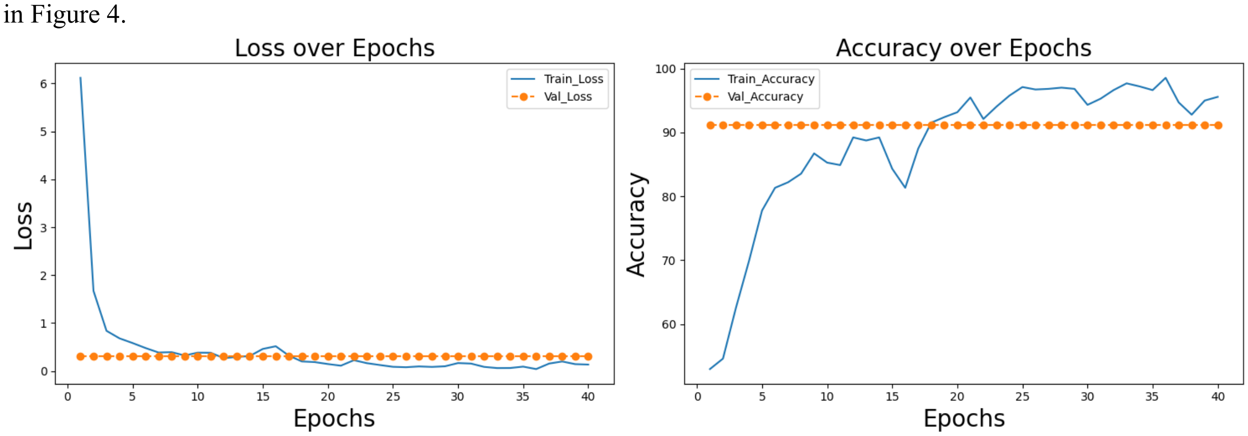

Figure 4. Accuracy and Loss over Epochs for the EfficientNet Model during Training and Validation.

3.2.3 DenseNet

A neural network architecture known for its distinctive dense connectivity pattern is DenseNet (Huang et al., 2017), which is short for Dense Convolutional Network. In contrast to conventional CNNs, which only connect to the subsequent layer, DenseNet provides that each layer receives input from all preceding layers. DenseNet is more efficient in capturing and utilizing information throughout the network due to this dense connectivity, which enhances feature propagation, promotes feature reuse, and minimizes the vanishing gradient issue. The DenseNet model employed binary cross-entropy as the loss function, with a learning rate of 0.0001. Adam was chosen as the optimizer, and the model was trained over 40 epochs. The performance of the DenseNet model is illustrated in Figure 5.

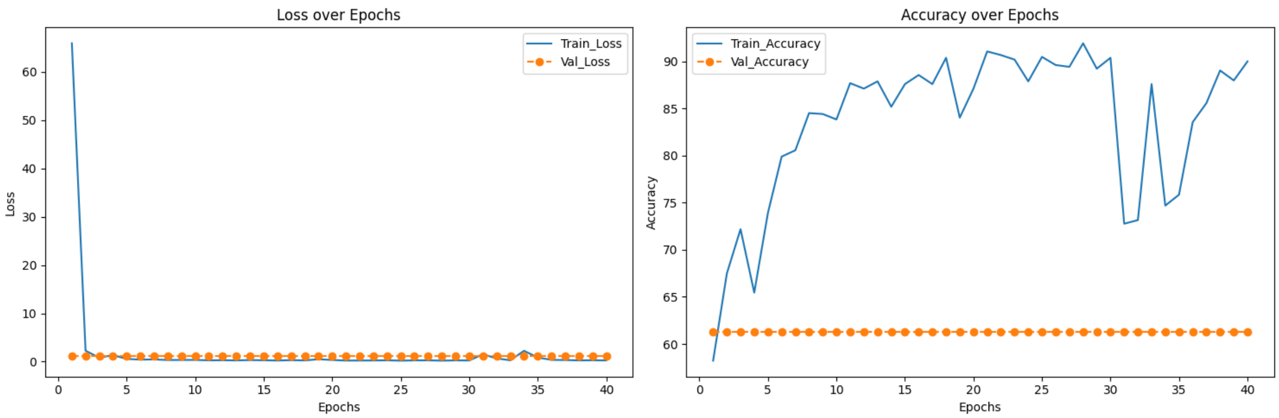

Figure 5. Accuracy and Loss over Epochs for the DenseNet Model during Training and Validation.

3.2.4 WideResNet

The WideResNet architecture, introduced by Zagoruyko & Komodakis (2017), is an improved variant of the ResNet model. It enhances performance by expanding the network's layers while maintaining a relatively shallow depth. WideResNet is highly efficient for a variety of image classification tasks due to its ability to effectively capture a more varied range of complex features through broader convolutional layers. This expanded design enables the network to optimize its efficiency and accuracy by balancing depth and breadth. The WideResNet model was configured with binary cross-entropy as its loss function, ensuring effective handling of the binary classification task. Adam, a widely used optimizer known for its adaptability and efficient gradient computation, was employed with a learning rate of 0.0001 to optimize the model's performance. The training process was carried out over 40 epochs, allowing sufficient iterations for the model to learn and generalize effectively. Figure 6 provides a comprehensive visualization of the model's performance during the training procedure.

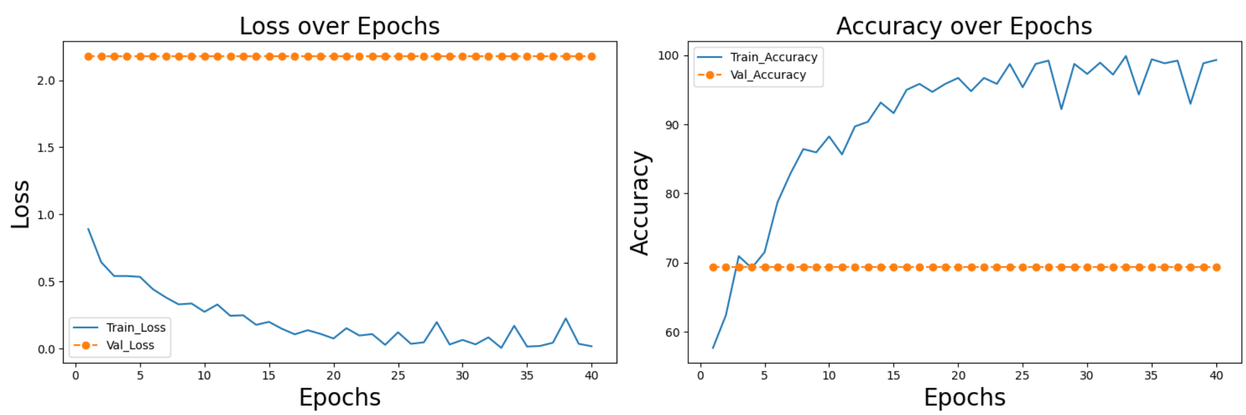

Figure 6. Accuracy and Loss over Epochs for the WideResNet Model during Training and Validation.

3.2.5 SqueezeNet

SqueezeNet is a CNN architecture that is compact and optimized for efficiently operating resource-limited devices, such as mobile phones and embedded systems. In 2016, researchers from UC Berkeley and DeepScale (Iandola et al., 2016) developed SqueezeNet, which employs various innovative design strategies to considerably reduce model size and computational demands while preserving high accuracy, making it ideal for use in low-power environments. The SqueezeNet model utilized binary cross-entropy as the loss function to handle the classification task effectively. Adam was selected as the optimizer, with a learning rate of 0.0001, ensuring efficient optimization. The training was conducted over 40 epochs, allowing the model to learn and refine its predictions. A comprehensive depiction of the model's performance is provided in Figure 7.

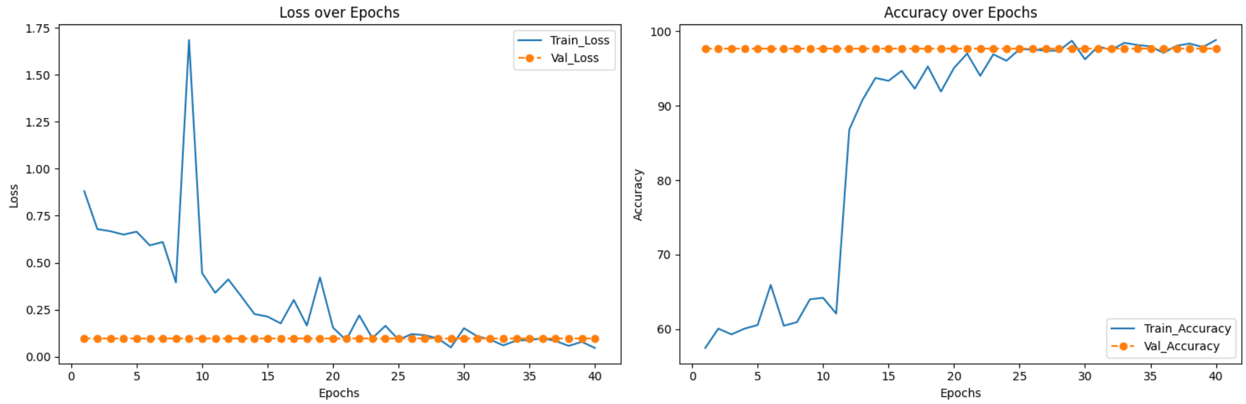

Figure 7. Accuracy and Loss over Epochs for the SqueezeNet Model during Training and Validation.

3.2.6 Custom CNN Model

The architecture of a DL model comprises numerous layers of linked nodes, which are more generally referred to as neurons. The layers are organized in a hierarchical structure, with each layer tasked with processing and altering the data input into the system. CNN converts an input picture into a feature map. This feature map is subjected to several convolutional and pooling layers to provide a predicted output. The three primary components of a typical CNN architecture are the input, hidden, and output layers, as depicted in Figure 8. Passing the input image to the hidden layers, which consist of numerous convolutional and pooling layers, is the function of the input layer. The predicted output layer provides class labels or probability scores for each class.

The proposed custom CNN model is designed for effective feature extraction and classification in die-casting part defect detection. The architecture consists of two main components: convolutional layers for feature extraction and

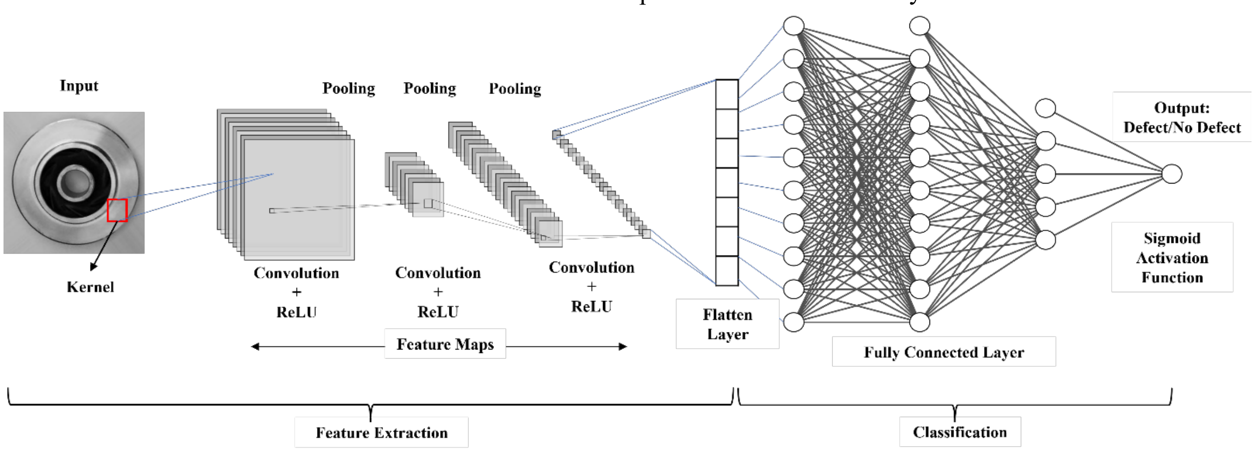

Figure 8. The architecture of the DL model used for improving die-casting part Classification

fully connected layers for classification. The convolutional block includes four layers, each comprising a convolution operation followed by ReLU activation and max-pooling. The model is able to learn hierarchical features as the number of filters increases progressively from 16 to 128, the spatial dimensions are reduced, and translation invariance is introduced through max-pooling with a 2×2 kernel. After the feature maps are flattened, they are passed through a series of fully connected layers. These layers progressively reduce dimensionality from 512 to 256, 128, and finally to the number of output classes, with each layer (except the last) using ReLU activation and dropout for regularization. Adam is employed to optimize the model, which is trained over 40 epochs with a learning rate of 0.0001 and employs binary cross-entropy as the loss function. This design ensures robust feature extraction, efficient optimization, and accurate classification. The performance metrics and results of the proposed model are detailed in subsequent sections. Figure 9 illustrates the performance of the custom CNN model.

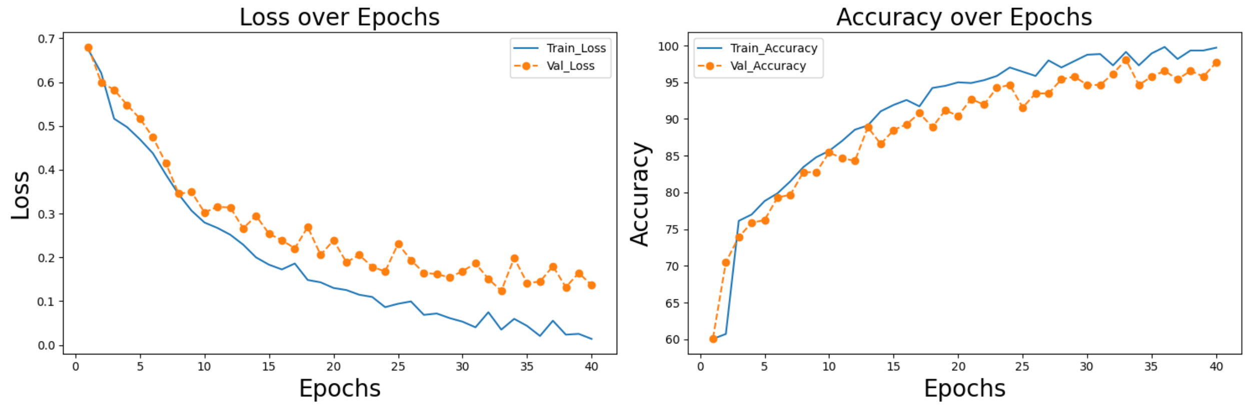

Figure 9. Accuracy and Loss over Epochs for the CNN Model during Training and Validation.

3.3 Evaluation Metrics

This study employs four commonly utilized performance metrics: Accuracy, Precision, Recall, and F1 Score (eq1-eq4).

$$Accuracy = \frac{\sum_{i=1}^{k} correct\ prediction_i}{N} \qquad eq.1$$

$$Precision = \frac{TP}{TP+FP} \qquad eq.2$$

$$Recall = \frac{TP}{TP+FN} \qquad eq.3$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad eq.4$$

Where TP, FP, and FN are True Positives, False Positives, and False Negatives, respectively.

3.4 Model Selection

The performance matrices of the EfficientNet model obtained 91% by precision, recall, and f1 score when the loss matrix was set to the binary cross entropy. SqueezeNet and AlexNet demonstrated exceptional defect classification abilities, with SqueezeNet having 97% precision and recall and 98% F1-score. AlexNet similarly achieved 95% of precision, recall, and f1 score. Compared to the Custom CNN and SqueezeNet, AlexNet, and EfficientNet demonstrated moderate performance, with 95% and 91% accuracy, respectively, indicating their suitability for the classification task. However, they were surpassed by the latter two. Conversely, DenseNet and WideResNet demonstrated substantially inferior performance, with DenseNet attaining an accuracy of only 61% and WideResNet achieving 69%. This implies these models may not be optimally adapted for this dataset or application.

Table 3. Summary of the performance of all models with the loss matrix 'binary cross entropy'.

Model Name	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
AlexNet	95	95	95	95
EfficientNet	91	91	91	91
DenseNet	61	76	61	48
WideResNet	69	80	69	63
SqueezeNet	97.70	97	97	98
Custom CNN (Proposed)	**98.08**	**98**	**98**	**98**

The proposed custom CNN model demonstrated the highest performance in all metrics, with an F1 score of 98%, precision of 98%, recall of 98%, and accuracy of 98.08%. This illustrates its superior capacity to classify defects precisely compared to the pre-trained models. SqueezeNet also demonstrated satisfactory performance, attaining an accuracy rate of 97.70%. However, its overall metrics could not have been better than the proposed models. Table 3 summarizes the classification performance of various models.

3.5 Experimental Setup

The experimental setup for training DL models is crucial as it impacts the performance and the efficiency of the training process. We perform all of our experiments on Kaggle to utilize the GPU-accelerated environment powered by the NVIDIA Tesla P100 GPU, significantly speeding up the training process. The programming was conducted in Python, which was chosen for its extensive support and libraries in the data science domain. To implement the machine learning pipeline, we leveraged Scikit-learn and PyTorch, which offer efficient data mining and analysis tools, including functionalities for preprocessing data, implementing various machine learning algorithms, and evaluating model performance.

5. Experimental Results

5.1 Ablation Study

In this study, both the Adam optimizer and stochastic gradient descent (SGD) were utilized to evaluate the performance of the proposed custom CNN model under various parameter settings. The learning rates were adjusted between 0.001 and 0.0001, while the number of training epochs varied between 20, 30, and 40. For the Adam optimizer, the model achieved its highest accuracy of 98.08% with a learning rate of 0.0001 and 40 epochs, alongside strong performance metrics of 98% for precision, recall, and F1-score, with a minimal loss of 13.69%. Similarly, for the SGD optimizer, the best accuracy was 97% at a learning rate of 0.001 and 40 epochs, coupled with precision, recall, and F1-score values of 97% and a loss of 18%. The performance comparison across these configurations, summarized in Table 4, highlights the model's ability to adapt to optimizers and parameter combinations, showcasing its robustness and reliability.

Table 4. The summarized description of the ablation study of our custom CNN model

Optimizer	Learning rate	epochs	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)	Loss (%)
Adam	0.0001	20	91	91	90	91	20.71%
Adam	0.001	30	96	96	96	96	34
Adam	0.001	40	96	96	96	96	26
Adam	0.001	20	95	94	95	94	16.88
Adam	0.0001	30	95	95	95	95	19
Adam	**0.0001**	**40**	**98.08**	**98**	**98**	**98**	**13.69**
SGD	0.001	20	60	30	50	38	68.89
SGD	0.001	30	97	96	97	97	19
SGD	0.001	40	97	96	97	97	18
SGD	0.0001	30	97	96	97	97	18
SGD	0.0001	20	60.15	96	97	97	69.03
SGD	0.0001	40	97	96	97	97	18

5.2 Graphical Results

The model classified instances of images in the die-casting part classification task, as illustrated in Figure 10. An image in the upper left-hand corner depicts a part that was correctly designated as "No Defect." The remaining images showcase defective parts, all of which were accurately classified as "Defects." This shows the model's exceptional capacity to distinguish between defective and non-defective parts. However, while the confusion matrix indicates a high level of accuracy (with 153 true positives and 103 true negatives), it also demonstrates some misclassification cases. Specifically, four defective parts were misclassified as "No Defect," and one non-defective part was labeled as "Defect." Figure 11 visualizes the ROC-AUC curve for the custom CNN model, which achieved a perfect area under the curve (AUC) of 0.99 for both defect and no-defect classifications. This underscores CNN’s superior ability to distinguish between the two classes. Overall, the results indicate that CNN is the most effective model for defect detection in this study.

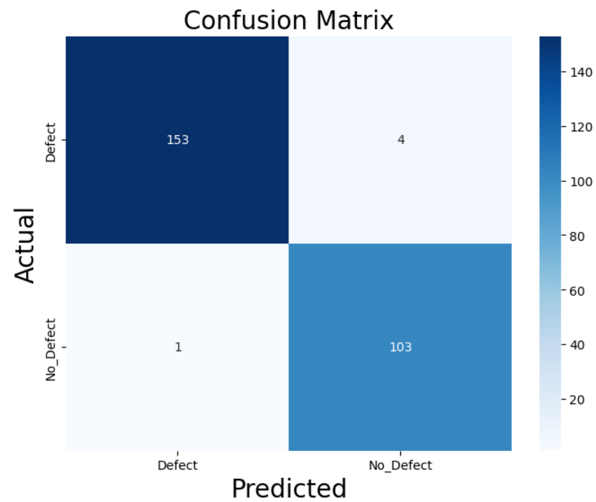

Figure 10. Confusion Matrix for proposed CNN Model

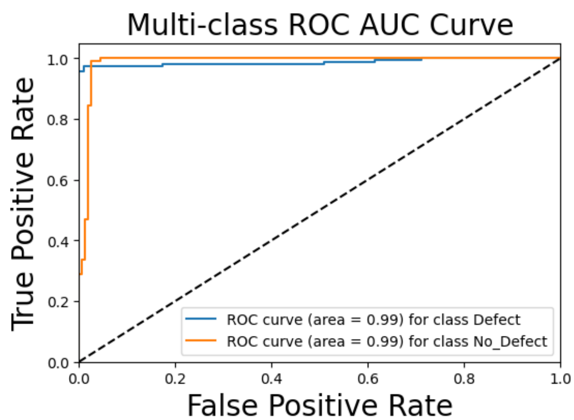

Figure 11. ROC-AUC curve of Custom CNN model, showing a perfect AUC of 0.99 for defect and no-defect classification

5.2 Comparative Performance Analysis with Existing Work

Table 5 compares the proposed CNN model with various kinds of state-of-the-art techniques for defect classification, such as traditional machine learning methods (RBF-SVM, SVM) and deep learning models (VGG16Inception-V3, CNN, YOLOv5m). Conventional approaches, which achieve an accuracy of approximately 97%, and other deep learning models, which exhibit accuracies ranging from 92.77% to 95.9%, are considerably outperformed by the proposed CNN model, which achieves an accuracy of 98.08%. This emphasizes the CNN model's exceptional performance and its potential for real-world defect classification in die-casting products.

Table 5. Comparison of our proposed Custom CNN model with state-of-art techniques.

Reference	Model	Year	Method	Accuracy %
(Ireri et al., 2019)	RBF-SVM	2019	Machine Learning	97.09
(Sahoo et al., 2019)	SVM	2019	Machine Learning	97.5
(Liqun et al., 2020)	VGG16Inception-V3	2020	Deep Learning	95.29
(Hu & Wang, 2020)	CNN	2020	Deep Learning	92.77
(Parlak & Emel, 2023)	YOLOv5m	2023	Deep Learning	95.9
	Our Proposed CNN Model	**2024**	**Deep Learning**	**98.08**

In addition to quantitative metrics, the proposed CNN model's superiority is further supported by a qualitative comparison. For instance, although DenseNet attains moderate accuracy, its visual performance discloses specific constraints. DenseNet frequently experiences false positives or false negatives in extreme cases due to its difficulty in identifying slight defect patterns. In contrast, the proposed CNN model exhibits robustness in the classification of both defective and non-defective parts, as illustrated in Figure 10, where it accurately classifies tiny defects with high precision.

6. Conclusion and Future Recommendation

This research has concentrated on categorizing die-casting components to enhance the quality inspection processes in industrial sectors. The primary objective of this study was to design and develop a deep learning model that can accurately classify defects in casting products. The ultimate goal was to enhance manufacturing processes and reduce the number of defective outputs. We attempted to improve the generalization capability and robustness of the overfitting of the models by diversifying the dataset and enhancing image quality through augmentation techniques. Additionally, the dataset enabled a thorough comparison of the performance of five distinct DL models. Our custom CNN model was the most successful model, demonstrating 98.08% accuracy. This shows the proposed CNN model's ability to classify casting products accurately. The manufacturing industry is significantly impacted by the results of this research, particularly in the context of die-casting part classification. Manufacturers can improve consumer satisfaction and brand reputation by reducing the occurrence of faulty products, enhancing their quality inspection processes, and leveraging DL models. Future research could investigate the inclusion of fusion models, which combine multiple deep-learning architectures to improve defect detection accuracy. Additionally, to test the robustness of these methods under adverse environmental conditions, diverse datasets employing techniques like (Islam et al., 2024) can be involved to validate the performance of these models in real-life scenarios. These models can potentially deliver more reliable and robust classification outcomes in die-casting applications by capitalizing on the strengths of more diverse data.

References

Chigateri, K. B., Poojary, S., & Padmashali, S. , Recognition and classification of casting defects using the CNN algorithm. *Materials Today: Proceedings*, *92*, 121–130,2023. https://doi.org/10.1016/j.matpr.2023.03.818

Dong, X., Taylor, C. J., & Cootes, T. F. ,Defect Detection and Classification by Training a Generic Convolutional Neural Network Encoder. *IEEE Transactions on Signal Processing*, *68*, 6055–6069, 2020. https://doi.org/10.1109/TSP.2020.3031188

Dongling, Y., Xiaohui, Z., Jianzhen, Z., & Nanxing, W. , An enhancement algorithm based on adaptive updating template with Gaussian model for Si3N4 ceramic bearing roller surface defects detection. *Ceramics International*, *48*(5), 6672–6680,2022. https://doi.org/10.1016/j.ceramint.2021.11.217

Duan, Z., Chen, W., Pei, X., Hou, H., & Zhao, Y. (2023). A multimodal data-driven design of low pressure die casting gating system for aluminum alloy cabin. *Journal of Materials Research and Technology*, *27*, 2723–2736. https://doi.org/10.1016/j.jmrt.2023.10.076

Gupta, R., Anand, V., Gupta, S., & Koundal, D. ,Deep learning model for defect analysis in industry using casting images. *Expert Systems with Applications*, *232*, 120758,2023. https://doi.org/10.1016/j.eswa.2023.120758

Habibpour, M., Gharoun, H., Tajally, A., Shamsi, A., Asgharnezhad, H., Khosravi, A., & Nahavandi, S. (2021). *An Uncertainty-Aware Deep Learning Framework for Defect Detection in Casting Products* (No. arXiv:2107.11643). arXiv. https://doi.org/10.48550/arXiv.2107.11643

Hu, C., & Wang, Y. (n.d.). An Efficient CNN Model Based on Object-level Attention Mechanism for Casting Defects Detection on Radiography Images. *IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS*.

Huang, G., Liu, Z., van der Maaten, L., & Weinberger, K. Q. , *Densely Connected Convolutional Networks*. 4700–4708,2017. https://openaccess.thecvf.com/content_cvpr_2017/html/Huang_Densely_Connected_Convolutional_CVPR_2017_paper.html

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., & Keutzer, K. (2016). *SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5MB model size* (No. arXiv:1602.07360). arXiv. https://doi.org/10.48550/arXiv.1602.07360

Ireri, D., Belal, E., Okinda, C., Makange, N., & Ji, C., A computer vision system for defect discrimination and grading in tomatoes using machine learning and image processing. *Artificial Intelligence in Agriculture*, *2*, 28–37,2019. https://doi.org/10.1016/j.aiia.2019.06.001

Islam, M. T., Rahim, N., Anwar, S., Saqib, M., Bakshi, S., & Muhammad, K. ,HazeSpace2M: A Dataset for Haze Aware Single Image Dehazing. *Proceedings of the 32nd ACM International Conference on Multimedia*, 9155–9164,2024. https://doi.org/10.1145/3664647.3681382

Jiang, L., Wang, Y., Tang, Z., Miao, Y., & Chen, S. , Casting defect detection in X-ray images using convolutional neural networks and attention-guided data augmentation. *Measurement*, *170*, 108736,2021. https://doi.org/10.1016/j.measurement.2020.108736

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Commun. ACM*, *60*(6), 84–90. https://doi.org/10.1145/3065386

Lal, R., Bolla, B. K., & Sabeesh, E. (2023). Efficient Neural Net Approaches in Metal Casting Defect Detection. *Procedia Computer Science*, *218*, 1958–1967. https://doi.org/10.1016/j.procs.2023.01.172

Liqun, W., Jiansheng, W., & Dingjin, W. (2020). Research on Vehicle Parts Defect Detection Based on Deep Learning. *Journal of Physics: Conference Series*, *1437*(1), 012004. https://doi.org/10.1088/1742-6596/1437/1/012004

Mery, D. , Aluminum Casting Inspection using Deep Object Detection Methods and Simulated Ellipsoidal Defects. *Machine Vision and Applications*, *32*(3), 72,2021. https://doi.org/10.1007/s00138-021-01195-5

Oh, S., Cha, J., Kim, D., & Jeong, J. (2020). Quality Inspection of Casting Product Using CAE and CNN. *2020 4th International Conference on Imaging, Signal Processing and Communications (ICISPC)*, 34–38. https://doi.org/10.1109/ICISPC51671.2020.00014

Parlak, İ. E., & Emel, E.,Deep learning-based detection of aluminum casting defects and their types. *Engineering Applications of Artificial Intelligence*, *118*, 105636,2023. https://doi.org/10.1016/j.engappai.2022.105636

Pilot Technocast. (n.d.). *Pilot Technocast | Precision Core Casting Solutions*. Retrieved December 1, 2024, from https://pilottechnocast.com/

Pranav, G., Sonam, T., & Sree Sharmila, T. (2023). Defect Detection with less training samples using Deep Neural Networks. *2023 2nd International Conference on Smart Technologies and Systems for Next Generation Computing (ICSTSN)*, 1–4. https://doi.org/10.1109/ICSTSN57873.2023.10151506

Sahoo, S. K., Mahesh Sharma, M., & Choudhury, B. B. (2019). A Dynamic Bottle Inspection Structure. In H. S. Behera, J. Nayak, B. Naik, & A. Abraham (Eds.), *Computational Intelligence in Data Mining* (pp. 873–884). Springer. https://doi.org/10.1007/978-981-10-8055-5_77

Tan, M., & Le, Q. ,EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *Proceedings of the 36th International Conference on Machine Learning*, 6105–6114,2019. https://proceedings.mlr.press/v97/tan19a.html

Wang, P., & Jing, P. ,Deep learning-based methods for detecting defects in cast iron parts and surfaces. *IET Image Processing*, *18*(1), 47–58,2024. https://doi.org/10.1049/ipr2.12932

Yousef, N., & Sata, A. , Innovative Inspection Device for Investment Casting Foundries. *International Journal of Metalcasting*, *17*(4), 2663–2673,2023. https://doi.org/10.1007/s40962-023-01051-4

Zagoruyko, S., & Komodakis, N. (2017). *Wide Residual Networks* (No. arXiv:1605.07146). arXiv. https://doi.org/10.48550/arXiv.1605.07146

Zhao, Z., & Wu, T. ,Casting Defect Detection and Classification of Convolutional Neural Network Based on Recursive Attention Model. *Scientific Programming*, *2022*, 1–11,2022. https://doi.org/10.1155/2022/4385565

Biographies

Al Sani is an Industrial & Production Engineering graduate with a strong foundation in Data Science, Data Analysis, and Optimization, specifically Machine Learning (ML), Deep Learning (DL), Computer-Aided Design, Computer-Aided Manufacturing, and Supply Chain Management.

Khalilur Rahman received a B.Sc. degree in Computer Science and Engineering from Sylhet Engineering College. His current research interests include Machine Learning, Computer Vision, and natural language processing (NLP).

Md Nuruzzaman excels in machine learning, computer vision, NLP, web scraping, automation, and Django with a Computer Science and Engineering background.

Md Mahfuzur Rahman is an Assistant Professor in the Department of Industrial and Production Engineering at Jashore University of Science and Technology (JUST). His current research interests include Machine Learning, Computation of Material, and Nanomaterials.