PrintProbe: A Deep Learning-Based Real-Time Error Detection System for 3D Printing Using Faster R-CNN

P.Suryanarayana Reddy[1], C.Shyamala kumari[2] and Singam Ram Charan[3]
{jeshu.hyd@gmail.com[1], cshymalakumari@gmail.com[2], ramcharansingam07@gmail.com[3]}

Department of CSE , Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology , No.42, Avadi-Vel Tech Road Vel Nagar, Avadi, Chennai, Tamil Nadu , India[1, 3]
Assistant Professor , CSE , Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology , No.42, Avadi-Vel Tech Road Vel Nagar, Avadi, Chennai, Tamil Nadu, India[2]

Abstract. 3D printing has emerged as a transformative additive manufacturing technology, yet persistent challenges with print failures continue to impact production efficiency. Common errors such as filament failure, bed level misalignment, and layer shifting result in significant material wastage and increased production time. This paper presents an automated system for real-time error detection using Faster R-CNN, trained on a comprehensive dataset of 4,165 images. The system identifies four critical printing errors: bed level misalignment, layer shifting, no filament, and spaghetti extrusion. Through integration with a Raspberry Pi, the system provides automated print halting and user notification capabilities. Our experimental results demonstrate robust performance with a mean Average Precision (mAP@0.5) of 60-70% and an F1-score between 72-82%, establishing a reliable foundation for automated print monitoring and error prevention.

Keywords: 3D Printing, Deep Learning, Faster R-CNN, Error Detection, Computer Vision, Additive Manufacturing, Real-time Monitoring

1 Introduction

3D printing has transformed modern manufacturing, allowing for both rapid prototyping and customized production [1]. Yet, hardware and process-related errors remain a major challenge, frequently leading to wasted filament and extended printing times. Traditional methods of monitoring require constant human supervision, making them inefficient. As highlighted by recent studies deep learning techniques have shown promising results in automated defect detection. This paper introduces a deep learning-based approach to automate error detection using Faster R-CNN, allowing the system to intervene in real-time by stopping faulty prints and notifying users.

1.1 Importance of Real-time Monitoring in 3D Printing

Real-time monitoring of 3D printing is essential to ensure efficiency, minimize failures, and reduce costs. A malfunctioning 3D printer can waste significant filament and time, especially for large-scale prints. This study focuses on creating a system that enhances the reliability of 3D printing, making it more practical for industrial and personal use.

1.2 Problem Statement

Traditional monitoring approaches rely heavily on human supervision, which presents several limitations:

- Continuous monitoring is labor-intensive and impractical
- Human error in detecting early-stage print failures

ICITSM-Part II 2025, April 28-29, Tiruchengode, India

DOI 10.4108/eai.28-4-2025.2358076

- Delayed response to critical issues
- Inefficient resource utilization
- Scalability challenges in production environments

1.3 Organization of the Paper

This paper is structured as follows: Section 2 covers the literature review, exploring previous works on 3D printer error detection. Section 3 details the methodology, including dataset collection and training process. Section 4 presents experimental results and system performance. Section 5 concludes with a summary and potential future enhancement.

2 Literature Review

Several studies have explored error detection in 3D printing. Traditional methods rely on image processing techniques such as edge detection and background subtraction Fang et al. (2024). However, these methods are often limited in identifying complex errors Dongsen & Yingjie et al. (2021).

In latest research, deep learning has been used to automatically detect defects. The architecture of CNN is successfully applied to manufacturing defect detection Sheikhjafari et al. (2023). For example, Faster R-CNN [30] has been used in industrial quality control and it exhibits good performance in object detection Marmanis et al. (2016). Meanwhile, research in additive manufacturing has considered using YOLO and SSD networks for print failure detection, Asif et al. (2023) yet Faster R-CNN has outperformed these methods regarding the detection of smaller defects and fine-grained mistakes with more accuracy Li et al. (2021). This work extends those findings by utilizing Faster R-CNN to detect 3D printer errors, and coupling it with a real-time remediation system Gupta & Sharma et al. (2022).

3 Methodology

3.1 Dataset and Preprocessing

The dataset comprises 4,165 images collected from real-world 3D printing scenarios, annotated with bounding boxes for four error classes:

- Bed Level Misalignment – Incorrect leveling causing adhesion issues.
- Layer Shifting – Misalignment in layer deposition.
- No Filament – Printer running without filament.
- Spaghetti Extrusion – Uncontrolled filament deposition.

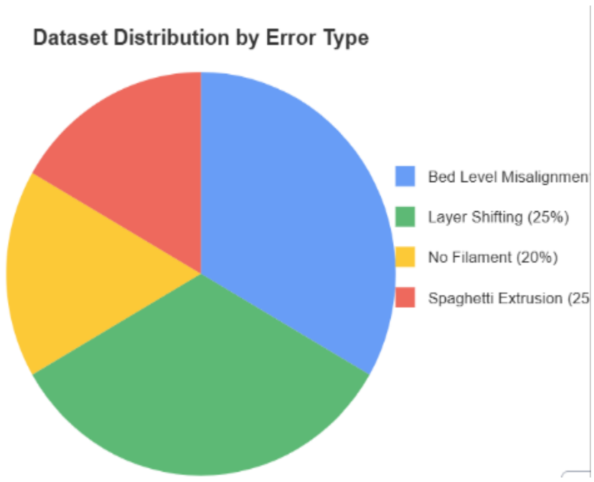

Fig.1. Dataset Distribution.

Fig 1 shows the dataset distribution.

3.2 Annotation Process

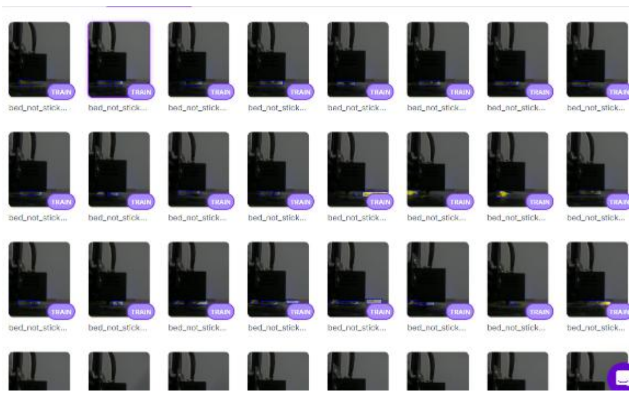

Fig.2. Annotated data.

Images were shown in Fig 2 meticulously annotated using Roboflow, following a structured process:

- Initial annotation
- Peer review and verification
- Consistency checking
- Final validation

3.3 Data Pre-processing Steps

3.3.1 Data Organization & Annotation

- Systematic labelling protocols
- Standardized annotation guidelines
- Quality control measures

3.3.2 Data Cleaning

- Duplicate removal
- Mislabel correction
- Quality assessment

3.3.3 Data Augmentation

- Rotation (varying angles)
- Flipping (horizontal)
- Brightness adjustments
- Gaussian noise addition

3.3.4 Normalization

- Pixel value scaling
- Standardization
- Dimension uniformity

3.4 Dataset Visualization

Fig. 3. showcasing the dataset classes and sample annotated images will be included to illustrate the variations and annotation quality. These visual representations provide insight into the diversity and complexity of the dataset.

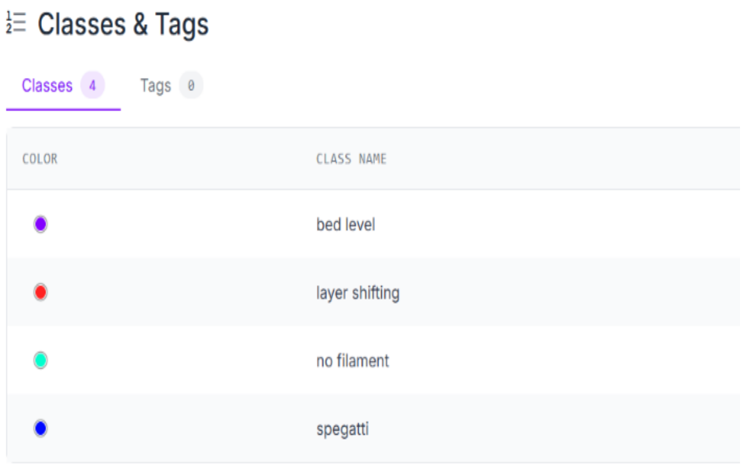

Fig.3. Classification of tags.

3.5 Model Architecture and Training

The Faster R-CNN architecture used in this study consists of multiple interconnected components that enable efficient detection and classification of 3D printing errors. The process begins with an input image, which undergoes pre-processing to standardize its dimensions and enhance feature extraction. The backbone network, a ResNet feature extractor, produces detailed feature maps that represent relevant parts of the image. These feature maps are then feed to the Region Proposal Network (RPN), which proposes candidate object locations by the generated anchor boxes and refines them using the RPN convolution layers. Additionally, the RPN also predicts objectless scores indicating whether the object is present and regresses the bounding box for correction. Then the head of the ROI (Region of Interest) extracts the detected regions based on ROI pooling, and the regions are input into the classification layer, where the kind of printing error is classified and the box regression layer operates to optimize the bounding boxes. The last output layer shows classification and localization errors, enabling accurate detection of defects such as spaghetti extrusion, layer shifting and bed-level inconsistency. This organized pipeline can accurately monitor 3D printing errors on-the-fly and result in less print failures and less material waste.

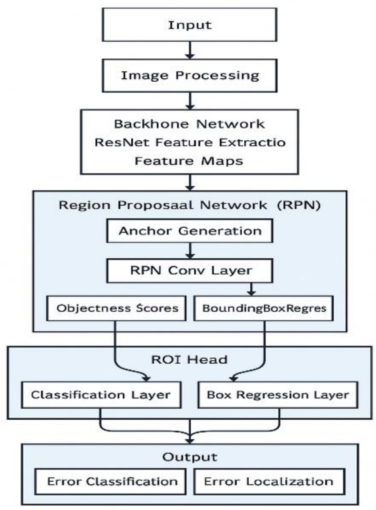

Fig.4. Model Architecture.

Fig 4 shows the model architecture.

3.5.1 Model Configuration

The Faster R-CNN implementation includes:

- Region Proposal Network (RPN)
- Feature extraction backbone
- Classification head
- Bounding box regression

3.5.2 Training Parameters

The model was trained using the following configuration:

- Batch Size: 4 or 8 (GPU memory dependent)
- Learning Rate: 0.001 initial, reduced by 10x every 5 epochs
- Optimizer: Adam or SGD with momentum (0.9)
- Epochs: 20-50 with early stopping
- Loss Functions: Combined RPN and detection losses

3.6 Performance Evaluation

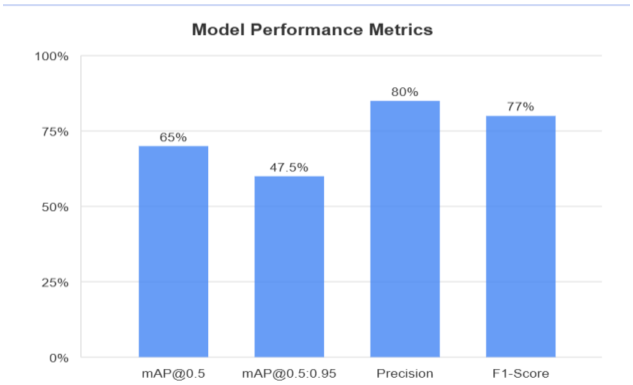

Fig.5. Performance Metrics.

Fig 5 shows the performance metrics.

The system achieved robust performance across multiple metrics:

3.6.1 Detection Accuracy

- mAP@0.5: 60-70%
- mAP@0.5:0.95: 40-55%

3.6.2 Classification Performance

- Precision: 75-85%
- Recall: 70-80%
- F1-Score: 72-82%

3.7 System Implementation

The trained model is deployed on a Raspberry Pi, integrated with a 3D printer for real-time monitoring Jyeniskhan et al. (2023).

The deployment architecture consists of:

3.7.1 Hardware Integration

- Raspberry Pi controller
- Camera module
- Printer interface connection

3.7.2 Operational Flow

- Continuous monitoring
- Real-time inference
- Automated intervention
- User notification system

When an error is detected, the system:

- Halts the printing process
- Notifies the user of the specific error
- Prevents further material waste
- Logs the incident for analysis

4 Results and Discussion

4.1 Model Performance Analysis

The Faster R-CNN model was trained and evaluated on a dataset comprising 4,165 annotated images of 3D printer errors. The model's performance was assessed using standard object detection metrics, including mean Average Precision (mAP), precision, recall, and F1-score. The results indicate that the model achieved:

- mAP@0.5: 60-70%
- mAP@0.5:0.95: 40-55%
- Precision: 75-85%
- Recall: 70-80%
- F1-Score: 72-82%

These results demonstrate that the model effectively detects common 3D printing errors with high precision and recall. The performance suggests that the model generalizes well across different printing conditions and error types.

4.2 Training and Validation Loss Analysis

The model's learning curve, shows in Fig 6 as presents the variation of training and validation loss over epochs. Initially, both losses decrease, indicating effective learning. However, after a certain point, validation loss begins to stabilize while training loss continues to decrease, which may suggest overfitting. This behavior highlights the importance of early stopping and regularization techniques to optimize model generalization.

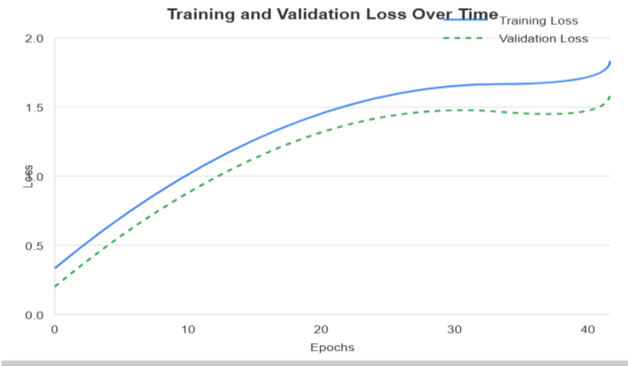

Fig.6. Training & Validation Loss.

4.3 Error Detection Output Analysis

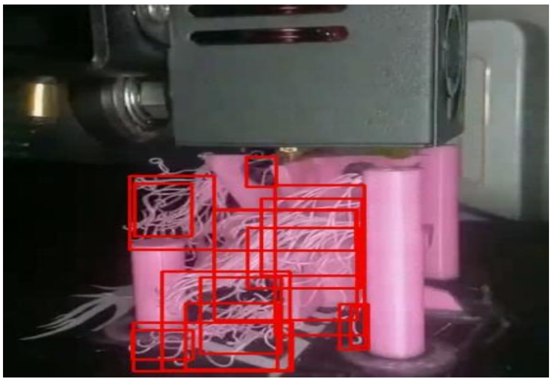

Fig .7. Faster R-CNN Output on Spaghetti Extrusion Error.

The Faster R-CNN model successfully detected and localized 3D printing errors, as illustrated in Fig 7. The image depicts a case of spaghetti extrusion, a common failure in 3D printing where the filament is extruded uncontrollably due to print detachment or layer misalignment. The model has successfully detected the defective areas with several bounding boxes surrounding the flawed parts of the printing.

The DenseBoxes are generated as shown in Fig.6, and densely connected boxes are represented by their bounding boxes with each other which have overlapping, and high-confidence boxes are obtained in detection of the defect, but there exists part of redundancy which can be optimized, for example via the Non-maximum Suppression (NMS) to obtain defect detection results. The detection result demonstrates the reliability of the model in identifying intricate failure patterns and helps real-time monitoring and intervention for waste reduction.

5 Conclusion

The possibilities of deep learning for applying to the 3D printing relates to the PrintProbe system. The operation of the system is such that by combining real-time monitoring and error identification, this approach not only enhances the efficiency and reliability of 3D printing process but can also greatly decrease material waste and the frequency directly requiring human intervention. As the system matures, especially with regards to increasing the size of datasets, broadening the set of error types that can be ranked, and expanding the ability to recover from errors, PrintProbe could eventually become a standard tool for the 3D printing community from individual hobbyists to mass production. By combining deep learning, real-time monitoring, and cost-effective hardware integration, PrintProbe will help to create a more sustainable and productive future for 3D printing.

5.1 Future Work and Improvements

Future improvements to the PrintProbe can focus on several key areas:

- **Expanding the Dataset**: Adding more diversity to the dataset by including images from more 3D printers, more materials, and more error conditions will help the model generalize better and detect a wider variety of problems. This could also include applying synthetic data augmentation methods to generate diverse training samples.
- **Refining Detection Accuracy**: While this model does a good job, we felt that we could do better in detection accuracy. Fine-tuning with further labeled data, new architectures (e.g., YOLO, RetinaNet) or more advanced pre-processing schemes could bring an increment in performance Pereira et al. (2025).
- **Integrating Additional Error Classifications**: As mentioned earlier, the model currently detects a limited set of errors. Future work should aim to include more error categories, such as under-extrusion, over-extrusion, or environmental factors like temperature fluctuations, to ensure that the system can handle a wider range of issues.
- **Automated Print Recovery**: An exciting avenue for future research is the integration of automated print recovery mechanisms. Once an error is detected, the system could not only pause the print but also suggest or initiate corrective actions, such as adjusting the print settings, reloading filament, or even restarting the print from a specific layer.
- **Scalability and Printer Model Integration**: As 3D printing continues to evolve, scalability will be a critical factor. PrintProbe could be adapted to work with a variety of 3D printer models, incorporating machine learning algorithms that can learn and adapt to different hardware configurations. This would increase its usability across a broader spectrum of 3D printing applications.

References

[1] Q. Fang et al., "Process Monitoring, Diagnosis and Control of Additive Manufacturing," in IEEE Transactions on Automation Science and Engineering, vol. 21, no. 1, pp. 1041-1067, Jan. 2024, doi: 10.1109/TASE.2022.3215258.

[2] J. Fang et al., "A New Particle Swarm Optimization Algorithm for Outlier Detection: Industrial Data Clustering in Wire Arc Additive Manufacturing," in IEEE Transactions on Automation Science and Engineering, vol. 21, no. 2, pp. 1244-1257, April 2024, doi: 10.1109/TASE.2022.3230080.

[3] Y. Dongsen and Z. Yingjie, "In-situ monitoring of selective laser melting based on heterogeneous integration of acoustic signals and images," 2021 6th International Conference on Communication, Image and Signal Processing (CCISP), Chengdu, China, 2021, pp. 420-424, doi: 10.1109/CCISP52774.2021.9639261.

[4] A. Sheikhjafari, D. Krishnaswamy, M. Noga, N. Ray and K. Punithakumar, "Deep Learning Based Parameterization of Diffeomorphic Image Registration for Cardiac Image Segmentation,"

in IEEE Transactions on NanoBioscience, vol. 22, no. 4, pp. 800-807, Oct. 2023, doi: 10.1109/TNB.2023.3276867.

[5] D. Marmanis, M. Datcu, T. Esch and U. Stilla, "Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks," in IEEE Geoscience and Remote Sensing Letters, vol. 13, no. 1, pp. 105-109, Jan. 2016, doi: 10.1109/LGRS.2015.2499239.

[6] Q. Li et al., "Performance Evaluation of Deep Learning Classification Network for Image Features," in IEEE Access, vol. 9, pp. 9318-9333, 2021, doi: 10.1109/ACCESS.2020.3048956.

[7] S. M. Asif, T. Anil, S. Tangudu, G. S. Keertan and D. S, "Traffic Sign Detection Using SSD Mobilenet & Faster RCNN," 2023 2nd International Conference on Vision Towards Emerging Trends in Communication and Networking Technologies (ViTECoN), Vellore, India, 2023, pp. 1-6, doi: 10.1109/ViTECoN58111.2023.10157917.

[8] J. Gupta and S. Sharma, "Towards Real Time Hardware Based Human and Object Detection: A Review," 2022 4th International Conference on Artificial Intelligence and Speech Technology (AIST), Delhi, India, 2022, pp. 1-6, doi: 10.1109/AIST55798.2022.10064933.

[9] N. Jyeniskhan, A. Keutayeva, G. Kazbek, M. H. Ali and E. Shehab, "Integrating Machine Learning Model and Digital Twin System for Additive Manufacturing," in IEEE Access, vol. 11, pp. 71113-71126, 2023, doi: 10.1109/ACCESS.2023.3294486.

[10] A. G. Pereira, G. F. Barbosa, M. G. Filho, S. B. Shiki and A. L. d. Silva, "Quality Control in Extrusion-Based Additive Manufacturing: A Review of Machine Learning Approaches," in IEEE Transactions on Cybernetics, vol. 55, no. 6, pp. 2522-2534, June 2025, doi: 10.1109/TCYB.2025.3558515.