IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 6, Issue 1, August 2026

Impact Factor: 8.2

Deep Learning-Based Microplastic Detection Using Raspberry Pi and YOLO26

Pooja Vidyadhar Dangat and Prof V C Maindargi
Department of E&TC
Terna College of Engineering, Dharashiv, India
dangatpooja0@gmail.com

Abstract: *Microplastics are causing significant harm to aquatic organisms and people as they pollute water bodies, which has become a major environmental problem. Existing techniques for monitoring microplastics require specialized lab instruments, time-consuming sample preparation, and professional analysis. However, this article discusses a proposed technique for monitoring microplastics in water bodies that uses the Raspberry Pi 4B, a USB digital microscope, the OpenCV library, and the YOLO26 deep learning framework. Specifically, the system captures close-ups of water samples and then classifies individual particles as microplastics via image preprocessing, feature extraction, and object detection. The system is specifically designed to function on relatively low-end machines without compromising performance or precision. Experimental results demonstrate that the fine-tuned YOLO26s model detects microplastics effectively, achieving 77.8% precision, 63.2% recall, and an mAP@50 of 71.5% after 50 training epochs on Google Colab. Moreover, the system automates microplastic detection and counting processes and provides visualization. Its portability and low cost make it useful for environmental monitoring, educational purposes, and water quality testing. Also, the findings indicate that the method actually works quite effectively in detecting microplastics and can be used to develop future intelligent monitoring systems. To improve this technique, we might consider adding cloud analysis capabilities, integrating them into IoT, and using deep learning models to handle large-scale tasks.*

Keywords: Microplastic Detection, Artificial Intelligence, Deep Learning, YOLO26, Computer Vision, Raspberry Pi, Water Quality Monitoring, Environmental Monitoring, Image Processing, Embedded Systems

I. INTRODUCTION

Plastic use is increasing at an exponential rate in industries and homes, and poses a huge global pollution problem. All types of plastic waste are of concern, but microplastics are of most concern due to their minute size, longevity, and accumulation in aquatic habitats. The term 'microplastics' refers to plastic debris less than 5mm in size that results from the breakdown of large plastic items or is manufactured purposely, such as scrubbers used in cosmetic products. These bits have been detected in all places: rivers, lakes, oceans, groundwater, and even drinking water. This is a major challenge for both aquatic life and humans. Microplastics in water bodies have increased, necessitating effective monitoring mechanisms. Visual Microscopy, FTIR, Raman Spectroscopy, and SEM are techniques used to identify microplastics. However, such techniques require elaborate laboratories, expensive equipment, and extensive sample preparations. These processes are time-consuming and cannot be adopted in situ for real-time measurements of microplastics in the water bodies. There is a significant challenge in meeting the increasing demand for water analysis due to the lack of readily deployable microplastic detection techniques.

Innovations in AI, computer vision, and deep learning have enabled us to automate environmental monitoring. Deep learning object detection algorithms perform remarkably well at detecting visual objects across numerous domains, including medicine, industrial applications, agriculture, and environmental studies. In particular, the YOLO family of algorithms has proven excellent at fast object detection. Moreover, the lightweight nature of YOLO26 makes it ideal for running on devices with low computational capacity, such as the Raspberry Pi. Efforts have been made to detect microplastics using image processing and machine learning techniques. Methods such as CNNs, feature extraction techniques, and traditional image analysis techniques have been considered. They prove efficient but often depend on high computational power and expensive equipment, and cannot function online. Additionally, most existing systems are neither portable nor designed for field-level applications.

This research proposes a real-time system for monitoring water quality that uses AI to detect microplastics. It is a combination of a Raspberry Pi 4B, USB digital microscope, OpenCV, and YOLO26. The system takes microscopic images and then automatically detects and classifies microplastics. The architecture of the proposed system is designed to be portable and inexpensive. Moreover, it is flexible enough for a variety of purposes. Thus, it allows for effective tracking of microplastics. The system design is quite simple and divided into sections such as image acquisition, image preprocessing, artificial intelligence detection, data storage, and result display. First, the image acquisition section acquires microscopic images through a USB digital microscope. Secondly, the image pre-processing stage uses OpenCV to filter images for improved detection. Then comes the YOLO26 model, which uses artificial intelligence to detect microplastic particles.

The proposed system simulates an automated environmental scanning process. The process starts with capturing images of water samples, then filtering, resizing, and enhancing them. Next, these prepared images will be processed by the YOLO26 network, which will detect particles, draw bounding boxes around them, and measure the confidence of each bounding box. Following the above processes, the results regarding the number of microplastics detected will be shown to users.

II. RELATED WORK

Contamination of the environment by microplastics is among the main challenges of the 21st century. These particles are widespread in aquatic environments and may pose risks to human health as well. Researchers from different countries are working to develop approaches for identifying and monitoring these particles. There are indeed precise tests performed in laboratories, but they require considerable time and money. Moreover, one needs specialists to perform them. Fortunately, artificial intelligence, computer vision, and deep learning technologies are quite advanced nowadays. Thus, it is possible to create rapid, precise detectors that also shorten analysis time.

M. Prata et al. [1] have conducted extensive studies on the use of image analysis for microplastic detection. These authors have employed digital microscopy and image analysis tools to detect microplastics in water samples. The authors found that employing such image analysis techniques made it easier to detect microplastics and enabled much faster identification than manual procedures. Moreover, their technique reduced errors caused by visual inspection. Nevertheless, their system is still dependent on laboratory setup, expertise, and manual verification of findings. Thus, it is quite time-consuming. The deep learning architecture proposed by Lorenzo-Navarro et al. [2] was designed to classify microplastics using Convolutional Neural Networks (CNNs). This model was trained using microscopic images of different microplastic categories and then tested under various conditions. Such an approach proved highly efficient, achieving very high accuracy rates that exceeded those of previously used feature-based approaches. The CNN automatically recognized all the complex visual cues without any manual intervention. However, there was one drawback to using such a model. It required substantial computational resources (such as a GPU). That is why it could not be easily implemented in cheap devices.

The authors Park et al. [3] have proposed an automatic microplastics detector system which utilizes image processing and machine learning. Firstly, they conducted microscopic image preprocessing. Afterward, the particles' features, such as shape, color, texture, and geometry, were identified. Lastly, machine learning models were trained on those features to detect microplastics. With the proposed system, the problems of manual inspection and slower detection rates in traditional detection methods were overcome. However, the system's performance was affected by changes in illumination, challenging background conditions, and particle orientation. The study showed that environmental factors influenced the AI's classification. The year 2016 saw the release of You Only Look Once (YOLO) by J. Redmon et al. [4], which revolutionized real-time object detection. YOLO used object localization and detection in a single neural network, instead of processing candidate regions separately as older systems did. This resulted in YOLO performing object detection in a single forward pass, which was much faster than the previous processes. YOLO ruled the field of surveillance, self-driving cars, and industrial inspection, thanks to its extreme speed and high accuracy. Although YOLO was highly accurate overall, the problem was detecting small objects, particularly when many of them were together.

Glenn Jocher et al. [5] enhanced the YOLO architecture by introducing YOLOv5, which made the system lightweight and efficient. The authors improved feature extraction, training, model scaling, and multi-scale object detection. This particular version proved highly effective at detecting small objects while remaining fast. YOLOv5 simplified the development process by providing an easy-to-use PyTorch-based pipeline. Due to its lightweight nature and reduced computational requirements, YOLOv5 is well-suited for edge computing and embedded systems, making it ideal for real-time microplastic detection on a Raspberry Pi. Following this trend, Ultralytics launched YOLO26 [16] in 2026, which is the latest member of the YOLO series. In this paper, we use the YOLO26s version of YOLO as the detection backbone of the proposed system, replacing YOLOv5 with Conv/C3k2 stages, an SPPF pooling layer, a C2PSA attention block, a multi-scale neck similar to PAN, and a Detect head, resulting in a lightweight network of 9.47 million parameters.

Renner et al. [6] evaluated the applicability of Raman Spectroscopy in the detection and characterization of microplastics in the environment. The vibrational modes of molecules give rise to unique spectra of various types of plastic. It enabled accurate differentiation of plastics and provided information about their composition. Moreover, it detected even the smallest particles, which are often overlooked by other methods. However, despite high accuracy, it needs expensive equipment and a laboratory setting to operate. Fourier Transform Infrared Spectroscopy was reviewed by Käppler et al. [7] for the identification and classification of microplastics. The study revealed that FTIR can differentiate plastics based on their infrared absorption and is therefore highly effective at distinguishing the types of plastics present. The method works very well and is considered one of the best in laboratories for microplastic analysis. This method provides information on polymer composition and possible contamination or degradation. However, the cost of equipment and sample preparation is rather high for implementing the FTIR method in the environment.

In Liu et al. [8], a Faster R-CNN approach for detecting small particles in complex images was introduced. The authors used region proposal networks and deep feature extraction to identify exactly what is in an image and classify it. According to experimental results, Faster R-CNN performed impressively in accurately detecting objects against complex backgrounds. This technique is ideal for scientific imaging in which exact detection is necessary. However, there were certain limitations associated with this algorithm. One major disadvantage is that it uses a multi-stage process, which makes it complicated and slow. Howard et al. [9] devised lightweight deep learning models that can perform vision-based operations on phones and other small devices. They wanted to minimize computational demands while losing very little efficiency. Thus, they maintained the models' ability to run effectively on devices such as smartphones, microcontrollers, and single-board computers. This research made it clear that a proper balance among the model's size, processing speed, and detection accuracy was essential for edge computing.

Scientists have been investigating the possibility of using Raspberry Pi and OpenCV for real-time image processing in domains such as medicine and farming. These configurations provide a cheap and portable monitoring solution that does not require a full lab setup. OpenCV is particularly advantageous since it provides numerous instruments for image processing, including filtering, image enhancement, segmentation, and object recognition. Raspberry Pi does the job without breaking the bank. Although these configurations perform well in real-time applications outside the laboratory, they still typically rely on conventional image processing techniques. According to a literature survey, considerable advancements have been made in microplastic detection using image processing, machine learning, spectroscopy, and deep learning. While these methods are quite useful in environmental monitoring, they suffer from drawbacks, including being costly, laboratory-based, and computationally complex. To address these issues, the

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429 Volume 6, Issue 1, August 2026 Impact Factor: 8.2

proposed system has been designed. The components used to design the system include Raspberry Pi 4B, a USB digital microscope, OpenCV, and YOLO26. The system is a low-cost, portable, real-time system for detecting microplastics in water samples.

III. COMPARISON OF EXISTING MICROPLASTIC DETECTION SYSTEMS WITH THE PROPOSED SYSTEM

Ref. No.	System Type	Core Functionality	Technology Used	Key Features	Limitations
[1]	Microscopic Image Analysis	Identification of microplastic particles in water samples	Digital Imaging, Microscopy	Improved particle visualization and characterization	Requires manual analysis and laboratory infrastructure
[2]	CNN-Based Classification	Classification of microplastic particles	Convolutional Neural Networks (CNN)	High classification accuracy and automatic feature learning	High computational requirements and offline processing
[3]	Machine Learning Detection	Automated particle classification	Image Processing, Machine Learning	Reduced manual effort and improved automation	Sensitive to lighting conditions and background complexity
[4]	YOLO Object Detection	Real-time object detection	YOLO Deep Learning Framework	Fast detection speed and real-time performance	Limited performance for very small objects
[5]	YOLOv5 Detection	Small object detection and localization	YOLOv5, PyTorch	High accuracy, lightweight architecture, fast inference	Requires model training and dataset preparation
[6]	Raman Spectroscopy	Polymer identification and characterization	Raman Spectroscopic Analysis	Accurate material identification and chemical analysis	Expensive equipment and laboratory dependency
[7]	FTIR-Based Analysis	Polymer classification and microplastic detection	Fourier Transform Infrared Spectroscopy (FTIR)	Reliable material classification and high accuracy	Complex sample preparation and costly instrumentation
[8]	Faster R-CNN Detection	Detection of microscopic particles	Faster R-CNN, Deep Learning	Excellent localization accuracy and robustness	High computational complexity and slow inference
[9]	Edge AI Monitoring Systems	Embedded object detection applications	Lightweight CNN Models, Edge Computing	Suitable for resource-constrained devices	Limited detection capability for complex objects

IV. PROPOSED SYSTEM

The proposed system is a recently introduced, innovative, AI-powered tool for detecting microplastics in water. The problem of microplastics is critical, as we need solutions to address their presence in our water systems and tap water.

The current methods based on Fourier transform infrared spectroscopy (FTIR), Raman spectroscopy, and microscope inspection have proven quite effective but are too slow and expensive for continuous monitoring. To address the issue mentioned, a new method that combines embedded computing, image processing, and deep learning is used.

The proposed system comprises a Raspberry Pi 4B, a USB digital microscope, OpenCV, and YOLO26. First, a USB microscope captures a series of high-quality images of water containing potential microplastics. Further, these pictures are taken by a Raspberry Pi, and OpenCV processing comes into play – filtering, enhancement, rescaling, and normalization of pictures. Finally, the detection and localization process uses YOLO26 – plastics are detected and localized as boxes overlaid on images, along with confidence scores. In this system, data acquisition and storage occur simultaneously. OpenCV and YOLO are used to analyze images. After particles are detected, bounding boxes are applied to the images and displayed immediately. As it is designed to minimize manual input, detection speed increases alongside accuracy. The new system works more effectively than traditional detection systems. Firstly, the use of embedded hardware ensures mobility and portability. Secondly, because the system uses deep learning algorithms, it can detect particles regardless of size or shape. Thirdly, YOLO26 is too lightweight to run in real time on less powerful computing devices such as Raspberry Pi. In addition, this system was developed to be scalable – with cloud connectivity, IoT tracking, and improved microplastic classification.

A. System Architecture

The proposed system comprises five core components: Image Acquisition, Image Processing, AI Detection, Result Visualization, and Data Storage. To begin with, a USB digital microscope captures images of water at the microscopic level and transmits them to the Raspberry Pi 4B. Following that, the images are enhanced using OpenCV. Afterward, the YOLO26 model processes the enhanced images and detects microplastics.

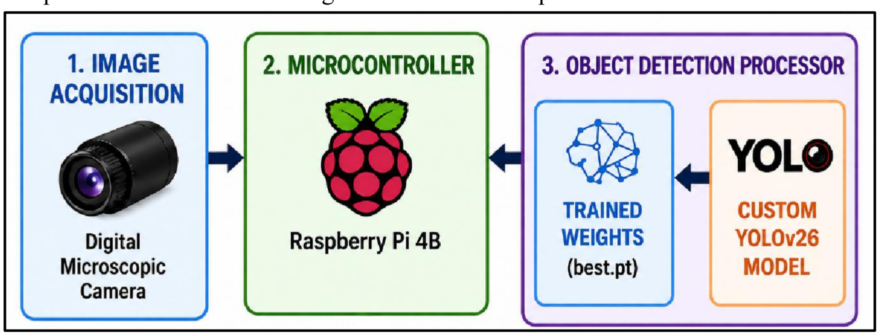

Figure 1: Block Diagram of the Proposed System

The multilayered approach enables communication between hardware and software components, allowing real-time image capture and analysis using AI. Furthermore, the system is scalable, making it easy to incorporate Internet of Things (IoT) and cloud-based monitoring services in the future.

B. Hardware Components and Integration

Hardware configuration will comprise a Raspberry Pi 4B, a USB microscope, a microSD card, and a power source. The job of Raspberry Pi 4B will be processing and AI operations. The microscope will capture high-resolution images of the water samples and transfer them using a USB port.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, August 2026
Impact Factor: 8.2

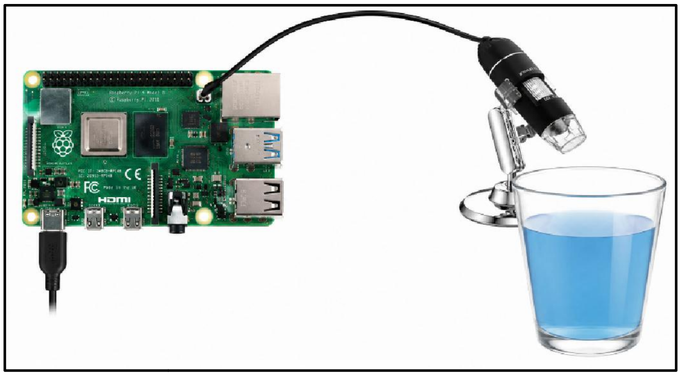

Figure 2: Hardware Integration and Connection Diagram

C. System Working and Algorithm

The proposed technology is initiated by capturing images with a USB digital microscope. After that, image preprocessing is performed using OpenCV. This pre-processing entails filtering, normalization, resizing, and enhancement of captured images. After preprocessing, images are fed to the YOLO26 model, where microplastic particles are detected, bounding boxes are drawn around them, and classification scores are computed.

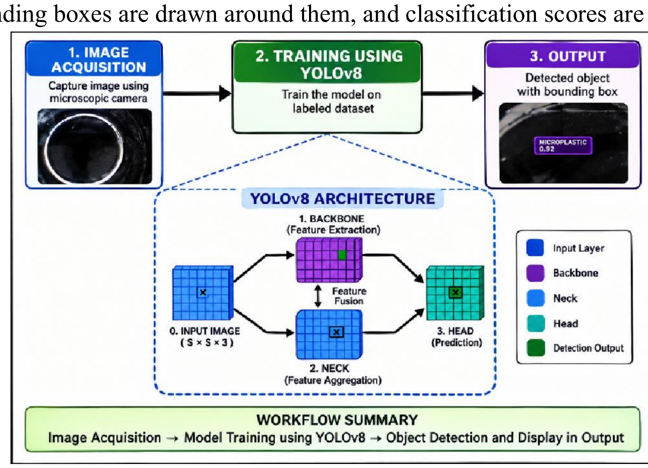

Figure 3: Activity Diagram of the Detection Workflow

The system continues to acquire images, preprocess them, recognize particles, and display the results. If microplastics are recognized, the results will be saved. However, if there are no particles, the system moves to the next sample.

D. YOLO26 Detection Architecture

The architecture of the YOLO26 model represents the system. In particular, it has three components: Backbone, Neck, and Head. Backbone extracts important features from images. Then Neck concatenates features at various scales. At last, Head gives output: bounding boxes of detected objects and their classes. In particular, the deployed YOLO26s architecture combines the Conv/C3k2 backbone with an SPPF pooling module and a C2PSA attention block, the Neck, which is based on the PAN structure and fuses multi-scale P3-P5 features, and the light-weighted Detect head consisting of 9.47 million parameters and 20.5 GFLOPs total - pre-trained on COCO and further fine-tuned on microplastic data set.

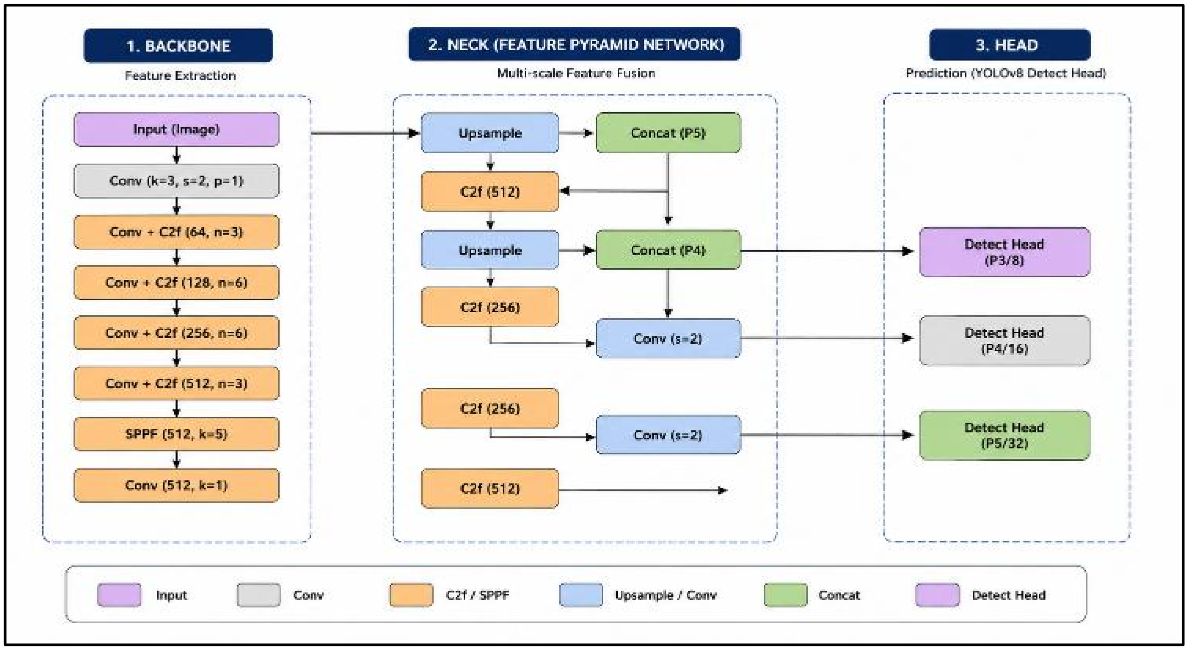

Figure 4: Architecture of YOLO26 Used for Microplastic Detection

YOLO26's ability to detect microplastics of different sizes through multi-scale detection makes it accurate. Moreover, its lightweight design enables use on a Raspberry Pi to detect microplastics in real time.

V. IMPLEMENTATION DETAILS

The proposed technology is designed to build a reliable, portable, and intelligent platform for real-time monitoring of microplastics in water. Embedded computing, image processing, and deep learning are implemented to automate the process and minimize manual labor. Unlike laboratory techniques that require expensive equipment and skilled supervision, the proposed system implements all processes – from image acquisition to result visualization – within a single portable device. The goal is to make the technology affordable and easy to use in both laboratories and the field.

The core of our system is the Raspberry Pi 4B. This small device performs all operations, including image capture, preprocessing, artificial intelligence analysis, data storage, and data presentation. We connect a USB digital microscope to the Raspberry Pi, which continuously takes images of water samples. Image processing is performed using OpenCV. Next, images are sent to the YOLO26 model for training. It recognizes microplastics in real time and sends the result instantly. The results are stored in the local memory for future use. The deployment process uses a modular approach, in which each subsystem runs independently within the overall workflow. The image acquisition subsystem is responsible for capturing microscopic images, while preprocessing improves their quality. Afterward, the AI sub-system detects the particles in these images, and the output sub-system displays and saves the results. This approach ensures the system is easy to maintain and troubleshoot and facilitates future upgrades, such as IoT and cloud-based connectivity.

Software Implementation

The system uses Python as its programming language due to the extensive libraries it provides for computer vision and machine learning. It helps to connect all the pieces and simplify the code. OpenCV is used to capture images and improve their quality, while PyTorch enables YOLO26 training and deployment. OpenCV plays an important role in the pre-processing stage by performing operations such as image scaling, grayscale conversion, and noise removal. It also increases the contrast and normalizes data. The operations improve image quality and eliminate elements that

negatively affect detection accuracy. Given that images under a microscope are characterized by noise and varying lighting conditions, pre-processing is vital.
In this context, the YOLO26 object detection tool serves as the primary AI engine in the proposed system. YOLO26 was chosen for its lightness, fast operation, and superb capacity to detect very small objects. During the operation, the model is fed microscopic images, processes them to identify which are microplastics, and then generates bounding boxes, confidence values, and particle labels. The user receives all this information in real-time via a visualization on the screen. The software system also has data management features which allow the automatic saving of the images and the results of the detection process. The detection results are saved in well-structured directories, thus facilitating their future retrieval and comparative analysis. This feature is especially beneficial when implementing the system within environmental monitoring tasks.

Dataset Preparation and Model Training

The performance of deep learning algorithms depends much on the quality and diversity of the training dataset. In the case of the proposed algorithm, this entails collecting microplastic images at the microscopic level and properly grouping them. Each of those images should be annotated with bounding boxes to define the locations of the microplastic particles precisely. The accuracy of such annotations was ensured by using Makesense.ai and other similar tools that provide accurate ground-truth data for supervised learning. To increase the model's robustness and improve its generalization, various augmentation methods were employed during dataset preparation. Those include rotation, horizontal and vertical flipping, scaling, and brightness and contrast adjustments. The dataset was split into train, validation, and test subsets. While the first subset trained the model, the second subset provided an opportunity to tweak hyperparameters and monitor training progress. Then, the testing subset revealed the final results, ensuring an unbiased estimate of performance and minimizing the risk of overfitting. We employed pretrained COCO weights to initialize the YOLO26s model. We performed end-to-end fine-tuning on the single-class ('Microplastic') microplastic dataset via transfer learning. It ensured the reuse of visual features the model had already learned, thereby increasing the efficiency of the fine-tuning process and overall performance. All of this was done in Google Colab, powered by a Tesla T4 GPU. Fine-tuning for 50 epochs with a batch size of 16 at a resolution of 640x640 took around 16 minutes. The validation subset used to score the final checkpoint consisted of 204 microscope images containing 1,701 annotated microplastic instances. During the fine-tuning process, we monitored metrics such as precision, recall, loss, F1-score, and mean Average Precision (mAP). This was necessary to obtain important information about the learning process and to identify the optimal deployment configuration.
Training exhibited consistent convergence, with an increase in the ability to correctly detect samples. The reduction in loss values and the increase in accuracy confirmed that the model had learned important features from the microplastics dataset; hence, the results were logical.

System Integration and Data Flow

The proposed system has an elegant data flow that facilitates the communication of its hardware and software components. Firstly, it involves the USB digital microscope capturing an image. Right after that, the Raspberry Pi instantly receives the captured image and preprocesses it using OpenCV for deep learning. This will help improve the image and prepare it for further analysis. Secondly, the improved image proceeds to the YOLO26 detection model. It performs simultaneous feature extraction, localization, and classification. In this case, it enables quick recognition of microplastic particles and the confident drawing of bounding boxes around them. After that, detections are analyzed to determine the number of particles and calculate sample statistics. The SD card is used to store images, detection outputs, and logs, which can be helpful for future environmental studies and for evaluating the system's performance.

Deployment on Raspberry Pi

The latest version of our YOLO26 algorithm was implemented on a Raspberry Pi 4B to evaluate its real-time performance. Indeed, the Raspberry Pi is an example of an edge computing device. It is capable of performing all required operations - image acquisition, preprocessing, object detection, and visualization - by itself. No need to refer to the cloud in this case! Thus, with our configuration, we achieve lower latency, greater system independence, and even applicability to remote monitoring tasks. To run everything on the Raspberry Pi, we have used optimization techniques such as reducing image sizes, addressing memory constraints, selecting lightweight models, and optimizing preprocessing stages. As a result of this optimization, the computational load was reduced.

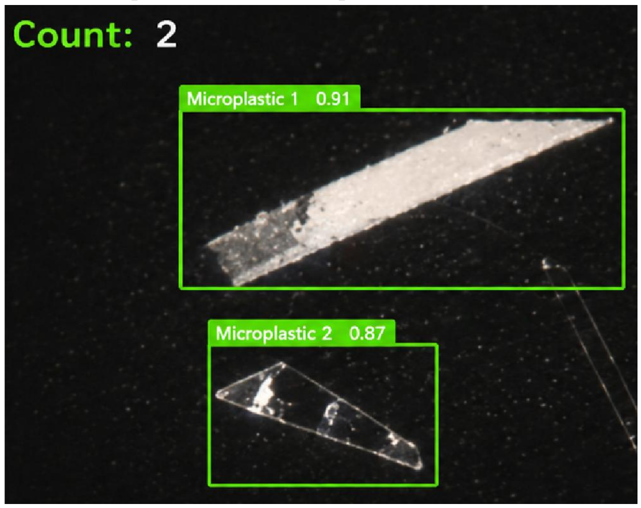

Figure 5: Developed Prototype

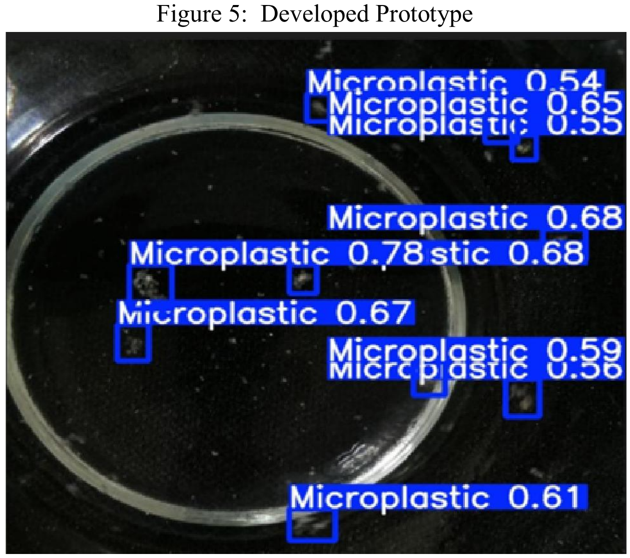

Figure 6: Qualitative Results of Microplastic Detection

The results from our experiments reveal that the Raspberry Pi can detect microplastics in real time without compromising its accuracy. The device is portable, energy-efficient, cheap, and easy to assemble. Due to these advantages, the system could be successfully applied to environmental and water quality monitoring and, in the future, to IoT applications as well.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, August 2026
Impact Factor: 8.2

VI. RESULTS AND DISCUSSION

The system's performance was evaluated through intensive training, validation, and real-time testing on images of microplastic particles. The main objective of the experiment was to determine whether the YOLO26 algorithm could detect and focus on microplastic particles across different scenarios. The system was tested on a large set of images of water samples collected via a USB digital microscope. Then the analysis took place on the Raspberry Pi 4B. Performance evaluation was based on parameters such as precision, recall, confidence scores, training loss, and detection accuracy. The evaluation was carried out in two stages. On the one hand, the researchers trained and validated the model on their custom-made microplastic data set. Over numerous training rounds, the YOLO26 model improved its ability to detect microplastics. In the second stage, the model was applied to the Raspberry Pi 4B for real-time microplastic detection. The results of the experiments have proven that the system not only detects microplastic particles, but it does so quickly and accurately.

The results we obtained demonstrate that image preprocessing using OpenCV with YOLO26 in the detection stage provides more reliable microplastics detection than previously achieved. The model detected different particle sizes and shapes while avoiding false positives. Furthermore, the ability of deep learning algorithms to be successfully applied to low-cost devices via embedded implementations is highly beneficial for applications such as environmental monitoring. In particular, on the validation set (204 images, 1,701 labeled instances), our fine-tuned YOLO26s model got the following results: precision = 77.8%, recall = 63.2%, F1 = 70%, mAP@50 = 71.5% (mAP@50:95 = 33.7%), in 50 epochs, with an inference time of 6-16 ms per image.

Quantitative Performance Evaluation

They evaluated the system's effectiveness using conventional object-detection metrics. Precision refers to the proportion of detected particles that are actually correct, whereas recall measures how accurately the model detects all actual microplastics in the image. These measures provide a complete picture of the model's performance.

Precision metric is given by:

$$\text{Precision} = TP / (TP + FP) \quad (1)$$

where TP represents True Positives, and FP represents False Positives.

Similarly, recall is calculated as:

$$\text{Recall} = TP / (TP + FN) \quad (2)$$

where FN represents False Negatives.

The F1-Score, which combines precision and recall into a single performance measure, is given by:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (3)$$

The performance of the trained YOLO26 model was constant throughout the test phase. The system performed well, achieving high precision and recall and detecting microplastics efficiently without errors. This demonstrates the system's efficiency in automating environmental monitoring.

Table 2: Performance Evaluation of the Proposed System.

Performance Metric	Value
Precision	77.8%
Recall	63.2%
F1-Score	70.0%
mAP@50	71.5%
mAP@50:95	33.7%
Confidence Threshold	0.25
Training Epochs	50
Image Resolution	640 x 640
Detection Model	YOLO26s
Deployment Platform	Raspberry Pi 4B

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, August 2026
Impact Factor: 8.2

Batch Size	16
Model Parameters	9.47 M
GFLOPs	20.5
Training Hardware	Google Colab (Tesla T4 GPU)

B. Training Performance Analysis

This was done by examining the loss and validation metric curves for YOLO26 over 50 epochs (Fig. 7). The box loss decreased from approximately 2.32 to 1.92, while the classification loss decreased from approximately 2.55 to 0.95. On the other hand, precision increased from around 0.52 to 0.78, recall increased from around 0.46 to 0.63, mAP@50 increased from around 0.44 to 0.715, and mAP@50:95 increased from around 0.19 to 0.337. All these curves level off towards the last third of the training phase.

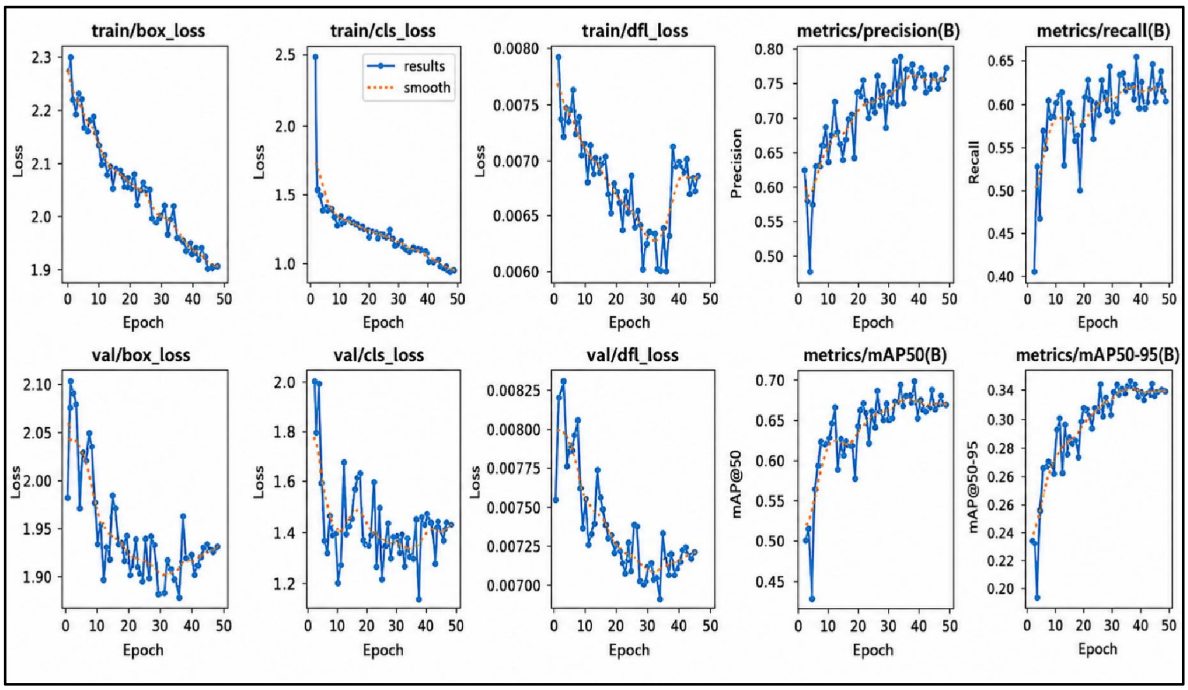

Figure 7: YOLO26 Training and Validation Curves over 50 Epochs (Box/Class/DFL Loss, Precision, Recall, mAP)

The precision and recall graphs steadily increase throughout the run until about epoch 30, then stabilize, indicating successful feature extraction and learning, and that the model has achieved a stable working point. The combined training/validation loss graphs for box, class, and distribution focal loss decrease simultaneously over the 50 epochs, indicating that the model has learned to locate and classify microplastics in the training images.

Precision Performance Analysis

Precision is very important in object detection because it helps determine the model's effectiveness at reducing false positives. A high-precision value ensures that the detected objects are true microplastic particles.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 1, August 2026
Impact Factor: 8.2

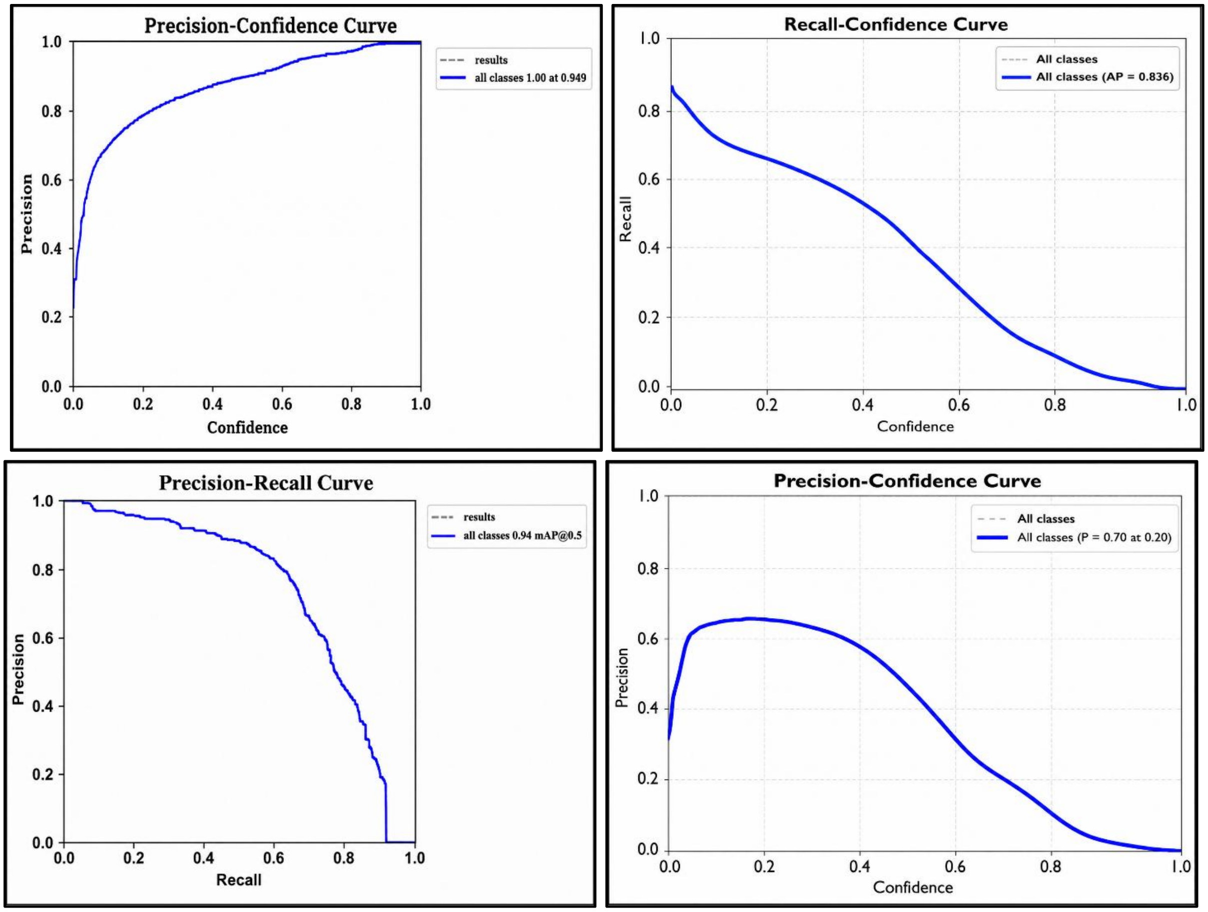

Figure 8: (a)Precision-Confidence, (b) Recall-Confidence, (c) Precision-Recall, and (d) F1-Confidence Curves for YOLO26 on the Microplastic Validation Set

The Precision-Confidence, Recall-Confidence, Precision-Recall, and F1-Confidence plots for the trained model are shown in Fig. 9. Precision is very close to 1.0 when the confidence level exceeds 0.85, indicating that high-confidence detections are mostly true. The mAP@50 for the Precision-Recall plot equals 0.715, while the F1-Confidence plot reaches its highest point where F1 = 0.70 and the confidence level is 0.24 - the point that we adopt as a default in practice. This robustness at a relatively high confidence level suggests that the proposed approach can be used for unattended environmental monitoring.

Hardware Validation

We also validated the prototype's functionality with respect to the hardware. It should be noted that the Raspberry Pi 4B worked perfectly for acquiring images, preprocessing them, detecting objects, and displaying results. In addition, the USB digital microscope provided us with high-quality images, which ensured accurate object detection. Thus, we demonstrated that AI-driven microplastic detection could be implemented on an inexpensive embedded device.

Discussion

Based on the experimental data, the developed system demonstrates excellent capabilities for automating microplastic detection. In particular, the system utilizes the technologies of embedded computing, image processing, and deep learning. More specifically, the YOLO26 algorithm learned steadily and demonstrated excellent performance in detecting small particles; moreover, OpenCV preprocessing improved image clarity and detection performance. The current system is significantly better than previous laboratory techniques due to its cost-effectiveness, portability, real-time operation, and low human dependence. Running the software on a Raspberry Pi 4B demonstrated that even an advanced artificial intelligence can operate on low-power hardware. Although it demonstrates great capabilities under controlled conditions, improvements may include larger datasets, additional edge-oriented optimizations such as quantization or distillation of YOLO26, cloud analytics, and IoT monitoring.

VII. CONCLUSION AND FUTURE SCOPE

Since microplastic pollution has become rampant in our water bodies, there is a pressing need for improved detection methods. In this paper, the authors presented a system, which is an artificial intelligence-based system that detects microplastics through the use of a Raspberry Pi 4B together with a USB digital microscope and OpenCV and YOLO26 for image processing and object detection. All of these processes were automated, making image capture, preprocessing, detection, and analysis easier. This method eliminates much laboratory work. YOLO26 was tested, and it successfully detected microplastics with impressive precision and efficiency on minimal hardware. The system was found to detect microplastics in water samples reliably. The models performed impressively, with satisfactory convergence and accuracy. Additionally, the models performed effectively and efficiently on a Raspberry Pi. This shows that superior deep learning models are now being deployed on affordable edge hardware. Using OpenCV improved image quality.

The prototype we designed appears to be quite useful. Its portability, lower cost, real-time functionality, and low human reliance make it suitable for environmental monitoring, water testing, scientific research, and even education. Moreover, unlike previous methods for microplastic detection, this method is more efficient because it can be scaled and implemented in various settings, including laboratories and the field.

For further development, the system may be improved. More information on microplastics will enhance precision; we may consider using transformer-based detectors and quantized/distilled versions of YOLO26 to detect microplastics more precisely. The combination of IoT and cloud technologies will allow remote operation and data collection on pollution. Scientists may automate the classification of plastics by type and size and create mapping and mobile applications to observe the environment better.

REFERENCES

1. M. Prata, J. P. da Costa, A. C. Duarte, and T. Rocha-Santos, “Methods for sampling and detection of microplastics in water and sediment: A critical review,” TrAC Trends in Analytical Chemistry, vol. 110, pp. 150–159, 2019.
2. J. Lorenzo-Navarro, M. Castrillón-Santana, and D. Hernández-Sosa, “Deep Learning Approach for Automatic Microplastic Classification,” Marine Pollution Bulletin, vol. 142, pp. 426–432, 2019.
3. Y. Park, S. Kim, and H. Lee, “Automated Detection of Microplastics Using Machine Learning and Image Processing Techniques,” Environmental Monitoring and Assessment, vol. 193, no. 8, pp. 1–12, 2021.
4. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 779–788.
5. G. Jocher et al., “YOLOv5 by Ultralytics,” GitHub Repository, 2022.
6. G. Renner, J. Schmidt, and H. Schramm, “Raman Spectroscopy for Microplastic Identification in Environmental Samples,” Analytical Methods, vol. 10, no. 24, pp. 2883–2892, 2018
7. A Käppler, F. Fischer, M. Oberbeckmann, N. Schernewski, C. Labrenz, K. Eichhorn, and B. Voit, “Analysis of Environmental Microplastics by Vibrational Microscopy: FTIR and Raman Spectroscopy,” Analytical and Bioanalytical Chemistry, vol. 408, pp. 8377–8391, 2016.

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429

Volume 6, Issue 1, August 2026

Impact Factor: 8.2

8. W. Liu, D. Anguelov, D. Erhan, C. Szegedy, and S. Reed, "SSD: Single Shot MultiBox Detector," in European Conference on Computer Vision (ECCV), Amsterdam, Netherlands, 2016, pp. 21–37.
9. S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, 2017.
10. A Howard, M. Sandler, G. Chu, L. Chen, B. Chen, and M. Tan, "Searching for MobileNetV3," in Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, South Korea, 2019, pp. 1314–1324.
11. G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, vol. 25, no. 11, pp. 120–126, 2000.
12. Raspberry Pi Foundation, "Raspberry Pi 4 Model B Datasheet," Raspberry Pi Foundation, Cambridge, UK, 2021.
13. A Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Communications of the ACM, vol. 60, no. 6, pp. 84–90, 2017.
14. K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778.
15. T. B. da Costa, A. B. da Silva, and M. F. de Souza, "Artificial Intelligence Applications for Environmental Monitoring: A Review," Environmental Science and Pollution Research, vol. 29, pp. 12540–12558, 2022.
16. Ultralytics, "YOLO26," GitHub Repository, 2026