Journal of Science, Computing and Engineering Research (JSCER)
Volume-9, Issue-3, March 2026.

DOI: *https://doi.jscer.org/10.46379/jscer.2025.080533*

WATER QUALITY PREDICTION USING DEEP LEARNING

[1]J.R. Arun Kumar,[2]Mayank Kumar, [3]Vaibhav Sharma, [4]Naman Jain, [5]Harshit Sharma

[1]Professor, Department of CSE & AI, Modern Institute of Technology and Research Centre, Rajasthan, India.

[2,3,4,5,6] UG Student, Department of AI&DS, Modern Institute of Technology and Research Centre, Rajasthan, India

Article Information

Received : Mar 29 2026

Revised : Mar 29 2026

Accepted : Mar 30 2026

Published : Mar 31 2026

Corresponding Author:

Mayank Kumar

Abstract— Access to safe drinking water is a critical public health requirement. In many regions, water quality is still assessed manually or using laboratory-based chemical tests, which are often time-consuming, costly, and not scalable for continuous monitoring. This project presents Water Quality Prediction Using Deep Learning, a system that utilizes computer vision and deep learning to estimate water turbidity from images and optionally integrates user-provided TDS values to determine overall potability. The system allows the user to capture or upload an image of water in a container. A convolutional neural network (CNN)-based deep learning model predicts an approximate turbidity range directly from the image. The solution follows a modular approach: image acquisition, preprocessing, turbidity estimation using a deep learning model, fusion with optional TDS input, and final classification into Drinkable or Not Drinkable based on predefined thresholds inspired by standard water quality guidelines. The system can be deployed as a simple web application, enabling non-expert users to quickly check water quality without requiring chemical labs.

Keywords: Water Quality, Deep Learning, CNN, ResNet, Turbidity Prediction, Computer Vision, Potability Classification, Image Analysis, FastAPI, TDS

Citation: J.R.Arun Kumar, Mayank Kumar, Vaibhav Sharma, Naman Jain, Harshit Sharma, "WATER QUALITY PREDICTION USING DEEP LEARNING", Journal of Science, Computing and Engineering Research, 9(03), March 2026

I. INTRODUCTION

Water Quality Prediction using Deep Learning is an artificial intelligence approach designed to estimate the safety of water for human consumption by analyzing an image of a water sample. Traditionally, water quality assessment depends on laboratory testing or chemical test kits, which are reliable but require time, cost, trained technicians, and proper sampling procedures. In many regions, especially in rural or low-income communities, such facilities are not easily available, and people often judge water quality by visual inspection. This method is subjective, inaccurate, and may lead to serious health risks.

To address these limitations, the proposed system uses Convolutional Neural Networks (CNNs), which are capable of extracting complex visual patterns from images. The system focuses on turbidity, a key indicator of water quality that refers to the cloudiness of water caused by suspended particles such as silt, clay, organic matter, or microorganisms. High turbidity increases the probability of contamination and water-borne diseases. International standards, including WHO and BIS,

recommend that drinking water should have low turbidity for safe consumption and effective disinfection.

In this system, users capture or upload an image of water in a transparent container. The CNN model analyzes visual cues such as color intensity, light scattering, and particle presence to estimate a turbidity level or range. Based on predefined threshold values, the system classifies the sample into Drinkable (Safe for Consumption) or Not Drinkable (Unsafe for Consumption)This approach is highly accessible because it requires only a camera, commonly available in smartphones and low-cost digital devices. Predictions are generated within seconds, enabling early detection of potentially contaminated water and reducing the risk of water-borne health issues.

The end goal is simple: provide users with a reliable, adaptive, and efficient AI-based water quality prediction tool that can be accessed anytime, enhancing water safety without the need for laboratory infrastructure.

II. PROBLEM STATEMENT

The availability of clean and safe drinking water is one of the most critical requirements for human survival, public health, and socio-economic development. However, the modern water safety landscape is undergoing a significant transformation due to increasing contamination, industrial waste discharge, agricultural runoff, and irregular sanitation practices. Whether in urban or rural regions, determining water potability has become a growing concern as pollutants continue to enter freshwater sources such as rivers, lakes, groundwater reserves, and storage systems.

Traditional water quality assessment methods rely heavily on laboratory chemical analysis, physical testing, or specialized instruments such as turbidity meters and microbial test kits. These approaches, although scientifically reliable, are often time-consuming, expensive, require trained personnel, and depend on physical infrastructure that may not be available to a large portion of the population. People in many low-resource or rural settings still rely on visual inspection of water, which can be highly misleading and dangerous.

Turbidity is considered one of the most important preliminary indicators of water quality because it reflects the cloudiness or haziness caused by suspended solids such as clay, silt, organic matter, algae, and microbial contamination. Higher turbidity increases the risk of water-borne diseases including diarrhea, cholera, and gastrointestinal infections. Both the World Health Organization (WHO) and the Bureau of Indian Standards (BIS) emphasize the need to maintain turbidity at low levels to ensure effective disinfection and safe consumption.

Deep learning and computer vision are emerging as powerful techniques to address these accessibility issues. Modern CNNs have demonstrated strong capabilities in extracting complex visual patterns from images, making it possible to analyze physical properties of water without direct contact or chemical testing. By processing a simple image captured from a smartphone, a deep learning model can estimate turbidity levels and classify the water sample as Drinkable or Not Drinkable based on defined safety thresholds.

III. PROPOSED METHOD

The proposed system is designed to provide accurate, accessible, and automated water quality assessment by combining image analysis with a deep learning model. It uses a Convolutional Neural Network (CNN) trained to detect visual characteristics associated with turbidity, enabling classification of water samples as Drinkable or Not Drinkable without laboratory equipment.

A. Image Acquisition

The process begins when the user provides an image of a water sample in a transparent glass or plastic container. Images can be captured using a smartphone camera, laptop webcam, or uploaded as pre-stored image files. The frontend interface provides an image upload mechanism with file validation for supported formats such as JPG and PNG.

B. Image Preprocessing

Raw captured images vary in size, orientation, background, lighting, and noise. The system performs preprocessing operations including resizing the image to a fixed dimension (224x224 pixels) matching the CNN input requirement, converting to RGB format, and normalizing pixel values to a uniform range. Random augmentations such as brightness and contrast adjustments are applied during training to improve robustness.

C. Deep Learning Model (CNN)

The core of the system is a Convolutional Neural Network using a ResNet-based backbone initialized with ImageNet pre-trained weights. The convolutional layers generate high-level feature maps capturing texture, color transitions, edge patterns, and scattering behavior linked

D. Turbidity Estimation

After feature extraction, the model produces a turbidity prediction which is mapped to human-readable turbidity ranges. The system then applies threshold-based classification using reference guidelines from WHO and BIS. If turbidity falls below the defined safety limit, water is classified as Drinkable; if it exceeds the limit, it is classified as Not Drinkable.

E. Backend API Layer

The backend is implemented using FastAPI, acting as the orchestration hub linking the frontend, deep learning model, and database. It exposes REST API endpoints for image upload, prediction requests, and health checks. The backend handles image reception, validation, preprocessing, model inference, and returns results as JSON responses to the frontend.

F. Frontend Interface

The frontend serves as the primary interaction layer, allowing users to upload water images and receive turbidity predictions in a simple and accessible manner. Users can upload or drag an image of a water sample and initiate the prediction process. After backend analysis, the frontend displays an image preview, estimated turbidity level, and the final drinkability classification.

G. Database and Logging

An optional database layer stores uploaded image metadata, prediction results, and user interaction history. This supports future analysis, model retraining with new data, and tracking of water quality assessments over time. Structured data is stored in a relational database while prediction logs are maintained for system monitoring.

H. System Architecture and Deployment

The system follows a modular architecture ensuring scalability and maintainability. The frontend is implemented with React and TailwindCSS. FastAPI manages backend services and API orchestration. The deep learning model is exported in PyTorch or TensorFlow format and loaded by the inference service. The system is designed to be deployed as a web application accessible from any device with a browser and internet connection.

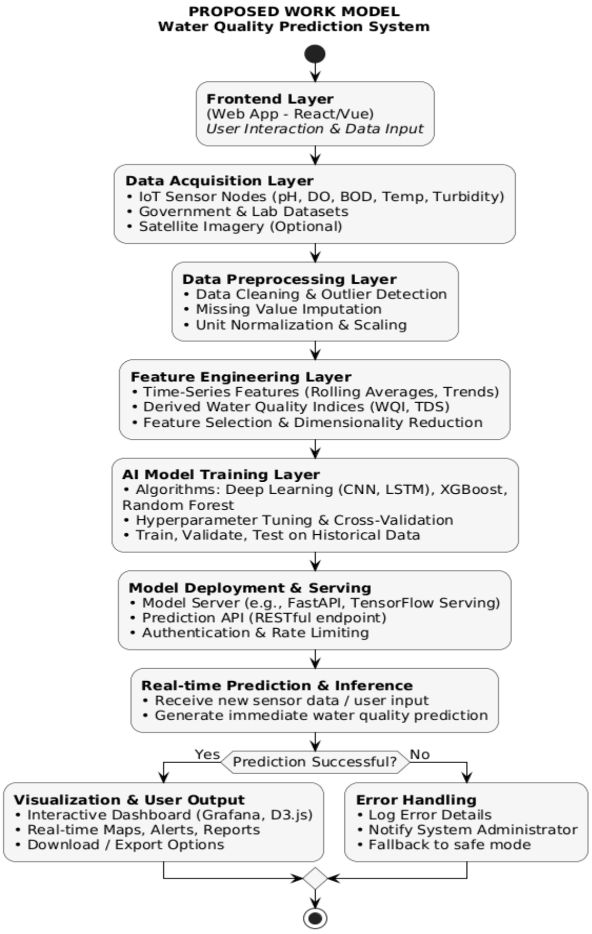

Fig. 1. Proposed Work Model

IV. TECH STACK

A. Frontend Technologies

The frontend is built using React.js for its component-based architecture. TailwindCSS is integrated for responsive and consistent UI styling. Interactive charting libraries such as Chart.js and Plotly are used to visualize turbidity trends and prediction outputs. React hooks and context API manage application state efficiently, while Axios handles asynchronous HTTP communication with the backend API.

B. Backend Technologies

The backend is built using FastAPI, a modern Python framework supporting high-performance API development. It manages image reception and validation, preprocessing orchestration, model inference requests, and result formatting as JSON responses. FastAPI's built-in support for asynchronous request handling and automatic OpenAPI documentation simplifies both development and testing of API endpoints.

C. Artificial Intelligence and Deep Learning

The system is powered by a CNN based on the ResNet architecture, pre-trained on ImageNet and fine-tuned for turbidity estimation. The model is trained using TensorFlow or PyTorch and extracts visual features such as color distribution, particle density, and scattering patterns from water images.

. D. Database Technologies

The system uses a relational database such as PostgreSQL or MySQL to store user records, image metadata, prediction results, and system logs. Images are saved in local server storage with only file paths stored in the database.

E. Image Processing

The trained model is exported as a TensorFlow SavedModel or PyTorch .pt file and loaded by the FastAPI inference service. TensorFlow Serving can be used for production-level model serving with support for batching, versioning, and GPU acceleration. Model versioning strategies ensure seamless updates and rollbacks without service interruption, maintaining continuous availability during redeployment cycles.

F Model Deployment and Serving

The system implements JWT-based authentication, ensuring only authorized users can upload images and view predictions. Sensitive configuration values such as database credentials and API keys are stored in environment variables using a .env file. Additionally, input validation and file type verification are enforced on all uploaded images to prevent malicious file injection, and HTTPS is recommended for all client-server communication in production.

G. Authentication and Security

The system implements JWT-based authentication, ensuring only authorized users can upload images and view predictions. Sensitive configuration values such as database credentials and API keys are stored in environment variables using a .env file.

H. Development Tools and Deployment Support

The project is developed using Visual Studio Code. Git and GitHub are used for version control. Docker containerization enables consistent deployment across environments. The system supports cloud deployment on platforms such as AWS, GCP, or Azure for scalability. CI/CD pipelines configured through GitHub Actions automate testing, building, and deployment workflows, reducing manual overhead and ensuring code quality across all stages of development.

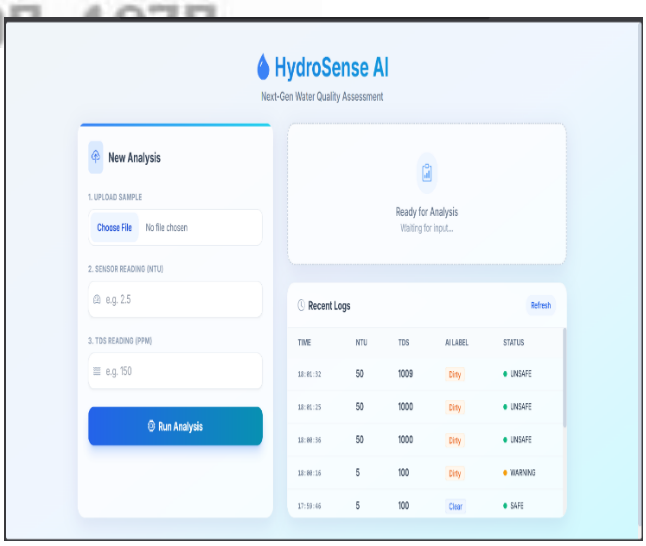

Fig. 2. Web Application Interface for Water Quality Prediction

V. RESULTS

The developed Water Quality Prediction Using Deep Learning system successfully demonstrates the practical application of computer vision and deep learning for accessible water safety assessment. The system accurately analyzes uploaded water images and classifies them as

Drinkable or Not Drinkable within seconds. During testing, the CNN-based ResNet model demonstrated strong performance in distinguishing between clear and turbid water samples across varying lighting conditions and container types. The web application interface allows non-technical users to easily upload images and receive immediate predictions. The modular architecture ensures each component operates reliably and independently. Overall, the results show that the proposed system can effectively function as a fast, low-cost pre-screening tool for water quality assessment.

VI. CONCLUSION

The Water Quality Prediction Using Deep Learning project demonstrates how modern artificial intelligence techniques can address real-world environmental and public health challenges. The developed system utilizes a CNN based on ResNet to analyze visual characteristics of water samples. By examining features such as color distribution, suspended particle density, and turbidity patterns, the model estimates the turbidity level and classifies the sample as drinkable or not drinkable using thresholds from recognized water quality standards. This automated approach reduces dependency on manual inspection and laboratory equipment, enabling users to obtain quick preliminary assessments within seconds. Although the model does not replace certified laboratory testing, it serves as an effective pre-screening mechanism that can be extended to incorporate additional water quality parameters, mobile deployment, and real-time IoT sensor integration.

REFERENCES

[1]. Central Pollution Control Board. (2023). Water quality standards and monitoring data. Ministry of Environment, Forest and Climate Change, Government of India. https://cpcb.nic.in

[2]. U.S. Environmental Protection Agency. (2023). Water quality guidelines and standards. https://www.epa.gov

[3]. Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[4]. Abadi, M., et al. (2016). TensorFlow: A system for large-scale machine learning. Proceedings 12th USENIX OSDI, 265-283.

[5]. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. IEEE CVPR.

[6]. World Health Organization. (2017). Guidelines for Drinking-water Quality, 4th edition. WHO Press.

[7]. Bureau of Indian Standards. (2012). IS 10500: Drinking water - Specification (2nd revision). BIS, New Delhi.

[8]. World Health Organization. (2022). Drinking-water quality surveillance and monitoring. WHO Technical Report Series. https://www.who.int

[9]. Chollet, F. (2017). Deep learning with Python. Manning Publications.

[10]. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems (NeurIPS), 25, 1097–1105.