IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

An AI-based Extreme-Edge TCN-Based Low-Latency Collision-Avoidance Safety System for Industrial Machinery

Garde Ashwini Rajendra
Department of e&tc engineering
Shivanagar vidya prasarak mandal
college of engineering Malegaon

Dr..Doshi N.A
Department of e&tc engineering
Shivanagar Vidya Prasarak Mandal
College of Engineering, Malegaon

***Abstract*— The rise of autonomous and semi-autonomous machinery in industrial settings necessitates advanced safety mechanisms to ensure smooth operation while preventing collisions and protecting nearby workers. This project proposes an AI-based Extreme-Edge Collision-Avoidance System utilizing a Temporal Convolutional Network (TCN) deployed on an STM32 microcontroller. The system integrates multiple sensing technologies, including LIDAR, Camera Modules, Ultrasonic Sensors, and Infrared Sensors, to comprehensively monitor the machine's surroundings. At the system's core, the STM32 microcontroller processes real-time data from the sensors via a driver circuit, ensuring ultra-low-latency response. AI-based monitoring runs on a Raspberry Pi, analyzing time-series sensor data using the TCN model to detect potential hazards in real-time. The AI algorithm predicts collision risks and enhances situational awareness, ensuring timely interventions critical for industrial safety. Upon detecting an obstacle, the STM32 microcontroller triggers immediate corrective actions, controlling machinery operations while engaging an alarm and alert system to notify nearby workers. The integration of Raspberry Pi extends computational flexibility, supporting data logging, visualization, and remote monitoring, while enabling machine learning model updates. The system ensures robust performance in noisy and dynamic industrial environments by leveraging sensor fusion and AI. Extreme-edge processing minimizes latency, optimizes energy consumption, and maintains a compact memory footprint. Designed for seamless integration into industrial applications, this real-time collision avoidance system significantly enhances workplace safety, reduces accidents, and improves operational efficiency.**

Keywords— Collision avoidance, Industrial safety, Edge computing, Ultrasound sensors, Temporal Convolutional Network (TCN), Machine learning, Low-power MCU, Raspberry Pi, Real-time processing, Sensor fusion, Acoustic noise robustness, Embedded systems, Proximity sensing, Smart manufacturing, Industrial automation.

I. Introduction

In industrial environments, the increasing complexity and automation of machinery present new challenges for worker safety and operational efficiency. The rise of smart manufacturing and Industry 4.0 has led to the development of interconnected and autonomous systems, where machines perform tasks with minimal human intervention. However, as industrial machinery operates in dynamic environments, ensuring robust, low-latency collision avoidance becomes crucial to safeguarding workers and preventing costly equipment damage. Traditional collision-avoidance systems often rely on centralized processing, where sensor data is transmitted to cloud-based platforms for decision-making. While effective in some scenarios, this approach introduces higher latencies that may compromise safety in industrial settings, where split-second responses are required. Additionally, industrial environments are characterized by high levels of acoustic and electromagnetic noise, vibrations, and variable lighting conditions, which can degrade the accuracy of many sensing technologies. Existing solutions must address these challenges while adhering to strict constraints related to power, memory, and processing capabilities, particularly when deployed on resource-limited embedded devices.

To overcome these challenges, Edge AI offers a promising solution by enabling real-time computation directly at the machine level, reducing communication delays and dependency on external cloud infrastructure. Edge AI systems must be low-power, compact, and efficient, particularly in environments where energy efficiency and real-time performance are critical, such as in battery-powered or autonomous systems.

In this proposed work, Temporal Convolutional Networks (TCNs) emerge as an ideal solution for processing time-series data generated by industrial sensors. Unlike traditional machine learning models or Recurrent Neural Networks (RNNs), TCNs provide several advantages: they maintain long memory, efficiently learn temporal dependencies, and process sequences in parallel, enabling faster inference. These features make TCNs highly suitable for real-time prediction tasks, such as collision avoidance, where rapid decision-making is essential. Furthermore, TCNs' ability to handle noise and their efficient use of computational resources make them ideal for deployment in industrial environments. This work proposes an AI-based extreme-edge collision-avoidance system utilizing a TCN model, deployed on an STM32 microcontroller, with additional computational support from a Raspberry Pi. The system integrates Ultrasonic (US) sensors, LIDAR, camera modules, and infrared sensors to detect potential collisions or obstacles in real-time. Ultrasonic sensors, in particular, are well-suited for industrial applications due to their ability to measure proximity accurately, even in conditions of low visibility or excessive noise, making them more reliable than vision-based systems in certain scenarios.

The STM32 microcontroller serves as the primary real-time processing unit, efficiently handling sensor data and executing the TCN model with ultra-low latency. Meanwhile, the Raspberry Pi plays a complementary role, offering additional computational capabilities for AI model execution, data logging, and remote monitoring. This hybrid approach balances energy efficiency with high-performance processing, ensuring that the system operates autonomously while remaining capable of incremental learning and future updates.

A sensor-fusion dataset, collected from Ultrasonic sensors mounted on an industrial woodworking machine, forms the basis for training and refining the TCN model. By leveraging incremental learning, the system continuously adapts to evolving environmental conditions and operational nuances, improving its accuracy over time. This adaptability is particularly advantageous in industrial settings, where machinery behaviour and surrounding environments fluctuate due to factors such as wear and tear, workload variations, acoustic noise, and vibrations. By combining STM32 for ultra-low-latency execution, Raspberry Pi for AI-driven processing and monitoring, and a TCN-based AI model for predictive analytics, this work presents a highly efficient, real-time collision-avoidance system. The proposed solution enhances industrial safety, minimizes accidents, and ensures seamless integration into existing manufacturing environments, marking a significant advancement in AI-powered edge computing for smart manufacturing.

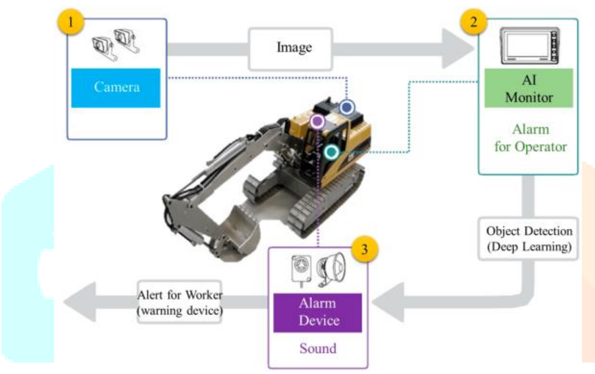

Fig 1. Diagram of proposed work

The diagram illustrates an AI-powered collision avoidance system designed for industrial machinery, integrating real-time object detection and deep learning to enhance worker safety and prevent accidents. The system consists of three main components: camera modules, an AI monitoring system, and an alarm device. Cameras mounted on the machinery capture real-time images of the surroundings and send them to the AI monitoring unit for processing. Using deep learning-based object detection algorithms, particularly a Temporal Convolutional Network (TCN) model deployed on STM32 and Raspberry Pi, the system identifies workers, obstacles, or potential hazards in the machinery's path. If a collision risk is detected, the system triggers an alarm for the machine operator.
Additionally, an alarm device is activated to provide sound alerts, ensuring that workers in the danger zone receive immediate warnings. This multi-modal alert mechanism helps operators and workers react swiftly, preventing accidents. The project aims to enhance industrial safety by utilizing Edge AI for ultra-low latency response, enabling real-time hazard detection and intelligent decision-making at the machine level. By deploying deep learning models at the edge, the system ensures faster processing and immediate action without relying on cloud connectivity. Ultimately, this AI-powered collision avoidance system improves safety in industrial environments, significantly reducing accident risks through intelligent automation and real-time alerts.

II. PROBLEM STATEMENT:

Design and Development of an AI-driven low-latency Extreme-Edge Collision Avoidance Safety System for Industrial Machinery Using TCN, Sensor Fusion, and Embedded Controllers. Industrial machinery operates in complex and dynamic environments where the risk of collisions poses significant threats to worker safety, equipment longevity, and operational efficiency. Traditional safety approaches, such as manual monitoring and reactive safety measures, are often slow, inconsistent, and prone to human error. These limitations result in delayed responses to potential hazards, leading to accidents, costly downtime, and damage to both personnel and machinery.
The primary challenge is to develop a real-time, intelligent collision-avoidance system that can reliably detect and prevent collisions with minimal human intervention. Existing solutions struggle in high-noise environments, have limited adaptability to rapidly changing industrial conditions, and often suffer from high latency. As manufacturing processes become increasingly automated and high-speed machinery dominates industrial workflows, there is an urgent need for low-latency, AI-powered safety solutions capable of instant hazard detection and prevention without disrupting operations.
This project aims to leverage Artificial Intelligence (AI), Temporal Convolutional Networks (TCN), and sensor fusion techniques to create an advanced safety system that ensures continuous real-time monitoring and precise decision-making. By deploying AI inference directly at the edge using embedded controllers, the system minimizes processing delays, making it highly suitable for high-speed industrial environments. This innovation enables rapid hazard detection, proactive collision prevention, and seamless adaptability to diverse industrial settings.

Significance of the Problem Statement:

1. Clear Focus: Addresses the inefficiencies of traditional safety systems and emphasizes the need for AI-driven automation.
2. Technology Integration: Highlights the role of AI, sensor fusion, and embedded controllers in achieving real-time safety monitoring and response.
3. Impact & Justification: Demonstrates the necessity of improved worker protection, enhanced machine efficiency, and reduced industrial hazards.
4. Success Metrics: Defines performance expectations, including real-time detection, ultra-low latency response, environmental adaptability, and robust industrial safety enhancements.

Objective:

1. Designing a retractor mechanism that can be controlled automatically with the help of the application.
2. Integrating AI to adjust retraction pressure and positioning dynamically based on real-time feedback.
3. Utilizing IoT to monitor and control the retractor remotely, ensuring precise and consistent retraction.
4. Testing and validating the design through simulations and practical trials to ensure its effectiveness and safety.
5. Evaluating the performance of the automatic retractor in comparison to traditional manual methods in terms of precision, efficiency, and user satisfaction.

Scope of the project:

a. **Designing a multi-sensor collision-avoidance** system that integrates various sensors such as LIDAR, ultrasonic, IR, and camera modules for real-time monitoring and detection of obstacles in industrial environments. The system will be controlled automatically through embedded algorithms running on a microcontroller.
b. **Integrating AI** (Temporal Convolutional Network - TCN) to analyze sensor data and dynamically adjust the machinery's response to potential hazards. The AI will help in predicting and preventing collisions by adapting to real-time environmental changes and providing low-latency responses.
c. **Utilizing IoT technology** to monitor the system remotely, allowing for remote alerts, diagnostics, and control of the safety system. This will ensure continuous operation, even in complex industrial settings, by providing live feedback to operators and remote monitoring units.
d. **Testing and validating** the collision-avoidance system through simulations and practical trials in industrial environments to ensure its accuracy, reliability, and safety. Validation will include testing in various noise and lighting conditions to confirm robustness.
e. **Evaluating** the performance of the automated collision-avoidance system against traditional safety mechanisms in terms of precision, latency, energy efficiency, and overall improvement in operator safety. The evaluation will focus on improvements in real-time adaptability, reduction of human error, and overall system efficiency.

III. Literature Review

Functional safety of electrical/electronic/programmable electronic safety-related systems - Part 1: General requirements (see Functional Safety and IEC 61508): IEC 61508-1:2010 covers those aspects to be considered when electrical/electronic/programmable electronic (E/E/PE) systems are used to carry out safety functions. A major objective of this standard is to facilitate the development of product and application sector international standards by the technical committees responsible for the product or application sector. This will allow all the relevant factors, associated with the product or application, to be fully taken into account and thereby meet the specific needs of users of the product and the application sector. A second objective of this standard is to enable the development of E/E/PE safety-related systems where product or application sector international standards do not exist. This second edition cancels and replaces the first edition published in 1998. This edition constitutes a technical revision. It has been subject to a thorough review and incorporates many comments received at the various revision stages. It has the status of a basic safety publication according to IEC Guide 104. This publication is of high relevance for Smart Grid.

K. Cao, Y. Liu, G. Meng, and Q. Sun, ''An overview on edge computing research,'' IEEE Access, vol. 8, pp. 85714–85728, 2020: Edge computing is a novel computing paradigm designed to address the limitations of traditional cloud computing in the era of the Internet of Everything (IoE). As the number of smart devices increases, generating vast amounts of data, issues like bandwidth load, slow response times, security, and privacy concerns become more prominent. Edge computing addresses these challenges by performing data processing closer to the data source, providing faster, real-time, and secure services.

F. Wang, M. Zhang, X. Wang, X. Ma, and J. Liu, ''Deep learning for edge computing applications: A state-of-the-art survey,'' IEEE Access, vol. 8, pp. 58322–58336, 2020: With the booming development of Internet-of-Things (IoT) and communication technologies such as 5G, our future world is envisioned as an interconnected entity where billions of devices will provide uninterrupted service to our daily lives and the industry. Meanwhile, these devices will generate massive amounts of valuable data at the network edge, calling for not only instant data processing but also intelligent data analysis to fully unleash the potential of the edge big data. Both traditional cloud computing and on-device computing cannot sufficiently address this problem due to the high latency and the limited computation capacity, respectively. Fortunately, emerging edge computing sheds light on the issue by pushing the data processing from the remote network core to the local network edge, remarkably reducing the latency and improving the efficiency. Besides, the recent breakthroughs in deep learning have greatly facilitated the data processing capacity, enabling a thrilling development of novel applications, such as video surveillance and autonomous driving.

D. L. Dutta and S. Bharali, ''Tiny ML meets IoT: A comprehensive survey,'' Internet Things, vol. 16, Dec. 2021, Art. no. 100461: The rapid growth in miniaturization of low-power embedded devices and advancement in the optimization of machine learning (ML) algorithms have opened up a new prospect of the Internet of Things (IoT), tiny machine learning (Tiny ML), which calls for implementing the ML algorithm within the IoT device. Tiny ML framework in IoT is aimed to provide low latency, effective bandwidth utilization, strengthen data safety, enhance privacy, and reduce cost. Its ability to empower the IoT device to reliably function without consistent access to the cloud services while delivering accurate ML services makes it a promising option for IoT applications seeking cost-effective solutions. Especially in settings where inadequate connectivity is common, Tiny ML aims to provide on-premise analytics which will add substantial benefit to IoT services. In this article, we introduce the definition of Tiny ML and provide background information on diverse related technologies stating their strengths and weaknesses. We then show how Tiny ML-as-a-service is implemented through efficient hardware-software co-design. This article also introduces the role of 5G in the Tiny ML-IoT scenario. Furthermore, it touches on the recent progress in Tiny ML research in both academia and industry along with future challenges and opportunities. We believe that this review will serve as an information cornerstone for the IoT research community and pave the way for further research in this direction.

P. P. Ray, ''A review on Tiny ML: State-of-the-art and prospects,'' J. KingSaud Univ. Comput. Inf. Sci., vol. 34, no. 4, pp. 1595–1623, Apr. 2022: Machine learning has become an indispensable part of the existing technological domain. Edge computing and the Internet of Things (IoT) together present a new opportunity to imply machine learning techniques at the resource-constrained embedded devices at the edge of the network. Conventional machine learning requires an enormous amount of power to predict a scenario. Embedded machine learning – The tiny ML paradigm aims to shift such a plethora from traditional high-end systems to low-end clients. Several challenges are paved while doing such a transition such as maintaining the accuracy of learning models, providing a train-to-deploy facility in resource-frugal tiny edge devices, optimizing processing capacity, and improving reliability. In this paper, we present an intuitive review of such possibilities for Tiny ML. We first, present the background of Tiny ML. Secondly, we list the tool sets for supporting Tiny ML. Thirdly, we present key enablers for the improvement of Tiny ML systems. Fourthly, we present state-of-the-art frameworks for Tiny ML. Finally, we identify key challenges and prescribe a future roadmap for mitigating several research issues of Tiny ML.

IV. Methodology

To develop an AI-based Extreme-Edge TCN-Based Low-Latency Collision-Avoidance Safety System, a systematic approach is necessary to ensure efficiency, real-time responsiveness, and safety in industrial environments. The methodology follows a structured framework encompassing system architecture, sensor integration, AI model development, edge computing deployment, testing, and optimization to enhance collision avoidance capabilities.

System Architecture Design

The system architecture is designed to integrate AI algorithms, real-time sensors, and edge computing to detect and prevent collisions. The essential components of the system include:

1. Embedded Microcontroller: A low-power AI-compatible microcontroller (e.g., STM32, ESP32, or an ARM Cortex-based MCU) to perform AI inference at the edge.
2. Sensors: A combination of LIDAR, ultrasonic sensors, infrared sensors, and cameras to detect objects, movements, and distances in industrial environments.
3. Actuators and Alarms: Emergency braking mechanisms, visual indicators, and sound alarms that activate in case of a predicted collision.

To enhance the accuracy and reliability of collision detection, **multi-sensor fusion** is implemented. Each sensor provides a different perspective of the environment, improving the system's ability to detect obstacles effectively. The steps involved include:

1. **Data Collection:** Continuous acquisition of environmental data from multiple sensors.
2. **Noise Reduction:** Filtering out erroneous readings using signal-processing techniques.
3. **Data Synchronization:** Aligning sensor inputs in a unified format for accurate analysis.
4. **Feature Extraction:** Identifying critical parameters such as object proximity, velocity, and movement direction.

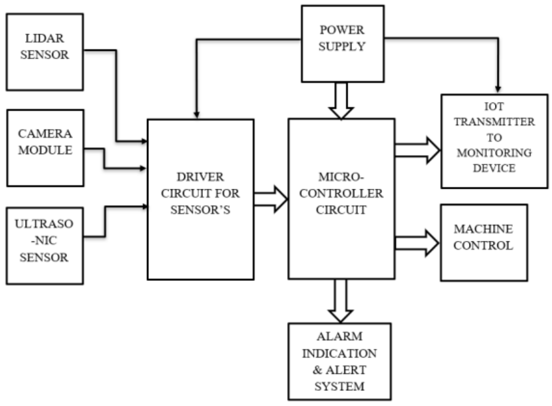

Fig 3. Block Diagram of collision avoidance system

AI Model Development Using Temporal Convolutional Networks (TCN): The AI model is developed using a Temporal Convolutional Network (TCN), which processes time-series sensor data to predict potential collisions. TCN is preferred over conventional Recurrent Neural Networks (RNNs) or Long Short-Term Memory (LSTM) models due to its advantages in low-latency processing, ability to capture long-range dependencies, and efficiency in time-series prediction.

AI Model Development Steps:

Dataset Collection: Gathering real-world collision scenarios using industrial sensor data. **Preprocessing:** Normalizing sensor values, eliminating outliers, and labeling collision events. **Feature Engineering:** Extracting key indicators like object trajectory, acceleration, and risk factors. **Model Training:** Training the TCN model with **supervised learning** techniques on time-series data. **Optimization:** Applying **quantization and pruning** to minimize the model size for embedded system deployment.

Edge Computing Implementation

To ensure ultra-low latency, the AI model is deployed on an **edge computing device**, enabling real-time processing without reliance on external servers. This step ensures: **Faster decision-making:** AI inference is executed directly on the microcontroller, avoiding delays from cloud-based processing. **Energy efficiency:** Optimized algorithms reduce power consumption for prolonged operation. **Offline functionality:** The system remains fully operational even in network-constrained environments.

Collision Prediction and Preventive Actions

Once deployed, the system operates in real-time, continuously monitoring the surroundings and predicting possible collisions. The execution process follows these steps: **Real-Time Data Acquisition:** Sensors collect and transmit data at high frequency. **Data Processing:** The microcontroller preprocesses sensor inputs and runs AI inference. **Collision Prediction:** The AI model assesses time-series data and identifies potential obstacles. **Preventive Actions:** If a collision is predicted, the system triggers: **Emergency braking mechanisms** for machinery. **Visual and audio alarms** to alert nearby personnel. **Vibration feedback systems** for operator notification.

System Testing and Validation

To evaluate the system's performance, rigorous testing is conducted in both simulated and real-world industrial environments. The testing process includes: **Accuracy Testing:** Comparing AI predictions with actual obstacle occurrences. **Response Time Measurement:** Assessing the latency from detection to preventive action. **Robustness Evaluation:** Testing performance under varying environmental conditions (e.g., lighting changes, sensor interference). **Power Consumption Analysis:** Ensuring energy efficiency for long-term industrial usage. Hardware-in-the-loop (HIL) simulations and **real-time industrial field trials** validate the system's effectiveness in preventing collisions.

Block diagram explanation:

1. **LIDAR Sensor, Camera Module, and Ultrasonic Sensor:**

The LIDAR Sensor, Camera Module, and Ultrasonic Sensor play a crucial role in detecting obstacles and monitoring the environment in real time. The LIDAR sensor provides precise distance measurements, the camera module captures visual data, and the ultrasonic sensor detects nearby objects. These sensors collect environmental data and send it as raw signals to the Driver Circuit for Sensors for further processing.

2. **Driver Circuit for Sensors:**

The **Driver Circuit for Sensors** is an intermediary between the sensors and the microcontroller. It processes the raw signals from the LIDAR, camera, and ultrasonic sensors, ensuring proper voltage levels and signal conditioning. Once processed, the refined sensor data is sent to the **Microcontroller Circuit**, which serves as the system's central processing unit.

3. **Microcontroller Circuit:**

The Microcontroller Circuit is responsible for analyzing sensor data and making real-time decisions to prevent collisions. It receives inputs from the driver circuit and determines whether an obstacle is present. Based on the detected risk, the microcontroller sends control signals to the Machine Control system to adjust movement and avoid accidents. Additionally, it activates the Alarm Indication & Alert System to warn nearby workers and transmits real-time data to the IoT Transmitter for remote monitoring.

4. **Power Supply:**

The Power Supply is essential for the proper functioning of the entire system. It provides electrical energy to all components, including the sensors, driver circuit, microcontroller, alarm system, and IoT transmitter. This ensures stable operation and reliable performance. It is battery operated; a 12V dc power supply is used here to provide power to the entire circuit.

5. **Alarm Indication & Alert System:**

The Alarm Indication & Alert System is designed to notify workers of potential hazards. It receives activation signals from the microcontroller whenever an obstacle is detected. Upon receiving the signal, the system triggers warning mechanisms such as buzzers, flashing lights, or voice alerts to ensure worker safety.

6. **IoT Transmitter to Monitoring Device:**

The IoT Transmitter to Monitoring Device enables remote supervision and real-time data transmission. It receives processed information from the microcontroller and transmits it wirelessly to a cloud-based dashboard, computer, or mobile application. This allows industrial supervisors to monitor machine operations and safety status remotely.

7. **Machine Control:**

The Machine Control system is responsible for adjusting the machine's movements in response to potential collision risks. It receives control signals from the microcontroller and executes corrective actions, such as stopping, slowing down, or changing direction. This ensures that the machine operates safely without causing accidents.

Role of each block:

1. LIDAR Sensor, Camera Module, Ultrasonic Sensor

These sensors collect environmental data, detecting obstacles, distance, and movement. The output of sensor signals are sent to the Driver Circuit for Sensors for processing.

2. Driver Circuit for Sensors

It receives raw signals from LIDAR, Camera Module, and Sensor. Then, it sends the signals to the processor circuit and conditions them before sending them to the microcontroller circuit.

3. Microcontroller Circuit

The microcontroller receives processed sensor data from the driver circuit and analyzes sensor data using algorithms (TCN model) to detect potential collisions. The Microcontroller sends signals to the Machine Control for corrective actions. Activates Alarm Indication & Alert System for warnings. Transmits safety data to the IoT Transmitter to the Monitoring Device.

4. Power Supply

Takes a 12V DC power source. Supplies power to all system components, ensuring stable operation.

5. Alarm Indication & Alert System

Receives activation signals from the Microcontroller Circuit when a collision risk is detected. Then it triggers warnings (buzzers, flashing lights, or voice alerts) to notify nearby workers**.**

6. IoT Transmitter to Monitoring Device

It receives processed safety data from the Microcontroller Circuit. And transmits real-time data wirelessly to remote dashboards, mobile apps, or cloud-based monitoring platforms.

7. Machine Control

Receives control signals from the Microcontroller Circuit in response to potential collisions. Executes corrective actions (stopping, slowing down, or changing direction) to avoid accidents.

Let's take an example for the safety System for Crane Operations:-

cranes play a crucial role in the heavy equipment and construction industry, enabling the transportation of heavy materials across worksites. However, crane operations come with several safety challenges, including the risk of collisions with structures, equipment, and workers. One of the most significant hazards is the presence of workers in the crane's operational area, which can lead to severe accidents. Additionally, cranes often operate in constrained spaces where poor visibility, blind spots, and high wind conditions make it difficult for operators to maneuver safely. Load swinging due to sudden movements or external factors such as wind can also destabilize lifting operations, increasing the risk of dropping materials. Delayed response times in traditional crane operations further increase the chances of accidents, as manual control relies on human reflexes, which may not always be fast enough to prevent collisions.

To address these challenges, an AI-based low-latency collision-avoidance system is integrated into crane operations, utilizing a combination of sensors such as LIDAR, AI-powered cameras, ultrasonic sensors, infrared sensors, and accelerometers. The system continuously monitors the crane's surroundings and detects obstacles in real-time. If a worker is detected in the crane's movement path, AI vision and LIDAR immediately trigger an emergency stop, preventing accidents. If an obstacle such as a nearby building or equipment is detected, the system calculates a new path and adjusts crane movement accordingly. For load stabilization, inertial sensors and accelerometers detect excessive swinging and automatically adjust the crane's speed to ensure balance. In cases of poor visibility due to fog, dust, or nighttime operations, thermal cameras enhance object detection, allowing the system to alert operators about potential hazards.

The output of this intelligent system includes multiple preventive actions to enhance safety and efficiency. If an obstacle or worker is detected, the crane's movement is automatically halted to prevent accidents. In cases where excessive load swinging occurs, the system dynamically adjusts crane speed to stabilize the load, reducing the risk of material loss. Additionally, the system issues real-time audio-visual alerts to the operator if an obstruction is detected, allowing for quick decision-making. When a potential collision is identified, AI algorithms calculate an alternative movement path, ensuring smooth and safe operations. All detected incidents are logged into a cloud-based platform for continuous monitoring and safety improvements.

By implementing this AI-based collision-avoidance system, crane operations become significantly safer and more efficient. The system enhances worker safety by preventing fatal accidents, reduces downtime by avoiding machinery damage, and ensures precise load control for stable lifting operations. Furthermore, the optimization of crane movement improves operational efficiency while maintaining compliance with strict safety regulations. This advanced technology transforms traditional crane operations into a safer, more reliable, and intelligent process, reducing risks and increasing overall productivity in the construction and heavy equipment industry.

SIMULATION OF THE CIRCUIT:

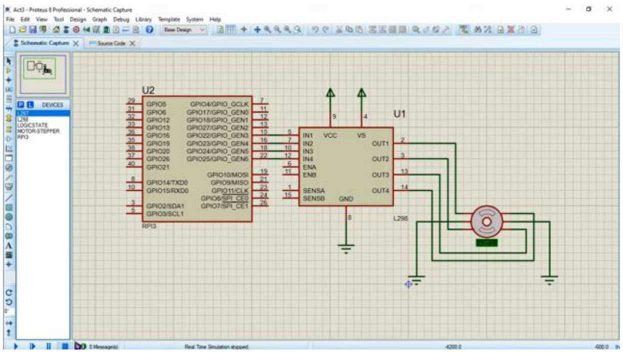

Fig 4. Machine control i.e. motor driving simulation

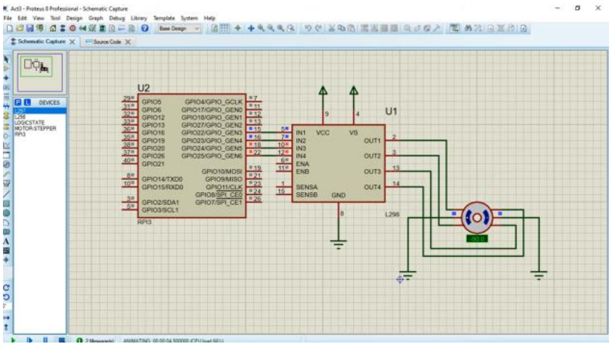

Fig 5. Machine control simulation

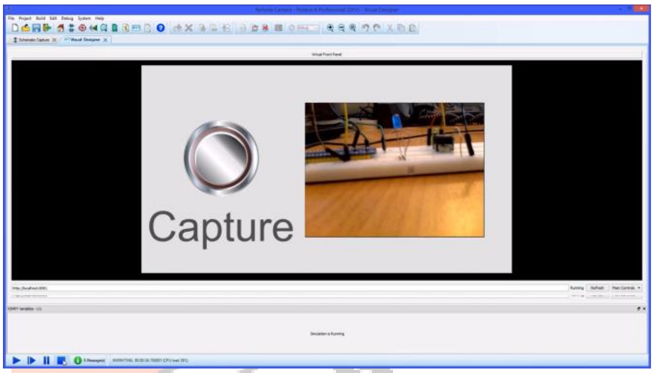

Fig 6. Camera detection simulation

V. CONCLUSION

The AI-based Extreme-Edge TCN Low-Latency Collision-Avoidance System effectively enhances industrial safety by integrating real-time sensor fusion, AI-driven time-series analysis, and edge computing. The system ensures ultra-fast collision prediction and response, minimizing accidents and improving operational efficiency. By leveraging low-power microcontrollers and optimized AI models, it operates with high accuracy and reliability. Continuous monitoring and adaptive learning further enhance its effectiveness, making it a robust and scalable solution for industrial environments.

VI. REFERENCES

1.Bai, S., Kolter, J. Z., & Koltun, V. (2018). An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling. arrive preprint arXiv:1803.01271. https://doi.org/10.48550/arXiv.1803.01271

2.Chen, Z., & Hussein, A. E. (2017). Sensor Fusion Techniques for Autonomous Vehicles. IEEE International Conference on Intelligent Transportation Systems, 1825-1830. https://doi.org/10.1109/ITSC.2017.8317924

3. Lea, C., Flynn, M. D., Vidal, R., Reiter, A., & Hager, G. D. (2017). Temporal Convolutional Networks for Action Segmentation and Detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 156-165. https://doi.org/10.1109/CVPR.2017.935

4.Shi, W., Cao, J., Zhang, Q., Li, Y., & Xu, L. (2016). Edge Computing: Vision and Challenges. IEEE Internet of Things Journal, 3(5), 637-646. https://doi.org/10.1109/JIOT.2016.2579198.
5.Kajita, S., Hirukawa, H., Harada, K., & Yokoi, K. (2015). Collision Avoidance in Industrial Robots Using Multiple Sensors. IEEE Robotics and Automation Magazine, 22(2), 34-43. https://doi.org/10.1109/MRA.2015.2414495
6.Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., & Mane, D. (2016). Concrete Problems in AI Safety. arXiv preprint arXiv:1606.06565. https://doi.org/10.48550/arXiv.1606.06565
7.Wang, X., Han, Y., Leung, V. C., Niyato, D., Yan, X., & Chen, X. (2020). Convergence of Edge Computing and Deep Learning: A Comprehensive Survey. IEEE Communications Surveys & Tutorials, 22(2), 869-904. https://doi.org/10.1109/COMST.2020.2970550
8. Siemens. (2019). Machine Safety and Collision Avoidance: Best Practices in Industrial Safety Systems. Siemens White Paper. Available at https://new.siemens.com/global/en.html