**A**

**PROJECT REPORT**

**ON**

**“<<Bullet Detection Project>>”**

**SUBMITTED TO**

**SHIVAJI UNIVERSITY, KOLHAPUR**

**IN THE PARTIAL FULFILLMENT OF THE REQUIREMENT**

**FOR THE AWARD OF DEGREE**

**BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND ENGINEERING**

**SUBMITTED BY**

|  |  |  |
| --- | --- | --- |
| **MR.** | **Harshal Makarand Mali** | **22UAD038** |
| **MR.** | **Karan Mahendra Nagure** | **22UAD046** |
| **MR.** | **Harshvardhan Pratap Londhe** | **22UAD036** |
|  |  |  |

**UNDER THE GUIDANCE OF**

**Mr. S. P. Pise**

 



**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE ENGINEERING**

**DKTE SOCIETY’S TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI**

**(AN EMPOWERED AUTONOUMOUS INSTITUTE)**

**2024-2025**

**D.K.T.E. SOCIETY’S**

**TEXTILE AND ENGINEERING INSTITUTE, ICHALKARANJI**

**(AN EMPOWERED AUTONOUMOUS INSTITUTE)**

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE ENGINEERING**



**CERTIFICATE**

**This is to certify that, project work entitled**

**“<<Bullet Detection Project >>”**

**is a bonafide record of project work carried out in this college by**

|  |  |  |
| --- | --- | --- |
| **MR.** | **Harshal Makarand Mali** | **22UAD038** |
| **MR.** | **Karan Mahendra Nagure** | **22UAD046** |
| **MR.** | **Harshvardhan Pratap Londhe** | **22UAD036** |
|  |  |  |

**is in the partial fulfillment of award of degree Bachelor of Technology in Artificial Intelligence and Data Science Engineering prescribed by Shivaji University, Kolhapur for the academic year 2024-2025.**

**MR. S. P. PISE**

**(PROJECT GUIDE)**

**PROF. (DR.) T. I. BAGBAN PROF.(DR.) L.S.ADMUTHE**

**(HOD AI & DS DEPT.) (DIRECTOR)**

**EXAMINER: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**DECLARATION**

We hereby declare that, the project work report entitled “<<project title>>” which is being submitted to D.K.T.E. Society’s Textile and Engineering Institute Ichalkaranji, affiliated to Shivaji University, Kolhapur is in partial fulfillment of degree B.Tech.(AI & DS). It is a bonafide report of the work carried out by us. The material contained in this report has not been submitted to any university or institution for the award of any degree. Further, we declare that we have not violated any of the provisions under Copyright and Piracy / Cyber / IPR Act amended from time to time.

|  |  |  |  |
| --- | --- | --- | --- |
| **Title** | **Name of the Student** | **PRN** | **Signature** |
| MR. | Harshal Makarand Mali | 22UAD038 |  |
| MR. | Karan Mahendra Nagure | 22UAD046 |  |
| MR. | Harshvardhan Pratap Londhe | 22UAD036 |  |
|  |  |  |  |

**ACKNOWLEDGEMENT**

With great pleasure we wish to express our deep sense of gratitude to Mr. S. P. Pise for his valuable guidance, support, and encouragement in the completion of this project report.

Also, we would like to take the opportunity to thank our head of department Dr. T. I. Bagban for his cooperation in preparing this project report.

We feel gratified to record our cordial thanks to other staff members of the Artificial Intelligence and Data Science Department for their support, help, and assistance which they extended as and when required.

Thank you,

|  |  |  |
| --- | --- | --- |
| **Title** | **Name of the Student** | **PRN** |
| MR. | Harshal Makarand Mali | 22UAD038 |
| MR. | Karan Mahendra Nagure | 22UAD046 |
| MR. | Harshvardhan Pratap Londhe | 22UAD036 |
|  |  |  |

**ABSTRACT**

The Bullet Detection Project leverages the power of computer vision and artificial intelligence to detect bullets in video footage and real-time camera feeds. Using the YOLO (You Only Look Once) object detection model, we have trained the system on a large dataset of bullet visuals to recognize and track bullet trajectories accurately. The system also incorporates speed and direction analysis from real-time camera input. To enhance usability, we developed a frontend interface that allows users to upload videos, which are stored in a backend database. After processing, the system displays before and after detection results for visualization and analysis.

**INDEX**

1. **Introduction 1** 
   1. Problem definition 1
   2. Aim and objective of the project 1
   3. Scope and limitation of the project 1
2. **Background study and literature overview 2**
   1. Literature overview 2
   2. Investigation of current project and related work 2
3. **Requirement analysis 3**

a. Requirement Gathering 3

b. Requirement Specification 3

c. Use case Diagram 3

1. **System design 4**
   1. Architectural Design 4
   2. Flow Chart 4
   3. System Modeling 4
      1. Dataflow Diagram 4
2. **Implementation 5**

a. Agile Methodologies 5

b. Development Model 5

1. **Future Scope 5**
2. **References (Public repository GitHub source code links) 5**

**1. Introduction**

The ever-growing demand for public safety has increased the need for intelligent surveillance systems. Detecting projectiles such as bullets in high-risk environments can be critical for real-time alerts and forensic analysis. This project aims to automate bullet detection using a robust machine learning approach with real-time capabilities and a user-friendly interface.

1. **Problem definition**

Current surveillance systems lack the ability to detect fast-moving projectiles like bullets in real-time. Manual detection is inefficient and error-prone. There is a need for an automated, real-time system to detect, track, and analyze bullet movement for forensic, military, and safety applications.

1. **Aim and objective of the project**

* To build an automated system for bullet detection in videos and real-time feeds.
* To integrate YOLO-based object detection for fast and accurate recognition.
* To calculate the direction and speed of the detected bullet.
* To design a frontend platform for video upload and result visualization.
* To store video data and results in a database for further analysis.

1. **Scope and limitation of the project**

**Scope:**

* Real-time bullet detection using live camera feeds.
* Video-based analysis with upload and output features.
* Usability in security, defense, and forensic fields.

**Limitations:**

* Accuracy depends on camera resolution and lighting.
* Limited to bullet types and angles trained during dataset preparation.

**2. Background study and literature overview**

1. **Literature overview**

Object detection using deep learning has advanced significantly in recent years, primarily through the use of convolutional neural networks (CNNs). Among the most prominent object detection algorithms are YOLO (You Only Look Once), SSD (Single Shot Detector), and Faster R-CNN. YOLO is known for its ability to process images in real time with a high level of accuracy, making it highly suitable for applications like surveillance, autonomous driving, and smart traffic systems.

YOLO works by dividing the image into a grid and predicting bounding boxes and class probabilities for each grid cell. This single-shot approach offers a remarkable balance between speed and precision, making it ideal for detecting fast-moving objects such as bullets. In contrast, SSD and Faster R-CNN offer slightly better accuracy in some scenarios but at the cost of slower processing speeds.

1. **Investigation of current project and related work**

In the domain of video surveillance and threat detection, several projects have addressed broader challenges such as detecting violent behavior, weapon presence, or unauthorized entry. Technologies like facial recognition, object detection, and motion tracking are increasingly applied in security systems. However, a critical shortcoming of these implementations is the limited capacity to detect fast, small, and dynamic objects such as bullets. Bullet detection, due to its complexity and the high velocity involved, has largely remained unaddressed.

Our investigation reveals that most existing systems rely on static camera footage analysis or event-triggered recording, which often fails to provide real-time data essential for high-risk environments. Moreover, general-purpose object detection frameworks are not sufficiently trained to handle such specialized scenarios without customized datasets and refined models.

**3. Requirement analysis**

1. **Requirement Gathering**

* Functional Requirements:
* Upload video from frontend.
* Detect bullets using YOLO.
* Display detection result with bullet speed and direction.
* Non-Functional Requirements:
* Real-time processing capability.
* Efficient storage and retrieval system.
* User-friendly frontend.

1. **Requirement Specification**

* Language: Python, JavaScript
* Frameworks: Flask, YOLOv8, OpenCV
* Database: MySQL or MongoDB
* Frontend: HTML, CSS, JavaScript

1. **Use case Diagram**

**Actors:**

* User
* System

**Use Cases:**

* Upload video
* Detect bullet
* Store data
* Display results

**4. System design**

1. **Architectural Design**

* User uploads video via frontend
* Video is sent to backend (Flask server)
* YOLO model processes video for detection
* Result (video with detection) is stored and returned to frontend
* Before and after visuals are displayed.

1. **Flow Chart**

Start → Video Upload → Detection Process → Speed/Direction Calculation → Store Result → Display Output → End.

1. **System Modeling**

* **Dataflow Diagram**

1. User submits video.
2. System receives video.
3. YOLO model detects bullet.
4. System calculates bullet speed and direction.
5. Store and display result.

**5. Implementation**

1. **Agile Methodologies**

* Weekly sprints with iterative development.
* Frequent testing and debugging.
* Regular feedback sessions.

1. **Development Model**

* Waterfall + Agile hybrid: Initial planning followed by iterative development.

**6. Future Scope**

* Integrate GPS tracking for geolocation tagging.
* Enable multi-camera setup for 3D trajectory analysis.
* Incorporate deepfake detection to prevent tampered inputs.
* Support alert system for real-time notification.

**7. References (public repository GitHub source code links)**

* GitHub: https://github.com/7HarshvardhanLondhe/Bullet-Detection-Project
* YOLOv8 Documentation: https://docs.ultralytics.com/
* OpenCV Docs: https://docs.opencv.org/
* Flask Docs: https://flask.palletsprojects.com/