VisionLock: Military Grade Single Object Tracking System

Technical Report

Authors: - Rayyan Ahmed: 22F-BSAI-11 - Agha Harris: 22F-BSAI-27 - Irtat Mobin: 22F-BSAI-29 - Omaid Ejaz: 22F-BSAI-45 - Wajhi Qureshi: 22F-BSAI-50

Date: January 11, 2026

Abstract

VisionLock is a real-time computer vision system engineered for the precision tracking of military ground targets. Utilizing adaptive template matching with OpenCV, the system provides robust single-object tracking capabilities. It features a user-friendly interface built with Streamlit, allowing for video upload, manual initialization of a bounding box (Region of Interest - ROI), and visualization of target lock indicators. The system is designed to process video streams, identify and track designated targets, and export the processed output for analysis. This report details the system's architecture, technical implementation, key features, and specifications.

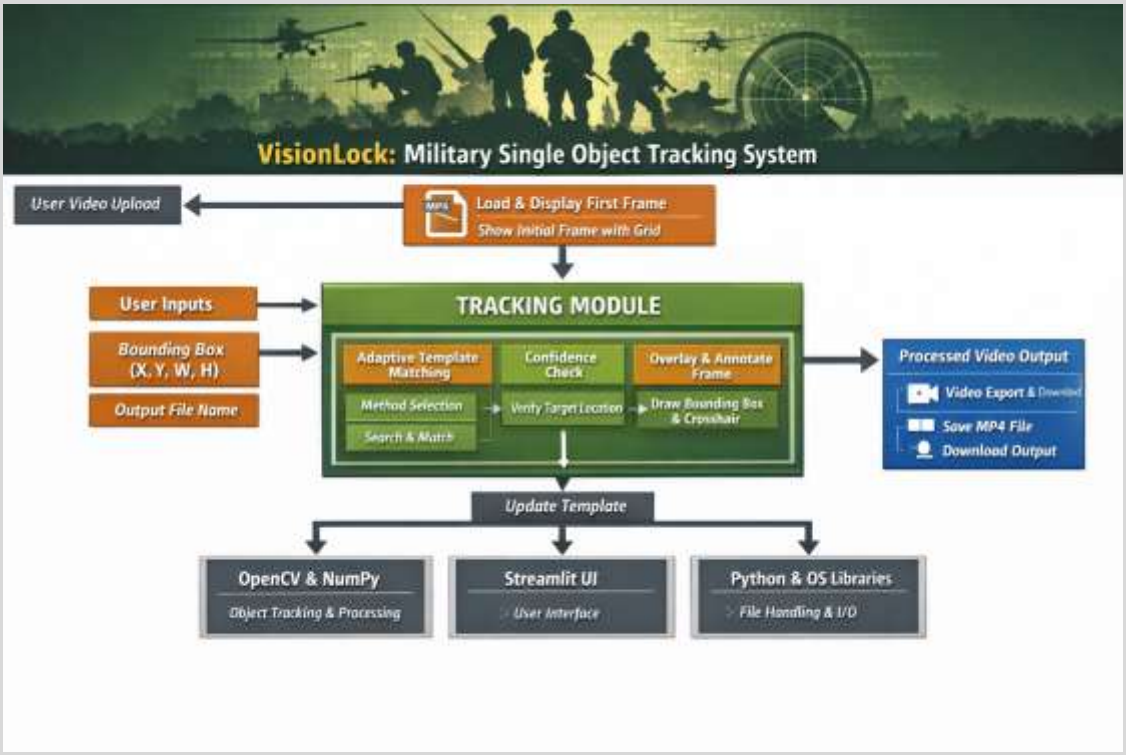
1. Introduction

Modern military operations demand highly accurate and reliable target tracking systems for surveillance, reconnaissance, and engagement. Traditional object tracking methods often struggle with variations in target appearance, lighting conditions, and occlusions in dynamic environments. VisionLock addresses these challenges by providing a robust, real-time single object tracking (SOT) solution specifically tailored for military ground targets.

This report provides a comprehensive overview of the VisionLock system, outlining its design principles, architectural components, and technical implementation. It further highlights the system's key features and advantages, demonstrating its potential as a valuable tool in military intelligence and operational support.

2. System Architecture

The VisionLock system is structured into several interconnected modules, as depicted in the architecture diagram below, to facilitate a streamlined object tracking workflow.



VisionLock System Architecture

Figure 1: VisionLock System Architecture

2.1. User Video Upload

The process begins with the user uploading a video file (e.g., MP4, AVI, MOV) to the system. This input serves as the primary data source for target tracking.

2.2. Initial Frame Processing

Upon video upload, the system loads and displays the first frame of the video. This initial frame is presented with a pixel grid to assist the user in accurately defining the initial bounding box for the target.

2.3. User Inputs

Before initiating the tracking process, the user provides critical inputs: - **Bounding Box (X, Y, W, H):** The coordinates (x, y) of the top-left corner, width (w), and height (h) of the initial target bounding box are manually specified by the user on the first frame. - **Output File Name:** The desired name for the processed output video file.

2.4. Tracking Module

The core of the VisionLock system is the **Tracking Module**, responsible for continuously identifying and tracking the designated object across subsequent video frames. This module comprises several sub-components:

2.4.1. Adaptive Template Matching

This sub-module employs template matching techniques to locate the target in each new frame. It includes: - **Method Selection:** Dynamically chooses between `cv2.TM_CCOEFF_NORMED` and `cv2.TM_SQDIFF_NORMED` based on the template's mean and standard deviation to optimize matching accuracy. - **Search & Match:** Performs the template matching operation within a defined search region of the current frame.

2.4.2. Confidence Check

After template matching, a confidence score is calculated to assess the accuracy of the target's location. This involves: - **Verify Target Location:** Based on the confidence score, the system determines if the tracked location is reliable. If confidence is low, the search expansion area is increased in subsequent frames to relocate the target.

2.4.3. Overlay & Annotate Frame

Once the target's location is verified, the system overlays visual indicators on the current frame: - **Draw Bounding Box & Crosshair:** A bounding box is drawn around the tracked target, along with a crosshair and an aim circle at its center, providing clear target lock indicators.

2.4.4. Update Template

To maintain tracking accuracy amidst changes in target appearance (e.g., rotation, scale, lighting), the template used for matching is periodically updated. This adaptive mechanism ensures robustness over time.

2.5. Processed Video Output

Upon completion of the tracking process, the system generates a processed video output, which includes: - **Video Export & Download:** The tracked video, with all annotations, is exported as an MP4 file. - **Save MP4 File:** The output video is saved to a temporary location. - **Download Output:** The user can download the final processed video file.

2.6. Underlying Technologies

The VisionLock system leverages several key technologies: - **OpenCV & NumPy:** Utilized for core object tracking and image processing functionalities, including adaptive template matching, frame manipulation, and numerical operations. - **Streamlit UI:** Provides the interactive web-based user interface for video upload, input specification, and result display. - **Python & OS Libraries:** Handle file management, input/output operations, and other system-level tasks.

3. Technical Implementation

3.1. Core Tracking Algorithm: Adaptive Template Matching

The VisionLock system employs an adaptive template matching algorithm to achieve robust single object tracking. The primary steps involve:

1. **Initial Template Extraction:** From the first frame, a grayscale template of the target is extracted based on the user-defined bounding box.
2. **Dynamic Method Selection:** To enhance matching performance, the system dynamically selects the template matching method (`cv2.TM_CCOEFF_NORMED` or `cv2.TM_SQDIFF_NORMED`) based on the statistical properties (mean and standard deviation) of the current template. If the standard deviation is low (indicating low contrast), `TM_CCOEFF_NORMED` is preferred; otherwise, `TM_SQDIFF_NORMED` is used. Additionally, if the mean intensity is high, the template is inverted to improve matching in certain scenarios.
3. **Search Region Definition:** For each subsequent frame, a search region is defined around the previously tracked target location. This region is expanded or contracted based on the tracking confidence to balance computational efficiency and the ability to re-acquire lost targets.
4. **Template Matching:** The selected template matching method is applied within the search region to find the best match for the target.
5. **Confidence Evaluation:** A confidence score is derived from the matching result (e.g., `max_val` for `TM_CCOEFF_NORMED`, `1.0 - min_val` for `TM_SQDIFF_NORMED`). If the confidence falls below a predefined threshold (e.g., 0.55), it indicates a potential loss of track, and the search expansion is increased for the next frame.
6. **Template Update:** To adapt to changes in the target's appearance, the template is periodically updated (e.g., every 10 frames) using the newly tracked region of the target. This ensures that the system remains robust against variations in scale, rotation, and lighting conditions.

3.2. User Interface with Streamlit

The user interface, developed using Streamlit, provides an intuitive platform for interacting with the VisionLock system:

- **Video Upload:** Users can easily upload video files in supported formats (MP4, AVI, MOV).
- **First Frame Reference:** The first frame of the uploaded video is displayed with a pixel grid, aiding users in precisely defining the initial bounding box for the target.
- **Bounding Box Input:** Numerical input fields are provided for `x`, `y`, `width`, and `height` to specify the initial ROI.
- **Output File Naming:** Users can specify a custom name for the output video file.
- **Tracking Initiation:** A dedicated button (`▶ Start Tracking`) initiates the object tracking process.
- **Progress Indicator:** A spinner provides visual feedback during video processing.
- **Download Processed Video:** Upon completion, a download button allows users to retrieve the tracked video.

3.3. Video Processing and Annotation

During the tracking process, each frame is annotated with visual indicators:

- **Bounding Box:** A red rectangle outlines the tracked target.
- **Crosshair and Aim Circle:** A crosshair and a concentric circle are drawn at the center of the bounding box, simulating a target lock mechanism.
- **Status Text:** Text overlays such as "Military Vehicle Targeted Successfully." and "FPV Drone AIM Locked." provide real-time feedback.
- **FPS Display:** The frames per second (FPS) of the processing is displayed, indicating the system's real-time performance.

4. Key Features and Advantages

VisionLock offers several key features and advantages that make it a robust solution for military-grade single object tracking:

- **Precision Tracking:** Designed for accurate tracking of military ground targets, even in challenging environments.
- **Adaptive Robustness:** The adaptive template matching algorithm, coupled with dynamic method selection and template updating, ensures resilience against changes in target appearance, scale, and lighting.
- **Real-time Performance:** Optimized for real-time video processing, crucial for operational scenarios.
- **Intuitive User Interface:** The Streamlit-based UI simplifies video input, ROI definition, and result retrieval.
- **Clear Target Lock Indicators:** Visual overlays provide immediate and unambiguous feedback on target acquisition and tracking status.
- **Exportable Output:** Processed videos can be easily exported for further analysis, debriefing, or integration into other systems.

5. Technical Specifications

Component	Specification
Programming Language	Python
Core Libraries	OpenCV (for computer vision and tracking) NumPy (for numerical operations)
User Interface	Streamlit
Input Video Formats	MP4, AVI, MOV
Output Video Format	MP4
Tracking Method	Adaptive Template Matching (TM_CCOEFF_NORMED, TM_SQDIFF_NORMED)
Operating System	Platform-independent (Python-based)

6. Conclusion

VisionLock represents a significant step towards developing effective and reliable single object tracking systems for military applications. By integrating adaptive template matching with a user-friendly interface, it provides a powerful tool for real-time surveillance and target acquisition. The system's ability to dynamically adjust to environmental changes and maintain tracking confidence underscores its potential for deployment in diverse operational contexts. Future enhancements could explore the integration of deep learning models for improved object re-identification and multi-object tracking capabilities.

Streamlit App Link:

https://visionlock---military-sot-system-11-27-30-45-50.streamlit.app/

Github Link:

https://github.com/CodingRayyan/VisionLock---Military-SOT-System