

Threat Sentinel AI: Real-Time Security Threat Detection and Alert System

¹Vaidya Akshaya, ²Mr.Y.Subramanyam,

¹M.Tech Scholar, Dept. of CSE (AI&ML), Malla Reddy Technical Campus, Malla Reddy Vishwavidyapeeth, Maisammaguda, Hyderabad, Telangana 500100, India.

Mail id: vaidyaakshaya04@gmail.com

²Assistant Professor, Dept. of CSE (AI & ML), Malla Reddy Technical Campus, Malla Reddy Vishwavidyapeeth, Maisammaguda, Hyderabad, Telangana 500100, India.

Mail id: subrahmanyamedida@gmail.com

Article Info

Received: 23-03-2026

Revised: 02-04-2026

Accepted: 10-04-2026

Published: 21-04-2026

ABSTRACT

To combine cyber and physical surveillance, Threat Sentinel GPT provides a real-time security architecture powered by artificial intelligence. Delays in threat identification due to human error, siloed security technologies, and inefficient analysis of continuous CCTV feeds and system records are being tackled by this initiative. There is an increase in danger and harm because traditional security systems do not react quickly enough to physical incursions or cyberattacks. Computer vision and natural language processing are two examples of the sophisticated technological and algorithmic models used by the suggested system to analyze live camera feeds and system logs, respectively, in order to fix these difficulties. To find intruders, weapons, and suspicious behavior, object detection algorithms are used, and cyber risks are categorized using log data by machine learning models. The system keeps a constant eye on both settings, finds patterns in the events it detects, and automatically ranks their severity. With little to no human input required, real-time warnings are produced, allowing for more precise security responses and enhanced safety in general.

PROBLEM STATEMENT

Cyber and physical security threats have never been more intertwined for modern organizations. Ransomware, phishing, insider threats, and other cyberattacks are becoming smarter all the time and often work in tandem with physical breaches to do the most harm. Cybersecurity and physical security are two separate but related aspects of traditional security systems. Constant human attention is required by CCTV systems, which may lead to supervision and exhaustion. The massive amounts of log data produced by network monitoring technologies are either not evaluated at all or just reviewed on a periodic basis. Delays in detection, more false positives, and ineffective responses are the outcomes of this disjointed strategy, which leaves businesses open to assaults that might have been averted with quicker detection. Cyber and physical monitoring systems do not work together, which leads to gaps in coverage. Take the hypothetical case of an attacker who manages to breach a secure facility while also compromising any devices or networks that are linked to it. Traditional methods overlook the interconnected nature of these occurrences, handling them as separate episodes rather than a planned assault. This restriction hinders security teams' ability to comprehend the

whole extent and gravity of the problem, which in turn delays threat detection. Contemporary security threats are too complex and too fast for manual monitoring and reactive response processes to handle. In addition, massive amounts of data may be quite overwhelming for human operators. There are a lot of false positives produced by alerting mechanisms, a lot of logs from network monitoring systems, and constant video feeds from CCTV cameras. Being through all of these streams by hand is a waste of time and might lead to important dangers being unnoticed. Serious issues including data breaches, financial losses, bodily harm, or compromised staff safety might result from false negatives in detection. An immediate need exists for smart systems that can analyze data in real-time, integrate several modalities, and automatically prioritize security events.

OBJECTIVES OF THE PROJECT

A complete AI-powered security framework that can identify, analyze, and respond to threats in real time across cyber and physical domains is the primary goal of ThreatSentinelGPT. Here are the particular goals: 1. Real-time Monitoring: Set up round-the-clock monitoring of live CCTV feeds and network activity

records so that abnormalities may be detected automatically.

2. Embrace multi-modal integration: Merge cyber and physical data sources for a holistic view of security occurrences.

3. Developing machine learning models to categorize occurrences according to kind, severity, and probable effect is the third step in advanced threat classification. Fourthly, automated alerting allows security teams to respond quickly by generating prioritized alarms in real time.

5. Use computer vision for object detection to spot weapons, suspicious activity, unattended items, and unwanted entry in monitored locations.

6. Use AI models that can foresee security events using both historical and real-time data for predictive analytics.

7. Adaptive Learning: Improve system accuracy and resilience over time by dynamically updating models in response to new threats.

8. Visualization and Reporting: Make available dashboards that track connections between incidents, trends over time, and current occurrences.

9. Decrease Reliance on Humans: Eliminate Operator Overwork and Error by Automating Repetitive Monitoring and Prioritization Tasks.

10. Adhere to all industry rules and use encryption and access control to keep sensitive data safe.

Eleventh, Implement Proactive Security Management: Shift from Reactive Monitoring to Predictive Threat Mitigation Driven by Intelligence.

12. Supporting growth: Create a system that can include more sensors, handle massive amounts of data, and expand to cover more locations.

13. Make the most of your resources: Use predictive risk score to make the most effective use of your human and security resources.

SCOPE OF THE PROJECT

An comprehensive AI-based security platform is under the purview of Threat Sentinel GPT, which includes its design, development, testing, and deployment. In order to identify cybersecurity and physical security risks, the system keeps tabs on live CCTV feeds, system logs, network traffic, and access control data. Machine learning classifiers for threat prediction and severity assessment, natural language processing models for log analysis, and computer vision algorithms for object identification and anomalous behavior recognition based on deep learning are all part of the project. An interactive dashboard will be provided for monitoring, analysis, and incident management, and the system will work in real time, delivering alerts with minimum latency. All of your current security hardware, such as cameras, firewalls, IDS/IPS, and network monitoring software, will

function in tandem with Threat Sentinel GPT. In order to enhance detection accuracy when new danger patterns arise, the research also involves developing adaptive learning methods. An essential part of the project scope is testing and validation. This includes evaluating the system in situations such as noisy video, incomplete logs, many threats at once, and high-volume network traffic. The system is designed to safely handle data and comply with privacy regulations, all while being scalable to accommodate many locations or operations. The end objective is to provide a smart security framework that can foresee potential problems, take proactive measures to prevent them, and increase corporate safety while decreasing human effort and reducing risks from cyber and physical events.

INTRODUCTION

The current urban infrastructure cannot function without public safety measures. They are crucial for keeping social stabilization via safeguarding people during crises and sustaining vital services including healthcare, transportation, communication, and electricity delivery [1]. The proliferation of digital technologies and interconnected gadgets is rapidly improving the intelligence of these systems [2]. Unfortunately, cyber dangers are becoming more common as they improve in technology [3]. A public system's overall security may be severely compromised by a single, little flaw. There may be vehicular mayhem if an intruder compromised a traffic control system. A potential lag in the emergency response might be caused by malfunctioning communication equipment. Some instances may include the theft of confidential information or the shutdown of systems. The systems and the people who rely on them are both vulnerable to these threats, which are quite real. Preventing harm to the public requires knowledge of potential dangers. The term for this procedure is threat modeling [4]. Systematic analysis is examining a system's parts to identify potential failure points and then devising solutions to either eliminate or mitigate such issues. There are a number of well-known models that can be used for this purpose, including STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege) [5], MITRE ATT&CK [6], CVE (Common Vulnerabilities and Exposures) databases [7], NIST (National Institute of Standards and Technology) guidelines [8], and CISA (Cybersecurity and Infrastructure Security Agency) vulnerability reports [9]. Although these models provide powerful approaches, they may be difficult to use, particularly for those without cybersecurity expertise. The majority of threat modeling technologies available today are either too

complex, inflexible, or both. They aren't well-suited to various user or system types [10]. To implement the frameworks properly, many need in-depth understanding of security or physical labor. This could be very stressful for those who work as healthcare administrators, city planners, developers, or safety officers. An simpler, quicker, and more human-centered threat modeling tool is required.

With this goal in mind, we created ThreatGPT, an Agentic AI [11], [12] helper that is specifically tailored to aid individuals in doing threat analysis inside public safety systems. It generates structured threat models based on common frameworks, analyzes system components, and understands user input using the capabilities of large-scale language models. Based on the user-described approach, it offers both broad recommendations and specific, actionable insights. To provide a versatile, intelligent, and context-aware threat modeling experience, ThreatGPT combines Aggressive AI capabilities with well-established cybersecurity frameworks. As an interactive helper, ThreatGPT can grasp user-defined system components, guide the analysis process, and adapt to different levels of technical skill, unlike traditional solutions that just provide static checklists or demand expert-level knowledge. Structured threat assessments are generated in accordance with industry standard techniques by using big language models and few-shot learning [13]. To guarantee that ThreatGPT provides accurate, relevant, and actionable threat models for various public safety systems, it combines generative AI [14] with established cybersecurity expertise. Through the use of an intuitive conversation, users are able to participate in high-quality threat analysis. This allows stakeholders, from technical teams to public safety authorities, to proactively identify risks and increase system resilience. ThreatGPT democratizes access to this analysis. Agentic AI is designed to be more of a helpful colleague than a conventional AI that only follows orders. The ability to ask questions, advise users, and adapt to different contexts is a remarkable feat. An agentic AI collaborates with the user to solve problems as they arise, rather than waiting for flawless instructions. With ThreatGPT, the AI does more than simply respond to questions; it guides the user step-by-step through threat modeling, answering questions, and providing clarification when necessary. Particularly for those lacking in in-depth technical understanding, this makes the experience more intuitive and helpful. A public safety system's design, context, and risk profile are all distinct from one another. An intelligent traffic management system may be the subject of one user's description, while the IT network architecture of a hospital might be that of another [15]. It is unrealistic and restricts the

adaptability of traditional systems to predefine and hardcode threat modeling methodologies for each possible use case. The use of few-shot learning is crucial in dealing with this unpredictability. Without needing massive labelled datasets or intensive retraining, it allows the AI model to generalize from a limited selection of user-provided examples, adapting dynamically to new system descriptions and producing applicable threat models.

LITERATURE REVIEW

Several recent papers have used AI for threat analysis, and we will go over them here. In a world where risks are always developing, the necessity for proactive cyber-risk management is becoming more apparent, as Hamza *et al.* pointed out [22]. When it comes to modern, sophisticated assaults, old, ineffective solutions like rule-based and signature-based detection just won't cut it. They investigated the potential of Generative AI models like GANs and VAEs to mimic hitherto undiscovered cyberattack situations, allowing businesses to fine-tune their defenses. Better readiness against complex dangers, such advanced persistent threats (APTs), is made possible by moving away from reactive threat modeling and toward proactive threat modeling. Additionally, they highlighted the need of integrating AI-driven tactics with traditional security procedures to create cybersecurity systems that are both flexible and robust. In their proposal for a cyber threat detection system, Mohamed *et al.* [23] combined AI, NLP, and malware analysis. By using several analytical methods, their strategy sought to identify vulnerabilities such as the Follina vulnerability [24]. Even if AI and NLP work, the research found that unstructured data and heavy computing needs remain their Achilles' heels. To better understand the consequences and how to prevent them, Malatji *et al.* [25] looked at the many facets of cyberattacks powered by artificial intelligence. Strong protection mechanisms are emphasized in the research that examines the pros and cons of AI technology in cybersecurity. The rapid advancement of AI has been a major obstacle, necessitating ongoing adaption as new security measures are developed.

In their study, Roshanaei *et al.* [26] looked at ways to improve cyber security by using AI and ML. Integrating AI into security operations is crucial, as the research highlighted real-time threat detection and forecasting capabilities. But the authors stressed the legal and ethical issues with AI deployment, drawing attention to the dangers of relying too much on automated systems without sufficient human control. Protecting digital infrastructures was Schmitt's primary interest, and he worked on cyber threat

detection using AI [27]. To tackle the difficulties in implementing AI-enabled cybersecurity solutions, the study assessed ML classifiers for intrusion detection and anomaly-based malware. Although the research offered helpful information on how to incorporate AI into security systems, it also acknowledged the difficulty in adjusting these models to the ever-changing cyber threats and the need of ongoing learning processes. It is clear from the literature study that the majority of current research is centered on using machine learning models to identify and assess risks in different systems. We were unable to find any previous efforts that targeted the development of an AI agent for automated threat modeling. The goal of our work is to improve cybersecurity and guarantee public safety by introducing an agentic AI strategy that is specifically designed for threat modeling.

PROCEDURE

To help with intelligent, context-aware, and security-oriented query processing, we built an Agentic AI system in this study. The four main layers that make up our proposed Agentic AI system are the AI Agent Layer, the Knowledge Base/Training Dataset Layer, the Pretrained Large Language Model (LLM) Layer, and the Command-Line Interface (CLI) Layer. Figure 1 shows the workflow of the system.

Software & Hardware Requirements

Table 1. Hardware Requirements

Component	Specification
Processor	Intel Core i5 or above
RAM	8 GB (Minimum)
HardDisk	500 GB

Table 2. Software Requirements

Software Component	Specification
Operating System	Windows 10/Linux (Ubuntu)
Coding Language	Python
Deep Learning Framework	TensorFlow
Computer Vision Library	OpenCV
Development Environment	IDE/Anaconda/VS Code/Pycharm
Library	Nltk

IMPLEMENTATION

Security monitoring and threat detection are two of the many important functions performed by the several modules that make up the ThreatSentinelGPT system.

The Data Collection and Ingestion Module is the first component that collects data from many sources such as cyber logs, CCTV cameras, and physical sensors placed in locations that are under surveillance. You may see what people are up to online by looking at cyber logs, which contain things like firewall warnings, records of network activity, authentication attempts, and notifications from the system itself. Closed-circuit television (CCTV) feeds provide live video streams that record any motion, objects, or unusual physical activity. Data collected from sensors, such as those for temperature, motion, and door access, enriches the picture of the monitored environment. To make sure that the data is in sync and ready to be processed in the next modules, the ingestion module standardizes these diverse data streams.



Fig. 1. ThreatGPT System Workflow

Data Preprocessing and Feature Extraction is the second module that gets the raw data ready to be analyzed. In order to make computer vision algorithms more accurate, video frames are resized, normalized, and noise reduced. Frame differencing, edge detection, and background removal are some of the techniques used to isolate moving objects and eliminate noise. Using natural language processing (NLP) methods such as tokenization, stemming, and the elimination of unnecessary or duplicate items, cyber logs are processed and organized. Attributes including IP addresses, login frequency, mobility patterns, item size, and position coordinates are uncovered using feature extraction. The inputs are prepared for real-time AI/ML processing by this module, which guarantees that they are clean and organized. The system's vital component is the third module, AI/ML Threat Detection. Convolutional neural networks (CNNs) and other computer vision models identify objects, follow motion, and categorize suspicious actions in video frames. In order to identify patterns or anomalies in behavior across time, recurrent neural networks (RNNs) and long short-term memory networks (LSTMs) examine sequences of frames. Machine learning classifiers, such as deep neural networks, random forests, and gradient boosting, analyze log data to identify possible intrusions, suspicious access patterns, and abnormalities in cyber threat detection. Algorithms for natural language processing decipher hidden signs of bad conduct by extracting semantic meaning from textual input. With

confidence levels and possible severity ratings attached, this module generates preliminary threat notifications.

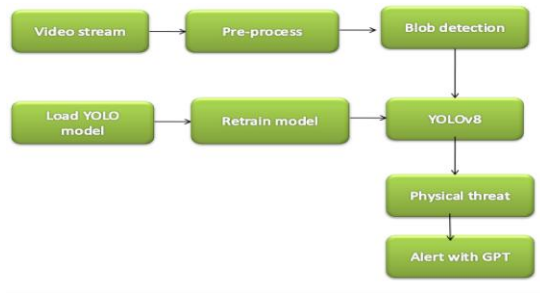


Fig 2: System Architecture

To find out whether there is a connection between cyber and physical occurrences, the fourth module, Threat Correlation Engine, merges the outputs from the AI/ML detection module. abnormalities seen in CCTV feeds are cross-referenced with suspected login attempts, abnormalities in system activity, or data from sensors. Such a situation would be considered linked and of high importance if, for example, an unlawful login to the network occurred at the same time as physical entry to a restricted location. The correlation engine prioritizes real dangers and decreases false positives by scoring occurrences according to their severity, frequency, and kind. This prioritizes alerting operators to significant issues so they can respond and mitigate them quickly. The generation and notification of alerts for security incidents in real-time to operators, SOC teams, or automated response systems is handled by the fifth module. All of the relevant details, such as the danger kind, impacted area, time, severity level, and suggested countermeasures, are included in the alerts. Notifications may be sent via several channels, such as email, SMS, push, and dashboard alerts. In addition, the system keeps track of each warning for auditing reasons, which allows for examination in retrospect and the ongoing enhancement of detection algorithms. Monitoring, trend analysis, and decision-making are made easier with the interactive interfaces provided by the sixth module, Visualization and Dashboard Integration. Live CCTV feeds, trends, historical data, and predicted insights may all be seen by security staff. Investigating occurrences in detail is made possible with features like filtering, magnification, and drill-down into events. All monitored facilities may be centrally managed thanks to the dashboard's integration with SOC tools. Thanks to the system's scalability, adding more cameras, sensors, and network nodes is a breeze. The modular architecture of

ThreatSentinelGPT makes it strong and flexible enough to handle enterprise-wide installations on a massive scale.

EXPERIMENT AND RESULTS

A local client configuration allowed access to the free version of the Google Gemini API [28], which was used to perform the tests. Powered by a Ryzen 7 5700U CPU with integrated graphics, the machine ran Ubuntu 20.04 LTS inside an environment. This system came with a 1 TB solid-state drive and 16 GB of random-access memory. We did not utilize any external GPU acceleration. The local computing resources were used for all processes on the client side, including preprocessing, quick building, and postprocessing. The Gemini model was still housed on Google's servers remotely, but API interactions and light processing duties were mostly handled by local hardware.

Figures 3 and 4 show the AI model's produced prompts and answer examples, respectively. Assets, Entry Points, Attacker Models, Threats and Vulnerabilities, and Mitigation Strategies are all elements of a threat model that the AI model arranges in a systematic way. The first step for ThreatGPT is to create a threat model. If it determines that the model needs to be refined, it proceeds to ask for clarification questions. The figures only display the crucial parts of the whole response for the sake of demonstration.

```

ThreatGPT: Ready to generate threat models. Write your system information. (Type 'exit' to quit.)
You: write a threat model for drone delivery system

ThreatGPT: Planning steps to generate the threat model...
1. Understand system components.
2. Identify entry points and assets.
3. Map STIGME threats to components.
4. Cross-reference with CVEs, NVD, NIST, MITRE ATT&CK, and CISA references.
5. Suggest mitigation strategies.
6. Review and refine the threat model if necessary.

Initial ThreatGPT output (after self-check):
Threat Model: Drone Delivery System

1. Assets
- Drone Hardware (airframe, motors, sensors, camera, payload bay)
- Flight Control System (autopilot, navigation, obstacle avoidance)
- Communication System (radio, cellular, satellite)
- Ground Control Station (GCS) (mission planning, monitoring, control)
- Regional Management System (package securing, delivery mechanics)
- Navigation Database (maps, waypoints, geographic information)
- User Accounts and Authentication System (operator, administrators)
- Order Management System (customer orders, delivery routing)
- Payment Processing System (credit card, mobile payments)
- Delivery Tracking System (real-time location, status updates)
- Data Storage and Logs (flight data, telemetry, images)
- APIs and Integrations (weather, traffic, customer services)

2. Entry Points
- Remote Controller Interface (GCS)
- Wireless Communication Link (Wi-Fi, cellular, satellite)
- Drone Sensors (camera feeds, LIDAR, GPS)
- Firmware Update Interfaces (OTA, USB)
- API endpoints for GCS, order management, and tracking
- Payload compartment access
- Network connections (GCS to internet, drone to network)
- Mobile application (for customer tracking or delivery confirmations)
  
```

Fig. 3. Initial Threat Model

With the use of three guidelines, we tested our concept. The name of the system and the names of the individual components make up the basic prompt. It facilitates the model's comprehension

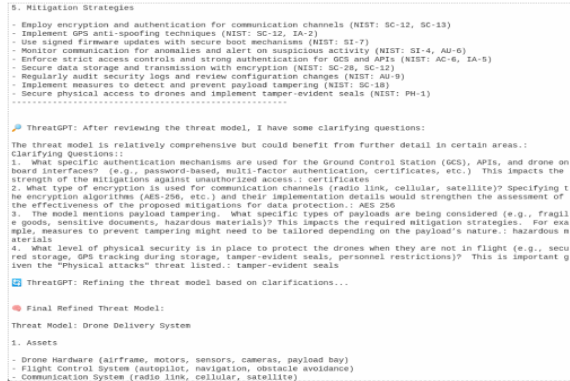


Fig. 4. Clarifying questions for refined threat model

communities depends on the prompt detection and elimination of dangers.

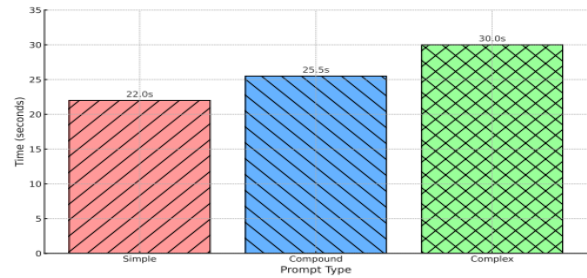
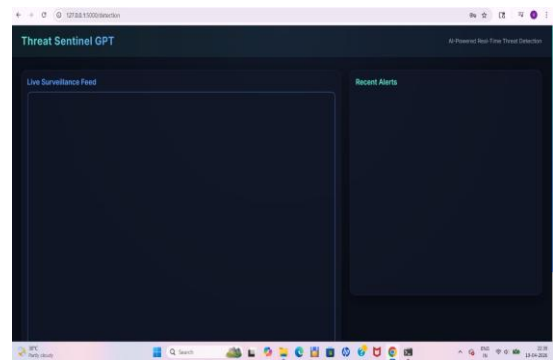
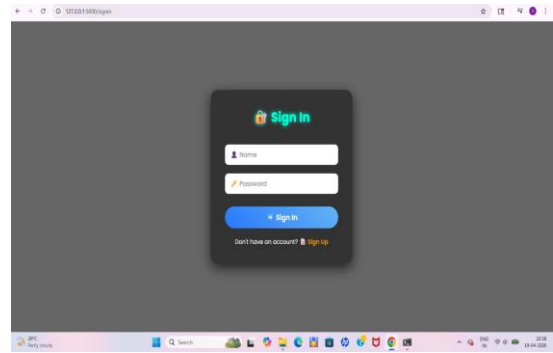
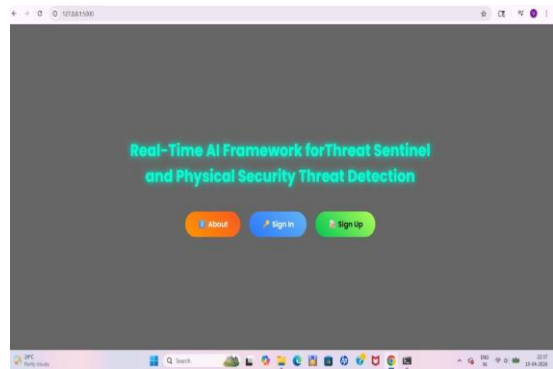
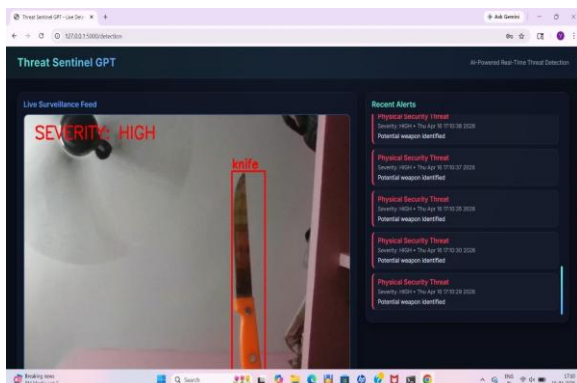


Fig. 5. Response time comparison

RESULTS

in order to construct the threat model. It is unnecessary for it to consider the parts. Names of several components and the system itself make up the compound prompt. The model must consider the specific parts and create a danger model for them. Lastly, the complicated question simply displays the name of the system. As a whole, the model considers all the parts and how they work together to form the threat model. The following are examples of prompts that were used in each category: Easier Exercise: Create a risk assessment for a drone delivery management system that uses a scheduling server and a mobile app to fulfill client requests. Drones with built-in GPS systems may carry out delivery jobs given to them by the server. A cloud-based dashboard allows for constant monitoring of operational data from drones, while the tracking module provides real-time updates on package progress. When dealing with the identification of assets, entry points, threats, vulnerabilities, attacker models, and mitigation, it is important to take into account the relationships and data flow between these components. Formulate a danger assessment for a drone dispatching system; this is a compound prompt. A client mobile app, a delivery scheduling server, a package tracking module, a drone navigation system, and a cloud-based monitoring dashboard are the major components. Difficult Question: In order to better manage drone deliveries, you need develop a threat model. Figure 5 shows how long each of these three prompts take. It is clear from the chart that the reaction time for each sort of prompt stays between twenty and thirty seconds. Our suggested model completes the work in 30 seconds, whereas experts in the field estimate that humans need around 40 hours to create a threat model for a particular system [29]. It follows that our technique provides a very efficient substitute for manual approaches, drastically cutting down the time required for threat model building. To improve public safety, this capacity to respond quickly is of the utmost importance, as safeguarding vital infrastructure and





CONCLUSION

With its innovative approach to integrated security systems, ThreatSentinelGPT fills in important gaps left by outdated methods that depend on disparate monitoring instruments and human threat detection. The framework offers a comprehensive approach to security by integrating physical surveillance with cybersecurity monitoring. This enables the real-time identification and mitigation of both digital and physical threats. To provide precise detection and intelligent threat categorization, the system makes use of state-of-the-art technologies including computer vision, machine learning, and natural language processing. Together, object detection algorithms and machine learning models can effectively categorize cyber threats from system logs and identify intrusions, weapon presence, and anomalous activities inside monitored regions in real-time. Timely reactions are made possible by effectively contextualizing potential threats and assigning severity levels by the capacity to continually connect events from numerous sources. By keeping human interaction at a minimum while yet achieving high reliability, ThreatSentinelGPT frees up security staff to concentrate on making decisions and managing crucial incidents. Damage, loss, or compromise of sensitive assets might be lessened due to improved situational awareness and the real-time alarm system. The system's resilience under varied scenarios has been shown by testing methodologies such as unit, integration, system, performance, stress, and security testing. This confirms the system's capabilities to withstand many concurrent attacks. The system's adaptive machine learning models bolster its resilience even further. These models adapt to new threat patterns and improve detection accuracy as the system evolves. Deploying the framework across several physical locations, diverse camera networks, and vast log monitoring infrastructures is made possible by its modular architecture, which promotes scalability. With ThreatSentinelGPT, you can be

certain that warnings will be both accurate and responsive. The system uses severity levels and clear prioritization to guide operational actions. Thorough testing and improvement of the model have reduced the occurrence of false positives and negatives, guaranteeing that security warnings are only triggered by legitimate threats. Overall security is improved by detecting complex, multi-domain attack scenarios via the combination of cyber and physical surveillance. Even under heavy loads, resource optimization—which includes effective control of the central processing unit, graphics processing unit, and memory—allows for uninterrupted operation without performance loss. To aid in forensic investigations and ensure conformity with regulatory requirements, automated logging and audit trails provide openness and responsibility. System learning from operator input and false-positive situations allows detection models to be refined for increased accuracy via iterative improvements and continuous feedback loops.

FUTURE ENHANCEMENTS

Upgrades to ThreatSentinelGPT in the future will address new security threats by enhancing detection accuracy, broadening the capabilities of the system, and making operations more efficient. Incorporating predictive analytics, which examine past data from CCTV feeds and system records to anticipate possible security events before they happen, is one potential advance. Preemptive alarms, area lockdowns, or heightened surveillance in high-risk zones might be made possible with this. By connecting to the internet of things (IoT) and smart devices, environmental sensors, access controls, and other smart home gadgets may all contribute data to the system, enhancing situational awareness and enabling more comprehensive monitoring. To further reduce network latency and guarantee quick danger identification even with large data quantities, the system might use edge computing methods to do local analysis at camera nodes. The detection of suspicious actions and high-risk persons might be enhanced with the use of advanced video analytics, including face recognition, behavior prediction, and crowd density analysis. Retraining machine learning models on real-time operating data is one way to increase their accuracy, decrease the number of false positives, and make them better at detecting new physical and cyber risks. Automated detection of established attack patterns, malware signatures, or hostile IP addresses might be achieved by integration with threat intelligence feeds. To better comprehend security incidents, multi-modal threat correlation might be enhanced to include operational and environmental context, including things like fire alarms, building access records, or

network abnormalities. Based on the kind and intensity of the danger, the system might implement automatic reaction procedures like locking doors, generating alarms, or segmenting the network. Reducing response delays, improvements to alert prioritization algorithms might guarantee that serious occurrences are escalated to the right persons quickly. To help operators better understand the situation and make informed decisions, natural language generation might be used to provide short, actionable summaries of security incidents. Interactive dashboards, real-time maps, and 3D representations of monitored regions might greatly improve the user interface and make it easier for operators to handle complicated events intuitively.

REFERENCES

- [1] Y. Chen, Y. Tian, and M. He, "Monocular human pose estimation: A survey of deep learning-based methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020.
- [2] C. Zheng, W. Wu, C. Chen, and M. Shah, "Deep learning-based human pose estimation: A survey," *IEEE Access*, 2020.
- [3] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2020.
- [4] Y. Chen *et al.*, "Monocular human pose estimation: A survey of deep learning-based methods," *IEEE Access*, 2020.
- [5] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson, 2020.
- [6] H. Kaur and A. Singh, "Video surveillance systems: Current trends and future directions," *IEEE Access*, 2020.
- [7] J. Xu, B. Ni, and X. Yang, "Deep kinematics analysis for monocular 3D human pose estimation," *IEEE Trans. Image Process.*, 2020.
- [8] M. S. Hossain *et al.*, "Smart surveillance: Deep learning approaches for real-time threat detection," *Sensors*, vol. 21, no. 6, p. 1987, 2021.
- [9] F. Zhang, X. Zhu, and C. Wang, "Single person pose estimation: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2021.
- [10] M. S. Hossain, "Smart surveillance: Deep learning approaches for real-time threat detection," *Sensors*, 2021.
- [11] H. Chen, R. Feng, S. Wu, and H. Xu, "2D human pose estimation and computer vision techniques," *IEEE Access*, 2022.
- [12] N. Ahmed *et al.*, "AI-driven security monitoring systems: A comprehensive review," *J. Inf. Secur. Appl.*, vol. 65, p. 103032, 2022.
- [13] M. M. E. Yurtsever and S. Eken, "BabyPose: Real-time decoding of baby's non-verbal communication using pose estimation," *IEEE Access*, 2022.
- [14] E. Alam, A. Sufian, and P. Dutta, "Vision-based human fall detection systems using deep learning: A review," *IEEE Access*, 2022.
- [15] H. Chen, R. Feng, S. Wu, and H. Xu, "2D human pose estimation," *IEEE Access*, 2022.
- [16] J. W. Kim, J. Y. Choi, E. J. Ha, and J. H. Choi, "Human pose estimation using MediaPipe Pose and optimization method based on a humanoid model," *Applied Sciences*, vol. 13, no. 2, pp. 1–15, 2023.
- [17] J.-W. Kim, J.-Y. Choi, E.-J. Ha, and J.-H. Choi, "Human pose estimation using MediaPipe Pose and optimization method based on a humanoid model," *Applied Sciences*, 2023.
- [18] A. Bochkovskiy *et al.*, "You Only Look Once (YOLO): Object detection models," *IEEE Access*, 2023.
- [19] J. W. Kim *et al.*, "Human pose estimation using MediaPipe Pose," *Applied Sciences*, 2023.
- [20] "A systematic survey on human pose estimation: Approaches and applications," *Springer*, 2024.