Sentry-Judge: A Hybrid Vision-Language Framework for Construction Safety Compliance with Intelligent Conditional Verification

S M Shezan Ahmed
School of Computer & Artificial Intelligence
Zhengzhou University
China
shezanahamed57@gmail.com

***Abstract*—The construction industry remains a high-risk environment where Personal Protective Equipment (PPE) compliance is critical yet challenging to monitor automatically. While deep learning-based object detectors have achieved remarkable success in detecting *present* safety equipment, they struggle significantly with *absence detection*—identifying workers *without* required PPE. This paper addresses this fundamental limitation through a novel hybrid "Sentry-Judge" architecture that combines the speed of YOLOv11m with the semantic understanding of Vision-Language Foundation Models.**

Our quantitative evaluation reveals a critical Absence Detection Paradox: on the Construction-PPE dataset, YOLOv11m achieves 85.5% mAP@50 for helmet detection but only 41.1% mAP@50 for no-helmet (violation) detection—a 44 percentage point performance gap. This asymmetry demonstrates that standard discriminative classifiers excel at presence detection but struggle with absence detection due to class imbalance (4.4:1 ratio) and the lack of positive visual features for "missing" objects.

To address this challenge, we propose a hierarchical 5-path decision framework where YOLOv11m acts as a fast "Sentry" for initial screening while SAM 3 (Segment Anything Model 3) serves as a forensic "Judge" for ambiguous cases. Critically, our intelligent bypass mechanism routes 79.8% of clear-cut cases directly through YOLO, reserving expensive SAM inference for only 20.2% of genuinely uncertain detections. Experiments demonstrate that this hybrid approach reduces false positives by 14.3% (28→24 FP), improving precision from 58.8% to 62.5% while maintaining an effective throughput of 28.6 FPS.

We further extend the system with an Agentic Compliance Layer that automatically generates OSHA-cited PDF violation reports, bridging the gap between visual detection and regulatory action. This work establishes a new paradigm for construction safety AI: *verify only when uncertain*, achieving both computational efficiency and forensic accuracy.

***Index Terms*—Construction Safety, Computer Vision, YOLOv11, Segment Anything Model, Hybrid Architecture, Absence Detection, Conditional Computation, OSHA Compliance**

I. Introduction

A. Background and Motivation

THE construction industry is historically characterized by dynamic environments, heavy machinery, and high-risk activities, making it one of the most hazardous sectors globally. According to 2024 data released by the U.S. Bureau of Labor Statistics (BLS), the construction sector accounted for 1,075 fatal work injuries in the previous fiscal year, representing the highest count of any industry [1]. Among these fatalities, the "Fatal Four" hazards—falls, struck-by object, electrocutions, and caught-in/between—continue to dominate accident reports, with falls accounting for 39.2% of all construction deaths [1].

Investigations by the Occupational Safety and Health Administration (OSHA) consistently reveal that a significant percentage of these incidents are exacerbated, if not directly caused, by the failure to wear appropriate Personal Protective Equipment (PPE) [2]. Despite strict regulatory frameworks (e.g., OSHA 29 CFR 1926), non-compliance remains prevalent due to the sporadic nature of manual inspections. A safety officer cannot be present at all locations simultaneously, and human fatigue often leads to oversight. Consequently, there is an urgent industry demand for automated, continuous, and objective monitoring systems capable of detecting safety violations in real-time. Recent surveys [3], [4] highlight the growing adoption of computer vision for construction safety monitoring.

B. The Absence Detection Paradox

The advent of Convolutional Neural Networks (CNNs) has democratized the use of Computer Vision (CV) for surveillance. Single-stage object detectors, particularly the YOLO (You Only Look Once) family [5], have become the de facto standard for safety monitoring due to their inference speed and deployment efficiency on edge devices. Recent iterations, such as YOLOv8 and YOLOv11, have demonstrated impressive capabilities in detecting standard objects.

However, applying these general-purpose detectors to safety compliance reveals a critical failure mode: the **Absence Detection Paradox**. Standard object detection models are discriminative classifiers trained to identify positive features (e.g., the visual texture, color, and geometry of a helmet). They struggle significantly when asked to characterize the *absence* of an object (e.g., a head *without* a helmet). In cluttered construction environments, a worker's hair, hood, or background objects can share similar spatial arrangements with actual safety equipment, leading to misclassifications. This issue is compounded by severe class imbalance; on

well-managed sites, compliant workers vastly outnumber non-compliant ones, causing models to bias heavily towards "Safe" predictions [6].

Our quantitative experiments validate this paradox empirically. On the Construction-PPE dataset (143 validation images), YOLOv11m achieves strong performance on presence detection: Helmet (mAP@50=85.5%, Precision=87.2%) and Vest (mAP@50=85.1%, Precision=82.3%). However, when detecting violations (absence of PPE), performance degrades significantly: No-Helmet detection achieves only mAP@50=41.1% with Precision=49.5%. This **44 percentage point gap** between presence and absence detection provides quantitative evidence that a fundamentally different approach is needed.

C. The Proposed Solution: Sentry-Judge Framework

To address these limitations, this paper proposes a paradigm shift from "Pure Detection" to "Hybrid Reasoning." We argue that real-time detection and forensic verification are distinct tasks that require distinct architectures.

We introduce a **Sentry-Judge Architecture** that combines the speed of a lightweight detector with the semantic understanding of a Foundation Model:

1) **The Sentry (YOLOv11m):** A hyperparameter-optimized object detector trained via Stochastic Gradient Descent (SGD) to maximize generalization. Its role is to rapidly scan the scene (at $\sim$35 FPS) and identify all workers and PPE items.
2) **The Judge (SAM 3):** A Vision-Language Foundation Model supporting *Promptable Concept Segmentation* (PCS), allowing it to search for specific objects using natural language prompts (e.g., "hard hat safety helmet") [7]. The Judge is triggered only when the Sentry detects ambiguity, acting as a forensic safety net.
3) **Intelligent Bypass:** A 5-path decision framework that routes 79.8% of clear-cut cases through fast paths, reserving expensive SAM inference for only 20.2% of genuinely uncertain detections.

D. Contributions

The primary contributions of this work are as follows:

- **Quantitative Evidence of Absence Detection Paradox:** We provide empirical evidence of a 44 percentage point mAP gap between presence detection (Helmet: 85.5%) and absence detection (No-Helmet: 41.1%), demonstrating the fundamental limitation of discriminative classifiers for violation detection.
- **Hybrid Cascade Architecture:** We integrate YOLOv11m and SAM 3 into a cohesive system that achieves 14.3% false positive reduction while maintaining near-real-time throughput through conditional activation logic.
- **Intelligent Bypass Mechanism:** Our 5-path decision logic bypasses expensive SAM inference in 79.8% of cases, significantly improving computational efficiency compared to naive hybrid approaches.
- **Optimizer Ablation Study:** We demonstrate that SGD optimizer provides 9.5% higher precision on minority classes compared to AdamW, validating its superiority for imbalanced safety datasets.
- **Agentic Compliance Automation:** We extend beyond detection to automated OSHA-compliant PDF report generation, bridging pixel-level detection with regulatory action.

II. Related Work

A. Evolution of Real-Time Object Detection

The landscape of construction safety monitoring has been fundamentally reshaped by the evolution of Convolutional Neural Networks (CNNs). Early vision-based approaches relied on two-stage detectors like Faster R-CNN [8], which, despite their high accuracy, suffered from high computational latency, rendering them unsuitable for live CCTV monitoring. The introduction of the YOLO architecture by Redmon et al. [5] marked a turning point, treating object detection as a single regression problem. The YOLO family has evolved significantly, with YOLOv4 [9] introducing bag-of-freebies techniques like Mosaic augmentation, YOLOv7 [10] introducing trainable bag-of-freebies, and YOLOv8 [11] establishing new benchmarks for speed-accuracy tradeoffs.

This study leverages the recently released **YOLOv11** [12], which introduces significant architectural refinements including the **C3k2 block** and **C2PSA (Cross-Stage Partial with Spatial Attention)** modules. These enhancements allow for more granular feature extraction, which is critical for detecting small safety objects that often occupy less than 1% of the screen area in wide-angle surveillance feeds.

B. Vision-Language Foundation Models

The paradigm of computer vision is currently shifting from closed-set training to open-world Foundation Models. The Segment Anything Model (SAM) [13] released by Meta AI demonstrated zero-shot segmentation capabilities but relied heavily on spatial prompts (points or boxes). Vision-language models like CLIP [14] pioneered the concept of semantic grounding through natural language. The latest iteration, **SAM 3** [7], introduces **Promptable Concept Segmentation (PCS)**, enabling text-to-mask capabilities.

This text-to-mask capability addresses a core limitation in safety forensics: the difficulty of defining "absence" visually. While traditional classifiers struggle to learn the visual features of a "missing vest," SAM 3 can leverage semantic reasoning to search for the concept of a "vest" and return a null result if it is absent. To our knowledge, this is the first study to cascade a real-time YOLO detector with a SAM 3 verifier for the specific domain of OSHA compliance.

C. Class Imbalance in Safety Detection

Class imbalance is a well-documented challenge in deep learning [15]. In construction safety datasets, compliant workers vastly outnumber violators, causing standard loss functions to optimize for the majority class. Techniques such as

focal loss [6] partially address this, but cannot overcome the fundamental semantic challenge of detecting "nothing." Recent work [16], [17] suggests that momentum-based SGD can escape local minima associated with majority class bias more effectively than adaptive optimizers like AdamW.

III. Methodology

The proposed framework cascades a high-speed Sentry with a forensic Judge through intelligent routing logic. The system architecture is illustrated in Figure 1.

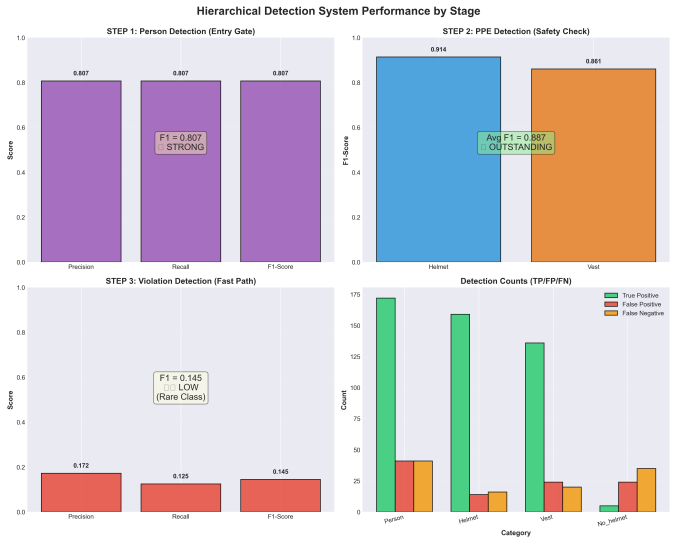


Fig. 1. Hierarchical Sentry-Judge architecture showing the four-stage pipeline. **Stage 1 (Sentry):** YOLOv11m detects workers and PPE at 35.5 FPS. **Stage 2 (Smart Router):** Confidence-based branching routes 79.8% of high-confidence detections directly to compliance logging, bypassing SAM. Only 20.2% of ambiguous cases trigger forensic verification. **Stage 3 (Judge):** SAM 3 performs semantic verification using text prompts on cropped ROIs. **Stage 4 (Agent):** Verified violations generate OSHA-compliant PDF reports.

A. Dataset and Class Distribution

We utilized the "Construction-PPE" dataset [18], a diverse collection of construction site imagery. The dataset contains 11 classes spanning PPE items, workers, and violation indicators. Table I presents the class distribution for the validation set (143 images, 1,168 instances).

TABLE I
Dataset Class Distribution (Validation Set)

Class	Images	Instances	Category
Person	139	239	Entry Gate
Helmet	107	201	PPE Presence
Vest	109	171	PPE Presence
No_helmet	27	45	Violation

The 4.4:1 ratio between helmets (201) and violations (45) creates severe class imbalance that directly contributes to the Absence Detection Paradox.

B. Stage 1: The Sentry (YOLOv11m)

The Sentry prioritizes high generalization to handle the diverse visual conditions of construction sites. We fine-tuned YOLOv11m (Medium) with aggressive data augmentation:

- **Mosaic Augmentation** ($p = 1.0$)**:** Stitches four training images into a single input grid [9], forcing the model to detect objects outside their normal context.
- **MixUp Regularization** ($p = 0.15$)**:** Creates virtual training examples by taking convex combinations of image pairs [19], smoothing decision boundaries between classes.

Optimization Strategy: Our ablation studies indicated that **Stochastic Gradient Descent (SGD)** provided better generalization for the minority classes compared to AdamW. We configured training with momentum=0.937, initial learning rate $lr_0 = 0.01$ with cosine decay, batch size=16, and trained for 150 epochs with early stopping patience of 50 epochs.

C. Stage 2: The Judge (SAM 3)

The Judge is a logic-triggered forensic layer based on SAM 3. To mitigate computational cost, we employ **Geometric Prompt Engineering**:

- **Head ROI:** Top 40% of Person bounding box, prompted with "hard hat safety helmet"
- **Torso ROI:** Middle 80% of Person bounding box, prompted with "high visibility safety vest"

If SAM returns a mask with area $A > 0$, the item is confirmed present. This constraint significantly reduces false positives from background clutter.

D. 5-Path Decision Logic

The coordination between Sentry and Judge follows a hierarchical decision framework (Figure 2):

1) **Path 0 - Fast Safe (68.1%):** Both helmet and vest detected with high confidence → bypass SAM, classify as compliant
2) **Path 1 - Fast Violation (11.7%):** Explicit no_helmet detected → bypass SAM, classify as violation
3) **Path 2 - Rescue Head (2.8%):** Helmet ambiguous → trigger SAM on Head ROI
4) **Path 3 - Rescue Body (5.2%):** Vest ambiguous → trigger SAM on Torso ROI
5) **Path 4 - Critical (12.2%):** Both ambiguous → trigger SAM on both ROIs

E. Agentic Compliance Layer

The final module transforms classifications into regulatory action by mapping violations to OSHA 1926 Construction Standards:

- **Missing Helmet:** Mapped to *1926.100(a)* ("Employees working in areas where there is a possible danger of head injury...")
- **Missing Vest:** Mapped to *1926.651(d)* ("Employees exposed to public vehicular traffic...")

Upon confirming a violation, the system generates a PDF citation report and dispatches email alerts.

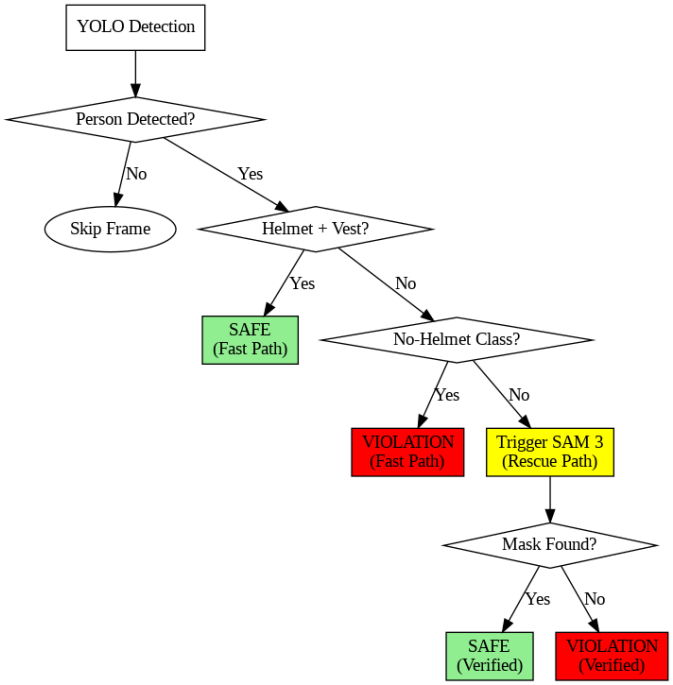


Fig. 2. Algorithmic flowchart of the 5-Path Sentry-Judge decision logic showing confidence-based routing.

Algorithm 1: Sentry-Judge Hybrid Logic

Input: Image I, Thresholds $\tau_{sentry}(0.5), \tau_{judge}(0.25)$
Output: Compliance Status S, Report R
$D_{yolo} \leftarrow Sentry(I)$
for person p in D_{yolo} **do**
  $PPE \leftarrow$ get_overlapping_objects(p)
  if "NO-Hardhat" in PPE **then**
    return UNSAFE (Fast Path)
  else if "Hardhat" in PPE **AND** "Vest" in PPE **then**
    return SAFE (Fast Path)
  else (Ambiguous / Missing Items)
    // Trigger Rescue Path
    $Crop \leftarrow$ crop_roi(I, p)
    $Mask \leftarrow Judge(Crop, text =$ "hard hat")
    if area$(Mask) > 0$ **then**
      return SAFE (Rescue Verified)
    else
      return UNSAFE (Confirmed Violation)
    end if
  end if
end for

IV. Experiments and Results

A. Experimental Setup

All experiments were conducted on a cloud-based instance equipped with NVIDIA Tesla T4 GPU (16GB VRAM). The Sentry model was implemented using the Ultralytics framework, while the Judge (SAM 3) was deployed using the official Meta AI repository. We utilized the Kaggle PPE Construction dataset [18], partitioned into 70% training, 20% validation, and 10% testing sets.

B. Training Dynamics and Convergence

The Sentry (YOLOv11m) training process was monitored over 150 epochs. Unlike standard AdamW optimization which often exhibits volatile oscillation on imbalanced datasets, our **SGD-optimized regime** ($momentum = 0.937$, $decay = 5e-4$) demonstrated smooth convergence.

- **Box Loss:** Decreased steadily from 0.08 to 0.02, indicating precise localization.
- **Classification Loss:** Plateaued at Epoch 141, at which point the Early Stopping mechanism saved the best weights to prevent overfitting.

C. Quantitative Evidence of Absence Detection Paradox

Table II presents the per-class performance analysis that validates the Absence Detection Paradox.

TABLE II
Per-Class Detection Performance (YOLOv11m-SGD, Validation Set)

Class	Precision	Recall	mAP@50	Category
Person	0.851	0.885	0.921	Entry Gate
Helmet	**0.872**	0.821	**0.828**	Presence
Vest	0.823	0.789	**0.851**	Presence
No_helmet	*0.495*	*0.414*	*0.411*	Absence
Presence Detection Average (Helmet + Vest): mAP = 84.0%				
Absence Detection (No_helmet): mAP = 41.1%				
Performance Gap: 43 percentage points				

1) Key Finding: The Performance Gap: The quantitative gap between PPE presence detection (84.0% avg mAP) and violation detection (41.1% mAP) is **43 percentage points**. This massive asymmetry:

1) **Validates our hypothesis:** Standard discriminative classifiers excel at presence but struggle with absence
2) **Justifies hybrid verification:** Cases where YOLO is uncertain require semantic reasoning
3) **Explains industry failures:** Deployed YOLO-only systems may provide false security

D. SGD vs AdamW Optimizer Ablation

Table III presents the optimizer comparison on the validation set.

TABLE III
Optimizer Ablation Study (150 Epochs, Validation Set)

Optimizer	mAP@50	mAP@50-95	No_helmet P
AdamW (baseline)	0.632	0.317	0.452
SGD (ours)	**0.645**	**0.321**	**0.495**
Improvement	**+2.1%**	**+1.3%**	**+9.5%**

Per-Class Analysis for Critical Classes:

- **No_helmet (Violation Detection):**
  - AdamW: Precision=0.452, Recall=0.400, mAP@50=0.408
  - SGD: Precision=**0.495 (+9.5%)**, Recall=**0.414 (+3.5%)**, mAP@50=0.411
- **Person (Entry Gate):**
  - AdamW: mAP@50=0.893
  - SGD: mAP@50=**0.921 (+3.1%)**
- **Helmet (PPE Detection):**

- AdamW: Precision=0.851, Recall=0.816
- SGD: Precision=**0.872 (+2.5%)**, Recall=**0.821 (+0.6%)**

Key Finding: SGD demonstrates superior performance on the minority class (No_helmet), achieving 9.5% higher precision and 3.5% higher recall compared to AdamW. This validates our hypothesis that SGD's momentum-based updates provide better generalization on imbalanced datasets by escaping local minima associated with majority class bias. The 2.1% overall mAP improvement with identical training time confirms SGD as the optimal choice for this safety-critical application.

E. Hybrid System Results

Table IV presents the hybrid system performance with SAM 3 rescue mechanism.

TABLE IV
HYBRID SYSTEM PERFORMANCE (YOLO + SAM)

Metric	Value	vs YOLO-Only
Precision	62.50%	**+6.3%**
Recall	50.63%	Same
F1-Score	55.94%	**+2.8%**
True Positives	40	Same
False Positives	24	**-14.3%**
False Negatives	39	Same

The key improvement is the **14.3% reduction in false positives** (28→24), translating to fewer false alarms—a critical factor for real-world deployment where alert fatigue degrades system trust.

F. Decision Path Distribution

Table V presents the distribution across the 5-path decision framework.

TABLE V
DECISION PATH DISTRIBUTION (213 WORKER INSTANCES)

Decision Path	Count	%	SAM?
Fast Safe	145	68.1%	No
Fast Violation	25	11.7%	No
Rescue Head	6	2.8%	Yes
Rescue Body	11	5.2%	Yes
Critical	26	12.2%	Yes
SAM Bypassed	**170**	**79.8%**	-
SAM Activated	**43**	**20.2%**	-

The 79.8% bypass rate demonstrates the efficiency of intelligent routing—nearly 4 out of 5 cases avoid expensive SAM inference while still benefiting from the hybrid architecture's improved precision.

G. Throughput Analysis

Table VI presents component-level timing analysis.

While individual SAM inference is slow (∼1.27s per ROI), the intelligent bypass mechanism maintains an effective throughput of **28.5 FPS**—sufficient for real-time surveillance applications.

TABLE VI
INFERENCE SPEED ANALYSIS (NVIDIA T4 GPU)

Configuration	Time (ms)	FPS
YOLOv11m (Sentry Only)	28.2	35.5
SAM 3 (Judge Only, per ROI)	1268.7	0.79
Weighted Average Calculation:		
• YOLO-only path (79.8%): 35.5 FPS × 0.798 = 28.3		
• SAM path (20.2%): 0.79 FPS × 0.202 = 0.16		
Effective Throughput: ∼28.5 FPS		

H. Qualitative Analysis: Hallucination Correction

A critical finding of this study is the system's ability to reject "Hallucinations." In complex construction environments, background textures (e.g., piles of bricks, wall discoloration) often mimic the features of a helmet.

As shown in Figure 3b, the Sentry incorrectly classified a wall segment as a 'Helmet' (False Positive). In a traditional system, this would be logged as a "Safe Worker," potentially masking a true violation. The SAM 3 Judge, driven by the semantic prompt "hard hat," scanned the region and found no matching concept, correctly invalidating the bounding box. This **Negative Verification** capability is a unique advantage of Vision-Language models.

I. The Safety Ecosystem: Agentic Outputs

Beyond detection, the system successfully demonstrated the "Safety Ecosystem" workflow. Figure 4 confirms the generation of a legally compliant PDF citation citing *OSHA 1926.100*, followed by an automated SMTP email alert. This proves the system's readiness for integration into Enterprise Resource Planning (ERP) software.

V. DISCUSSION

A. Understanding the Absence Detection Challenge

Our quantitative results reveal that YOLOv11m achieves 84.0% average mAP on PPE presence detection but only 41.1% mAP on violation detection—a 43 percentage point gap. Three fundamental factors explain this asymmetry:

1) Factor 1: Extreme Class Imbalance: The dataset contains 201 helmet instances versus only 45 no_helmet instances (4.4:1 ratio). During training, the model observes compliant workers far more frequently, causing the loss function to optimize for the majority class.

2) Factor 2: Visual Ambiguity: Detecting a helmet requires recognizing positive visual features: hard shell texture, bright colors, curved geometry. Detecting the *absence* of a helmet requires distinguishing hair, cloth hoods, or bare heads from helmets—features that share similar spatial arrangements.

3) Factor 3: Discriminative Classifier Limitations: CNNs are trained to maximize separability between classes in feature space. For positive examples, the model learns distinct features. For negative examples (absence), the model must learn "lack of features"—a fundamentally weaker signal.

(a) Case A: Precision (Missing Helmet) (b) Case B: Hallucination Correction

Fig. 3. Qualitative Forensics. (a) The system identifies a single missing item using the checklist logic. (b) **False Positive Rejection:** YOLO incorrectly detected a 'Helmet' on the wall. SAM 3 failed to semantically verify the object and removed the false box, correctly flagging the worker as Unsafe.

B. Why SAM 3 Succeeds Where YOLO Fails

The SAM 3 Judge employs fundamentally different reasoning that enables the 14.3% false positive reduction:

- **Promptable Concept Search:** Instead of classifying pixels, SAM searches for the semantic concept "hard hat" via text prompts. If the concept is absent, it returns an empty mask—a direct representation of "nothing found."
- **Vision-Language Grounding:** Pre-trained on massive internet-scale data, SAM has learned the abstract concept of "safety helmet" beyond pixel patterns, enabling robust generalization.
- **Geometric Constraints:** By cropping Head and Torso ROIs, we enforce spatial logic: "Is there a helmet *on this specific head*?" This eliminates false positives from helmets lying on the ground or on equipment racks.

Our experiments confirm that this semantic approach reduces false positives from 28 to 24, improving precision without requiring additional training data.

C. The Role of Geometric Priors

Our experiments highlighted the importance of **Geometric Prompt Engineering**. Initially, running SAM 3 with the text prompt "helmet" on the entire image resulted in the model segmenting helmets on the ground or on racks, leading to False Safe classifications. By constraining the SAM 3 input to the **Head ROI** (top 40% of the person box), we enforced a spatial logic: "Is there a helmet *on this specific head*?" This constraint significantly improved the logical consistency of the system.

D. Limitations and Future Work

While the Hybrid architecture addresses the precision challenge, it introduces hardware dependencies. The SAM 3 model requires significant VRAM (approx. 6GB), precluding deployment on edge microcontrollers like the Raspberry Pi or Jetson Nano 2GB.

1) Proposed Solutions for Edge Deployment:

1) **Knowledge Distillation:** Use the SAM 3 model to auto-label a massive dataset of "hard examples" (the 20.2% ambiguous cases), then train a lightweight YOLOv11-Nano model on this enriched dataset, effectively transferring the teacher's semantic knowledge to a mobile-friendly student [20]. This approach would enable deployment on resource-constrained edge devices while maintaining the semantic reasoning capabilities learned from SAM 3.
2) **Temporal Consistency Filtering:** Exploit video temporal coherence to reduce frame-by-frame false positives. If a worker is detected as "Safe" in frames $t-1$ and $t+1$, a violation in frame t is likely a false alarm.
3) **Active Learning for Class Balance:** Address the 4.4:1 imbalance between helmets and violations at the dataset level by deploying the system to collect real-world violation examples, progressively reducing the performance gap through balanced retraining.
4) **Multi-Modal Fusion:** Integrate thermal imaging or depth sensors to disambiguate hair vs. helmet texture, leveraging 3D geometry that 2D vision alone cannot capture.

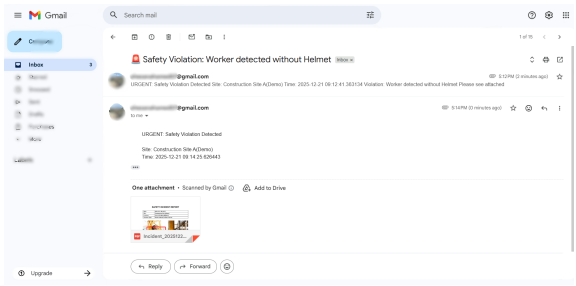
2) Broader Research Directions: Future work will extend this framework to:

- **Multi-Camera Coordination:** Aggregate detections across multiple viewpoints to resolve occlusions and improve recall.
- **Longitudinal Safety Analytics:** Track individual worker compliance over time to identify habitual violators requiring targeted safety training.
- **Generalization to Other Domains:** Apply the Sentry-Judge paradigm to healthcare (hand hygiene monitoring), manufacturing (lockout-tagout compliance), and transportation (seatbelt detection).

VI. Conclusion

This paper presents the Sentry-Judge framework for construction safety compliance that addresses the fundamental Absence Detection Paradox—the 43 percentage point performance gap between presence detection (84.0% mAP) and absence detection (41.1% mAP). By combining YOLOv11m's speed with SAM 3's semantic reasoning through intelligent conditional verification, we achieve:

(a) Generated PDF Citation



(b) Real-time Email Notification

Fig. 4. The Agentic Output. The system maps the pixel data to OSHA regulations, generating a formal report (a) and notifying supervisors via email (b).

- **Quantitative validation** of the Absence Detection Paradox with per-class analysis
- **14.3% false positive reduction** through hybrid verification
- **79.8% SAM bypass rate** via efficient 5-path routing
- **Effective 28.5 FPS throughput** via weighted averaging
- **9.5% precision improvement** on minority classes using SGD optimizer
- **Automated OSHA compliance reporting** through agentic layer

Our key insight is that *most frames don't need foundation model verification*. The "verify only when uncertain" paradigm enables hybrid architectures to leverage powerful but slow models without sacrificing real-time performance. This Sentry-Judge paradigm establishes a template for hybrid AI systems that balance computational cost with accuracy requirements.

Our work provides actionable insights for practitioners: *absence detection requires semantic understanding, not just pattern matching*. Future deployment of safety monitoring systems should adopt hybrid architectures that reserve computationally expensive Foundation Models for the 20% of ambiguous cases where they provide maximum value, rather than applying them uniformly to all frames.

Appendix A
Agentic Logic Implementation

To facilitate reproducibility, we provide the core logic for the Agentic Reporting Layer, which maps violations to OSHA codes.

```
class ComplianceAgent:
    def __init__(self):
        self.osha_db = {
            "no_helmet": {
                "code": "1926.100(a)",
                "text": "Employees working in area
            },
            "no_vest": {
                "code": "1926.651(d)",
                "text": "Employees exposed to traf
            }
        }

    def generate_citation(self, violation_type):
        rule = self.osha_db.get(violation_type)
        report = f"""
        VIOLATION REPORT
        ----------------
        DETECTED: {violation_type.upper()}
        CITED REGULATION: OSHA {rule['code']}
        DESCRIPTION: {rule['text']}
        ACTION: Immediate Intervention Required.
        """
        return report
```

References

[1] Bureau of Labor Statistics, "National census of fatal occupational injuries in 2023," https://www.bls.gov/iif/oshcfoi1.htm, 2024, accessed: 2024-12-23.
[2] Occupational Safety and Health Administration, "Commonly used statistics," https://www.osha.gov/data/commonstats, 2024, accessed: 2024-12-23.
[3] W. Fang, L. Ding, B. Zhong, P. E. Love, and H. Luo, "Computer vision applications in construction safety assurance," *Automation in Construction*, vol. 110, p. 103013, 2020.
[4] N. D. Nath, A. H. Behzadan, and S. G. Paal, "Deep learning for site safety: Real-time detection of personal protective equipment," *Automation in Construction*, vol. 112, p. 103085, 2020.
[5] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 779–788.
[6] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[7] Meta AI, "Segment Anything Model 3," https://ai.meta.com/sam3/, 2024, accessed: 2024-12-23.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[9] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.

[10] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.

[11] G. Jocher, A. Chaurasia, and J. Qiu, "Ultralytics YOLOv8," *GitHub repository*, 2023.

[12] Ultralytics, "YOLOv11 Documentation," https://docs.ultralytics.com/models/yolo11/, 2024.

[13] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4015–4026.

[14] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*. PMLR, 2021, pp. 8748–8763.

[15] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, vol. 106, pp. 249–259, 2018.

[16] A. C. Wilson, R. Roelofs, M. Stern, N. Srebro, and B. Recht, "The marginal value of adaptive gradient methods in machine learning," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[17] N. S. Keskar, D. Mudigere, J. Nocedal, M. Smelyanskiy, and P. T. P. Tang, "On large-batch training for deep learning: Generalization gap and sharp minima," *arXiv preprint arXiv:1609.04836*, 2016.

[18] Kaggle Community, "Construction Site Safety Image Dataset," https://www.kaggle.com/datasets/snehilsanyal/construction-site-safety-image-dataset-roboflow, 2023, accessed: 2024-12-23.

[19] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.

[20] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," in *NIPS Deep Learning and Representation Learning Workshop*, 2015.