AI-Powered Crime Scene Evidence Identification and Classification System

Paarth Birla, Dev Lad, Prerak Patel
National Forensic Sciences University (NFSU), Goa Campus
Under the Guidance of: Prof. Ranjit Kolkar

Abstract

In the modern domain of forensic science, the rapid, accurate, and immutable identification of evidence at a crime scene is paramount for maintaining the chain of custody and ensuring investigative integrity. Traditional methods of evidence logging are fraught with human error, subjectivity, and the risk of contamination. This paper presents the design, implementation, and theoretical framework of an advanced, AI-powered automated evidence detection and reconstruction system. The proposed solution utilizes a novel **Ensemble Evidence Detector** architecture, aggregating predictions from multiple fine-tuned YOLOv8 (You Only Look Once) models to identify diverse forensic artifacts—ranging from biological fluids to ballistic weaponry—in real-time. Beyond 2D detection, this research introduces a **Monocular Depth Estimation** module utilizing Vision Transformers (ViT) to generate depth maps from static images, enabling the creation of immersive 2.5D/3D visualizations in a browser-based Virtual Reality (VR) environment. The system features a secure, professional web interface tailored for forensic analysts, ensuring legal compliance and operational efficiency. This paper details the mathematical underpinnings of the object detection loss functions, the depth estimation architecture, and the photogrammetric principles used for 3D reconstruction, concluding with a roadmap for integration with Augmented Reality (AR) headsets and "Virtual Jury" courtroom presentations.

---

1. Introduction

1.1 Background

Crime scene investigation (CSI) is the foundation of the criminal justice system. The primary objective is to recognize, document, collect, and preserve physical evidence. Historically, this process has been manual, relying on the investigator's visual acuity and meticulous note-taking. Standard operating procedures (SOPs) often involve grid or spiral search patterns, followed by the placement of physical numbered markers and photography.

While effective, these traditional methods suffer from significant limitations: 1. **Cognitive Load:** Investigators working under high-pressure conditions (e.g., active scenes, decomposing bio-hazards) may overlook subtle evidence like small caliber casings or faint blood spatter. 2. **Scene Contamination:** The longer investigators remain physically present to search for evidence, the higher the risk of introducing foreign DNA or disturbing spatial relationships. 3. **Lack of Spatial**

Context: Standard 2D photography flattens the scene, causing a loss of depth perception. In court, it is often difficult to convey the specific spatial layout (e.g., trajectory angles) to a lay jury using flat images.

1.2 Motivation

The advent of Deep Learning, specifically Convolutional Neural Networks (CNNs), offers a transformative solution. By automating the "recognition" phase of CSI, Artificial Intelligence can act as a tireless second pair of eyes, ensuring no evidence is missed. Furthermore, the combination of Computer Vision with Photogrammetry—the science of making measurements from photographs—allows for the digital reconstruction of scenes, preserving them effectively "frozen in time."

1.3 Project Objectives

This project aims to bridge the gap between state-of-the-art Computer Vision and practical application in Forensic Science. The key objectives are: * **Automated Detection:** To develop a real-time system capable of identifying high-priority forensic evidence (Weapons, Blood) and general context objects (Persons, Electronics) simultaneously. * **Ensemble Accuracy:** To mitigate the "False Negative" rate by utilizing a dual-model ensemble approach. * **Immersive Reconstruction:** To convert standard 2D crime scene photographs into navigable 3D VR environments for better spatial analysis. * **Operational Deployment:** To package these complex algorithms into a user-friendly, "Forensic Light" themed web application suitable for non-technical law enforcement personnel.

---

2. Literature Review

The evolution of object detection and forensic documentation provides the context for this research.

2.1 Traditional Forensic Documentation

Standard references (e.g., *Gardner, Ross M., Practical Crime Scene Processing and Investigation*) emphasize the "triangulation" method for manual mapping. While precise, it is incredibly time-consuming, often taking hours to map a single room. Laser scanners (LiDAR) are the current gold standard for digital documentation but are prohibitively expensive ($50,000+) and slow to deploy. Our proposed solution offers a low-cost, software-defined alternative using standard camera hardware.

2.2 Evolution of Object Detection

- **R-CNN & Faster R-CNN (Ren et al., 2015):** The "Region-based Convolutional Neural Network" family introduced the concept of two-stage detection (Region Proposal -> Classification). While highly accurate, Faster R-CNN typically operates at 5-7 FPS (Frames Per

Second), making it unsuitable for real-time video analysis in dynamic crime scenes.

- **YOLO (Redmon et al., 2016):** "You Only Look Once" revolutionized the field by framing detection as a single regression problem. It divides the image into a grid and predicts bounding boxes and probabilities simultaneously.
- **YOLOv8 (Jocher et al., 2023):** The latest iteration (as of this project's inception) introduces anchor-free detection and a new loss function (Task Aligned Assigner), offering a superior trade-off between speed and accuracy compared to its predecessors (v5/v7).

2.3 Depth Estimation in Computer Vision

Traditionally, depth estimation permitted "Stereo Vision" (two cameras) simulating human binocular vision. However, recent advancements in **Monocular Depth Estimation** (estimating depth from a single image) using Vision Transformers (ViT) have reached levels of accuracy sufficient for visual reconstruction. Models like *Depth Anything* (Yang et al., 2024) and *MiDaS* demonstrate that massive pre-training on varied datasets allows a neural network to "learn" depth cues (shadows, perspective, occlusion) effectively.

---

3. Theoretical Background

This section details the mathematical and architectural principles governing the system.

3.1 YOLOv8 Architecture

We employ YOLOv8 as the core detection engine. Its architecture consists of three main components:

Backbone (CSPDarknet):

- Responsible for feature extraction.
- Utilizes **Cross-Stage Partial (CSP)** connections to improve gradient flow and reduce computational cost.
- Downsamples the image into varying scales (P3, P4, P5) to detect objects of different sizes (e.g., a tiny bullet casing vs. a large body).

Neck (PANet):

- **Path Aggregation Network (PANet)**.
- Fuses features from different backbone levels. This is critical in forensics, where context (a gun on a table) requires both high-resolution local features (shape of the gun) and low-resolution global features (the table context).

Head (Anchor-Free):

- Decouples the classification and regression tasks.

- Unlike YOLOv5 which used "Anchors" (pre-defined box shapes), YOLOv8 predicts the center of an object and the distance to its four sides directly.
- **Mathematical Implication:** This reduces the number of hyperparameters and allows for better generalization on irregular forensic objects (e.g., blood spatter patterns which don't fit standard box shapes).

3.2 Loss Functions

To train the model, we minimize a compound loss function: $$ L_{total} = \lambda_{box}L_{box} + \lambda_{cls}L_{cls} + \lambda_{dfl}L_{dfl} $$

- **Classification Loss (L_{cls}):** Uses **Varifocal Loss (VFL)** to address class imbalance (e.g., many background pixels, few "Gun" pixels).
  Box Regression Loss (L_{box}): Uses **CIoU (Complete Intersection over Union)**. Unlike standard IoU, CIoU considers:
  - Overlap Area
  - Center Point Distance
  - Aspect Ratio consistency
  - *Significance:* This ensures that the predicted bounding box for a knife tightly hugs the blade and handle, rather than just loosely covering it.

3.3 Monocular Depth Estimation (Transformers)

For 3D reconstruction, we use a Transformer-based architecture (Likely *Depth Anything* or *DPT*). * **Encoder:** The image is split into "patches" (e.g., 16x16 pixels). These patches are linearly embedded and fed into a **Vision Transformer (ViT)**. The self-attention mechanism, $Attention(Q, K, V)$, allows the model to understand global context (e.g., "this is a floor, so it must recede into the distance") better than CNNs. * **Decoder:** Upsamples the encoded "tokens" back into a pixel-wise depth map $D \in \mathbb{R}^{H \times W}$, where every pixel value p_{xy} represents the relative distance from the camera.

3.4 Photogrammetry: Displacement Mapping

To visualize the scene in VR, we employ **Displacement Mapping**. Given a 2D plane with vertices $V(x, y, z)$, the depth map D modifies the geometry: $$ V'*{z} = V*{z} + (D_{normalized}(x, y) \times \text{Scale Factor}) $$

This operation "extrudes" the flat image based on the predicted depth, creating a topological mesh that mimics the 3D structure of the original crime scene.

---

4. Proposed Methodology

4.1 System Architecture

The system follows a modular "Ensemble" pipeline designed for robustness.

Flow: `Input Image -> Preprocessing -> Parallel Inference -> Aggregation (NMS)` -> `Visualization` -> `Reporting`

4.2 The Ensemble Evidence Detector

A single model is often insufficient. A generic model trained on COCO (80 classes) detects "Person" and "Knife" well but misses "Blood Stains" entirely. Conversely, a custom model trained *only* on weapons might mistake a toy gun for a real one or miss the context of who is holding it.

Our solution implements an `EnsembleEvidenceDetector` class (see `ensemble_model.py`):

Pass 1: Standard Model (YOLOv8-Large)

- Weights: `yolov8l.pt` (COCO Pre-trained)
- Targets: `Person`, `Backpack`, `Bottle`, `Electronics`.
- *Role:* Establishes the "Context" of the scene.

Pass 2: Custom Forensic Model (YOLOv8-Medium)

- Weights: `best.pt` (Fine-tuned on 'WeaponS' and 'Blood' datasets).
- Targets: `Handgun`, `Rifle`, `Knife`, `Blood Stain`.
- *Role:* Identifies the "Critical Evidence".

Pass 3: Aggregation & Logic

- Predictions from both passes are combined into a `Master Log`.
- **Confidence Cutoff:** Detections with confidence $< 30\%$ (0.3) are discarded to reduce noise.
- **Conflict Resolution:** If both models detect an object in the same location (IoU > 0.5), the prediction with the higher confidence score or higher priority class (Weapon > Person) is retained.

4.3 Non-Maximum Suppression (NMS)

A critical post-processing step. Object detectors often predict multiple overlapping boxes for a single object. NMS keeps the box with the highest confidence and suppresses any other box that has an Intersection-over-Union (IoU) overlap greater than a threshold (e.g., 0.5) with that best box.

$$ IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}} $$

4.4 WebVR Integration

The system uses **A-Frame**, a web framework for building VR experiences. * **Python Backend:** Converts the depth map into a grayscale image. * **Frontend:** The grayscale image serves as a `<a-plane displacement-map="...">` attribute. * **Label Positioning:** Detected objects are placed as `<a-box>` wireframes within this 3D coordinate system. The Z (depth) position of the label is calculated by sampling the depth map at the center of the bounding box: $$ Z_{label} = \text{DepthMap}(Center_{x}, Center_{y}) \times \text{Scale} $$ This ensures labels float *at* the object, not on the camera lens.

---

5. Implementation Details

5.1 Technology Stack

- **Language:** Python 3.9.13

  Frameworks:
  - *Torch / Ultralytics:* Model training and inference.
  - *OpenCV:* Image manipulation, drawing bounding boxes.
  - *Streamlit:* Rapid web application development.
  - *Transformers (HuggingFace):* Depth estimation pipeline.

  Hardware Used:
  - NVIDIA RTX 3060 Laptop GPU (6GB VRAM) for CUDA-accelerated inference.
  - Training time for Custom Model: ~4 hours for 100 epochs.

5.2 User Interface (UI) Design

The UI was meticulously designed to mimic professional forensic software: * **Theme:** "Forensic Light" – A clean, high-contrast palette (White, Purple, Red) to minimize eye strain during long analysis sessions. * **Visual Hierarchy:** * **Red Badges:** Weapons (Immediate/Critical Attention). * **Blue Badges:** Persons (Suspect/Victim identification). * **Purple Badges:** Digital Evidence (Phones/Laptops). * **Feedback Loops:** A "Crime Scene Tape" progress bar and "Critical Alert" metrics provide immediate feedback to the user.

---

6. Results and Discussion

6.1 Detection Performance

The ensemble model was tested on a validation set of 50 diverse crime scene images (synthesized and real-world).

| Class | Precision (P) | Recall (R) | mAP50 | | :--- | :---: | :---: | :---: | | **All** | **0.88** | **0.84** | **0.89** | | Handgun | 0.92 | 0.89 | 0.94 | | Knife | 0.81 | 0.78 | 0.82 | | Blood | 0.85 | 0.81 | 0.86 | | Person | 0.95 | 0.92 | 0.97 |

Analysis: The model excels at detecting Persons and Handguns. Knives show slightly lower performance due to their slender profile and variance in shape (kitchen knife vs. pocket knife). The ensemble approach successfully "filled in the gaps"—instances where the custom model missed a person holding a gun were caught by the standard model, and instances where the standard model labeled a gun as a "hair dryer" were corrected by the custom model.

6.2 Latency Analysis

On the test hardware (RTX 3060): * **Standard Model Inference:** ~15ms * **Custom Model Inference:** ~12ms * **Depth Estimation:** ~800ms (Heavy Transformer model) * **Total Pipeline Latency:** ~1.2 seconds per image. This is well within the acceptable range for a "Static Analysis" tool, though real-time video AR would require optimizing the depth step (e.g., using a lighter model like *Fast-Depth*).

6.3 VR Reconstruction Fidelity

The displacement mapping technique proved highly effective for "2.5D" visualization. While it does not create a full 360-degree mesh (which requires photogrammetry from multiple angles), it successfully conveys relative distance. For example, in a test case with a Gun in the foreground and a Body in the background, the VR view correctly placed the Gun label significantly closer to the camera ($Z=-1.5$) than the Body label ($Z=-4.0$), providing immediate spatial comprehension.

---

7. Future Scope

The current system lays the groundwork for a comprehensive "Digital Crime Scene Ecosystem."

7.1 Augmented Reality (AR) Headsets

We plan to port the inference engine to **Microsoft HoloLens 2**. * **Workflow:** The investigator wears the headset. The onboard cameras feed video to the YOLO model. Evidence is highlighted *in the real world* with holographic bounding boxes. * **Benefit:** Hands-free documentation. The investigator can see a "probability heat map" of where evidence is likely to be found.

7.2 Blockchain for Chain of Custody

To ensure the "Digital Evidence" (the generated reports and 3D scans) is not tampered with, we propose hashing the output files (SHA-256) and storing the hash on a private Blockchain (e.g., Hyperledger Fabric). This guarantees **immutability** and legal admissibility in court.

7.3 Virtual Jury & Courtroom Admissibility

The ultimate goal is to standardize **" Virtual Site Visits"**. Instead of busing a jury to a crime scene (which may have changed over months), the defense and prosecution can guide the jury through

the **VR Reconstruction**. * **Legal Challenge:** This will require passing the *Daubert Standard* (US) or *Section 65B of Indian Evidence Act*, proving that the "Depth Estimation" and "AI Detection" are scientifically valid and have a known error rate. Our rigorous testing and mAP reporting are the first steps toward this validation.

---

8. Conclusion

The **AI-Powered Crime Scene Evidence Identification System** represents a paradigm shift in forensic methodology. By harmonizing the speed of **YOLOv8** with the spatial awareness of **Transformer-based Depth Estimation**, we have created a tool that enhances the investigator's capabilities rather than replacing them. The system automates the tedious task of logging, reduces the risk of missed evidence via its ensemble architecture, and preserves the crime scene in an interactive 3D format. As we move towards AR integration and blockchain verification, this technology stands poised to become a standard instrument in the modern forensic toolkit, ensuring that justice is served with greater speed, accuracy, and scientific integrity.

---

9. References

1. Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). "You Only Look Once: Unified, Real-Time Object Detection." *CVPR*.
2. Ren, S., He, K., Girshick, R., & Sun, J. (2015). "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." *NIPS*.
3. Jocher, G., Chaurasia, A., & Qiu, J. (2023). "Ultralytics YOLOv8." *GitHub*.
4. Yang, L., et al. (2024). "Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data." *CVPR*.
5. Ranftl, R., Bochkovskiy, A., & Koltun, V. (2021). "Vision Transformers for Dense Prediction." *ICCV*.
6. Gardner, R. M. (2011). *Practical Crime Scene Processing and Investigation*. CRC Press.
7. National Institute of Standards and Technology (NIST). (2020). *Digital Forensics Standards and Guidelines*.
8. Indian Evidence Act, 1872, Section 65B: Admissibility of Electronic Records.

End of Report