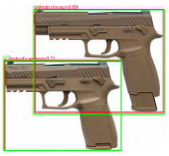
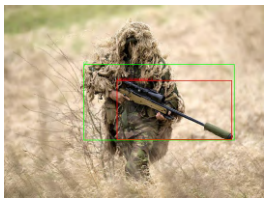
BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

http://fit.vut.cz

Project practice 2025/2026

Development of a Forensic Analytical Tool

Daniel Čerešňa*

Abstract

In this work, we explore the capabilities of state-of-the-art zero-shot object detection models as a viable alternative for violence detection in forensic contexts with a focus on weapons.
We conduct a comprehensive assessment of several candidate models, examining their detection performance across different violent scenarios including weapons, altercations and knife violence, their capability to generalize to unfamiliar visual environments, and their computational efficiency in real-world scenarios. In addition, we analyze the limitations of current zero-shot approaches. The results of this study provides valuable insight into the feasibility of applying zero-shot object detection in forensic settings and offer guidance for integrating such technologies into operational tools used by police forces. In this work, we found that the combination of performance and precision of the OWLv2 model will be the best to use in forensic contexts. It achieved the accuracy of 71% and recall of 91% showing that the zero-shot object detection models are a viable alternative for violence and general object detection used by police forces. The end of the work includes integrating the Developped Python module into a Forensic Analytical Tool made with C++ through pybind.

Keywords: Zero-shot learning — Weapon detection — Object detection

Supplementary Material: Demonstation video — Downloadable code

*xceresd00@stud.fit.vut.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

The increasing number of firearms violence forces us to think of and study possible solutions to these challenges. The growing infrastructure of closed-circuit television (CCTV) systems and police forces using body cameras helps address this challenge by providing a possible monitoring device for such events.

The increasing number of video data sources has made it impractical for human operators to manually inspect all available footage for signs of weapons or violent behavior. The widespread adoption of body-worn cameras by police forces provides a rich source of real-time visual data capturing critical incidents, including weapon presence, physical altercations, and knife-related violence. These systems operate in uncontrolled environments, ranging from low-light urban settings to rapidly evolving confrontations, creating both challenges and opportunities for automated violence and threat detection.

Yet, this vast footage remains underutilized without advanced tools to sift through it efficiently, prompting our study into zero-shot detection as a practical way to unlock its forensic value.

Our solution explores the use of zero-shot and open-vocabulary object detection models for detecting weapons and visually observable violent behavior in images and videos, with a particular focus on firearms. The approach aims to minimize the need for

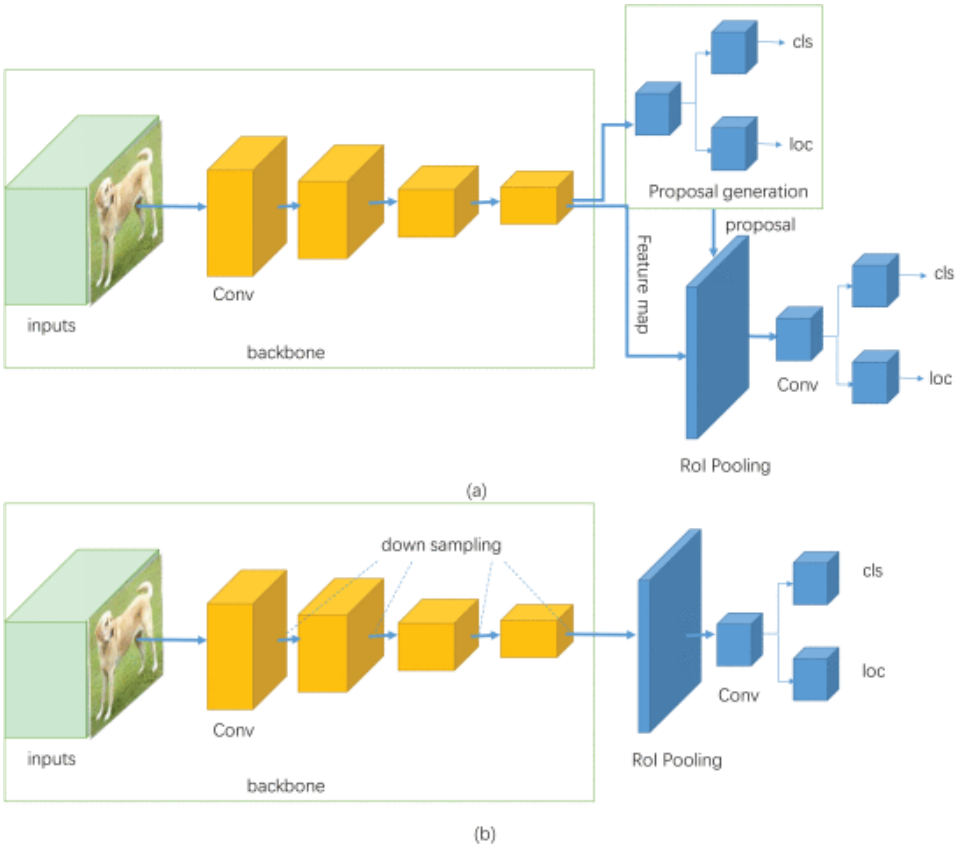


Figure 1. Architecture comparison of one and two stage detectors. [4]

task-specific training data by avoiding fully supervised detectors trained on fixed weapon classes and instead evaluating modern vision–language models capable of localizing objects and violent cues based on textual prompts.

The work follows a multi-stage process. First, several candidate zero-shot detection models are evaluated on a small, curated dataset to identify those most suitable for real-time and forensic scenarios involving weapons or violence with weapons such as knives. The selected models are then optimized through threshold tuning and assessed on both video data and a larger annotated image dataset. Runtime performance and detection accuracy are evaluated separately to reflect real-world deployment requirements. This approach enables flexible detection of violent scenarios, while maintaining an emphasis on firearm-related threats and balancing detection quality, speed, and practical applicability.

2. Existing Solutions

2.1 Background and Tradiditional Methods

Weapon and violence detection in images and video streams is commonly approached using modern object detection architectures, which can be broadly divided into two-stage and one-stage detectors. Two-stage detectors, such as Faster R-CNN and its variants, first generate region proposals and then classify each proposed region. This design allows for more precise localization and classification, which can be beneficial when detecting small or partially occluded objects such as firearms or knives. As a result, two-stage methods are often favored in scenarios where accuracy is prioritized over inference speed. However, their multi-step processing pipeline introduces higher computational cost, making real-time deployment on high-resolution video streams more challenging.

One-stage detectors, including the YOLO family and SSD-based models, perform object localization and classification in a single forward pass. These models are optimized for speed and are widely used in real-time surveillance and video analytics systems. Recent versions of one-stage detectors achieve competitive accuracy while maintaining high frame rates, making them suitable for time-critical applications such as live monitoring of violent incidents. Nevertheless, conventional one-stage detectors are typically trained on fixed sets of object classes and require extensive labeled datasets, limiting their flexibility when encountering novel weapon types or diverse violent scenarios.

These models combine the efficiency of one-stage or transformer-based detection architectures with semantic understanding derived from large-scale image–text pretraining. Unlike classical detectors, they

do not require retraining when new weapon categories or violent concepts are introduced, enabling detection based on textual descriptions alone. This shift reflects a broader trend toward adaptable detection systems that can generalize across different environments, object types, and forms of visual violence.

Similarly, many modern security systems use trained object detectors like YOLO or Faster R-CNN that are tuned to specific weapon categories. These approaches rely on large annotated datasets of weapon images and learn a fixed set of classes. While effective for known weapon types under expected conditions, they struggle to generalize to novel weapons or atypical appearances outside the training distribution. In particular, detectors like YOLO [12] or DETR [1] operate in a closed-set manner: they can only recognize objects belonging to the classes seen during training. This limitation poses a problem for surveillance scenarios, where new weapon designs or unanticipated poses may appear.

2.2 Vision-Language Models

To address closed-vocabulary limitations, recent research has turned to vision-language models that enable open-vocabulary, zero-shot object detection. The key idea is to leverage models pretrained on large-scale image–text data so that the system can relate visual features to arbitrary textual queries. A prime example is OpenAI's CLIP, which was trained on 400 million image-text pairs [11]. CLIP embeds images and textual descriptions into a shared latent space such that corresponding images and phrases have similar embeddings [2]. Because of this, CLIP can zero-shot identify objects it has never been trained on by simply providing a new text label. In practice, zero-shot detection can be performed by encoding a class name (or phrase) with CLIP's text encoder and comparing it with features extracted from image regions. For instance, a common approach splits an image into many overlapping windows or proposals, computes a CLIP image embedding for each patch, and then scores these embeddings against the target class embedding. This produces a relevance heatmap over the image. High-response regions indicate likely object locations; bounding boxes can then be drawn around the top-scoring regions. In this way, CLIP enables object localization and detection without any further training on the specific classes of interest.

Beyond CLIP, several models explicitly integrate language into detection. GLIP (Grounded Language-Image Pre-training) [7] trains on 27 million images with paired captions and object labels, learning to associate phrases with image regions. GLIP achieved strong zero-shot and few-shot performance by merging the supervision from captions and detection

2.3 Open Vocabulary Detectors

Open-vocabulary object detectors represent a recent shift in object detection research, aiming to overcome the closed-set limitations of traditional one-stage and two-stage detectors. This work focuses on state-of-the-art open-vocabulary detection models proposed within the past two to three years, which integrate language supervision into modern detection architectures. These models leverage large-scale vision–language pretraining to enable detection of previously unseen object categories based on textual descriptions alone.[3] By conditioning object localization on natural language, open-vocabulary detectors provide a flexible foundation for weapon and violence detection in dynamic and uncontrolled environments, where fixed class definitions are often insufficient.

2.3.1 OWL-ViT and OWLv2

Another line of work extends the CLIP paradigm to full object detectors. Google's OWL-ViT (and its successor OWLv2) are open-vocabulary detection models based on vision transformers. OWL-ViT applies a

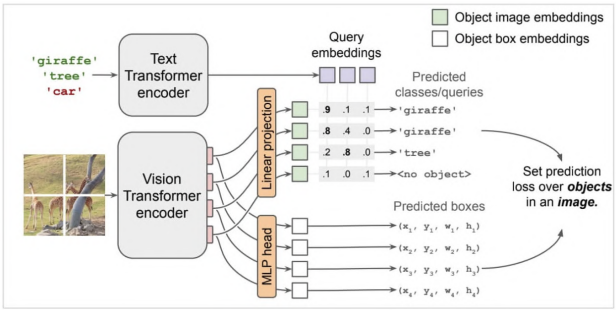


Figure 2. OWL-ViT Detection Pipeline [16]

Vision Transformer (ViT) backbone and uses a CLIP-like image encoder along with CLIP's text embeddings to score object proposals. In 2023, Minderer and others introduced OWLv2 [9] and a large-scale self-training recipe (OWL-ST) to dramatically scale open-vocabulary detection. The core idea is to use a smaller open-vocab detector (OWL-ViT) to generate pseudo-labels (bounding boxes and classes) on billions of unlabeled web images, then train a larger detector on this pseudo-labeled data. OWLv2 was trained on over a billion examples, enabling vast improvement in recognizing rare or unseen categories. For example, OWL-ST (self-trained OWLv2) improved average precision on LVIS rare categories (which have no box annotations) from 31.2% to 44.6%, a relative gain of 43%. Architecture-wise, OWLv2 is similar to OWL-ViT but adds an objectness classifier to the detection

head; this predicts the likelihood that a proposed box actually contains an object.

The objectness score can be used to filter or rank detections independently of the text query. In summary, OWLv2 demonstrates that scaling open-vocabulary detection to web-scale data yields far stronger zero-shot capability than training on conventional datasets alone. This model, like CLIP, uses a shared image–text embedding space: at inference, one encodes each candidate region and the class label text, and the model outputs matching scores and bounding boxes.

2.3.2 Grounding DINO

labels. [2] Building on GLIP [7] and transformer detectors, Grounding DINO is an open-world detection model that can take text input (class names or free-form descriptions) and detect arbitrary objects that match that text. Technically, Grounding DINO combines a DETR-like transformer detector (called DINO) with language grounding. It encodes the text prompt with a language encoder and visual features with a Swin Transformer backbone. A feature enhancer with deformable self-attention fuses multi-scale image features, while a language-guided query selection stage filters object queries based on the text. Finally, a cross-modality decoder jointly treats both visual and textual features to predict bounding boxes and labels. [8] In essence, Grounding DINO "introduces language to a closed-set detector" to allow open-set generalization. This model achieves impressive zero-shot results: for example, it reports 52.5 AP on COCO objects without any COCO training data, and it also handles referring expressions and attributes. In practical terms, a text query can be supplied as "person with a gun" or simply "handgun", and Grounding DINO will detect instances of this concept if present. Thus, vision-language detectors can recognize weapons from natural-language cues alone, without the need for specific firearm training images.

2.3.3 GroundVLP

GroundVLP, which stands for Grounding Vision-Language Pre-training, is a zero-shot model designed to perform visual grounding tasks, such as Referring Expression Comprehension (REC), without requiring explicit training on every new object. Rather than relying on exhaustive annotation, the model leverages the generalized knowledge inherent in Vision-Language Pre-training (VLP) models and Open-Vocabulary Detectors (OVD).

Technically, GroundVLP is based on the ALBEF architecture (Align Before Fuse) [6], which employs a contrastive loss to align image and text representations prior to combining them in a multimodal encoder. A key feature of the model is its dual-stream fusion approach. In the first stream, Grad-CAM is applied to the VLP model to generate semantic heatmaps that highlight regions of the image most relevant to the text query. [14] In parallel, the second stream uses an open-vocabulary detector to generate a set of candidate bounding boxes.

These two streams are then integrated using a weighted scoring strategy, which combines the detector's confidence with the intensity of the semantic heatmap to rank the candidates. This approach effectively bridges the gap between high-level language understanding and precise spatial localization, enabling the model to identify specific objects in a scene based solely on natural language descriptions. [15].

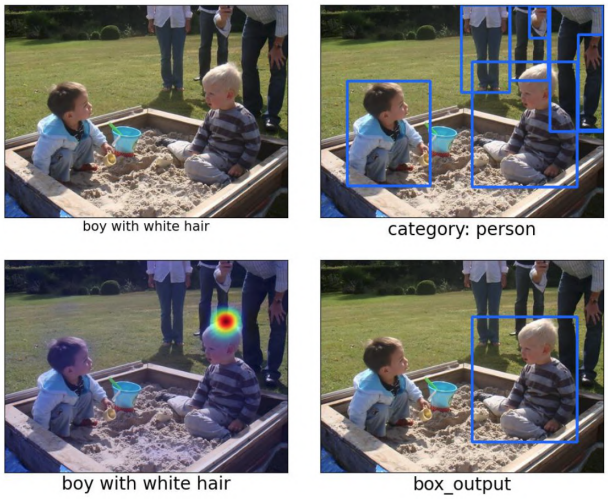


Figure 3. Example of GroundVLP thought process [15]

2.3.4 YOLOv8-World

YOLOv8-World-m is a single-stage open-vocabulary object detector with approximately 52M parameters, it was implemented using the Ultralytics Python framework. All images were resized to an input resolution of 640×640 pixels. Inference was performed using FP16 precision, a confidence threshold of 0.25. The model aligns image features with text embeddings to produce open-vocabulary bounding box predictions.

2.4 Model comparison

2.4.1 OWL-ViT and OWLv2

As the final detection model, we initially used the OWL-ViT model and later switched to the improved OWLv2 version. OWL-ViT and OWLv2 are transformer-based open-vocabulary object detectors, with OWLv2 containing approximately 300M parameters. Input images were kept at their original 1920×1080 resolution. Inference was performed using FP16 precision

and similarity-based scoring between region features and text embeddings, with a confidence threshold of 0.4 applied to filter predictions. The text prompts "a photo of a gun" and "a photo of a weapon" were used, and bounding boxes were filtered using the model's objectness and confidence scores.

2.4.2 Grounded SAM (GSAM): Grounding DINO with Segment Anything

The Segment Anything Model (SAM) [5] is a general-purpose segmentation model that can produce object masks from simple prompts, but it does not perform object detection or semantic classification on its own. To overcome this limitation, Grounded SAM (GSAM) combines Grounding DINO [8] with SAM to enable text-guided detection and segmentation in a single pipeline.

In GSAM, Grounding DINO is first used to detect objects based on a textual prompt, such as "gun" or "handgun". It outputs bounding boxes that correspond to regions in the image that match the given text description, even for object categories not explicitly seen during training. These bounding boxes are then used as input prompts for SAM, which generates precise segmentation masks for each detected object.

This approach allows GSAM to perform zero-shot instance segmentation, meaning that weapons can be localized and segmented without requiring weapon-specific training data or pixel-level annotations. Compared to using bounding boxes alone, GSAM provides more accurate spatial information, which can be useful in surveillance applications where precise localization of weapons is important. As a result, GSAM is well suited for zero-shot weapon detection tasks that require both flexibility and detailed object representation.

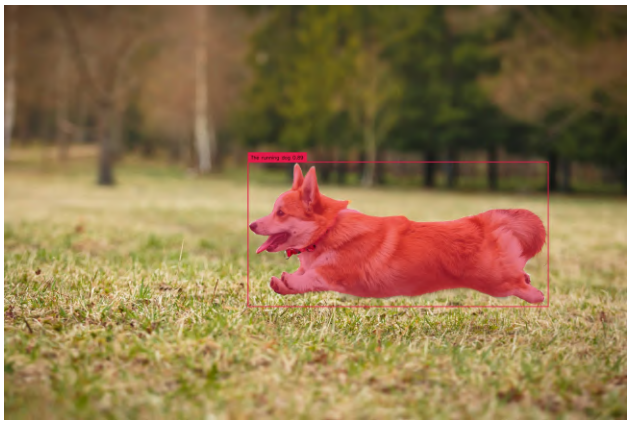


Figure 4. Example GSAM detection

3. Methodology

This section outlines the complete experimentation phase for violence detection, with a particular focus on weapons, in uncontrolled environments using zero-shot learning models and their evaluation methodology. Our approach extends beyond weapon detection alone to identify broader violent behaviors while emphasizing scenarios that involve weapons.

The dataset consists of 14 curated images depicting people in scenarios involving weapons. Weapon categories include pistols, concealed pistols, assault rifles, knives, and an umbrella for false detections. Among the dataset pictures, there are 2 pictures of people with umbrellas holding them similarly as one would hold a weapon. The pictures were included to test for false positives and how that might affect their use in forensic contexts.

3.1 Scenario Definition

To systematically evaluate our models, we first created controlled testing scenarios. Initial experiments were conducted on static images, including frames from movies and images of individuals holding firearms. Subsequently, we extended the testing to video sequences to assess the performance of the model in dynamic, real-world situations.

3.2 Candidate Model Evaluation

The first part of picking out the zero-shot learning models was finding 5 candidate text-query models implemented for object detection, sorted by their popularity and usability on the Github seach page.

This stage was purely focused on reducing the number of candidate models from 5 to 3 before diving deeper with tweaking and refining the detection accuracy. The models were evaluated on a small dataset of 14 pictures containing different subclasses of weapons.

3.2.1 Evaluation Process

Each model was run on the same dataset and the resulting output was presented in a table ?? in a binary found/not found format. A detection was considered correct if the model produced at least one bounding box overlapping the target object. The runtime of the model did not play an important role in the evaluation. Due to the limited size of the dataset, results should be interpreted as indicative rather than definitive.

3.2.2 Selection

Model selection was based on the total number of correct detections across the dataset. Although control images containing umbrellas were included to evaluate false positives, all evaluated models incorrectly classified the umbrellas as potential weapons, limiting their usefulness as a discriminatory factor at this stage. No formal quantitative metrics such as mAP, precision, or recall were computed, as this evaluation was

Table 1. Comparison of model outputs across different categories. C - Concealed Weapon; F - Fight; G - Gun; V - Violence; K - Knife; FP - False Positive (eg. Umbrella)

Category	C1	C2	C3	C4	F1	F2	G1	G2	G3	V1	NG1	FP1	FP2	K1	Total
GSAM	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	✓	12
GroundVLP	×	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	✓	9
YOLOv8-World	✓	×	×	✓	✓	×	×	✓	✓	×	×	×	×	×	5
OWL-ViT	✓	✓	✓	✓	×	×	✓	✓	✓	×	✓	×	×	×	8
Grounding DINO	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	×	×	✓	11

Figure 5. Example images from the dataset, including concealed weapons, guns and an umbrella

intended to be exploratory rather than a comprehensive benchmark.

Table 2. Inference speed comparison

Model	Speed
GSAM	1.2 f/s
OWL-ViT	13 f/s
Grounding DINO	3.1 f/s
OWLv2	3 f/s

This process reduced the set of candidate models from five to three. Based on the correct detection counts of the evaluated images, the models were classified as follows: GSAM (12), Grounding DINO (11) and OWL-ViT (8). OWL-ViT was selected over GroundVLP due to its compatibility with the newer OWLv2 model and its established use in recent literature.

3.3 Model Evaluation and Optimization for Video Data

This phase focused on improving the 3 chosen models by adjusting their input configuration as well as measuring the inference speed on various datasets. The threshold tuning was performed after the initial model evaluation and was done only on the 3 chosen models.

The phase also included transferring the environment from a local machine to a remote server with 4 Nvidia RTX A5000 graphics cards used to detect weapons in videos faster.

3.3.1 Threshold Adjustment

Adjusting of the threshold of each model individually was done with a manual, trial and error approach. Each model was independently tuned, resulting in each model having different threshold values. The goal of the tuning was to improve the precision and recall of each model to its greatest potential. The results of the threshold adjustment were tested on a *Weapon Detection Dataset*. [13]

3.3.2 Hardware and Execution Enviroment

All experiments were conducted on a server equipped with four NVIDIA RTX A5000 GPUs. While multiple GPUs were available, each video was processed on a single GPU at a time, with different videos assigned to different GPUs to allow parallel experimentation. This setup ensured consistent runtime measurements per model. All models were executed with GPU acceleration, and runtime performance was evaluated under identical hardware conditions.

3.3.3 Video-Based Evaluation

The first evaluation stage was performed on a set of videos to assess runtime behavior and practical detection performance. The videos included clips from movies as well as real-world scenarios [10], which reflect the intended deployment environment of the models. All videos were processed at their original resolutions without resizing. Runtime performance was measured as the average frames per second (FPS) over the full duration of each video. Detection quality during this stage was qualitatively assessed through visual inspection, focusing on detection stability and consistency rather than formal metrics.

Table 3. OWLv2 Evaluation results for different IOU and threshold settings.

IOU	Threshold	TP	FP	FN	Precision	Recall
0.5	0.4	180	61	39	0.74	0.82
0.5	0.35	191	89	28	0.68	0.87
0.5	0.3	198	131	21	0.60	0.90
0.4	0.4	186	55	33	0.77	0.85
0.4	0.35	198	82	21	0.71	0.90
0.4	0.3	205	124	14	0.62	0.94
0.3	0.4	188	53	31	0.78	0.85
0.3	0.35	200	80	19	0.71	0.91
0.3	0.3	206	123	13	0.62	0.94

Figure 6. Screenshot from a dataset video showing detected weapons in an uncontrolled enviroment. Detected by OWLv2.

3.3.4 Model Selection After Video Evaluation

Following the subjective video-based evaluation, a single model was selected for further experimentation. OWL-ViT was chosen due to its higher inference speed, which is a critical factor in time-sensitive applications such as law enforcement scenarios. While other models occasionally produced more precise detections, the qualitative differences were not considered substantial enough to outweigh the runtime advantages observed with OWL-ViT.

However, during this stage, it was observed that OWL-ViT performed poorly on low-resolution video inputs, failing to produce reliable detections on a publicly available low-resolution video dataset. To address this limitation, the evaluation was continued using the newer OWLv2 model, which builds upon OWL-ViT and offers improved robustness while maintaining similar open-vocabulary capabilities.

3.3.5 Image-Based Quantitative Evaluation

The final evaluation stage was conducted using OWLv2 on an image dataset consisting of 143 annotated images sourced from Kaggle. The dataset includes a variety of weapon categories such as automatic rifles, bazookas, handguns, knives, grenade launchers, shotguns, submachine guns, sniper rifles, and swords. Although fine-grained labels were available, all detections were evaluated under a unified gun/weapon class, and some images contained multiple weapons. Ground-truth annotations were provided in YOLO-format bounding boxes, enabling quantitative evaluation.

Detection performance was measured using precision and recall, calculated using IoU-based matching. Multiple IoU thresholds (0.3, 0.4, and 0.5) were evaluated in combination with different detection thresholds. The resulting true positive, false positive, and false negative counts are reported in a dedicated table.

3.3.6 Separation of Evaluation

The evaluation methodology intentionally separated runtime and accuracy objectives. Video-based experiments prioritized real-time performance and practical behavior, assessed through FPS measurements and qualitative inspection, while image-based experiments focused on quantitative detection accuracy, measured using precision and recall. This separation reflects the differing requirements of real-world deployment versus controlled accuracy evaluation and allows each model to be assessed within an appropriate experimental context.

3.4 Error Analysis

The error analysis focuses on understanding the most common failure cases observed during both video-based and image-based evaluation. One recurring issue was the reduced detection performance on low-resolution inputs, where weapons appeared small or lacked sufficient visual detail. In such cases, the models frequently failed to produce any detections or produced low-confidence predictions that were filtered out by the selected thresholds.

Another common failure mode involved partial occlusion of weapons. When only a small portion of a weapon was visible, such as a concealed pistol or a knife partially covered by clothing, detections were often missed. This behavior was observed across all evaluated models and is likely caused by the limited visual cues available for reliable vision–language alignment. False positives were also observed, most notably in

Figure 7. False positive detection detected by GSAM

images containing umbrellas. Despite being included as control samples, all models incorrectly classified objects resembling weapons such as umbrellas, when held in a manner visually similar to firearms. This highlights a limitation of zero-shot detection approaches, where semantic similarity inferred from visual appearance may override object-specific details.

In video sequences, detection instability was observed in some cases, where detections intermittently appeared and disappeared across consecutive frames. Since the evaluated models process frames independently, no temporal consistency constraints were enforced, leading to flickering detections in challenging scenarios.

4. Integration of the Detection Module

The integration of the violence and weapon detection module into the forensic analysis tool was carried out by establishing a communication bridge between the existing C++ codebase and the Python inference module. The forensic tool, originally implemented entirely in C++, required the ability to invoke advanced computer vision models written in Python without disrupting its existing workflow. To achieve this, we employed pybind, which allows seamless binding between C++ and Python, enabling function calls and data exchange across the two languages. This approach preserved the high-performance nature of the C++ tool while leveraging the flexibility and rapid prototyping capabilities of Python for model inference.

The Python module, referred to as InferenceCV Python, serves as the core of the detection pipeline. When the C++ tool identifies a segment of interest in an image or video, it passes the corresponding data to the Python module through the pybind interface. Within the module, the zero-shot learning models perform both violence detection and weapon-focused analysis, generating metadata that characterizes the detected events, such as object type, probability scores, and contextual flags. This metadata is then transmitted back to the C++ layer, where it is incorporated into the forensic tool's analytical pipeline for further processing and visualization.

This integration design ensures modularity and maintainability of the system. By isolating the computationally intensive detection processes within a dedicated Python module, the main C++ application remains stable and responsive, while still benefiting from cutting-edge machine learning models. Additionally, this approach facilitates future updates and experimentation with different models or detection strategies, as modifications can be confined to the Python module without requiring extensive changes to the underlying

C++ codebase.

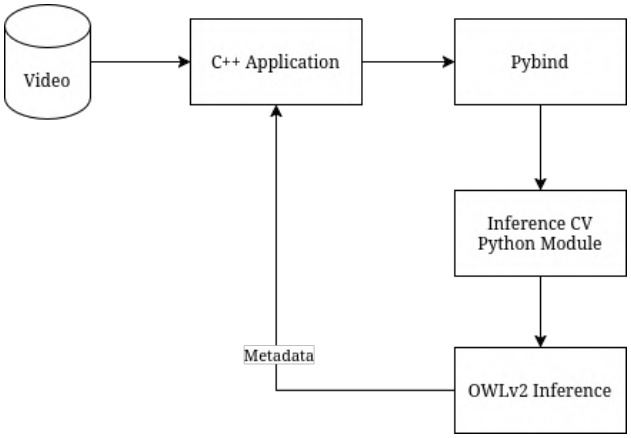


Figure 8. Application communicating with the developped module.

5. Conclusion

This work explored the capabilities of state-of-the-art zero-shot object detection models as a viable alternative for violence and weapon detection in forensic contexts, conducting a multi-stage assessment of models such as OWL-ViT, OWLv2, Grounding DINO, and GSAM. By moving beyond traditional closed-set detectors that require extensive task-specific training data, the study demonstrated how vision-language models can localize objects based on textual prompts alone, offering the flexibility needed for the uncontrolled and dynamic environments typical of police work. The research followed a rigorous process of evaluating candidate models on curated datasets and video sequences, optimizing performance through manual threshold tuning and hardware acceleration using NVIDIA RTX A5000 GPUs. Ultimately, the study concluded that the OWLv2 model provides the most effective balance of precision and inference speed for forensic applications, and this capability was successfully integrated into a high-performance C++ forensic tool via a pybind11-based communication bridge.

The quantitative results underscore the practical utility of these models, with initial evaluations on a small, curated dataset showing GSAM and Grounding DINO achieving high detection counts of 12 and 11 out of 14 scenarios, respectively. Inference speed measurements identified OWL-ViT as the fastest model at 13 FPS, followed by Grounding DINO at 3.1 FPS and the more robust OWLv2 at 3 FPS. Detailed testing of OWLv2 on a 143-image weapon dataset revealed that at an IoU of 0.3 and a detection threshold of 0.35, the model achieved a precision of 71% and a recall of 91%. Further optimization showed that the model could reach a peak recall of 94% at lower thresholds, demonstrating its high sensitivity in identifying critical threats like firearms and knives even in challenging conditions.

The primary contributions of this work include a comprehensive performance benchmark of modern open-vocabulary models specifically for forensic weapon detection, filling a critical gap in research regarding their real-world deployment. Additionally, the study provides a technical blueprint for the modular integration of Python-based machine learning modules into existing C++ forensic infrastructures using pybind11, preserving system performance while enabling rapid adoption of new AI technologies. Finally, this research verifies the forensic viability of zero-shot learning, proving that police forces can effectively utilize models trained on web-scale data to recognize unfamiliar weapon designs and violent cues without the need for costly, domain-specific dataset annotation. Future research will focus on addressing the primary limitations identified during the evaluation, particularly the lack of temporal consistency in video-based detection that leads to flickering results across consecutive frames. Integrating object tracking algorithms or temporal windowing techniques would likely stabilize detections and reduce noise in dynamic forensic footage, ensuring a more reliable user experience for law enforcement operators. Another critical area for improvement is the model's performance on low-resolution inputs, which currently hampers reliability in scenarios where weapons appear small or lack sufficient visual detail. Future iterations could explore the integration of lightweight super-resolution pre-processing or the development of models specifically robust to the visual artifacts of body-worn and CCTV cameras.

To further reduce false positives, such as the observed instances where umbrellas were incorrectly classified as weapons, advanced prompt engineering or multi-stage verification modules could be employed to improve semantic discrimination. This might involve refining textual queries to include negative prompts or contextual cues that help the model distinguish between similar-shaped objects. Beyond technical refinements, the scope of detection can be expanded to include a wider array of forensic markers and more complex criminal behaviors beyond the current focus on firearms and physical altercations. While the current system utilizes a robust C++ to Python bridge via pybind11 to maintain performance, future work will aim to further optimize these models for real-time execution on edge devices. This would enable immediate, on-device threat detection directly on police hardware, significantly reducing the time required for forensic

analysis in the field.

Acknowledgements

I would like to thank my supervisor Ing. Tomáš Goldmann Ph.D. for his help.

References

[1] CARION, N., MASSA, F., SYNNAEVE, G., USUNIER, N., KIRILLOV, A. et al. *End-to-End Object Detection with Transformers*. 2020. Available at: https://arxiv.org/abs/2005.12872.

[2] CHHIPA, P. C., DE, K., CHIPPA, M. S., SAINI, R. and LIWICKI, M. *Open-Vocabulary Object Detectors: Robustness Challenges under Distribution Shifts*. 2024. Available at: https://arxiv.org/abs/2405.14874.

[3] CHHIPA, P. C., DE, K., CHIPPA, M. S., SAINI, R. and LIWICKI, M. *Open-Vocabulary Object Detectors: Robustness Challenges under Distribution Shifts*. 2024. Available at: https://arxiv.org/abs/2405.14874.

[4] JIAO, L., ZHANG, F., LIU, F., YANG, S., LI, L. et al. A Survey of Deep Learning-Based Object Detection. *IEEE Access*. 2019, vol. 7, p. 128837–128868.

[5] KIRILLOV, A., MINTUN, E., RAVI, N., MAO, H., ROLLAND, C. et al. Segment Anything. *ArXiv:2304.02643*. 2023.

[6] LI, J., SELVARAJU, R. R., GOTMARE, A. D., JOTY, S., XIONG, C. et al. *Align before Fuse: Vision and Language Representation Learning with Momentum Distillation*. 2021. Available at: https://arxiv.org/abs/2107.07651.

[7] LI, L. H., ZHANG, P., ZHANG, H., YANG, J., LI, C. et al. *Grounded Language-Image Pre-training*. 2022. Available at: https://arxiv.org/abs/2112.03857.

[8] LIU, S., ZENG, Z., REN, T., LI, F., ZHANG, H. et al. *Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection*. 2024. Available at: https://arxiv.org/abs/2303.05499.

[9] MINDERER, M., GRITSENKO, A. and HOULSBY, N. *Scaling Open-Vocabulary Object Detection*. 2024. Available at: https://arxiv.org/abs/2306.09683.

[10] PINTÉR, P. *Airsoft Videos*.

[11] RADFORD, A., KIM, J. W., HALLACY, C., RAMESH, A., GOH, G. et al. *Learning Transferable Visual Models From Natural Language Supervision*. 2021. Available at: https://arxiv.org/abs/2103.00020.

[12] REDMON, G. F. You Only Look Once: Unified, Real-Time Object Detection. *CVPR*. 2016.

[13] SANYAL, S. *Weapon Detection Dataset*. Available at: https://www.kaggle.com/datasets/snehilsanyal/weapon-detection-test/data.

[14] SELVARAJU, R. R., COGSWELL, M., DAS, A., VEDANTAM, R., PARIKH, D. et al. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *International Journal of Computer Vision*. Springer Science and Business Media LLC. october 2019, vol. 128, no. 2, p. 336–359. Available at: http://dx.doi.org/10.1007/s11263-019-01228-7. ISSN 1573-1405.

[15] SHEN, H., ZHAO, T., ZHU, M. and YIN, J. GroundVLP: Harnessing Zero-shot Visual Grounding from Vision-Language Pre-training and Open-Vocabulary Object Detection. *ArXiv preprint arXiv:2312.15043*. 2023.

[16] ZAX, D. How to use OWL ViT – A groundbreaking zero-shot model. *Ezml.io*. 2023.