Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2024.056816

ARTICLE

MARIE: One-Stage Object Detection Mechanism for Real-Time Identifying of Firearms

Diana Abi-Nader[1], Hassan Harb[2], Ali Jaber[1], Ali Mansour[3], Christophe Osswald[3], Nour Mostafa[2,*] and Chamseddine Zaki[2]

[1]Computer Science Department, Faculty of Sciences, Lebanese University, Beirut, 146404, Lebanon

[2]College of Engineering and Technology, American University of the Middle East, Egaila, 54200, Kuwait

[3]Lab-STICC, CNRS UMR 6285, ENSTA-Bretagne, Brest, 29200, France

*Corresponding Author: Nour Mostafa. Email: nour.moustafa@aum.edu.kw

Received: 31 July 2024 Accepted: 24 October 2024 Published: 17 December 2024

ABSTRACT

Security and safety remain paramount concerns for both governments and individuals worldwide. In today's context, the frequency of crimes and terrorist attacks is alarmingly increasing, becoming increasingly intolerable to society. Consequently, there is a pressing need for swift identification of potential threats to preemptively alert law enforcement and security forces, thereby preventing potential attacks or violent incidents. Recent advancements in big data analytics and deep learning have significantly enhanced the capabilities of computer vision in object detection, particularly in identifying firearms. This paper introduces a novel automatic firearm detection surveillance system, utilizing a one-stage detection approach named MARIE (Mechanism for Real-time Identification of Firearms). MARIE incorporates the Single Shot Multibox Detector (SSD) model, which has been specifically optimized to balance the speed-accuracy trade-off critical in firearm detection applications. The SSD model was further refined by integrating MobileNetV2 and InceptionV2 architectures for superior feature extraction capabilities. The experimental results demonstrate that this modified SSD configuration provides highly satisfactory performance, surpassing existing methods trained on the same dataset in terms of the critical speed-accuracy trade-off. Through these innovations, MARIE sets a new standard in surveillance technology, offering a robust solution to enhance public safety effectively.

KEYWORDS

Firearm and gun detection; single shot multi-box detector; deep learning; one-stage detector; MobileNet; inception; convolutional neural network

1 Introduction

Homicide remains a significant threat to human life, with studies indicating a rising trend in homicide rates over the years, predominantly due to firearm incidents [1–3]. According to the latest global report by the United Nations Office on Drugs and Crime (UNODC), firearms were identified as the principal cause of homicides, accounting for 54% victims in 2017 [4]. This escalation corresponds with an increase in civilian possession of weapons, particularly noted in the United States, which

leads global statistics [5]. Notably, the United States has seen a marked increase in school-associated homicide incidents from 2009 to 2018, with the majority involving firearms [6]. For instance, a reduction in gun homicides in the state of Minneapolis within a single year catalyzed the creation of 80 jobs and generated an additional $9.4 million in sales for local businesses the following year [7]. On a sociological level, gun violence instills fear and contributes to psychological distress among victims and communities, exacerbating the broader social impact of this issue.

Gun violence remains a critically understudied area of public safety, necessitating further research to decrease gun-related fatalities and prevent such incidents [8]. One effective strategy to mitigate homicides is early incident detection, facilitating rapid police response. In this regard, an innovative approach was explored using a camera-embedded automatic firearm detection system. The system leverages Convolutional Neural Networks (CNNs), which are adept at feature selection and are position invariant, to perform object detection [9,10]. Despite their effectiveness, CNNs require extensive datasets and the use of Graphics Processing Units (GPUs) for precise training [9,11]. CNNs are generally divided into two classes: one-stage and two-stage detectors. One-stage detectors provide immediate predictions, whereas two-stage detectors create region proposals and extract features prior to making predictions. Notably, while one-stage detectors excel in speed, two-stage detectors typically offer greater accuracy [12].

Firearm detection is a nascent field of research. Initial studies on gun detection were conducted using X-ray and millimeter wave imaging with traditional machine learning techniques [13,14]. More recent research has employed CNNs to enhance detection capabilities, focusing particularly on two-stage detectors like the Faster-Region based CNN (R-CNN) using the Visual Geometry Group (VGG-16) classifier [15,16]. The primary challenges in developing a real-time automatic firearm detection system include achieving rapid detection without compromising accuracy, which is crucial for swift police intervention. Since the predominant challenge in object detection models is balancing speed and accuracy, the proposed research aims to optimize this trade-off specifically for firearm detection. Additional hurdles include constructing and labeling image datasets, a manual and time-consuming process.

This paper makes several significant contributions to the field of firearm detection through object detection technologies. Key contributions of the paper are as follows:

- **Innovative Detection Mechanism:** This paper introduces a one-stage object detection mechanism, named MARIE, specifically designed for real-time identification of firearms. The use of the Single Shot Multibox Detector (SSD) model is proposed, incorporating modifications to the base architecture by integrating MobileNetV2 and InceptionV2. This adaptation aims to enhance the model's efficiency and accuracy in detecting firearms.
- **Model Training and Fine-Tuning:** A pre-trained model was utilized from the Microsoft Common Objects in Context (COCO) dataset [17], and fine-tuned it using a specifically constructed and labeled image dataset by Olmos et al. [15]. This process tailors the model more closely to the nuances of firearm detection.
- **Focus on Held-Gun Types:** The proposed technique specifically addresses the detection of various types of held guns, refining the model's applicability to real-world scenarios where different firearm types are present.
- **Comparative Analysis:** The performance of the proposed modified SSD model is rigorously compared with existing methods, including traditional classifiers [18] and other deep learning models [15,19], all trained and tested on the same firearm dataset. This comparison highlights the strengths and potential improvements in the proposed approach.

The remaining of the paper is structured as follows. Section 2 shows related models used in object and firearm detection. Section 3 presents the architecture of the model used in the proposed work. Section 4 presents the classifier variations used. Section 5 explains the performance metrics and the obtained results. Finally, the paper is concluded and future directions are given in Section 6.

2 Related Work

Object detection algorithms have undergone significant evolution since their inception, advancing from basic traditional machine learning techniques to sophisticated deep learning networks [20]. A variety of methods have been developed to tackle object detection broadly [21–25]. While specific techniques have been crafted for the detection of particular objects, such as firearms and other weapons [26].

2.1 General Detection Techniques

Object detection models are primarily categorized into two types: two-stage and one-stage detectors. Among the two-stage detectors, the R-CNN family is notably popular. The initial R-CNN model was introduced in [21]. Enhanced object detection by reducing the regions analyzed to 2000 proposals, which are processed by a VGG-based CNN for feature extraction before being evaluated by a Support Vector Machine (SVM) to produce predictions. Subsequent developments led to Fast R-CNN, which increased processing speed by extracting image features prior to generating region proposals and replaced the SVM with a softmax layer [22]. The introduction of Faster R-CNN further advanced this model by substituting the selective search with a Region Proposal Network (RPN), rendering the model fully trainable and significantly improving both speed and accuracy [23]. Among its peers, Faster R-CNN has demonstrated superior accuracy [27].

The early one-stage detector "Overfeat" [12] uses the sliding window approach for feature extraction and shares it with the classification and localization layers to improve accuracy [28]. This approach affects the detection speed because it generates a large number of windows for processing [29]. Later, the fastest known model "You Only Look Once" (YOLO) was first introduced in [24]. It divides the input image into a 7×7 grid cells and predicts the bounding box and the class of an object using a single network. However, YOLO suffered from the detection of small objects. YOLOv2 was developed in order to address YOLO drawbacks while maintaining a real-time detection speed [30]. In [31], YOLOv3 used multi-scale detection to improve the previous versions. Similarly, the SSD model, proposed in [25], adopted VGG-16 as its base network and introduced anchor boxes from Faster R-CNN to enhance accuracy while maintaining real-time inference speeds. In this study, utilizing SSD for firearm detection is proposed to leverage its balance of speed and precision.

2.2 Firearm Detection Techniques

The current state of the art in firearm detection can be categorized as follows: traditional machine learning techniques, generic detection models tailored for firearm detection, and specialized models developed specifically for identifying firearms.

Earlier techniques in image feature extraction employed methods such as edge detection and multi-modal image matching, utilizing Radiation-Invariant Feature Transform (RIFT) and Scale-Invariant Feature Transform (SIFT) [13], along with basic local density descriptors [32]. Additionally, Haar-like features were used for feature extraction as noted in [14]. Subsequent studies [33,34] refined this approach by combining SIFT with K-means clustering before classification via Support Vector Machines (SVM). Another innovative framework was proposed in [35,36] for Red-Green-Blue (RGB)

images, which involved color segmentation using Speeded Up Robust Features (SURF) [35], further enhanced with K-means and the Harris interest point detector paired with Fast Retina Keypoint (FREAK) [36]. However, these techniques often faced limitations due to their sensitivity to noise [29] and slow inference speeds.

Initial studies employing Convolutional Neural Networks (CNNs) for firearm detection advocated for the use of transfer learning applied to Faster R-CNN with VGG-16 as the base network. These authors achieved high accuracy, surpassing that of traditional machine learning techniques [15,37]. Specifically, Verma et al. focused on detecting handguns in cluttered scenes [37], while Olmos et al. optimized the base classifier using both a sliding window approach and a Region Proposal Network (RPN) with Faster R-CNN, which yielded superior results [15]. Furthermore, another study [38] utilized the same dataset to implement the Overfeat classifier using TensorFlow, where Overfeat-3 achieved high accuracy but fell short in real-time detection capabilities. Additionally, Olmos et al. explored enhancements in detection accuracy by integrating binocular image fusion into Faster R-CNN [39]. In related research, the effectiveness of soft deep CNNs was examined to address computational resource limitations [19].

The authors of [40] explored several key aspects of designing and testing an automated detection system aimed at locating and identifying small firearms either left at a training range or found on a battlefield. The system they proposed utilizes a small unmanned aerial system (sUAS) equipped with a modern electro-optical (EO) sensor, which operates based on a convolutional neural network (CNN) trained specifically for this purpose. Furthermore, the system employs a YOLOv2 CNN framework, utilizing a ResNet-50 as the backbone network to train the model using ground truth data.

Recent studies have furthered the application of neural networks in firearm detection. Mehta et al. [41] employed YOLOv3 to detect handguns and conducted evaluations across various benchmark datasets. Romero et al. [42] and Basit et al. [43] developed models for human-firearm detection using the VGG-16 architecture; the former utilized YOLO while the latter implemented Faster R-CNN. Additionally, Iqbal et al. [26] introduced a novel two-stage system called Orientation Aware Object Detection (OAOD), which incorporates VGG-16 to specifically tackle challenges related to the angle variations and occlusions encountered in firearm detection.

While existing approaches provide effective solutions, results in this paper indicate a need for further improvement. In this paper, MARIE is proposed, which utilizes a Single Shot Multibox Detector (SSD) as a one-stage detector. This method not only offers enhanced speed in detection but also achieves accuracy comparable to that of Faster R-CNN [12,27]. The proposed methodology involves training the SSD using MobileNetV2 [44] and InceptionV2 [45] architectures on the dataset compiled by Olmos et al. [15]

3 Deep Learning Model

SSD is a deep learning model that utilizes multi-scale feature maps for making predictions [12] (see Fig. 1). It employs a base network for feature map extraction, followed by the application of convolution filters to detect objects. Drawing inspiration from Faster R-CNN, SSD incorporates aspect ratios in a manner that preserves real-time processing speeds by utilizing default boxes, which eliminate the need for resampling the features within them [25].

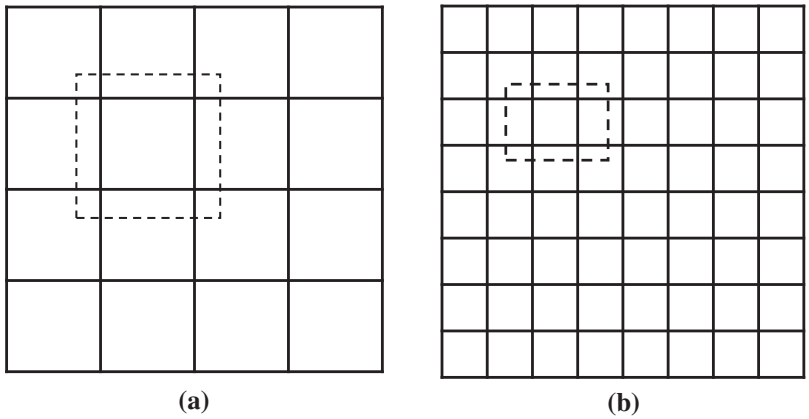

Figure 1: SSD multi-scale feature map: (a) feature map with 4×4, (b) feature map with 8×8

The SSD architecture comprises a base network and six additional convolution layers that generate multi-scale feature maps [25]. Each layer can make 4 or 6 predictions in the same location independently from other layers by applying 3×3 convolution filters to each cell. For example, after feature extraction from VGG-16, the first feature map is 38×38, where each cell undergoes four predictions [46]. The complete SSD architecture is depicted in Fig. 2.

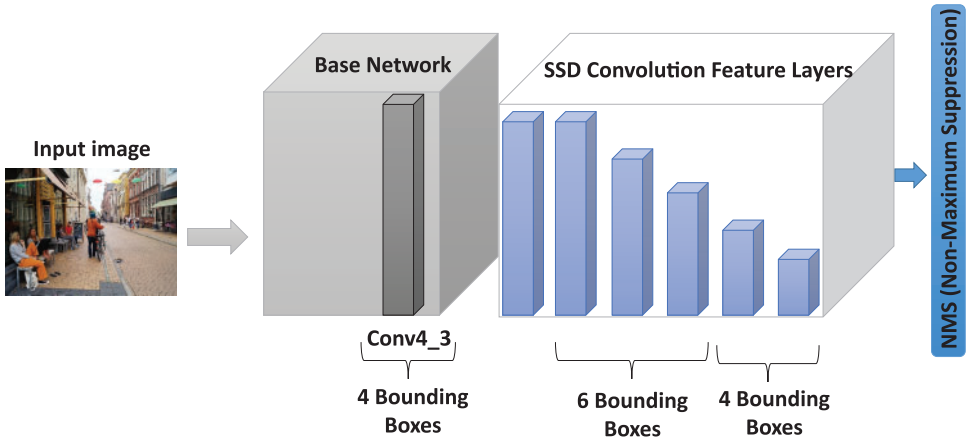

Figure 2: SSD main architecture

The convolutional feature layers progressively reduce the spatial dimensions of the feature maps, enabling the detection of objects of varying sizes. As spatial dimensions diminish, so does resolution, with lower-resolution layers being utilized for detecting larger objects. Additionally, each cell is assigned a list of initial boxes with varying aspect ratios to accommodate the detection of different shapes (illustrated in Fig. 3). The scaling of default boxes across feature map layers is detailed in the following formula:

$$s_k = \frac{(k-1) \times s_{max} + (m-k) \times s_{min}}{m-1}, \quad k \in [1, m] \tag{1}$$

where $s_{min} = 0.1$ or $s_{min} = 0.2$ is the smallest scale, $s_{max} = 0.9$ is the largest scale and m is the number of feature maps.

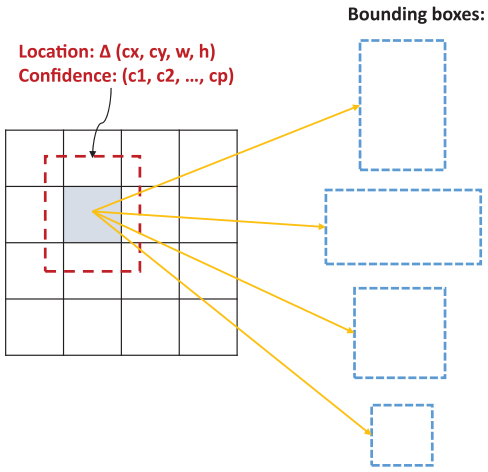

Figure 3: Default boxes aspect ratio variation

Additionally, the weights (w_k) and heights (h_k) of the default boxes are calculated in relation to the scale and different aspect ratio values (a_r) as follows:

$$w_k = s_k\sqrt{a_r} \quad and \quad h_k = s_k/\sqrt{a_r} \tag{2}$$

where $a_r \in \{0.5, 1/3, 1, 2, 3\}$.

Each default box yields to a boundary box prediction offsets of the object location and to $c = N+1$ confidence scores, where $c = N + 1$ is the expected class number plus an additional class for no object (background) predictions (Fig. 4). The model calculates the offsets of the initial boxes and picks the class with the largest confidence score.

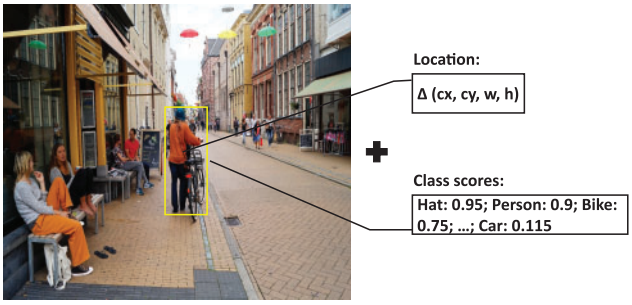

Figure 4: SSD default box output prediction

SSD employs a matching strategy to select default boxes and calculate localization loss. It identifies default boxes as positive matches with the ground truth if they achieve a Jaccard overlap threshold greater than 0.5. These positive matches serve as references for the predicted boxes and are used in the calculation of localization loss. In contrast, negative matches are disregarded. The localization loss is computed using a Smooth L1 loss, which is calculated between the coordinates of the predicted box (l) from positive matches and the ground truth bounding box (g), according to the formula:

$$L_{loc}(x, l, g) = \sum_{i \in Pos}^{N} \sum_{m \in \{c_X, c_y, w, h\}} x_{ij}^{p} \times smooth_{L1}(l_i^m - g_j^{\prime m}) \tag{3}$$

where N in the localization loss is the fitted boxes number, i is the default box index, j is the ground truth box index and

$$x_{ij}^{p} = \begin{cases} 1 & \text{if } IoU > 0.5 \text{ between } i \text{ and } j \text{ on class } p \\ 0 & \text{otherwise} \end{cases}$$

The ground truth bounding box offsets values on each feature map ($g_j^{'m}$) are calculated using the following formulas:

$$g_j^{'cx} = (g_j^{cx} - d_j^{cx})/d_i^{w} \quad \text{and} \quad g_j^{'cy} = (g_j^{cy} - d_j^{cy})/d_i^{h}$$

$$g_j^{'w} = \log\left(\frac{g_j^{w}}{d_i^{w}}\right) \quad \text{and} \quad g_j^{'h} = \log\left(\frac{g_j^{h}}{d_i^{h}}\right) \tag{4}$$

where d is the default box, and c_x and c_y are the left firearm corner coordinates of the bounding boxes.

The confidence loss is defined as the loss in making a class prediction and calculated as the softmax loss over multiple classes (c):

$$L_{conf}(x, c) = -\sum_{i \in Pos}^{N} x_{ij}^{p} \log(c_i^{'p}) - \sum_{i \in Neg} \log(c_i^{'0}) \tag{5}$$

where $c_i^{'p} = \dfrac{\exp(c_i^{p})}{\sum_p \exp(c_i^{p})}$.

The final loss function is a weighted sum of the localization loss and the confidence loss as follows:

$$L(x, c, l, g) = \frac{1}{N}(L_{conf}(x, c) + \alpha L_{loc}(x, l, g)) \tag{6}$$

where α is the weight.

To make final predictions, SSD employs Non-Maximum Suppression (NMS) to eliminate redundant predictions for the same classes. This is achieved by first arranging the confidence scores within the same class in descending order and then comparing the value of the current prediction against that of the previous one using an Intersection over Union (IoU) threshold of 0.45. Any prediction with an IoU greater than 0.45 relative to a preceding prediction is disregarded. Additionally, SSD facilitates the use of various data augmentation techniques to enhance both training and detection accuracy [25].

4 Base Classifiers Architecture

Base classifiers in object detection models are typically CNNs that extract features from input images before these features are classified by subsequent classification layers. Consequently, object detection models utilize these base classifiers as the backbone for their networks, which are then referred to as base networks [25]. The evolution of CNNs has spurred the development of various new base classifiers. In the proposed research, two variations of base classifiers have been employed for the SSD architecture. This section outlines the details of these two base architectures.

4.1 MobileNetV2

MobileNet was initially developed with the goal of reducing the high computational costs and complexity typically associated with CNNs, without compromising performance [46]. As a lightweight model, MobileNet is particularly well-suited for integration into resource-constrained devices, playing a crucial role in facilitating the deployment of CNNs in real-time applications.

A key innovation of MobileNet is the introduction of depth-wise separable convolution. This technique begins by applying a 3 × 3 convolution filter to each input channel independently with a depth of 1. Subsequently, it uses point-wise convolution to combine these outputs through a 1 × 1 kernel. This approach significantly cuts computational costs by reducing the number of parameters by a factor of nine compared to standard convolution [46].

MobileNetV2 builds on this by further decreasing the number of parameters [44]. It replaces point-wise convolutions with a bottleneck layer designed to minimize information loss during RELU activation while maintaining low end-to-end dimensionality and reducing parameter count. Inverted residual connections are used between bottleneck blocks to preserve gradients during transmission through the layers. Additionally, a 1 × 1 expansion layer is introduced prior to the depth-wise separable convolution to increase the number of channels based on an expansion factor, thereby enhancing the effectiveness of depth-wise separable convolutions [44,46,47] (see Fig. 5).

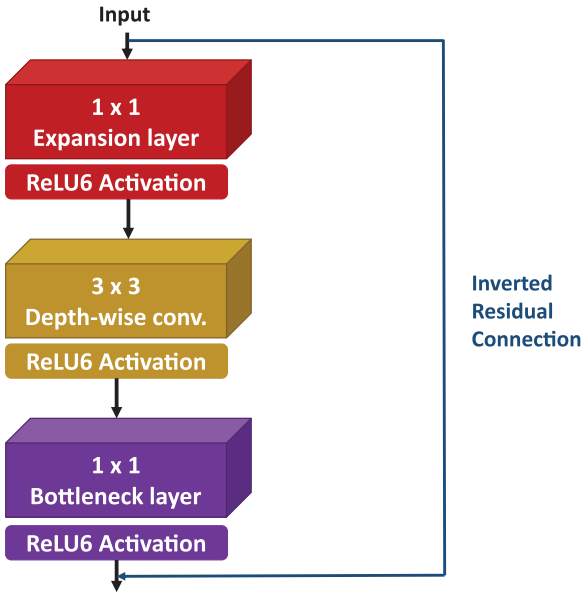

Figure 5: MobileNetV2 building block

4.2 InceptionV2

Inception, a complex CNN architecture, was developed to address three primary challenges. Firstly, selecting an appropriate kernel size is challenging due to the varying sizes of objects. Secondly, increasing the depth of the model often leads to difficulties in propagating gradients through its layers, which can cause overfitting. Additionally, constructing larger networks incurs substantial computational costs. To mitigate these issues, Inception adopts a wider rather than deeper architecture [46]. It employs three parallel convolution filters of sizes 1 × 1, 3 × 3 and 5 × 5 to manage computational efficiency and reduce input dimensions. A 1 × 1 convolution layer precedes the 3 × 3 and 5 × 5 filter to further decrease computational demands and condense input dimensions. The outputs from these filters are concatenated into a single vector, serving as the input for subsequent layers. Fig. 6 illustrates the original Inception architecture as described [47].

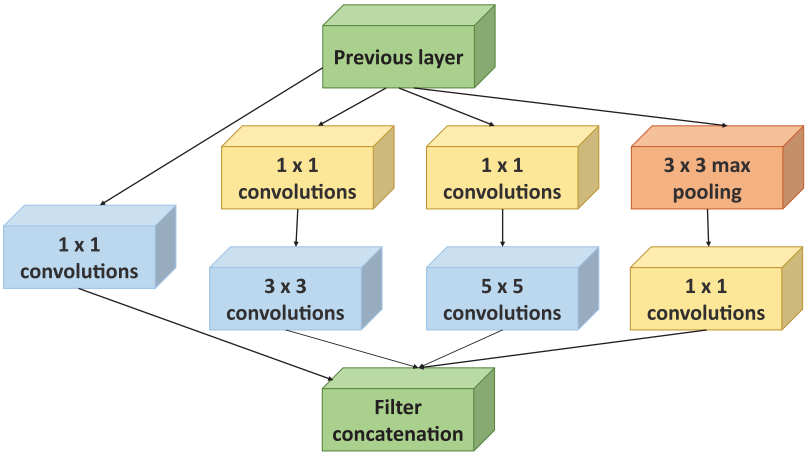

Figure 6: Inception model architecture

Large convolution sizes can drastically reduce input dimensions, leading to information loss and decreased accuracy, in addition to raising the network's complexity and computational demands. Inception V2 was introduced to enhance the accuracy of its predecessor while further diminishing computational costs. This enhancement was achieved through the use of scaled-up networks that incorporate efficient factorized convolutions. The 5×5 convolution layer of size $(2n-1) \times (2n-1)$ in the original model, which is 2.78 times more expensive than a 3×3 convolution, is replaced by two 3×3 convolution layers. Convolution filters of size $n \times n$ are factorized to a combination of $1 \times n$ and $n \times 1$ convolutions where two-layer convolutions are 33% cheaper in computational complexity compared to a single convolution. Finally, the filter banks convolutions are expanded as shown in Fig. 7 to go wider instead of deeper. This prevents excessive dimensions reduction and excessive information loss [45]. Essentially, Inception works on multi-level feature extraction [47].

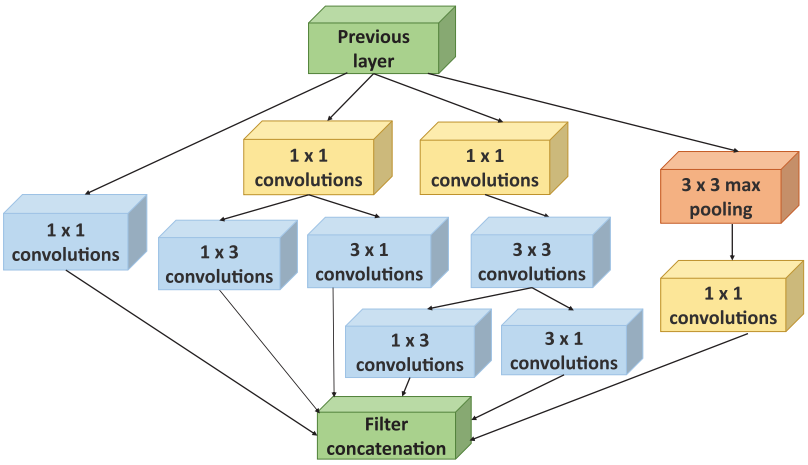

Figure 7: InceptionV2 architecture

5 Experiments and Results

Numerous research efforts in firearm detection have involved the collection and labeling of proprietary datasets. For the proposed study, a firearm dataset has been utilized initially created by the pioneering research that applied CNNs for firearm detection. The dataset was processed, the model parameters were set up and configured, and subsequently, two variations of the model were trained and tested. To assess the effectiveness of these models, two distinct tests were conducted. The results of these tests, along with a detailed analysis of the detection performance, are discussed in this section.

5.1 Data Collection and Preprocessing

The original handgun dataset compiled by the University of Granada was utilized [15], which focuses on the types of handguns most frequently used in crimes, including revolvers, automatic and semi-automatic pistols, six-gun shooters, horse pistols, and derringers. This dataset comprises 3000 images of firearms, each enriched with contextual details and sourced from various websites [15].

Training effective models require substantial data; however, video surveillance data are often limited and confidential. Importantly, using different datasets for experiments can preclude direct model comparisons, as the dataset significantly influences the learning process. To facilitate research consistency, the dataset collected and labeled by Olmos et al. [15] was employed, and formatted according to the PascalVOC standard [48].

For the training and testing of the proposed models, the dataset was divided into two parts: 2400 images randomly selected for training and 600 for testing. The annotations were transformed from Extensible Markup Language (XML) files into two Comma Separated Values (CSV) files, one for each set. During this process, a single box position mismatch was identified and corrected by removing the erroneous record from the CSV file. Subsequently, the two CSV files were converted into TensorFlow binary files (TFRecords). Fig. 8 illustrates the data preprocessing steps implemented.

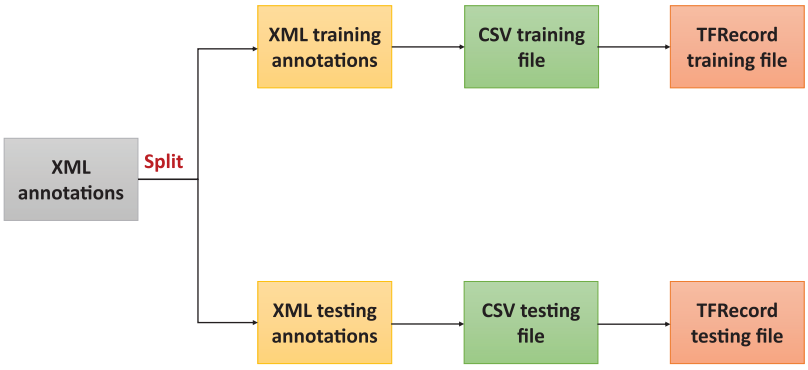

Figure 8: Data preprocessing steps

5.2 Experimental Setup

In this study, pre-trained TensorFlow models were utilized based on the COCO dataset, which were further fine-tuned. These models are accessible through the TensorFlow Model [49].

To facilitate weapon identification, a single class label "gun" was defined and categorized all other objects as background. The dataset comprises 3000 images, of which 2400 were allocated for training and 600 for testing. All images were resized to 300 × 300 pixels. Identical configurations were maintained for both model variations. The settings included a batch size of 16, an initial learning rate

of 0.003, and a momentum of 0.9, a weight decay of 0.001, over a total of 200,000 steps. The RMSprop optimizer was employed to enhance the training process. To mitigate overfitting, a dropout strategy was implemented with a probability of 0.8. Additionally, to augment the data, random horizontal flips were applied, random contrast adjustments, and SSD random crops. The proposed models were implemented using the GPU capabilities of Google Colab [50]. Table 1 provides a summary of the configuration parameters used.

Table 1: Simulation environment

Model parameter	Parameter value
Size of image	300 × 300
Size of batch	16
Initial learning rate	0.003
Momemtum	0.9
Weight	0.001
Step	200,000
Dropout probability	0.8
Detection per class	16
Used optimizer	RMSprop
Data augmentation methods	Random adjust contrast, SSD random crop, Random horizontal flip

5.3 Comparison of the SSD Base Network Variations

Two SSD base network variations were implemented using MobileNetV2 and InceptionV2 feature extractors for firearm detection. After 200,000 training steps, SSD with MobileNetV2 reached a total loss of 4.267 while SSD InceptionV2 based reached a total loss of 4.189. Fig. 9 shows the SSD decrease in total loss in correspondence to the number of steps and the evolution of the training model over time. The results reveal the following observations:

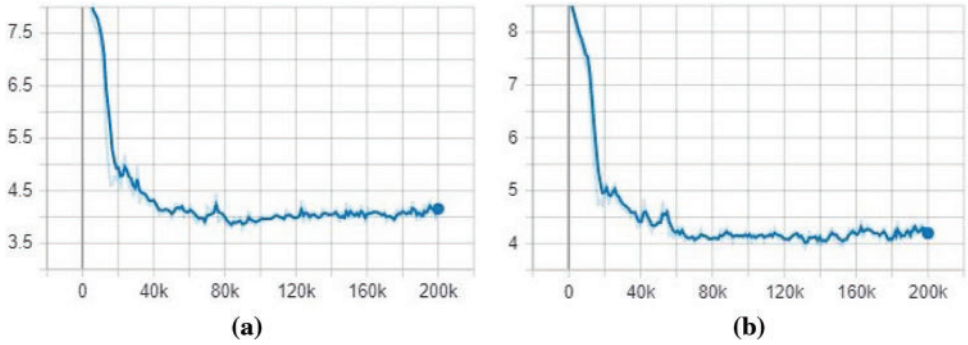

Figure 9: Total loss evolution during training: (a) SSD with MobileNetV2, (b) SSD with InceptionV2

While the loss decreased for both models, the Inception-based model exhibited a more substantial reduction in loss compared to the MobileNet-based model. At the final measurement step, the SSD

model with Inception reported a loss value of 1.552, whereas the SSD model with MobileNet recorded a loss value of 2.428, as detailed in Table 2.

Table 2: Comparison of the SSD model backbone variations

Backbone variation	Learning time	mAP		Total loss	Final step loss
		@IoU = 0.5	@IoU = 0.75		
MobileNetV2	24 h 22 min	0.800	0.615	4.267	2.428
InceptionV2	19 h 10 min	0.794	0.621	4.189	1.552

Additionally, the total loss curve, representing the weighted sum of classification and localization losses, exhibits periods of stagnation and decline at certain points. This phenomenon occurs because, although the localization loss consistently decreases, the classification loss encounters difficulties in showing similar reductions. A potential cause for this issue could be the suboptimal resolution of the dataset. Consequently, over time, the classification loss function in the Inception variant shows a greater reduction compared to that of the MobileNet variant.

To assess the performance of object detection models, the mean Average Precision (mAP) metric was employed [48], which quantifies the overlap between detected bounding boxes and ground truth using an Intersection over Union (IoU) threshold. Typically, an IoU of 0.5 or higher is required to classify detections as True Positives (TP). For evaluating the proposed SSD variants, the COCO mAP metrics were utilized [17].

Fig. 10 illustrates the increase in mean Average Precision (mAP) in relation to the number of training steps over time. The performance of two model variations was assessed using the 600-image testing set, applying IoU thresholds of 0.5 and 0.75, respectively. At an IoU of = 0.5, the SSD model equipped with MobileNetV2 achieved an 80% accuracy rate for firearm detection at the final measurement step, with a peak accuracy of 81.65%. This slightly exceeded the performance of the SSD model with InceptionV2, which reached an accuracy of 79.4% at the final step and a maximum accuracy of 80.91%. However, at the higher IoU threshold of = 0.75, the SSD with the InceptionV2 model outperformed the MobileNetV2 variant by achieving an accuracy of 62.1%, compared to 61.5% for MobileNetV2. Table 2 summarizes the key metrics from the training and testing phases.

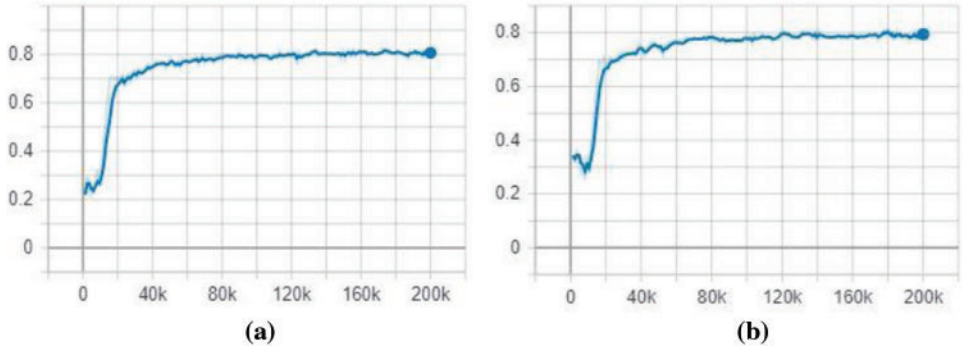

Figure 10: (Continued)

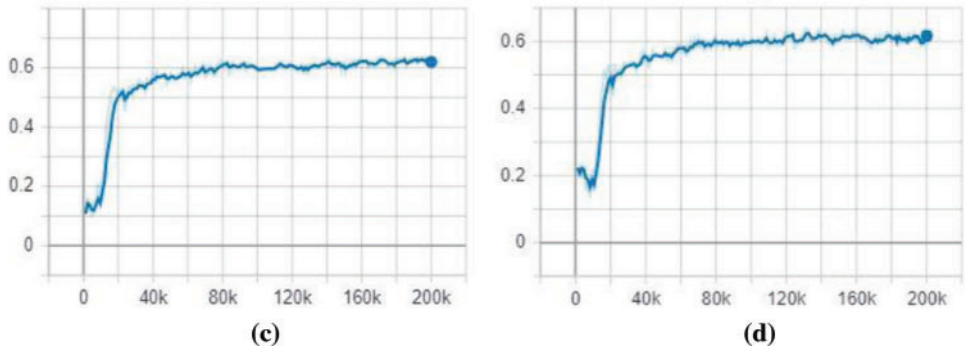

Figure 10: mAP evolution during training of base network SSD model: (a) MobileNetV2 mAP @ IoU = 0.5, (b) InceptionV2 mAP @ IoU = 0.5, (c) MobileNetV2 mAP @ IoU = 0.75 and (d) InceptionV2 mAP @ IoU = 0.75

Increasing the Intersection over the Union (IoU) threshold tends to reduce model accuracy, and the variations observed can be explained as follows. When using the COCO mAP metrics to evaluate the mean Average Precision (mAP) across different firearm sizes, the results generally indicate that the SSD model performs better at detecting larger firearms. Specifically, the SSD model utilizing the InceptionV2 architecture excels in detecting medium and large firearm sizes, whereas the SSD with MobileNetV2 architecture shows superior performance in detecting smaller firearm sizes, as illustrated in Fig. 11. As the threshold is raised, smaller firearms are often dismissed due to their confidence scores falling below the heightened threshold, resulting in an accuracy decline for MobileNetV2 and allowing InceptionV2 to surpass it.

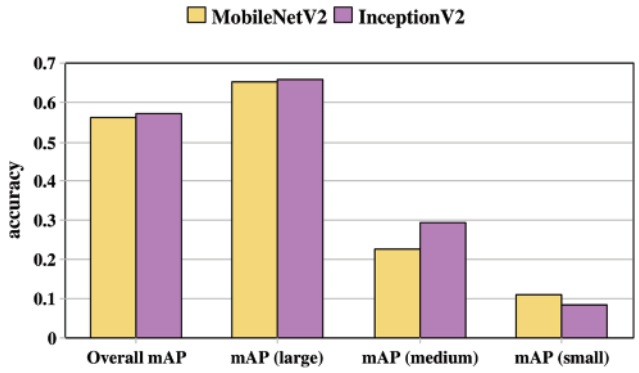

Figure 11: Accuracy grouped by gun size and base network

Overall, the SSD model based on the InceptionV2 architecture outperforms the MobileNetV2-based model in terms of firearm detection accuracy. As shown in Fig. 11, the InceptionV2-based model achieved a 57.2% overall accuracy, while the MobileNetV2 achieved a 56.3% overall size-related accuracy. These results are consistent with previous findings reported in [27] regarding general object detection. In summary, the more complex feature extraction capabilities of InceptionV2 allow it to capture a broader range of features, thereby improving accuracy. Conversely, the lighter architecture of MobileNetV2 facilitates faster detection times, albeit at a slight compromise in accuracy. Table 3 summarizes the accuracy values obtained for the two model variations at the final training step, detailing performance relative to different firearm sizes.

Table 3: Comparison of the accuracy at final step of the model variations training on different firearms sizes

AccuracyBackbone	MobileNetV2	InceptionV2
Overall mAP	0.563	0.572
mAP (large)	0.654	0.661
mAP (medium)	0.229	0.294
mAP (small)	0.110	0.087

Fig. 12 shows some results from testing the SSD-based MobileNetV2 on firearms images, while Fig. 13 shows some images from the SSD-based InceptionV2 testing results.

Figure 12: Detection results in SSD-based MobileNetV2

Figure 13: Detection results in SSD-based InceptionV2

5.4 Comparison with Previous Firearms Detection Approaches

In this section, the performance of the SSD model was analyzed, focusing on the balance between speed and accuracy in firearm detection. Table 4 presents the accuracy values achieved by various base classifiers at an Intersection over Union (IoU) of = 0.5. Although VGGNet is commonly used in firearm detection, the results in this paper demonstrate that MobileNetV2 and InceptionV2 surpass both VGGNet and Overfeat, achieving accuracies of 81.65% and 80.91%, respectively.

Table 4: Comparison of the firearms detection base classifiers

Base classifier	IoU threshold	Testing accuracy	Testing execution time (ms per image)
MobileNetV2	0.5	0.81	7
InceptionV2	0.5	0.80	16
VGGNet [38]	0.5	0.46	36
Overfeat-2 [38]	0.5	0.64	29
Overfeat-3 [38]	0.3	0.89	22

Lowering the threshold value generally improves model accuracy, as observed with Overfeat-3. Among the classifiers, MobileNetV2 stands out as the fastest, making it highly suitable for real-time firearm detection due to its rapid feature extraction capabilities and ease of deployment in surveillance systems. Conversely, Overfeat-3, with its slower detection speed [38], is less appropriate for real-time applications. For benchmarking, all base classifiers were trained and tested using the same dataset of 3000 firearm images [38], which was preprocessed and augmented with additional images from another dataset. This standardization makes them comparable. In previous studies [38], 2535 images were used for training and 218 for testing. In contrast, proposed models were trained on 2400 images and evaluated on 600 images. Additionally, Table 4 details the execution time per image during the testing phase, measured in milliseconds for each model. The results highlight several key findings: 1) Proposed trained models (MobileNetV2 and InceptionV2) in the MARIE system demonstrate superior performance compared to the models proposed in [38] (VGGNet and its improved variants). 2) MobileNetV2 outperforms the InceptionV2 model in reducing image detection time, enhancing its efficiency for real-time applications.

To maintain consistency with existing research, the handgun dataset was utilized, originally created by Olmos et al. [15], which has been subsequently employed by other studies addressing firearm detection. Unlike previous studies that trained SSD models on all 3000 images, the dataset was randomly divided into training and testing subsets. Furthermore, to facilitate comparison with prior research, an additional test was conducted using the same test dataset that previous works used to assess model accuracy. The test set comprised 608 images, evenly split between 304 handgun and 304 non-handgun images. During preprocessing, errors in the dataset were corrected while maintaining the original composition and the image size of 160 $\times$ 120 pixels. Specifically, 11 images (5 handgun were replaced and 6 non-handgun images) with similar class images sourced from the internet and another dataset containing a pistol class.

To evaluate and benchmark proposed models against previous approaches, three metrics were employed: Precision, Recall, and the F1 measure. Precision calculates the percentage of handguns correctly detected, Recall assesses the retrieval rate from the 304 handgun images, and the F1 measure provides a balance between Precision and Recall. These metrics were computed using the following formulas:

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$F1\ measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{9}$$

where TP = True Positives, FP = False Positives, FN = False Negatives and TN = True Negatives.

For real-time firearms detection, precise localization of the firearm is less critical than ensuring its detection. Therefore, reducing FN was prioritizing and increasing TP to minimize missed detections and maximize the identification of present firearms. This approach aims to enhance the model's recall and the inference speed in firearm detection. Table 5 presents a comparison of various previous firearm detection models, all implemented using the same training and testing datasets. While all models utilized the entire test set, the MobileNet-CNN in [19] was tested on a subset of 420 images. The results indicate that SSD outperforms the previous models, achieving 100% recall with the fastest detection speed. Although Faster R-CNN also reaches 100% recall with higher precision, it, along with Fast R-CNN both being two-stage detectors exhibits slower real-time detection speeds compared to SSD. Overall, as a one-stage detector, SSD demonstrates comparable performance in firearm detection and excels in optimizing the speed/accuracy trade-off.

Table 5: Comparison of the SSD model variation with previous works

Model	TP	FN	TN	FP	Precision	Recall	F1 measure
SSD with MobileNetV2	304	0	199	102	74.87%	100%	85.63%
SSD with InceptionV2	304	0	206	98	75.62%	100%	86.11%
Faster R-CNN [23]	304	0	247	57	84.21%	100%	91.43%
Fast R-CNN [22]	232	72	248	56	80.76%	76.31%	78.37%
MobileNet-CNN [19]	156	54	168	42	78.78%	74.28%	76.46%

Further, authors [43] suggested that additional work is needed to train their proposed models, including MobileNet-CNN, on more powerful machines. However, the F1 measure reveals that SSD with MobileNetV2 and InceptionV2 outperformed Fast R-CNN and MobileNet-CNN, achieving 85.63% and 86.11%, respectively. In terms of the speed/accuracy trade-off, SSD proves effective as a one-stage detector for real-time surveillance systems.

The variations in the base network using MobileNetV2 and InceptionV2 showed comparable performance. While SSD based on InceptionV2 slightly surpasses SSD based on MobileNetV2 in accuracy, the latter excels in terms of detection speed. Therefore, the selection of a specific SSD base network depends on the operational context. For environments with limited hardware resources aiming to optimize detection speed, SSD with MobileNetV2 is preferred. Conversely, in settings where powerful equipment is available and greater detection accuracy is prioritized, InceptionV2 would be the preferred choice for the SSD base network.

6 Conclusion

Despite significant global development, firearm incidents continue to rise, resulting in substantial human casualties. This escalating trend necessitates multifaceted interventions to mitigate and prevent homicides. This paper, introduced MARIE, a one-stage object detection mechanism designed for the real-time identification of firearms. MARIE incorporates SSD with two variations in the base network MobileNetV2 and InceptionV2 to optimize the speed/accuracy trade-off essential for prompt law enforcement responses. Transfer learning was applied to a handgun dataset previously collected and labeled, and results were benchmarked against existing techniques using the same dataset.

Both architectures demonstrated high efficiency, surpassing VGG-16, which is commonly used in firearm detection. Remarkably, SSD achieved 100% recall with zero false negatives, a significant count of true negatives, and commendable precision while maintaining rapid detection speeds, outperforming Faster R-CNN and other advanced two-stage detectors. The chosen training dataset contributed to this success by providing images with distinct, clear features, enhancing training effectiveness. Conversely, the use of video surveillance data can detract from learning outcomes due to cluttered and noisy scenes.

Looking ahead, the aim is to enhance the proposed model's precision for small objects and reduce false positives by integrating a Feature Pyramid Network (FPN). Also, there is a plan to assess the speed/accuracy trade-off using newer SSD base networks, such as MobileNetV3 and InceptionV3. Furthermore, there is the intention to train and evaluate the proposed model variations across diverse benchmark firearm datasets to explore how different datasets influence detection performance.

Acknowledgement: None.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: Diana Serhal Abou Nader: Writing—original draft preparation, Methodology, Software, Experimental work. Hassan Harb: Writing—original draft preparation, Related works, Methodology, Data curation. Ali Jaber and Ali Mansour: Writing—original draft preparation, Conceptualization of this study, Methodology, Data curation. Christophe Osswald: Related works, Methodology, Software. Nour Mostafa and Chamseddine Zaki: Conceptualization of this study, Related works. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: We agree to make data and materials supporting the results or analyses presented in the paper available upon reasonable request.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Naghavi M, Marczak LB, Kutz M, Shackelford KA, Arora M, Miller-Petrie M, et al. Global mortality from firearms, 1990–2016. JAMA. 2018;320:792–814. doi:10.1001/jama.2018.10060.
2. Abdallah R, Harb H, Taher Y, Benbernou S, Haque R. ICAD: an intelligent framework for real-time criminal analytics and detection. In: International Conference on Web Information Systems Engineering, 2023 Oct 21; Singapore: Springer Nature Singapore; p. 300–15.

3. Abdallah R, Harb H, Taher Y, Benbernou S, Haque R. CRIMEO: criminal behavioral patterns mining and extraction from video contents. In: 2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA), 2023; Thessaloniki, Greece: IEEE; p. 1–8.
4. UNODC. Global study on homicide 2019: Executive summary; 2019. Available from: https://www.unodc.org/documents/data-and-analysis/gsh/Booklet1.pdf. [Accessed 2024].
5. Karp A. Estimating global civilian-held firearms numbers. Ginebra, Suiza: Small Arms Survey; 2018.
6. Curtin SC, Garnett MF. Suicide and homicide death rates among youth and young adults aged 10–24: United States, 2001–2021. NCHS Data Brief. 2023 Jun;(471):1–8.
7. Irvin-Erickson Y, Bai B, Gurvis A, Mohr E. The effect of gun violence on local economies: gun violence, business, and employment trends in Minneapolis, Oakland, and Washington, DC. In: Research report. USA: Urban Institute; 2016. Available from: https://books.google.com.kw/books?id=8E1-0AEACAAJ. [Accessed 2024].
8. Manley NR, Fischer PE, Sharpe JP, Stranch EW, Fabian TC, Croce MA, et al. Separating truth from alternative facts: 37 years of guns, murder, and violence across the US. J Am Coll Surg. 2020;230(4):475–481. doi:10.1016/j.jamcollsurg.2019.12.040.
9. Latha RS, Sreekanth GR, Suganthe RC, Selvaraj RE. A survey on the applications of Deep Neural Networks. In: 2021 International Conference on Computer Communication and Informatics (ICCCI), 2021; Coimbatore, India; p. 1–3. doi:10.1109/ICCCI50826.2021.9457016.
10. Gulli A, Kapoor A, Pal S. Deep learning with TensorFlow 2 and Keras: regression, ConvNets, GANs, RNNs, NLP, and more with TensorFlow 2 and the Keras API. Birmingham, UK: Packt Publishing, Limited; 2019.
11. Voulodimos A, Doulamis N, Doulamis A, Protopapadakis E. Deep learning for computer vision: a brief review. Comput Intell Neurosci. 2018;2018(1):1–13. doi:10.1155/2018/7068349.
12. Wu X, Sahoo D, Hoi SC. Recent advances in deep learning for object detection. Neurocomputing. 2020;396:39–64. doi:10.1016/j.neucom.2020.01.085.
13. Gesick R, Saritac C, Hung CC. Automatic image analysis process for the detection of concealed weapons. In: Proceedings of the 5th Annual Workshop on Cyber Security and Information Intelligence Research: Cyber Security and Information Intelligence Challenges and Strategies, 2009; Oak Ridge, TN, USA; p. 1–4.
14. Xiao Z, Lu X, Yan J, Wu L, Ren L. Automatic detection of concealed pistols using passive millimeter wave imaging. In: IEEE International Conference on Imaging Systems and Techniques (IST), 2015; Macau, China: IEEE; p. 1–4.
15. Olmos R, Tabik S, Herrera F. Automatic handgun detection alarm in videos using deep learning. Neurocomputing. 2018;275(9):66–72. doi:10.1016/j.neucom.2017.05.012.
16. Alaqil RM, Alsuhaibani JA, Alhumaidi BA, Alnasser RA, Alotaibi RD, Benhidour H. Automatic gun detection from images using faster R-CNN. In: 2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH), 2020; Riyadh, Saudi Arabia; p. 149–54.
17. Lin TY, Maire M, Belongie S, Hays J, Perona P, Ramanan D, et al. Microsoft COCO: common objects in context. In: European Conference on Computer Vision, 2014; Zurich, Switzerland: Springer; p. 740–55.
18. Aggarwal P, Thapliyal S, Singh CR, Kukreja V, Mehta S. Advanced computational approaches to gun detection with CNN-SVM model. In: 2024 5th International Conference for Emerging Technology (INCET), 2024; Belgaum, India: IEEE; p. 1–6.
19. Elmir Y, Laouar SA, Hamdaoui L. Deep learning for automatic detection of handguns in video sequences. In: 3rd Edition of the National Study Day on Research on Computer Sciences (JERI), 2019; Saida, Algeria.
20. Zou Z, Shi Z, Guo Y, Ye J. Object detection in 20 years: a survey. arXiv preprint arXiv:190505055. 2019.
21. Girshick R, Donahue J, Darrell T, Malik J. Rich feature hierarchies for accurate object detection and semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2014; Columbus, OH, USA; p. 580–87.

22. Girshick R. Fast R-CNN. In: IEEE International Conference on Computer Vision (ICCV), 2015; Las Condes, Chile; p. 1440–8.
23. Ren S, He K, Girshick R, Sun J. Towards real-time object detection with region proposal networks. In: NeurIPS Proceedings, 2015; Montreal, QC, Canada; p. 91–9.
24. Redmon J, Divvala S, Girshick R, Farhadi A. You only look once: unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016; Las Vegas, NV, USA; p. 779–88.
25. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: single shot multibox detector. In: European Conference on Computer Vision, 2016; Amsterdam, The Netherlands: Springer; p. 21–37.
26. Iqbal Z, Liu F, Khan M, Shen X, Yuan Y, Ullah I, et al. End-to-end deep learning model for firearms identification in video. IEEE Access. 2021;9:87007–19.
27. Cherpanath ED, Fathima Nasreen PR, Pradeep K, Menon M, Jayanthi VS. Food image recognition and calorie prediction using faster R-CNN and mask R-CNN. In: 2023 9th International Conference on Smart Computing and Communications (ICSCC), 2023; Kochi, Kerala, India; p. 83–9.
28. Sermanet P, Eigen D, Zhang X, Mathieu M, Fergus R, LeCun Y. Overfeat: integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:13126229. 2013.
29. Warsi A, Abdullah M, Husen MN, Yahya M. Automatic handgun and knife detection algorithms: A review. In: 14th International Conference on Ubiquitous Information Management and Communication (IMCOM), 2020; Taichung, Taiwan; p. 1–9.
30. Redmon J, Farhadi A. YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017; Honolulu, HI, USA; p. 7263–71.
31. Redmon J, Farhadi A. YOLOv3: an incremental improvement. arXiv preprint arXiv:180402767. 2018.
32. Flitton G, Breckon TP, Megherbi N. A comparison of 3D interest point descriptors with application to airport baggage object detection in complex CT imagery. Pattern Recognit. 2013;46(9):2420–36. doi:10.1016/j.patcog.2013.02.008.
33. Hosam O, Alraddadi A. K-means clustering and support vector machines approach for detecting fire weapons in cluttered scenes. Life Sci J. 2014;11(9).
34. Halima NB, Hosam O. Bag of words based surveillance system using support vector machines. Int J Secur Appl. 2016;10(4):331–46. doi:10.14257/ijsia.2016.10.4.30.
35. Tiwari RK, Verma GK. A computer vision based framework for visual gun detection using SURF. In: International Conference on Electrical, Electronics, Signals, Communication and Optimization (EESCO), 2015; Visakhapatnam, India; p. 1–5.
36. Tiwari RK, Verma GK. A computer vision based framework for visual gun detection using Harris interest point detector. Procedia Comput Sci. 2015;54:703–12. doi:10.1016/j.procs.2015.06.083.
37. Verma GK, Dhillon A. A handheld gun detection using faster R-CNN deep learning. In: Proceedings of the 7th International Conference on Computer and Communication Technology, 2017; Chengdu, China; p. 84–8.
38. Lai J. Developing a real-time gun detection classifier; 2017. Available from: https://api.semanticscholar.org/CorpusID:54033357. [Accessed 2024].
39. Olmos R, Tabik S, Lamas A, Pérez-Hernández F, Herrera F. A binocular image fusion approach for minimizing false positives in handgun detection with deep learning. Inf Fusion. 2019;49(2):271–80. doi:10.1016/j.inffus.2018.11.015.
40. Ma J, Yakimenko OA. The concept of sUAS/DL-based system for detecting and classifying abandoned small firearms. Def Technol. 2023;30(10):23–31. doi:10.1016/j.dt.2023.04.017.
41. Mehta P, Kumar A, Bhattacharjee S. Fire and gun violence based anomaly detection system using deep neural networks. In: International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020; Coimbatore, India; p. 199–204.

42. Romero D, Salamea C. Convolutional models for the detection of firearms in surveillance videos. Appl Sci. 2019;9(15):2965. doi:10.3390/app9152965.
43. Basit A, Munir MA, Ali M, Mahmood A. Localizing firearm carriers by identifying human object pairs. arXiv preprint arXiv:200509329. 2020.
44. Sandler M, Howard A, Zhu M, Zhmoginov A, Chen LC. MobileNetV2: inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018; Salt Lake City, UT, USA; p. 4510–20.
45. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016; Las Vegas, NV, USA; p. 2818–26.
46. Liu Y, Mehta S. Hands-on deep learning architectures with python. Birmingham, UK: Packt Publishing; 2019.
47. Dhillon A, Verma GK. Convolutional neural network: a review of models, methodologies and applications to object detection. Prog Artif Intell. 2020;9(2):85–112. doi:10.1007/s13748-019-00203-0.
48. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes (VOC) challenge. Int J Comput Vis. 2010;88(2):303–38. doi:10.1007/s11263-009-0275-4.
49. Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, et al. Tensorflow object detection API; 2017. Code: https://github.com/tensorflow/models.git. Documentation: https://modelzoo.co/model/objectdetection. [Accessed 2024].
50. Research G. Google Colab. Available from: https://colab.research.google.com/notebooks/intro.ipynb [Accessed 2024].