ArmVision: An Approach to Improve Police Officers Response Times to Gun Violence and Put a Stop to Armed Crimes Before They Occur

Vinay Venkatesh
Lynbrook High School
San Jose, United States of America
vinayven01@gmail.com

***Abstract*—An estimated 19,223 people lost their lives due to gun violence in 2020. ArmVision prevents arm related crimes before they occur and brings quicker attention to them if they do with its efficient notification system. Its wide range of applicability expresses a bright future for its development. ArmVision uses a complex machine learning (ML) algorithm called You Only Look Once (YOLO) and Convolutional Neural Networks (CNNs) to optimize its performance.**

Keywords—Artificial Intelligence, Machine Learning, Object Detection, School Safety

I. Introduction

Since 2013, an estimated 618 school shootings have taken place. From 1999 to 2017, U.S. gun deaths have reached nearly 40,000, exhibited in Fig. 1.[8] Due to the lack of productive means of stopping gun crimes, they continue to rise daily. Innocent people from elementary schools, universities, convenience stores, and more are killed, injured, or traumatized. The resulting trauma on the surviving community can be lifelong and devastating. In any environment, safety is crucial.

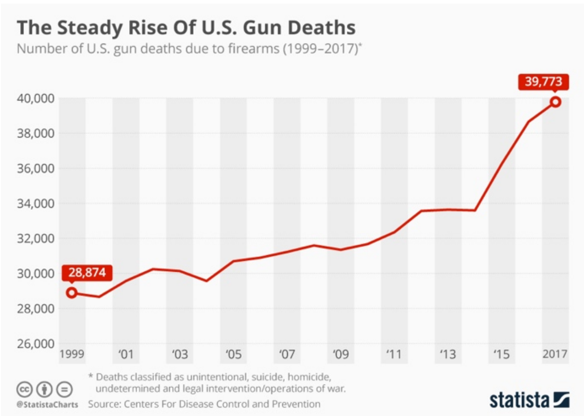

Fig. 1. Number of U.S. deaths due to firearms (1999-2017)

ArmVision, a firearm detection software powered by real-time object detection algorithms, is an innovative solution to provide security. Its capability to lower firearm-related crime rates and bring immediate assistance to crime scenes will save lives and improve citizen safety at a national level.

II. Design Methodology

ArmVision's design methodology consists of three major parts: data importation, data pre-processing, and YOLO algorithm. The dataset used in this project contains several different types of guns including, but not limited to, pistols, rifles, and shotguns. There were two sections to the dataset: label and image sets. Labels corresponding to an image signal where the desired object to be classified is located in the image. After the sets were uploaded into the integrated development environment (IDE), samples of images were examined for errors to ensure high performance. Fig. 2, below, is an example of an image used to train the dataset.

Fig. 2. Image of a gun used to train the object detection model

A. Data Importation

The initial data set of 3000 images from Roboflow were separated into groups of images and corresponding pre-annotated labels. Due to the lack of a test set, using Python's scikit-learn library, a test set was manually created by splitting the existing images and labels. Images were then displayed to ensure quality, which allowed them to be directly used in training the model.

B. Data Pre-processing

All images were processed already in a 416 x 416 pixel size, allowing image processing steps to be avoided.

C. YOLO Algorithm

YOLO is a real-time object detection algorithm that is beneficial because of its drastic advancements in speed and accuracy compared to Fast/Faster R-CNN. The YOLO

979-8-3503-4616-9/23/$31.00 ©2023 IEEE

algorithm is based on regression, meaning classes are predicted and bounding boxes are placed in a single run. Every bounding box has four descriptors: center, width, height, and object class, as shown in Fig. 3.

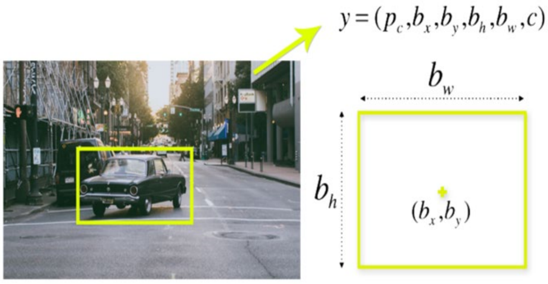

Fig. 3. Representation of bounding box descriptors

The YOLO algorithm splits images into cells, and each cell is responsible for predicting five bounding boxes. A representation of an image split into cells is shown in Fig. 4.[3]

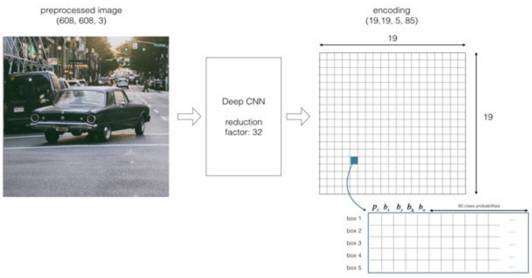

Fig. 4. Representation of bounding box descriptors

Using a concept called non-max suppression, bounding boxes with low accuracy are removed from the split cells by performing the Intersection over Union (IoU), shown in Fig. 5.[3]

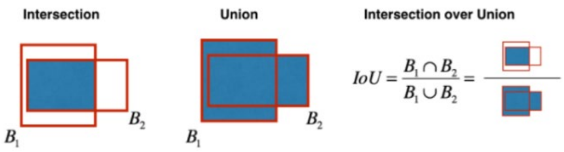

Fig. 5. Diagram of IoU operation

Score is the final probability that the object is present, calculated by the equation below:

$$score_{c,i} = p_c \times c_i \quad (1)$$

After the non-max suppression leaves the bounding box with the highest probability score, the algorithm returns a vector containing the details of the bounding box. The architecture of the YOLO algorithm is shown below in Fig. 6.[3]

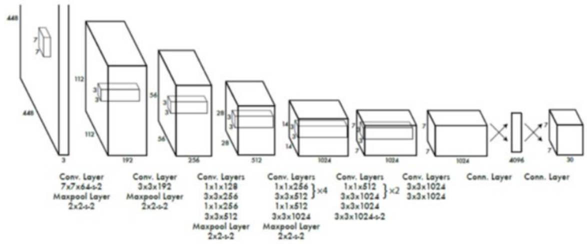

Fig. 6. YOLO algorithm architecture

III. Results and Analysis

The model trained over 50 epochs, or training iterations, and in 19 minutes, a set of optimized weights were trained. Over the course of the epochs, the weights accuracy steadily increased as shown in Fig. 7.

TABLE I. Training Accuracies

Metric	Peak Accuracy
mAP_0.5	0.93
mAP_0.5:0.95	0.665
precision	0.7768
recall	0.9234

Fig. 7. Highest training accuracy

Using the trained weights alongside Python's OpenCV library, guns can be detected in real-time video and in images as proven in Fig. 8.

Fig. 8. Gun detected with 91% certainty

Using TensorBoard's data log shown in Fig. 9, the change in accuracy over each epoch can be visualized.

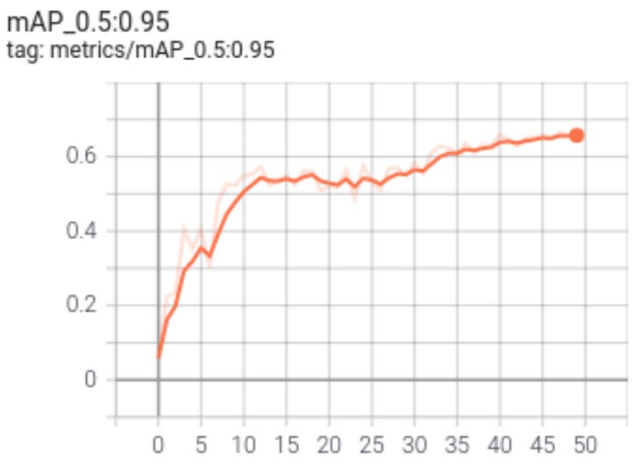

Fig. 9. Visual representation showing steady increase of mAP_0.5:0.95 accuracy

IV. Challenges

During the span of the project, understanding how to process thousands of images in a short period of time while retaining quality of the weights was an issue. By increasing the timeout duration and stretching the ram usage limits, a compromise on quality was not necessary and the model was able to train exponentially faster.

In addition to challenges while developing ArmVision, there were several roadblocks after its creation. Applying ArmVision to the real world will surely be of benefit, but how can the help it serves be maximized? Can ArmVision be scaled to the extent of national benefit? ArmVision cannot detect concealed firearms, and firearms are usually hidden while carried. This drastically reduces its effectiveness.

Another issue is surveillance quality and lighting during the event. In Fig. 10, an inaccurate result is shown due to issues with lighting and camera quality.

Fig. 10. Example of an innacurate result due to poor camera quality and lighting

How would ArmVision function on every surveillance camera in the world? Because of the vast number of unique security cameras and software they use, evolving ArmVision's compatibility with different types of security cameras and growing its renown is a challenge sure to be faced.

Inaccuracies like shown in Fig. 10 are an obstacle for the future but can be fixed with improvements in size and quality of the dataset ArmVision is trained on. Optimal size and quality of data that the model is trained on will play a large role in the accuracy of detecting firearms.

V. Strengths of Model

After viewing Fig. 8, it is evident that the detection of a gun is clear and accurate. In public places, this model will perform at a high accuracy with hundreds of gun varieties due to the training dataset being composed of a large variety of guns. With further training and larger datasets, higher accuracy and farther detection distances can be achieved.

VI. Applications

Though ArmVision's main intended use is in schools, its areas of use include public places, areas with high gun violence rates and more. Once ArmVision detects a gun in these areas, the police will be notified immediately.

ArmVision can be implemented in security cameras using real-time surveillance to identify firearms in the live video. Once a firearm is detected with a certain accuracy level, local police officers and stations would be alerted via an ArmVision notification app.

The ArmVision notification app will incorporate sign ups, location tracking, and live viewing. The sign-up feature will be used to avoid confusion within police officers regarding which officer is addressing what crime notification. Location tracking will use a map to show the exact location of the crime. Much like a standard map application, travel times and ideal routes to arrive will be provided. For all gun identification alerts, live viewing will be available to avoid false positives. With a click of one button, officers will be able to see the surveillance feed that ArmVision provides. Live viewing is especially useful when firearm probability levels are low.

VII. Conclusion

As visible in Fig. 11, many schools across America have started to use security cameras to monitor their campus. From 2009 to 2020, security camera usage in public schools has risen about 30%.[7] The sharp growth is a testament for the necessity of school surveillance.

How effective is surveillance in schools? Although this is a step forward in improving school safety, there is no guarantee that someone reliable is constantly monitoring the surveillance for threats. Using ArmVision, it is possible to not only constantly monitor the surveillance, but also detect one of the largest school threats, guns, with high accuracy.

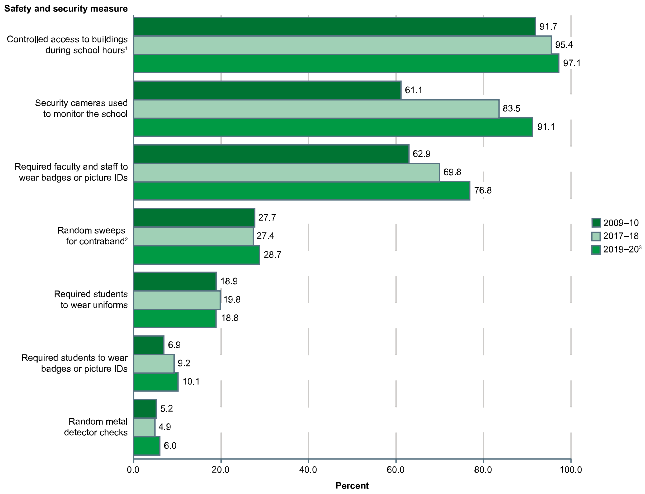

Fig. 11. Percentage of public schools that used selected safety and security measures: School years 2009-10, 2017-18, 2019-20

Several sets of post training data indicate that the hyper parameters, training data size, and kinds of training images play a large impact on epoch speeds and model accuracy. The batch size is an especially important hyper parameter because it controls how many images are propagated through the network each epoch. After the hyper parameters and all other training factors were optimized, the greatest training speeds and accuracies were achieved, drastically increasing ArmVision's efficiency.

References

[1] Brownlee, Jason. "How to Control the Stability of Training Neural Networks With the Batch Size." Machine Learning Mastery, 27 Aug. 2020, machinelearningmastery.com/how-to-control-the-speed-and-stability-of-training-neural-networks-with-gradient-descent-batch-size/.

[2] Gunfire on School Grounds in the United States, 11 Feb. 2019, maps.everytownresearch.org/gunfire-in-school/#ns.

[3] Gupta, Manish. "YOLO-You Only Look Once." Medium, Towards Data Science, 30 May 2020, towardsdatascience.com/yolo-you-only-look-once-3dbdbb608ec4.

[4] Redmon, Joseph, et al. Computer Vision Foundation, 2016, pp. 1–10, You Only Look Once: Unified, Real-Time Object Detection.

[5] "Statistics." Giffords, 19 Dec. 2020, giffords.org/lawcenter/gun-violence-statistics/.

[6] Świeżewski, Jędrzej. "YOLO Algorithm and YOLO Object Detection: An Introduction - Appsilon: End to End Data Science Solutions." Appsilon, 20 Oct. 2020, appsilon.com/object-detection-yolo-algorithm/.

[7] "The NCES Fast Facts Tool Provides Quick Answers to Many Education Questions (National Center for Education Statistics)." National Center for Education Statistics (NCES) Home Page, a Part of the U.S. Department of Education, https://nces.ed.gov/fastfacts/display.asp?id=334.

[8] McCarthy, Niall, and Felix Richter. "Infographic: The Steady Rise of U.s. Gun Deaths." Statista Infographics, 13 Dec. 2018, https://www.statista.com/chart/16421/the-number-of-us-gun-deaths -due-to-firearms/.

[9] H. Jain, A. Vikram, Mohana, A. Kashyap and A. Jain, "Weapon Detection using Artificial Intelligence and Deep Learning for Security Applications," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 193-198, doi: 10.1109/ICESC48915.2020.9155832.

[10] M. Kmiec, A. Glowacz and A. Dziech, Towards Robust Visual Knife Detection in Images: Active Appearance Models Initialised with Shape-specific Interest Points, In Proceedings of 5th International Conference on Multimedia Communications, Services and Security pp. 148–158, (2012).

[11] Tiwari RK, Verma GK (2015) A computer vision based framework for visual gun detection using Harris interest point detector. Procedia Computer Science 54:703–712