BORDER SURVEILLANCE SYSTEM USING DEEP LEARNING

Aditya Nahata, Nanhay Singh

Netaji Subhas University Of Technology, Dwarka, New Delhi-110078

Abstract: The old notion of physical borders between nations has become obsolete with the onset of globalization and the greater mobility of global inhabitants. Physical boundaries that depend on military forces and powerful weaponry need a large amount of labor, are prone to human mistake, and could be harmful to the environment, especially in rocky areas. This project seeks to introduce a "Smart Border Surveillance System" as a novel way to overcome the constraints of physical borders. This system replaces the traditional strategy of armed patrolling with cutting-edge surveillance technology and includes an integrated Intruder Alert System. By doing this, border security concerns are addressed without the need for armed patrols and physical barriers, which frequently result in the loss of human life and put a burden on technology. The given system makes use of sensors and surveillance cameras which can detect any kind of threats like drones and weapons through machine learning algorithms and alerts the officials in case of such detections.

1. Introduction

The focus of this chapter is to provide a brief introduction to the background of our thesis, the motivation, scope and the organization of our thesis. We will explore the significance of Border surveillance and how cutting-edge algorithms like Deep learning can play a major role in the management of it.

Border surveillance is a vital task for ensuring national security and sovereignty. However, it is also a challenging task due to the large and complex nature of the border areas, which may span different terrains, climates, and lighting conditions. Moreover, the border areas may be subject to various types of threats and disturbances, such as illegal crossings, smuggling, terrorism, wildlife intrusion, etc. Therefore, it is essential to have an effective and efficient system that can monitor the border areas and alert the authorities in case of any abnormal or suspicious events.

Object detection, a subfield of computer vision that seeks to locate and identify various types of objects in an image or video, is one of the promising technologies that can improve border surveillance capabilities. In order to investigate the situation and take the necessary action, object detection might yield useful information on the existence, location, size, shape, and category of the objects in the scene. Object detection, for instance, can assist in spotting cars, weapons, contraband, or intruders in border areas and notifying law enforcement or border guards of their presence.

However, object detection is also a difficult problem due to the high variability and complexity of the objects and the scenes. For instance, the objects may have different appearances, poses, orientations, scales, occlusions, or deformations. The scenes may have different backgrounds, clutter, illumination, or weather conditions. Moreover, the object detection system should be able to handle large amounts of data and process them in real time or near real time. Therefore, there is a need for developing advanced and robust methods and models for object detection that can overcome these challenges and achieve high performance.

In this project, We suggest utilizing object detection methods based on deep learning techniques to create and operate a border monitoring system. In order to extract complicated features and patterns from massive volumes of data, a subsection of machine learning which is called deep learning that uses a multi-layer combination of networks called artificial neural networks(ANN). Many computer vision fields, including image classification, face recognition, semantic segmentation, and others, have witnessed impressive outcomes using deep learning. Using various architectures and methodologies, deep learning has learned powerful features and representations for object recognition and classification, enabling state-of-the-art performance in object detection.

We plan to use one of the existing deep learning models for object detection, such as Faster R-CNN or YOLO as the backbone of our system. We will also customize and fine-tune the model according to our specific requirements and scenarios. We will evaluate the performance of our system on various datasets and metrics. We will also compare our system with other existing methods and systems for border surveillance and object detection.

2. Literature Review:

The artificial intelligence subfield of machine learning has grown significantly in the last several years. It combines a number of methods, including as supervised and unsupervised learning, deep learning, and reinforcement learning, with the goal of creating algorithms that can perform better thanks to data-driven learning procedures.

The many facets of machine learning will be covered in this overview of the literature, along with its historical evolution, real-world uses, and present research directions. We will also look at other machine learning algorithms, including supervised and unsupervised learning, reinforcement learning, and deep learning.

Within the field of artificial intelligence, machine learning involves instructing computers on how to find patterns and correlations in vast amounts of data. Applications for machine learning get better with time and grow more precise as they handle more data. Numerous industries, including healthcare, shopping carts, entertainment, and homes, use machine learning.

To put it briefly, machine learning is a programming approach that allows programmers to educate and train any machine to perform, hence enabling automation. This is how a model for a robot is trained to be intelligent and capable of self-driving. Fig. 5 illustrates the operation of machine learning.

Why Deep Learning ?

Neural networks are trained on big datasets as part of a process known as deep learning in machine learning. It has shown promise in a variety of applications, such as speech recognition, natural language processing, and picture and video analysis.

Deep learning is frequently chosen for object recognition jobs because it does not require human feature engineering because it can automatically extract features from the data.. This implies that deep learning models can frequently outperform humans on tasks involving object detection.

Because deep learning models can learn hierarchical representations of pictures, convolutional neural networks (CNNs) in particular, are especially well-suited for image classification and object detection tasks.

Overall, Deep learning has produced cutting-edge outcomes on a variety of object detection tasks and is often the method of choice for these tasks. However, other machine learning techniques, such as Haar cascades and histogram of oriented gradients (HOG), may still be appropriate in certain situations, such as when computational resources are limited.

Related deep learning algorithms for object detection task:

1. Haar cascades:

Haar Cascades are an effective object detection method that Paul Viola and Michael Jones introduced in their 2001 publication "Rapid Object Detection using a Boosted Cascade of Simple Features". The majority of their uses are in real-time applications because of their high efficiency. They may be trained to identify many different kinds of objects, but face detection is the most common application for them.

2. The Histogram of Oriented Gradients (HOG)

It is a feature descriptor that is utilized in image processing and computer vision to depict an object's geometry. It has shown effectiveness in object detection applications, especially pedestrian recognition, and is based on the way that an image's edge directions or intensity gradients are distributed.

The feature descriptor algorithm known as Histogram of Oriented Gradients, or HOG, is frequently used in computer vision and image processing to recognise objects. Navneet Dalal and Bill Triggs presented it in their CVPR article "Histograms of Oriented Gradients for Human Detection" from 2005.

The fundamental tenet of HOG is that edge directions or intensity gradient distributions can be used to characterize the appearance and form of local objects within an image.

3. Scale-Invariant Feature Transform (SIFT):

This feature descriptor is employed in order to identify and characterize local characteristics in pictures. It has demonstrated performance in numerous object recognition tasks and is independent of both image rotation and scale.

An approach in computer vision called Scale-Invariant Feature Transform, or SIFT, is used to identify and characterize local features in images. David Lowe created the technique in 1999, and it has since been widely applied in a variety of fields, including gesture recognition, video tracking, 3D modeling, object detection, and panoramic image stitching.

4. Speeded Up Robust Feature (SURF):

This is a feature descriptor and detector that is faster than SIFT and has been successful in object recognition tasks.
Speeded Up Robust Features (SURF) is a robust image detector and descriptor. It was introduced by Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool in 2006 as a faster and more efficient alternative to SIFT (Scale-Invariant Feature Transform).
SURF shares many characteristics with SIFT but considers speed and performance in its design.

5. Convolutional neural networks:

CNNs are a special kind of neural network that work particularly well for tasks involving object detection and picture classification. CNNs have achieved state-of-the-art performance on several object detection benchmarks and are able to learn hierarchical representations of images.

Overview of C.N.N. for Object Detection:

One kind of neural network that works especially well for tasks involving object detection and picture classification is the convolutional neural network (CNN). They are able to identify things at various sizes and locations within the image by learning hierarchical representations of the images.

There are several approaches to object detection using CNNs, including:

Single Shot Detectors (SSDs): These are fast object detection systems that use a single CNN to predict object classes and locations in an image. SSDs are able to process images in real-time and have been successful in a variety of object detection tasks.

Region-Based CNNs (R-CNNs): These are object detection systems that use a CNN to classify object proposals generated by a separate region proposal network. R-CNNs have produced cutting-edge outcomes on certain object detecting benchmarks, but are slower than SSDs due to the need to process each region proposal separately.

Faster R-CNNs: These are an extension of R-CNNs that use a "region proposal network" to generate object proposals and a CNN to classify and locate objects within those proposals. Faster R-CNNs are faster than R-CNNs due to the use of a shared convolutional feature extractor for both the region proposal and detection networks.

6. YOLOv8:

This is the most recent iteration of the well-known object identification and image segmentation model, YOLO (You Only Look Once) algorithm. It was built by Ultralytics, the same company that produced the well-known YOLOv5 model that defined an industry.

A number of changes are introduced by YOLOv8, including modules for context aggregation, feature fusion, and spatial attention. YOLOv8 is now one of the most important object detection algorithms in the industry because of these enhancements, which lead to faster and more accurate object detection.

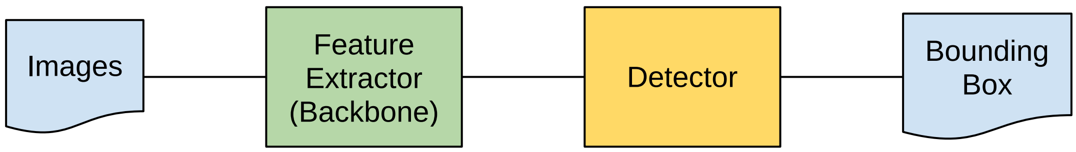

Figure 1. YOLO network architecture

3. Methodology:

Steps followed in Training the Model :

1. Assemble and annotate a collection of pictures that have the objects you wish to detect. One tool that can be used for this is LabelImage.
2. To get the dataset ready for training, preprocess it. This can involve scaling the images, standardizing the data, and splitting the data into training and validation sets and normalizing the pixel values.
3. Select a Yolov8 object detection model that has been pre-trained, then train it to perform detection tasks for specific classes.
4. Test the model with both saved photos and a webcam or Raspberry Pi camera in real time.
5. Analyze the outcomes for each class and document the results of the tests.

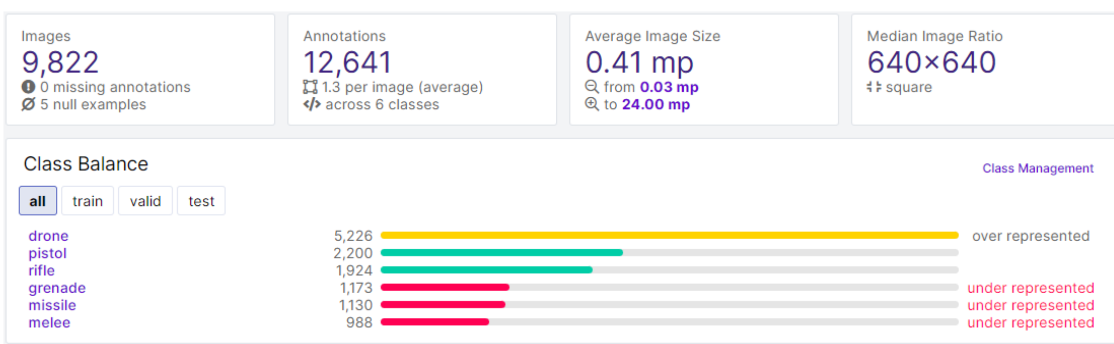

Figure 2. Prepared dataset

The following block diagram that depicts the system architecture and the system overview that follows serve to explain the methodology and operation of the system.

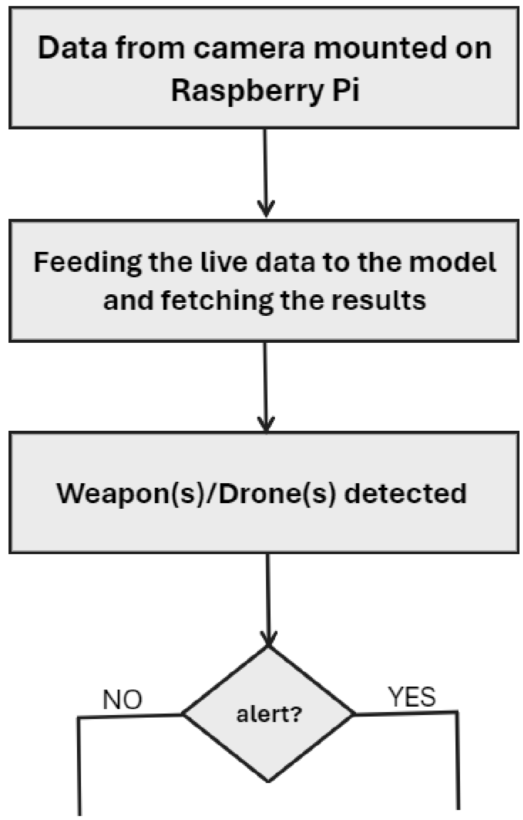

Figure 3. Flowchart of work

Three essential metrics are used to evaluate a model's performance in data science and machine learning: precision, recall, and loss. These metrics are especially useful in classification and detection tasks.

Precision: The relevancy of a model is gauged by its precision. It is computed by dividing the total number of false positives (FP) and true positives (TP) by the sum of the two. Stated differently, precision provides an answer to the following query: "Of all the instances the model predicted as positive, how many are actually positive?"

Precision = TP / (TP + FP)

A low percentage of false-positive predictions is shown by high precision. But accuracy by itself doesn't provide a whole picture. Even while a model with great precision can miss a lot of real positive cases, it might not be very useful if it just predicts a small number of positive predictions.

Recall (Sensitivity): Recall quantifies the comprehensiveness of a model. It is computed by taking the total number of false negatives (FN) and dividing it by the number of true positives. Stated differently, recall provides an answer to the following query: "How many of the actual positive instances did the model correctly identify?"

Recall = TP / (TP + FN)

A low percentage of false-negative predictions is indicated by a high recall. Like precision, recollection on its own, though, can be deceptive. A model with perfect recall (but horrible precision) will be one that just predicts positive for every occasion.

Loss (Cost Function): Another name for the cost function is the loss function, which is a way to gauge a machine learning model's effectiveness. It calculates the discrepancy between the actual and expected output. The goal during training is to minimize this difference, or "loss". The kind of machine learning problem determines the precise form of the loss function. (regression, classification, etc.).

For instance, Mean Squared Error (MSE), which computes the average of the squares of the variations between the expected and actual values, is a popular loss function in regression applications. Binary Cross-Entropy, which determines the negative average of the log of the predicted probabilities for the actual classes, is a popular loss function in binary classification problems.

Various models could necessitate distinct loss functions, and the selection of a loss function can greatly influence the model's performance.

Although recall and accuracy are mainly focused on classification issues, the loss function is a broader idea that may be used for nearly any kind of machine learning work. In order to obtain a thorough grasp of a model's performance, it is imperative to take into account each of these indicators collectively.

3. Results:

The model was able to identify and class the images properly for the validation and correctly made the bounding boxes as well as the probability of the given class. The response time of the model is also low and has almost negligible latency.

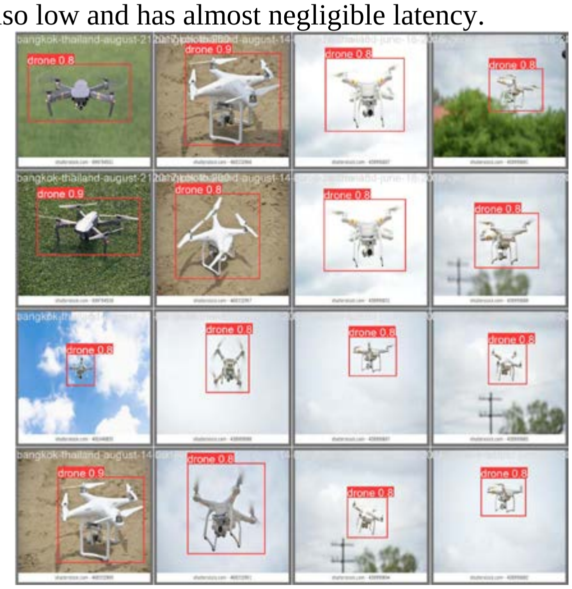

Figure 4. Predictions on validation set

In comparison to other models like faster R-CNN with high parameters, the yolo model was able to perform at par and also provide significantly better results. This can be very helpful in case of live predictions.

Table 1. Model comparison

Model Name	**Speed (ms)**	**COCO mAP**	**Outputs**
SSD MobileNet v2 320x320	19	20.2	Boxes
Faster R-CNN ResNet152 V1 640x640	64	32.4	Boxes
Faster R-CNN Inception ResNet V2 640x640	206	37.7	Boxes
Faster R-CNN Inception ResNet V2 1024x1024	236	38.7	Boxes
Yolov8 m 2.9 M params 640x640	221	33.2	Boxes

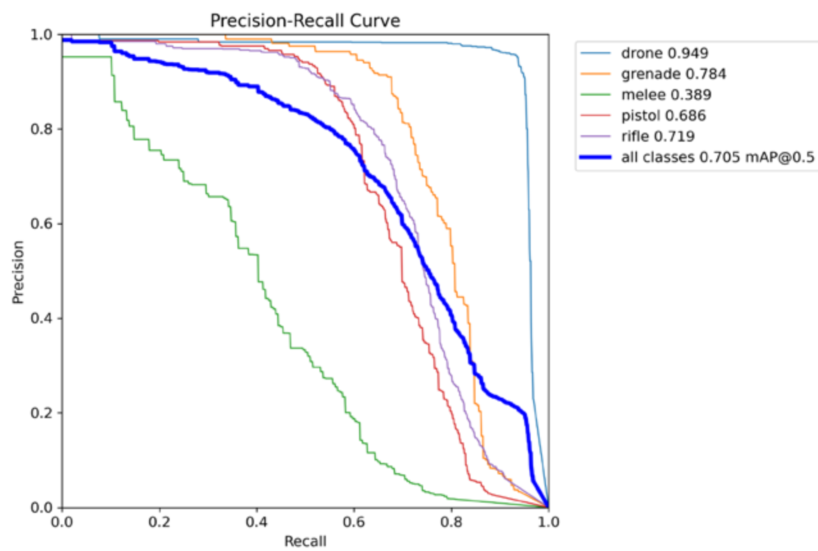

Figure 5. Precision-recall curve

As we can see from the above graph, the accuracy of the model for the prediction of drones is **94.9%**, which means that our performed very well given that this is performed for real-life scenarios where the response time should be low and also the model has very small fraction of time to actually capture the right object and make the class prediction.

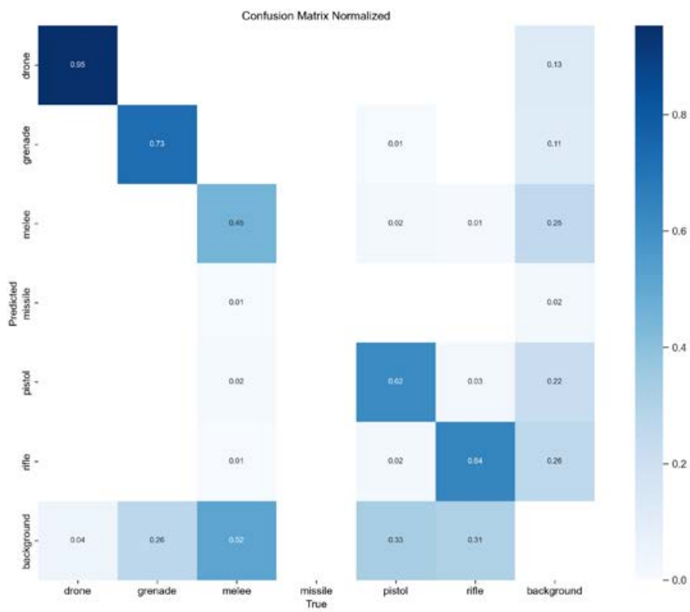

Figure 6. Confusion matrix on validation set

4. Future Scope:

The lives of army officials and troops are safeguarded by the smart border surveillance system, which provides a subtle vigilance without the need for strongly fortified and militarized borders, particularly in places with rough terrain and bad weather. The system is a prime illustration of how technology may be used to secure borders between nations, particularly in the modern, globalized world, without negating the need for human intelligence and judgment. A safe access and memory-efficient system solution that is workable and reasonably priced has been described.

We enumerated the primary obstacles that our soldiers face when protecting the border. Automated border control systems that process travelers at border crossings with accuracy and efficiency can be developed with machine learning. For example, a traveler's identity and authorization to enter the country can be verified using facial recognition technology.

The model we have created is highly trainable, i.e., it can be trained to recognize and detect a variety of objects and people, thanks to the fast and reliable YOLOv*8 algorithm. With a few modifications, it can be used to identify more firearms, vehicles, and important objects in the future. Our goal is to inspire and spark interest in security-related subjects with this post.

Future work

Future work that can enhance the system and make the system more practical includes:

1. Deploy the given model on an IOT device like Raspberry Pi and use the model for live detections using a camera connected to the device.

2. Developing an ALERT System by creating an API that can send requests to the system and email the officials when a threat is detected.
3. Building a Client-GUI to feed and monitor video, as well to upload threat detected frames as evidence.

5. Acknowledgement:

I would like to express my deepest gratitude to my supervisor, Dr. NANHAY SINGH , for his valuable guidance and support throughout the course of my thesis. His expertise and insights were instrumental in helping me to develop my research, and I am deeply grateful for their patience and encouragement.

I am also grateful to my fellow students and colleagues, who provided valuable feedback and support during the research process.

Finally, I would like to express my gratitude to my family and friends for their love and support throughout my studies. Without their unwavering encouragement and support, I would not have been able to complete this work.

Thank you all for helping me to achieve this important milestone in my academic journey.

6. References:

[1] Goyal, A., Anandamurthy, S.B., Dash, P., Acharya, S., Bathla, D., Hicks, D., Bhan, A. and Ranjan, P., 2019. Automatic border surveillance using machine learning in remote video surveillance systems. In Emerging Trends in Electrical, Communications and Information Technologies (pp. 751-760). Springer, Singapore.

[2] Nguyen, T.T. and Tran, T.H., 2022. A real-time border surveillance system using deep learning and edge computing. In 2022 RIVF International Conference on Computing and Communication Technologies (RIVF) (pp. 1-6). IEEE.

[3] Sharma, R., 2021. Machine learning-based smart surveillance and intrusion detection system for border security. In Advances in Intelligent Systems and Computing (pp. 197-207). Springer Singapore

[4] Dillon Reis, Jordan Kupec, Jacqueline Hong, Ahmad Daoudi. (2023). Real-Time Flying Object Detection with YOLOv8. arXiv preprint arXiv:2305.09972.

[5] Bellazi, K.M., Marino, R., Lanza-Gutierrez, J.M., & Riesgo, T. (2020). "Towards a Machine Learning-Based Edge Computing Oriented Monitoring System for the Desert Border Surveillance Use Case." IEEE Access, 8, pp. 218304-218322. DOI: 10.1109/ACCESS.2020.3042699.

[6] Laouira, M.L., Abdelli, A., Ben Othman, J., & Kim, H. (2021). "An Efficient WSN Based Solution for Border Surveillance." IEEE Transactions on Sustainable Computing, 6(1), pp. 54-65. DOI: 10.1109/TSUSC.2019.2904855.

[7] Purohit, M., Singh, M., Yadav, S., Singh, A.K., Kumar, A., & Kaushik, B.K. (2022). "Multi-Sensor Surveillance System Based on Integrated Video Analytics." IEEE Sensors Journal, 22(11), pp. 10207-10222. DOI: 10.1109/JSEN.2021.3131579.