AUTO TRACKING AND TARGET LOCKING OF STATIC OBJECT USING MACHINE VISION

Project/Thesis ID. 2022: 1

Session: BS. Fall 2019

Project Supervisor: Dr. Adnan Saeed

Submitted By

Abdul Moaiz

Hamdan Bin Tanveer

Abdur Rehman

Muhammad Raheem

Electrical Engineering Department

National University of Technology (NUTECH)

Certification

It is certified that the project titled "Auto Tracking and Target Locking of Static Object using Machine Vision" carried out by Abdul Moaiz (F19603002), Hamdan Bin Tanveer (F19603015), Abdur Rehman (F19603020), and Muhammad Raheem (F19603032) Under the supervision of Dr. Adnan Saeed, National University of Technology (NUTECH), Islamabad, is fully adequate in scope and in quality, as a capstone project for the degree of BS of Electrical Engineering.

External Examiner

[Name of Examiner]

[Designation]

Project Supervisor

Dr. Adnan Saeed

Assistant Professor

Chairman

Department of Electrical Engineering, National University of Technology (NUTECH)

Sustainable Development Goals

(Please tick the relevant SDG(s) linked with FYDP)

SDG No	Description of SDG	SDG No	Description of SDG
SDG 1	No Poverty	**SDG 9**	**Industry, Innovation, and Infrastructure**
SDG 2	Zero Hunger	SDG 10	Reduced Inequalities
SDG 3	**Good Health and Well Being**	SDG 11	Sustainable Cities and Communities
SDG 4	Quality Education	SDG 12	Responsible Consumption and Production
SDG 5	Gender Equality	SDG 13	Climate Change
SDG 6	Clean Water and Sanitation	SDG 14	Life Below Water
SDG 7	Affordable and Clean Energy	SDG 15	Life on Land
SDG 8	**Decent Work and Economic Growth**	SDG 16	Peace, Justice and Strong Institutions
		SDG 17	Partnerships for the Goals

Range of Complex Problem Solving				
	Attribute	**Complex Problem**		
	Range of conflicting requirements	Involve wide-ranging or conflicting technical, engineering and other issues.		
	Depth of analysis required	Have no obvious solution and require abstract thinking, originality in analysis to formulate suitable models.		
	Depth of knowledge required	Requires research-based knowledge much of which is at, or informed by, the forefront of the professional discipline and which allows a fundamentals-based, first principles analytical approach.		
	Familiarity of issues	Involve infrequently encountered issues		
	Extent of applicable codes	Are outside problems encompassed by standards and codes of practice for professional engineering.		
	Extent of stakeholder involvement and level of conflicting requirements	Involve diverse groups of stakeholders with widely varying needs.		
	Consequences	Have significant consequences in a range of contexts.		
	Interdependence	Are high level problems including many component parts or sub-problems		
Range of Complex Problem Activities				
	Attribute	**Complex Activities**		
	Range of resources	Involve the use of diverse resources (and for this purpose, resources include people, money, equipment, materials, information and technologies).		
	Level of interaction	Require resolution of significant problems arising from interactions between wide ranging and conflicting technical, engineering or other issues.		
	Innovation	Involve creative use of engineering principles and research-based knowledge in novel ways.		
	Consequences to society and the environment	Have significant consequences in a range of contexts, characterized by difficulty of prediction and mitigation.		
	Familiarity	Can extend beyond previous experiences by applying principles-based approaches.		

Abstract

Auto tracking and target locking is an automated weapon system that eliminates the need for human intervention in defense operations. Traditionally, such systems have relied on radar and Opto-electrical technology, which are susceptible to jamming and come with high procurement costs and technical challenges. This proposal suggests an alternative approach using visual perception through cameras. The system would utilize a camera to capture visual information in the form of image data. The operator would select a target reference in the image, and this image data would then be processed to estimate and track the target's position within the image. Based on the estimated target position, the system would generate motion commands for the gun platform or laser light, aligning it with the target and locking onto it. At this point, the operator can make a decision regarding whether to engage the target or disregard it. It's important to note that the performance of the system would not be affected by variations in gun prototypes. By leveraging visual perception and image processing, this proposal aims to address the vulnerabilities, difficulties, and high costs associated with existing auto tracking and target locking systems. The use of cameras provides an alternative means of tracking targets, reducing the system's susceptibility to jamming and potentially lowering procurement expenses. But here we would be using A Laser Beam instead of gun mechanism. Camera module works on the principle of image processing using CNN, Roboflow, Epochs, PyTorch, YOLOv5 libraries. Image processing is the method to perform some operation on image in order to get an enhanced image or to extract some useful information from it. It is the type of signal processing in which input is the image and the output is characteristic or feature of the image. Image processing includes:

- Importing the image via image acquisition tools
- Analyzing and manipulating the image.
- Output in which result can be an altered image or analysis based on image.

Image processing is done by python in camera module then the information of image send to microcontroller. Then by coding after face recognition the microcontroller move the laser and target the static object.

Undertaking

I certify that the project Auto Tracking and Target Locking of Static Object using Machine Vision is our own work. The work has not, in whole or in part, been presented elsewhere for assessment. Where material has been used from other sources it has been properly acknowledged/ referred.

[Abdul Moaiz]

[Hamdan Bin Tanveer]

[Muhammad Raheem]

[Mehran Ali]

Acknowledgement

All the recognition and appreciation are for all-powerful ALLAH who made the universe and presented the humanity with information and intelligence to look for its privileged insights. I feel extraordinary joy and distinction to communicate my most profound feeling of appreciation, earnest sentiments and respects to my Supervisor Dr. Adnan Saeed for his proficient direction, huge assistance and unique method of guidance for the fulfillment of my investigations. I additionally prefer to stretch out my unique on account of Dr. Abdullah Waqas, Dr. Muhammad Abu Bakr, Dr. Khalid Iqbal, Dr. Waqar ud Din, Dr. Shahid Iqbal, Lec. Abdul Basit Taj, Lec. Syed Shahzad, and our Honorable HOD Dr. Nauman Razzaq for their consolation and backing. I have acquired a great deal from NUTECH, Islamabad during most recent Four years. I wish to express my heartiest thanks to all my colleagues for their support in the studies and helping me out when in need.

DEDICATED

TO

NATIONAL UNIVERSITY OF TECHNOLOGY (NUTECH),

ISLAMABAD, PAKISTAN

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF GRAPHS

LIST OF ABBREVIATIONS

ABSTRACT

Auto tracking and target locking is an automated weapon system that eliminates the need for human intervention in defense operations. Traditionally, such systems have relied on radar and Opto-electrical technology, which are susceptible to jamming and come with high procurement costs and technical challenges. This proposal suggests an alternative approach using visual perception through cameras. The system would utilize a camera to capture visual information in the form of image data. The operator would select a target reference in the image, and this image data would then be processed to estimate and track the target's position within the image. Based on the estimated target position, the system would generate motion commands for the gun platform or laser light, aligning it with the target and locking onto it. At this point, the operator can make a decision regarding whether to engage the target or disregard it. It's important to note that the performance of the system would not be affected by variations in gun prototypes. By leveraging visual perception and image processing, this proposal aims to address the vulnerabilities, difficulties, and high costs associated with existing auto tracking and target locking systems. The use of cameras provides an alternative means of tracking targets, reducing the system's susceptibility to jamming and potentially lowering procurement expenses. But here we would be using A Laser Beam instead of gun mechanism. Camera module works on the principle of image processing using CNN, Roboflow, Epochs, PyTorch, YOLOv5 libraries. Image processing is the method to perform some operation on image in order to get an enhanced image or to extract some useful information from it. It is the type of signal processing in which input is the image and the output is characteristic or feature of the image. Image processing includes:

- Importing the image via image acquisition tools
- Analyzing and manipulating the image.
- Output in which result can be an altered image or analysis based on image.

Image processing is done by python in camera module then the information of image send to microcontroller. Then by coding after face recognition the microcontroller move the laser and target the static object.

Chapter 1

INTRODUCTION

1.1 Background

Auto tracking and target locking on static objects using machine vision have emerged as critical components in various fields, including surveillance systems, robotics, and autonomous vehicles. This background section provides an overview of the fundamental concepts, techniques, and applications of auto tracking and target locking on static objects based on machine vision. Machine vision is a branch of artificial intelligence that enables machines to perceive and interpret visual information, similar to human vision. It involves the use of cameras, image processing algorithms, and pattern recognition techniques to analyze and understand visual data. Machine vision systems play a crucial role in object recognition, tracking, and targeting applications. Auto tracking refers to the automated process of following and tracking a target object without requiring manual intervention. In the context of static objects, auto tracking involves detecting and maintaining the position and orientation of an object within a given field of view. Auto tracking algorithms continuously analyze visual input to track the target object's movement and maintain its position relative to the camera. Target locking is the process of acquiring and maintaining focus on a specific target object. It ensures that the system maintains accurate tracking of the target, even in the presence of occlusions, background clutter, or changes in lighting conditions. Target locking algorithms aim to keep the tracked object consistently centered in the field of view, facilitating further analysis or control. Object detection algorithms are used to identify and locate static objects within an image or video stream. These algorithms employ various techniques, including Haar cascades, Convolutional neural networks (CNNs), and region-based approaches like YOLO. Once an object is detected, tracking algorithms are employed to follow its movements over time. These algorithms leverage methods such as optical flow, correlation filters.

1.2 Problem Statement

With machines replacing humans, can a target locking system be developed that can be deployed in a hostile environment which is swift, reliable, robust and accurate?

1.3 Aims and Research Objectives

Aims and Objective is to develop a hardware-based prototype able to:

- To track the static objects using computer vision
- To aim/point laser once the target is locked
- To used for surveillance mission and target acquisition.
- Accuracy of gun system up to approximately 85%.

1.4 Scope of Research

This research is aimed at developing a sustainability assessment tool dedicated to geotechnical projects. The scope of this research is:

1.5 Project Specification

Machine vision
Machine vision is a process to remove picture data to get significant data of an item. The data will be additionally handled as per wanted goals. In the field of industry, machine vision can be utilized as supplant of conventional sensors. For the case of machine vision application is counting the quantity of jugs and arranging the containers utilizing machine vision instead of various sensors. By utilizing machine vision that replaces numerous sensors, it would make process more productive and practical.

Target Tracking
The objective tracking is a process to gauge and track the place of an objective in the succession of pictures. Optical flow is a strategy that ordinarily use for the objective following reason. Optical flow is an assessment of the development from a piece of the picture in view of the subsidiary of the light force in the grouping of pictures. In two aspects space, this mean how far a picture pixel exchanging between two progressive pictures outlines, they are available casing and past edge. Computation of the subordinate depends on changes in light forces between present casing and past edge. Changes in light force that happens in the piece of picture can be brought about by the development made by the item.

Human Detection
In the process of target selection for static defense application, to know what things are at in front of the camera will greatly help the operator to interact with it. Human detection can be used as one of feature to provide help for the operator. Human detection is a method for detecting presence or absence of human in the image sequence.

1.6 Outcomes

- A complete hardware capable of detecting a target using Camera.
- Gun locks the target by aiming at it and emits the laser beam on the target.
- The target’s color affect to the tracking process in image frame. If the color blends with the background, the Static Defense System is unsuccessful to track and lock the target, otherwise it is successful.

1.7 Challenges and Future Directions:

Despite significant progress, auto tracking and target locking on static objects based on machine vision still face challenges, such as occlusion handling, illumination variations, and object appearance changes. Future research efforts are focused on improving robustness, efficiency, and adaptability to complex environments, as well as integrating machine learning and deep learning techniques for enhanced performance.

Chapter 2

EXISTING TECHNOLOGIES

In this section, the current technique for target following and locking is talked about. In this day and age, innovation assumed a huge part in fighting. Military innovation is frequently explored and created by researchers and specialists explicitly for use fighting by the military. Numerous new innovations came because of the tactical subsidizing of science.

Knowing the presence of the foe is a significant calculates any war zone. Target following aides in identifying the adversary presence and target locking assists with keeping up with lock on the foe targets.

2.1 RADAR

RADAR stands for Radio Detection and Ranging. Radar is an article location framework that utilizations radio waves to decide the reach, point, or speed of items. It might be used to distinguish plane, ships, space contraption, coordinated rockets, motor vehicles, environment courses of action, and domain. A radar structure contains a transmitter making electromagnetic waves in the radio or microwaves space, a sending getting wire, a getting wire and a beneficiary and processor to choose properties of the thing. Radio waves from the transmitter skip off the article and return to the recipient, giving information about the thing's region and speed. Radar was developed secretly for military use by a couple of nations in the period beforehand and during The Subsequent Extraordinary Conflict.

Figure 2.1 Radar screen

2.2 Electro-Optical Sensors

Electro-Optical Sensors keep on advancing in execution and goal, requiring more signals handling power for better objective separation, lower mistake rates, and further developed precision. Programmed target following empowers the sensor subsystem to distinguish and follow different focuses through mess, climate, air aggravations, and countermeasures intended to cloud the objective. Electro-optical sensors are utilized broadly in ground-to-air, air-to-endlessly air to-ground fire control frameworks, reconnaissance, security, and border guard frameworks. Sensor pictures are handled by a global positioning framework to distinguish and follow items like airplane, vehicles, or faculty that are moving inside a bunch of recommended qualities for those specific sorts of articles. Sensor innovation keeps on developing quickly, requiring a tantamount expansion in signal handling power as pixel count, outline rates, and simple to-computerized transformation goals increment. Ongoing higher-goal sensors send video in computerized designs, including Camera Connection, DVI, HD-SDI, or GbE Vision. The crude computerized video information could contain a lot of commotion or mess or have unfortunate difference. This frequently applies across the entire picture, so a profoundly equal handling gadget, for example, a FPGA is appropriate to sifting the substance, leaving all the more plainly discernable objective information for nitty gritty examination. GE Clever Stages' Skilled 5000 elite execution, embeddable tracker joins these key FPGA, DSP, and 3G Cell phone figuring innovations to make a rough, center following module sufficiently little to be mounted on a PC/104-Larger estimated (3.6 x 3.8 inches90 x 96 mm) I/O base card.

Figure 2.2 ADEPT 5000

2.3 Infrared thermography

Infrared thermography (IRT), thermal imaging, and thermal video are examples of infrared imaging science. Thermographic cameras typically distinguish radiation in the long-infrared scope of the electromagnetic range (about 9,000-14,000 nanometers or 9-14 μm) and produce pictures of that radiation, called thermograms. Since infrared radiation is discharged by all items with a temperature above outright zero as indicated by the dark body radiation regulation, thermography makes it conceivable to see one's current circumstance regardless of noticeable light. How much radiation transmitted by item increments with temperature; subsequently, thermography permits one to see varieties in temperature. At the point when seen through a warm imaging camera, warm items stand apart well against cooler foundations; people and other warm-blooded creatures become effectively noticeable against the climate, day or night. Subsequently, thermography is especially helpful to the military and different clients of observation cameras. Warm pictures, or thermograms, are really visual presentations of how much infrared energy discharged, communicated, and reflected by an article. Since there are various wellsprings of the infrared energy, it is hard to get an exact temperature of an article utilizing this strategy. A warm imaging camera is equipped for performing calculations to decipher that information and fabricate a picture.

Figure 2.3 IRT display screen

2.4 CHALLENGES FACED BY EXISTING TECHNOLOGY

Soldiers experience a few life and passing circumstances. Among them some perhaps connected with the hardware which they are utilizing. A portion of the difficulties looked by existing innovation are expressed underneath.

- Requires a couple of moments to lock on, in disaster areas consistently could welcome another danger there by perhaps neglecting to counter go after on time.
- Powerless against sticking.
- In IRT numerous cameras don't give the irradiance estimations used to develop the result picture, the deficiency of this data without a right adjustment for emissivity, distance, and surrounding temperature and relative moistness involves that the resultant pictures are intrinsically wrong estimations of temperature.
- Preparing administrator is troublesome, as the tasks are a piece muddled.

Chapter 3

MACHINE VISION BASED AUTO TRACKING AND TARGET LOCKING SYSTEM

Machine Vision based Auto Tracking and Target Locking System is a system which tracks and locks enemy targets

3.1 BLOCK DIAGRAM

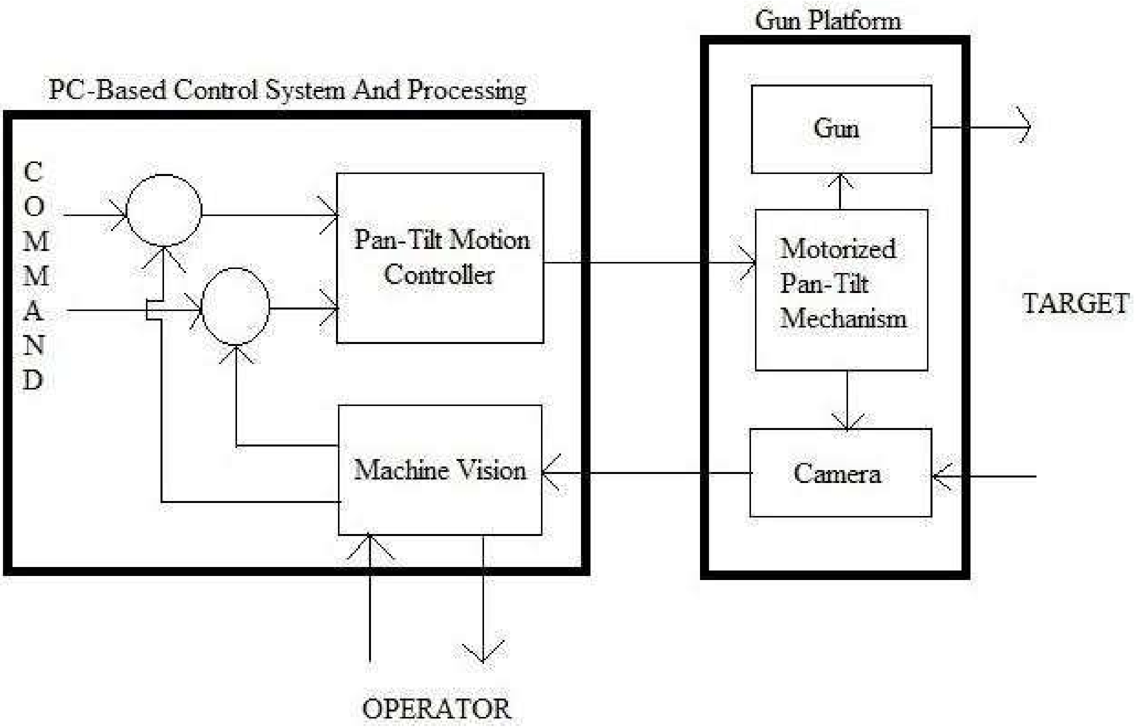


Figure 3.1 Defense Equipment

3.2 BLOCK DIAGRAM DESCRIPTION

The Defense Equipment's block diagram is shown in Figure. 3.1 The Defense Equipment mainly consists of

- PC-Based Control and Vision Processing
- Gun Platform (laser light).

The PC- Based Control and Vision Processing use image from camera placed on Gun Platform as primary vision data to be processed. The Pan-Tilt Motion Controller is used to control the motion of Gun Platform through the Motorized Pan-Tilt Mechanism. The Gun Platform is also equipped with Gun to shoot the Target if necessary.

3.2.1 Machine Vision

To achieve the adequate performance for the vision processing, the Machine Vision utilizes image pixel resolution of 720p resolution with image frame rates of 25 frames per second. The center position in image frame is used as command positions for locking the target.

Machine vision is a process to extract image information to get important information of an object. The information will be further processed in accordance with desired objectives. In the field of industry, machine vision can be used as replace of traditional sensors. For the example of machine vision application is counting the number of bottles and sorting the bottles using machine vision rather than multiple sensors. By using machine vision that replaces multiple sensors, it would make process more efficient and cost-effective. In the military field, machine vision can be used to detect the presence of enemies without direct contact of military personnel with the enemies. This can prevent unnecessary casualties on both sides. Machine vision is used for unmanned system that operated at far position from operator or frontline personals that make operator safer from direct contact with the hostile enemy.

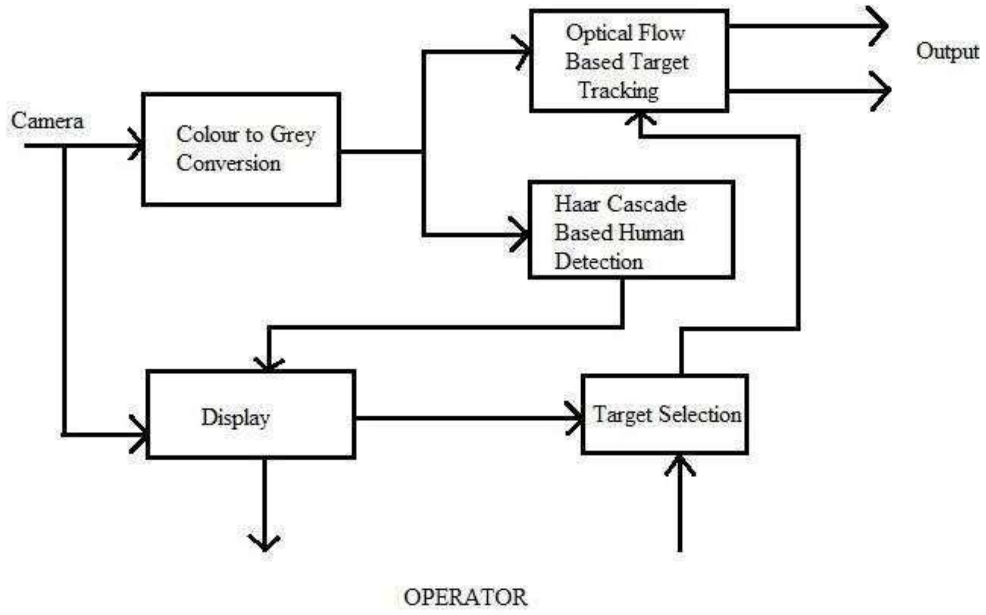


Figure 3.2 Block diagram of machine vision.

Mechanism:
The block diagram of the Machine Vision is shown in Figure 3.2. Machine Vision's block in the PC-Based Control and Vision Processing utilizes the color image captured from camera to be processed further. The color or black and white image is displayed to the Operator. The image is used as input of the CNN Algorithm in front of the camera and then displayed to the Operator. Based on image on Display, the Operator can decide the order to select the Target if necessary. When the operator has selected a Target, then the grey image will be processed with optical flow method to track the current positions of Target in image frame.

3.2.2 Target

We used the Bullseye image, Birds and Drones as our desired targets that has to be track and targeted

Figure 3.3 Bullseye Target image.

3.2.3 CNN Machine Learning Algorithm

Convolutional Neural Networks (otherwise called CNN or ConvNet) in deep learning, especially when it comes to Computer Vision applications. In significant learning, a Convolutional brain organization (CNN/ConvNet) is a class of significant mind associations, by and large normally applied to look at visual imagery. As of now when we consider a mind network we contemplate framework increments anyway that isn't correct with ConvNet. It uses a special strategy called Convolution. By and by in science convolution is a mathematical strategy on two capacities that conveys a third ability that imparts how the condition of one is changed by the other. Convolutional mind networks are made from various layers of phony neurons. Fake neurons, a cruel pantomime of their regular accomplices, are mathematical capacities that work out the weighted measure of various information sources and results inception regard. Right when you input an image in a ConvNet, each layer makes a couple of establishment works that are given to the accompanying layer.

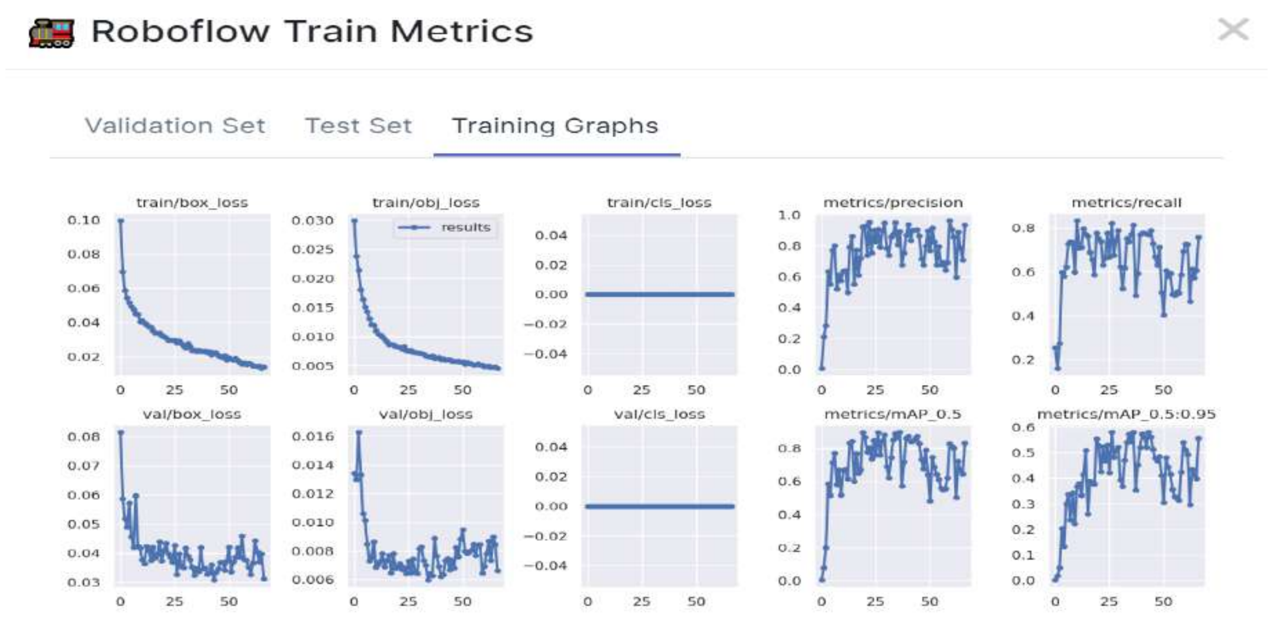
The principal layer, generally speaking, isolates key features like level or slanting edges. This outcome is given to the accompanying layer which recognizes more astounding features like corners or combinational edges. As we move further into the association it can perceive altogether more muddled features like articles, faces, etc. Considering the order guide of the last convolution layer, the gathering layer yields a lot of conviction scores (esteems some place in the scope of 0 and 1) that decide how likely the image is to have a spot with a "class." For instance, in case you have a ConvNet that perceives cats, canines, and horses, the consequence of the last layer is the probability that the data picture contains any of those creatures.

3.2.4 Roboflow

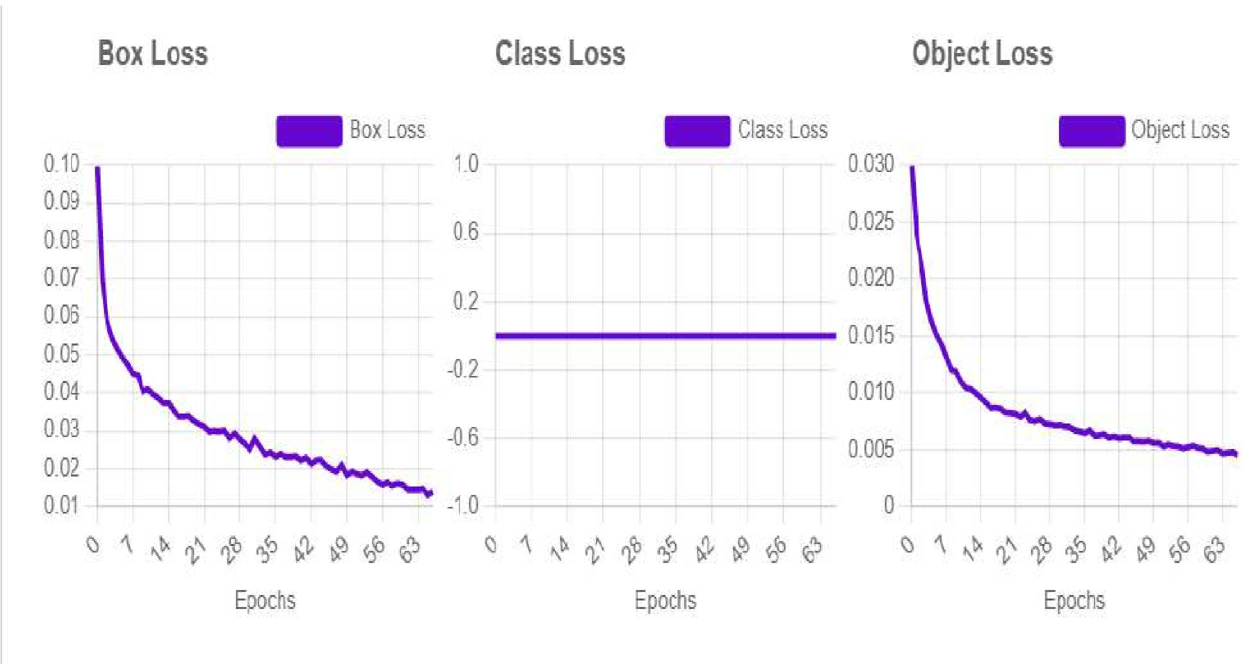
Roboflow is a Computer Vision designer system for better information assortment to preprocessing, and model preparation strategies. Roboflow has public datasets promptly accessible to clients and approaches for clients to transfer their own custom information moreover. Roboflow acknowledges different comment designs. In information pre-handling, there are steps included like picture directions, resizing, differentiating, and information expansions.

The whole work process can be co-ordinate with groups inside the structure. For model preparation, there's a lot of model libraries currently present like Efficient Net, Mobile Net, Consequences be damned, Tensor Flow, PyTorch, and so on. From there on model arrangement and perception choices are additionally accessible thus enveloping the whole

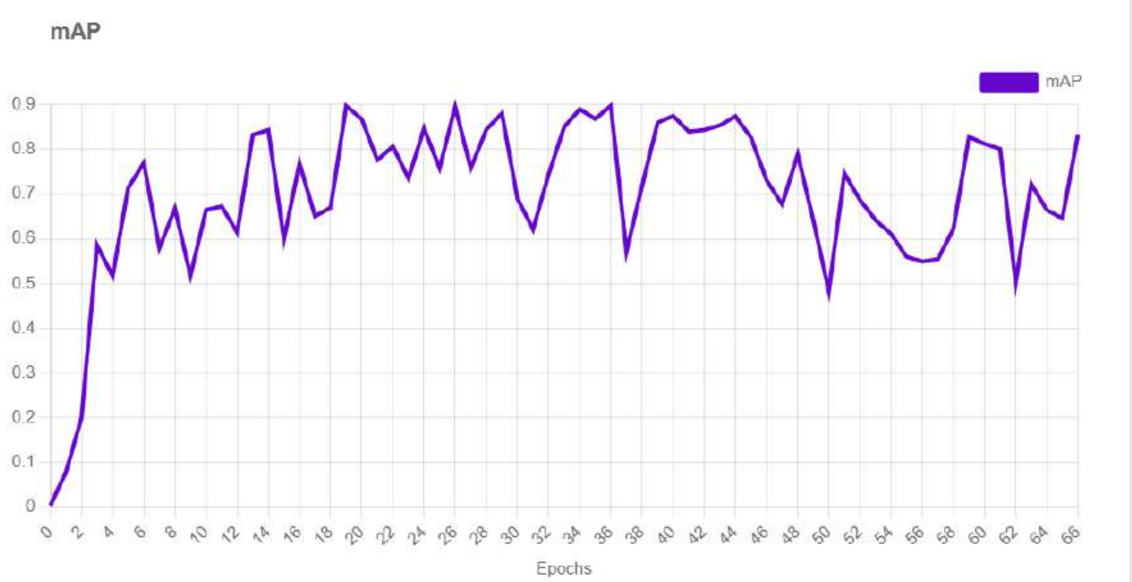
condition of-workmanship.
Roboflow is utilized in different computer vision businesses for use cases, for example, - gas spill discovery, plant versus weed identification, plane support, rooftop harm assessor, satellite symbolism, self-driving vehicles, traffic counter, trash cleaning, and some more.
We uploaded 3000 images of our target (Bullseye image) in given link;
https://app.roboflow.com/national-university-of-technology-islamabad-kms39/target-ir9dp/deploy/1
From here the data set is generated after we trained the images in Roboflow that provided the below results;



Graph 3.1 Roboflow Train Metric.



Graph 3.2 Epochs results for Bullseye target.



Graph 3.3 mean Average Precision results for Bullseye target

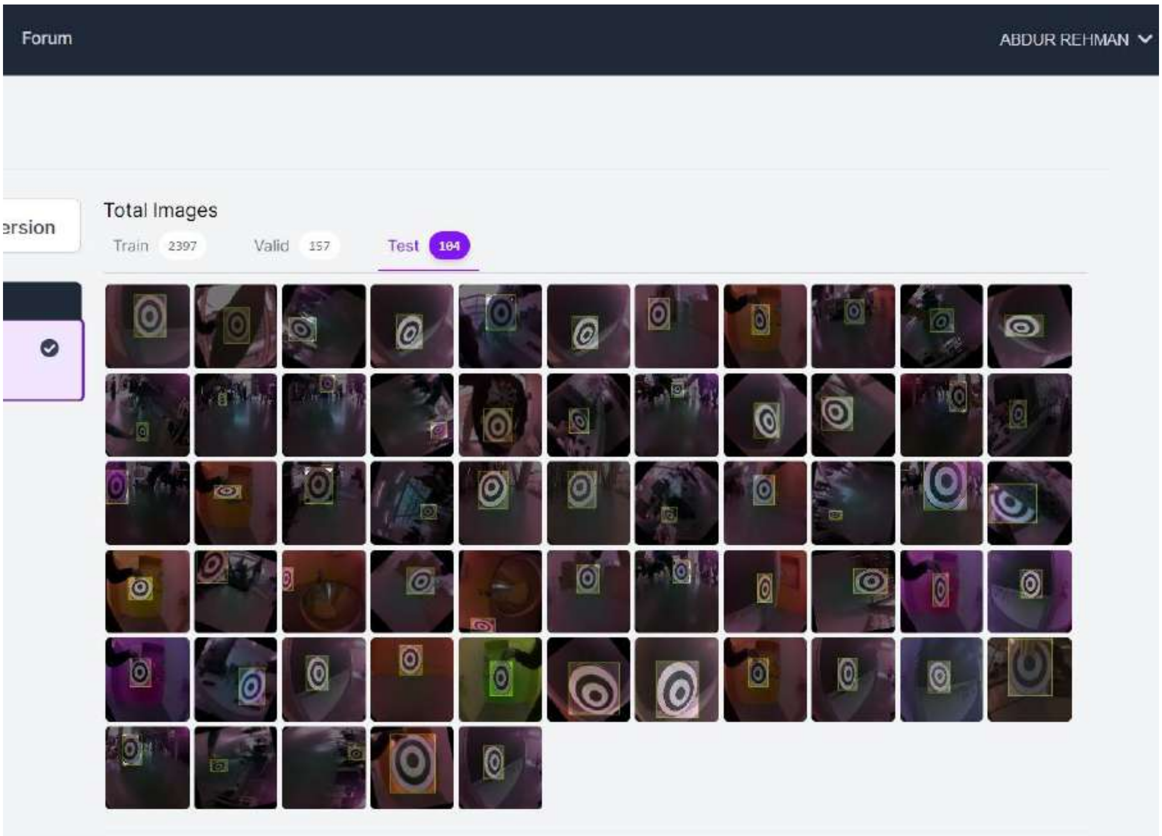


Figure 3.4 Roboflow after trained model for Bullseye target.

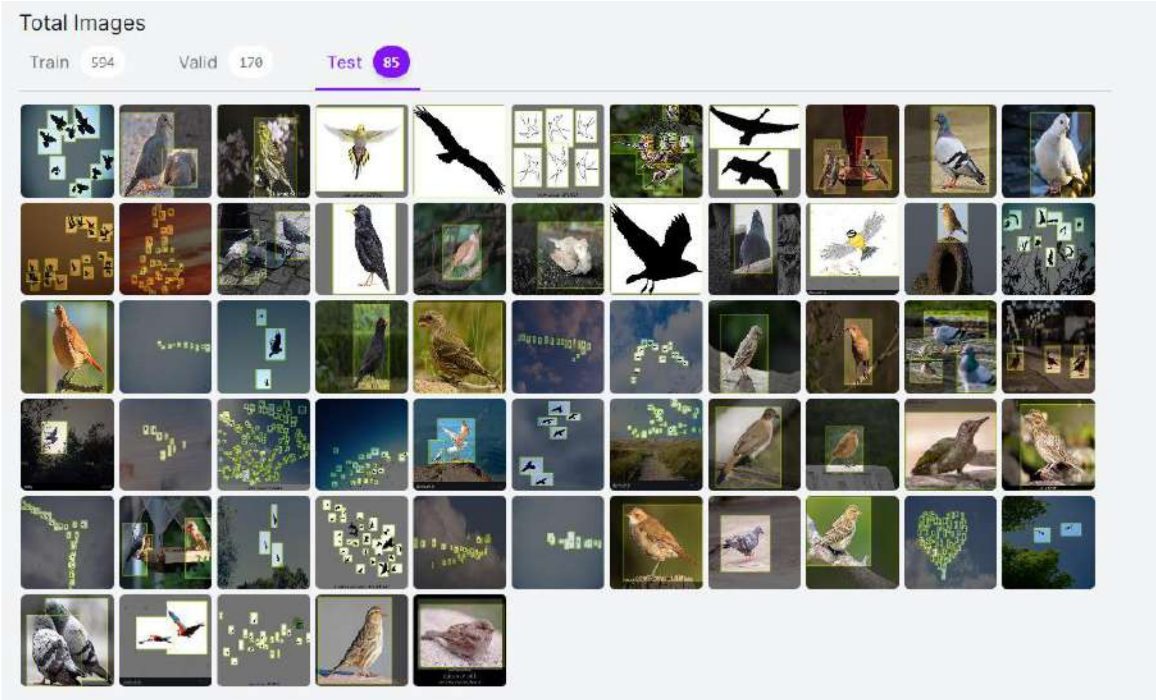


Figure 3.5 Roboflow after trained model for Birds as target.

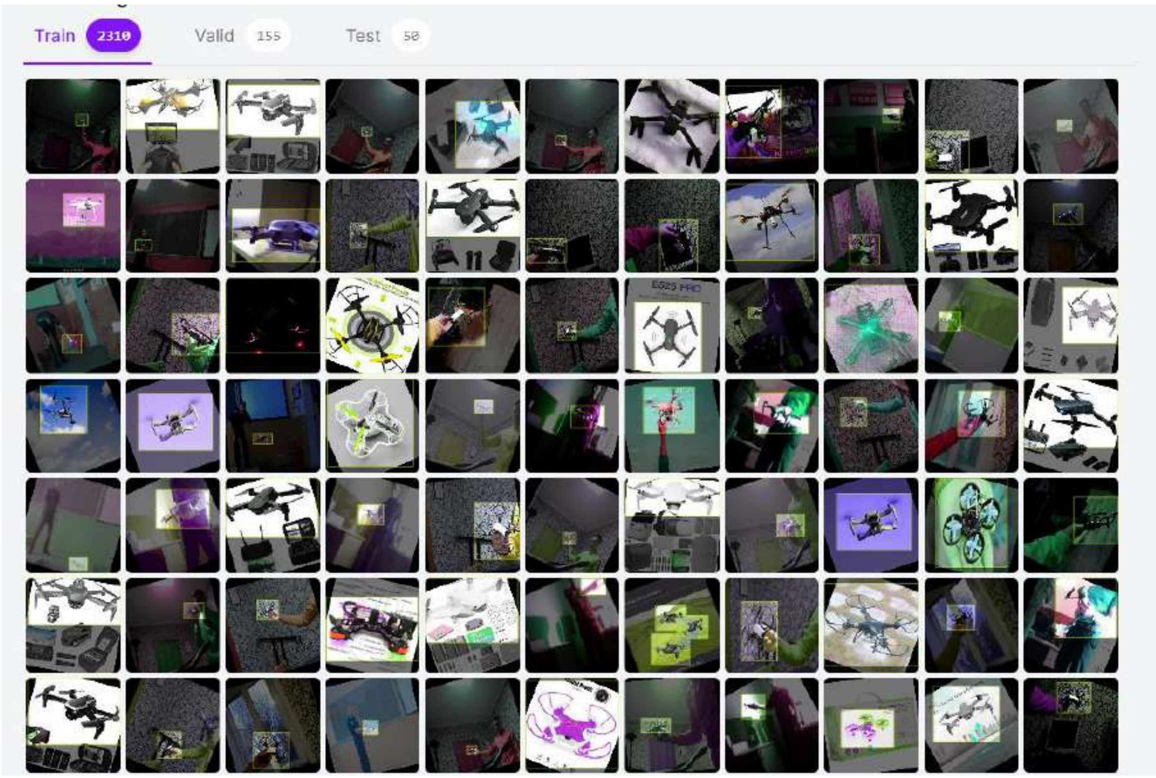


Figure 3.6 Roboflow after trained model for Drone as target.

3.2.5 PyTorch

PyTorch is an open source machine learning (ML) framework based on the Python programming language and the Torch library. Torch is an open source ML library used

for creating deep neural networks and is written in the Lua scripting language. It's one of the preferred platforms for deep learning research. The framework is built to speed up the process between research prototyping and deployment.
The PyTorch framework supports over 200 different mathematical operations. PyTorch popularity continues to rise, as it simplifies the creation of artificial neural network models.

3.2.6 YOLOv5

YOLOv5 is a model in the You Only Look Once (YOLO) family of computer vision models. YOLOv5 is commonly used for detecting objects. YOLOv5 comes in four main versions: small (s), medium (m), large (l), and extra large (x), each offering progressively higher accuracy rates. Each variant also takes a different amount of time to train. YOLOv5 derives most of its performance improvement from PyTorch training procedures, while the model architecture remains close to YOLOv4.
We used the YOLOv5 instead of YOLOv4 because the v5 has more accuracy but slow speed as compare to v4 that has high speed and low accuracy.
Then we exported our data set from Roboflow by format YOLOv5-PyTorch by the bellow Code:

```
!pip install roboflow
from roboflow import Roboflow
rf = Roboflow(api_key="B0df2s3iZ7PvWgVQMAsk")
project = rf.workspace("national-university-of-technology-islamabad-kms39").project("target-ir9dp")
dataset = project.version(1).download("yolov5")
```

3.2.7 Epochs

The quantity of epochs is a hyper parameter that characterizes the number times that the learning algorithm will manage the whole preparation dataset.
One epoch implies that each example in the preparation dataset has had an amazing chance to refresh the interior model boundaries. An epochs is included at least one clumps. For instance, as over, an epoch that has one clump is known as the group slope drop learning algorithm.
An epoch is the point at which all the preparation information is utilized immediately and is characterized as the all out number of iterations of all the preparation information in one cycle for preparing the ML model.
One more method for characterizing an epochs is the quantity of passes a preparation dataset takes around a calculation. One pass is counted when the informational index has done both forward and in reverse passes.

Confusion Matrix

A confusion matrix is a grid of information that shows the number of True Positives [TP], False Positives [FP], True Negatives [TN], and False Negatives [FN] returned when applying a test set of data to a classification algorithm. Using a confusion matrix, you can learn how many times your model makes correct and false predictions.
Confusion matrices can be used in both single-label and multi-label classification. Single-label classification is where a single label is assigned to classify an image. Single-label classification algorithms will be plotted on a 2x2 confusion matrix. Multi-label classification is where multiple labels can be assigned to classify an image. This method

of classification will be plotted on a larger matrix, depending on how many labels should be assigned to each image.

F1 Confidence Curve
F1 score is a machine learning evaluation metric that measures a model's accuracy. It combines the precision and recall scores of a model. The accuracy metric computes how many times a model made a correct prediction across the entire dataset.

Precision Confidence Curve
The precision-recall curve shows the tradeoff between precision and recall for different threshold. A high area under the curve represents both high recall and high precision, where high precision relates to a low false positive rate, and high recall relates to a low false negative rate.

Precision Recall Curve
The precision-recall curve shows the tradeoff between precision and recall for different threshold. A high area under the curve represents both high recall and high precision, where high precision relates to a low false positive rate, and high recall relates to a low false negative rate.

We finded the different results at the different Quantities of Epochs by code:

!python train.py --img 416 --batch 16 --epochs 50 --data {dataset.location}/data.yaml -- weights yolov5s.pt –cache

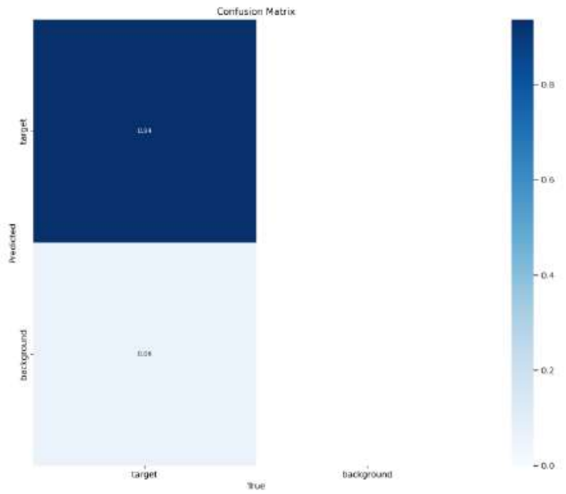
Epochs Results for Bullseye Target

For Epochs 50 we got the below graphs:

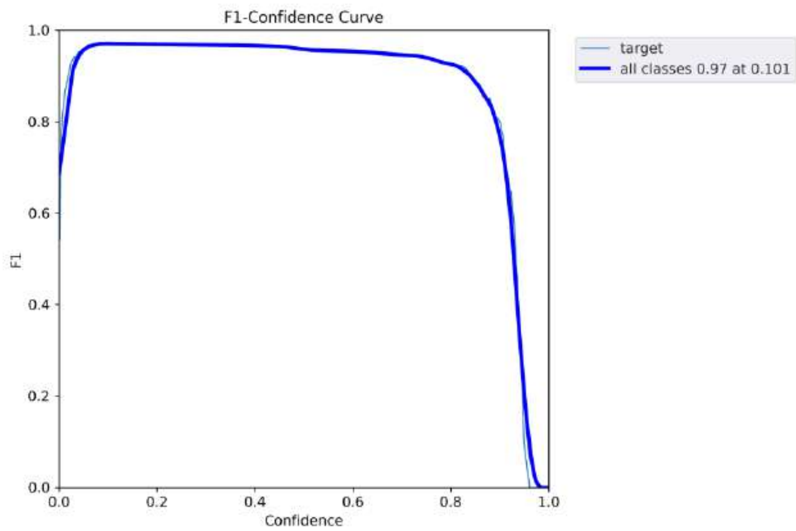
```
50 epochs completed in 0.316 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 14.3MB
Optimizer stripped from runs/train/exp/weights/best.pt, 14.3MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:03<00:00,  1.51it/s]
                   all        157        157          1      0.942      0.975      0.771
Results saved to runs/train/exp
```

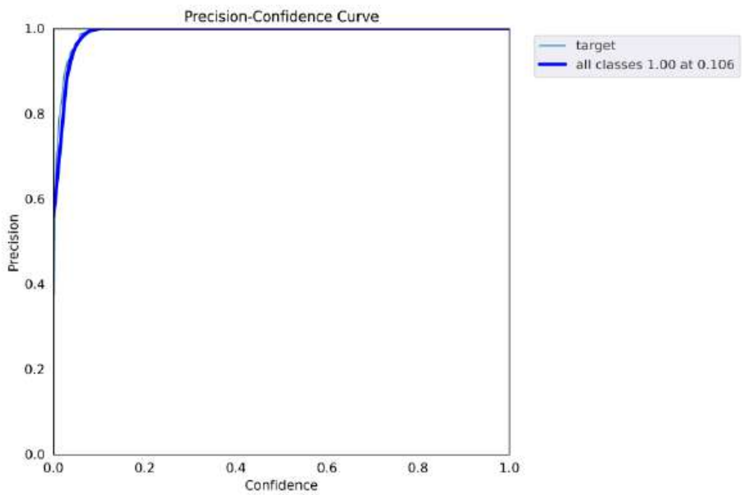
Figure 3.7 Epochs for 50



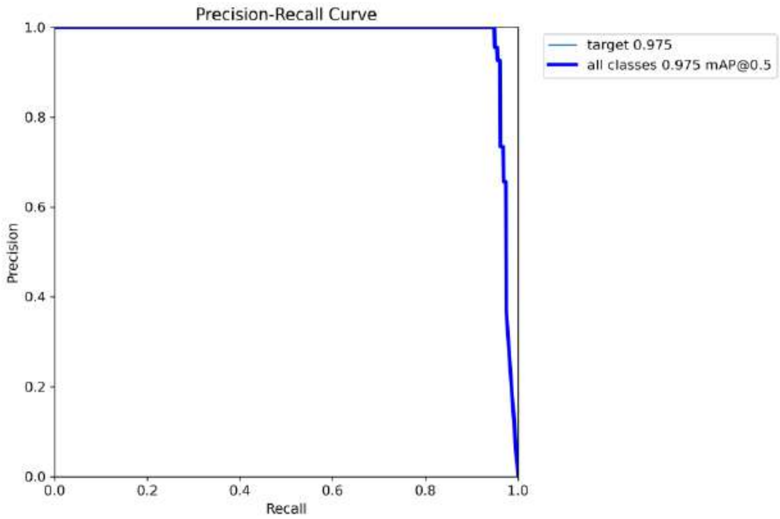
Graph 3.4 Confusion Matrix epochs 50



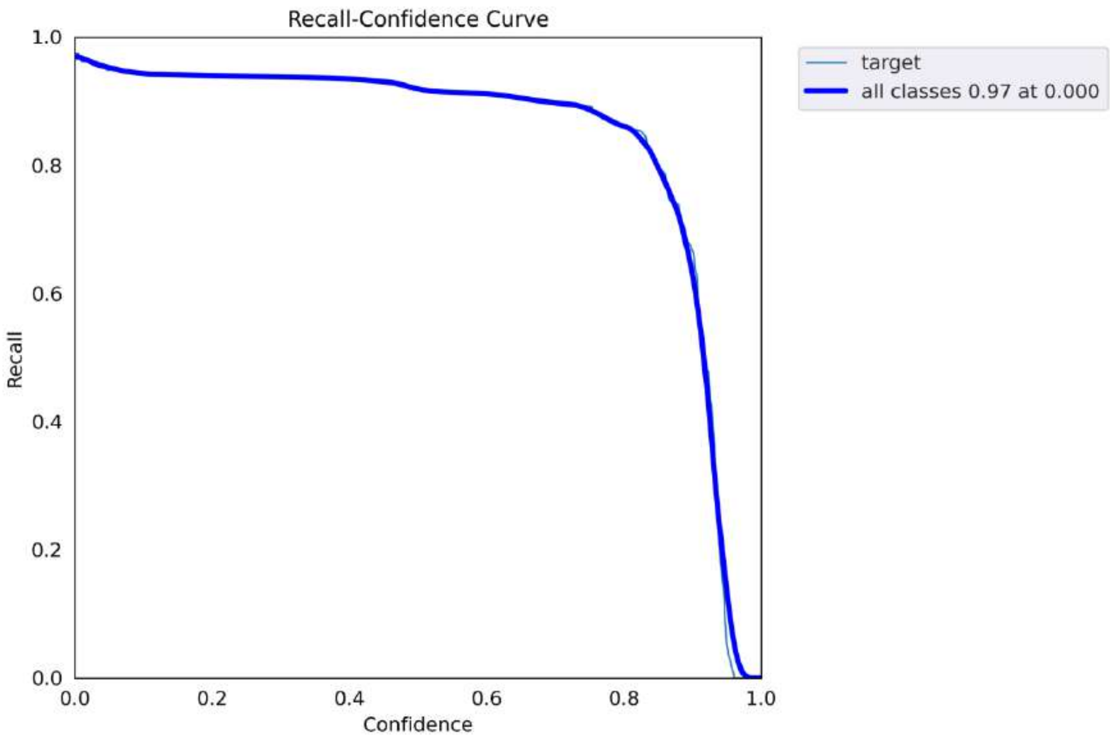
Graph 3.5 F1 Curve epochs 50



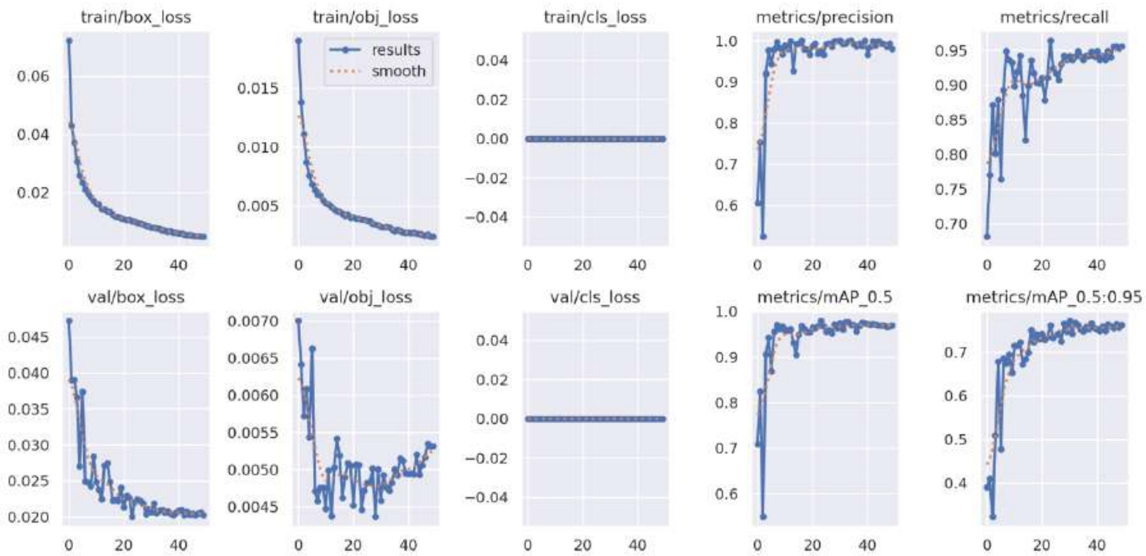
Graph 3.6 P Curve epochs 50



Graph 3.7 PR Curve epochs 50

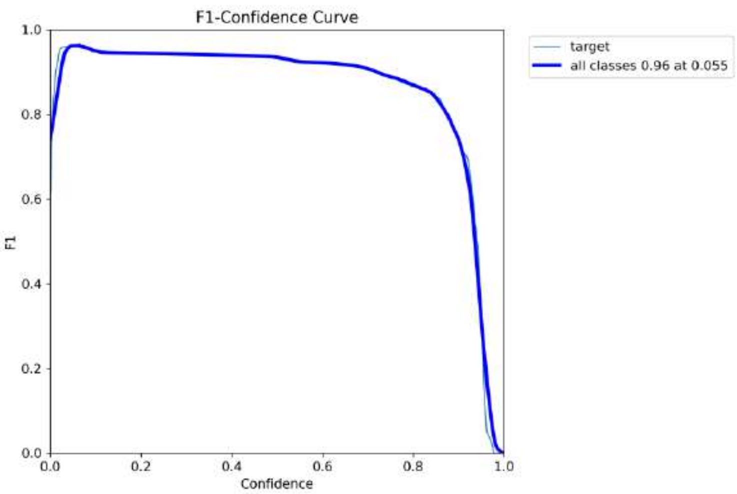


Graph 3.8 R Curve epochs 50

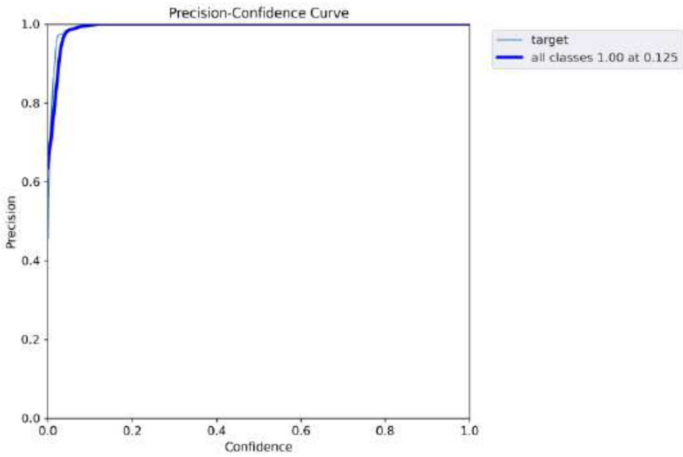


Graph 3.9 Results epochs 50

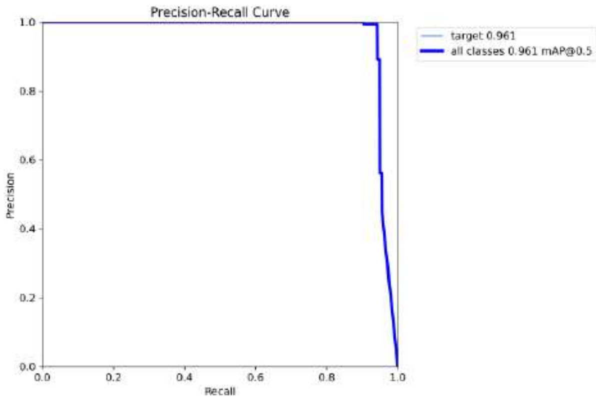
For Epochs 100 we got the below graphs:



Graph 3.10 F1 Curve epochs 100



Graph 3.11 P Curve epochs 100



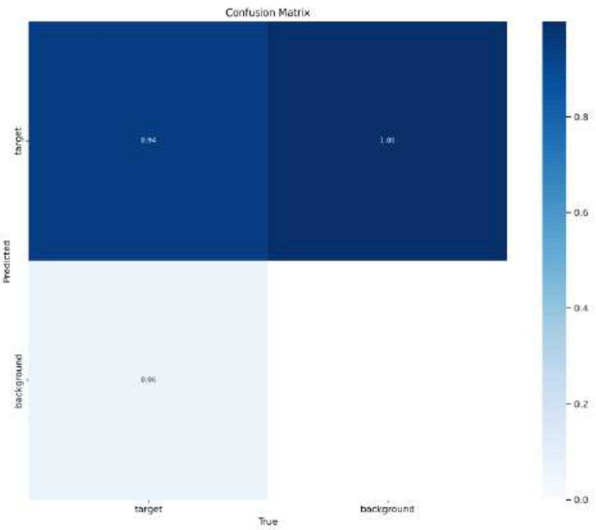
Graph 3.12 PR Curve epochs 100

For Epochs 150 we got the below graphs:

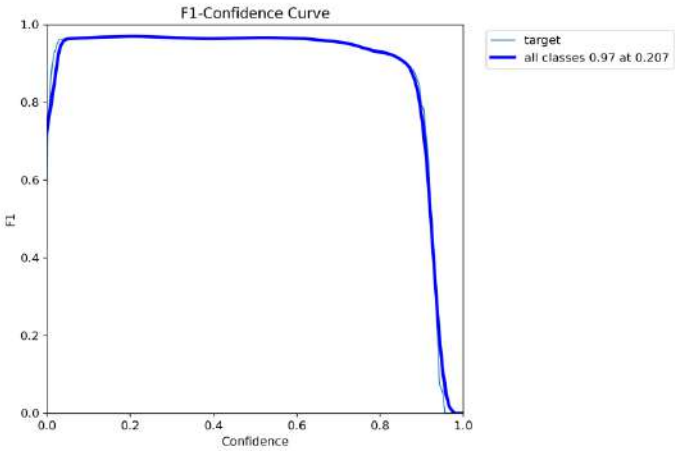
```
145 epochs completed in 0.977 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 14.3MB
Optimizer stripped from runs/train/exp/weights/best.pt, 14.3MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:03<00:00,  1.58it/s]
                   all        157        157      0.993      0.949      0.972      0.775
Results saved to runs/train/exp
```

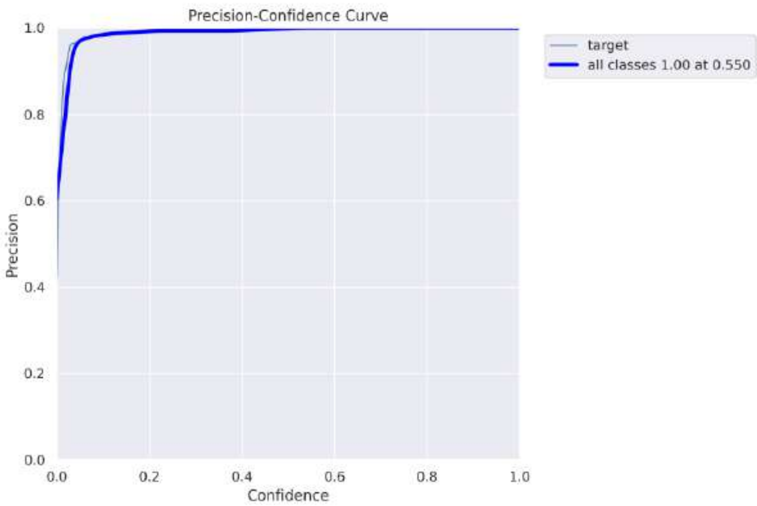
Figure 3.8 Epochs for 150



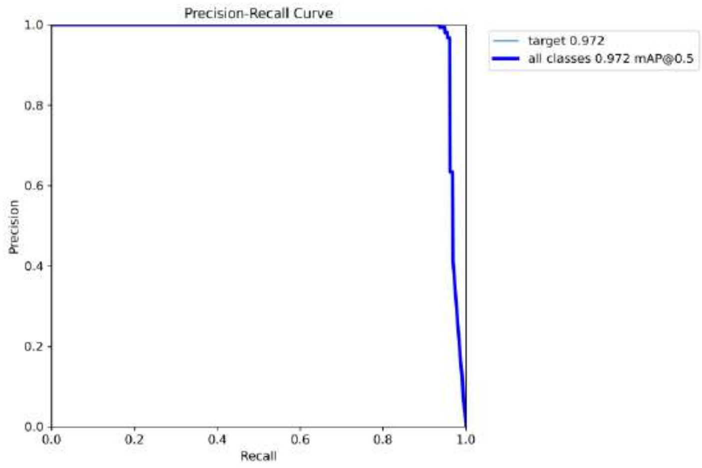
Graph 3.13 Confusion Matrix epochs 150



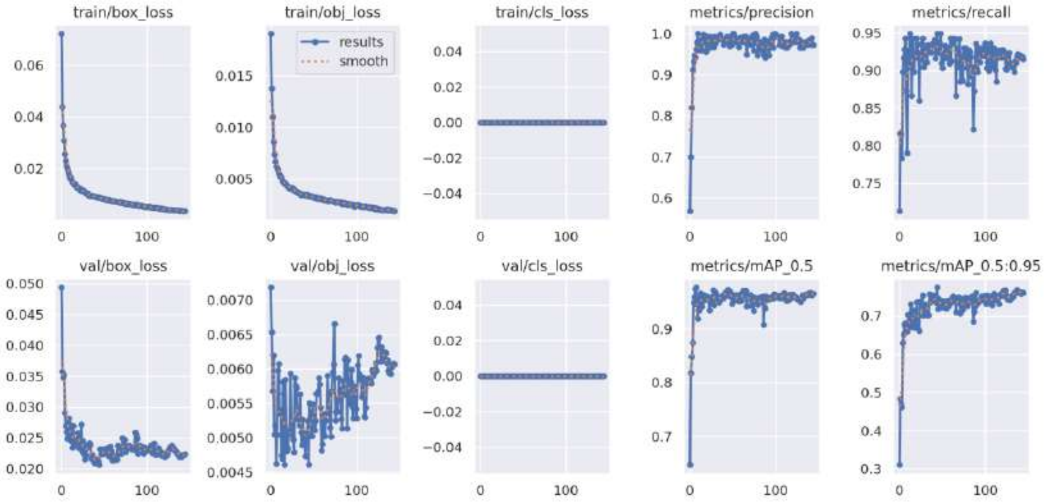
Graph 3.14 F1 Curve epochs 150



Graph 3.15 P Curve epochs 150



Graph 3.16 PR Curve epochs 150



Graph 3.17 Results epochs 150

Epochs Results for Birds as Target

```
     Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
     48/49       1.7G    0.04566    0.04096          0         11        416: 100% 38/38 [00:05<00:00,  6.46it/s]
                Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 6/6 [00:01<00:00,  5.00it/s]
                  all        170       2011      0.761      0.631      0.658      0.237

     Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
     49/49       1.7G    0.04684    0.04072          0          6        416: 100% 38/38 [00:07<00:00,  5.22it/s]
                Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 6/6 [00:01<00:00,  5.17it/s]
                  all        170       2011      0.766      0.628      0.656      0.236

50 epochs completed in 0.117 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 14.3MB
Optimizer stripped from runs/train/exp/weights/best.pt, 14.3MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs
                Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 6/6 [00:10<00:00,  1.80s/it]
                  all        170       2011       0.76      0.638      0.665      0.238
Results saved to runs/train/exp
```

Figure 3.9 Epochs for Bird as target

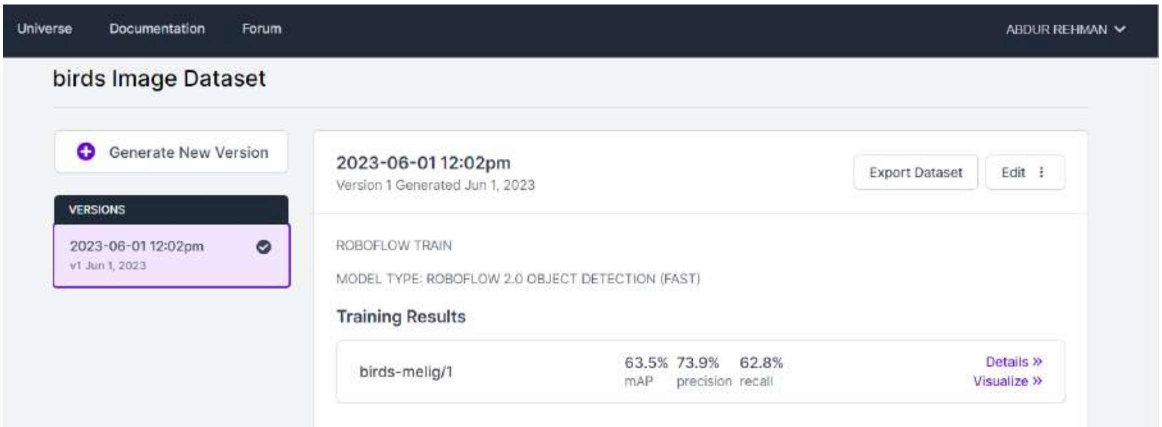
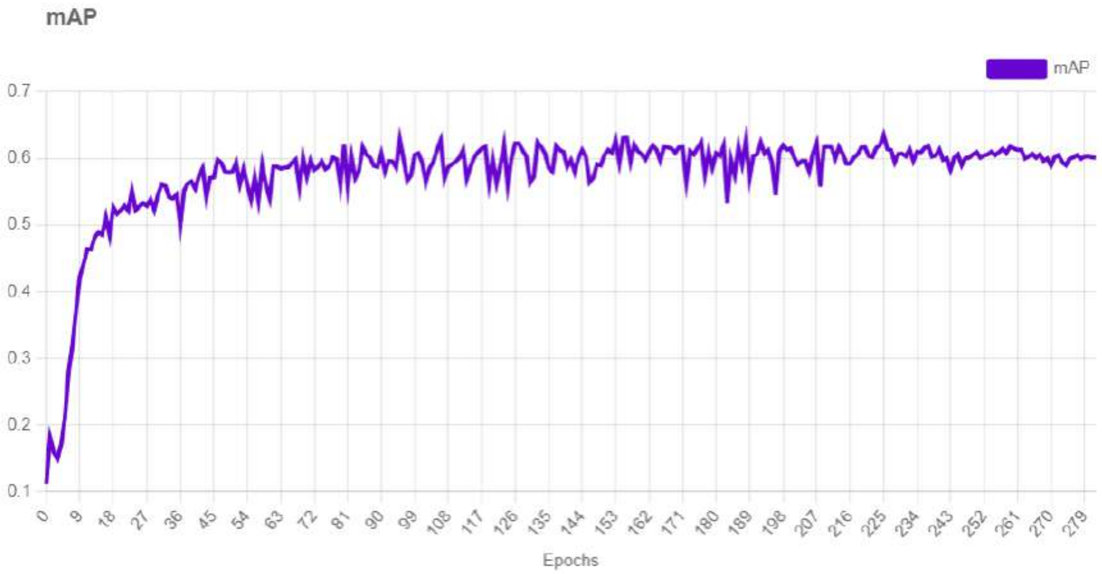


Figure 3.10 Birds image dataset



Graph 3.18 mean Average Precision results for Birds as target

Graph 3.19 Epochs results for Birds as target.

Epochs Results for Drones as Target

```
      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      46/49      1.72G    0.01988   0.008852          0         12        416: 100% 145/145 [00:19<00:00,  7.34it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:00<00:00,  5.14it/s]
                   all        155        169      0.995      0.941      0.975      0.631

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      47/49      1.72G    0.02016   0.008874          0          9        416: 100% 145/145 [00:19<00:00,  7.43it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:01<00:00,  3.76it/s]
                   all        155        169          1      0.929      0.977      0.644

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      48/49      1.72G    0.01983    0.00881          0         16        416: 100% 145/145 [00:19<00:00,  7.50it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:00<00:00,  5.25it/s]
                   all        155        169      0.976      0.941      0.975      0.639

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      49/49      1.72G    0.01929    0.00859          0         12        416: 100% 145/145 [00:19<00:00,  7.35it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:01<00:00,  3.74it/s]
                   all        155        169      0.994      0.934      0.976      0.635

50 epochs completed in 0.291 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 14.3MB
Optimizer stripped from runs/train/exp/weights/best.pt, 14.3MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7012822 parameters, 0 gradients, 15.8 GFLOPs
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 5/5 [00:03<00:00,  1.56it/s]
                   all        155        169      0.993      0.941      0.976      0.655
Results saved to runs/train/exp
```

Figure 3.11 Epochs for Drone as target

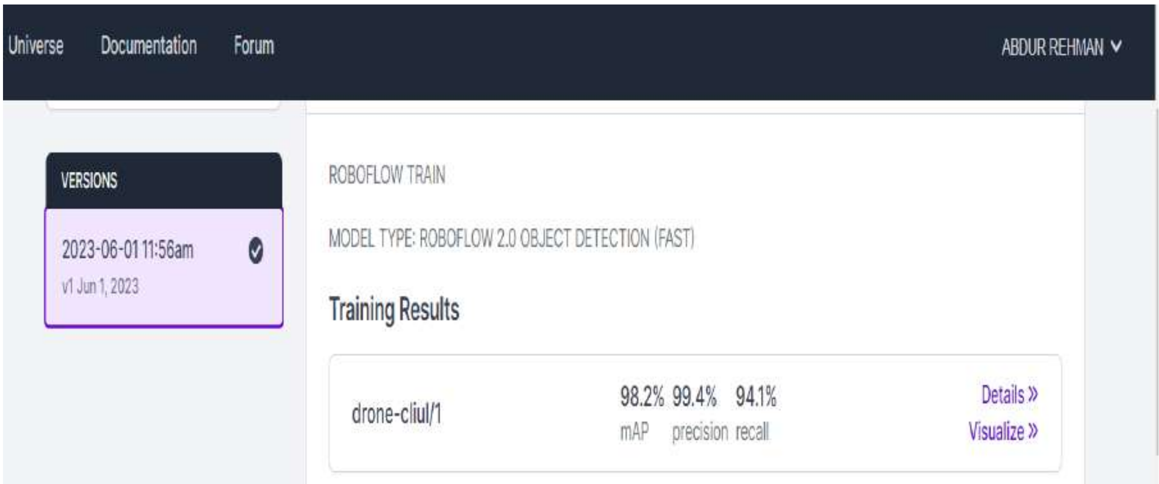
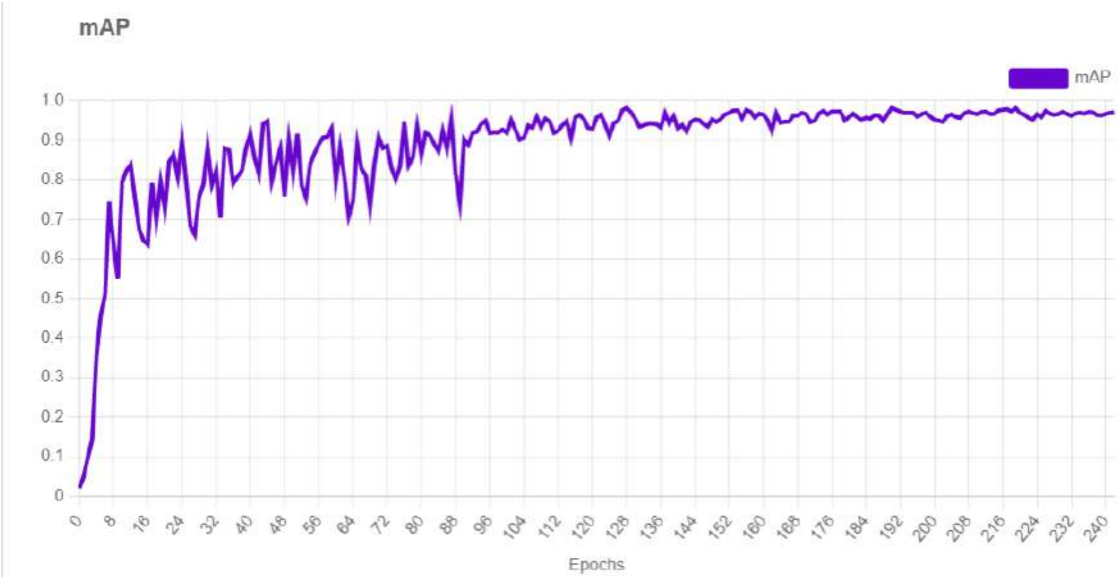


Figure 3.12 Drones image dataset



Graph 3.20 mean Average Precision results for Drone as target

Graph 3.21 Epochs results for Drone as target.

3.2.8 Main Code for Python IDE PyCharm

After training the model from the above process we generate file and use this file for our code that we write in python IDE PyCharm, the code is as below:

```
import cv2
import torch
import serial
import time
ser = serial.Serial('COM5', 115200)  # Replace 'COM3' with the appropriate port for your Arduino
model = torch.hub.load('yolov5', 'custom', path='targetservo.pt', source='local')  # localrepo
```

```
cap = cv2.VideoCapture(0)
last_detection_time = time.time()
box_center_x = 0
box_center_y = 0
while True:
    ret, frame = cap.read()
    result = model(frame, size=340)
    df = result.pandas().xyxy[0]
    for ind in df.index:
        x1, y1 = int(df['xmin'][ind]), int(df['ymin'][ind])
        x2, y2 = int(df['xmax'][ind]), int(df['ymax'][ind])
        label = df['name'][ind]
        conf = df['confidence'][ind]
        if float(conf.round(decimals=2)) >= 0.90:
            cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 0, 255), 2)
            cv2.putText(frame, label, (x1, y1 - 5), cv2.FONT_HERSHEY_PLAIN, 2,
(0, 0, 255), 2)
            # Calculate the center of the box
            box_center_x = int((x1 + x2) / 2 * (180 / frame.shape[1]))
            box_center_y = int((y1 + y2) / 2 * (180 / frame.shape[0]))
            print('box_center_x:', box_center_x)
            print('box_center_y:', box_center_y)
            ser.write(("!" + str(box_center_x) + "@" + str(box_center_y) + "$" +
str(1)+"^").encode())
            ser.write(b'\n')
        else:
            ser.write(("!" + str(box_center_x) + "@" + str(box_center_y) + "$" +
str(0)+"^").encode())
            ser.write(b'\n')

    # Display the frame with bounding boxes
    cv2.imshow('Video', frame)
    cv2.waitKey(1)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

3.2.9 Pan-Tilt Motion Controller

The Pan-Tilt Motion Controller utilizes 2 servo motor drivers to control the motion of the Motorized Pan-Tilt Mechanism which uses 2 Servo motor to perform motion in pan and tilt direction. The x-axis and y-axis in picture outline are utilized as reference for dish movement and slant movement, separately. The dish and slant not entirely set in stone by the control calculation with inputs come from the order positions and the ongoing places of Target assessed in Machine Vision. The standard Corresponding Necessary Subsidiary (PID) control calculation is applied in the Container Slant Movement Regulator. The schematic of the Container Slant Stage (PTP) has two levels of opportunity. The PTP incorporates a base, a rotatable skillet component and a rotatable slant structure for supporting and situating the gadget in an ideal course. It can ceaselessly spin about the skillet pivot and 90 levels of movement range in the slant hub. The skillet instrument (base) is rotatable about a container pivot, and the slant system is rotatable about a slant hub upheld on the base, that is opposite to the dish hub. A dish engine and a slant engine drive the PTP. There is stuff on the shaft of the engine. Through the component of stuff, sprocket and belt, the force is moved to the construction.

3.3 HARDWARE EXPLANATION

The hardware of the Gun Platform consists of a camera for image capturing, Personal Computer (PC), Servo motors for pan-tilt movement, a gun mechanism (laser light) and Arduino Uno r3.

3.3.1 Gun mechanism (laser light)

The gun types used as gun platform is Laser light module that is connected to Arduino it will focus on the desired target and will track that target within the frame of Camera.

Figure 3.13 Laser Light Module

3.3.2 Camera

There isn't much difference between machine vision cameras and normal cameras. Machine vision cameras are normal working cameras with similar sensors and elements which are programmed or devised especially for monitoring purposes. They are designed to obtain some specific information in various scenarios. It is like an automated eye which picks out the objects depending on the algorithm fed into it. Generally the process consists of a specific place where the object to be monitored is placed and a camera mounted on some stand and then the readings/output of the camera is processed in different image processing software's to check for desired parameters. Machine vision cameras can take a lot of pressure, operate best in very low or very high amounts of light, and be set up in pairs for 3D vision, operate in UV or infrared.

3.3.3 Servo Motors for tilt motion

A servo motor is a kind of engine that can pivot with incredible accuracy. Ordinarily this sort of motor comprises of a control circuit that gives input on the ongoing place of the motor shaft, this criticism permits the servo motor to pivot with extraordinary accuracy. To pivot an item at a few explicit points or distance, then, at that point, you utilize a servo motor. It is comprised of a basic motor which goes through a servo component. A servo motor typically accompanies a stuff plan that permits us to get an extremely high force servo motor in little and lightweight bundles. A servo comprises of a motor, a potentiometer, gear get together, and a controlling circuit. As a matter of some importance, we use gear gathering to diminish RPM and to expand force of the motor. Say at starting place of servo motor shaft, the place of the potentiometer handle is with the end goal that there is no electrical sign created at the result port of the potentiometer. Presently an electrical sign is given to one more information terminal of the blunder finder enhancer. Presently the contrast between these two signs, one comes from the potentiometer and one more comes from different sources, will be handled in a criticism component and result will be given as far as blunder signal. This mistake signal goes about as the contribution for endlessly motor turns over pivoting. Presently motor shaft is associated with the potentiometer and as the motor turns so the potentiometer and it will create a sign. So as the potentiometer's precise position changes, it result criticism signal changes. After at some point the place of potentiometer comes to at a place that the result of potentiometer is same as outside signal gave. At this condition, there will be no result signal from the enhancer to the motor input as there is no contrast between outer applied signal and the sign created at potentiometer, and in this present circumstance motor quits turning.

3.3.4 Arduino Uno

Arduino Uno is a microcontroller board in light of the ATmega328P (datasheet). It has 14 computerized input/yield pins (of which 6 can be utilized as PWM yields), 6 simple information sources, a 16 MHz ceramic resonator (CSTCE16M0V53-R0), a USB association, a power jack, an ICSP header and a reset button. It contains everything expected to help the microcontroller; just interface it to a PC with a USB link or power it with an air conditioner to-DC connector or battery to begin. We can dabble with our Uno without stressing a lot over accomplishing something off-base, most dire outcome imaginable we can swap the chip for a couple of dollars and begin once more.

"Uno" signifies one in Italian and was decided to stamp the arrival of Arduino Programming (IDE) 1.0. The Uno board and variant 1.0 of Arduino Programming (IDE) were the reference renditions of Arduino, presently advanced to more up to date delivers. The Uno board is the primary in a progression of USB Arduino sheets, and the reference model for the Arduino stage; for a broad rundown of current, past or obsolete sheets see the Arduino record of sheets.

Code for the Arduino for the movement of the servo motors is as below:

```
#include <Servo.h>
Servo xx;
Servo yy;
unsigned long startTime = 0;  // Variable to store the start time
// Convert extracted string to integer
int extractedValue1 = 90;
int extractedValue2 = 90;
int extractedValue3 = 0;
int state = 0;
void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  xx.attach(9);
  yy.attach(10);
  xx.write(90);
  yy.write(60);
  pinMode(2, OUTPUT);

}
```

```
void loop() {
  if (Serial.available()) {
    String inputString = Serial.readStringUntil('\n'); // Read data from serial until
newline character
    int pos1 = inputString.indexOf('!'); // Find the position of '!' character
    int pos2 = inputString.indexOf('@'); // Find the position of '@' character
    int pos3 = inputString.indexOf('$'); // Find the position of '$' character
    int pos4 = inputString.indexOf('^'); // Find the position of '$' character
    if (pos1 != -1 && pos2 != -1 && pos3 != -1 && pos4 != -1) { // Check if both '!'
and '@' characters are found
      String extractedString1 = inputString.substring(pos1 + 1, pos2); // Extract
substring between '!' and '@' characters
      String extractedString2 = inputString.substring(pos2 + 1, pos3); // Extract
substring between '!' and '@' characters
      String extractedString3 = inputString.substring(pos3 + 1, pos4); // Extract
substring between '!' and '@' characters
      extractedValue1 = extractedString1.toInt();
      extractedValue2 = extractedString2.toInt();
      extractedValue3 = extractedString3.toInt();
    }
    if (extractedValue3 == 1)
    {
      digitalWrite(2, HIGH);
    }
    else
    {
      delay(50);
      digitalWrite(2, LOW);
      extractedValue1 = 90;
      extractedValue2 = 60;
    }
    extractedValue1 = map(extractedValue1, 20, 160, 75, 105);
    extractedValue2 = map(extractedValue2, 20, 160, 60, 80);
      if (extractedValue1 > 105 or extractedValue2 > 80)
      {
        extractedValue1 = 105;
        extractedValue2 = 80;
      }
      if (extractedValue1 < 75 or extractedValue2 < 60)
      {
        extractedValue1 = 75;
        extractedValue2 = 60;
      }
    xx.write(180 - extractedValue1);
    yy.write(extractedValue2 - 20);
  }
}
```

3.3.5 Final Hardware Pictures

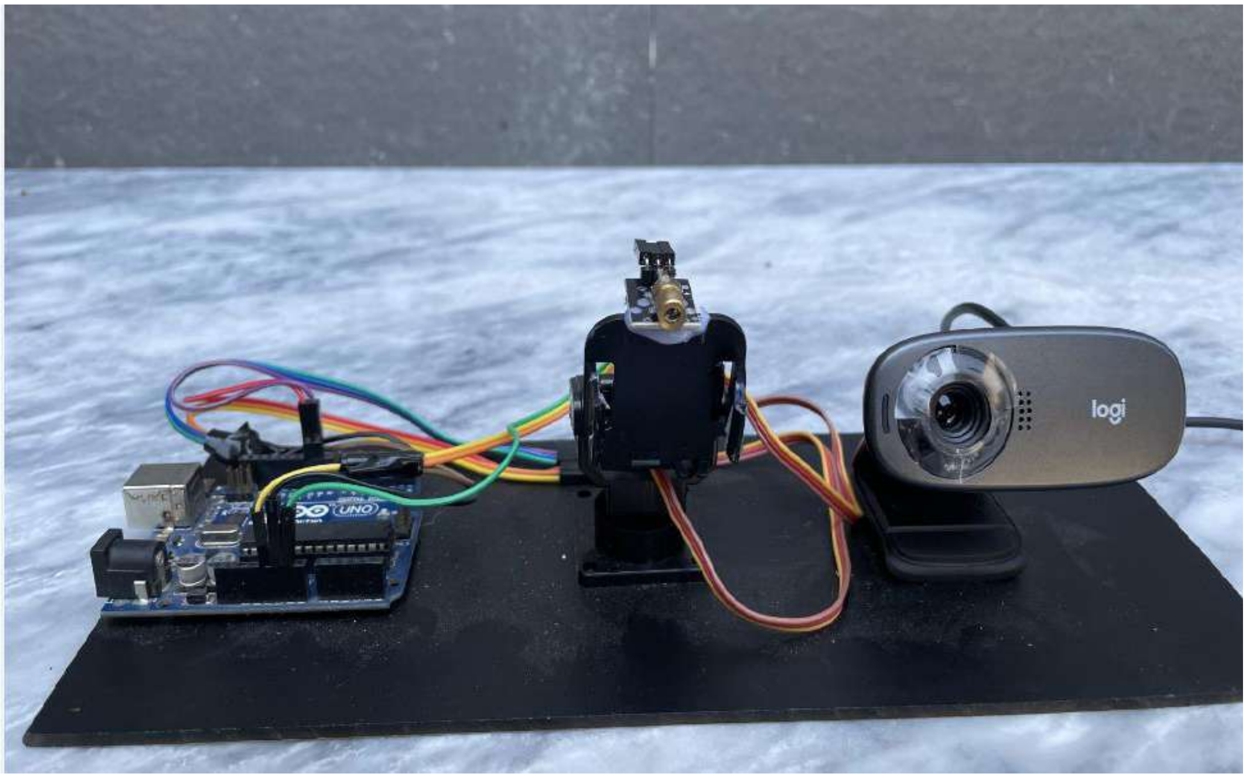


Figure 3.14 Final Prototype side wise Picture

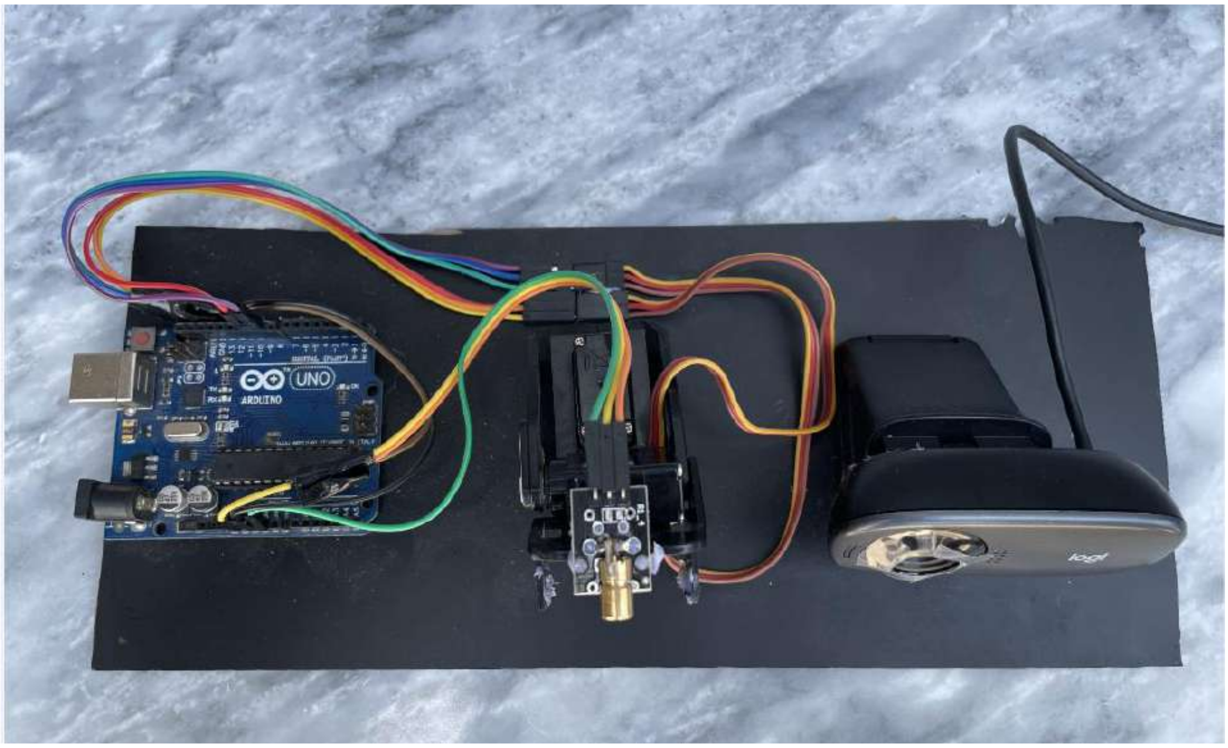


Figure 3.15 Final Prototype front wise Picture

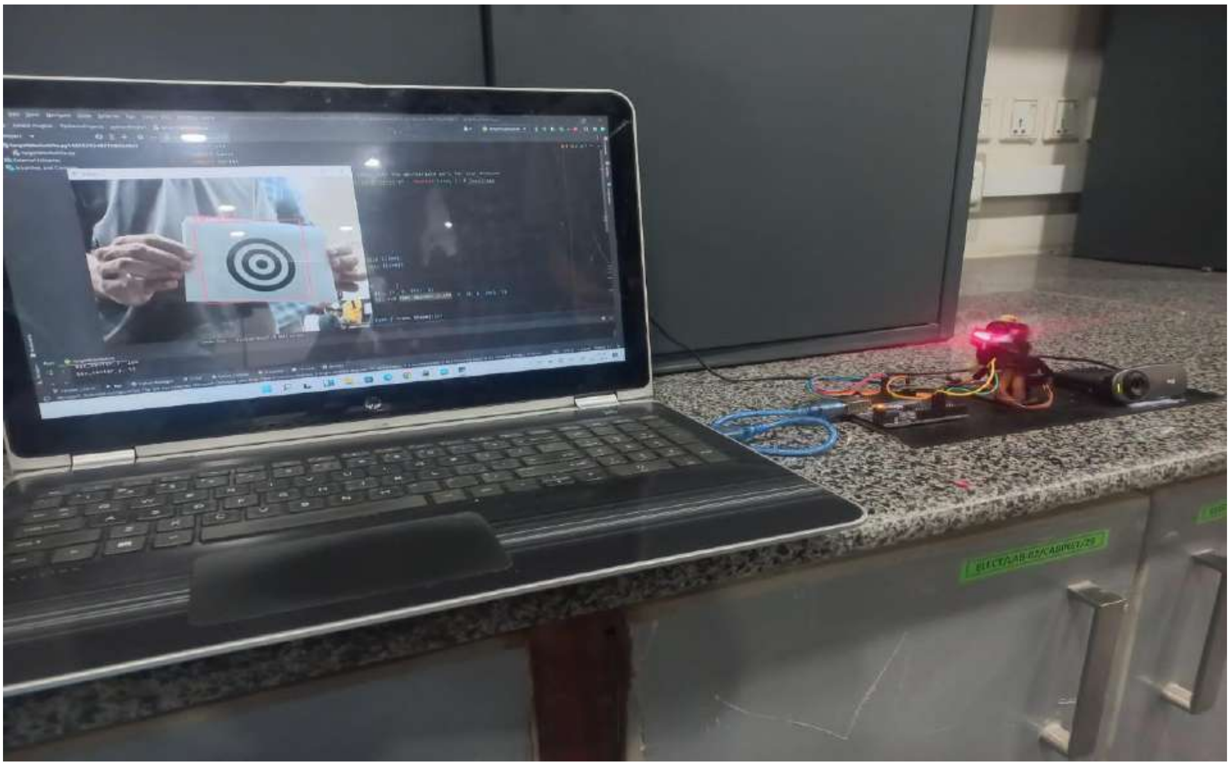


Figure 3.16 Tracking and target locking on Bullseye image as target

Figure 3.17 Tracking and target locking on Drone as target

Chapter 4

APPLICATIONS

Modifications made in the system will be of great use in upcoming time. Some applications of the system are;

1. Used in Modular Advanced Armed Robotic Systems (MAARS). MAARS are a powerful combat ready robot which can operate for reconnaissance mission, surveillance mission and target acquisition. The presence of MAARS can make frontline operator in conflict area safer because MAARS can replace operator presence in dangerous area. MAARS can be regarded as one of example of a defense system. The model of MAARS is shown as below:

Figure 4.1 Modular Advanced Armed Robotic System model

2. Can be used to guard army bases.
3. Can be used in places where human entry is difficult.
4. With slight adjustments it can be programmed to detect mines and other obstructions.
5. One of the main applications is to detect the birds near the airport area and to target or shoot these birds by the Gun instead of Laser light, as the birds near airports pose a significant risk to aircraft safety. Bird strikes occur when birds collide with aircraft during takeoff, landing, or in-flight. These collisions can damage critical aircraft components, including engines, windshields, and control surfaces, leading to accidents or emergency landings. Bird strikes can result in loss of control of the aircraft, engine failure, or reduced visibility for the pilots. These incidents jeopardize passenger safety and can cause injuries or fatalities. Therefore, the presence of birds near the airport area can have severe consequences for aviation safety.

FUTURE WORK

Speed or Accuracy can be increased by using Raspberry Pi 4 in the place of Arduino Uno because it is fast and can perform more accurately and in a best way because it is the latest product in the Raspberry Pi.

OPERATIONAL PROCEDURE

- Firstly we gather picture of our desired target.
- Then compile these pictures in a folder and named it as target.
- Then create our account in ROBOFLOW.
- Import these files in ROBOFLOW we get our desired data set splited into Test and train.
- Then export this data set to Google Colabs where we create a virtual workspace and train our target file according to desired epoch.
- Once the processing done in Google colab we get a Best.pt file which contains all the calculation and operations of our training data set.
- Download best.pt file and import it to the code written in python (Py charm).
- Then write the code for Arduino for controlling Pan-Tilt motion and leaser light.
- Set the baud rate 11520 for both Arduino and camera code for smooth communication between camera module and Arduino.
- Make sure the port of Arduino connected with PC must be same as in code.
- Then we turn on camera for real-time input.

CONCLUSION

This presents an alternative approach to auto tracking and target locking systems by leveraging visual perception through cameras. The system utilizes image processing techniques, such as Convolutional neural networks (CNNs), Roboflow, Epochs, PyTorch, and YOLOv5 libraries, to capture visual information and estimate the position of static objects within an image. By using cameras instead of traditional radar or opto-electrical technology, the system aims to address vulnerabilities, reduce susceptibility to jamming, and lower procurement costs. The system offers several advantages, including automation of defense operations, improved accuracy, and the ability to operate in hostile environments. The hardware-based prototype developed as part of this research is capable of tracking static objects using computer vision, aiming the laser beam at the target once it is locked, and can be used for surveillance missions and target acquisition. The accuracy of the gun system (Laser light) reaches approximately 85%, providing reliable and robust performance. The scope of this research encompasses various aspects of machine vision, including target tracking, human detection, and object recognition. Challenges in auto tracking and target locking systems, such as occlusion handling, illumination variations, and object appearance changes, still need to be addressed in future research. Further efforts can focus on improving robustness, efficiency, and adaptability to complex environments by integrating machine learning and deep learning techniques. The proposed system has a wide range of applications, including military defense, guarding army bases, areas with difficult human entry, and detecting mines and obstructions. Additionally, the system can be employed to detect birds near airport areas, reducing the risk of bird strikes and enhancing aviation safety.

This presents a hardware-based prototype utilizing machine vision and image processing techniques to develop an automated auto tracking and target locking system. By leveraging cameras and advanced algorithms, the system offers a reliable, accurate, and cost-effective solution for defense operations, surveillance missions, and target acquisition. With further advancements and improvements, this technology holds great potential for enhancing security and safety in various applications.

REFERENCES

1. Lars Lindner, Paolo Mercorelli; “Machine Vision System for Unmanned Aerial Vehicle Navigation”; IEEE 26th International Symposium on Industrial Electronics (ISIE); 19-21 June 2017.

2. Djoko Purwanto, Dani Prasetyawan, Muhammad Rivai; “Development of Auto Tracking and Target Locking on Static Defence Based on Machine Vision”; IEEE Application for Technology of Information and Communication; March 9, 2017.

3. Yuichi Motai, Sumit Kumar Jha, Daniel Kruse; “Human tracking from a mobile agent: Optical flow and Kalmanfilter arbitration”; Image Communication; Aug 7, 2011.

4. Imran S. Sarwar, Afzaal M. Malik; “Modeling, analysis and motion control of a Pan Tilt Platform based on linear and nonlinear systems”; WSEAS Transactions on Systems and Control. Volume 4, Issue 8; August 2009.

5. Cheng-Ming Huang, Su-Chiun Wang, Li-Chen Fu, Pei-Ying Chen, dan Yu-Shan Cheng, “A Robust Visual Tracking of an Arbitrary-Shaped Object by a New Active Contour Method for a Virtual Reality Application”, Proceedings of lEEE International Conference on Networking, Sensing & Control , 2004, pp. 64-69.

6. Tingting Wang, Guodong Liu, and Wenfang Xie, “Visual Servoing Control of Video Tracking Sistem for Tracking a Flying Target”, IEEE / ASME International Conference on Advanced Intelligent Mechatronics, 2011, pp. 850 – 855.

7. Purwanto, D., Prasetyawan, D. and Rivai, M., 2016, August. Development of auto tracking and target locking on static defence based on machine vision. In 2016 International Seminar on Application for Technology of Information and Communication (ISemantic) (pp. 290-294). IEEE.

8. Müller, M., Casser, V., Lahoud, J., Smith, N. and Ghanem, B., 2018. Sim4cv: A photo-realistic simulator for computer vision applications. International Journal of Computer Vision, 126, pp.902-919.

9. Shailendra Kumar Singh, Utkarsh Sharma; “Simulink Model For Object Tracking using Optical Flow”; International Journal of Science and Research (IJSR). Volume 4, Issue 6; June 17, 2015.

10. Tingting Wang, Guodong Liu, and Wenfang Xie, “Visual Servoing Control of Video Tracking Sistem for Tracking a Flying Target”, IEEE / ASME International Conference on Advanced Intelligent Mechatronics, 2011, pp. 850 – 855.

CONTACT INFORMATION

1) Name: Abdul Moaiz
   ID: F19603002
   Mobile No: +92 335 5807259

2) Name: Hamdan Bin Tanveer
   ID: F19603015
   Mobile No: +92 319 0972447

3) Name: Abdur Rehman
   ID: F19603020
   Mobile No: +92 335 5807259

4) Name: Muhammad Raheem
   ID: F19603032
   Mobile No: +92 345 7279551

Supervisor:

Name: Dr. Adnan Saeed