

SMART SURVEILLANCE SYSTEM FOR INTRUSION DETECTION

A PROJECT REPORT

Submitted by

JAYANTH KRISHNAMOORTHY	160801034
KARTHI KUMAR	160801039
KISHORE S	160801042

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

INFORMATION TECHNOLOGY



SRI VENKATESWARA COLLEGE OF ENGINEERING

(An Autonomous Institution)

SRIPERUMBUDUR

NOVEMBER 2019

SRI VENKATESWARA COLLEGE OF ENGINEERING

(An Autonomous Institution)

BONAFIDE CERTIFICATE

Certified that this project report “SMART SURVEILLANCE SYSTEM FOR INTRUSION DETECTION” is the bonafide work of **JAYANTH KRISHNAMOORTHY (160801034), KARTHI KUMAR R (160801039) and KISHORE S (160801042)** who carried out the project work under my supervision.

SIGNATURE

Dr. V. Vidhya, M.E, Ph.D.,

HEAD OF THE DEPARTMENT

Department of Information Technology

Sri Venkateswara College of Engineering

Sriperumbudur Tk. - 602 117

SIGNATURE

Mr. AR. Guru Gokul, M.E.,

SUPERVISOR

Assistant Professor

Department of Information Technology

Sri Venkateswara College of Engineering

Sriperumbudur Tk. - 602 117

Submitted for the Project Viva-Voice held on at
Sri Venkateswara College of Engineering, Sriperumbudur.

INTERNAL EXAMINER

EXTERNAL EXAMINER

ABSTRACT

Video Surveillance has been used in many applications including elderly care and home nursing etc. There are situations in which ordinary video surveillance systems are incapable of preventing intrusion. This happens predominantly in areas close to protected areas that are often prone to animal intrusion. Smart video surveillance systems are capable of enhancing situational awareness across multiple scales of space and time. It describes mobile based remote control and surveillance architecture. Objective of this project is to develop a Smart Surveillance System using Computer Vision, which detects the animal intrusion in estates near protected areas through the CCTV (Closed Circuit Television) cameras deployed and alerts the admin. The proposed model makes use of some library to capture camera images and detect intrusion using image comparison technique. Once the comparison is done and an intrusion is found, it sends the streamed video from server to remote administrator over android phone. Admin can then take appropriate action and alert local security. Smart Surveillance is the use of automatic video analytics to enhance effectiveness of surveillance systems. The user can view the particular video. This system maintains the security situation at estates and this reduces the incidence of animal intrusion cases and avoid life loss (Both domestic animal and humans).

ACKNOWLEDGEMENT

We thank our principal **Dr. Ganesh Vaidyanathan, Ph.D.**, Principal, Sri Venkateswara College of Engineering (Autonomous) for being the source of inspiration throughout our study in this college.

We express our sincere thanks to **Dr. V. Vidhya, M.E., PhD**, Head of Department, Information Technology for her permission and encouragement accorded to carry out this project.

With profound respect, we express our deep sense of gratitude and sincere thanks to our guide, **Mr. A R. Guru Gokul, M.E., Assistant Professor**, for his continuous and valuable guidance throughout this project.

We are also thankful to **Ms. N. Devi, M.E., Mr. V. Rajaram, M.E., Mr. V. Ranjith, M.E.**, project coordinators for their continual support and assistance.

We are also thankful to all faculty members and supporting staff of the Department of Information Technology, Sri Venkateswara College of Engineering (Autonomous) for rendering their support.

JAYANTH KRISHNAMOORTHY
KARTHI KUMAR R
KISHORE S

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	iii
	LIST OF TABLES	vii
	LIST OF FIGURES	viii
	LIST OF ABBREVIATIONS	ix
1	INTRODUCTION	1
	1.1 COMPUTER VISION	1
	1.1.1 NEURAL NETWORKS	2
	1.1.2 APPLICATIONS OF COMPUTER VISION	3
	1.2 OBJECTIVE OF THE PROJECT	4
	1.3 ORGANIZATION OF THE REPORT	6
2	LITERATURE SURVEY	7
	2.1 AGRICULTURAL INTRUSION DETECTION USING WIRELESS SENSOR NETWORK	7
	2.2 IOT-BASED WILD ANIMAL INTRUSION DETECTION SYSTEM	8
	2.3 FARM MONITORING AND SECURITY SYSTEM	8
	2.4 PHOTSENSITIVE SECURITY SYSTEM FOR THEFT DETECTION AND CONTROL USING GSM TECHNOLOGY	9
	2.5 MOTION BASED ANIMAL DETECTION IN AERIAL VIDEOS	10
	2.6 AN ANIMAL DETECTION PIPELINE FOR	11

	IDENTIFICATION	
	2.7 AUTOMATIC WILD ANIMAL MONITORING: IDENTIFICATION OF ANIMAL SPECIES IN CAMERA-TRAP IMAGES USING VERY DEEP CONVOLUTIONAL NEURAL NETWORKS	11
	2.8 AGREEMENT BETWEEN PASSIVE INFRARED DETECTOR MEASUREMENTS AND HUMAN OBSERVATIONS OF ANIMAL ACTIVITY	12
	2.9 CLASSIFICATION OF WILD ANIMALS BASED ON SVM AND LOCAL DESCRIPTORS	13
	2.10 AUTOMATIC WILD ANIMAL DETECTION IN LOW QUALITY CAMERA-TRAP IMAGES USING TWO-CHANNELLED PERCEIVING RESIDUAL PYRAMID NETWORKS	14
3	ARCHITECTURE OF PROPOSED SYSTEM	15
	3.1 PROPOSED ARCHITECTURE	15
	3.2 SYSTEM	16
	3.2.1 DATA SET	16
	3.2.2 OBJECT ANNOTATION AND TRAINING SYSTEM	18
	3.2.3 OBJECT DETECTION SYSTEM	19
	3.2.4 MOBILE APPLICATION	20
	3.2.5 DATABASE	20
	3.3 PROCESSING	20
	3.3.1 INPUT DATA	20

	3.3.2 OUTPUT	21
4	DATA EXTRACTION AND LABELLING	22
	4.1 DATA EXTRACTION FOR DATASET	22
	4.2 ANNOTATION AND LABELLING	23
	4.3 DARKNET TRAINING	24
5	OBJECT RECOGNITION	26
	5.1 LOSS FUNCTIONS EXPLANATION	26
	5.2 THE PREDICTIONS VECTOR	27
	5.3 INTERNAL WORKING OF YOLO	28
6	IMPLEMENTATION AND EXPERIMENTAL RESULTS	30
7	CONCLUSION AND FUTURE WORK	33
	7.1 CONCLUSION	33
	7.2 FUTURE WORK	33
	REFERENCE	34

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	Proposed System Architecture	17
4.1	Frame Extraction	17
4.2	Annotator Tool LableImg	23
4.3	Annotator Tool Output	24
5.1	Detection Of Objects Using Yolo	27
5.2	Yolo v3 Network Architecture	29
6.1	Training Dataset Using Darknet	30
6.2	Single Elephant Intrusion Detection Using Yolo	31
6.3	Multiple Elephant Intrusion Detection	31
6.4	Tiger Intrusion Detection	32

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
AOI	Annotation Of Intererst
API	Application Programming Interface
AVR	Advanced Virtual RISC
CCTV	Closed Circuit Televison
CUDA	Compute Unified Device Architecture
CNN	Convolutional Neural Network
GPS	Global Positioning System
GSM	Global System for Mobile
GPU	Graphical Processing Unit
LDR	Light Dependent Resistor
PIR	Passive Infrared Sensor
RCNN	Recurrent Convolutional Neural Network
RNN	Recurrent Neural Network
ROI	Region of Interest
SIFT	Scale Invariant Feature Transform
SMS	Short Message Service
SURF	Speeded up Robust Features
SSE	Sum of Square Error
SIFT	Support Vector Machine
XML	Extensible Markup Language
YOLO	You Look Only Once

CHAPTER 1

INTRODUCTION

This chapter gives a detailed introduction about the domain and also discusses the various existing solutions available along with their disadvantages. It also emphasises and highlights the need for the proposed system.

1.1 COMPUTER VISION

Computer Vision is the process of using machines to understand and analyse imagery (both photos and videos). It deals with how computers can be made to gain high-level understanding from digital images or videos. From the perspective of engineering, it seeks to automate tasks that the human visual system can do. Computer vision is concerned with the automatic extraction, analysis and understanding of useful information from a single image or a sequence of images. It involves the development of a theoretical and algorithmic basis to achieve automatic visual understanding.

While most of the Computer Vision algorithms have been around in various forms for very long time, recent advances in Machine Learning, as well as huge leaps forward in data storage, computing capabilities, and cheap high-quality input devices, have driven major improvements in how well our software can explore this kind of content.

Computer vision is currently one of the areas in Machine Learning where core concepts are already being integrated into major products that we use every day. Google is using maps to leverage their image data and identify street names, businesses, and office buildings. Facebook is using computer vision to identify people in photos, and do a number of things with that information.

1.1.1 NEURAL NETWORKS

Neural networks are a set of algorithms, modelled loosely after the human brain, that are designed to recognize patterns. They interpret sensory data through a kind of machine perception, labeling or clustering raw input. Neural networks help us cluster and classify. Neural Networks can be assumed as a clustering and classification layer on top of the data you store and manage. They help to group unlabelled data according to similarities among the example inputs, and they classify data when they have a labelled dataset to train on. Neural networks can also extract features that are fed to other algorithms for clustering and classification. Few major types of Neural Networks are:

1. Artificial Neural Network (ANN): An artificial neuron network (ANN) is a computational model based on the structure and functions of biological neural networks. Information that flows through the network affects the structure of the ANN because a neural network changes - or learns, in a sense - based on that input and output. ANNs are considered nonlinear statistical data modelling tools where the complex relationships between inputs and outputs are modelled or patterns are found.

2. Convolutional Neural Network (CNN): A Convolutional Neural Network (ConvNet/CNN) is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other. The pre-processing required in a CNN is much lower as compared to other classification algorithms. While in primitive methods filters are hand-engineered, with enough training, CNN have the ability to learn these filters/characteristics.

3. Recurrent Neural Network (RNN): A recurrent neural network (RNN) is a class of artificial neural networks where connections between nodes form a directed graph along a temporal sequence. This allows it to exhibit temporal dynamic behaviour. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. This makes them applicable to tasks such as unsegmented, connected handwriting recognition or speech recognition.

1.1.2 APPLICATIONS OF COMPUTER VISION

Object Detection: Object detection is the task of image classification with localization, although an image may contain multiple objects that require localization and classification. This is a more challenging task than simple image classification or image classification with localization, as often there are multiple objects in the image of different types. Often, techniques developed for image classification with localization are used and demonstrated for object detection.

Optical Character Recognition: An AI system can be trained through Computer Vision to identify and read text from images and images of documents, and use it for faster processing, filtering, and on-boarding.

Facial recognition: It is a category of biometric software that maps an individual's facial features mathematically and stores the data as a faceprint. The software uses Computer Vision to compare a live capture or digital image to the stored faceprint in order to verify an individual's identity.

Self-driving Cars: Computer Vision is the fundamental technology behind developing autonomous vehicles. Most leading car manufacturers in the world are reaping the benefits of investing in artificial intelligence for developing on-road versions of hands-free technology.

Augmented & Virtual Reality: Computer Vision is central to creating limitless fantasy worlds within physical boundaries and augmenting our senses. Most of the Augmented Reality and Virtual Reality fantasies would be impossible without Computer Vision.

1.2 OBJECTIVE OF THE PROJECT

Agriculture is the backbone of our country. Plantation and Vegetation is mainly done in almost all parts of our country. They are predominantly done in places near protected areas where fertility is high. They not only yield crops but also provide employability for people in the locality. They yield highly valuable products because of their fertile nature and hence they yearn high profits to the landlords of the locality. Protected areas are clearly defined geographical space,

recognised, dedicated and managed, through legal or other effective means, to achieve the long term *conservation* of nature with associated ecosystem services and cultural values. The locality near protected areas are often prone to wild animal attacks which cause damage to crops in the vegetation around. There are many places in which crops are damaged due to wild animal attacks and it has also led to loss of life for some people. Several private owned estates and have CCTV cameras for surveillance. Idea behind the proposed system is to deploy a smart surveillance system in which the CCTV cameras monitoring the area can intimate the securities if any intrusion of animals is detected.

Intrusion detection is a smart surveillance system that can detect the intrusion of any object based on the training through the neural network. It can also be trained with neural network that can detect both animals and human. Thus it can be used to detect the intrusion of any unauthorised person into a private owned prohibited areas. Video surveillance has become most basic component in private owned areas. Since the proposed system can also detect the intrusion of unauthorised person it can be deployed in the private areas. Thus the proposed system reduces from full time monitoring of surveillance system to monitoring only when an intrusion is detected. The proposed system uses Open Computer Vision and Neural Network to detect the intrusion through CCTV footages. The proposed system attempts to replace the existing smart surveillance systems that uses sensors and provide better performance than the previous one.

1.3 ORGANIZATION OF THE REPORT

The report is organized as follows: Chapter 2 presents Literature Survey. Chapter 3 discusses the Software Requirement specifications. Chapter 4 discusses the architecture of the proposed work. Chapter 5 presents the modules of the

proposed work. Chapter 6 presents the implementation and experimental results. Chapter 7 specifies Conclusion and Future work to be done.

CHAPTER 2

LITERATURE SURVEY

This chapter presents a detailed Literature Survey of all the existing work carried so far by various authors along with the pros and cons of each.

[1] A prototype for Agricultural Intrusion Detection using wireless sensor network

Sanku Kumar Roy and Arijit Roy (2015) proposed a prototype “Agricultural Intrusion Detection using wireless sensor network” to detect the intrusion of animals into farming lands using Advanced Virtual RISC (AVR) micro-controller-based wireless sensor boards over an outdoor environment and evaluate the performance. This system helps to generate alarms in the farmer's house and at the same time transmits a text message to the farmer's cell phone when an intruder enters into the field. The sensors in the field and the alarm system are connected with each other and are in same location.

Merits:

- It uses AVR micro controller and intimates the intrusion immediately.
- Sensors provides accurate results and works fine all time.

Demerits:

- Using sensors increases cost implementation.
- Sensors can detect only short range and hence for covering large areas many sensors are required.

[2] IOT-based Wild Animal Intrusion Detection System

Prajna P, Soujanya B.S, and Divya M (2018) proposed a “IoT- based Wild Animal Detection System” in International Journal for Engineering and Technology. This system is an IoT based solution in which as soon as intrusion is detected the image of the camera from camera is obtained and it is compared with the classified image datasets and then intimated to the field owner. Since it is an IoT based solution it can intimate the field owner even if the field area is at certain distance from the owners location.

Merits:

- This system can intimate the field owner even at any distant from the field.
- Provides accurate real time data.

Demerits:

- It can intimate the users only after the animal enter the field and thus crop loss can be reduced and it is not suitable for preventing.
- For better comparisons at night time, night vision cameras are required.

[3] Farm Monitoring and Security System

Emmanuel Onwuka Ibam and Mark O Afolabi (2018) proposed “Farm Monitoring and Security System”. This system was proposed to detect any kind of human intrusion to the farm fields. This system is deployed using Wireless Network Sensors (WNS) and GSM module to intimate the field owner on detection of any intrusion. The architecture of the WSNs system comprises of

a set of sensor nodes, surveillance facilities and a base station that communicate with each other and gather information to make decisions about the situation at hand. The system overcomes the limitation of building fences using sticks which can be very much stressful. When the intruder stays for more than 30 seconds on the farmland, the GSM module is used for sending SMS to the farm owner indicating the nature of intrusion.

Merits:

- The proposed system uses wireless sensors and it yields accurate results.
- This system has large capability, wide area range, low operation costs, effective and strong expandability.

Demerits:

- This WNS System can intimate only there is some intrusion in the field and it cannot describe the nature of the intrusion.
- Usage of sensors requires that it has to be maintained regularly.

[4] Photosensitive security system for theft detection and control using GSM technology

Kushal Voona, Satya Ravi Teja and Sai Srikar (2015) proposed "Photosensitive security system for theft detection and control using GSM technology" which uses photosensitive sensor LDR (Light Dependent Resistor) based sensor which acts as an electronic eye for detecting the theft or attempt, and a signaling procedure based on SMS using GSM (Global Systems for Mobile communications) technology. The GSM based communication helps the owner

and concerned authorities to take necessary and timely action in order to prevent the theft. The LDR circuit is interfaced using a relay circuit with an Arduino microcontroller board. Efficacy of the proposed system can be seen in its immediate intimation regarding the incident.

Merits:

- This is very useful in jewellery shops and museum like places because of its effectiveness.

Demerits:

- This system is very costly to implement and sensor inaccuracy may lead to false intimation.
- This system cannot be used to detect the intrusion nature.

[5] Motion Based Animal Detection in Aerial Videos

Fang Y (2010), “Motion Based Animal Detection in Aerial Videos” discussed a technique to move animal detection by taking benefit of global patterns of pixel motion. In the dataset, where animals make obvious movement against the background, motion vectors of every pixel were estimated by applying optical flow techniques. A coarse segmentation then eliminates most parts of the background via applying a pixel velocity threshold. Using the segmented regions, another threshold was used to filter out negative candidates, which could belong to the background

Merits:

- Global pixel motion difference between the animal and the background is used to detect the animals or objects.
- Highly efficient system.

Demerits:

- In this approach, more effective local threshold selection methods are not used.

[6] An Animal Detection Pipeline for Identification

Parham J (2014), “An Animal Detection Pipeline for Identification” proposed a 5-component detection pipeline to utilize in a computer vision-based animal recognition system. The result of this approach was a collection of novel annotations of interest (AoI) with species and viewpoint labels. The concept of this approach was to increase the reliability and automation of animal censusing studies and to offer better ecological information to conservationists.

Merits:

- This approach is provided better ecological information to conservationists.

Demerits:

- Practically identification of animals using pipeline provides low accuracy.

[7] Automatic Wild Animal Monitoring: Identification of Animal Species in Camera-trap Images using Very Deep Convolutional Neural Networks

Villa A. G. (2017), “Automatic Wild Animal Monitoring: Identification of Animal Species in Camera-trap Images using Very Deep Convolutional Neural Networks” stated the main problems inherent to camera trapping images automatic species identification. Through numerous experiments the capacity of very deep convolutional neural networks for automatizing species classification in cameratrap images was proved. Unbalanced, balanced, foreground objects selection and segmented versions of Snapshot Serengeti dataset were utilized for studying how a powerful learning algorithm performs in presence of four of the main problems inherent to camera trapping acquired data: unbalanced samples, empty frames, incomplete animal images and objects too far from focal distance.

Merits:

- High performance in detection of animals from the classified set.
- Reasonable detection accuracy.

Demerits:

- Provides low accuracy in low light illumination.
- It cannot detect the animals during night period.

[8] Agreement between passive infrared detector measurements and human observations of animal activity

Besteiro R. (2010), “Agreement between passive infrared detector measurements and human observations of animal activity” evaluated the agreement among two animal activity measurement techniques: a Passive infrared (PIR) detector that was used the sensor’s digital signal for executing measurements and human observations of activity in a group of 50 weaned piglets on a commercial farm.

The location chosen for the sensor allowed for the recording of the main transverse movements with respect to the orientation of the sensor that maximized its detection capacity. Human observation revealed two kinds of behavioral activity, feeding (eating or drinking) and playing. Additionally, animal weight affected the quality of measurements that decreased with the increase in the ratio among kg of live weight and area covered by the sensor. The kind of activity affected the precision of PIR detectors that better detected playing activities that were more intense than feeding activities.

Merits:

- Good estimation of the relative activity of the group.
- Acceptable and accountable technique performance is obtained.

Demerits:

- Less intense for feeding activities.
- The run time of this algorithm is not yet satisfactory.

[9] Classification of Wild Animals Based on SVM and Local Descriptors

Matuska S. (2011), “Classification of Wild Animals Based on SVM and Local Descriptors” discussed a new approach for object recognition by using hybrid local descriptors. This approach was utilized a combination of a few techniques (SIFT - Scale-invariant feature transform, SURF - Speeded Up Robust Features) and consists of second parts. The applicability of the presented hybrid techniques were demonstrated on a few images from dataset. Dataset classes represent big animals situated in Slovak country, namely wolf, fox, brown bear, deer and wild boar.

Merits:

- Promising results comparable with other key point detectors.
- Highly accurate results are obtained.

Demerits:

- Poor results with success rate of classification around 50% only.

[10] Automatic Wild Animal Detection in Low Quality Camera-trap Images Using Two-channelled Perceiving Residual Pyramid Networks

Zhu C. and Rey N. (2013), “Automatic Wild Animal Detection in Low Quality Camera-trap Images Using Two-channelled Perceiving Residual Pyramid Networks” introduced a two-channelled perceiving residual pyramid networks towards automatic wild animal detection in low quality camera-trap images. This paper was extracted depth cue from the original images and used two-channelled perceiving model as input to training a networks. The three-layer residual blocks were used for merging the entire information and generating full size detection results. In addition, a novel high quality dataset with the complex wild environment was built using dataset design principles.

Merits:

- Improves the quality of wild animal detection and more robustness.

Demerits:

- This approach does not perform high speed applications.

The proposed system discussed in chapter 4 overcomes all the above-mentioned disadvantages. It certainly modernizes the current system.

CHAPTER 3

ARCHITECTURE OF PROPOSED SYSTEM

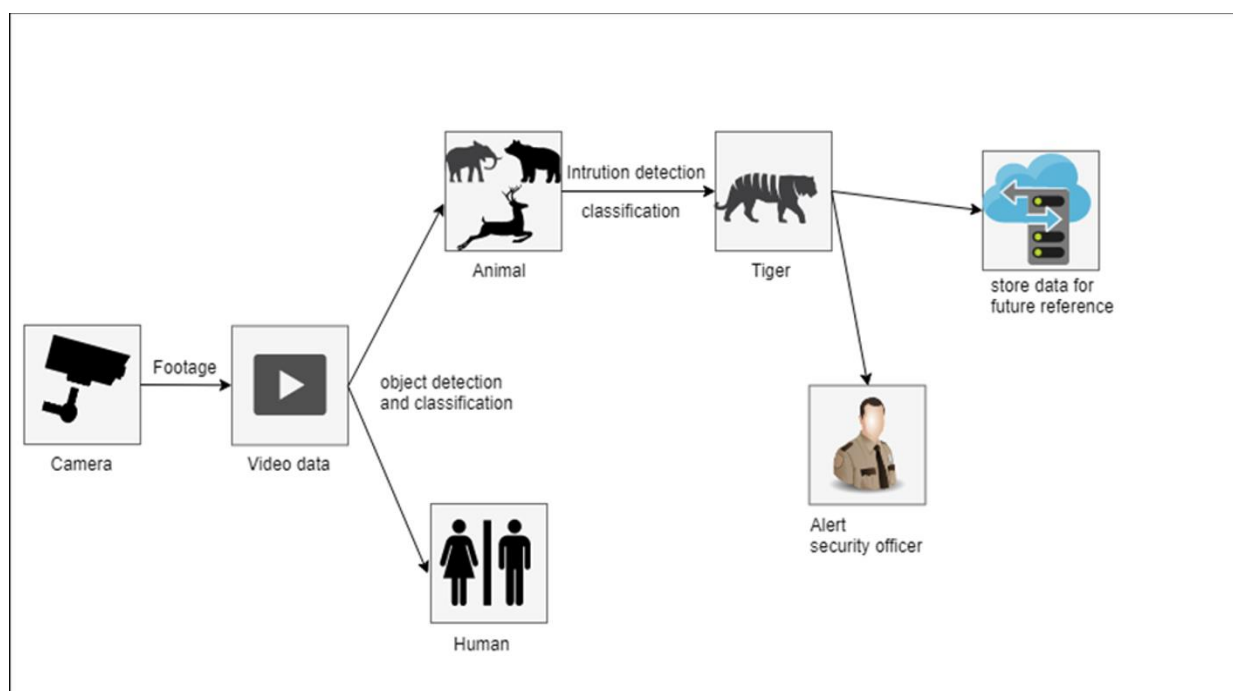


FIGURE 3.1 PROPOSED SYSTEM ARCHITECTURE

3.1 PROPOSED ARCHITECTURE

The figure 3.1 demonstrates the proposed architecture. The proposed system uses Convolution Neural Network (CNN) to detect animal intrusion through camera videos. Firstly, video is captured from the camera fixed intervals. From the live video footage the images are extracted so that it can be used as a dataset for training. The image is pre-processed in order to achieve better accuracy. Later, the image is processed with sliding window technique in order to identify the animal in the image. The System is trained with the help of large number of data sets to detect the animals. The images are annotated with required Region of

Interest (ROI). The most widely used annotation tool for YOLO training is BBox-Label-Tool (Bounding Box). It is also easier to install and simpler to use than other alternatives. Label-box is a data labelling platform for enterprises to easily train expert machine learning applications. It is agnostic to data type and has an open source labelling frontend with already built interfaces for image classification and segmentation, text, audio and video annotation. Label-box supports custom built labelling interfaces using a java script API (labeling-api.js). The annotated images are the input for training through the use of YOLO (You Only Look Once) framework. The items are classified during training itself. The real time video data from the camera is sent to the system where the frames from the video is extracted at regular intervals and the object detection algorithm is performed on each and every frame. Object detection is a field of Computer Vision that detects instances of semantics objects in images/videos (by creating bounding boxes around them in our cases). On detecting the intrusion, a signal is sent to the cloud, from which the user or security officer is alerted.

3.2 SYSTEM

The system has five sub components

- Dataset
- Object Annotation and Training
- Object Detection System
- Mobile Application
- Database

3.2.1 DATA SET

As with any deep learning task, the first most important task is to prepare the

dataset. A dataset is a collection of instances that can be used by the proposed system for both testing and training and when working with machine learning methods the proposed system typically need a few datasets for different purposes. Especially in Object Detection, the datasets are images of the required objects. The proposed system detects the intrusion of animals from the video surveillance camera. The appropriate datasets for the proposed system are the images of the animals that are prone to intrude the field more frequently. The image dataset of animals can be collected in different ways. The images can be either directly downloaded or they can be extracted from the video of the animals. Even some of the datasets are provided by organisations such as Kaggle. The dataset has to be selected carefully so that the images from the dataset is sufficient to train the object detector. A weight file that is trained using good dataset can detect the animals even in bad quality. The proposed System will use the Tiger and Elephant images from Google's OpenImagesV4 dataset that are publicly available online. It is a very big dataset with around 600 different images. As a whole, the dataset is more than 50GB, download the images with clear visibility of Tiger and elephants only.

TRAIN-TEST SPLIT

Any machine learning training procedure involves first splitting the data **randomly** into two sets.

1. **Training set:** This is the part of the data that are fed into our machine learning algorithm on which are used to train the model. The efficiency and accuracy of the proposed system is directly proportional to the training dataset. More the images used to train the proposed system, more the efficiency and accuracy of the proposed system. The quality of the images is also a factor determining the proposed system efficiency. A good dataset contains the animal

images in high quality and in different postures. Depending on the amount of data you have, you can randomly select between 70% to 90% of the data for training.

2. **Test set:** This is the part of the data on which are used to test our model. A dataset that are used to validate the accuracy of our model but is not used to train the model. Typically, this is 10-30% of the dataset. No image should be part of the both the training and the test set. This part of dataset should contain the images in different quality and in different postures so that they can be used in calculating the accuracy of the trained weight file. Split the images inside the dataset folder into the train and test sets.

3.2.2 OBJECT ANNOTATION AND TRAINING SYSTEM

Annotation in machine learning is the process of labelling the data which is in the form of image, video, text etc. This is done manually by the labelling the objects region using bounding box. The region that is covered by bounding box is called ROI (Region Of Interest). The tool used as image annotator is Labellmg in which the required region along with the name of the object is marked. Annotating process generates a text file for each image, contains the object class number and coordination for each object in it, as this format "(object-id) (x-centre) (y-centre) (width) (height)" in each line for each object. Co-ordinations values (x, y, width, and height) are relative to the width and the height of the image. Create a folder contains images files and name it "images" and annotations files and name it as "labels". Both folders must be in the same directory. The system also **Transfer Learning** enabling reusing an efficient pre-trained model.

Darknet is an open source neural network framework supports CPU and GPU computation. Before starting the training process, create a folder "**custom**" in the main directory of the darknet. After that, start training via executing this command from the terminal " darknet detector train custom/trainer.data custom/yolov3-tiny.cfg darknet53.conv.74 ". Weights will be saved in the backup

folder every 100 iterations till 900 and then every 10000. The training is accomplished by running the iterations on a 2GB nvidia 940mx GPU with CUDA enabled.

3.2.3 OBJECT DETECTION SYSTEM

Object detection is a computer vision technique for locating instances of objects in images or videos. Object detection algorithms typically leverage [machine learning](#) or [deep learning](#) to produce meaningful results. Popular deep learning-based approaches using [convolutional neural networks](#) (CNNs), such as R-CNN and YOLO v2, automatically learn to detect objects within images. There are many more algorithms in use and every algorithm has its pros and cons. The model here is the **You Only Look Once** (YOLO) algorithm that runs through a variation of an extremely complex Convolutional Neural Network architecture. YOLO is mostly used in detecting objects in real time. Prior detection systems repurpose classifiers or localizers to perform detection. They apply the model to an image at multiple locations and scales. High scoring regions of the image are the objects. A more enhanced and complex YOLO v3 model is being used here. Also, the python **cv2** package has a method to setup Darknet from our configurations in the yolov3.cfg file. The path to the weight file path, config file path, and the label path are specified in the code. The video is extracted into frames and the objects are detected in the frames.

YOLO is not a traditional classifier that is repurposed to be an object detector. YOLO actually looks at the image just once, hence its name You Only Look Once, but in a clever way. YOLO divides up the image into a grid of cells. Each of these cells is responsible for predicting bounding boxes. A bounding box describes the rectangle that encloses an object. YOLO also outputs a confidence score that tells us how certain it is that the predicted bounding box actually

encloses some object. This score doesn't say anything about what kind of object is in the box, just if the shape of the box is any good. The confidence score for the bounding box and the class prediction are combined into one final score that tells us the probability that this bounding box contains a specific type of object. The detected region is enclosed by rectangle with the help of coordinates provided by the YOLO output and methods by some cv2 modules.

3.2.4 MOBILE APPLICATION

A mobile application is an android application built in android studio. This mobile application is used to send alert message to the user when an intrusion is detected by the system. This application uses publish-subscribe model to intimate the intrusion detection.

3.2.5 DATABASE

The database has a repository of all reports of animals that intruded and its images, these are non-volatile and stays forever unless until the user himself wishes to wash it off from the system memory, these data can be fetched by the user from the app after proper authentication.

3.3 PROCESSING

The required input and the processed output of the proposed architecture is discussed in the following sections.

3.3.1 INPUT DATA

The extracted images are the input to the annotator tool. The images are annotated by the annotator tool and annotated images and text file corresponding to the images are input for training the model. The video captured by the surveillance camera is the input to the trained model. The model extracts the frames from the video and perform object detection to find the objects. In python, cv2 package

provides method to capture video data from the camera in real time. It also provide methods to extract images form the data. The extracted images are the required input.

3.3.2 OUTPUT

Once a animal intrusion is detected by the system it creates a alert to the user mobile application updates it to the database, overlay the boxes on the objects detected and return the stream of frames as a video playback. The system captures the frame when the intrusion is detected with the time stamp it saves it in database for future reference.

DATA EXTRACTION AND LABELLING

The required data for the proposed system is the images of the animals for which the model is to be trained. Although the images can be downloaded from the internet, they can't help to generate efficient weight file since it is difficult for downloading the images of the animals in different postures. Thus, to get efficient dataset the images are obtained through video. The idea of using extracted frames as required dataset images helps in obtaining the user with different postures of the animals in different scales. Thus, for efficiency of the dataset, the images extracted from different videos of different qualities and used for training the weight file. The training data determines the efficiency and accuracy of the data. Python provides methods to read and write image frames under cv2 package.



The Figure 4.1 represents the images extracted. Multiple clarity images in multiple postures is obtained through this extraction. These images might look similar but since its extracted from video it contains minute chnages which will help the model to get trained more accurate.

4.2 ANNOTATION AND LABELLING

The images thus obatained are annotated with the required objects. The area of annotation is called Region of Interest (ROI). The process of labelling the data which is in the form of image is done manually by the labelling the objects region using bounding box. This manual labelling process is called annotation. In the proposed system annotation is done using labellmg. The Figure 4.2 represents the image annotator with annotated object.

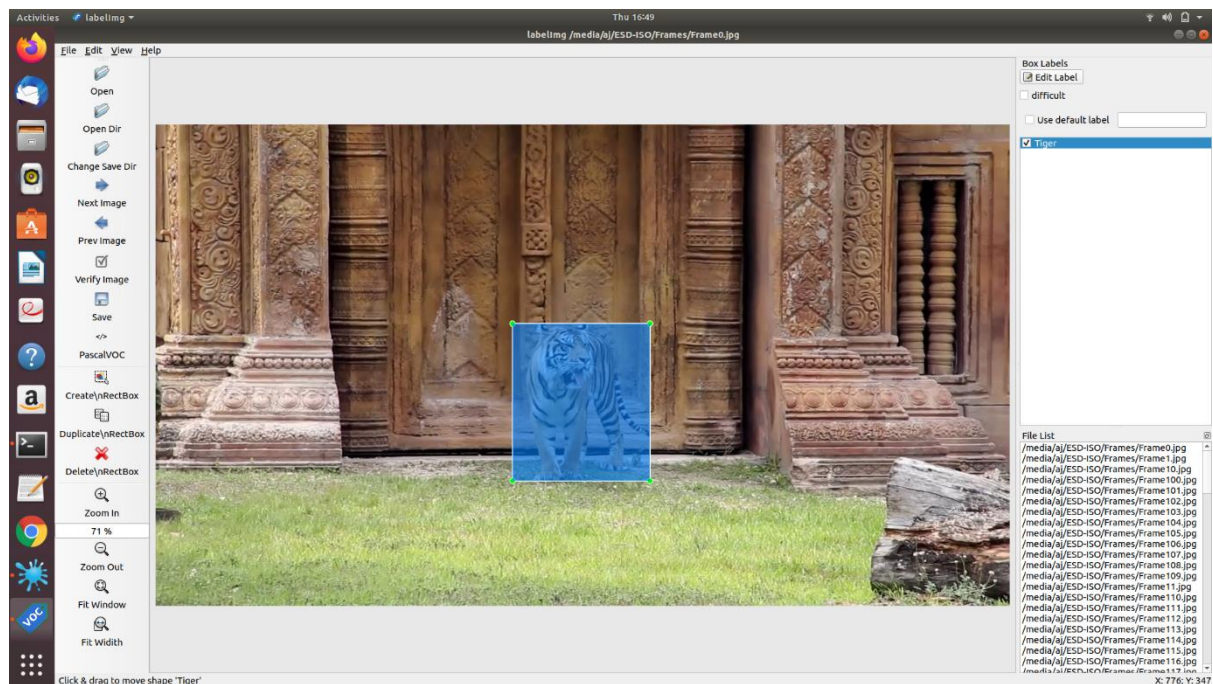
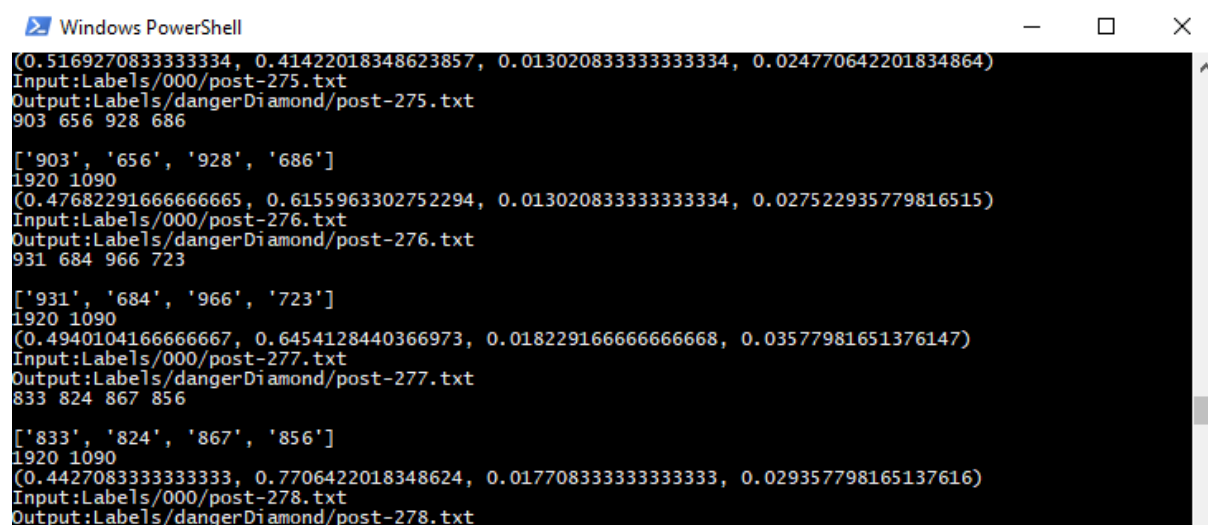


FIGURE 4.2 ANNOTATOR TOOL - LABLEIMG

LabelImg is a graphical image annotation tool and label object bounding boxes in images. It is written in Python and uses Qt for its graphical interface. The main purpose of using this annotation technique is reduce the range of search for those

object features conserving the resources used in computing and helps to solve the computer vision problems. Annotation through bounding box produce the coordinates of the location of an object with respect to the image and stores the values as text file. LableImg provides us facility of specifying the annotated objects under a class name. Labellmg facilitates us by storing the annotated image file and corresponding text file for the image in separate directories. The text files contain the values of x_center, y_center, width, height in the mentioned sequence. Annotations are saved as XML files in PASCAL VOC format, the format used by [ImageNet](#), also supports YOLO format. Annotated images are saved as text file where the ROI values are calculated and stored in numbered tuple format. This text files are later user for the training purpose in the Darknet to generate the weight file. The Figure 4.3 represents the annotator output.



```

Windows PowerShell
(0.5169270833333334, 0.41422018348623857, 0.013020833333333334, 0.024770642201834864)
Input:Labels\000/post-275.txt
Output:Labels/dangerDiamond/post-275.txt
903 656 928 686

['903', '656', '928', '686']
1920 1090
(0.47682291666666665, 0.6155963302752294, 0.013020833333333334, 0.027522935779816515)
Input:Labels\000/post-276.txt
Output:Labels/dangerDiamond/post-276.txt
931 684 966 723

['931', '684', '966', '723']
1920 1090
(0.49401041666666667, 0.6454128440366973, 0.018229166666666668, 0.03577981651376147)
Input:Labels\000/post-277.txt
Output:Labels/dangerDiamond/post-277.txt
833 824 867 856

['833', '824', '867', '856']
1920 1090
(0.4427083333333333, 0.7706422018348624, 0.017708333333333333, 0.029357798165137616)
Input:Labels\000/post-278.txt
Output:Labels/dangerDiamond/post-278.txt

```

FIGURE 4.3 ANNOTATOR TOOL OUTPUT

4.3 DARKNET TRAINING

The annotated images are fed into YOLOv3 which uses a neural network framework called darknet for training the object detector. When huge quantity of such annotated images are used to train an AI model through computer vision the

model give the predictions learn from these annotated images. This kind of training through images is called supervised learning, trying to learn through trial and error, constantly trying to predict the best outcome. Darknet is an open source neural network framework supports CPU and GPU computation. After collecting and annotating dataset, have two folders in the same directory the "images" folder and the "labels" folder. Now, split dataset to train and test sets by providing two text files, one contains the paths to the images for the training set (train.txt) and the other for the test set (test.txt). This can be done using the following script after editing the **dataset_path** variable to the location of your dataset folder. Then modify the YOLOv3 tiny model ([yolov3-tiny.cfg](#)) to train our custom detector.

CHAPTER 5

OBJECT RECOGNITION

Object recognition refers to a collection of related tasks for identifying objects in digital photographs. Region-Based Convolutional Neural Networks, or R-CNNs, are a family of techniques for addressing object localization and recognition tasks, designed for model performance. You Only Look Once, or YOLO, is a second family of techniques for object recognition designed for speed and real-time use.

5.1 LOSS FUNCTIONS EXPLANATION

There are **5 terms in the loss function** as shown above.

1. **1st term (x, y):** The bounding box x and y coordinates is parametrized to be offsets of a particular grid cell location so they are also bounded between 0 and 1. And the sum of square error (SSE) is estimated only when there is object.
2. **2nd term (w, h):** The bounding box width and height are normalized by the image width and height so that they fall between 0 and 1. SSE is estimated only when there is object. Since small deviations in large boxes matter less than in small boxes, square root of the bounding box width w and height h instead of the width and height directly to partially address this problem.
3. **3rd term and 4th term (The confidence)** (i.e. the IOU between the predicted box and any ground truth box): In every image many grid cells do not contain any object. This pushes the “confidence” scores of those cells towards zero, often overpowering the gradient from cells that do contain objects, and

makes the model unstable. Thus, the loss from confidence predictions for boxes that don't contain objects, is decreased, i.e. $\lambda_{noobj}=0.5$.

4. **5th term (Class Probabilities):** SSE of class probabilities when there is objects.

5. **λ_{coord} :** Due to the same reason mentioned in 3rd and 4th terms, $\lambda_{coord}=5$ to increase the loss from bounding box coordinate predictions.

5.2 THE PREDICTIONS VECTOR

The first step to understanding YOLO is how it encodes its output. The input image is divided into an $S \times S$ grid of cells. For each object that is present on the image, one grid cell is said to be “responsible” for predicting it. That is the cell where the center of the object falls into.

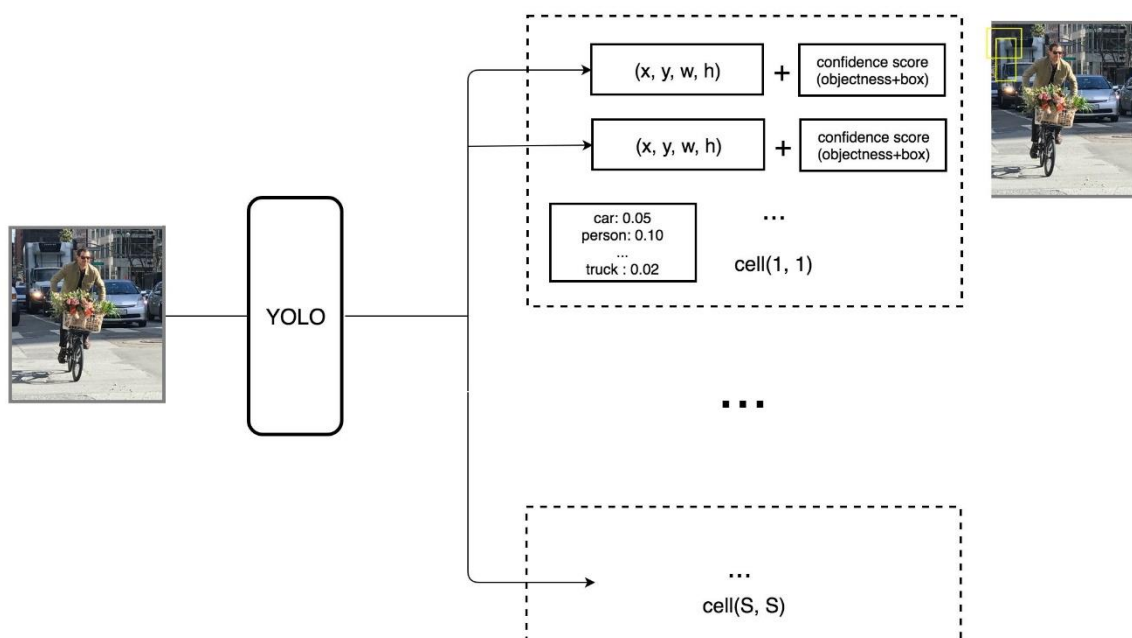


Fig: 5.1 DETECTION OF OBJECTS USING YOLO

The Figure 5.1 depicts the detection of objects using YOLO weights. Each grid cell predicts B bounding boxes as well as C class probabilities. The bounding box prediction has 5 components: $(x, y, w, h, confidence)$. The (x, y) coordinates represent the center of the box, relative to the grid cell location (remember that, if the center of the box *does not* fall inside the grid cell, this cell is not responsible for it). These coordinates are normalized to fall between 0 and 1. The (w, h) box dimensions are also normalized to $[0, 1]$, relative to the image size.

5.3 INTERNAL WORKING OF YOLO

Once the predictions are encoded, the rest is easy. The network structure looks like a normal CNN, with convolutional and max pooling layers, followed by 2 fully connected layers in the end. Each boundary box contains 5 elements: (x, y, w, h) and a box confidence score. The confidence score reflects how likely the box contains an object (objectness) and how accurate is the boundary box. Now, normalize the bounding box width w and height h by the image width and height. x and y are offsets to the corresponding cell. Hence, x, y, w and h are all between 0 and 1. Each cell has 20 conditional class probabilities. The conditional class probability is the probability that the detected object belongs to a particular class (one probability per category for each cell). So, YOLO's prediction has a shape of $(S, S, B \times 5 + C) = (7, 7, 2 \times 5 + 20) = (7, 7, 30)$. The Figure 5.2 depicts the YOLO Neural Network Architecture with different layers.

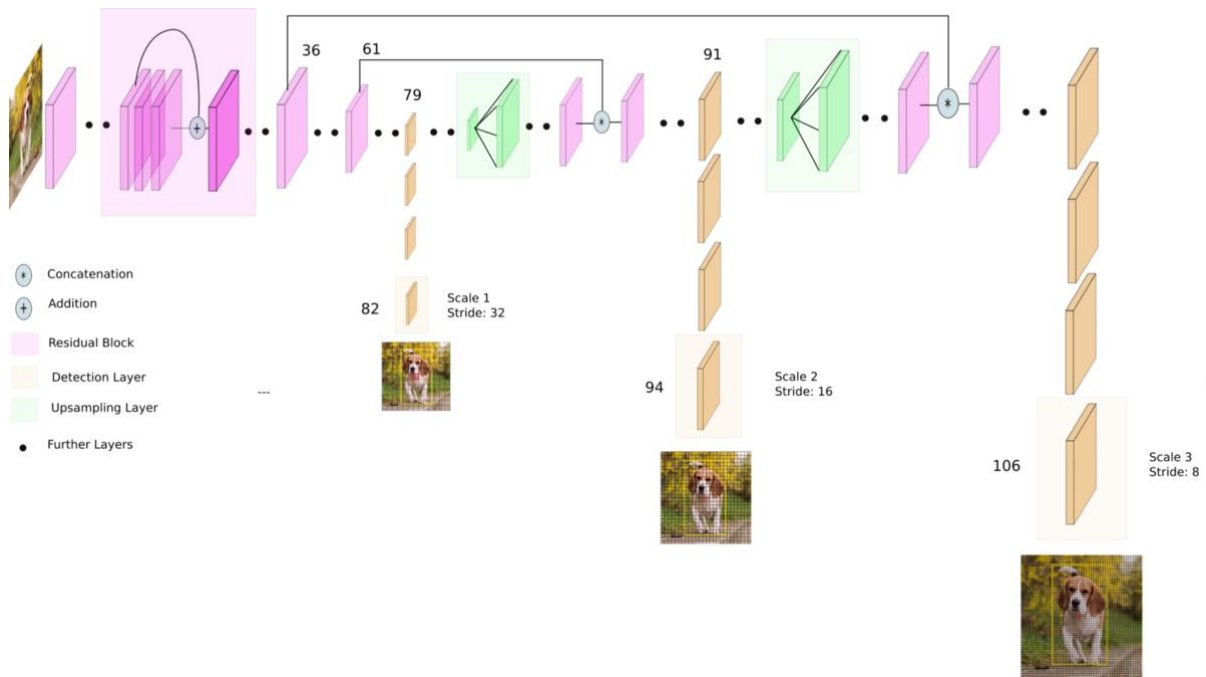


FIGURE 5.2 YOLO v3 NETWORK ARCHITECTURE

The major concept of YOLO is to build a CNN network to predict a (7, 7, 30) tensor. It uses a CNN network to reduce the spatial dimension to 7×7 with 1024 output channels at each location. YOLO performs a linear regression using two fully connected layers to make $7 \times 7 \times 2$ boundary box predictions. To make a final prediction, keep those with high box confidence scores (greater than 0.25) as our final predictions.

CHAPTER 6

IMPLEMENTATION AND EXPERIMENTATION RESULTS

```
Windows PowerShell
PS C:\Users\ibcn\Documents\darknetWindows\darknet\build\darknet\x64> ./darknet.exe detector train cfg/obj.data cfg/yolo-obj.cfg darknet19_448.conv.23
yolo-obj
layer  filters  size  input  output
0 conv  32  3 x 3 / 1  416 x 416 x 3  -> 416 x 416 x 32
1 max  2  2 x 2 / 2  416 x 416 x 32  -> 208 x 208 x 32
2 conv  64  3 x 3 / 1  208 x 208 x 32  -> 208 x 208 x 64
3 max  2  2 x 2 / 2  208 x 208 x 64  -> 104 x 104 x 64
4 conv  128 3 x 3 / 1  104 x 104 x 64  -> 104 x 104 x 128
5 conv  64  1 x 1 / 1  104 x 104 x 128 -> 104 x 104 x 64
6 conv  128 3 x 3 / 1  104 x 104 x 64  -> 104 x 104 x 128
7 max  2  2 x 2 / 2  104 x 104 x 128 -> 52 x 52 x 128
8 conv  256 3 x 3 / 1  52 x 52 x 128  -> 52 x 52 x 256
9 conv  128 1 x 1 / 1  52 x 52 x 256  -> 52 x 52 x 128
10 conv 256 3 x 3 / 1  52 x 52 x 128  -> 52 x 52 x 256
11 max  2  2 x 2 / 2  52 x 52 x 256  -> 26 x 26 x 256
12 conv 512 3 x 3 / 1  26 x 26 x 256  -> 26 x 26 x 512
13 conv 256 1 x 1 / 1  26 x 26 x 512  -> 26 x 26 x 256
14 conv 512 3 x 3 / 1  26 x 26 x 256  -> 26 x 26 x 512
15 conv 256 1 x 1 / 1  26 x 26 x 512  -> 26 x 26 x 256
16 conv 512 3 x 3 / 1  26 x 26 x 256  -> 26 x 26 x 512
17 max  2  2 x 2 / 2  26 x 26 x 512  -> 13 x 13 x 512
18 conv 1024 3 x 3 / 1  13 x 13 x 512  -> 13 x 13 x 1024
19 conv 512 1 x 1 / 1  13 x 13 x 1024 -> 13 x 13 x 512
20 conv 1024 3 x 3 / 1  13 x 13 x 512  -> 13 x 13 x 1024
21 conv 512 1 x 1 / 1  13 x 13 x 1024 -> 13 x 13 x 512
22 conv 1024 3 x 3 / 1  13 x 13 x 512  -> 13 x 13 x 1024
23 conv 1024 3 x 3 / 1  13 x 13 x 1024 -> 13 x 13 x 1024
24 conv 1024 3 x 3 / 1  13 x 13 x 1024 -> 13 x 13 x 1024
25 route 16
26 conv 64 1 x 1 / 1  26 x 26 x 512  -> 26 x 26 x 64
27 reorg  / 2  26 x 26 x 64  -> 13 x 13 x 256
28 route 27 24
29 conv 1024 3 x 3 / 1  13 x 13 x 256  -> 13 x 13 x 1024
30 conv 30 1 x 1 / 1  13 x 13 x 1024 -> 13 x 13 x 30
31 detection
Loading weights from darknet19_448.conv.23...Done!
Learning Rate: 0.001, Momentum: 0.9, Decay: 0.0005
Resizing
448
Loaded: 0.000000 seconds
Region Avg IOU: 0.204261, Class: 1.000000, Obj: 0.492693, No Obj: 0.486198, Avg Recall: 0.111111, count: 18
Region Avg IOU: 0.243076, Class: 1.000000, Obj: 0.504338, No Obj: 0.486026, Avg Recall: 0.047619, count: 21
Region Avg IOU: 0.325906, Class: 1.000000, Obj: 0.519450, No Obj: 0.485805, Avg Recall: 0.315789, count: 19
Region Avg IOU: 0.278630, Class: 1.000000, Obj: 0.504646, No Obj: 0.486410, Avg Recall: 0.111111, count: 18
1: 17.108604, 17.108604 avg, 0.001000 rate, 2.823000 seconds, 64 images
Loaded: 0.000000 seconds
Region Avg IOU: 0.247192, Class: 1.000000, Obj: 0.073060, No Obj: 0.078055, Avg Recall: 0.157895, count: 19
Region Avg IOU: 0.307193, Class: 1.000000, Obj: 0.044096, No Obj: 0.080080, Avg Recall: 0.074074, count: 27
Region Avg IOU: 0.343057, Class: 1.000000, Obj: 0.044127, No Obj: 0.075404, Avg Recall: 0.272727, count: 22
Region Avg IOU: 0.424638, Class: 1.000000, Obj: 0.048766, No Obj: 0.077142, Avg Recall: 0.263158, count: 19
2: 2.749439, 15.672688 avg, 0.001000 rate, 2.799000 seconds, 128 images
Loaded: 0.000000 seconds
Region Avg IOU: 0.369713, Class: 1.000000, Obj: 0.010010, No Obj: 0.007427, Avg Recall: 0.318182, count: 22
Region Avg IOU: 0.288321, Class: 1.000000, Obj: 0.005620, No Obj: 0.007315, Avg Recall: 0.166667, count: 18
Region Avg IOU: 0.321140, Class: 1.000000, Obj: 0.003912, No Obj: 0.007123, Avg Recall: 0.200000, count: 15
Region Avg IOU: 0.297358, Class: 1.000000, Obj: 0.004159, No Obj: 0.007017, Avg Recall: 0.222222, count: 18
3: 1.772242, 14.282643 avg, 0.001000 rate, 2.799000 seconds, 192 images
Loaded: 0.000000 seconds
Region Avg IOU: 0.370476, Class: 1.000000, Obj: 0.000793, No Obj: 0.000966, Avg Recall: 0.375000, count: 16
Region Avg IOU: 0.374387, Class: 1.000000, Obj: 0.000556, No Obj: 0.000985, Avg Recall: 0.260870, count: 23
Region Avg IOU: 0.289484, Class: 1.000000, Obj: 0.001393, No Obj: 0.001015, Avg Recall: 0.190476, count: 21
Region Avg IOU: 0.306027, Class: 1.000000, Obj: 0.001258, No Obj: 0.001070, Avg Recall: 0.117647, count: 17
4: 1.551824, 13.009562 avg, 0.001000 rate, 2.798000 seconds, 256 images
```

FIGURE 6.1 TRAINING DATASET USING DARKNET

This chapter represents the output of the images obtained through the implementation of the proposed system. The Figure 6.1 represents the output of training the dataset through darknet. The Figure 6.2 and Figure 6.3 represents the elephant object detected output. The Figure 6.4 represents the tiger object detected output.

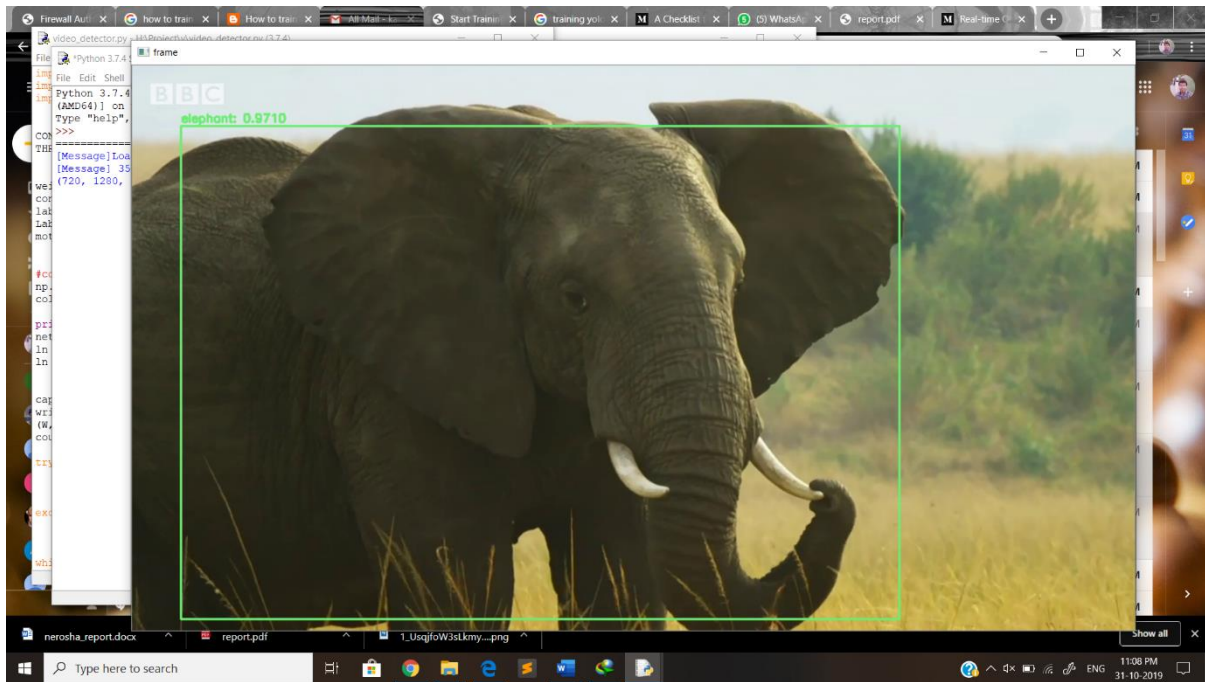


FIGURE 6.2 SINGLE ELEPHANT INTRUSION DETECTION USING YOLO

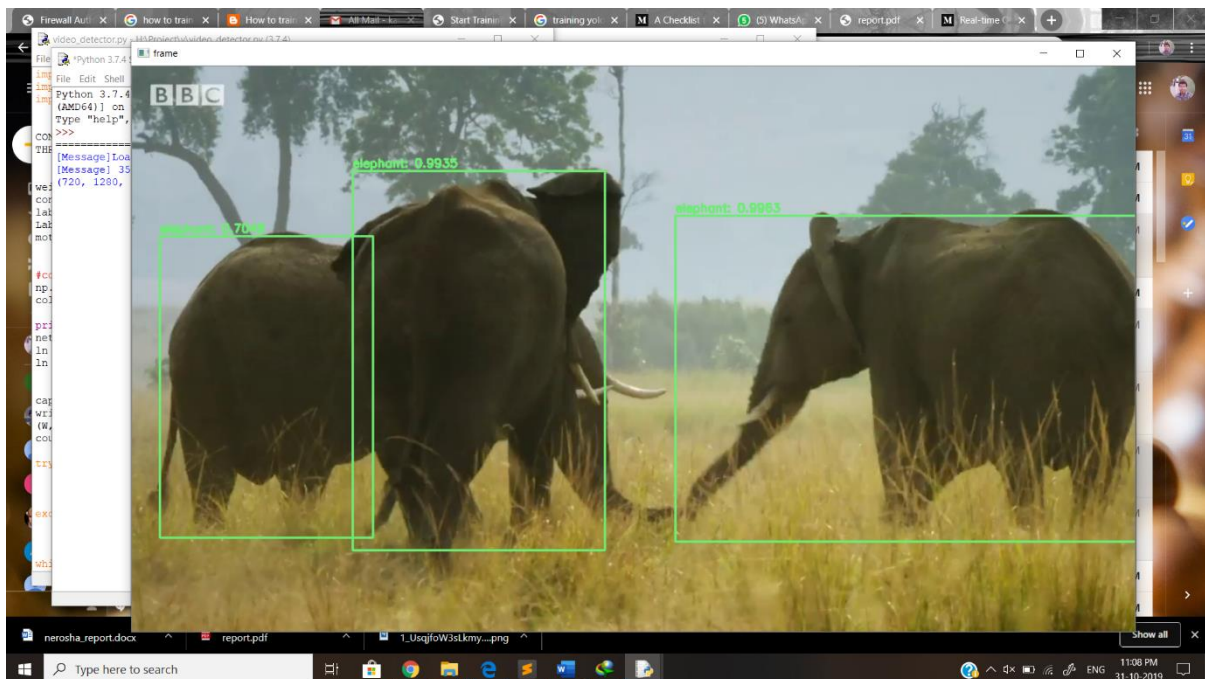


FIGURE 6.3 MULTIPLE ELEPHANT INTRUSION DETECTION

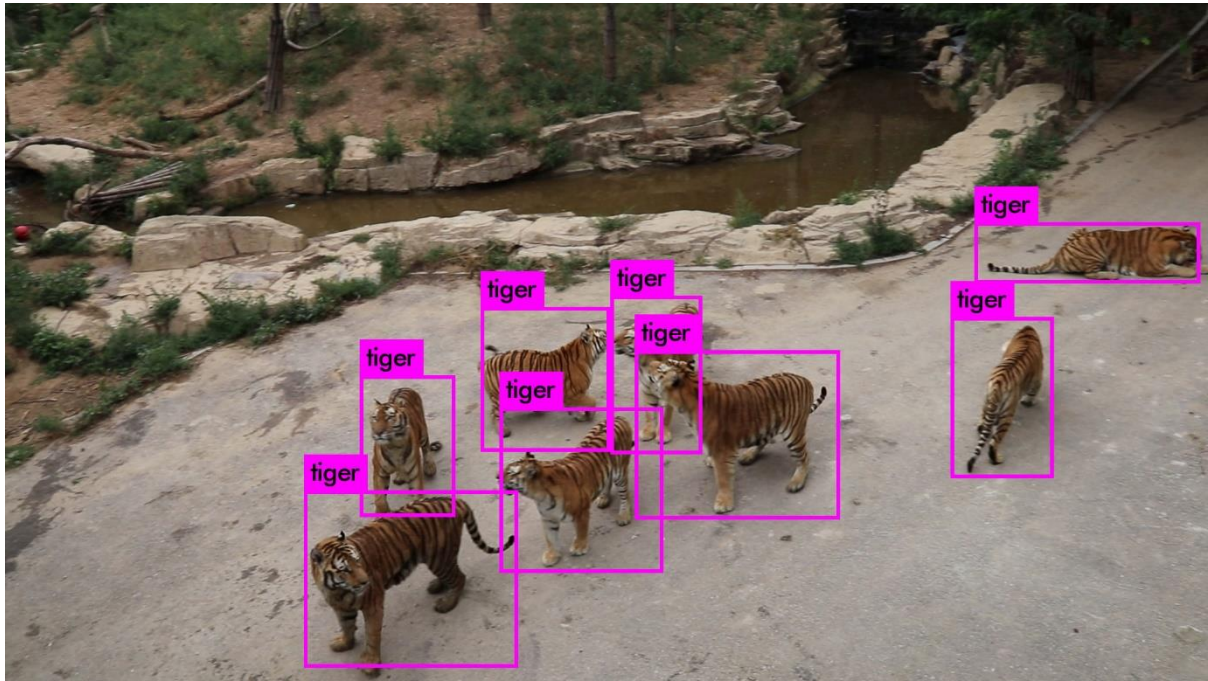


FIGURE 6.3 TIGER INTRUSION DETECTION

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

Video Surveillance monitoring is an important and integral component of daily life. The proposed work demonstrates the application of OpenCV and object detection algorithm to detect the intrusion of particular animal and helps to intimate the respected user about the intrusion. The exact location of the animals is plotted on the video. Moreover, the accuracy of our system is checked for different environments. It has been observed that the object having occlusion and noise has less accuracy as compared to other objects. Thus, by using this system intrusion detection can be done independently with minimal human intervention.

7.2 FUTURE WORK

No software can ever demand to be completely self complete and independent without requiring any future modification. The problem of intrusion detection using object detection is very broad field and many problems are yet to be solved. Future research work involves the classification of intrusion based on animals and humans

REFERENCES

- [1] Sanku Kumar Roy and Arijit Roy (2015) proposed a prototype “Agricultural Intrusion Detection using wireless sensor network” to detect the intrusion of animals into farming lands using AVR micro-controller.
- [2] Prajna P, Soujanya B.S, and Divya M (2018) proposed a “IoT- based Wild Animal Detection System” in International Journal for Engineering and Technology.
- [3] Emmanuel Onwuka Ibam and Mark O Afolabi (2018) proposed “Farm Monitoring and Security System” using Wireless Network Sensor and GSM Module.
- [4] Kushal Voona, Satya Ravi Teja and Sai Srikar (2015) proposed "Photosensitive security system for theft detection and control using GSM technology" which uses photosensitive sensor LDR and GSM module.
- [5] Fang Y (2010), “Motion Based Animal Detection in Aerial Videos” discussed a technique to move animal detection by taking benefit of global patterns of pixel motion.
- [6] Parham, J., Stewart, C., Crall, J., Rubenstein, D., Holmberg, J., & Berger-Wolf, T. (2018). An “Animal Detection Pipeline for Identification” IEEE Winter Conference on Applications of Computer Vision” (WACV).
- [7] Villa A. G. (2017), “Automatic Wild Animal Monitoring: Identification of Animal Species in Camera-trap Images using Very Deep Convolutional Neural Networks” stated the main problems inherent to camera trapping images automatic species identification.
- [8] Besteiro R. (2010), “Agreement between passive infrared detector measurements and human observations of animal activity” Passive infrared (PIR) detector and the sensor’s digital signal for executing

- [9] Matuska S. (2011), “Classification of Wild Animals Based on SVM and Local Descriptors” discussed a new approach for object recognition by using hybrid local descriptors.
- [10] Zhu C. and Rey N. (2013), “Automatic Wild Animal Detection in Low Quality Camera-trap Images Using Two-channelled Perceiving Residual Pyramid Networks” introduced a two-channelled perceiving residual pyramid networks towards automatic wild animal detection in low quality camera-trap images.
- [11] Abdesselam Bouzerdoun , Azeddine Beghdadi and Philippe L. Bouttefroy ,(2013) “On the analysis of background subtraction techniques using Gaussian mixture models,” IEEE International Conference on Acoustics, Speech, and Signal Processing (pp. 4042-4045). USA: IEEE.
- [12] Abhishek Kumar Chauhan and Prashant Krishan, April (2013)“Moving Object Tracking using Gaussian Mixture Model and Optical Flow,” International Journal of Advanced Research in Computer Science and Software Engineering , Volume 3, Issue 4.
- [13] Arti Tiwari & Jagvir Verma, June (2014) “Scene understanding using back propagation by neural network “International Journal of Image Processing and Vision Sciences (IJIPVS) ISSN (Print): 2278 – 1110, Vol-1 Iss-2.9. Simon Haykin,Neural Networks: A Comprehensive Foundation,Prentice Hall,1999.
- [14] Thierry Bouwmans, Fida El Baf, Bertrand Vachon, May (2010) “Background Modeling using Mixture of Gaussians for Foreground Detection - A Survey”.
- [15] Eduardo Velloso, Andreas Bulling, Hans Gellersen, Wallace Ugulino, and Hugo Fuks. April (2013). “Qualitative activity recognition of weight lifting exercises”.In Proceedings of the 4th Augmented Human International Conference. ACM,116–123.
- [16] Scott R Watterson, David Watterson, and Mark D Watterson, May (2013).

“Systems and Methods to Generate a Customized Workout Routine”. (Aug. 1 2013). US Patent App. 13/754,361.

[17] Schuldhaus D., Leutheuser H., and Eskofier B., September (2012) “Automatic Classification of Sport Exercises for Training Support,” Proceedings of Symposium der dvs- Sektion Sportinformatik, Konstanz, Germany.

[18] Seeger C., Buchmann A., and Van Laerhoven K., November (2011) “MyHealthAssistant: A Phone-based Body Sensor Network that Captures the Wearer’s Exercises throughout the Day,” Proceedings of the 6th International Conference on Body Area Networks (BodyNets), Beijing, China.

[19] F. Buttussi and L. Chittaro. Mopet: May (2010) “A context-aware and user-adaptive wearable system for fitness training”, Artificial Intelligence in Medicine.

[20] A. Giachetti, April (2011) “Matching Techniques to Compute Image Motion”. Image and Vision Computing 18(2000). Pp.247-260.