Engineering Clinics

Project Report

FALL SEM 2022-23

Title: Snake Detection

Submitted by

GAVARA SAI SREE HARSHITH (20BCE7398)
KAVALKUNTLA RISHI KARTHIKEYA REDDY (20BCI7016)
DASARI SHIVA (20BCE7075)
DASARI SRIKANTH (20BCE7076)
RAGHUNATH SINGH (20BCI7012)
BHAVANA VENKATESH (20BCE7064)

Guide:
Prof.Y Mohamed Sirajudeen.

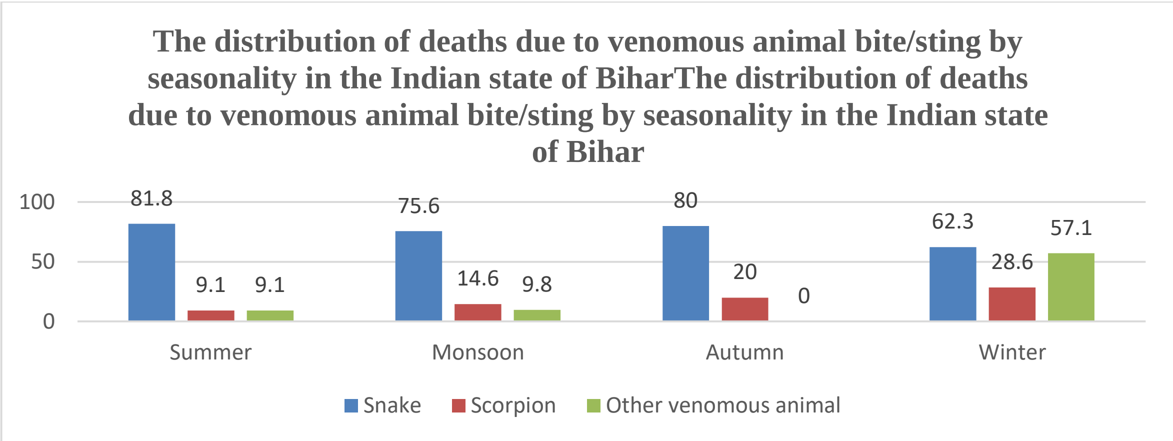
VIT-AP UNIVERSITY

AMARAVATI

ANDHRA PRADESH, INDIA

2022

Introduction:
Snakes are elongated and limbless carnivorous reptiles. Though snakes are said to maintain balance in the ecological systems, they have always remained a possible threat to humans. Snake's bites are a common fatality that causes death. Through the use of Arduino, we can easily detect the snake and catch it. The presentation will give deep insights into the project.
Real-life problems with snake bites:
Snakebites kill more than 46,000 people in India every year. They are probably the most 'neglected tropical disease', according to Study Protocols for Knowing the Incidence of Snakebites. Hospital-based data on snakebite admissions and anti-snake venom use have been underestimated, it stated.
A recent study on kraits, a highly venomous species of snake, has highlighted the lacunae in research into medically significant snakes and snake venom in India, an expert not associated with the study told Down to earth. India is home to over 300 species of snakes. Of these, only around 60 described snake species are capable of causing harm to humans. Nonetheless, over 58,000 people die in India every year and three times the number suffer permanent disabilities due to snakebites, according to a press statement the authors gave while releasing the study. Sunagar, Martin and others examined the phylogenetic histories and venoms of several 'look-alike' kraits in southern and western India to understand the differences in evolutionary histories and venom compositions and the consequent impact of this variation on the effectiveness of snakebite treatment.



How the proposed model will overcome the real-world problems?

1. Our model will help in alarming the people when there is a presence of a snake in their surroundings. Thus, the buzzer in our model can help people in safeguarding themselves. The Raspberry Pi model is efficient in working even in adverse climatic conditions like rains and storms.

2. Even the snake is discovered in the surroundings people generally tend to kill the snake out of fear. But, here our proposed bait system will put the snake in a black bag to be safely released into the forest area without it being killed.

Objectives:
1. To detect snake using Machine Learning Algorithms.
2. Suggest a possible bait system to catch snake.
3. To discover the poisonousness of the snakes detect using Machine Learning Algorithms.

Motivation:
Snakes often come in our university campus and it is very scary for the students. One day a student even died of snake bite and there are many such cases reported in the rural areas.
And also, Farmers are suffering a lot due to more snakes approaching their farm. So we were motivated by these situations.

Survey on similar Snake Detection products:

1. **A comparative study on image-based snake identification using machine learning:-**

   Automated snake image identification is important from different points of view, most importantly, snake bite management. Auto-identification of snake images might help the avoidance of venomous snakes and also providing better treatment for patients. In a similar study, for the first time, it's been attempted to compare the accuracy of a series of state-of-the-art machine learning methods, ranging from the holistic to neural network algorithms. Compared to holistic methods, convolutional neural networks show similar to better performance, and accuracy reaches 93.16% using MobileNetV2. Visualizing intermediate activation layers in VGG model reveals that just in deep activation layers, the colour pattern and the shape of the snake contribute to the discrimination of snake species. This study presents MobileNetV2 as a powerful deep convolutional neural network algorithm for snake image classification that could be used even on mobile devices. This finding pave the road for generating mobile applications for snake image identification.

2. **A Survey on Snake Species Identification using Image Processing Technique:-**

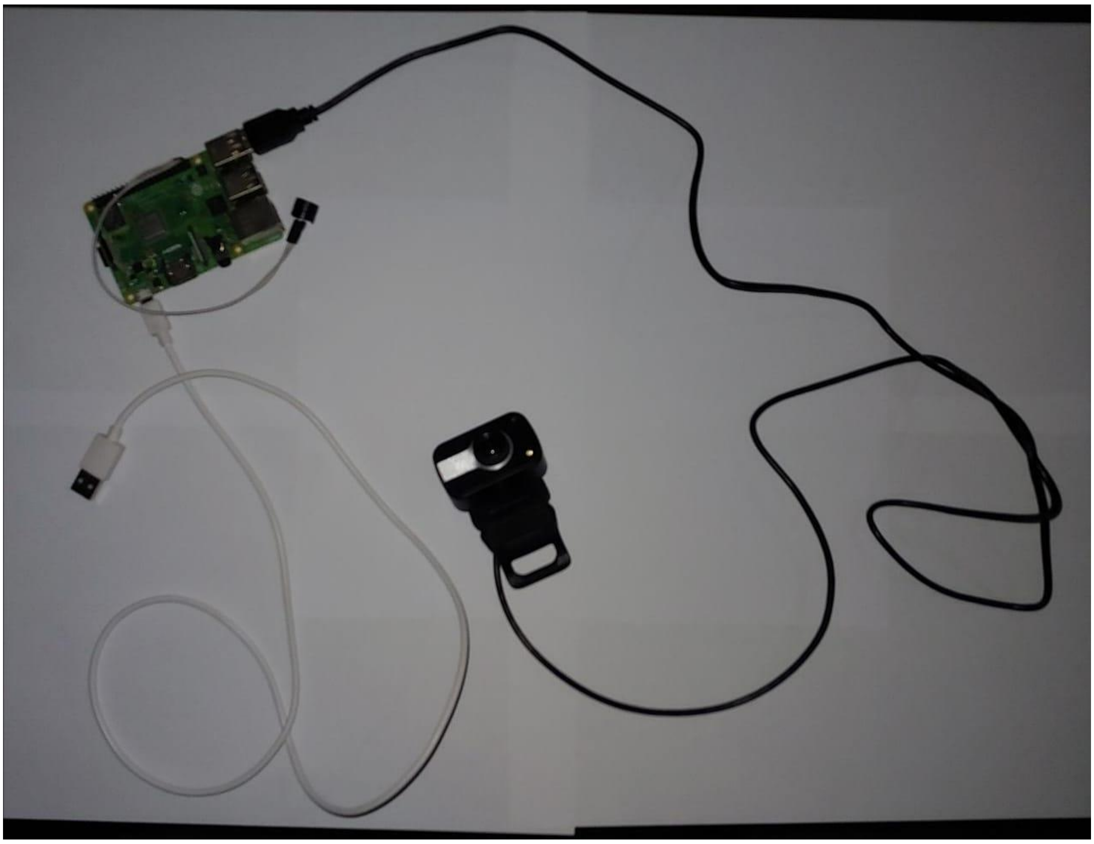
   Proposed system uses two modules, one of them is object detection and another one is classification. Object detection module will detect snake as an object from input and classification module will classify object into appropriate species of snake. Object detection module, uses YOLO (YOU ONLY LOOK ONCE) algorithm. YOLO uses deep learning and convolution neural networks (CNN) for object detection, as the name indicates it only needs to "see" each image once. This allows YOLO to be one of the fastest detection algorithms and can detect objects in real time (up to 30 FPS).

Technologies That can solve the Snake bites:
WHO has now launched the Snakebite Information and Data Platform, which is a confluence of technology with data science and geography, using a state-of-the-art solution ArcGIS. The Platform will serve as a hub with range maps depicting venomous snakes, identifying photos, and information supported by an integrated anti-venom products database. "This platform was developed with support from technology partner Esri, academic partner University of Geneva and the Foundations and our colleagues from the Department of Neglected Tropical Diseases,"Once the patient has arrived in hospital, prevention of morbidity and mortality will depend on the training of medical staff and provision of necessary equipment, antivenom and other drugs. Guidelines for the SEARO region, including India, have been published by WHO. The use of anti-venom is crucial for controlling the complications arising from snakebite envenoming. According to WHO, a significant challenge in the manufacturing of anti-venoms is the preparation of the correct immunogens (snake venoms). At present, very few countries have the capacity to produce snake venoms of adequate quality for anti-venom manufacture, and many manufacturers rely on common commercial sources. Additionally, lack of regulatory capacity for the control of anti-venoms in countries with significant snakebite problems results

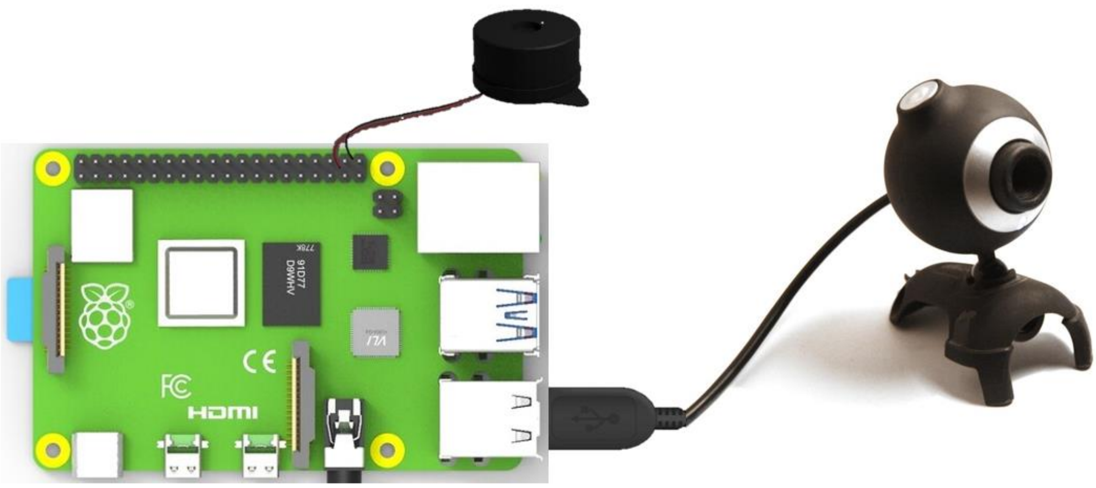
in the inability to assess the quality and appropriateness of the anti-venoms. Less than 2% of Africans currently have access to anti-venoms.

System Model:

- **Architecture Design:**



- **3D Prototype:**



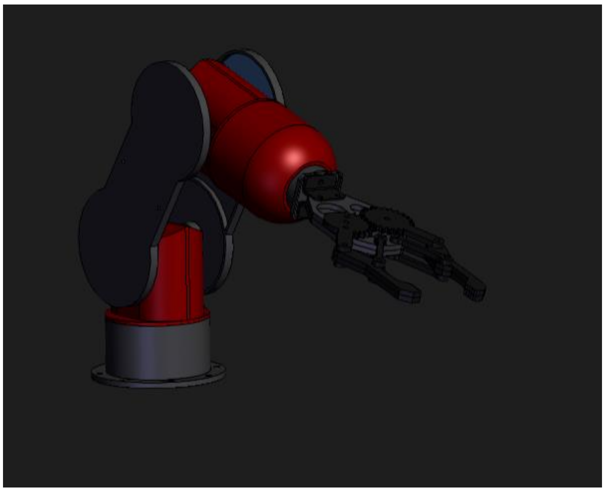
Components:

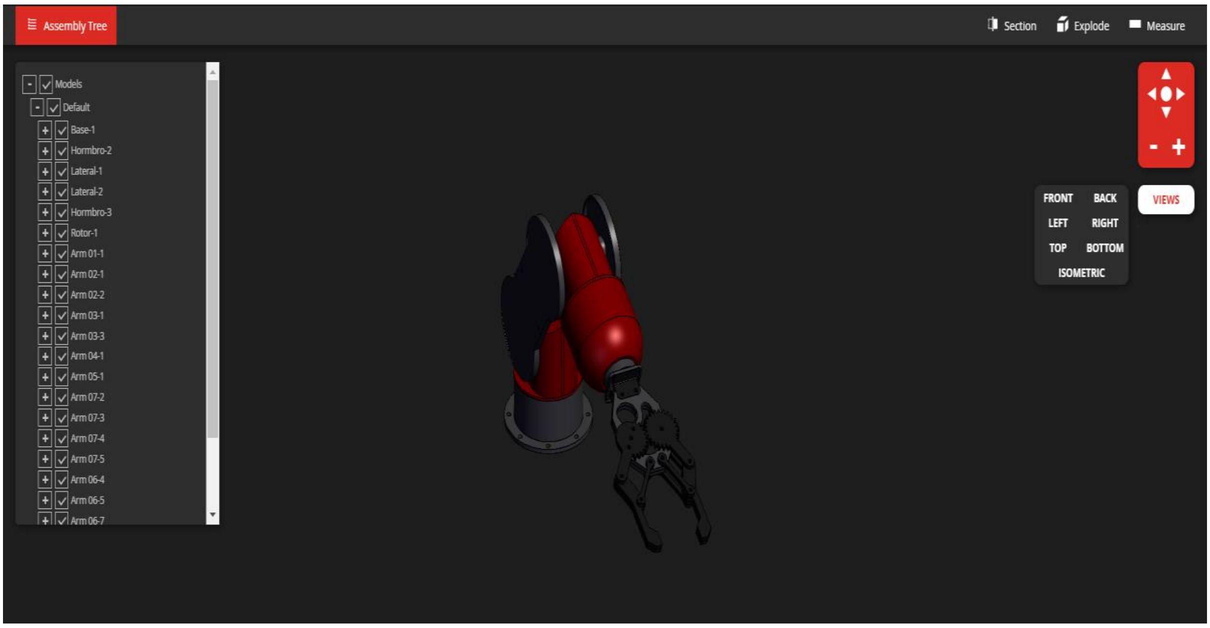
1. USB Camera:
It is used for image acquisition and processing application, based on TTL communication interface, very convenient to connect with Arduino controller, able to read image and data via UART and serial port.

2. Alarm (AE20M-24FA):
It will respond when the presence of snake was confirmed.

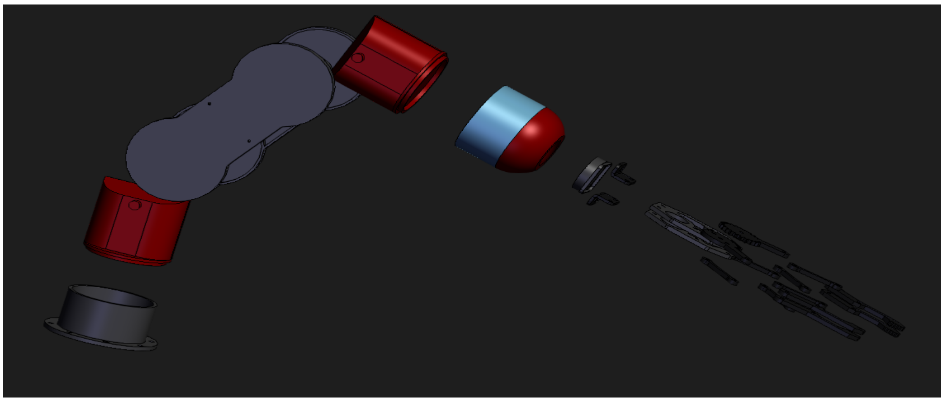
3. Raspberry Pi 4:
The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python.

4. Robotic Arm 3D Model Design:
The use of robotic arms also frees human workers from performing tasks that present a risk of bodily injury.





Camera capture the Snake and then the Buzzer will ring and then Robotic arm will catch the Snake.



Assembling components:

We have connected the ground pin on the buzzer to a GND pin on the Raspberry PI-3 and the positive buzzer pin to a standard GPIO pin on the Raspberry PI-3, The USB camera is connected to the Raspberry Pi USB port. The power supply cable has to be pinned to the USB port of a working laptop. The hardware assembling was done in the Engineering Clinics LAB of VIT-AP University.

Results:

Evaluation:
We have used a image dataset of various snake species found in India. The dataset is classified into train and test previously and then to Non-Venomous and Venomous. It contains images of cobras to vipers to rat snake and green tree vine. The size of every image is 400×400.

Snake Detection using Tensorflow Object Detection API

Tensorflow Object Detection API

Creating accurate machine learning models capable of localizing and identifying multiple objects in a single image remains a core challenge in computer vision. The TensorFlow Object Detection API is an open source framework built on top of TensorFlow that makes it easy to construct, train and deploy object detection models.

Requirements

1. Windows/Ubuntu
2. GPU (NVIDIA 1070 is used)
3. Tensorflow
4. Python v2.7 or Python > v3.0
5. OpenCV > v4.0

Directory structure recommondation:

```
.
├── data
│   ├── test_labels.csv
│   └── train_labels.csv
├── images
│   ├── test
│   │   └── test_image-1.jpg
│   └── train
│       └── trainin_image_1.jpg
├── training
└── xml_to_csv.py
```

Stages involved in training a model:

1. Data Collection.
2. Annotation.

3. Train-Test Data.
4. Generate TF Records files.
5. Model Architecture.
6. Train.
7. Frozen model generation.
8. Inference.

1. Data Collection:

a) Based on custom object, we have collected data.

b) We have Increase the data size by applying image processing techniques like R,G,B pixel intensity increment, rotation.

c) Collected the objects in varient possible distances.

2. Annotation:

Annotation of images also called as Labeling data. “Labeling” is the tool to be used for generation of annotation files. Labeling is a graphical image annotation tool .It is written in Python and uses Qt for its graphical interface. Annotations are saved as XML files in PASCAL VOC format, the format used by Image Net. Besides, it also supports YOLO format. Downloaded tool from github repot.

I. Tool usage:

a) Clone the repo.

b) Run ‘python labelImg.py’

II. label the image:

a) Click 'Change default saved annotation folder' in Menu/File.

b) Click 'Open Dir'.

c) Click 'Create Rectangle Box'.

d) Click and release left mouse to select a region to annotate the rectangle box.

e) You can use right mouse to drag the rectangle box to copy or move it. The annotation will be saved to the folder you specify.

3. Train-Test Data:

Split the data into Train and Test data. Normally, 70% of data will be used for Training and 30% data will be used for Testing.

4. Generate TF Record files:

TF record files are Binary representation of the data. Convert the data from stage #3 to train.tfrecord and test.tfrecord files respectively.
I. XML to CSV conversion : Convert all XMLs to a csv file using below python script (Many techniques can be used). Run the script for train_data and test_data.

II. CSV to TF Record file: Grab “generate_tfrecord.py” from tensorflow/models directory. And the only modification need to change is ‘class_text_to_int’ with the label assigned in stage #2.

5. Model Architecture:

1. Selecting a model architecture can be a pretrained model or new architecture. As new architecture takes days of time to train, prefer pretrained model. Download weights from tensorflow zoo. Select a model.
2. pbtxt file: Object-class ID represented using object_class.pbtxt file.
```
item {
  id: 1
  name: 'person'
}
```
ID number increases based on the number of objects that to be trained. (Note: label name must be same which is used in stage #2.
3. Configuration files: modify the configuration files for ‘num_classes’ and ‘PATH_TO_BE_CONFIGURED’ with the number of classes to be trained and train.tf record and test.tfrecord. (Note: total of 6 changes are important. Other changes depends on model, data, train/test examples).

6. Train:

1. clone the tensorflow models repo.
2. From models/research/object_detection, run python3 train.py --logtostderr --train_dir=training/ --pipeline_config_path=<config_file>.
3. On successful run, 3 files will be generated .data, .index, .meta which together called as “Unfrozen Model”.
4. Based on the configuration, unfrozen models are generated for various step_size.

7. Frozen Model Generation:

1. From models/research/object_detection, run python3 export_inference_graph.py --input_type image_tensor --pipeline_config_path training/ --trained_checkpoint_prefix training/model.ckpt-<step_size> --output_directory <output_dir>.
2. Result is frozen model “<output_name>.pb”.

Once the camera is detecting the snake image we have to manually enter the “S” button on keyboard for the device to select the image. The image is sent to the Raspberry Pi. There is the presence of operating system in Raspberry pi to run the python code for Machine Learning. After executing the ML Code the results are displayed.

CODE:

import tensorflow as tf

```
import sys
import time
import cv2
import RPi.GPIO as GPIO
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

buz=26
GPIO.setup(buz,GPIO.OUT)

GPIO.output(buz,0)


vs = cv2.VideoCapture(0)
time.sleep(3)
while(True):
        (grabbed, frame) = vs.read()
        cv2.imshow('input',frame)
        if cv2.waitKey(1) & 0xFF == ord('s'):
            break

cv2.imwrite('test.jpg',frame)
cv2.waitKey(10)

image_path = 'test.jpg'
graph_file = "Venomous_nonVenomous_inception.pb"
labels_txt = "snake_labels.txt"

with tf.gfile.FastGFile(graph_file, 'rb') as f:
        graph_def = tf.GraphDef()
        graph_def.ParseFromString(f.read())
        _ = tf.import_graph_def(graph_def, name='')
image_data = tf.gfile.FastGFile(image_path, 'rb').read()
label_lines = [line.rstrip() for line in tf.gfile.GFile(labels_txt)]


with tf.Session() as sess:
        softmax_tensor = sess.graph.get_tensor_by_name('final_result:0')
        predictions = sess.run(softmax_tensor,{'DecodeJpeg/contents:0': image_data})
        top_k = predictions[0].argsort()[-len(predictions[0]):][::-1]
        temp = ['','']
        i=0
        for node_id in top_k:

                human_string = label_lines[node_id]
                score = predictions[0][node_id]
                print('%s (score = %.5f)' % (human_string, score))
                print(score)
                i=i+1
                if(i==1 and score>0.5):
```

```
GPIO.output(buz,1)
print('buzzer on')
time.sleep(5)
print('buzzer off')
GPIO.output(buz,0)
```

Screenshots:

- **Input:**

- **Output:**

```
venomous (score = 0.89827)
0.898267
buzzer on
buzzer off
non venomous (score = 0.10173)
0.10173307
>>>
```

In the above Result it shows that how much the snake is Venomus.

Future Scope:

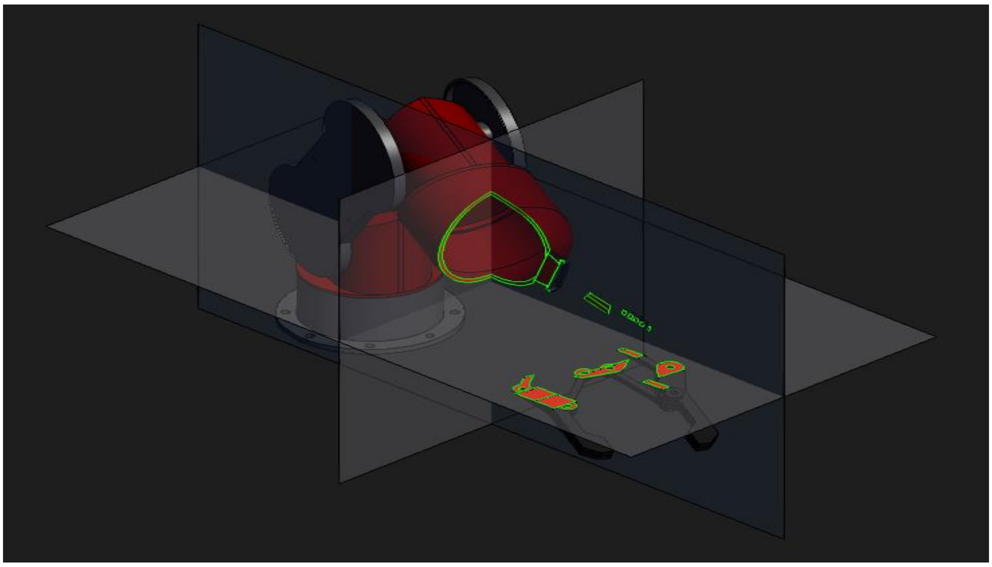
Because of more budget,We Designed a Robotic Arm in online Fusion 360 . In future we will proceed to buy robotic arm and perform physically.

A robotic arm is a type of mechanical arm, usually programmable, with similar functions to a human arm; the arm may be the sum total of the mechanism or may be part of a more complex robot. The links of such a manipulator are connected by joints allowing either rotational motion (such as in an articulated robot) or translational (linear) displacement.The links of the manipulator can be considered to form a kinematic chain. The terminus of the kinematic chain of the manipulator is called the end effector and it is analogous to the human hand. However, the term "robotic hand" as a synonym of the robotic arm is often proscribed.

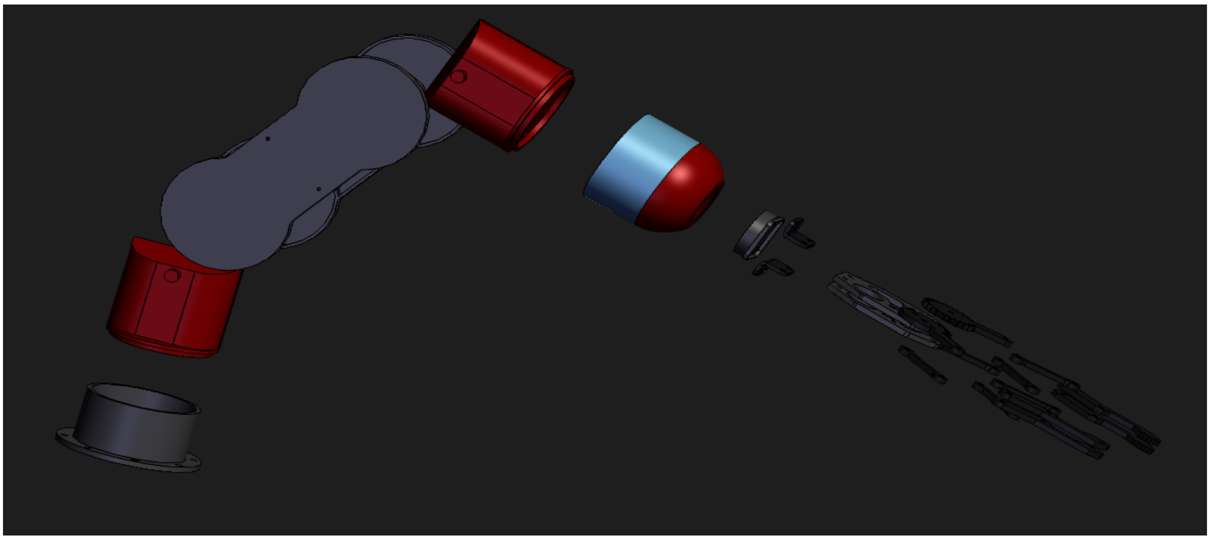
In our project the robotic arm is integrated to catch the snake and prevent it from harming us. The robotic arm is capable of lifting the moving snake and put it safely in a black bag, which will preserve the snake. The black bag can be transferred to the disaster management authorities to safely release the snake in the forest area.

The cost of manufacturing the robotic arm for snake bait will around Rs 22,000. Since we are sort of the required budget at the graduation level we have not incorporated it in the design. In future the Robotic arm will the center of research and innovation for the project. Our Peers can apply it to make a successful wait system.

XYZ PLANE ORIENTATION:



DIVISION OF PARTS:



THE END