SNAKES DETECTION IN AGRICULTURE FIELD USING MACHINE LEARNING

Minor project report submitted

In partial fulfilment of the requirement for award of the degree of

Bachelor of Technology

In

Electronics and Communication Engineering

By

U. Karuna (S180322)

Ch.Sonika (S180213)

A.Priyalatha (S180088)

M.Rani (S180752)

Y.Sai Santhoshi (S180066)

Under the guidance of Mrs. K.VASUNDHARA (M.Tech)

Department of Electronics and communication Engineering

Rajiv Gandhi University of Knowledge Technologies

SRIKAKULAM-532410 JULY 2023

Certificate

This is certify that the work entitled " **Snakes Detection in Agriculture Fields using Machine Learning**" is a bonafide record of authentic work carried out by S180322, S180088, S180066, S180752, S180213 under my supervision and guidance for the partial fulfilment of the requirement of the award of the degree of Bachelor of Technology in the department of Electronics and Communication Engineering at **RGUKT-SRIKAKULAM**

The results embodied in this work have not been submitted to any other university or institute for the award of any degree or diploma. This certifies, in our opinion, is worthy of consideration for the award of the degree of Bachelor of Technology in accordance with the regulations of the institute.

Mrs.K.Vasundhara (M.tech)

Assistant professor,

Department of ECE,

RGUKT- Srikakulam

Mr.N.Ramesh Babu (M.tech)

Head of the Department,

Department of ECE,

RGUKT- Srikakulam

Acknowledgement

I would like to express my special thanks of gratitude to my project guide **Mrs.K.Vasundhara** **(M.tech),** Assistant Professor, Department.of ECE, who gave me the golden opportunity to do this wonderful project on the topic "**Snakes Detection System**" which also helped me in doing a lot of research and I came to know about so many new things for which I am really thankful to them.

Electronics and communication Engineering

IIIT-SRIKAKULAM

Date: -03-2023

Place:-NUZVID

Table of Contents:

ABSTRACT

Snake encounters pose significant risks to both human populations and biodiversity. Developing an automated system to detect snakes in their natural habitats can greatly aid in preventing snakebite incidents and conserving endangered snake species. This research presents a novel snake detection system that leverages machine learning techniques to identify and classify snakes from images. The proposed snake detection system follows a multi-stage pipeline. First, a diverse dataset of snake images and videos is collected and annotated with bounding boxes around snake regions. Next, a feature extraction process utilizes deep convolutional neural networks (CNNs) to extract high-level representations from the images. The CNN model used for feature extraction is pre-trained on a large-scale dataset to learn generic visual features, followed by fine-tuning on the specific snake dataset to adapt to snakespecific characteristics.

Technology: Jupiter Notebook anaconda distribution, Convolutional Neural Network algorithm.

Keywords: CNN (Convolutional Neural Networks)

CHAPTER-1

INTRODUCTION

Snake encounters in natural habitats can be hazardous, posing risks to human populations and threatening the biodiversity of various ecosystems In regions where venomous snakes are prevalent, snakebites can lead to severe health consequences and even fatalities. Traditional methods of snake detection, such as manual surveys and human observation, are time-consuming, labour-intensive, and often inadequate for large-scale agricultural operations.

To address these challenges, Advancements in machine learning and computer vision offer a promising solution to automate the process of snake detection and classification. We designed the Snakes Detection System using Machine Learning. The proposed system follows a multi-stage pipeline that involves data collection, feature extraction, and classification, leading to accurate and efficient snake detection. By leveraging deep convolutional neural networks (CNNs) for feature extraction and various machine learning algorithms for classification, the system can learn intricate visual patterns and generalize across different snake species. The potential implications of this research extend beyond snakebite prevention and wildlife conservation, as the adaptability and scalability of the system can find broader applications in ecological research, biodiversity monitoring, and habitat management efforts.

1.1. Aim of the project:

The purpose of our project is to provide early warnings about the presence of snakes, enabling farmers to take preventive measures and mitigate potential risks. The aim of the snake detection using machine learning project is to leverage machine learning to automate the process of snake detection, enabling various practical applications such as wildlife conservation, snake-bite prevention, and public safety.

1.2. Project domain:

The above project primarily falls under the domain of computer vision and machine learning. It involves utilizing computer vision techniques to analyze images or video frames and applying machine learning algorithms to develop a snake detection system. The project's focus is on training a model to accurately identify and distinguish snakes from non-snake objects or backgrounds.

1.3. Significance of the project:

The significance of the snake detection project lies in its potential to positively impact wildlife conservation, snakebite prevention, public safety, research efficiency, educational outreach, and technological advancement. By leveraging machine learning to automate snake detection, the project addresses real-world challenges and contributes to the understanding, conservation, and coexistence with snake species.

1.4. Scope of the project:

The project emphasizes the prevention of snakebite incidents by alerting people about potential snake encounters, especially in snake-prone regions, promoting human safety. The system's scope extends to aiding wildlife conservation efforts by providing valuable data on snake populations, distribution, and behaviour, supporting researchers and conservationists in their conservation strategies .The project focuses on developing an efficient system that can run in real-time on edge devices, such as drones or smartphones, to enable timely snake detection in various environments.

1.5. Literature survey:

A review of literature shows that various Machine learning algorithms have been used for detecting and warning of snakes and insects in agriculture. Some of these systems have used sensors to detect movement, temperature, humidity, and other environmental parameters to determine the presence of snakes. Others have utilized machine learning algorithms to recognize and classify different types of snakes and insects based on their physical characteristics .In addition, some studies have explored the use of wireless communication modules such as Bluetooth to send warning messages to farmers' mobile phones or other devices.

1.6. Project Requirements:

In this project we used Python as the main programming language to develop the model because Python is a versatile programming language with a rich ecosystem of libraries and frameworks, making it well-suited for machine learning tasks. Its simplicity and ease of use allow developers to focus more on the logic and algorithms rather than dealing with complex syntax. It offers popular machine learning libraries such as Tensor Flow, Porch, and sickie-learn. These libraries provide pre-built functions and tools for building and training deep learning models, implementing classification algorithms, and evaluating model performance. Python-based deep learning frameworks like Tensor Flow and Porch are widely used for creating and training convolutional neural networks (CNNs). These frameworks provide highlevel APIs that simplify the process of building complex models for feature extraction .Below mentioned libraries are some additional requirements used in the development pro

CHAPTER-2

PROJECT DEFINITION

Introduction:

Traditional methods of snake detection, such as manual surveys and human observation, are time-consuming, labour intensive, and often inadequate for large-scale agricultural operations.

2.1. EXISTING SYSTEM:

There were several existing systems and approaches for snake detection like Traditional Rule-Based Systems, template Matching Techniques, and Handcrafted Feature Extraction.

Disadvantages:

The existing snake detection systems using machine learning showed progress, they faced challenges related to accuracy, real-time performance, adaptability, and dataset limitations.

2.2. PROPOSED SYSTEM:

The proposed Snake Detection System leverages state-of-the-art machine learning techniques to accurately detect and identify snakes in their natural habitats. The system follows a multi-stage pipeline Data Collection and Annotation, Feature Extraction with CNNs, Snake Species Classification, and Real-Time Deployment on Edge Devices.

Advantages:

The system's optimization for edge devices enables real-time detection, providing timely alerts and warnings about potential snake encounters, which is crucial for ensuring human safety. By utilizing deep learning and fine-tuned CNNs, the proposed system achieves high accuracy in identifying different snake species, minimizing false positives and ensuring reliable snake detection.

2.3. Research Papers:

- http://hdl.handle.net/10125/70760
- https://pubmed.ncbi.nlm.nih.gov/36848781/
- https://www.researchgate.net/publication/309338913_Image_Classification_for_Snake_Species_Using_Machine_Learning_Techniques

CHAPTER-3

PROJECT DESCRIPTION

3.1. Explanation to Machine Learning Technology:

Machine learning is a subfield of artificial intelligence that focuses on the development of algorithms and models that enable computers to learn and make predictions or decisions without being explicitly programmed. It is based on the idea that machines can learn from data, identify patterns, and make informed decisions or predictions.

At the core of machine learning is the concept of a model, which is a mathematical representation that captures patterns and relationships within the data. Models are trained using a learning algorithm, which iteratively adjusts the model's parameters to minimize the difference between its predictions and the actual data.

Machine learning models can be applied to various domains and tasks, such as image and speech recognition, natural language processing, recommendation systems, and anomaly detection. They can process large amounts of data quickly, identify complex patterns, and make predictions or decisions with high accuracy.

3.2 .Need of Snake Detection Systems:

Snake detection systems can contribute to public awareness and education about snake species, their habitats, and safety measures. Improved understanding of snakes can lead to better coexistence with these important members of ecosystems. These systems, particularly those deployed on edge devices in rural or remote areas, can serve as early warning systems, alerting communities and authorities about snake movements, especially during periods of increased snake activity.

3.3. Explanation to CNN Algorithm:

Convolutional Neural Networks (CNNs) are a class of deep learning algorithms primarily used for image recognition and computer vision tasks. They are inspired by the structure and functioning of the human visual system and have shown remarkable performance in various visual perception tasks .CNNs consist of multiple layers, each designed to perform specific operations, and they learn hierarchical representations of features from the input data. Convolutional Layers: These layers use filters or kernels to scan the input image, extracting local patterns or features. The filters slide over the entire image, computing dot products with the pixel values in each local region.

This operation generates feature maps, highlighting distinctive patterns such as edges, corners, and textures .After the convolution operation, an activation function, often Relook (Rectified Linear Unit), is applied to introduce non-linearity to the network. Relook sets all negative values to zero, allowing CNNs to learn complex non-linear relationships in the data. Pooling layers reduce the spatial dimensions of the feature maps, making the network more computationally efficient and reducing the risk of overfitting. Max pooling is a common technique where the maximum value within a local region is retained, discarding other values .After several convolutional and pooling layers, the output is flattened and passed through fully connected layers, also known as dense layers. These layers act as classifiers, making predictions based on the learned features from earlier layers .CNNs are trained through backpropagation, where the model's weights are adjusted based on the difference between predicted and actual output during the training process. The objective is to minimize the prediction error by updating the network's parameters.

3.4. Benefits:

The project's main benefit is its potential to prevent snakebite incidents and protect human populations living or working in snake-prone areas. By providing real-time alerts about snake presence, individuals can take necessary precautions, reducing the risk of snakebite accidents and potentially saving lives.

3.5. Limitations:

The proposed system has some limitations, such as occasional mistakes in identifying snakes correctly. It may not work well in densely covered environments, and its accuracy could vary for different snake species. Deploying the system on small devices might be challenging due to hardware constraints, and updating the model with new data may be necessary over time.

3.6. Project Outcomes:

Accurate detection of snakes, helping to prevent snakebites and keep people safe .Real-time warnings, giving people time to avoid potential snake encounters .Support for wildlife conservation by providing valuable data on snake populations and behaviour.

CHAPTER-4

PROJECT METHODOLOGY

4.1. Design and development:

In designing the Snake Detection System, we chosen to use the CNN (Convolutional Neural Network) algorithm, which is a type of deep learning model known for its excellent performance in image-related tasks. CNNs are particularly well-suited for tasks like object detection and image classification, making them an ideal choice for identifying snakes in images.

To implement the CNN algorithm and develop the Snake Detection System, we are used Jupiter Notebook, which is an interactive coding environment that allows us to write and execute Python code in a web browser. Jupiter Notebook provides a user-friendly interface for writing and running code, making it easier to experiment, iterate, and visualize the results. In Jupiter Notebook, we write Python code to define and train the CNN model on the snake dataset. We are used libraries like Tensor Flow , Numpy, Matplotlib to create and configure the CNN architecture, fine-tune it on the snake images, and train it to learn the features of different snake species.

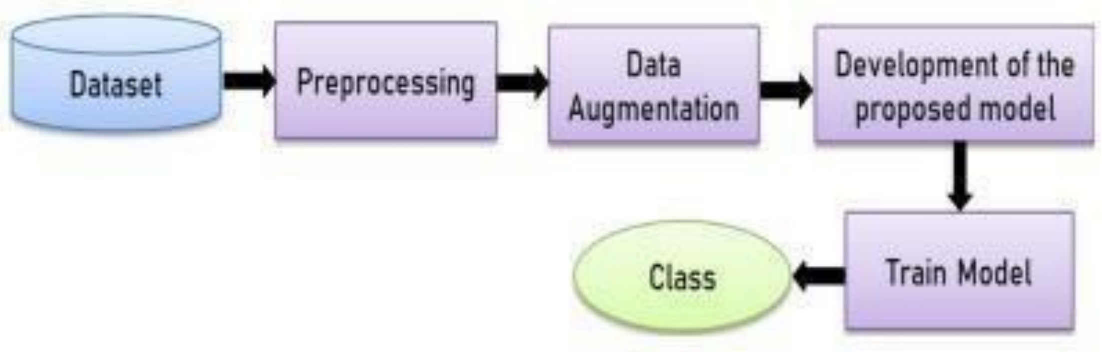


Fig.1. System Flowchart

4.2. About Dataset:

The set of data comes from kaggle.com and contains about 1766 snake images. Per photo was assigned to a category and was divided into groups by the respective class labels such as nonvenomous and venomous. Since reformatting is among the most important phases of data pre-processing, all images are reformatted to 224×224 pixels. Figure 2 and Fig. 3 depicts several photographs from the benchmark dataset. Figures 2 and 3 demonstrate that there are numerous differences between venomous and non-venomous snakes in terms of physical appearances such as head structure, eye shape, skin colour, and so on. The mentioned features will aid our proposed model in learning the distinctions between poisonous and non-poisonous snakes. The set of data has been split into train, validation, and test segments in an appropriate proportion.

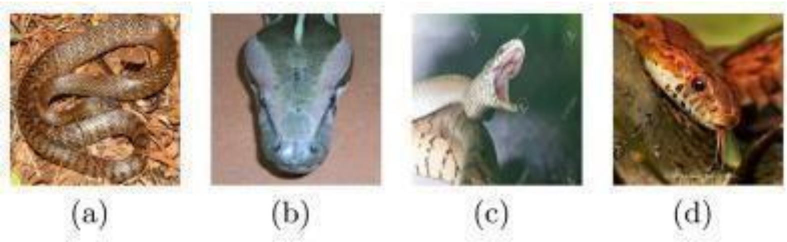


Fig.2.Non-Venomous Snake images

Fig.3.Venomous Snake images

4.3. Code:

from tensorflow.keras.optimizers import * from tensorflow.keras.models

import * from tensorflow.keras.preprocessing.image import * from

tensorflow.keras.callbacks import * from

tensorflow.keras.applications.efficientnet import *import numpy as np

import pandas as pd from pathlib import Path import os.path import

matplotlib.pyplot as plt from IPython.display import Image, display import

matplotlib.cm as cm import tensorflow as tf import os import shutil from

tqdm import tqdm from random import shuffle import cv2 from glob

import glob from tensorflow.keras import backend as K import random

import albumentations as A from sklearn.model_selection import

train_test_split, StratifiedKFold from tensorflow.keras.layers import

* image_dir = Path('../input/snake-dataset-india/Snake Images')

 # Get filepaths and labels filepaths = list(image_dir.glob(r'**/*.jpg'))

labels = list(map(lambda x: os.path.split(os.path.split(x)[0])[1], filepaths))

filepaths = pd.Series(filepaths, name='Filepath').astype(str) labels =

pd.Series(labels, name='Label') **# Concatenate filepaths and labels**

image_df = pd.concat([filepaths, labels], axis=1) image_df.head(5)

0	../input/snake-dataset-india/Snake Images/test...	Venomous
1	../input/snake-dataset-india/Snake Images/test...	Venomous
2	../input/snake-dataset-india/Snake Images/test...	Venomous
3	../input/snake-dataset-india/Snake Images/test...	Venomous
4	../input/snake-dataset-india/Snake Images/test...	Venomous

In [4]:
```
# Shuffle the DataFrame and reset index
image_df = image_df.sample(frac=1).reset_index(drop = True)

# Show the result
image_df.head(5)
```

Out[4]:

	Filepath	Label
0	../input/snake-dataset-india/Snake Images/trai...	Venomous
1	../input/snake-dataset-india/Snake Images/trai...	Venomous
2	../input/snake-dataset-india/Snake Images/test...	Non Venomous
3	../input/snake-dataset-india/Snake Images/trai...	Non Venomous
4	../input/snake-dataset-india/Snake Images/trai...	Non Venomous

In [5]:
```
image_df.shape
```

Out[5]: (1944, 2)

 # Display 20 picture of the dataset with their labels

fig, axes = plt.subplots(nrows=3, ncols=5, figsize=(15, 7),

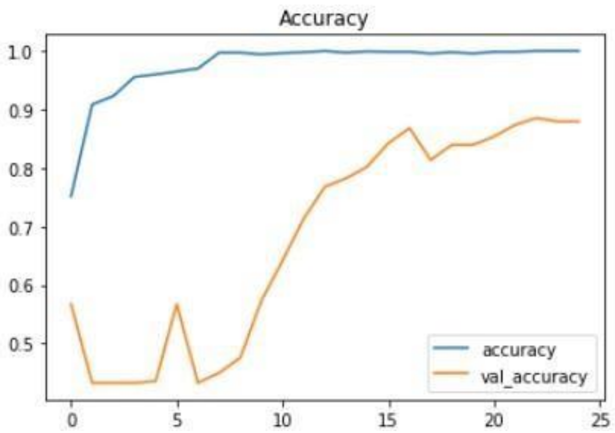
subplot_kw={'xticks': [], 'yticks': []}) for i, ax in

enumerate(axes.flat):

ax.imshow(plt.imread(image_df.Filepath[i]))

ax.set_title(image_df.Label[i]) plt.tight_layout()

plt.show()



 # Separate in train and test data train_df, test_df = train_test_split(image_df,

train_size=0.9, shuffle=True, random_state=1) train_df.shape

(1749, 2)

train_images = train_generator.flow_from_dataframe(

dataframe=train_df,   x_col='Filepath',

y_col='Label',

target_size=(224, 224),

color_mode='rgb',

class_mode='categorical',

batch_size=32,

shuffle=True,   seed=42,

subset='training'

)

val_images = train_generator.flow_from_dataframe(

dataframe=train_df,   x_col='Filepath',

y_col='Label',   target_size=(224, 224),

color_mode='rgb',   class_mode='categorical',

batch_size=32,   shuffle=True,   seed=42,

subset='validation'

)

test_images = test_generator.flow_from_dataframe(

dataframe=test_df,   x_col='Filepath',

y_col='Label',   target_size=(224, 224),

color_mode='rgb',   class_mode='categorical',

batch_size=32,   shuffle=False

)

Found 1400 validated image filenames belonging to 2 classes.

Found 349 validated image filenames belonging to 2 classes.

Found 195 validated image filenames belonging to 2 classes.

pd.DataFrame(history.history)[['accuracy','val_accuracy']].plot()

plt.title("Accuracy") plt.show() results =

model.evaluate(test_images, verbose=0) print("   Test Loss:

{:.5f}".format(results[0])) print("Test Accuracy:

{:.2f}%".format(results[1] * 100))

Test Loss: 0.41493
Test Accuracy: 88.72%

 # Predict the label of the test_images

pred = model.predict(test_images) pred

= np.argmax(pred,axis=1)

 # Map the label labels =

(train_images.class_indices) labels =

dict((v,k) for k,v in labels.items()) pred =

[labels[k] for k in pred] **# Display the**

result print(f'The first 5 predictions:

{pred[:5]}')

The first 5 predictions: ['Venomous', 'Venomous', 'Non Venomous', 'Venomous', 'Venomous']

from sklearn.metrics import classification_report

y_test = list(test_df.Label)

print(classification_report(y_test, pred))

	precision	recall	f1-score	support
Non Venomous	0.81	0.90	0.85	70
Venomous	0.94	0.88	0.91	125
accuracy			0.89	195
macro avg	0.87	0.89	0.88	195
weighted avg	0.89	0.89	0.89	195

 # Display 15 picture of the dataset with their
labels fig, axes = plt.subplots(nrows=3,
ncols=5, figsize=(15, 7),
subplot_kw={'xticks': [], 'yticks': []}) for i, ax
in enumerate(axes.flat):
ax.imshow(plt.imread(test_df.Filepath.iloc[i]))

ax.set_title(f"True: {test_df.Label.iloc[i]}\nPredicted: {pred[i]}")

plt.tight_layout() plt.show()

4.4. OUTPUT:


True: Venomous
Predicted: Venomous
True: Venomous
Predicted: Venomous
True: Venomous
Predicted: Non Venomous
True: Venomous
Predicted: Venomous
True: Venomous
Predicted: Venomous
True: Venomous
Predicted: Venomous
True: Non Venomous
Predicted: Non Venomous
True: Venomous
Predicted: Venomous
True: Non Venomous
Predicted: Non Venomous
True: Venomous
Predicted: Venomous
True: Venomous
Predicted: Venomous
True: Non Venomous
Predicted: Non Venomous
True: Venomous
Predicted: Venomous
True: Non Venomous
Predicted: Non Venomous
True: Venomous
Predicted: Venomous

CHAPTER-5

CONCLUSION

In conclusion, the proposed system is an innovative and socially relevant project aimed at protecting farmers from snake and insect bites. The project outcome is a smart system that can detect and warn of danger in real-time, allowing farmers to take appropriate precautions to protect themselves and their crops. This system can have a significant impact on the health and livelihoods of farmers, especially in areas where snakes and insect-related injuries are prevalent.

5.1. Future scope:

It is not only limited to snakes which are poisonous and non-poisonous but also can be implemented for dangerous insects and harmful creatures. The hardware can be developed such that it can sustain any environmental conditions darkness and can be made more flexible and adaptable. This system can face different challenges like water-Resistance wisely. It can be designed to given an immediate call to the specified number if bite occurs.

5.2. References:

- https://www.researchgate.net
- https://www.elprocus.com
- https://www.sparkfun.com
- https://maxbotix.com
- https://www.fierceelectronics.com