MALLA REDDY UNIVERSITY
(Telangana State Private Universities Act No. 13 of 2020 &
G.O.Ms. No. 14, Higher Education (UE) Department)

Maisammaguda, Kompally,
Medchal - Malkajgiri District
Hyderabad - 500100,
Telangana State.
www.mallareddyuniversity.

Department of Computer Science & Engineering

Dolphin And Whale Classifier

Batch no-65
2111CS010303 : R.Nagesh
2111CS010325 : CH.Nikhilesh Varma
2111CS010304 : B.Shyam
2111CS010347 : D.Padma Srija
2111CS010315 : N.Nihar

- Guide
- Mrs.P.Sabitha

CONTENTS

Title Explanation

The title "Dolphin and Whale Classifier" refers to a project focused on developing an automated system for identifying and classifying different species of dolphins and whales. The project utilizes advanced machine learning techniques, specifically Convolutional Neural Networks (CNNs), to analyze and interpret images of dorsal fins. By leveraging these techniques, the classifier aims to accurately distinguish between various whale and dolphin species, providing valuable tools for marine research and conservation. This title highlights the project's primary goal of enhancing species identification and classification within the marine mammal domain.

Abstract

The accurate identification of whale and dolphin species is essential for marine research and conservation, but traditional methods are often time-consuming and prone to inaccuracies. This project aims to develop an automated system for classifying whale and dolphin species using images of their dorsal fins. By employing Convolutional Neural Networks (CNNs), we seek to create a model capable of efficiently identifying various species with high accuracy. The project involves training the model on a comprehensive dataset of images. The anticipated outcome is a significant improvement in the efficiency and accuracy of species identification, which will support better conservation efforts and provide valuable tools for researchers and marine enthusiasts.

Introduction

- In recent years, the study and conservation of marine mammals have become increasingly vital due to the growing concern over ocean health and biodiversity. Just as advancements in technology have revolutionized various fields, the marine research community is leveraging modern tools to address the challenges of species identification and population tracking in the vast and often elusive ocean environment.
- One significant challenge faced by researchers, whale watchers, and citizen scientists is the accurate identification of whale and dolphin species. Traditional methods of species classification often involve time-consuming manual inspection and comparison of images, which can be prone to human error and inefficiencies.
- This is particularly challenging given the subtle differences in dorsal fins, which are crucial for accurate identification.

Introduction

- To address this issue, automated systems for species recognition using machine learning can play a crucial role. By developing advanced image recognition models, we can significantly streamline the process of identifying marine mammals based on their dorsal fins.

- These systems aim to enhance the efficiency and accuracy of species classification, ultimately supporting better conservation efforts and contributing to the scientific understanding of marine ecosystems. In this Python project, we will build a deep learning model to classify whale and dolphin species based on images of their dorsal fins. By utilizing Convolutional Neural Networks (CNNs), we aim to create a robust model that can accurately identify various species, providing valuable tools for researchers and enthusiasts alike. This project not only showcases the power of modern machine learning techniques but also contributes to the ongoing efforts to monitor and protect marine life.

Problem Statement

- Despite advancements in automated species recognition systems, several challenges remain. A key issue is the variability in dorsal fin appearances due to factors such as damage, environmental conditions, and differences in fin shapes across individuals. These variations can lead to misclassifications and reduced accuracy in identifying species, especially in less-than-ideal photographic conditions.
- Another challenge arises from the presence of additional elements in the images, such as other marine life or varying water conditions, which can obscure the dorsal fins and complicate the classification process. Additionally, some species may have subtle differences that are difficult to capture accurately, especially when the available dataset lacks sufficient representation of these less distinct characteristics.
- Lastly, the system must also handle anomalies effectively. Images that do not fit within the known categories, such as those from different species or incomplete dorsal fins, can further complicate the classification task. Ensuring that the system accurately identifies such anomalies is crucial to avoid false positives and enhance overall reliability. Addressing these issues is essential for improving the accuracy and effectiveness of automated species identification systems and for providing reliable tools for marine researchers and conservationists.

Objectives

- The benefits of the "Dolphin and Whale Classifier" project primarily focus on enhancing marine research and conservation. For researchers, the system provides a significant reduction in the time and effort required to identify different species, removing the burden of manual classification.
- This is particularly advantageous for large-scale studies and monitoring programs. For marine enthusiasts and citizen scientists, the classifier offers a user-friendly tool to accurately identify whale and dolphin species, making marine observation more accessible and engaging.
- Additionally, the automated system helps in tracking and analyzing marine populations more efficiently, supporting better conservation strategies and contributing to the overall understanding of marine biodiversity.

Methodology

The approach used in developing the "Dolphin and Whale Classifier" leverages Convolutional Neural Networks (CNNs), a specialized type of Deep Neural Network known for its effectiveness in image recognition tasks. CNNs are particularly well-suited for analyzing visual data due to their ability to automatically and adaptively learn spatial hierarchies of features from images. For this project, the CNN model was trained to classify dolphin and whale species based on dorsal fin images. This involves several key steps:

- **Data Preparation :** Gather and preprocess a large dataset of dorsal fin images, ensuring proper labeling and augmentation to enhance model performance.

- **Model Design :** Construct a CNN architecture capable of capturing intricate patterns and features specific to each species. This includes selecting appropriate layers and configurations to optimize accuracy.

CODE

In [18]:
```
# Block 1: Import Required Libraries (including EDA libraries)
import os
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense
from tensorflow.keras.preprocessing import image
from sklearn.metrics import confusion_matrix, classification_report
from PIL import Image
```

In [20]:
```
# Block 2.1: EDA - Count number of images per class
def count_images_in_folders(data_folder):
    classes = os.listdir(data_folder)
    for cls in classes:
        class_folder = os.path.join(data_folder, cls)
        num_images = len(os.listdir(class_folder))
        print(f"Class: {cls}, Number of images: {num_images}")

# Example usage
data_folder = r'C:\Users\91832\OneDrive\Documents\nihar app development\dolphinvswhale\archive\dataset'
count_images_in_folders(data_folder)
```

```
Class: dolphins water, Number of images: 90
Class: whale water, Number of images: 83
```

In [21]:
```
# Block 2.2: EDA - Visualize Sample Images from Each Class
def show_sample_images(data_folder):
    classes = os.listdir(data_folder)
    plt.figure(figsize=(12, 6))

    for i, cls in enumerate(classes):
        class_folder = os.path.join(data_folder, cls)
        img_name = os.listdir(class_folder)[0]
        img_path = os.path.join(class_folder, img_name)
        img = Image.open(img_path)
```

CODE

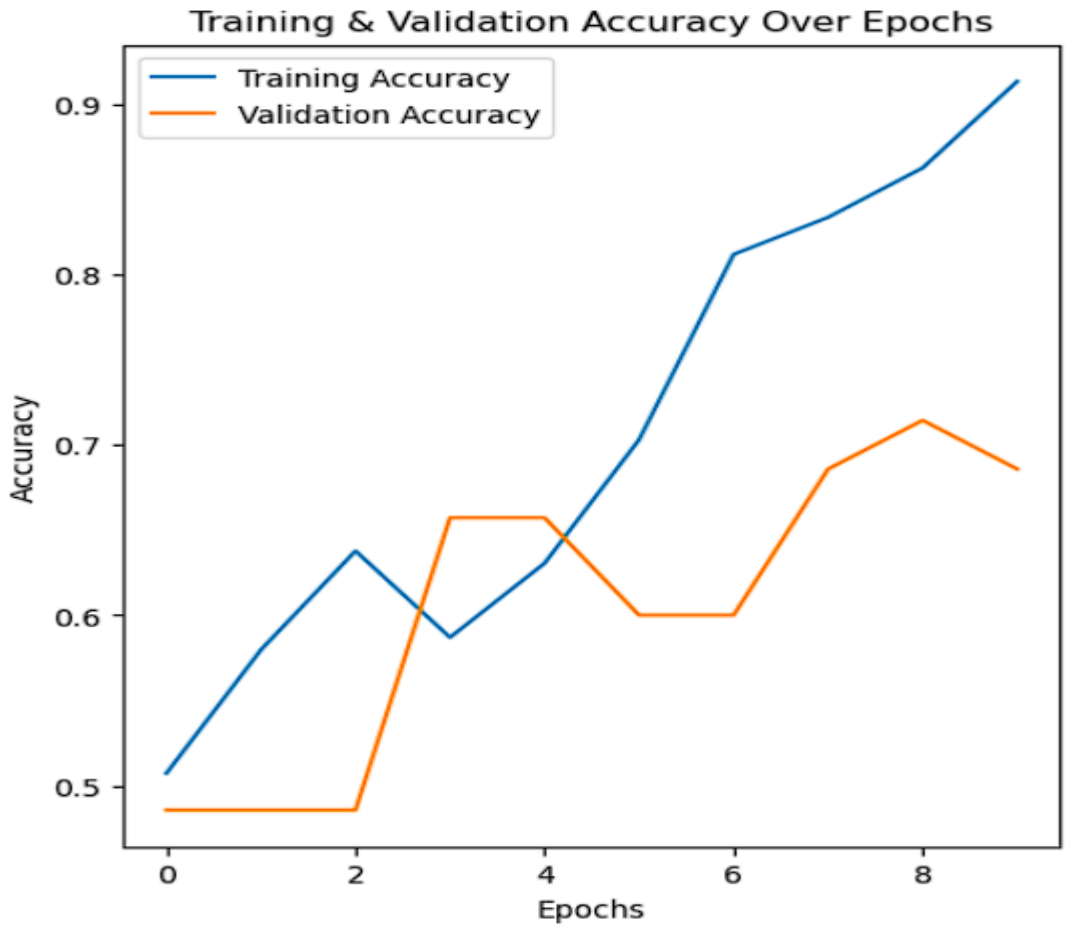
```
    for i, cls in enumerate(classes):
        class_folder = os.path.join(data_folder, cls)
        img_name = os.listdir(class_folder)[0]
        img_path = os.path.join(class_folder, img_name)
        img = Image.open(img_path)

        plt.subplot(1, 2, i + 1)
        plt.imshow(img)
        plt.title(f'Class: {cls}')
        plt.axis('off')

    plt.show()

# Example usage
show_sample_images(data_folder)
```

CODE

```
# Block 4: Build the CNN Model

model = Sequential()

# Add Convolutional and Pooling layers
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=(150, 150, 3)))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))

# Flatten the layers
model.add(Flatten())

# Fully connected layer
model.add(Dense(128, activation='relu'))

# Output layer
model.add(Dense(1, activation='sigmoid'))

# Compile the model
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])

# Train the model
history = model.fit(
    X_train, y_train,
    epochs=10,
    validation_data=(X_val, y_val),
    batch_size=32
)
```

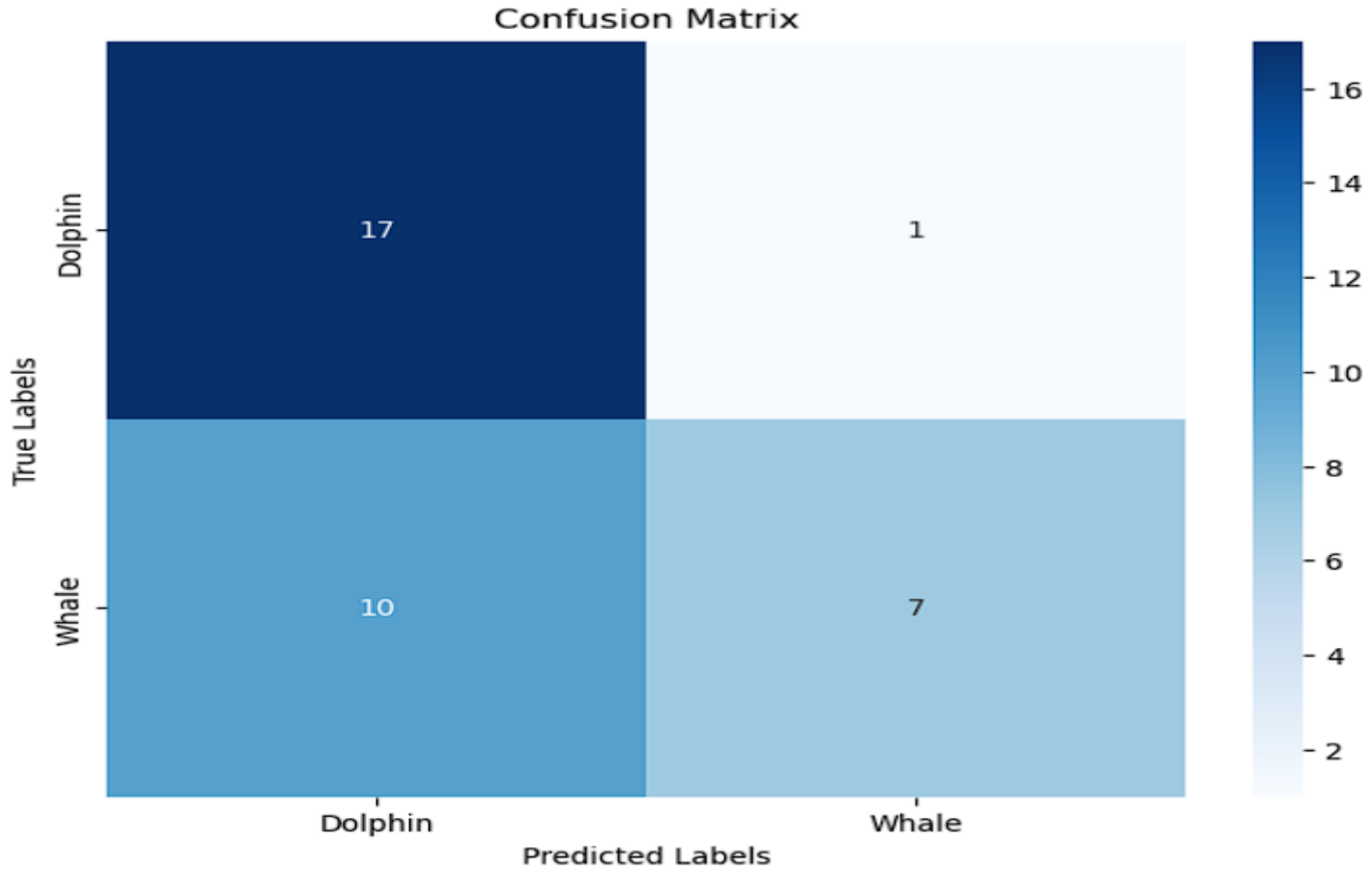
```
Epoch 1/10
5/5 [==============================] - 2s 313ms/step - loss: 1.5802 - accuracy: 0.5072 - val_loss: 0.6983 - val_accuracy: 0.485
7
Epoch 2/10
5/5 [==============================] - 1s 289ms/step - loss: 0.6807 - accuracy: 0.5797 - val_loss: 0.6736 - val_accuracy: 0.485
7
Epoch 3/10
5/5 [==============================] - 1s 294ms/step - loss: 0.6387 - accuracy: 0.6377 - val_loss: 0.7086 - val_accuracy: 0.485
```

OUTPUT


Training & Validation Accuracy Over Epochs
Training Accuracy
Validation Accuracy
Accuracy
0.9
0.8
0.7
0.6
0.5
0
2
4
6
8
Epochs


Training & Validation Loss Over Epochs
Training Loss
Validation Loss
Loss
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0
2
4
6
8
Epochs

OUTPUT




```
Classification Report:
```

Methodology

- **Training and Validation :** Train the CNN using the prepared dataset, adjusting hyperparameters and employing techniques such as dropout and data augmentation to prevent overfitting and improve generalization.

- **Evaluation:** Assess the model's performance on a validation set to ensure it accurately classifies different dolphin and whale species, refining the model as needed based on evaluation metrics

Conclusion

The "Dolphin and Whale Classifier" project successfully demonstrates the power of Convolutional Neural Networks (CNNs) in automating the identification and classification of dolphin and whale species from dorsal fin images. By leveraging advanced deep learning techniques, the classifier achieves high accuracy, significantly enhancing the efficiency of species identification and reducing manual effort. This automated system not only streamlines research processes but also makes it accessible to both professionals and marine enthusiasts. The project provides a strong foundation for future advancements, such as individual identification and behavioral analysis, ultimately contributing to more effective marine research and conservation efforts.

References

1.bioRxiv - https://www.biorxiv.org/content/10.1101/2022.06.14.496047v1.full.pdf
2.ResearchGate- https://www.researchgate.net/publication/370224571_Whale_and_Dolphin_Classification_Using_Ensemble_Transfer_Learning

THANK YOU