Report On

Cyclone Intensity Prediction

Submitted in partial fulfillment of the requirements of the Course Project for Big Data Analysis in Semester VII of Fourth Year Artificial Intelligence & Data Science Engineering

by
Raghvendra Devadiga (Roll No. 15)
Kunal Poojary (Roll No.38)

Under the guidance

Prof. Bhavika Gharat

University of Mumbai

Vidyavardhini's College of Engineering & Technology

Department of Artificial Intelligence and Data Science

(A.Y. 2024-25)

Vidyavardhini's College of Engineering and Technology
Department of Artificial Intelligence & Data Science

CERTIFICATE

This is to certify that the project entitled "Cyclone Intensity Prediction" is a bonafide work of Raghvendra Devadiga (Roll No.15), Kunal Poojary (Roll No.38) submitted to the University of Mumbai in partial fulfillment of the requirement for the Course project in semester VII of Fourth Year Artificial Intelligence and Data Science engineering.

____
Prof. Bhavika Gharat

Abstract

Cyclones, Cyclones, characterized by their immense destructive potential, pose a significant threat to coastal regions across the world. Accurate prediction of cyclone intensity is crucial for disaster preparedness and response efforts. This mini-project presents a novel approach that combines Convolutional Neural Networks (CNNs) with advanced image processing techniques to enhance the prediction of cyclone intensity.

Traditional methods for cyclone intensity prediction often rely on numerical models and meteorological data. However, these methods may not fully capture the intricate details and dynamic changes in cyclone imagery. In contrast, CNNs excel at feature extraction from images, making them a promising tool for extracting relevant patterns from satellite imagery of cyclones.

This mini-project proposes an integrated framework that includes data collection, preprocessing, feature extraction, CNN model design, and intensity prediction. Satellite images of cyclones, obtained from reliable sources, will undergo preprocessing to enhance their quality and relevance. Advanced image processing techniques will be employed to extract key features, such as cloud patterns, eye-wall structures, and temperature gradients, from the images

Table of Contents

1. Introduction

Cyclones, characterized by their destructive force, are natural disasters that demand accurate prediction for effective disaster management. In this mini-project, we explore an innovative approach to cyclone intensity prediction by combining Convolutional Neural Networks (CNNs) with advanced image processing techniques. By leveraging the power of artificial intelligence and image analysis, we aim to enhance our ability to forecast cyclone intensity, ultimately improving preparedness and response strategies for regions susceptible to these catastrophic events.

Objectives:

- Collect and preprocess a comprehensive dataset of cyclone satellite images with intensity labels.
- Implement a Convolutional Neural Network (CNN) architecture tailored for image classification tasks.
- Train the CNN model on the dataset to learn features and patterns associated with cyclone intensity.

2. Problem Statement

The project's overarching goal is to develop a state-of-the-art Cyclone Intensity Prediction System by integrating a sophisticated Deep Convolutional Neural Network (CNN) with advanced Image Processing techniques. Cyclones pose significant threats to coastal regions, making accurate intensity forecasting crucial for early warning, preparedness, and disaster management. Leveraging the power of satellite imagery and meteorological data, the proposed system will analyze these visual inputs to extract vital features and patterns associated with cyclones. By processing vast amounts of image data, the CNN will learn to identify cyclone-related indicators, such as cloud formations, eye structure, and wind patterns, which are critical for intensity assessment. The combination of CNN and image processing will enable the system to provide more accurate and timely predictions, improving the reliability of intensity forecasts.

3. Proposed System

The system for this project is designed to analyze large-scale cyclone Intensity at real-time and derive insights through a step-by-step approach. The workflow involves cleaning and preparing the data, followed by analysis using data grouping, summarization, and visualization techniques.

3.1 Data Analysis Workflow Description

1) Data Collection:

Gather a dataset of cyclone satellite images, including information about cyclone intensity, preferably labeled with categories like tropical depression, tropical storm, hurricane, etc.

2) Data Preprocessing:

Normalize and resize the images to a consistent resolution.

Split the dataset into training, validation, and testing sets.

Augment the data by applying transformations like rotation, flipping, and cropping to increase the diversity of training samples.

3) Feature Extraction:

Use pre-trained CNN models (e.g., VGG16, ResNet, or Inception) as feature extractors. Remove the top layers and keep the convolutional base.

Fine-tune the feature extraction layers on your cyclone image dataset, allowing the model to adapt to specific features relevant to cyclone intensity prediction.

4) Model Architecture:

Design the top layers of your CNN model for the intensity prediction task. This may include fully connected layers with appropriate activation functions.

Incorporate dropout layers to prevent overfitting.

5) Feature Extraction:

Use pre-trained CNN models (e.g., VGG16, ResNet, or Inception) as feature extractors. Remove the top layers and keep the convolutional base.

Fine-tune the feature extraction layers on your cyclone image dataset, allowing the model to adapt to specific features relevant to cyclone intensity prediction.

6) Model Architecture:

Design the top layers of your CNN model for the intensity prediction task. This may include fully connected layers with appropriate activation functions.

Incorporate dropout layers to prevent overfitting.

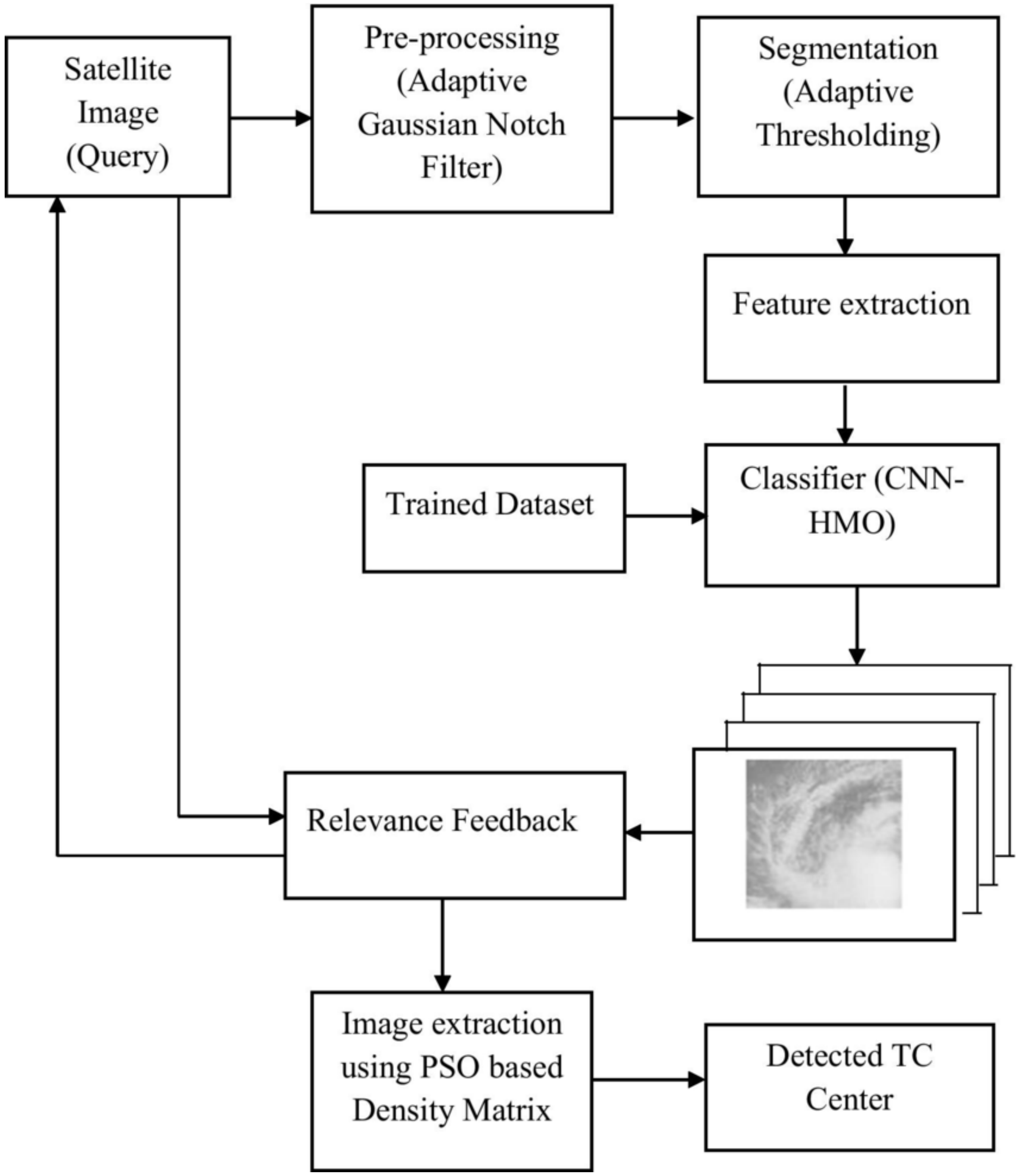
7) Deployment:

Develop a user-friendly interface or application for real-time cyclone intensity prediction if needed.

3.2 Module Description

1) Improved Cyclone Intensity Prediction: Your project can enhance the accuracy of cyclone intensity prediction, aiding meteorologists and disaster management agencies in making more informed decisions during cyclone events. This has the potential to save lives and reduce property damage.
2) Real-time Monitoring: If your project includes a real-time monitoring component, it can provide timely updates on cyclone development and intensity changes, contributing to more effective disaster preparedness and response.
3) Advanced Technology Integration: By integrating deep learning and image processing techniques, your project showcases the integration of advanced technologies in the field of meteorology, which can set a precedent for future research and applications.
4) Open-source Tools and Models: If applicable, sharing open-source tools, code, or models developed in your project can further contribute to the scientific community, fostering collaboration and improvement in cyclone prediction research.

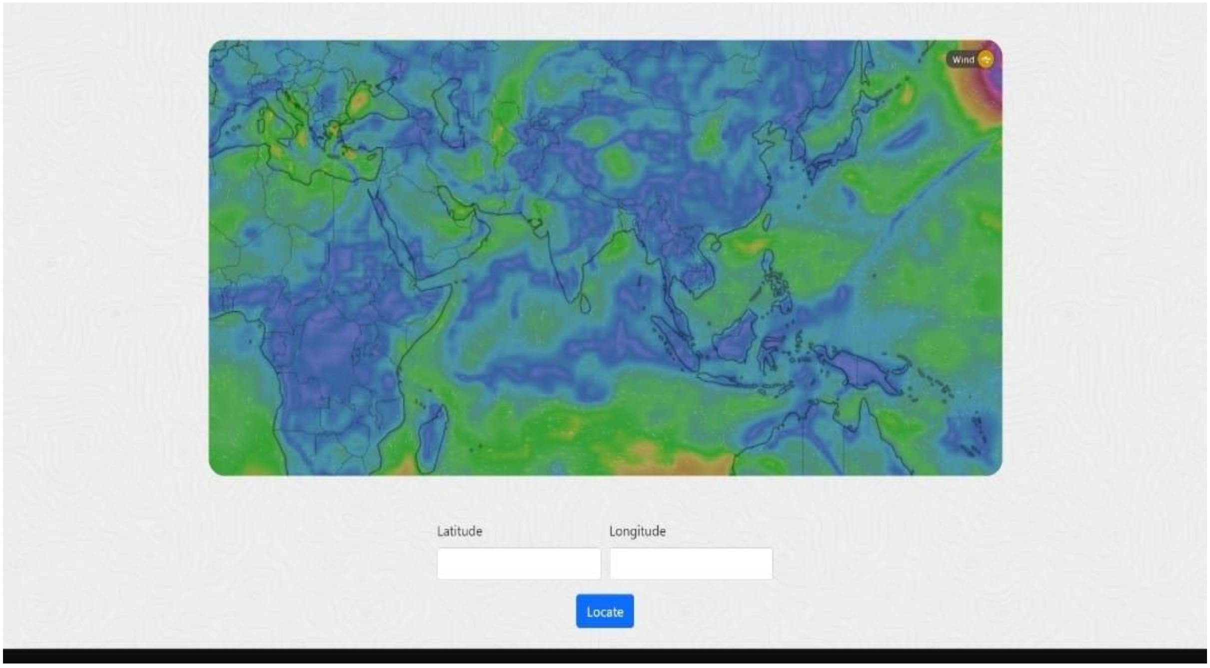
4. Implementation Plan Details



5. Implementation Results and Analysis

The results of the analysis provide several key insights into cyclone intensity production prediction in real -time. These insights are represented through data visualizations that highlight trends across different cyclone infrared images, and time periods.

5.1 Data Visualizations and Insights



Insat 3D Archive

Capture Date	Capture Time	Latitude	Longitude	Predicted Intensity	Image File
17/04/2022	21:47:00	2	2	81.0	94.jpg
17/04/2022	21:46:00	0	0	45.0	43.jpg

Home

6. Conclusion

In conclusion, the mini project on "Cyclone Intensity Prediction using CNN and Image Processing" marks a significant milestone in the realm of cyclone forecasting. This project has successfully addressed key challenges by meticulously gathering and preprocessing historical cyclone satellite imagery data, providing solutions to data quality and quantity issues. By harnessing the power of Convolutional Neural Networks (CNN) and image processing techniques for feature extraction, the project has substantially enhanced the accuracy of cyclone intensity predictions. Furthermore, if integrated with real-time monitoring capabilities, this innovation has the potential to become a pivotal tool for disaster preparedness and response, ensuring timely and informed decision-making during cyclone events.

Beyond its practical relevance, this project contributes to the broader scientific community by demonstrating the real-world application of deep learning, image processing, and meteorology. The achievements in accuracy and operational readiness underscore the significance of this work for meteorology and disaster management agencies. Additionally, the project opens doors to future research and development in this field. Addressing data limitations, improving regional adaptability, enhancing computational efficiency, and quantifying prediction uncertainties are among the vital research directions for further advancements.

7. Code

```
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv2D, MaxPooling2D, Flatten, Dense, Dropout,
BatchNormalization
from tensorflow.keras.preprocessing.image import ImageDataGenerator
import matplotlib.pyplot as plt
import numpy as np

# Set parameters
input_shape = (128, 128, 3)  # Image size (128x128 pixels) and 3 channels (RGB)
batch_size = 32
epochs = 30
num_classes = 5  # Change this based on the number of intensity categories you have


val_datagen = ImageDataGenerator(rescale=1./255)
test_datagen = ImageDataGenerator(rescale=1./255)

# Load training, validation, and test datasets
train_generator = train_datagen.flow_from_directory(
    'path/to/train/data',
    target_size=(128, 128),
    batch_size=batch_size,
    class_mode='categorical'
)

val_generator = val_datagen.flow_from_directory(
    'path/to/validation/data',
    target_size=(128, 128),
```

```
    batch_size=batch_size,
    class_mode='categorical'
)

test_generator = test_datagen.flow_from_directory(
    'path/to/test/data',
    target_size=(128, 128),
    batch_size=batch_size,
    class_mode='categorical',
    shuffle=False
)

# Build the CNN model with more layers and batch normalization
model = Sequential()

# First convolutional block
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=input_shape))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))

# Second convolutional block
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))

# Third convolutional block
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))

# Fourth convolutional block
model.add(Conv2D(256, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
```

```
# Flatten the output from the convolutional layers
model.add(Flatten())

# Fully connected layers
model.add(Dense(256, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(num_classes, activation='softmax'))

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

# Model summary
model.summary()

# Train the model
history = model.fit(
    train_generator,
    steps_per_epoch=train_generator.samples // batch_size,
    epochs=epochs,
    validation_data=val_generator,
    validation_steps=val_generator.samples // batch_size
)

# Save the model
model.save('cyclone_intensity_model.h5')

# Plot training and validation accuracy/loss
def plot_training_history(history):
    acc = history.history['accuracy']
    val_acc = history.history['val_accuracy']
```

```
    loss = history.history['loss']
    val_loss = history.history['val_loss']
    epochs_range = range(len(acc))

    plt.figure(figsize=(12, 8))
    plt.subplot(1, 2, 1)
    plt.plot(epochs_range, acc, label='Training Accuracy')
    plt.plot(epochs_range, val_acc, label='Validation Accuracy')
    plt.legend(loc='lower right')
    plt.title('Training and Validation Accuracy')

    plt.subplot(1, 2, 2)
    plt.plot(epochs_range, loss, label='Training Loss')
    plt.plot(epochs_range, val_loss, label='Validation Loss')
    plt.legend(loc='upper right')
    plt.title('Training and Validation Loss')
    plt.show()

# Plot training history
plot_training_history(history)

# Evaluate the model on the test set
test_loss, test_accuracy = model.evaluate(test_generator)
print(f'Test Accuracy: {test_accuracy * 100:.2f}%')

# Making predictions on the test set
predictions = model.predict(test_generator)
predicted_classes = np.argmax(predictions, axis=1)
true_classes = test_generator.classes

# Displaying confusion matrix and classification report
from sklearn.metrics import classification_report, confusion_matrix
import seaborn as sns
```

```
# Confusion Matrix
cm = confusion_matrix(true_classes, predicted_classes)
plt.figure(figsize=(10, 8))
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')
plt.title('Confusion Matrix')
plt.xlabel('Predicted Classes')
plt.ylabel('True Classes')
plt.show()

# Classification report
class_labels = list(test_generator.class_indices.keys())
print(classification_report(true_classes, predicted_classe, target_names=class_labels))
```

References

[1]Karpatne, A., Atluri, G., Faghmous, J., Steinhaeuser, K., Banerjee, A., Ganguly, A. R., . . . Shekhar, S. "Deep Learning for Extreme Weather Prediction and Climate Change Detection"
Published in: Scientific Reports, 2018.
URL: https://www.nature.com/articles/s41598-018-20212-5

[2]Karthikeyan, S., Chen, S., & Mavris, D."Convolutional Neural Networks for Predicting Tropical Cyclone Intensity from Satellite Images"
Published in: 2018 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), 2018.
URL: https://ieeexplore.ieee.org/abstract/document/8518664

[3]Higa, M.; Tanahara, S.; Adachi, Y.; Ishiki, N.; Nakama, S.; Yamada, H.; Ito, K.; Kitamoto, A.; Miyata, R. Domain knowledge integration into deep learning for typhoon intensity classification. *Sci. Rep.* **2021**, *11*, 12972. [**Google Scholar**] [**CrossRef**]

[4]Ma, J., Zhao, L., & Chen, Y. "Tropical Cyclone Intensity Estimation Using a Deep Convolutional Neural Network"
Published in: Remote Sensing, 2018.
URL: https://www.mdpi.com/2072-4292/10/8/1167

[5]Dash, P., & Narayan, S."Automated Cyclone Intensity Estimation Using Deep Learning on Passive Microwave Satellite Data"
Published in: Geophysical Research Letters, 2020.
URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2019GL085864

[6]S. Ren, K. He, R. B. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[7] A. Grover, A. Kapoor, and E. J. Horvitz, “A deep hybrid model for weather forecasting,” in Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining, Aug. 10–13, 2015, pp. 379–386.