PROJECT DOCUMENTATION

Enchanted Wings: Marvels of Butterfly Species

TEAM ID : LTVIP2025TMID32471

CONTENTS

Enchanted Wings: Marvels of Butterfly Species

TEAM ID - LTVIP2025TMID32471

1. Introduction

Project Title:
Enchanted Wings: Marvels of Butterfly Species

Team Members:

- Palla Bhavana(Team Leader)
- Mundru Udaysai
- Narapureddy Chandrakanth
- Munta Navadeep Venkata Sai Kumar

2. Project Overview

Purpose:
Enchanted Wings is an AI-powered system designed to automate the identification of butterfly species using advanced computer vision techniques and transfer learning. Leveraging pre-trained convolutional neural networks (CNNs), the platform enhances biodiversity research, ecological monitoring, and citizen science by providing fast and accurate species classification directly from images. The goal is to aid conservation efforts, facilitate ecological studies, and foster public engagement in butterfly biodiversity.

Key Features:

- **Image Preprocessing:** Loads and prepares butterfly images into training, validation, and test sets.
- **Transfer Learning:** Utilizes pre-trained CNN architectures (e.g., ResNet, EfficientNet) for feature extraction.
- **Model Training:** Trains a butterfly classification model optimized for high accuracy and efficiency.
- **Model Evaluation:** Provides performance metrics including accuracy, confusion matrices, and class-wise precision.
- **User Interface:** A user-friendly dashboard for uploading butterfly images and viewing classification results.
- **Scenario Integration:** Tailored applications for biodiversity monitoring, ecological research, and citizen science education.
- **Educational Info:** Displays species-specific details, facts, and conservation status.

3. Architecture

Enchanted Wings follows a modern client-server architecture:

Frontend:

- Built with Streamlit for an interactive, web-based dashboard.
- Allows users to upload butterfly images and view classification results.
- Communicates with the backend via HTTP API calls.

Backend:

- Developed using FastAPI, exposing RESTful endpoints for all functionalities.
- Handles image uploads, model predictions, and retrieval of species information.
- Implements logic for preprocessing, classification, and educational data retrieval.

Machine Learning Model:

- Employs transfer learning using pre-trained CNNs (e.g., ResNet, EfficientNet) for feature extraction and classification.
- Trained on a dataset of 6,499 butterfly images across 75 species.
- Designed for efficient, accurate predictions.

4. Setup Instructions

Prerequisites

- Python 3.8+
- pip (Python package manager)
- Flask (or any Python web framework you're using)
- TensorFlow or Keras (for loading vgg16_model.h5)
- HTML/CSS/JS knowledge for frontend edits

4.1 Installation

1. **Clone the Repository**

```
git clone https://github.com/yourusername/smartbridge.git
cd smartbridge
```

2. **Create Virtual Environment**

   python -m venv venv

   Activate it:

   venv\Scripts\activate

3. Install Requirements

```
pip install flask tensorflow keras
```

4. Run Your Application

```
cd ../frontend
pip install -r requirements.txt
```

5. **Run your Application**

   python app.py

5. Folder Structure

```
SMARTBRIDGE/
│
├── js/
│   └── script.js
│
├── static/
│   ├── css/
│   │   └── style.css
│   └── images/
│
├── templates/
│   ├── index.html
│   ├── input.html
```

```
│   └── output.html
│
├── app.py
└── vgg16_model.h5
```

6. Running the Application

Start the Flask Application:

```
python app.py
```

Access the Application:

- **Flask Development Server** starts (usually on port 5000).
- Visit http://localhost:5000 in your web browser.

7. Model Documentation & Interaction Flow

- **Model:** Pre-trained CNN model fine-tuned for butterfly species classification (e.g., ResNet, EfficientNet).
- **Interaction:**
  - The frontend sends an image file to the backend via HTTP POST.
  - The backend preprocesses the image and runs it through the ML model.
  - The backend retrieves educational info for the predicted species.
  - Results are returned to the frontend for display.
- **Example API Request:**

```
POST /predict
Content-Type: multipart/form-data


- file: butterfly_image.jpg
```

- **Example API Response:**

```
{
 "status": "success",
 "data": {
  "species_name": "Papilio demoleus",
  "confidence": 0.92,
  "info": {
   "description": "Common Lime Butterfly found across Asia.",
   "conservation_status": "Least Concern"
```

```
    }
  }
}
```

8. User Interface

Clean, interactive dashboardwith:

- Image upload feature.
- Display of predicted species name and confidence.
- Educational section with facts and conservation status.
- History panel (optional) for previous predictions.

9. Testing

Testing Strategy:

- **Backend:**
    - Unit tests for API endpoints using pytest.
    - Tests for preprocessing and prediction pipeline.
- **Model:**
    - Manual testing of various butterfly species images.
    - Evaluation using metrics like accuracy, confusion matrix.
- **Frontend:**
    - Manual testing for layout, responsiveness, and usability.

10. Known Issues

Model Accuracy Limitations:

- The model may misclassify visually similar species.

Processing Time:

- High-resolution images might increase inference time.

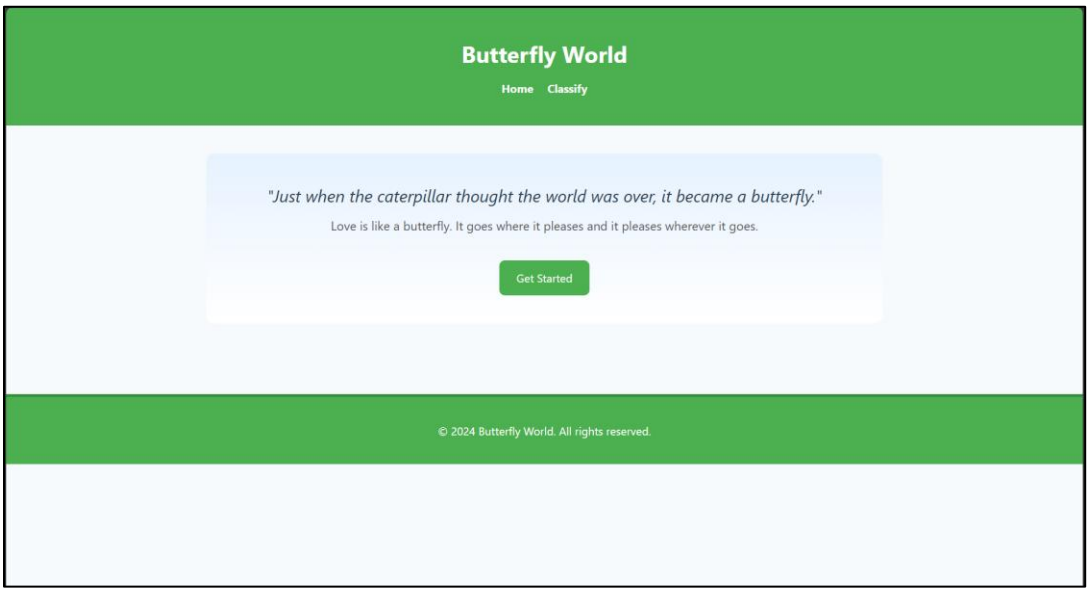
Class Imbalance:

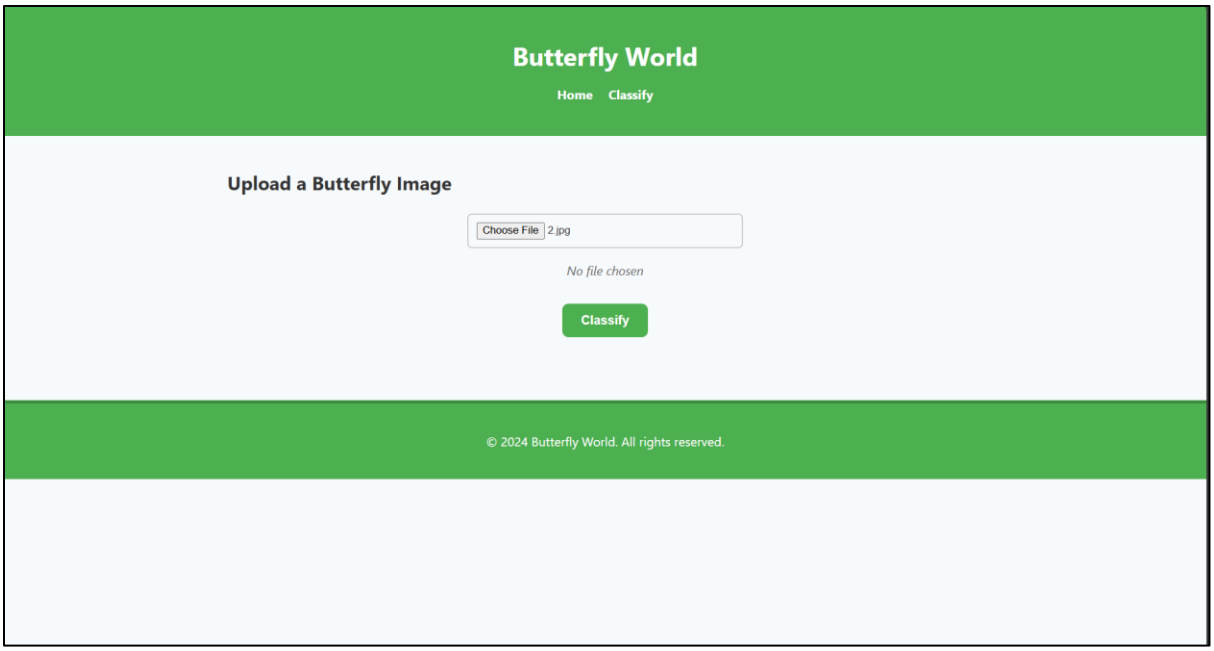
- Some butterfly species might be underrepresented in the training dataset.

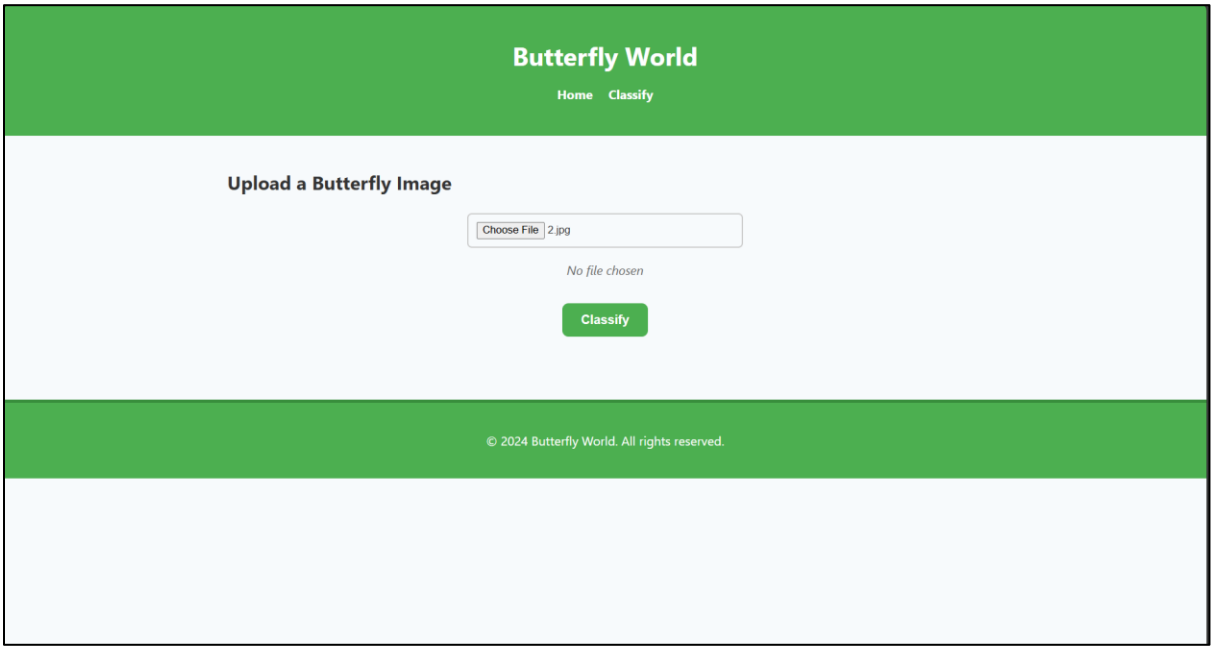
11. Future Enhancements

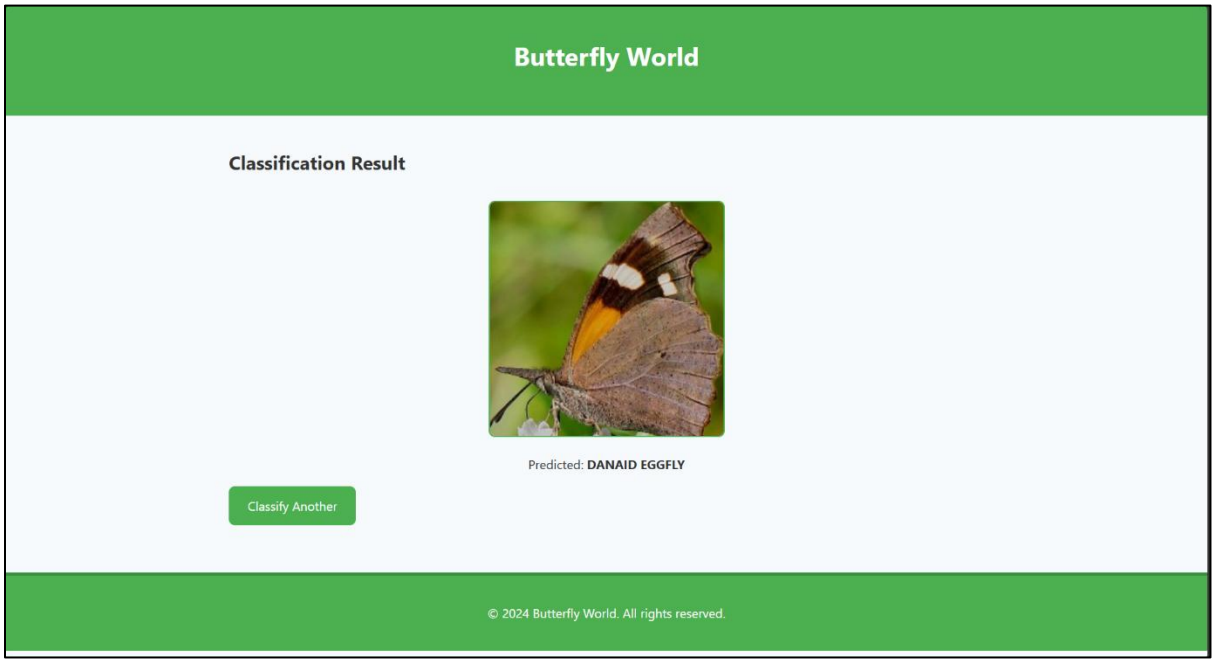
- Deploy online to support remote usage.
- Integration with mobile apps for field research.
- Incorporate real-time detection via camera streams.
- Extend to other insect or wildlife classifications.
- Add multilingual support in the UI.
- Integrate more educational content and interactive visuals.

12. Screenshots or Demo




Butterfly World
Home Classify
Upload a Butterfly Image
Choose File 2.jpg
No file chosen
Classify
© 2024 Butterfly World. All rights reserved.


Butterfly World
Home Classify
Upload a Butterfly Image
Choose File 2.jpg
No file chosen
Classify
© 2024 Butterfly World. All rights reserved.



13. License

This project is for educational purposes only.
No part of this code may be used, copied, or distributed without explicit permission from the authors.

14. Contact

For questions or contributions, please contact the team via GitHub issues or [muntanavadeepvenkatasaikumar.22.cse@anits.edu.in].

⁂