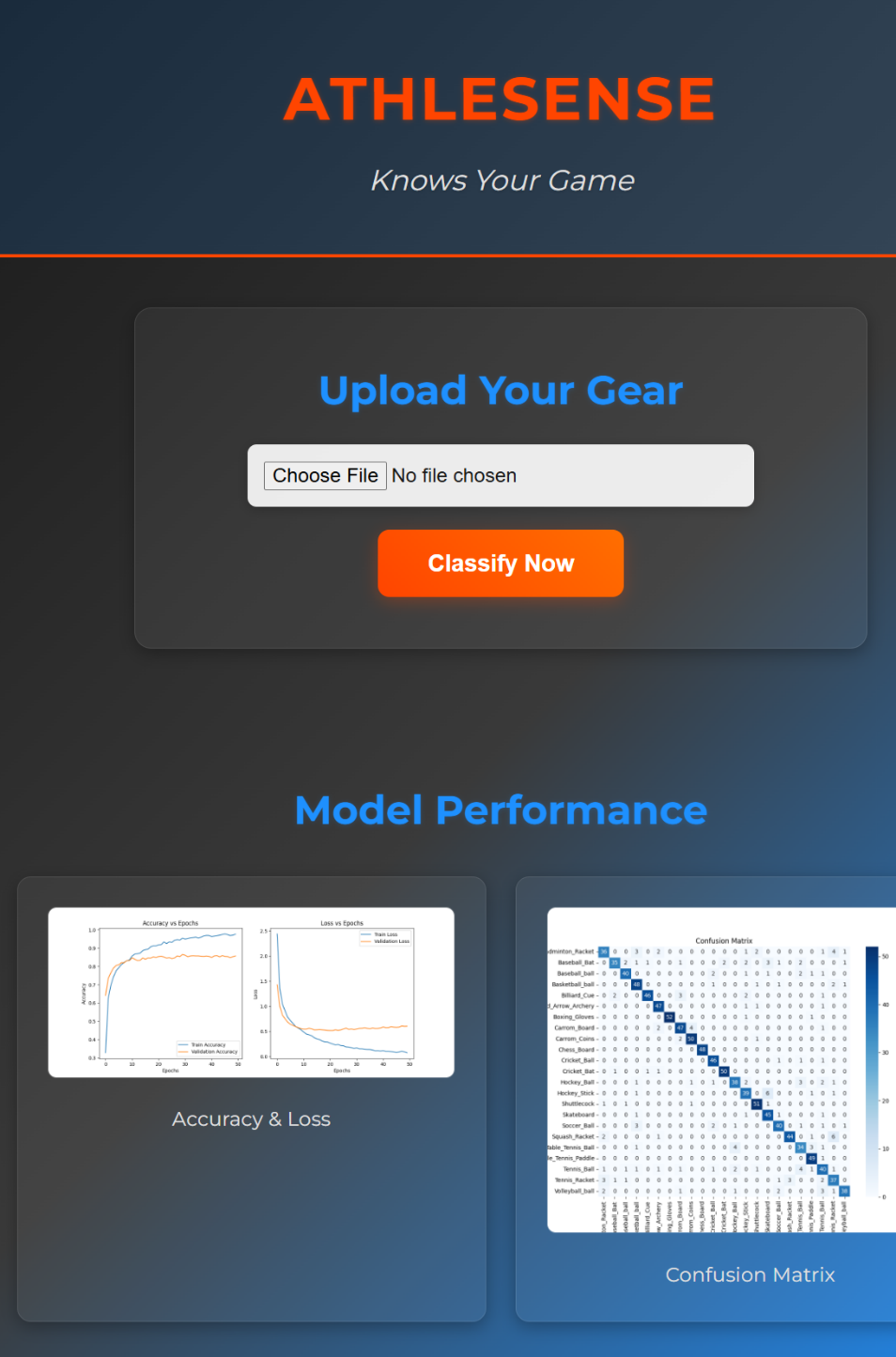
**ATHLESENSE - Knows Your Game 🏆🎾🏀⚽**

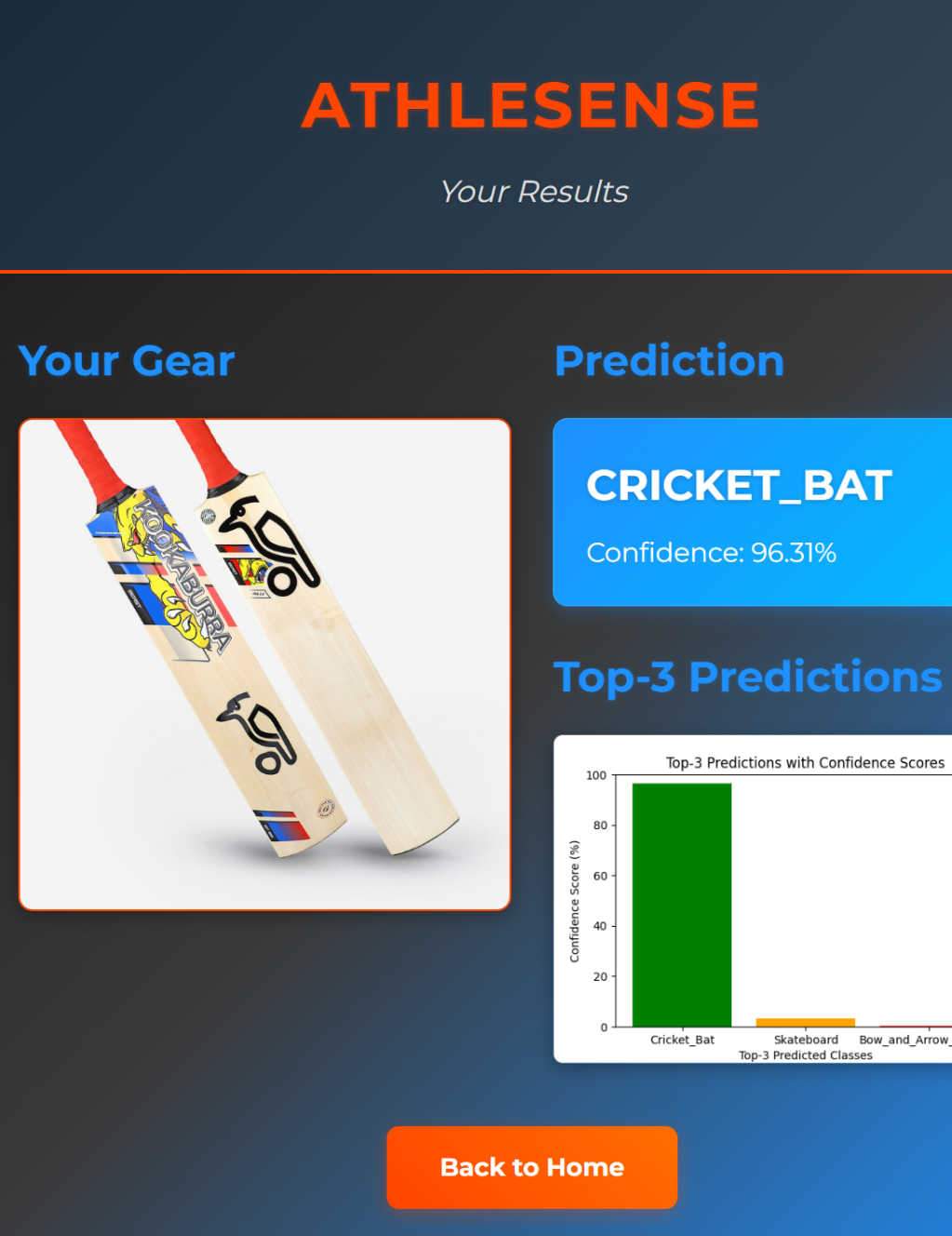
**[Deep Learning Flask App]**

**OVERVIEW :**

This project is a deep learning-based sports equipment classification system built using TensorFlow, Keras, and Flask. It allows users to upload an image of a sports item (e.g., Basketball, Football, Tennis Racket, etc.), and the trained model predicts the correct category with a confidence score. The project also includes a Flask web application to provide a user-friendly interface.

**PROJECT IMAGES :**





**FEATURES :**

* Image Upload:

Users can upload an image for classification.

* Deep Learning Model:

Uses MobileNetV2 for accurate image recognition.

* Training & Performance Visualizations:

Displays accuracy, loss curves, and confusion matrix.

* Top-3 Predictions:

Shows the most probable classifications.

* Flask Web Application:

Provides an interactive web interface.

* Data Preprocessing:

Filters out grayscale, low-quality, and corrupt images.

* Model Evaluation:

Generates a classification report for performance analysis.

**TECH STACK :**

* Python - Core programming language
* Flask - Web framework for UI & backend
* TensorFlow / Keras - Deep learning model training
* OpenCV - Image processing & resizing
* NumPy / Pandas - Data manipulation
* Matplotlib / Seaborn - Visualization (graphs, confusion matrix)
* ImageDataGenerator - Data augmentation
* imagehash - Duplicate image detection

**📂 PROJECT STRUCTURE :**

📂 ATHLESENSE

│── 📂 athlesense/ # Flask application

│ │── 📂 static/ # Stores images, CSS, and results

│ │ ├── uploaded\_image.jpg # Latest uploaded image

│ │ ├── accuracy\_loss.png # Training accuracy/loss graph

│ │ ├── confusion\_matrix.png # Model confusion matrix

│ │ ├── styles.css # CSS styles for UI

│ │── 📂 templates/ # HTML files

│ │ ├── index.html # Home page (upload image)

│ │ ├── result.html # Displays classification result

│ │── app.py # Flask application logic

│ │── train\_model.py # Trains the deep learning model

│ │── test\_image.py # Classifies a single image

│ │── new\_train\_model.py # Trains the deep learning model with new features

│── 📂 dataset/ # Raw dataset of images

│── 📂 processed\_data/ # Preprocessed dataset

│── 📂 train\_test/ # Train and Test Data splitup

│── web\_scraper.py # Scrapes images from Web for the Dataset

│── preprocess.py # Cleans and processes dataset images

│── dataset\_seperation.py # Train and Test Data splitup

│── requirements.txt # Python dependencies

│── README.md # Documentation

**MODEL TRAINING PROCESS :**

* Data Collection - Organized images of sports equipment into labeled folders.
* Data Preprocessing - Removed duplicate, grayscale, and small images; resized to 224x224.
* Data Augmentation - Applied transformations (rotation, zoom, brightness, etc.) to improve generalization.
* Model Selection - Used MobileNetV2 (pretrained on ImageNet) for feature extraction.
* Custom Model Architecture - Added a dense classification head with dropout layers.
* Training & Validation - Trained with categorical cross-entropy loss and Adam optimizer.
* Evaluation - Generated a confusion matrix and classification report.
* Deployment - Saved the trained model and integrated it into the Flask app

**FLASK WEB APP WORKFLOW :**

1. User uploads an image on the home page.

2. Backend processes the image and passes it to the trained model.

3. Model predicts the class and generates a probability score.

4. Result page displays the classification result and confidence score.

5. User can go back and upload another image.

**SETUP & INSTALLATION :**

1. Clone the Repository

**Bash:**

git clone https://github.com/HaswanthR-CIT/ATHLESENSE

cd ATHLESENSE

2. Install Dependencies

**Bash:**

pip install -r requirements.txt

3. Train the Model

**Bash:**

python train\_model.py

(This will train the model and save it as sports\_equipment\_model.h5.)

4. Run Flask Application

**Bash:**

python app.py

Open your browser and visit: http://127.0.0.1:5000/

**LICENSE :**

This project is open-source and available under the MIT License.

MIT License

Copyright (c) [2025] [Haswanth R]

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.