Project Title: Inclination Angle Estimation for Motorcycle Drivers Using a GoPro 360 Camera and Jetson Nano

Objective

Develop a system that estimates the inclination angle of a motorcycle driver using a GoPro 360 Max camera and machine learning, deployed on a Jetson Nano.

Expected Results

1. Real-time estimation of the driver's inclination angle.
2. A trained ML model optimized for Jetson Nano deployment.

Methodologies

1. **Data Acquisition:**
    - Mount the GoPro 360 Max on the motorcycle to record videos of the driver.
    - Annotate the data with ground-truth inclination angles (using IMU sensors).
2. **Model Selection and Training:**
    - **Model Type:**
        - **Pose Estimation Models:** OpenPose or BlazePose for analyzing the driver's posture.
        - **Regression Models:** Lightweight CNN or ResNet for predicting the inclination angle based on extracted features.
    - **Preprocessing:**
        - Normalize video frames and extract key points (e.g., body joints, bike tilt).
        - Synchronize video data with IMU measurements if applicable.
    - **Training :**
        - Train on annotated datasets using frameworks like TensorFlow or PyTorch.
3. **System Integration on Jetson Nano:**
    - Stream video from the GoPro to the Jetson Nano using Wi-Fi or USB-C.
    - Process frames in real time, extracting features and estimating angles.

- Visualize results on a connected display or log data for further analysis.

4. **ROS Implementation (Optional):**
    - Use ROS for modular development and integration.
    - Create ROS nodes for:
        - Camera input and frame publishing.
        - ML inference for angle estimation.
        - Output visualization or storage.
5. **Evaluation:**
    - Study the faisability.

Frameworks and Tools

1. **Hardware:**
    - GoPro 360 Max camera.
    - NVIDIA Jetson Nano.
    - IMU sensor (The GoPro camera has one IMU and GPS).
2. **Software:**
    - **Pose Estimation:** OpenPose, BlazePose, or MediaPipe.
    - **Machine Learning Frameworks:** TensorFlow, PyTorch, or Keras.
    - **ROS:** For integration and modularity.
    - **Data Annotation Tools:** LabelImg, CVAT.