🌾 Automatic Targeted Fertilizer Dropping Machine (AFDM)

🎯 Project Summary

The **Automatic Targeted Fertilizer Dropping Machine (AFDM)** is a robust, low-cost robotic solution developed for precision agriculture. The core objective of the project is to enhance farming efficiency and sustainability by replacing traditional, inefficient broadcast fertilization with **precise, plant-specific fertilizer application.**

The system integrates embedded computing, machine learning, and basic IoT concepts to autonomously navigate a field, detect individual plants, and deliver a measured dose of fertilizer directly to the base of the crop, significantly reducing fertilizer wastage and environmental impact.

✨ Key Features & Technological Stack

The AFDM is built upon high-performance, low-cost embedded components to achieve autonomous functionality.

Component	Function in the System	Technology/Concept
ESP32-CAM	Captures images and utilizes an **Edge Impulse**-trained Machine Learning (ML) model for **real-time plant detection.**	Machine Learning (ML), Computer Vision
Arduino UNO	The central control system that manages robot movement and fertilizer delivery based on data received from sensors (Ultrasonic, ESP32-CAM).	Embedded Systems, Microcontroller Programming
Ultrasonic Sensor (HC-SR04)	Measures the proximity of the detected plant. It signals the Arduino to **stop the robot** at a predefined, optimal distance for application.	Sensing, Proximity Measurement
Servo Motor	Activates the fertilizer dispensing mechanism to release a **measured amount** of fertilizer onto the plant.	Actuation, Precision Delivery
DC Motors & Motor Driver	Provide movement for the robot. The H-bridge motor driver (e.g., L298N) is controlled by the Arduino to drive the motors.	Robotics, Locomotion
Bluetooth Module (HC-05)	Enables manual control of the robot's direction via a mobile app, providing operational flexibility and remote override capability.	Internet of Things (IoT), Wireless Communication
Li-ion Battery Pack	Provides power to the entire system, ensuring extended operation in the field.	Power Management

⚙️ Detailed Operational Flow

The machine operates in a continuous loop in its primary autonomous mode:

1. **System Initialization:** The machine is powered on. The Arduino initializes all connected components (motors, sensors, servo), and the ESP32-CAM starts its detection loop.
2. **Autonomous Movement:** The DC motors are engaged, and the robot begins moving forward, systematically scanning the field.
3. **Plant Detection:** The **ESP32-CAM** captures images. The embedded ML model rapidly analyzes the image.
4. **Target Confirmation:** If the ML model identifies a target crop, a digital signal is sent from the ESP32-CAM to the **Arduino UNO**.
5. **Proximity Approach:** The robot continues moving forward while the **HC-SR04 Ultrasonic Sensor** constantly measures the distance to the detected target.
6. **Stop and Position:** When the measured distance equals or falls below the predefined proximity threshold (e.g., 10 cm), the Arduino commands the motor driver to **stop** the robot.
7. **Fertilizer Application:** The **Servo Motor** is actuated to open the dispensing channel. A precisely measured amount of fertilizer is delivered.
8. **Cycle Reset:** The Servo Motor closes the channel, and the Arduino commands the DC motors to move forward again, restarting the process of searching for the next plant.

Manual Control Override

The **HC-05 Bluetooth Module** allows an operator to interrupt the autonomous loop and send direct movement commands (forward, reverse, left, right, stop) to the Arduino via a smartphone application, which is crucial for maneuvering into new rows or overcoming obstacles.

🚀 Getting Started

1. **Hardware Setup:** Wire the components (Arduino, L298N, Servo, Ultrasonic Sensor, ESP32-CAM output, HC-05) according to the schematics provided in the `docs/` folder.
2. **Code Upload:** Open `AFDM_Arduino_Code.ino` in the Arduino IDE and upload it to the Arduino UNO board.
3. **ML Model:** Ensure the ESP32-CAM is running the compiled machine learning model from **Edge Impulse** and is correctly signaling the Arduino.
4. **Operation:** Power the system using the Li-ion battery pack and test the autonomous and manual control modes.