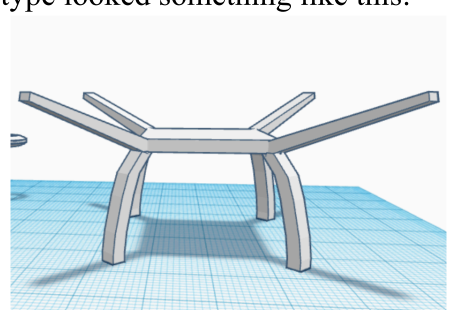
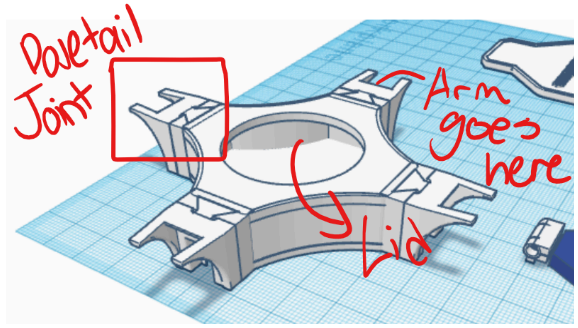
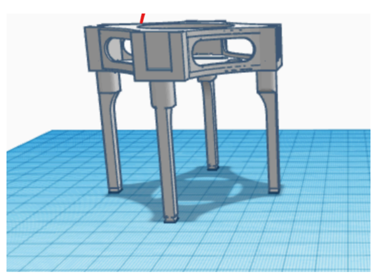
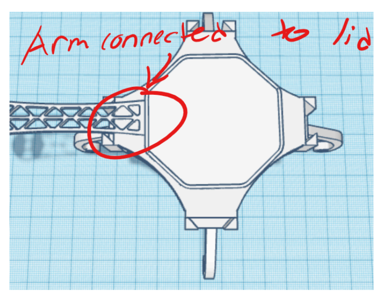
LOGBOOK

Are Drones the Future of Agriculture?

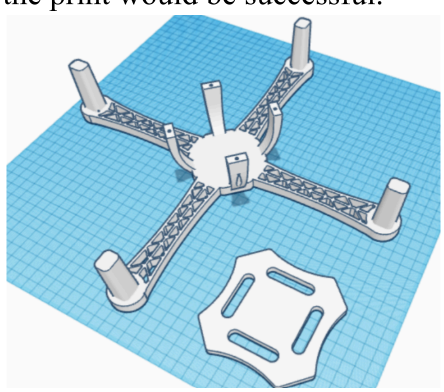
Name: Prabhuansh Sadana and Jogeshwor Mundi ***Grade: 8*** ***School: Khalsa School Calgary***	

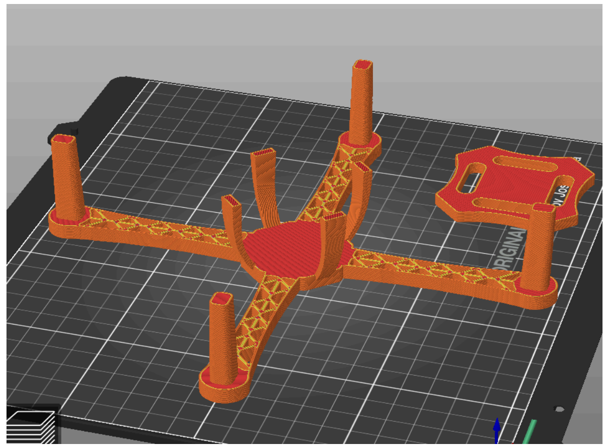
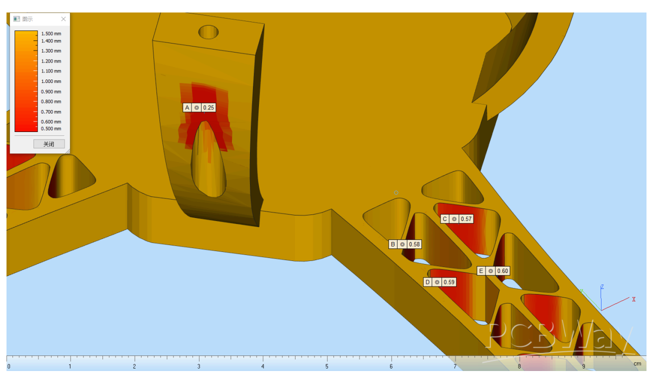
Timetable

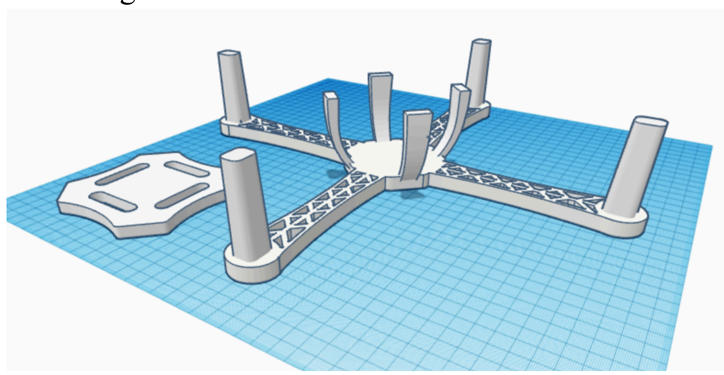
DATE	GENERAL CATEGORY	DESCRIPTION
August 1, 2023	Brainstorming	We decided to brainstorm about our topic for the project. We had initially wanted to do a drone, but we felt that a special feature was required in order to make the drone useful for more applications. We wanted it to base it off of an industry, specifically farming. Now, we needed to discuss what special features the drone would have.
August 2, 2023	Brainstorming	We discussed the features of the drone, which includes the following: ● Soil Moisture Detection These features would soon be the basis of our project and would be more refined as time went on.
August 8-10, 2023	Brainstorming & Designing	After finalizing the features and getting a general idea of the project, we decided to brainstorm about the materials and began to design the first prototype of the drone. We had initially designed it in Tinkercad, which we will be most likely using for our project. Our first prototype looked something like this: 
August 10-14, 2023	Brainstorming & Programming	First we decided what coding language we should use, we were debating on using either C or Python. After some thinking, we decided to use C. C is very common in robotics and it is very fast compared to Python which is relatively slow compared to most other languages.
August 15,	Brainstorming	Although we had decided on the features of the drone, we

2023		wanted to implement another feature that would change the project itself. Now, we will be testing an agricultural drone's efficiency to compare it to the efficiency of existing farming techniques and conclude whether or not drones will be sustainable in agriculture in the future. This aspect of the project is crucial to allow us to examine and conduct a variety of tests to conclude a result.
August 16-17, 2023	Designing	The first initial prototype was scrapped, and a new design was later created. It consisted of dovetail joints, and the arms would slide into those joints to create a strong and stable arm. It looked something like this: 
August 18, 2023	Designing	A new design for the legs was proposed and soon designed. Furthermore, we had proposed that the arms were to be connected onto the "lid" so that it could ensure that the arms were properly connected and that the drone remained strong throughout flight.  
August 31, 2023	Brainstorming	The final features of the drone were brainstormed and created. These included: ● High-resolution camera for detailed crop imaging, which can help with precision agriculture by monitoring crop yields and identifying areas for improvement.

		• Automated drone control system depending on land shape. • Automated water and fertilizer sprayer for targeted application of nutrients (scrapped later) • Soil moisture sensors for data collection and analysis
November 2-10, 2023	Research	Research was conducted and we analyzed a variety of sources to further strengthen our knowledge with the world of drones. Some research papers we analyzed today includes: • https://www.sae.org/publications/technical-papers/content/2021-01-0809/ • https://www.fao.org/e-agriculture/news/moocdrones-agriculture-prepare-and-design-your-drone-uav-mission • http://ijlemr.com/papers/volume2-issue7/18-IJLEMR-22328.pdf • https://oriontechnology.tech.blog/2023/04/25/steps-to-design-agricultural-drones/ • https://www.edx.org/learn/drones/wageningen-university-research-drones-for-agriculture-prepare-and-design-your-drone-uav-mission
November 14, 2023	Programming	Started coding in C, and got pretty far in the project after spending 2 hours. I also coded some python just to be safe if things don't go to plan. A lot of progress happened and we all felt more confident about the drone.
November 17, 2023	Brainstorming	Project was approved by our school following some questions that were asked by our coordinator.
November 27-28, 2023	Programming & Writing	Today we finished the programming for the last version of the code and started to wait for the model to be finished. We sent a few messages to CYSF regarding our project as per instruction by our coordinator and awaited for approval.
November 30-31, 2023	Brainstorming	We decided to do more research into how we are going to conduct our experiment. We concluded that we are going to be examining the efficiency by conducting a 'ground truth measurement', which is using a soil moisture sensor without the drone to compare it to the results and see if the drone's measurements were accurate. We are also going to research traditional farming methods and their impact to compare it to the drones data received by the tests to conclude how efficient

		the drone was. For the test itself, we will perform 3 trials, testing the soil moisture sensor, camera, and the irrigation system (which is a sensor, Arduino, and solenoid valve to water the soil) to gather our data.
December 1, 2023	Writing	Continued to work on the writing aspect for our project.
December 5, 2023	Brainstorming & Research	Today we researched a little bit about where we are going to be conducting the experiment, and also estimated when everything would be finished.
December 9, 2023	Designing	Did some more work on the 3D model; estimated overall dimensions and prepared the model for printing.
December 11, 2023	Designing and Brainstorming	Searched the web for 3D printing services. We also finalized our materials for the project and continued to prepare the model for 3D printing.
December 13-15, 2023	Designing	The old design of the drone was scrapped once more and a new design was created. It consisted of the same arms but slightly different base and completely new legs, as shown below. The drone would also need to be printed upside down to ensure that the print would be successful. 
December 16, 2023	Brainstorming	The materials were fully finalized and ready to be bought for our first prototype. The final materials were put in our 'Materials' section.
December 20, 2023	Designing	The model was put into a slicer that shows how the model would be printed. It looked something like this:

		
December 23, 2023	Materials	The materials are ordered from Amazon and will be shipped in a few days. Everything we ordered for prototype 1 was: • Propellers • Pi Camera • Motor Drivers • Motors • Soil Moisture Sensor • Battery • Arduino Uno R3 • Raspberry Pi Zero W
December 24, 2023	Materials	Drone is sent to the 3D printing service (PCBWay) for approval.
December 25, 2023	Materials	 Received response from the 3D printing service. Unfortunately, some of the walls were too thin (as shown in red) and would need to be fixed before the model could be printed.
December	Materials	The drone is now ordered and everything is set to arrive on

26, 2023		time. The final prototype for the drone consisted of all the walls being thickened and everything being finalized. It looked something like this: 
December 27-31, 2023	Materials & Building	All of the materials arrived and we are now currently waiting on the drone frame so we can begin the experiment.
January 1-5, 2024	Programming	Before the drone arrives, all of the circuits and code is tested to ensure that there will be no errors once we are testing the drone.
January 6-7, 2024	Testing	The soil moisture sensors and the motors were tested successfully. In this time we also performed our ground truth test.
January 8, 2024	Materials & Building	Today, the drone frame arrived! We encountered a slight issue with the code so the experiment would need to be delayed.
January 9-11, 2024	Building & Testing	The drone frame was constructed and the tests were conducted for our first prototype. We conducted a ground truth test first to ensure the collection of accurate data. The tests did not come as expected, as the DC motors were not powerful enough to generate much lift. It did hover, however
January 12-March 5th, 2024	The Expansion of Prototypes	During this period of time, we analyzed our results for our first prototype and went on to design our next prototypes. We added a camera for prototype 2 and completely redesigned prototype 3. We tested these prototypes and recorded all of the results. All of the science fair headings and such were completed in this time as well, and the project was finalized to prepare for the in-person judging.

Background Research

Modern Farming Techniques in Current Society

Many experts believe that, by 2050, food production will need to double to meet the world's growing demand. Without updating the current agricultural practices, doubling the world's food production would be very difficult. In fact, it would require the clearing of approximately one billion hectares of land. This would have a significant impact on climate change. However, drones can change that. There are many advantages towards using drones in agriculture. For one, drones are able to cover distant areas of farmland in a short period of time, which allows farmers to monitor crops more efficiently and accurately. This allows farmers to gather accurate information about crops and helps to identify potential issues. Second, farmers are able to save on labor costs as drones can perform tasks such as spraying pesticides, mapping, and crop monitoring, meaning that drones can reduce the need for expensive equipment.

Drones enable precision agriculture practices by providing real-time data and imagery. This allows farmers to decide how much fertilizer, pesticides, and water is needed with precision to reduce wasted chemicals and minimizes the environmental impact a drone has. Finally, drones equipped with specialized sensors and imaging technology have the ability to detect crops with disease, nutrient deficiencies, or other issues. This early detection allows farmers to take timely action, preventing the spread of diseases and minimizing crop losses.

Comparison Between Traditional and Modern Farming Techniques

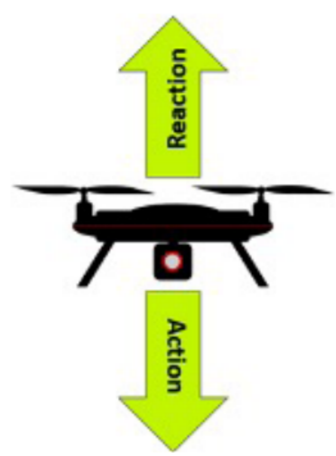
	Traditional Farming Practices	**Modern Farming Practices**
Approach	Traditional farming is mainly based on old practices that emphasize the use of manual labor. Traditional farming also relies on organic practices and the use of natural seeds.	Modern farming uses advanced technology to produce crops. Modern farming generally focuses on large-scale production to make more profits. Modern farming also implements precision agriculture to optimize the resources used. Finally, modern farming aims to be more productive and efficient than traditional farming.

Environmental Impact	Traditional farming has a generally low environmental impact due to the use of organic practices. Traditional farming also uses traditional methods, meaning there is limited use of chemicals and advanced technology.	Modern farming has a large environmental impact due to the chemical inputs that are released from modern farming practices. Modern farming mainly incorporates synthetic pesticides and fertilizers for pest management.
Productivity	Traditional farming has a generally lower productivity than modern farming. This is because traditional farming is commonly practiced for supplying local goods, and it is much more difficult to supply goods to a large population due to the lack of resources.	Modern farming has a high productivity due to many technological advancements. AI, drones, etc. all are astonishing technologies that have had a huge impact on the agricultural industry. Modern farming generally meets the needs of a large population.
Sustainability	Traditional farming promotes sustainability through organic practices and is well-suited for maintaining traditional practices and local ecosystems.	Modern farming does not promote sustainability due to the amount of resources that are used. Modern farming requires much more management than traditional farming to ensure that modern farming can have a lesser environmental impact.

Physics and Movement of a Quadcopter

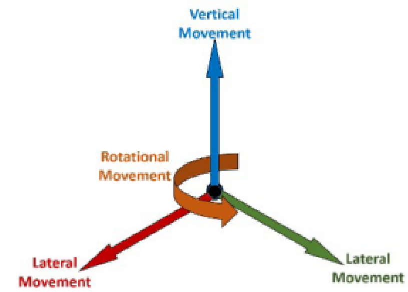
Newton's Third Law of Motion

Newton's third law of motion states that every action that occurs has to have an opposite and equal reaction that is generated. In the case of a quadcopter, the propellers pushing air downwards represents the action. The equal and opposite reaction is the upward force that is generated by the propellers force pushing air downwards. If the upward force is greater than the force of gravity, the quadcopter begins to propel upward.

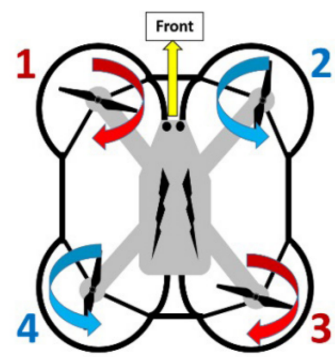


Movement of a Drone

A quadcopter must be able to perform different types of movements in order to be able to fly properly. This includes rotational, lateral, and vertical movement. As the propellers spin, they push air downward, which pushes the quadcopter up. In order for the drone to achieve lift, the force of lift must be greater than gravity. Moreover, the quadcopter is able to hover once the forces of gravity and lift are equal. Lateral (left and right) movement occurs when 2 of the motors on one side are spinning faster as compared to the 2 on the other side. The quadcopter will then move in the direction where less lift is created.



Each individual propeller produces torque when they rotate. In a helicopter, not only is there a main propeller, but also the tail rotor. The tail rotor helps to counteract the torque that is being created, which allows the helicopter to remain stable. Drones, on the other hand, are unable to have a tail rotor as instead of one propeller, there are 4. So, engineers have concluded that, in order to counteract this torque, two of the propellers have to be spinning clockwise, and two of the propellers have to be spinning counterclockwise. They must be diagonal from each other. So the forces cancel each other out and the drone is able to remain steady throughout flight.



Components of a Quadcopter

Unmanned Aerial Vehicle

A UAV (Unmanned Aerial Vehicle) is a flying device capable of flying a pre-set course with the help of GPS coordinates and autopilot. The term is commonly used for describing model planes and helicopters with both fixed and rotating wings.

Advantages

There are many advantages to using an Unmanned Aerial Vehicle. For one, UAVs can fly longer hours without human interference, doing the required tasks for however long the drone pilot requires under computer control. The drone operators are able to easily gain access to the drone and can manually control the drone as they wish.

Core Components

- **Chassis:** The chassis is where all of the components are connected to. In order for the drone to fly, the chassis must be lightweight and durable.

- **Motors:** The motors are used to spin the propellers so that the drone is able to fly. Motors are crucial for the success of the drone.

- **Propellers:** The propellers are attached to the motors and are what generate lift. The length of a propeller can be modified; longer propellers are able to achieve greater lift at a lower RPM (revolutions per minute) but take longer to speed up or slow down.

- **Flight Controller:** The onboard computer which controls the quadcopter based on signals sent from the pilot.

- **ESC:** Controls the speed of the motors; is crucial to give the motors a specific current depending on the required input.

- **Battery:** Supplies power to the drone and is crucial to the success of the drone. High power batteries will allow the drone to fly for long periods of time.

- **Camera:** Cameras are mounted for navigation and aerial photography.

How Does a Drone Fly?

A drone can be controlled manually with a handheld radio transmitter which manually controls the speed of the propellers. Sticks on the controller allow the drone to be moved in different directions. Screens can also be used to receive video footage from the cameras on the drone and to display sensor data. Drones are also able to be controlled autonomously, as flight controllers can use software to mark GPS waypoints that the vehicle will follow.

Problem

In modern society, many countries around the world have limited access to advanced farming equipment. This is mainly due to the cost of these equipment. In fact, agricultural drones can cost around 1,000 to 10,000 dollars if someone is looking for a drone with better sensors, cameras, and an increased payload. So, in order to find a solution to this problem, we wanted to see how we could make a budget friendly drone with an increased payload, quality cameras, and sensors so that people around the world who have limited access to this technology can have a better opportunity towards farming. However, once these technologies are accessed throughout the world, it may not be as sustainable and productive as compared to traditional farming practices. Farmers need to prepare their crops in a short period of time to ensure that the farmer can make more profits. In order to find a solution to this problem, however, we need to experiment and observe how long the drone takes to do a specific task (measure the soil moisture of plants) in order to conclude its efficiency. So, these 2 solutions not only allow farmers to have access to advanced farming technology but also make informed decisions about the efficiency of these technologies. With this information, we can conclude if drones are the future of agriculture.

Method

Prototype 1

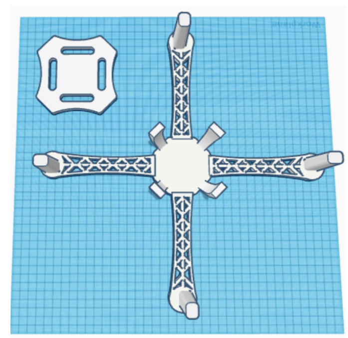
Design and Programming

Design

Our first prototype was our very first attempt at creating the agricultural drone. The prototype consisted of 2 motor chips, known as the L289N motor drivers, and 4 DC Motors. We also used an Arduino Uno and a Raspberry Pi Zero W to control 2 of the motors individually, and an additional Arduino Uno to control the soil moisture sensor. This prototype unfortunately had no video capabilities and was

lacking a very crucial component that many agricultural drones have that are currently in the agricultural industry. So, most of the footage shown from this prototype were from the ground and were not integrated into the model itself.

The chassis was created on a simple modeling software, Tinkercad, as we lacked the experience for any industrial modeling softwares. The model, though designed on Tinkercad, proved to be lightweight, durable, and efficient for our project. The quadcopter design did have many qualities and was used throughout the 3 prototypes that we created.

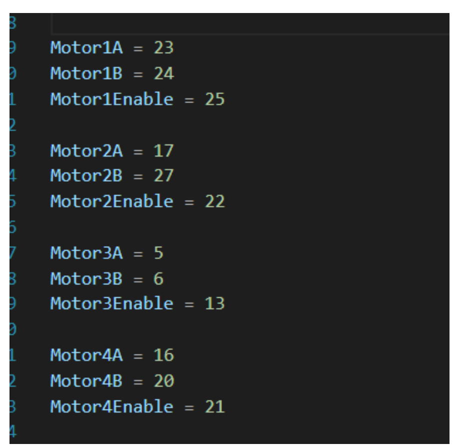


Programming

The drone was using 2 main coding languages, C++ (Arduino variation) and Python which was used in the Raspberry Pi. The Raspberry Pi in this project controlled two of the motors, while the Arduino Uno controlled the other two. The code was organized as follows:

Raspberry Pi Code (Python)

1. Define all wires and where they are at the microcontroller.
2. Create four sections for different code.



```
Motor1A = 23
Motor1B = 24
Motor1Enable = 25

Motor2A = 17
Motor2B = 27
Motor2Enable = 22

Motor3A = 5
Motor3B = 6
Motor3Enable = 13

Motor4A = 16
Motor4B = 20
Motor4Enable = 21
```

GPIOSTART

- ○ **1. GPIOSTART**
  - ■ This function is very important. This initializes the microcontroller and makes it ready to listen to commands.

```python
import RPi.GPIO as GPIO
import time


GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
```

- **2. DRONECONTROL**
  - ○ This function is also very important. This function controls the drone and tells it when the motors are supposed to be on and also when to turn off.

```python
for pins in motor_pins:
    for pin in pins:
        GPIO.setup(pin, GPIO.OUT)
        GPIO.output(pin, GPIO.LOW)

def turn_on_motors():
    for pins in motor_pins:
        GPIO.output(pins[0], GPIO.HIGH)
        GPIO.output(pins[1], GPIO.LOW)
        GPIO.output(pins[2], GPIO.HIGH)

def turn_off_motors():
    for pins in motor_pins:
        GPIO.output(pins[2], GPIO.LOW)

try:
    turn_on_motors()
    time.sleep(5)
    turn_off_motors()

except KeyboardInterrupt:
    turn_off_motors()

finally:
    GPIO.cleanup()
```

- **3. SUBMITDATA**
  - ○ This function sends sensor data to the ground where the operators will be and will receive data about the weather, gyroscope, and air pressure that is around the microcontroller.
  - ○ The microcontroller is at the bottom of the motors to make sure there is no interference from the motors blowing air into the sensor, therefore changing the result. Furthermore, the drone will try to stay in stable positions with the help of the built in gyroscope.

```cpp
const int soilMoisturePin = A0;
const int dryThreshold = 600;
const int wetThreshold = 300;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int soilMoistureValue = analogRead(soilMoisturePin);
  Serial.println(soilMoistureValue);

  if (soilMoistureValue >= dryThreshold) {
    Serial.println("Soil is dry - needs water!");
  } else if (soilMoistureValue <= wetThreshold) {
    Serial.println("Soil is wet enough");
  } else {
    Serial.println("Soil moisture is within acceptable range");
  }

  delay(1000);
}
```

Arduino Code (from the Arduino IDE)

1

```
// Motor A connections
int enA = 10;
int in1 = 9;
int in2 = 8;

// Motor B connections
int enB = 5;
int in3 = 7;
int in4 = 6;
```

2

```
  // Motors initially stopped
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
}
```

3

```
void loop() {
  // Run both motors forward for 5 seconds
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  analogWrite(enA, 255); // Motor A speed (0-255),

  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  analogWrite(enB, 255); // Motor B speed (0-255),
```

4

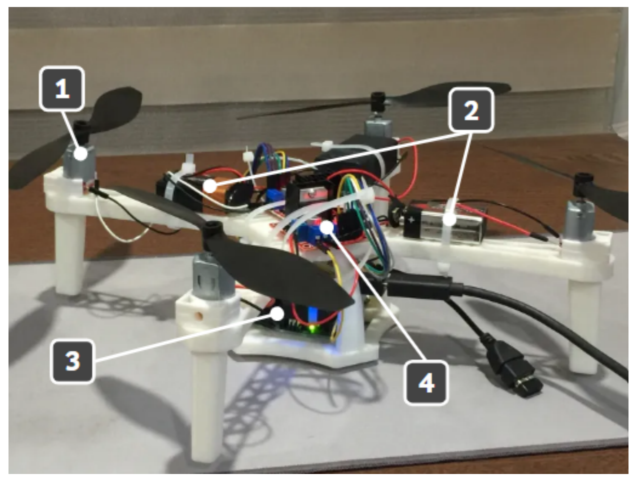
```
// Stop both motors after 5 seconds
digitalWrite(in1, LOW);
digitalWrite(in2, LOW);
analogWrite(enA, 0);

digitalWrite(in3, LOW);
digitalWrite(in4, LOW);
analogWrite(enB, 0);
```

Building and Functionality

The construction of the agricultural drone consisted of the flight controllers at the bottom of the drone, the motor drivers at the top of the drone, and the 4 DC motors on each arm of the drone. A soil moisture sensor was also integrated near the bottom of the drone to allow the drone to obtain the soil moisture of the soil samples. Most of the pieces were connected using super glue, and the batteries and such were connected using zip-ties. To connect each of the components to each other, a system of wires was integrated. The wires connecting to the motors were not very strong, which was a strong source of error for this prototype.

Upon further testing of the drone, we did not previously anticipate the amount of load the DC motors would be able to carry. As a result, the drone was not able to achieve very much lift, and was not able to fly very well. The DC Motors that we bought for this project were unable to produce heavy amounts of lift for the drone to stay up from the ground for extended periods of time. Due to this issue, the drone could only reach low altitudes. The drone was able to hover for a few seconds and then would lose altitude relatively quickly. This iteration of the project proved to be budget friendly and had already passed the first part of our problem statement. However, due to the limited functionality of the drone, the drone was very slow and could not fly very well.

To conclude this section, Although this prototype did in fact function properly, there were major flaws in the wiring and maneuvering components of the drone. For one, the drone was not able to achieve much lift, as mentioned previously. This was mainly due to the inefficient motors that we had purchased for the prototype. Secondly, the drone was not able to hover in a single position, and was not very stable. In the future, we would like to add things such as gyroscopes to aid in the steady maneuvering of the drone, so that the drone could steadily obtain the soil moisture of plants. Finally, it was very difficult to keep the whole machine compact as the wires made it difficult for the propellers to spin without disturbing the connection.



Prototype 2

Design and Programming

Design

For this prototype, the design remained the same, however this iteration was a minor improvement as compared to our previous prototype. Our major flaw in our previous prototype was the lack of video footage. The new prototype was equipped with the Pi Camera so that the drone would be able to record footage from a far distance. Although the Pi Camera did work, it was useless as the drone could not reach high altitudes so that the camera could obtain footage.

Programming

With the addition of the Pi Camera, the programming had to be minorly changed:

PICAMERA

- This function allows the onboard camera to record for 5 minutes. We will check the footage recorded after the flight is concluded.

```
import picamera
import time

camera = picamera.PiCamera()

camera.resolution = (640, 480)  # Adjust the resolution as needed

file_path = '/path/to/storage/dronevideofolder'  # Update the path as needed

try:
    camera.start_recording(file_path)

    camera.wait_recording(300)

    camera.stop_recording()
finally:
    camera.close()
```

Building, Testing, and Functionality

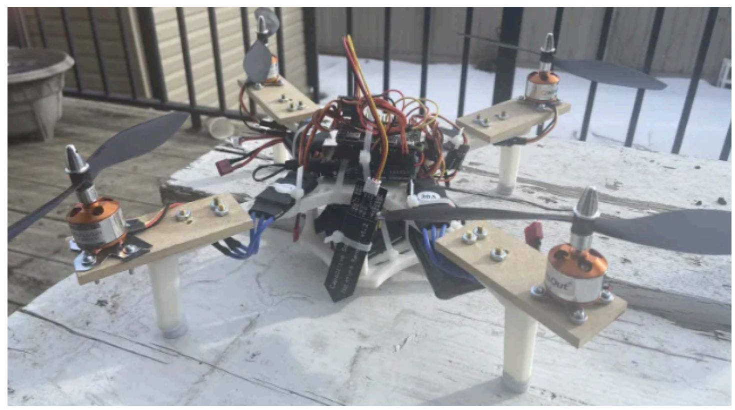
The functionality and results of the drone remained the same, the only difference obviously being the addition of the Pi Camera. The Pi Camera allowed us to receive video footage that we previously could not receive, and was overall very useful for our future prototypes. This prototype surpassed the first section of our problem once again, but still lacked the speed for the drone to be able to surpass human capabilities at a large scale.

Overall, although this iteration was successful in some areas, it did have many flaws that did affect the performance and accuracy of the drone itself. The code, design, etc. of this prototype remained the same, however, the only difference was the addition of the camera. We planned for our next prototype, aiming to make the flight performance and accuracy of the drone different in many aspects.

Prototype 3

Design and Programming

Design



For our last and current prototype, we decided to change many factors of the drone to allow the prototype to achieve lift effectively, remain stable throughout flight, and overall ease the wiring and other processes. For one, we decided to completely replace the DC motors with brushless motors, which allowed the propellers to reach staggering RPM's (revolutions per minute). Along with the brushless motors, we also needed to purchase ESC's which allow the motors to have a stable connection with our flight controllers. Furthermore, the Raspberry Pi, once used for the motors, was now changed to be used for the soil moisture sensor, and having the 2 Arduino Uno's control the motors. Finally, we decided to integrate a gyroscope (MPU 6050) into this prototype to allow the drone to be stable, and to connect 2 motors to the Arduino at one time. The soil moisture sensor and camera were in the same positions of the last 2 prototypes, however the programming for the soil moisture sensor had to be adjusted to accommodate for the changes we made in this prototype.

The chassis remained the same for this prototype as well. However, in order to connect the motors to the drone, adjustments would have to have been made. We had to use a wood piece to screw the motors on, and then connect that piece onto the drone, as the motors were quite bigger as compared to the drone's motor areas. The new design proved to be successful, as the new motors were powerful enough to generate the lift needed for the drone to hover in the air, and easily obtain the soil moisture.

Programming

Due to the change in the motors, new programming would have to be added to make the motors function. Furthermore, we also had to recode the soil moisture sensor with the Raspberry Pi now taking control of it. We also had to do many calculations to get the gyroscope correct and allow the drone to remain stable throughout flight. The code needed for the motors was now all Arduino C++

Arduino Code (Motors)

- Define all libraries needed for our project (there are red lines because my Visual Studio Code was not updated).

```
#include <Wire.h>
#include <Servo.h>
```

- Define all the variables needed for the MPU-6050 to function.

```
Servo right_prop;
Servo left_prop;

int16_t Acc_rawX, Acc_rawY, Acc_rawZ, Gyr_rawX, Gyr_rawY, Gyr_rawZ;
float Acceleration_angle[2];
float Gyro_angle[2];
float Total_angle[2];
```

- This is the setup function that is required, and it initializes the drone for flight.

```
void setup() {
  Wire.begin();
  Wire.beginTransmission(0x68);
  Wire.write(0x6B);
  Wire.write(0);
  Wire.endTransmission(true);
  Serial.begin(250000);
  right_prop.attach(3);
  left_prop.attach(5);

  time = millis();
  left_prop.writeMicroseconds(1000);
  right_prop.writeMicroseconds(1000);
  delay(7000);
}
```

- The void function tells commands for all ESCs to follow. The code later performs the PID control for the remainder of the flight (will be explained later in this section).

```
void loop() {
  timePrev = time;
  time = millis();
  elapsedTime = (time - timePrev) / 1000;

  // Check current stage and perform actions accordingly
  switch (stage) {
    case 0: // Full speed for 2 seconds
      if (millis() - stageStartTime < 2000) {
        throttle = 1050;
      } else {
        stageStartTime = millis();
        stage++;
      }
      break;
    case 1: // Half speed for 4 seconds
      if (millis() - stageStartTime < 4000) {
        throttle = 1050; // Half speed
      } else {
        stageStartTime = millis();
        stage++;
      }
      break;
    case 2: // Complete stop
      if (millis() - stageStartTime < 2000) {
        throttle = 1000; // Min speed
        // Set both motors to stop
        left_prop.writeMicroseconds(1000);
        right_prop.writeMicroseconds(1000);
      } else {
        stageStartTime = millis();
        stage = 0; // Reset to first stage
        delay(999999);
      }
      break;
  }
```

- Perform PID control (stabilizes drone during flight with the MPU-6050 chip).

```
void performPIDControl() {
  Wire.beginTransmission(0x68);
  Wire.write(0x3B);
  Wire.endTransmission(false);
  Wire.requestFrom(0x68, 6, true);

  Acc_rawX = Wire.read() << 8 | Wire.read();
  Acc_rawY = Wire.read() << 8 | Wire.read();
  Acc_rawZ = Wire.read() << 8 | Wire.read();

  Acceleration_angle[0] = atan((Acc_rawY / 16384.0) / sqrt(pow((Acc_rawX / 16384.0), 2) + pow((Acc_rawZ / 16384.0), 2))) * rad_to_deg;
  Acceleration_angle[1] = atan(-1 * (Acc_rawX / 16384.0) / sqrt(pow((Acc_rawY / 16384.0), 2) + pow((Acc_rawZ / 16384.0), 2))) * rad_to_deg;

  Wire.beginTransmission(0x68);
  Wire.write(0x43);
  Wire.endTransmission(false);
  Wire.requestFrom(0x68, 4, true);

  Gyr_rawX = Wire.read() << 8 | Wire.read();
  Gyr_rawY = Wire.read() << 8 | Wire.read();

  Gyro_angle[0] = Gyr_rawX / 131.0;
  Gyro_angle[1] = Gyr_rawY / 131.0;

  Total_angle[0] = 0.98 * (Total_angle[0] + Gyro_angle[0] * elapsedTime) + 0.02 * Acceleration_angle[0];
  Total_angle[1] = 0.98 * (Total_angle[1] + Gyro_angle[1] * elapsedTime) + 0.02 * Acceleration_angle[1];

  error = Total_angle[1] - desired_angle;

  pid_p = kp * error;

  if (-3 < error && error < 3) {
    pid_i = pid_i + (ki * error);
  }

  pid_d = kd * ((error - previous_error) / elapsedTime);

  PID = pid_p + pid_i + pid_d;
```

Soil Moisture Code

- Import the proper Python libraries that are needed.
- Find the correct threshold for the project and the soil moisture pin

```python
import RPi.GPIO as GPIO
import time
```

```python
soilMoisturePin = 17

dryThreshold = 600
wetThreshold = 300
```

- Setup the Raspberry Pi to receive soil moisture values from the sensor.

```python
dryThreshold = 600
wetThreshold = 300

def setup():
    GPIO.setmode(GPIO.BCM)
    GPIO.setup(soilMoisturePin, GPIO.IN)

def loop():
    try:
        while True:
            soilMoistureValue = analogRead(soilMoisturePin)
            print("Soil Moisture Value:", soilMoistureValue)

            if soilMoistureValue >= dryThreshold:
                print("Soil is dry.")
            elif soilMoistureValue <= wetThreshold:
                print("Soil is wet enough.")
            else:
                print("Soil moisture is within acceptable range.")

            time.sleep(1)

    except KeyboardInterrupt:
        print("Exiting...")

    finally:
        GPIO.cleanup()
```

- Read the analog input sent from the soil moisture sensor to the Raspberry Pi.

```python
def loop():
    try:
        while True:
            soilMoistureValue = analogRead(soilMoisturePin)
            print("Soil Moisture Value:", soilMoistureValue)

            if soilMoistureValue >= dryThreshold:
                print("Soil is dry.")
            elif soilMoistureValue <= wetThreshold:
                print("Soil is wet enough.")
            else:
                print("Soil moisture is within acceptable range.")

            time.sleep(1)

    except KeyboardInterrupt:
        print("Exiting...")

    finally:
        GPIO.cleanup()
```

- Run both functions at the end of the code.

```python
if __name__ == '__main__':
    setup()
    loop()
```

Building, Testing, and Functionality

The organization of the new drone had to be majorly altered, as new components such as the LiPo batteries and the ESC's had to be stored. The LiPo batteries were on the bottom of 2 of the drone's arms, while the ESC's were located on the top of each arm. One Arduino was stored at the bottom of the drone, alongside the Arduino batteries, and one Arduino and the Raspberry Pi were stored on the top of the drone. Overall, this design proved to be successful in our testing and overall stability.

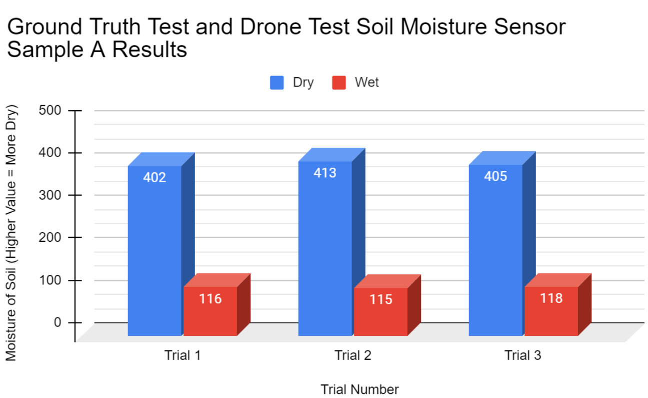
The functionality of the drone had completely changed. With the new additions of the brushless motors and the gyroscope, the flying process was much smoother, and allowed the drone to make more precise movements that once couldn't be performed. Our new drone, although somewhat heavy, could achieve lift and could easily hover in the air, allowing us to accurately obtain the soil moisture of plants without any interruptions. However, the testing phase of the drone was somewhat different. The ESC's had to be calibrated every now and then or the motors would malfunction. The calibration was a huge part of the drone, as without calibrating the ESC's, the drone would not have been able to fly.

Overall, this prototype proved to be the most successful out of the 3, and we have surpassed both of our problem statements with the new prototype. The agricultural drone is not only easy to automate, but also easy to maneuver, as the gyroscope is able to control the drone and maintain its balance.

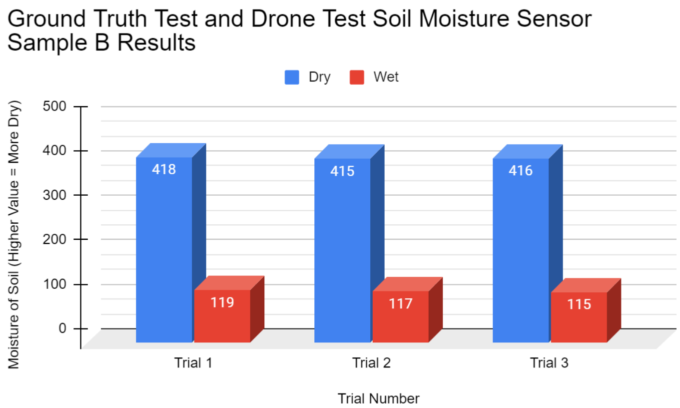
Analysis

The testing phase of the drone will consist of the drone flying in the air, hovering, obtaining the soil moisture of the plant via the soil moisture sensor near the bottom of the drone, and then landing. We will also be taking a 'ground truth test', which is when the soil moisture sensor is tested without the drone to ensure that the drone provides accurate and informative data. We will be using this test to represent our traditional farming method, or the method used by humans. We will be timing both processes to see how long the drone takes to measure the soil moisture of the plants, compare it to our ground truth tests taken previously, and to conclude if the drone was more efficient than without the drone so that we can conclude if the drone is able to perform better in terms of speed compared to traditional methods. Finally, we will be testing the soil moisture for 3 plants and 3 trials for each, each trial testing both dry and wet soil.

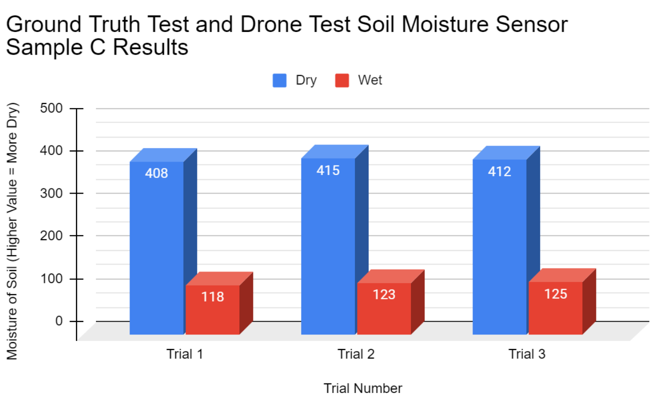
Each of the prototypes gave us the same soil moisture values, as shown below:



SAMPLE 1	Dry	Wet
Trial 1	402	116
Trial 2	413	115
Trial 3	405	118



SAMPLE 2	Dry	Wet
Trial 1	418	119
Trial 2	415	117
Trial 3	416	115

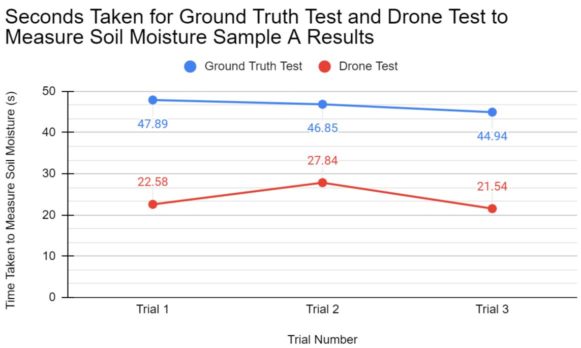


SAMPLE 3	Dry	Wet
Trial 1	408	118
Trial 2	415	123
Trial 3	412	125

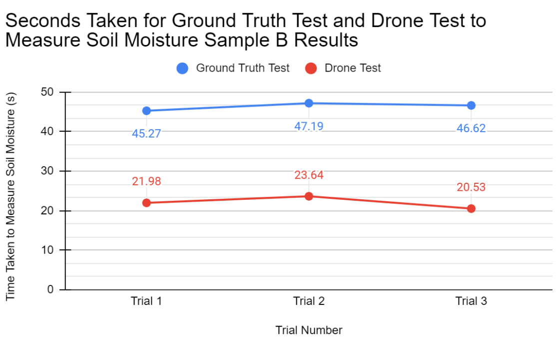
Prototypes 1 and 2

Prototypes 1 and 2 had the same results, as little to no changes were made regarding the motors and overall functionality (other than the camera). Our data from prototypes 1 and 2 showed that the drone was about 50% faster as compared to the ground truth tests. However, the drone did waste time as it took a while to get the drone off the ground and to maintain its hovering position. With this information, we decided to completely transform prototype 3 into something completely new, with many powerful motors that would help us to achieve that lift in a short amount of time.

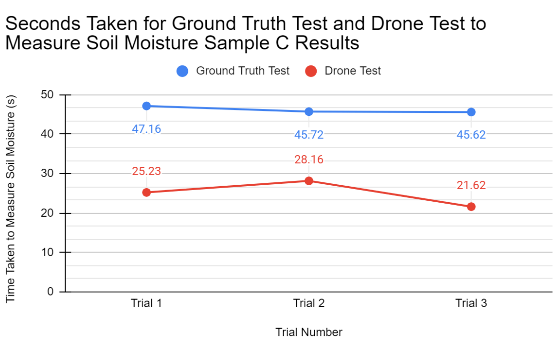
Despite the drone lacking in generating lift accordingly, the results were still informative and allowed us to come to a solid conclusion - the drone was cost-efficient and was relatively cheaper than industrial agri-drones, however the drone, although faster than human methods, was still quite slow and could be much faster with more enhancements. The results are shown below.



SAMPLE 1	Ground Truth Test	Drone Test
Trial 1	47.89	22.58
Trial 2	46.85	27.84
Trial 3	44.94	21.54



SAMPLE 2	Ground Truth Test	Drone Test
Trial 1	45.27	21.98
Trial 2	47.19	23.64
Trial 3	46.62	20.53

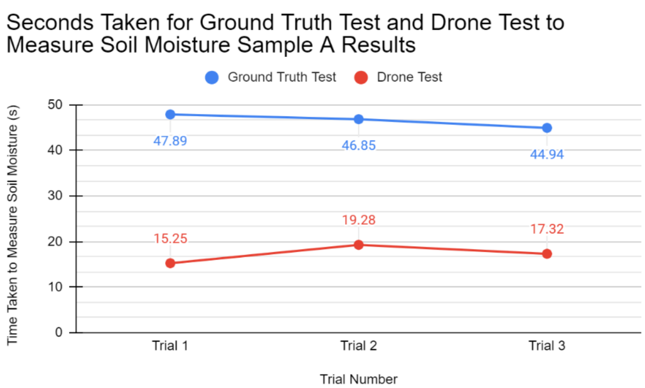


SAMPLE 3	Ground Truth Test	Drone Test
Trial 1	47.16	25.23
Trial 2	45.72	28.16
Trial 3	45.62	21.62

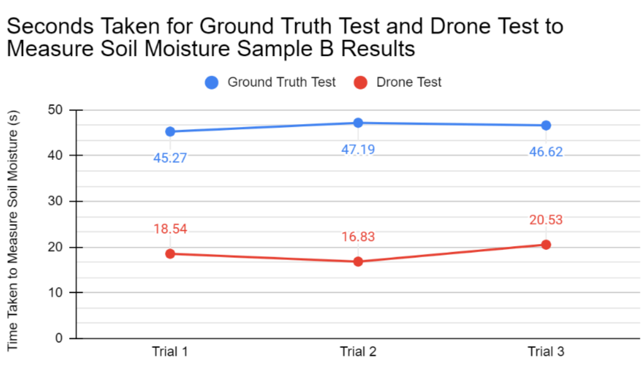
Prototype 3

Prototype 3 took much less time to measure the soil moisture of the soil as compared to the previous prototypes. The drone proved to be approximately 75% faster as compared to the ground truth tests. The drone took much less time to generate lift, and as a result, significantly decreased the time needed to hover in the air. Our new prototype showed that, through the use of drone technology, people can have a chance at accessing budget-friendly, advanced farming equipment so that they can be able to save time and money on expensive farming equipment. To conclude, our prototype successfully passed our initial problem statements, and overall was efficiently able to gather reliable and accurate data that we could use to innovate and expand the world of farming with.

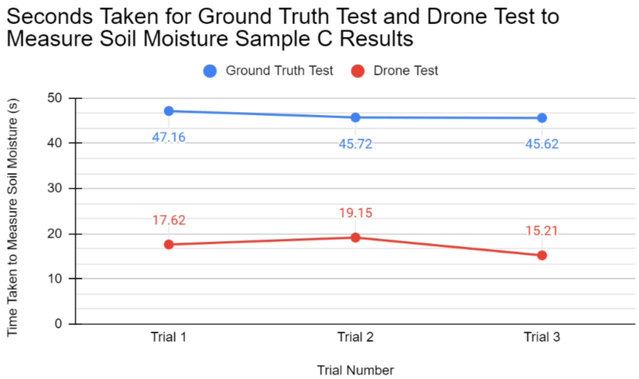
Below is the data collected for prototype 3 (keep in mind that the same ground truth test values were used for these ones as well).



SAMPLE 1	Ground Truth Test	Drone Test
	47.89	15.25
Trial 2	46.85	19.28
Trial 3	44.94	17.32



SAMPLE 2	Ground Truth Test	Drone Test
Trial 1	45.27	18.54
Trial 2	47.19	16.83
Trial 3	46.62	20.53



SAMPLE 3	Ground Truth Test	Drone Test
Trial 1	47.16	17.62
Trial 2	45.72	19.15
Trial 3	45.62	15.21

Conclusion

1. Yes, it is possible to create a budget friendly drone with similar functions as those in current society, as demonstrated in our experiment. In current society, drones are used to scout plants and monitor health, monitor livestock, release pesticides, etc. Our project showed that it is possible to create budget friendly drones with increased payloads, quality cameras and sensors, and shows that the future of agriculture is rapidly approaching, and that farmers around the world will soon have access to this groundbreaking technology.

2. Yes, a drone is able to perform better in terms of speed as compared to traditional farming methods. Traditional farming methods mainly relied on the use of manual labor and organic pesticides, however, our drone shows that using more advanced farming methods will not only have an increased benefit on the crops, but also the farmers themselves. Our drone has the ability to work autonomously, meaning that farmers do not need to control the drone. Also, the drone is able to do all the functions the farmer requires 24/7, as the drone does not need breaks and such. So, with our drone, farming will improve in many aspects and will be incredibly efficient.

Citations

Advantages of drones. (2021, May 30). COPTER.BG. Retrieved November 10, 2023, from https://www.copter.bg/en/blog/advantages-of-drones-and-drone-technology

Ahirwar, S. (2019, January 10). *Application of Drone in Agriculture*. https://www.researchgate.net/profile/Srinivas-4/publication/331017387_Application_of_Drone_in_Agriculture/links/5c6aacb04585156b57036824/Application-of-Drone-in-Agriculture.pdf

Allain, R. (2017, May 19). *The Physics of How Drones Fly*. WIRED. https://www.wired.com/2017/05/the-physics-of-drones/

Design and Development of Drone for Agricultural Applications. (n.d.). IJLEMR. http://ijlemr.com/papers/volume2-issue7/18-IJLEMR-22328.pdf

Drone Technology In Agriculture. (n.d.). Croptracker. Retrieved November 10, 2023, from https://www.croptracker.com/blog/drone-technology-in-agriculture.html

Dupuis, A. (2023, June 12). *Environmental Impact of Traditional Farming: 5 Effects*. Eden Green. https://www.edengreen.com/blog-collection/environmental-impact-of-traditional-and-vertical-farming-2021-report

Farm With A View: How Drone Technology Is Taking Agriculture To A New Level. (2023, February 23). Forbes. Retrieved November 10, 2023, from https://www.forbes.com/sites/stevensavage/2023/02/23/farm-with-a-view-how-drone-technology-is-taking-agriculture-to-a-new-level/?sh=2f05b3804962

Jao, E. (2023, July 31). *Traditional vs. Modern Farming Techniques: A Comparative Analysis*. Milk and Honey Ranch. https://milkandhoneyranch.com/gardening-and-farming/traditional-vs-modern-farming-techniques-a-comparative-analysis/

Johnston, M. (2023, September 7). *Thrust and Newton's Third Law in Aviation - Cal Aero Blog*. California Aeronautical University. https://calaero.edu/newtons-third-law-in-aviation/

Kamprad, D. (n.d.). *Urban Farming vs Traditional Farming: What's the Difference?* Impactful Ninja. https://impactful.ninja/urban-vs-traditional-farming-differences/

McSwine, D. (2023, November 7). *Implementation of drone technology for farm monitoring & pesticide spraying: A review*. https://www.sciencedirect.com/science/article/pii/S2214317322000087

McSwine, D. (2023, November 7). *Traditional Farming Practices and Its Consequences*. https://link.springer.com/chapter/10.1007/978-3-030-61010-4_6

STEM LEARNING: Advanced Air Mobility: The Science Behind Quadcopters Reader—Student Guide. (n.d.). NASA. https://www.nasa.gov/wp-content/uploads/2020/05/aam-science-behind-quadcopters-reader-student-guide_0.pdf

Daly, D. (n.d.). *How Farmers Are Using Drones in Agriculture*. Consortiq. https://consortiq.com/uas-resources/using-drones-in-agriculture-industry

DRONES, S. (2023, September 7). *Are agriculture drones cheap, the best cost, and where is the location?* Medium. https://medium.com/@samlisisi456/are-agriculture-drones-cheap-the-best-cost-a d-where-is-the-location-f253b90cf59d