# Agricultural image AI analysis: Healthy vegetation identification by drone

# 1. Plan goals

The goal of this project is to develop an artificial intelligence (AI) agricultural monitoring system based on drone images to quickly identify the vegetation health status of farmland or green areas. The system will use high-resolution image data and AI technology to automatically distinguish healthy vegetation from withered vegetation, achieve early problem detection, and provide data-driven decision support for agricultural managers, thereby improving management efficiency and accuracy and reducing pests and diseases and agricultural losses. .

# Project background

Plant health directly affects crop yield and quality. Factors such as pests and diseases, malnutrition or environmental stress may lead to a decline in vegetation health and even cause economic losses. In traditional agricultural monitoring methods, farmers usually rely on manual inspections to observe vegetation conditions. This method is not only time-consuming and labor-intensive, but also limited by human resources and observation accuracy. Especially in the management of large areas of farmland, efficiency issues are particularly prominent.

With the rapid development of drone technology, its application in the agricultural field is becoming more and more widespread. Drones can quickly and efficiently cover large areas of farmland and capture high-resolution image data, providing an important data source for agricultural monitoring. However, it is difficult to complete in-depth analysis of vegetation health using image data alone, which requires the support of artificial intelligence (AI) technology. AI-based image recognition technology, especially convolutional neural networks (CNN), has excellent performance in extracting image features and pattern recognition, and can be used to accurately distinguish the characteristics of healthy vegetation and withered vegetation, achieving fully automated vegetation health analysis.

## Needs and Challenges

- **Efficiency needs** : Manual inspections are time-consuming and inefficient, and efficient monitoring methods are urgently needed.
- **Accuracy requirements** : Traditional monitoring methods are susceptible to human factors and lack recognition accuracy, so they need to be supplemented by technical means.
- **Demand for scale** : With the large-scale development of agriculture, the monitoring scope is getting larger and larger. The combination of drones and AI technology is an effective way to solve the problem.

# 2. Solution proposal

With the rapid development of large-scale agriculture, efficient monitoring of crop health has become a key link in modern agricultural management. However, the traditional method of relying on manual inspections faces the following challenges:

- **Time and cost burden:** Manual inspections in large areas of farmland consume a lot of time and human resources.
- **Insufficient accuracy:** Human observation is

affected by experience and environment, and is prone to omissions or misjudgments.

- **Slow response:** Pests, diseases or environmental stress may spread rapidly, making it difficult for traditional methods to detect and respond in time.

Drones combined with artificial intelligence (AI) technology provide a new solution that can quickly cover large areas of farmland and automatically analyze vegetation health. This technology can not only reduce human input, but also improve monitoring accuracy and provide reliable decision-making support for agricultural production.

## 1. Project goals

This project aims to develop an AI agricultural monitoring system based on drone images to achieve the following goals:

- **Rapid health detection:** Automatically identify healthy and withered vegetation to achieve efficient monitoring of vegetation health.

- **Improve accuracy:** Use AI technology for precise image analysis to reduce errors in manual monitoring.
- **Data-driven decision-making:** Generate visual reports to help agricultural managers take timely response measures.

## 2. Solution details

### 2.1 System architecture design

The entire system consists of the following modules:

- **Drone equipment** : Drones equipped with high-resolution photography equipment for collecting farmland image data.
- **Image processing module** : Use image pre-processing techniques (such as cropping, normalization, noise reduction) to improve image quality.
- **AI model** : Image recognition model based on

convolutional neural network (CNN) for automatically distinguishing healthy and withered vegetation.

- **Data display and analysis** : Generate intuitive health status maps and reports to help users quickly understand monitoring results.

## 2.2 System implementation steps

A. **Technical preparation** :

- Select a drone to ensure it has stable flight capabilities and high-quality image shooting capabilities.
- Choose an appropriate AI technology framework (such as TensorFlow or PyTorch ) and design a preliminary model architecture.

B. **Data collection and annotation** :

- Use drones to capture image data of healthy and dead vegetation.
- Manually label data to provide

standardized samples for model training and testing.

C. **Model training and optimization**:

- annotation data to ensure the consistency of the image data.
- Use a CNN model for training, and improve accuracy by repeatedly adjusting parameters.

D. **System integration and testing**:

- Deploy the AI model to the drone control system to achieve real-time analysis.
- Conduct tests in a small area of farmland to ensure the flight stability of the drone and the reliability of the analysis results.

### 2.3 Solving technical challenges

- **High-quality data requirements**: improve the generalization ability of the model through diversified data collection covering different crops and environmental conditions.

- **Real-time analysis performance** : Optimize the model structure and deployment method to ensure that the drone can quickly process image data during flight.
- **Visualize results** : Design intuitive health maps and reporting systems that are easy for users to interpret.

3. Expected benefits

- **Improve efficiency** : Drones combined with AI technology can quickly complete health monitoring of large areas of farmland, significantly reducing labor costs.
- **Improve accuracy** : AI models can accurately identify plant health conditions, reduce human misjudgments, and improve monitoring quality.
- **Optimize agricultural management** : By generating instant health reports, it helps agricultural managers take effective measures

early to reduce losses from pests and diseases.

4. Future expansion

- **Multi-crop adaptation** : Enhance the model's ability to identify different crops and cover more agricultural scenarios.
- **Combining environmental data** : integrating weather, soil moisture and other data to provide more comprehensive health analysis and prediction.
- **Automated management suggestions** : Automatically generate management suggestions for fertilization, irrigation, etc. based on health status and environmental data.

# 3. Use generative artificial intelligence to solve the problem of vegetation health identification

## Generative AI solutions detailed

1. Data amplification and generation

Generative adversarial networks (GANs) are a typical generative AI model that can be used to create diverse vegetation image data.

- **Data collection and preprocessing** : Collect image data of healthy and withered vegetation captured by drones as the basis for training.
- **Data generation** : Use GAN models to generate simulated images covering a variety of health conditions and environmental conditions. These generated images can increase the diversity of the data set and improve the adaptability of the AI model to different scenarios.
- **Apply enhanced images** : Use the generated images as part of the training samples to improve the generalization ability of the model.


```python
import numpy as np
import matplotlib.pyplot as plt
from scipy.ndimage import gaussian_filter
import random

class VegetationHealthSimulation:
    def __init__(self, width=256, height=256):
        self.width = width
        self.height = height

    def generate_vegetation_image(self, health_status='healthy'):
        """
        模擬生成不同健康狀態的植被影像
        健康狀態：healthy, mild_withered, severe_withered
        """
        # 建立基礎綠色背景
        image = np.zeros((self.height, self.width, 3), dtype=np.float32)

        # 根據健康狀態調整顏色和紋理
        if health_status == 'healthy':
            # 深綠色，均勻紋理
            base_green = 0.6 + np.random.random() * 0.2
            image[:,:,1] = base_green  # 綠色通道

            # 添加輕微紋理
            noise = np.random.random((self.height, self.width)) * 0.1
            image[:,:,1] += noise

        elif health_status == 'mild_withered':
            # 淺綠色混合黃色，不均勻紋理
```

2. Image enhancement and noise reduction

Generative AI can also be used to process drone images to improve system analysis accuracy:

- Noise removal : Use an autoencoder or a dedicated denoising model to remove environmental noise in the image , such as weather effects or shooting shake.
- Image completion : For possible missing areas in drone photography, the generative model can

complete the missing parts and provide complete analysis data.


```python
        return image

    def add_drone_noise(self, image):
        """
        模擬無人機拍攝時的環境噪音
        """
        # 高斯模糊
        blurred = gaussian_filter(image, sigma=1)

        # 添加雜訊
        noise = np.random.normal(0, 0.05, image.shape)
        noisy_image = blurred + noise

        # 剪裁回有效範圍
        noisy_image = np.clip(noisy_image, 0, 1)

        return noisy_image

    def denoise_image(self, noisy_image):
        """
        簡單去噪方法：高斯濾波
        """
        return gaussian_filter(noisy_image, sigma=0.5)

    def analyze_vegetation_health(self, image):
        """
        根據影像特徵分析植被健康狀態
        """
        # 計算綠色和紅色通道比例
        green_ratio = np.mean(image[:,:,1])
```

## 3. Health indicator map generation

Image processing technology based on generative AI transforms vegetation health analysis results into health indicator maps:

- Labeling of classification results : Combined with AI model analysis, healthy and withered vegetation are marked in different colors on the map.

- **Generate suggested areas** : Use generative AI to simulate vegetation recovery conditions under different fertilization or irrigation conditions to provide reference for agricultural managers.

4. Technical architecture of the solution

- **Data collection and processing** : The drone captures images and performs basic cropping , standardization and denoising processing.
- **Data generation and enhancement** : Use GAN models to generate more diverse images of healthy and withered vegetation.
- **AI model analysis** : image recognition technology based on convolutional neural network (CNN) to identify health conditions.
- **Results visualization** : Use generative AI to transform analysis results into health indicator maps and data reports.

Technical framework diagram (can be inserted):

- Input data → Data processing and generation → AI analysis model → Health map and reporting

5. Expected benefits

- Improve data quality : Solve data shortage and quality issues by generating high-quality images.
- Improve analysis accuracy : Significantly improve model accuracy with enhanced imagery and more comprehensive training data.
- Promote management decisions : Generate intuitive health maps and recovery simulations to help agricultural managers make more scientific decisions.

## Future expansion direction

- Cross-crop model adaptation : Optimize generative AI to adapt to the characteristics and needs of multiple crops.
- Intelligent decision-making suggestions :

Combined with generative AI simulation, it provides intelligent irrigation, fertilization and other management suggestions.

- **Real-time monitoring application**: further optimize the generative AI algorithm to achieve real-time image generation and analysis during drone flight.

</> 無人機植被健康模擬系統
Click to open code

這個模擬程式展示了無人機植被健康檢測的完整流程：

1. 植被影像生成
    - 模擬三種健康狀態：健康、輕度枯萎、嚴重枯萎
    - 使用顏色和紋理變化表現不同健康程度
2. 無人機拍攝模擬
    - 添加高斯模糊和雜訊
    - 模擬真實拍攝環境中的影像失真
3. 去噪處理
    - 使用高斯濾波減少雜訊
4. 健康狀態分析
    - 根據綠色和紅色通道比例判斷植被健康程度
    - 生成對應顏色的健康指標地圖

程式將展示：

- 原始植被影像
- 模擬的無人機拍攝雜訊影像
- 對應的健康狀態指標地圖

運行後將顯示三種不同健康狀態植被的模擬結果。

```
模擬會生成9張子圖，分為3組，每組包含：
1. 原始植被影像
2. 添加無人機拍攝雜訊的影像
3. 健康狀態指標地圖

三種植被健康狀態：
- 健康植被：綠色為主
- 輕度枯萎：綠色與黃色混合
- 嚴重枯萎：棕色和黃褐色為主

健康狀態指標地圖顏色：
- 健康：綠色
- 輕度枯萎：黃色
- 嚴重枯萎：紅色

程式會模擬無人機拍攝過程中可能出現的噪音和失真，並嘗試通過去噪技術恢復影像。
```

Program code

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.ndimage import gaussian_filter
import random

class VegetationHealthSimulation :
    def __init __(self, width=256,
    height=256) :
        self.width = width
        self.height = height

    def generate_vegetation_image (self,
    health_status ='healthy'):
        """
        Simulate and generate vegetation images
        in different health states
        Health status: healthy , mild_withered ,
```

```
severe_withered
"""
# Create a basic green background
image = np.zeros (( self.height ,
self.width , 3), dtype =np.float32)

# Adjust colors and textures based on
health status
if health_status == 'healthy':
# Dark green, uniform texture
base_green = 0.6 +
np.random.random ( ) * 0.2
image[:,:,1] = base_green #Green channel

# Add slight texture
noise = np.random.random
(( self.height , self.width )) * 0.1
image[ :,:,1] += noise
```

```
elif health_status == '
mild_withered ':
# Light green mixed with yellow, uneven
texture
base_green = 0.4 +
np.random.random ( ) * 0.2
base_yellow = 0.2 +
np.random.random ( ) * 0.1

image[:,:,1] = base_green # Green channel
image[:,:,0] = base_yellow #Red channel

# Add more noise and uneven areas
noise = np.random.random
(( self.height , self.width )) * 0.2
dry_patches =
np.random.random ( ( self.height ,
self.width )) > 0.8
image[ :,:,1] -= dry_patches
```

```
* noise

elif health_status == '
severe_withered ':
# Brown and tawny, lots of dead areas
base_brown = 0.3 +
np.random.random ( ) * 0.2
base_yellow = 0.2 +
np.random.random ( ) * 0.1

image[:,:,0] = base_brown # red channel
image[:,:,1] = base_yellow # Green
channel

# Simulate large-scale withered areas
dry_regions =
np.random.random ( ( self.height ,
self.width )) > 0.6
image[ :,:,1] *= dry_regions
```

```
return image

def add_drone_noise (self, image):
"""
Simulate environmental noise during drone
shooting
"""
# Gaussian blur
blurred = gaussian_filter (image,
sigma=1)

# Add noise
noise = np.random.normal (0,
0.05, image.shape )
noise_image = blurred + noise

# Trim back to valid range
noisy_image = np.clip
```

```
( noisy_image , 0, 1)

        return noisy_image

    def denoise_image (self,
        noisy_image ):
        """
Simple denoising method: Gaussian filter
        """
        return gaussian_filter
( noisy_image , sigma=0.5)

    def analyze_vegetation_health (self,
        image):
        """
Analyze vegetation health status based on
        image features
        """
    # Calculate the green and red channel
```

```
ratio
green_ratio = np.mean
( image[:,:,1])
red_ratio = np.mean
( image[:,:,0])

# Health status judgment rules
if green_ratio > 0.5 and
red_ratio < 0.2:
return 'Healthy'
elif 0.3 < green_ratio <= 0.5 and
0.2 <= red_ratio < 0.4:
return 'Mild Withered'
else :
return 'Severe Withered'

def generate_health_map (self,
images):
"""
```

```
Generate health indicator maps
"""
health_maps = []
for image in images:
health_status =
self.analyze_vegetation_ health ( image)

# Create color mapping
health_map = np.zeros_ like
( image)
if health_status ==
'Healthy':
health_map [:,:] = [0, 1,
0] # Green
elif health_status == 'Mild
Withered':
health_map [:,:] = [1, 1,
0] # Yellow
else :
```

```
health_map [:,:] = [1, 0,
0] # red

health_maps.append
( health_map )

return health_maps

def main():
#Initialize the simulator
sim = VegetationHealthSimulation ()

# Simulate and generate images of
different health states
health_statuses = ['healthy', '
mild_withered ', ' severe_withered ']

# Set up drawing
plt.figure ( figsize =(15, 10))
```

```
for i , status in
enumerate( health_statuses ):
# Generate original vegetation image
original_image =
sim.generate_vegetation_image ( status )

# Add drone shooting noise
noisy_image = sim.add_drone_noise
( original_image )

# denoise
denoised_image =
sim.denoise_image ( noisy_image )

# Generate health indicator map
health_map = sim.generate_health_
map ( [ denoised_image ])[0]
```

```
# drawing
plt.subplot ( 3, 3, i *3+1)
plt.imshow ( original_image )
plt.title (f'Original vegetation
({ status. replace ( "_", " ")})' )
plt.axis ( 'off' )

plt.subplot ( 3, 3, i *3+2)
plt.imshow ( noisy_image )
plt.title ('drone shooting
noise' )
plt.axis ( 'off' )

plt.subplot ( 3, 3, i *3+3)
plt.imshow ( health_map )
plt.title ('Health status
indicator' )
plt.axis ( 'off' )
```

```
plt.tight_ layout ()
plt.show ()

if __name__ == "__main__":
main()
```

https://drive.google.com/file/d/1_-fxNo6ewfVhRnQaiwrBKRGqHDiz1RA2/view?usp=sharing