Remote Sensing of Crop Pathology through Computer Vision

Ankit Yande

Background Info

- Plants are subject to a variety of disorders caused by pathogens and environmental factors
- The ability to detect and diagnose these disorders remotely provides advantages for agriculture
  - Thermography, fluorescence imaging, and hyperspectral techniques are currently the most favorable indirect methods for plant disease detection. (Fang and Ramasamy 2015).
    - All these techniques use satellite imaging to collect and analyze data of the electromagnetic energy reflected off the Earth's surface which is expensive and only feasible for large scale agriculture in developed countries.
    - Computer vision is far more accessible and can be done from small aircrafts or drones (USGS).

Goal

To develop software in Java that can remotely discern crop pathology and distinguish between various stresses through a camera feed by observing changes in hue, saturation, and brightness. The area of the infected area should then be calculated using the known height of the remote sensing device and the area in pixels of the affected zone.

Methods

Phase I) Developing the Image Detection Program

- A Java program was designed to take in the feed from a camera and detect areas within the Hue, Saturation, and Value (HSV) threshold of infected plants. The program broke the video feed from a camera into individual frames that were converted into arrays of HSV values. The program then went through the array and searched for areas within the correct HSV threshold for infected plants to create a mask of the infected areas.
- The effectiveness of this image detection method was tested using computer-generated images of a gradient (Figure 7) and calculating the number of detected pixels within each strip (Table 1).
- The ability of the program to calculate area given distance was tested by taking readings of an object a various distances (Figure 8).

Phase II) Testing the Program's Ability to Identify and Distinguish between Various Stresses

- Flats of oats and radishes were grown at 25ºC in Miracle-Gro soil and watered twice a day. Subsets were then treated with the following stressors:
  - Dehydration: plants stopped being watered after they began to grow
  - Nutrient Deficiency: plants were grown in soil composed of perlite and vermiculite
  - Excess Salinity: plants were watered daily with 50 ml 0.6M NaCl solution daily
  - Cold Stress: plants were placed in freezer for an hour and then allowed to thaw
  - Pests: plants were placed chamber with whiteflies.
- The camera then captured images of the flats with portions exposed to each pathology and the image detection algorithm was run on each image to determine the software's ability to identify the presence of the pathology and distinguish what the cause of stress is. Each test was also done with the addition of a stress detection lense.

```
Scalar min = new Scalar(h, s, v);// HSV
Scalar max = new Scalar(H, S, V);// HSV

Mat imgThreshold = inputframe.clone();

// Detect the pixels in the range between the scalars min and max
Core.inRange(imgThreshold, min, max, imgThreshold);
```

Figure 1) The maximum and minimum HSV values were determined of the object that was to be detected. A threshold matrix was created that served as a mask of the pixels in the correct range. The pixels within the range turned white and the rest turned black

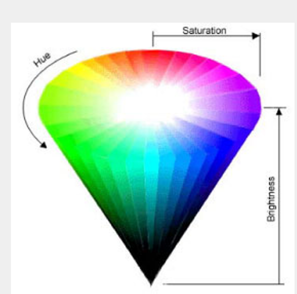
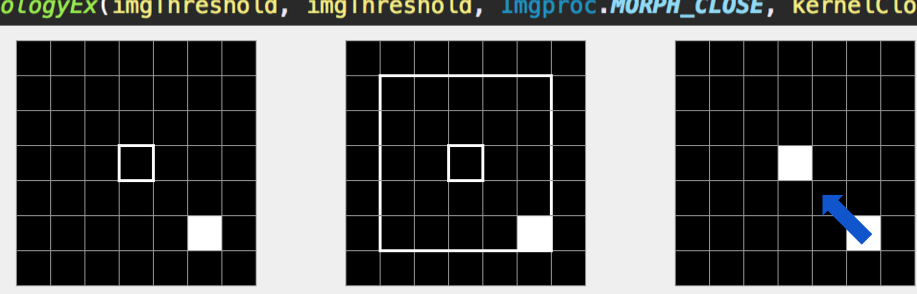


Figure 2) A visual representation of Hue Saturation and Brightness (Jewett, www.tomjewett.com).

```
Mat kernelOpen = Mat.ones(15, 20, 0);
Mat kernelClose = Mat.ones(50, 50, 0);
Imgproc.morphologyEx(imgThreshold, imgThreshold, Imgproc.MORPH_OPEN, kernelOpen);
Imgproc.morphologyEx(imgThreshold, imgThreshold, Imgproc.MORPH_CLOSE, kernelClose);
```



Figures 3&4) Matrices were created that were used to erode and dilate the mask. This method works by looping through each pixel (the anchor) and filling it with the highest brightness of a pixel within the kernel range around it. In the example the anchor is the middle square and the kernel is a 3x3

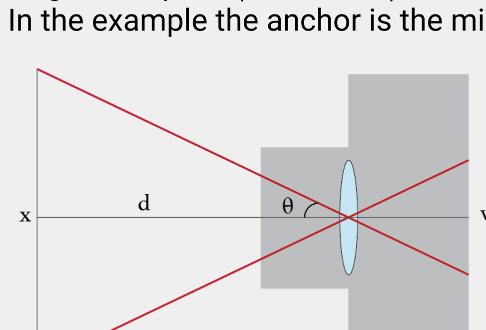


Figure 5) Diagram of how a camera captures images.
The area of a real world object can be calculated as:

$$A = a_{pixel}\left(\frac{w_{px}/h_{px}}{w_{px}h_{px}}\right) \cdot (2d\sin\theta)^2$$

where w_{px} is the width in pixels of the object, h_{px} is the height in pixels of the object, d is the distance from the object and θ is half the view angle of the camera

```
// Find the contours of the mask
Mat mat1 = new Mat();
List<MatOfPoint> contours = new ArrayList<MatOfPoint>();
Imgproc.findContours(imgThreshold, contours, mat1, Imgproc.RETR_EXTERNAL, Imgproc.CHAIN_APPROX_NONE);

//Create a rectangle around the large contour areas
Mat mat2 = new Mat();
Rect rect = null;
for (int i = 0; i < contours.size(); i++) {
    rect = Imgproc.boundingRect(contours.get(i));

    if (rect.width * rect.height > 15000 ) {
        Imgproc.rectangle(inputframe, rect.tl(), rect.br(), new Scalar(0, 255, 255), 5);
        Range xr = new Range((int) rect.tl().x,(int) rect.br().x);
        Range yr = new Range((int) rect.tl().y,(int) rect.br().y);
        mat2= new Mat(imgThreshold,yr,xr);
    }

}

//count the number of white pixels
double areaTot = Core.countNonZero(imgThreshold);
double areaDet = Core.countNonZero(mat2);
```

Figure 6) Using OpenCV's findContours method, an Array of points were created that outlined the white areas of the mask. A bounding rectangle was then created using the values of the contour and the number of pixels within the contours were calculated.

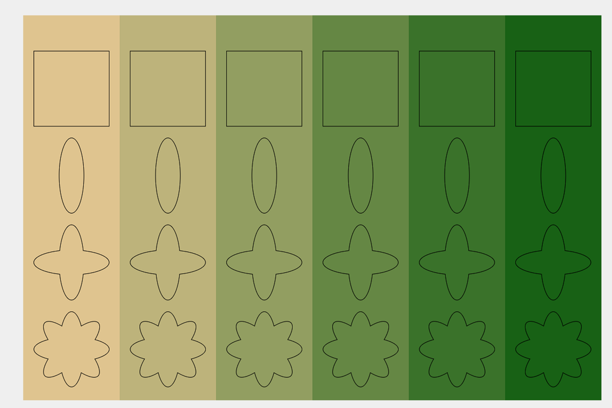


Figure 7) The computer-generated image of a gradient of HSV values within the range of healthy to diseased plants.

Figure 8) The setup used to test the program's ability to calculate the area of an object. The camera and detection object were placed on a track with marking to measure the distance between the camera and object

Results

Table 1: Pixels Detected in each Strip of the Gradient Image

	Detected Pixels	Actual Pixels	%Error
Strip 1	62129	62000	0.21%
Strip 2	63586	63350	0.37%
Strip 3	63753	63450	0.48%
Strip 4	63122	63000	0.19%
Strip 5	62333	62200	0.21%
Strip 6	62246	62000	0.40%

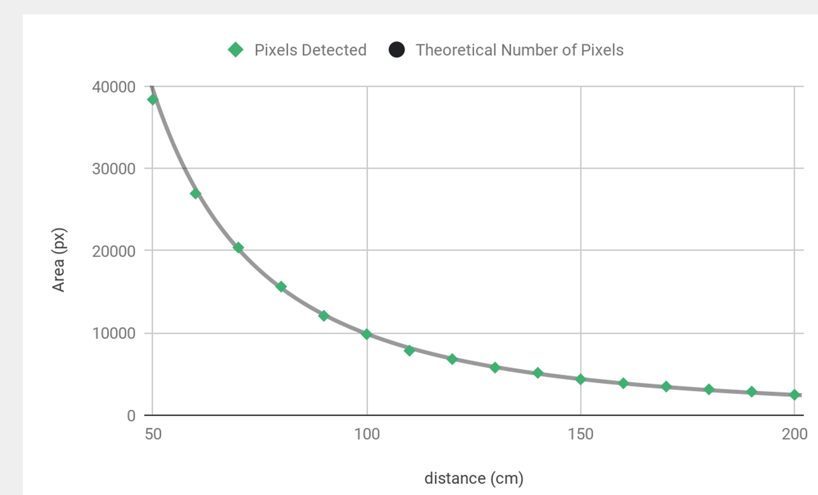


Figure 9) A plot of the number of pixels detected at each length from 50-100cm. All the computer calculated distances calculated were within 5% error. According to a paired, two-tailed T-Test, the difference between the samples was not statistically significant (p> 0.05).

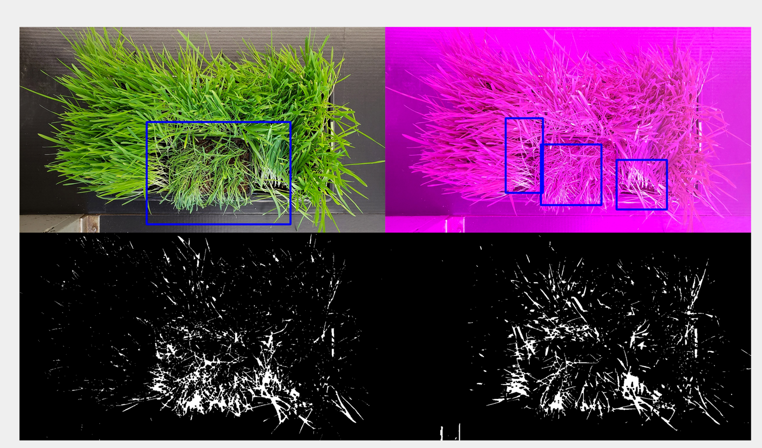


Figure 10) Oats treated with Dehydration
Raw Image
- Ratio of detected pixels within range: 69.69%
- HSV: min: (60, 20, 194) max: (99, 118, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 14.99%
- HSV: min: (0,0,0) max: (255, 154, 255)

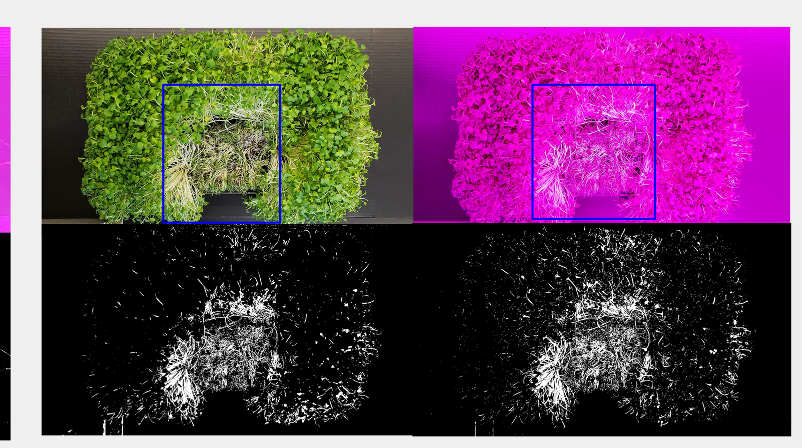


Figure 11) Radish treated with Dehydration
Raw Image
- Ratio of detected pixels within range: 74.40%
- HSV: min: (0, 0, 151) max: (255, 94, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 67.96%
- HSV: min: (0,0,0) max: (255, 203, 255)

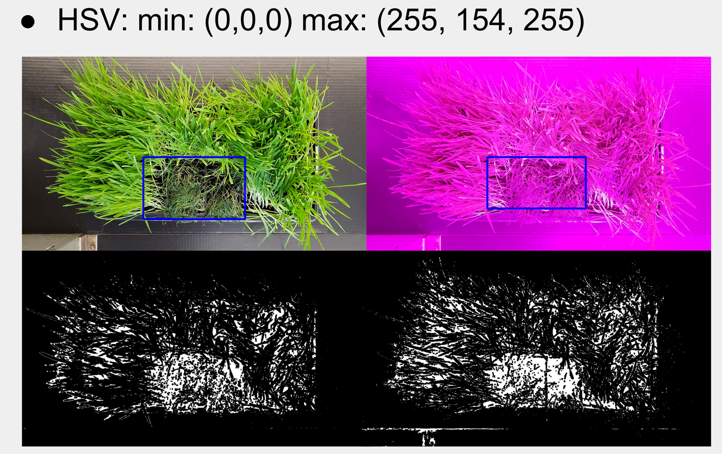


Figure 12) Oats treated with Cold Stress
Raw Image
- Ratio of detected pixels within range: 41.58%
- HSV: min: (0, 98, 0) max: (90, 255, 55)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 32.51%
- HSV: min: (0,244,0) max: (148, 255, 255)

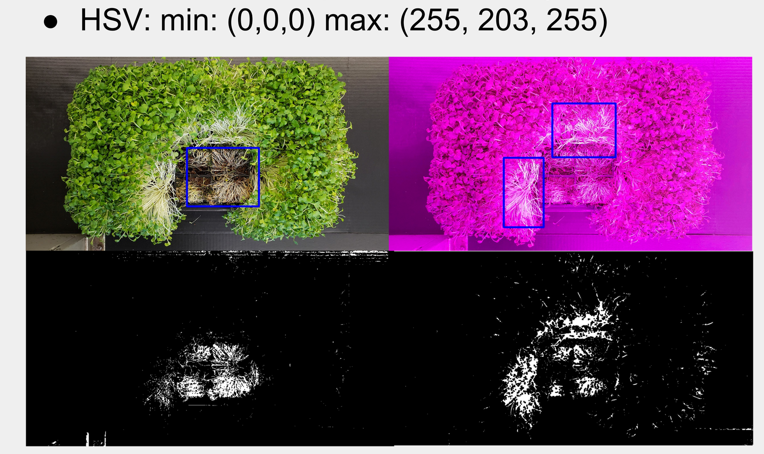


Figure 13) Radish treated with Cold Stress
Raw Image
- Ratio of detected pixels within range: 65.29%
- HSV: min: (94, 8, 139) max: (255, 255, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 31.10%
- HSV: min: (0,0,0) max: (255, 173, 255)

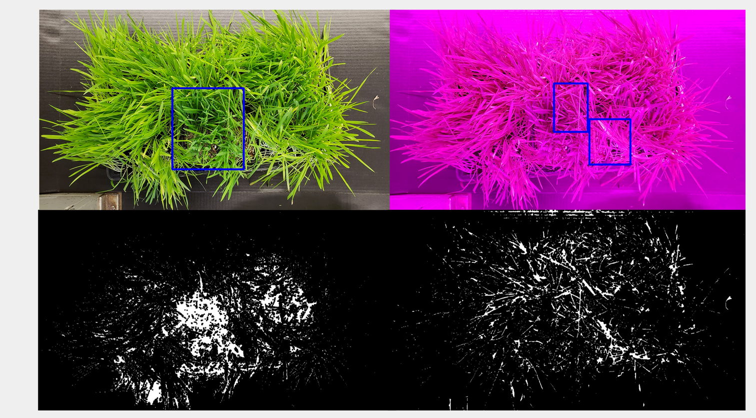


Figure 14) Oats treated with low nutrient soil
Raw Image
- Ratio of detected pixels within range: 37.27%
- HSV: min: (0, 132, 66) max: (71, 255, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 7.59%
- HSV: min: (0,0,0) max: (250, 198, 255)

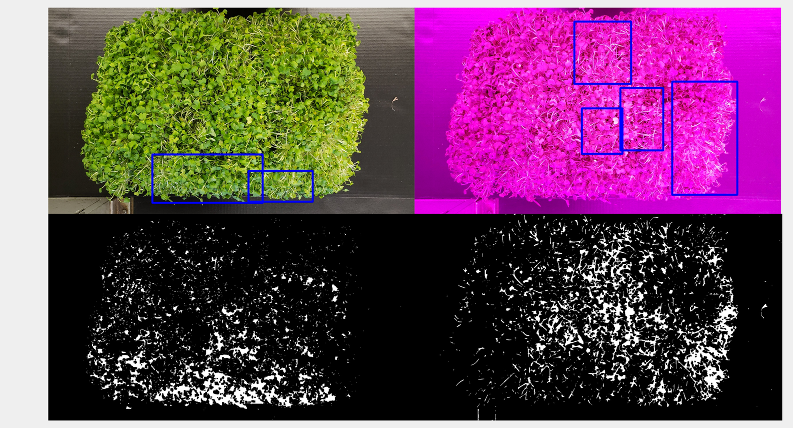


Figure 15) Radish treated with low nutrient soil
Raw Image
- Ratio of detected pixels within range: 30.21%
- HSV: min: (0, 0, 63) max: (76, 255, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 12.00%
- HSV: min: (0,0,0) max: (255, 178, 255)

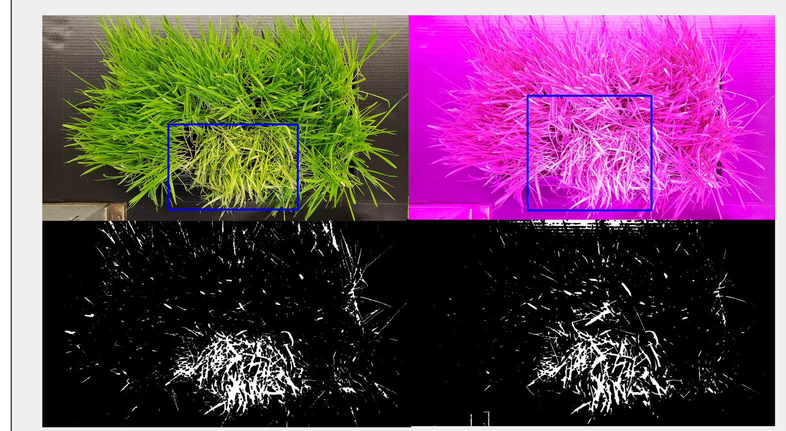


Figure 16) Oats treated with excess salinity
Raw Image
- Ratio of detected pixels within range: 78.68%
- HSV: min: (87, 77, 186) max: (255, 255, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 63.98%
- HSV: min: (0,0,0) max: (255, 88, 255)

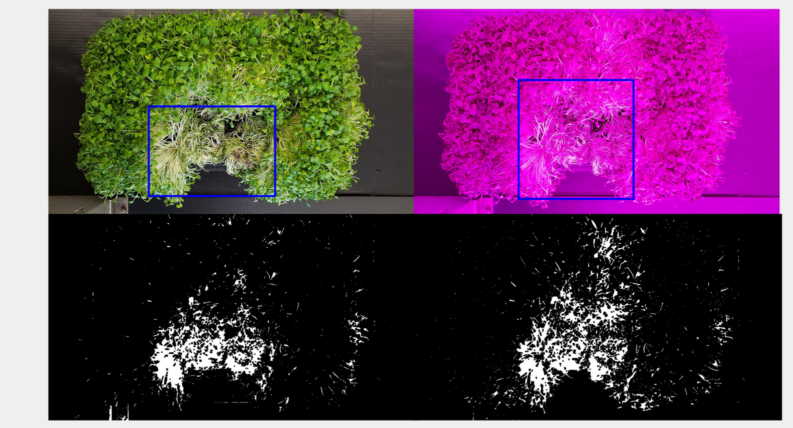


Figure 17) Radish treated with excess salinity
Raw Image
- Ratio of detected pixels within range: 81.96%
- HSV: min: (82, 0, 140) max: (255, 93, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 77.88%
- HSV: min: (0,0,0) max: (255, 195, 255)

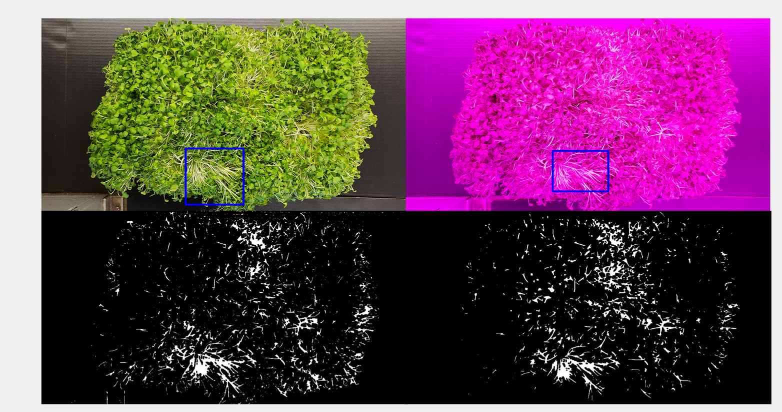


Figure 18) Radishes treated with whiteflies
Raw Image
- Ratio of detected pixels within range: 20.78%
- HSV: min: (0, 0, 192) max: (255, 121, 255)

Image with Stress Detection Glasses
- Ratio of detected pixels within range: 17.19%
- HSV: min: (0,0,0) max: (255, 168, 255)

Figure 19) More matured Radishes and Oats with chlorosis due to lack of nutrients. In this case, the addition of a stress detection lense lead to a better detection of the the affected area.

Discussion

Results suggest that computer vision analysis of diseased plant imagery provides a viable method of remotely diagnosing crop pathology. A program was successfully developed that could find various pathologies and differentiate between them. Some pathologies caused yellowing or chlorosis while other caused the plants to die and soil to be exposed. These changes allowed for the infected areas to be detected out of the control. The area of the infected region could also be reliably calculated and a basic path could be generated for a drone or low flying aircraft to deliver stress-relieving agents

Limitations
- Maximum Resolution of the Webcams used
- Time for plans to grow
- Greenhouse Temperature control unit failing

Possible application
- This project could be used to develop a fully functioning remote sensing system in which a drone or small aircraft scans the whole field and detects the areas that are infected. The drone can then identify the likely cause of the stress and deliver stress-relieving agents to help the plants grow.

Future experiments
- Calculating the maximum and minimum HSV values based on the averages of the image
- Testing the programs ability to detect illness on actual crop fields from a drone
- Developing more sophisticated detection algorithms or training a neural network to detect the infected area

Literature Cited

Admin. "Remote Sensing Applications in Agriculture." Grind GIS-GIS and Remote Sensing Blogs, Articles, Tutorials, Grind GIS, 11 Feb. 2018,grindgis.com/remote-sensing/remote-sensing-applications-in-agriculture.
"Evaluation of Stress Detection Glasses." USFS T&D Program, United States Forest Service, www.fs.fed.us/t-d/programs/im/stress_detection/stress_report.shtml.
Fang, Yi, and Ramaraja Ramasamy. "Current and Prospective Methods for Plant Disease Detection." Biosensors, vol. 5, no. 3, 2015, pp. 537–561., doi:10.3390/bios5030537.
Nowatzki, John, et al. "Agricultural Remote Sensing Basics." EXtension, 1 Dec. 2011, articles.extension.org/pages/9693/agricultural-remote-sensing-basics.
"Tracking Change Over Time—Understanding Remote Sensing." USGS: Science for a Changing World.
"What Is Computer Vision?" The British Machine Vision Association and Society for Pattern Recognition, BVMA, www.bmva.org/visionoverview.
"What Is Phytopathology or Plant Pathology?" American Phytopathological Society, www.apsnet.org/about/Pages/WhatisPhytopathology.aspx.