Computer Vision Problems in Plant Phenotyping (CVPPP)

Advisor: A P Reeves   Author: Yupeng Lin; Yunke Tang

Cornell University Electrical and Computer Engineering Department

Introduction

The topic is "Computer Vision Problems in Plant Phenotyping (CVPPP)". We aim to achieve the goal of identifying leaves form the plants given. Plant phenotyping is the identification of effects on plant structure and function resulting from genotypic differences and the environmental conditions a plant has been exposed to. We apply pre-processing, MAT, local maximum/minimum process, Depth First search, connected components labeling and other algorithms. Comparing with the ground truth, we performed well and get a satisfied result.

(1) **Outline**: Automatically plant phenotyping, in this project, is to automatically split a bunch of leaves and give labels of them.

(2) **Implementation step**: First, we need to get a well processed image from the original image. The quality of the processed image decides how we implement the further analysis steps and also the difficulty. Second, some significant features need to be extracted from the image to be further analysis. Third, through those features we can figure out the exact center points, branch points and end points, then the split points could be detected through those points with the related algorithms. Finally, by using the split points, we can get the split lines and label the split leaves.

(3) **Analysis step**: With a bunch of ground truth to be compared, we use the manually labeled ground truth to test if our system is good at automatically phenotyping leaves.

Algorithms

The algorithm in this project contained three part:

1 extract the leaf and its skeleton map from the background;

2 searched the split lines to separate the leaf in overlapping area;

3 label the connected component and assign different color for each leaf.

(1) Color space conversion & RGB increment

For the leaf extraction, the input image was read as RGB format file. First of all, the image was transformed into the L*a*b color space. The L*a*b color is better representing green than RGB does. The aChannel, where its value indicated green, was extracted among three channels.

(2) Pre-filtering with Gaussian & Connected Components Deleting

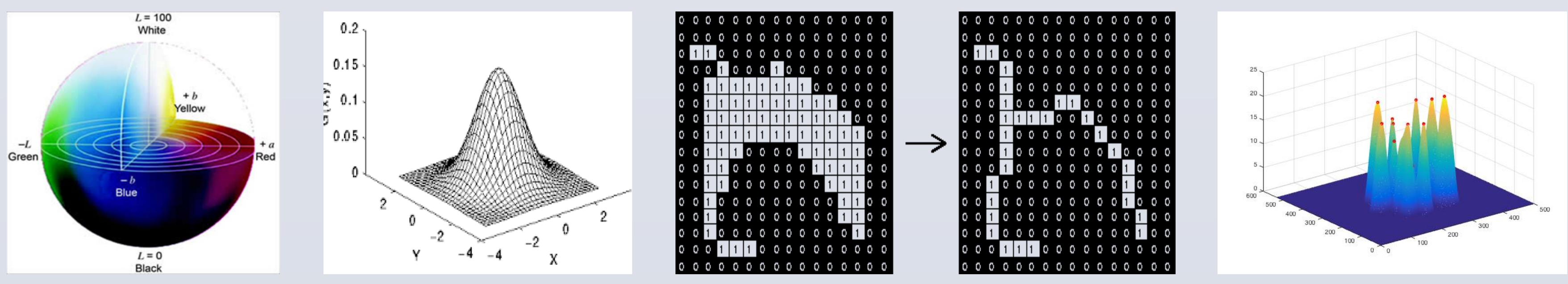
The threshold for background segment was used to segment the leaf from the background. The Gaussian filter was applied before the thresholding to remove the surrounding noise, which mainly came from moss around the plants. In additional, since the photographical issue, some image contained part of leaf from other plant, the connected component that has less than 80 pixels was removed from the image.

(3) MAT skeleton extraction

The image after above preprocessing only contained the leaf component, where the pixel value in background was zero and leaf component was one. The Euclidean distance transformed was performed at this image. The skeleton map based on medial axis transform was also found.

(4) Local maximum extraction

The center point was the important feature in the leaf segmentation, which is defined as the local maximum in the branch point. Usually, each leaf has its own center point. Before searching local maximum point, the mean filter was applied to distance-transformed-image to smooth .



(5) Center points correction

One thing worth noting was that local maximum point may not locate in the skeleton of image. Therefore, a helper function was help to adjust the position of center point (local maximum point) to the nearest skeleton map. The skeleton and center point were located in the image.

(6) Depth First searching

For the second part, search the split line and split point. The split line and split point was used to segment the overlapping part of leafs. Before searching split points. The skeleton map must be pruned so that every two center points were connected by a skeleton and skeleton which had open end (not end with center point) was discarded. The depth-first-search was used to recursively labeled the visited node and and skeleton.

(7) Split point searching

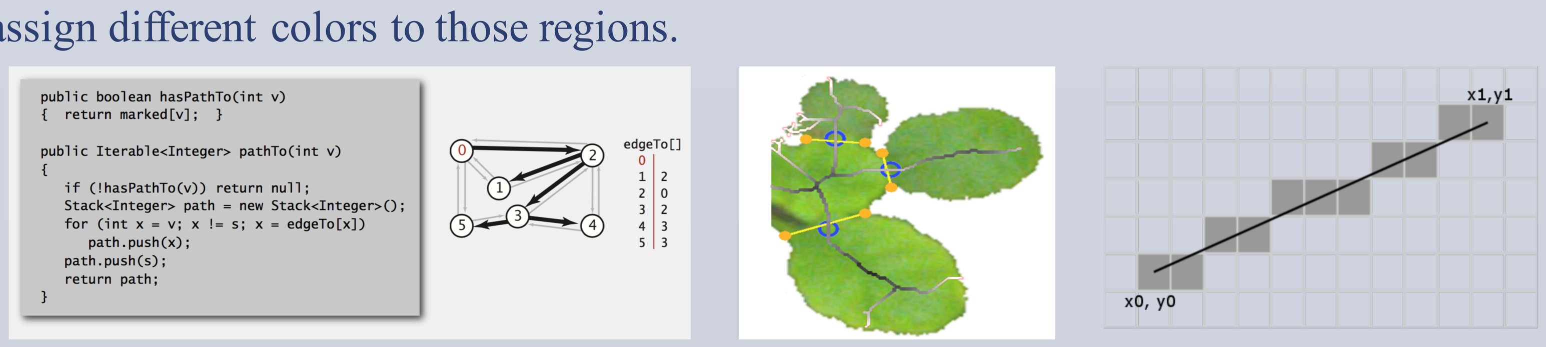
The split point is the edge point between two center points. Based on the calculated split points the exact split lines are needed to separate overlapping leaves. For each split point the nearest background point is searched. The second coordinate is searched at the opposite position relative to the split point. By examining these two splits points, the splint line was drawn.

(8) Straight line drawing

We were not going to simply plot a straight line on the resulting figure, but revise the image matrix entries to make a real line on the graph. One major thing is about "rasterization". So, we use interpolation method instead. It could make continuous line and successfully revise matrix.

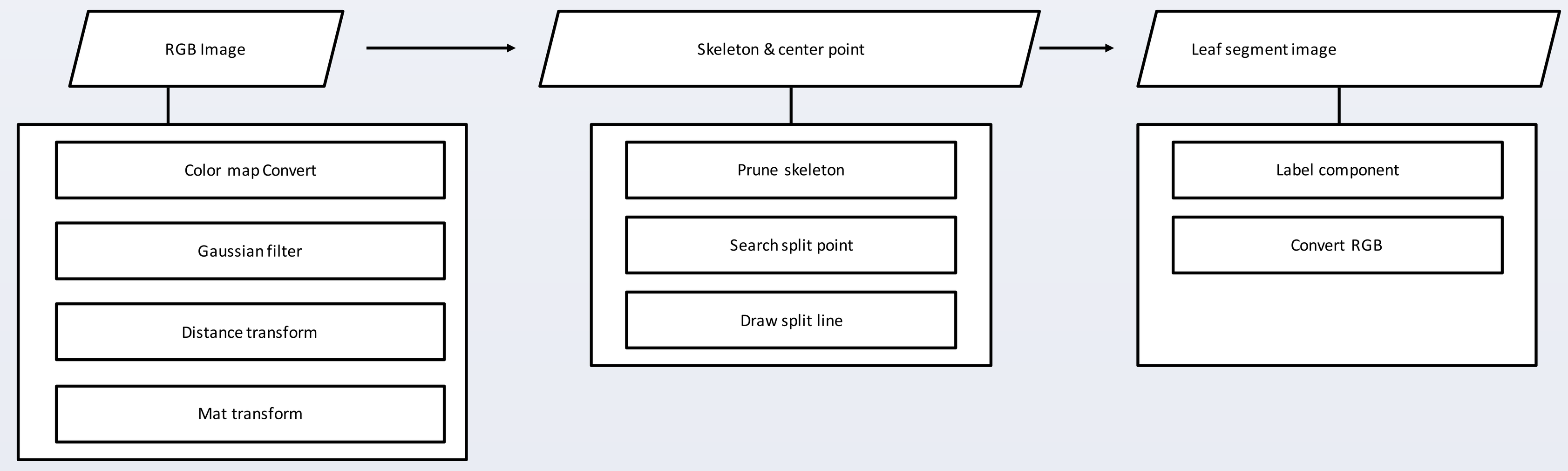
(9) Leaf segmentation & Labeling

For leaf segmentation, the method is used the 4 connection to find out the independent region and assign different colors to those regions.



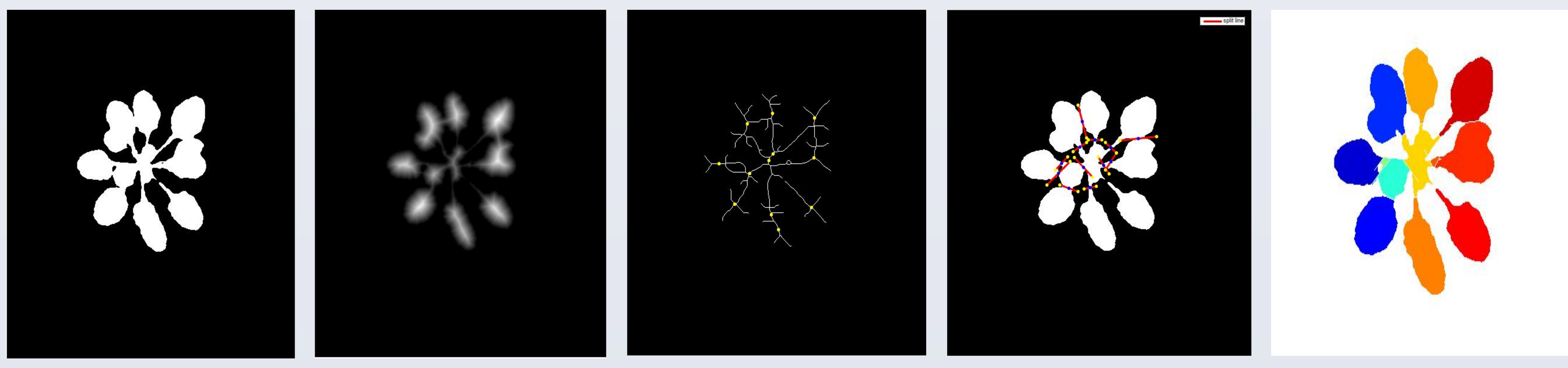
Experiment

The experiment implemented three-stage pipeline. (1) The first stage of experiment converts the rgb files to L*a*b color space. The gaussian filter is applied to image to reduce the noise. Automatic histogram in aChannel (green) component separates the leave from background. The skeleton structure and center point is determined by mat transform and distance transform. (2) The second stage experiment use the depth-first-search to prune the skeleton map so that each two center point was connect with one branch. The split points are also located by search the minimum distance point in branch. Split lines are drawn by rasterization once the splits point are given. (3) The third stage experiment labels the connected component and assigns the color to each component.



Results & Discussion

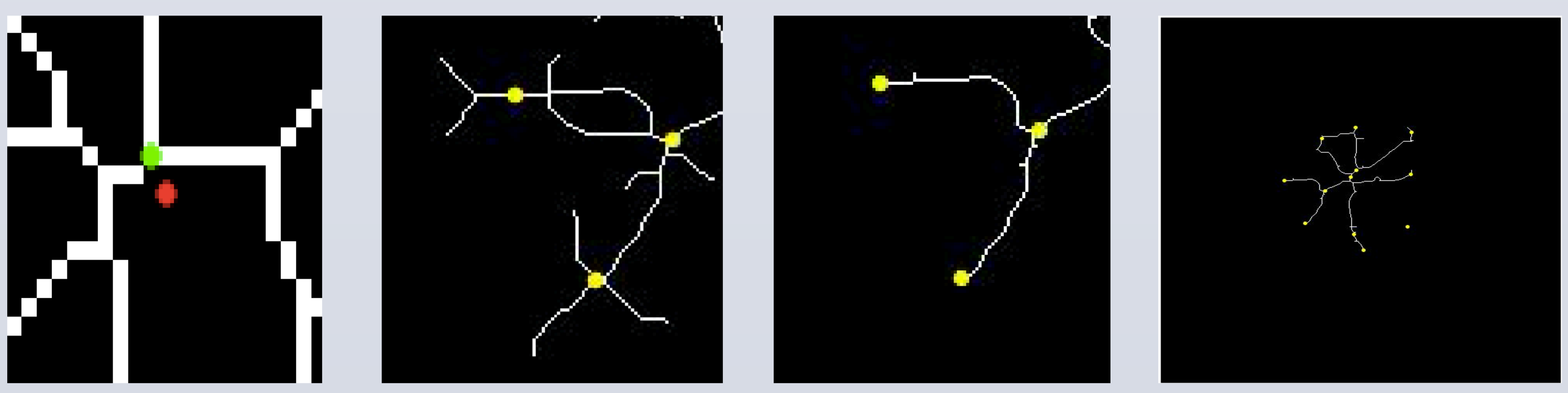
Result: The following image show the different stage image output. From left to right, image in L*a*b color space aChannel; leaf after the distance transform; the skeleton map of leaf and each leaf's center point. the split line and and split point in leaf; final segmentation result.



The dice coefficient for segmentation and foreground-background dice coefficient were calculated to evaluate the effect of segmentation. In the table, we randomly selected 6 image to evaluate the dice value and fgbg dice value.

Samples	Result Pixel	Origin Pixel	Overlap Pixel	Dice Value	FGBG Dice
plant005	25789	24438	24242	61.45%	96.53%
plant0012	14488	12503	12461	53.32%	92.33%
plant0015	30229	28708	28630	68.72%	97.15%
plant0024	41060	38759	38671	59.49%	96.90%
plant0038	44522	39878	39786	62.53%	94.28%
plant0055	47934	44463	44036	60.88%	95.32%

Discussion: (1) It is really important to engage an efficient automatic algorithm to adjust the pre-processing threshold and sigma. Thus, the processed image could be more robust and better prepared for the further analysis. (2) It is necessary to make the center points finding algorithm more robust, which means we have to find the center points on the skeleton. Due to the slight error in getting center points, we may see a little deviation of the center points from their skeletons. (3) How to choose the algorithm to deal with the split point finding issue could be another part of significance. Being aware of the endpoints and branch points, which may get in the way of finding split points, we used Depth First search to find the minimum, instead of local minimum, value after eliminating the influence of the elements mentioned.



Conclusion

The data set contains hundreds of real plant photographs. Each image was orderly read in and traversed, then placed into segmentation procedure. First, run the pre-processed program to make each image prepared. Then, we use the processed image to find its skeleton by using MAT. While the skeleton was found, we could implement the center points searching algorithm, which find the center point and make it right on the skeleton we've found. Next, using DFS, we make it possible to remove miscellaneous and trivial branches, leading to a graph that could be easier to analysis. On the reconstructed skeleton, we find split point and finally get the split line by algorithms showing above steps. Last but not least, we use these split lines to split each leaves and label them by different colors and evaluate our correctness.

Our project has extremely valuable meanings in not only research and analysis, but future applications. Because nowadays, we are stepping into information era, with more and more artificial intelligent application. Automatic plant labeling technic could bring huge benefit to the improvement of AI machines on not only agriculture, but like wild adventure, military applications and so on.