Crop Classification with Convolutional Neural Networks (CNNs)

Brody Vogel
Georgetown University

Abstract

Here, I use convolutional neural networks to classify poor-quality drone images of field segments into one of the following four classes: Soil, Grass, Soybean Plant, Broadleaf Weed. After pre-processing the images, I achieved an accuracy score of above 97%.

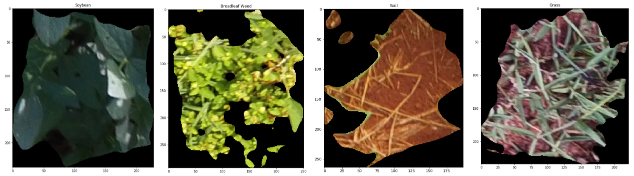
Introduction

The ability to accurately classify field imagery – i.e., identify hazardous weeds that harm yields – is immensely valuable. It can help cut down chemical cost and use (by highlighting patches of land that are more heavily populated by weeds) and aid in yield projections.

In this project, I perform the classification task described above, but on drone images of very poor quality. Strong results here, then, could cut further cut costs insofar as the technology needed to take poor-quality images is far less expensive than that necessary for high-resolution photos. My results *are* strong – using CNNs, I reached over 97% accuracy on a four-category classification task.

Data

The data set comes from Kaggle and contains a total of roughly 20,000 images from four classes; an example of each class is below:



The images are badly distorted by motion blurring, poor color contrasts, and bad cropping. Thus, I needed to perform pre-processing before feeding the images into the CNNs.

Methodology

My methodology falls into two categories: Image Pre-Processing and Model Fitting.

Image Pre-Processing

The pre-processing I performed on each image was a four-step process:

1.Extract the middle 150 x 150 pixels (to get rid of the weird black outlines)

2.Apply a 5 x 5 sharpening box filter (to correct for some of the blur)

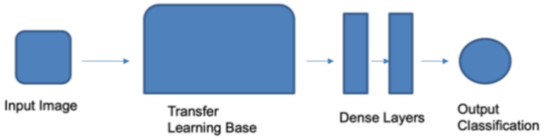
3.Use color thresholding to isolate only the shades of green (to accentuate the leaves)

4.Apply a 5 x 5 median blurring filter (to further stress the shapes of the leaves)

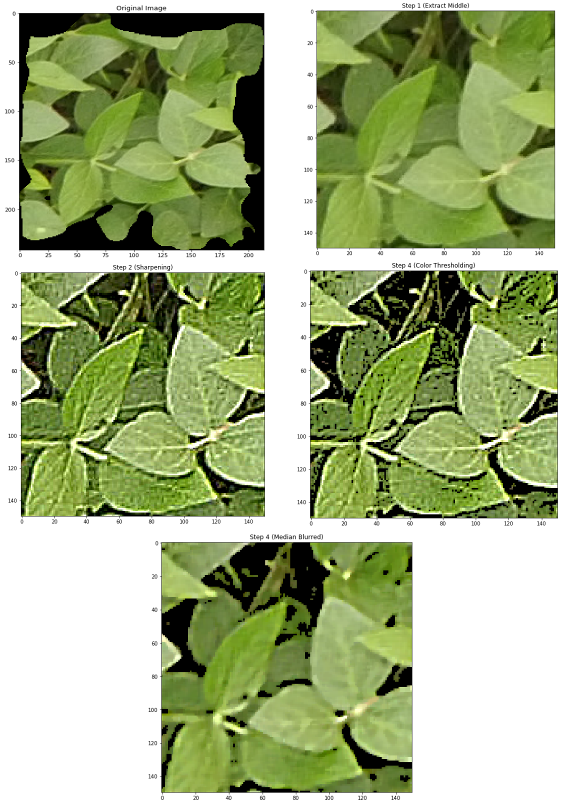
An example of this process is on the right of this blurb → The final image (bottom right) may look worse, but the important details – *the leaves* – stand out from the background, which has been removed.

Model Fitting

My final convolutional neural network is fairly simple. It takes a pre-processed image, runs it through the VGG16 imported base, then through two dense layers, and finally outputs a vector of four probabilities corresponding the the possible classes. The images is assigned the label for which the predicted probability is highest. A blueprint is below:



Example



Results

The results of my best models are listed below:

Images Used	Accuracy of VGG16 Model
Original Images	74%
Pre-Processed Images	> 97%
Black and White Pre-Processed Images	95%

For transparency, I also tested transfer bases from Google's Xception and Inception models, and VGG19. None of these performed as well on any of the tasks. Also, as the table mentioned, I also tried turning the images black and white, but this had little effect on the results

Conclusion

Working with a fairly simple model, I achieved an accuracy score on a four-category crop imagery classification task of over 97%. These are strong results, and with more images – from various heights, angles, etc. – it seems that number would only creep up. Suffice to say, I think there's ample ground for more work to be done here.

References

Background Removal (https://www.kaggle.com/ianchute/background-removal-cieluv-color-thresholding)
General Approach to Pre-Processing (https://medium.com/neuralspace/kaggle-1-winning-approach-for-image-classification-challenge-9c1188157a86)
Data (https://www.kaggle.com/fpeccia/weed-detection-in-soybean-crops/hom)
All project materials available here: https://github.com/BrodyVogel/Image-Processing/tree/master/Crop-Classification-Project