Fruit Counting for Automatic Inventory Management

1 Problem Statement

In the field of agriculture, yield estimation and mapping in orchards is vital for growers as it helps to utilize resources efficiently and improve returns per unit area and time. Having accurate knowledge of yield distribution and quantity, a grower can not only manage processes in the irrigation system such as chemigation, fertigation and thinning, but also plan ahead of time their harvest logistics, crop storage and sales. Producing yield information is currently done by manual sampling which is not only labour intensive but also expensive and time consuming. Such manual sampling also leads to inaccurate yield estimation. Hence, there is a need to develop machine vision systems to deal with the aforementioned problems in order to detect the fruits on each tree accurately, which in turn helps to reduce the errors in counting the total number of fruits in orchards.

2 Background Work

In traditional systems, fruit detection through key-point extraction and classification algorithms are often applied over vineyards and orchards. It exploits radial symmetries in the specular reflection of the individual berries to extract key-points, which are then classified as berries or not-berries. The detected regions are then used for yield estimation and prediction. Another approach uses simple color classifiers for key-point extraction for grape bunches and image patches are extracted around each key-point and a combination of color and texture filters are computed. The patches can then be classified as fruit or not-fruit using a trained classifier, such as a support vector machine (SVM) or a randomized KD-forest.

With the advancement of parallel computing using GPUs, deep neural network architectures, which host a significantly larger number of model parameters, are showing potential in capturing large variability in data. In a recent work, authors used multi-layered CNNs for image segmentation, in which individual patches representing contextual regions around pixels are densely classified in an image. More recently, convolutional neural network (CNN) has been shown to yield improved segmentation performance when a spatial prior on the classes is available. In another work, authors performed road image segmentation while incorporating the pixel positions to help the classifier in learning that road pixels are predominantly found near the bottom half of the images.

3 Materials and Methods

3.1 Datasets

3.1.1 Data Collection and Dataset Preparation: Images of trees can either be collected from any image database or captured through camera.

3.2 Methods

Figure 1 demonstrates the framework that can be applied for developing machine vision system in oder to detect fruits on each tree in an orchard.

3.2.1 Image preprocessing and Labeling: ZCA whitening can be used for preprocessing the data in order to remove pixel-level correlations on the image patches and force unit variance.

3.2.2 Training and Experimentation: In this step, training the convolution neural network for classifying an image as fruit or not-fruit. Subsequently, we can apply watershed transform method for yield estimation and test the performance afterwards.

3.2.3 Evaluation Measures: Measures such as accuracy and mean recall score, mean precision will be computed .We use the final mean F_1 score for the comparison of results across all of the different experimental configurations.

3.2.4 Deployment and analysis on real life scenario: The trained and tested fruit counting model will be deployed in a real-life scenario for further analysis where both positive and negative cases will be leveraged for further improvement in the methodology.

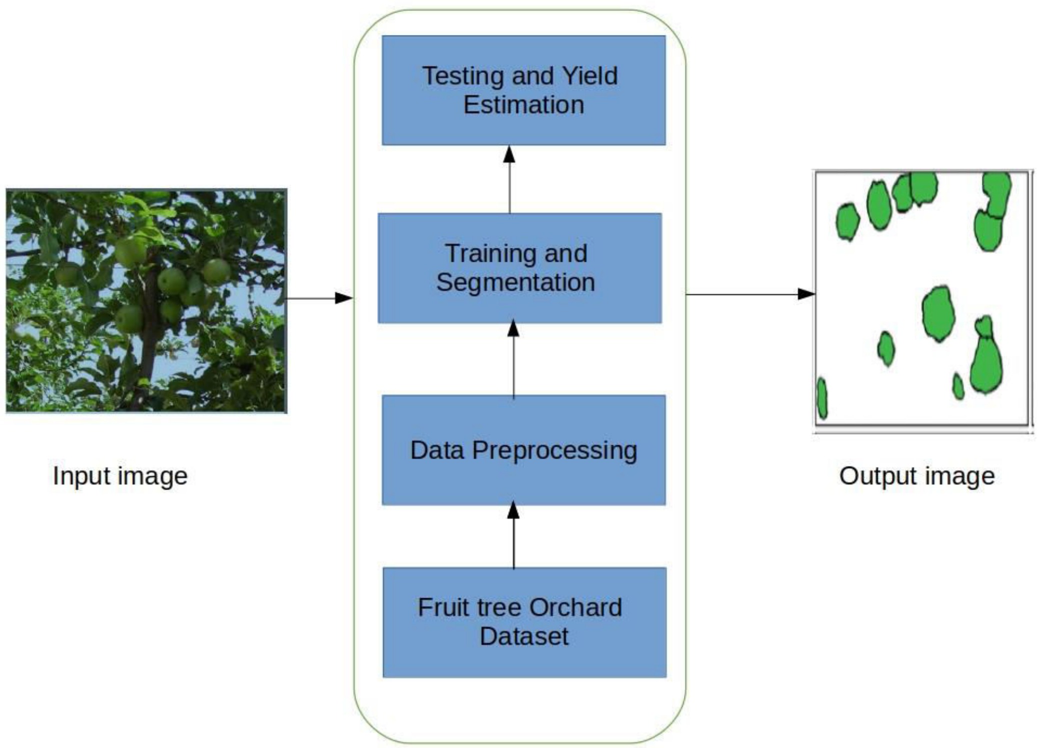

Figure 1: Architecture of fruit Counting for automatic inventory System

4 Experimental Design

4.1 Software and Hardware Requirements: Python based Computer Vision and Deep Learning libraries such as Pylearn2 will be exploited for the development and experimentation of the project. Training will be conducted on NVIDIA GPUs.