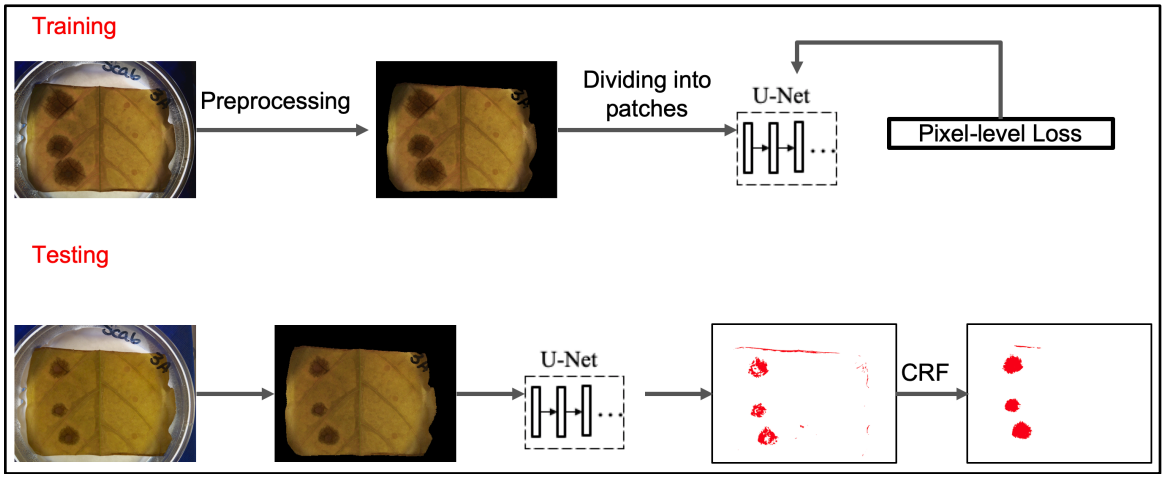
Leaf Segmentation



Structure

leaf-patch
|-- Metrics.py
|-- SaveWholeHeatmapAndMask.py
|-- leaf_divide_test.py
|-- leaf_divide_train.py
|-- merge_npz_final.py
|-- model
| |-- __init__.py
| |-- misc.py
| `-- u_net.py
|-- test.py
|-- train.py
`-- utils
  |-- __init__.py
  |-- datasets.py
  |-- joint_transforms.py
  |-- misc.py
  |-- mk_dataset.py
  |-- post-processing.py
  |-- preprocess.py
  `-- transforms.py

Prerequisites

Linux
NVIDIA GPU+CUDA 9.0

Dependencies
python 3.5.6, pytorch 1.1.0, opencv-python 4.1.1, scikit-image 0.14.0, scikit-learn 0.20.0, scipy 1.1.0, pydensecrf 1.0rc3, and etc.

Data Preparation
We use the leaf data from the directory *AutoPheno*. Before the mode training and testing, we first use the *utils/preprocess.py* to clean the data (mainly rename files) and split train/test dataset.

About Training
We first use *leaf_divide_train.py* to preprocess the leaves and divide them into several smaller image patches. Please change the path setting if necessary. (*root_pth* and *tumorname*)

Then we run *utils/mk_dataset.py* to make training dataset.

Finally, we run *train.py* to train the U-Net model for leaf segmentation. Please change the path setting if necessary. You may also change the following parameters:
BATCH_SIZE: the size of training image batch. Reduce or increase it according to the size of your GPU memory. Typically, larger is better.
MAX_EPOCH: the number of training epoch. One epoch means one traversal over the whole training set.
RESTORE_FROM: if you hope to train model from scratch, set "". If you hope to train model from previous saved model, set the path to the saved model here.
NUM_EXAMPLES_PER_EPOCH: the image number of training dataset. You can get it when run *mk_dataset.py*.
gpu: set the GPU id here.

About Testing
We first use *leaf_divide_test.py* to preprocess the leaves and divide them into several smaller image patches. Please change the path setting if necessary. (*root_pth* and *tumorname*)

Then we run *utils/mk_dataset.py* to make testing dataset.

We further run *test.py* to perform the inference. For brown disease, we use the trained model LEAF_UNET_B64_S4800.pth. For green disease, we use the trained model LEAF_UNET_B0064_S001800.pth. This can be set in the RESTORE_FROM. These pth file can be found in *trained_model* directory. You may need to change the following parameters: *test_dir*, *train_dir*, *NPZ_PATH*, *MAP_PATH*, *BATCH_SIZE*, *gpu*.

Subsequently, we run the *merge_npz_finally.py* to merge the segmentation results of each image patch for generating the segmentation map for the whole leaf. You may need to change: *dirlen*, *tumorname*, *srcpath*, and *model_id*.

Finally, please run *SaveWholeHeatmapAndMask.py* to perform CRF post-processing and save the visualization results. You can use the *metric.py* to evaluate the performance of model. Please change the path setting if necessary.

Contact

Please feel free to contact me (hfn5052@psu.edu) if you have any questions.

Reference

1. U-Net: https://github.com/zijundeng/pytorch-semantic-segmentation
2. CRF: https://github.com/lucasb-eyer/pydensecrf