

DAPD User manual



Diego Lozano-Claros, Xiangxiang Meng, Eddie Custovic, Guang Deng, Oliver Berkowitz, James Whelan, and Mathew G. Lewsey*

1. Overview

Digital Adjustment of Plant Development (DAPD) is a method that synchronizes shoot phenotypic measurements. It uses the plant leaf number to normalize time-series measurements, such as the projected rosette area and leaf area. This method improves accuracy by decreasing the statistical dispersion of time-series of quantitative traits (Figure 2). Also, it can identify more outliers than any other central tendency technique on the non-normalized dataset. The DAPD method was written in Python programming language and presented in Jupyter notebook pages. The DAPS program is available at https://github.com/diloc/DAPD_Normalization.git.

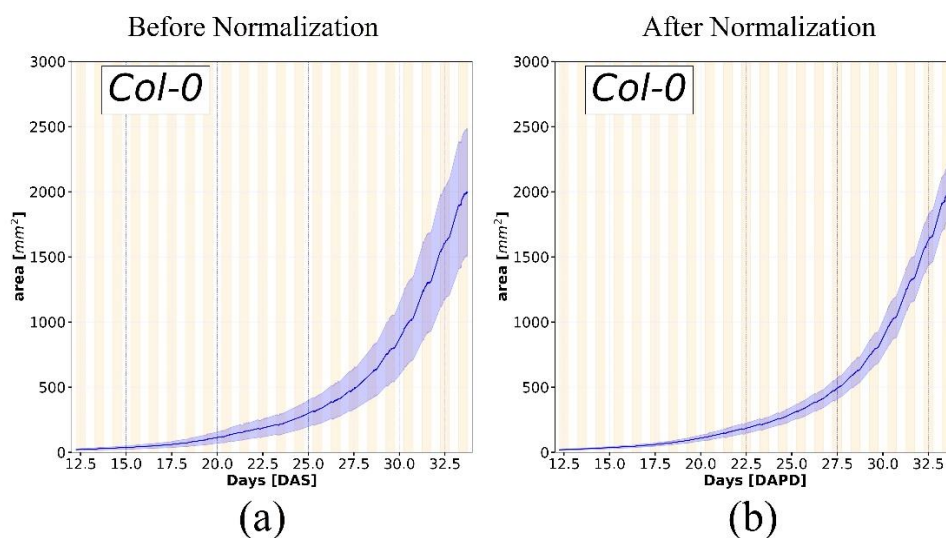


Figure 1 Mean and standard deviation of the non-normalized and normalized projected rosette area datasets. (a) the non-normalized datasets of Col-0 plants in experiment 1, (b) the normalized datasets of Col-0 plants in experiment 1. The light purple band indicates the standard deviation, and the solid blue curve shows the mean area.

2. Description

The DAPD method uses image processing algorithms to analyze and extract plant phenotyping traits. It starts by loading RGB images and other files such as camera parameters, which are used to correct the lens distortion. After, the image quality is improved by reducing the noise and correcting the color distortion. Then, the projected rosette is segmented from the pot image by removing the background automatically. Finally, phenotyping traits are obtained from the segmented image and write in CSV files. The traits include projected rosette area, leaf number, and perimeter (Figure 2). The user can run the DAPD image processing module to extract traits.

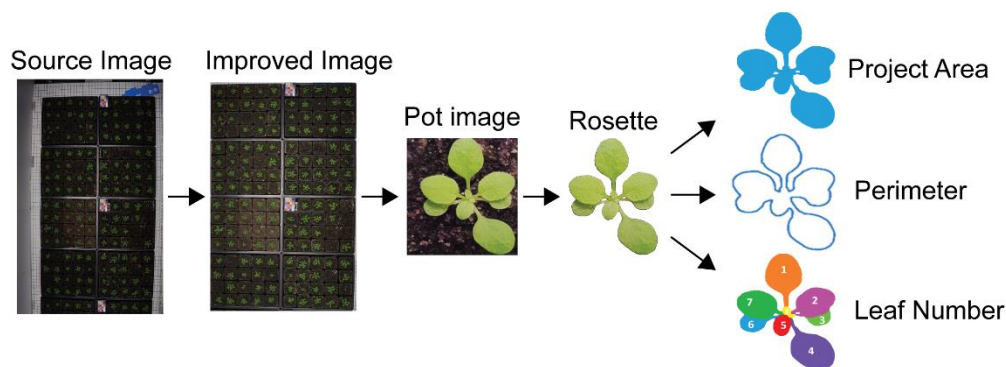


Figure 2 The most important image processing steps: Acquisition, correction and noise reduction, pot cropping, image segmentation, and phenotyping measurements.

The time-series of phenotyping traits are normalized to an early plant development stage. The number of leaves is used to identify a particular development stage among plants in an experiment (HTP scale). Then, the relationship between the development stages and time-series is studied by shifting the series timeline and calculating the regression.

3. Installation

The next step is to download and unzip the ZIP file in your computer from the repository: https://github.com/dilloc/DAPD_Normalization

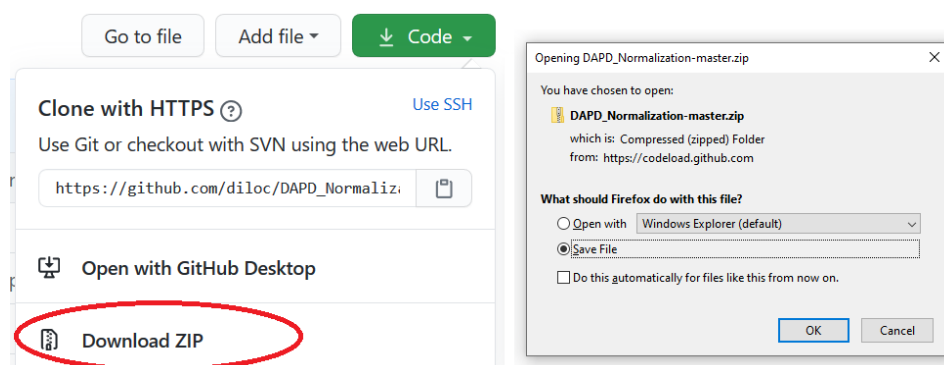


Figure 3 Download and unzip the repository ZIP file

4. Using the DAPD program

The DAPD program comes with two modules, image processing, and the data analysis module. To start with, the user has to unzip the ZIP file.

4.1. Image processing module

The image processing module is used to process RGB images. The user must open a Jupyter notebook interface in an Internet browser and find in the directory a Jupyter file called **"Image processing module.ipynb"**. If the file is opened on the browser, the user can run it. The program automatically identifies the images located in the Images folder and starts processing them.

The processing output is multiple CSV files that contain the phenotyping measurements of individual plants.

4.2. Data processing module 1

The data processing module 1 detects inconsistencies in the datasets, which can be corrupted during the data acquisition, storage, and image process. Then, the program can correct or remove the corrupted data. The user must open a Jupyter notebook interface in an Internet browser and find in the directory a Jupyter file called **"DAPD Normalization part1.ipynb"**. If the file is opened on the browser, the user can run it. The program automatically identifies the CSV files located in the Exp4_csvFiles folder.

The processing output is multiple CSV files that contain all the individuals that belong to a particular plant accession.

4.3. Data processing module 2

The data processing module 2 reads the cleaned CSV files and performs the developmental normalization of the phenotyping traits based on the number of leaves. The user must open a Jupyter notebook interface in an Internet browser and find in the directory a Jupyter file called **"DAPD Normalization part2.ipynb"**. If the file is opened on the browser, the user can run it. The program automatically identifies the CSV files located in the Exp4_csvFiles folder.

The processing output is multiple CSV files that contain the normalized datasets and curve plots, which represent the data before and after normalization.

5. Dependencies installation

The DAPD program requires the following dependencies to run on Python properly (3.7 or higher).

- Jupyter Notebook
- Pandas (1.0.3 or higher).
- OpenCV (4.2.0 or higher).
- Datetime

- Scipy (1.4.1 or higher).
- Matplotlib (1.18.1 or higher).

5.1. Quick Python installation

There are two popular ways to install the Python programming software and the required dependencies using: Python distribution and the Python standalone and PIP command.

5.1.1. Python distribution - Anaconda

One way is to download and install a Python distribution like Anaconda. The Anaconda distribution is an open-source, cross-platform language-agnostic package manager and environment management system. Anaconda (Python 3.7) can be downloaded from:

<https://docs.anaconda.com/anaconda/install/>

After installing Anaconda, the user must look for and open the Anaconda Prompt application on their computer. Then, the user can install the dependencies by typing conda command followed the name of the package:

- `conda install -c anaconda pandas`
- `conda install -c conda-forge opencv`
- `conda install -c anaconda scipy`

5.1.2. Python standalone and PIP command

The user must download and install the Python programming software from the official website:

<https://www.python.org/about/gettingstarted/>

Then, the user has to check if PIP is installed in the system if not, the user can install it from:

<https://pip.pypa.io/en/stable/>

After the Python and PIP installation is complete, the user can install the dependencies using PIP command on the command line or terminal:

- `pip install notebook`
- `pip install pandas`

- `pip install opencv-python`
- `pip install DateTime`
- `pip install scipy`
- `pip install matplotlib`