Phenotiki – How to setup the Phenotiki device

Authors: Massimo Minervini*
Valerio Giuffrida
Aliaksei Laurynovich
Software version: 1.2

January 4, 2018

Abstract

This document provides detailed instructions to setup and operate the Phenotiki device for affordable plant phenotyping. First, we provide a general overview of the RaspiCam image sensor and the web-based interface to operate the device using a browser. Next, we describe how to setup the device.

Contents

* E-mail: phenotiki@gmail.com

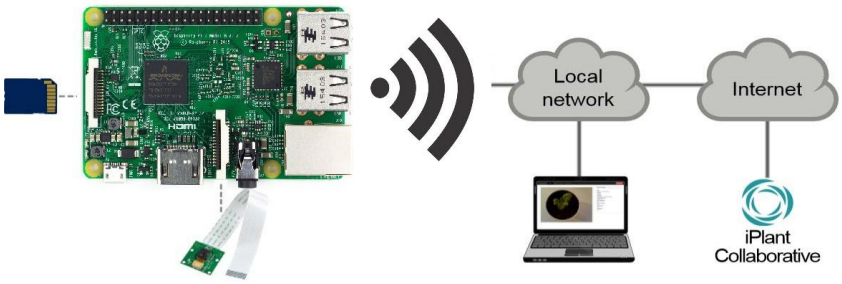


Figure 1: Schematic of our imaging solution based on the Raspberry Pi, showing the headless mode for image acquisition and transmission.

1 A smart sensing device based on the Raspberry Pi

The Phenotiki image acquisition device is based on the Raspberry Pi [1,2], a credit-card sized single-board computer designed and developed by the Raspberry Pi Foundation,[1] UK, as an educational tool for teaching computer science and programming [3, 4]. The Raspberry Pi is cheap (monetary cost is €40) and easy to use (it runs the Raspbian, a full-featured Linux operating system).

As illustrated in Figure 1, we use a Raspberry Pi, short-handed as *RPi*, equipped with an 8 megapixel 'RaspiCam' camera module, to capture static images of the scene. This solution offers great flexibility by having a complete yet small computer attached to the sensor. While the RaspiCam is capable of acquiring images of good quality that can satisfy a wide range of applications (the infrared camera module 'NoIR' is also available), in contexts where superior image quality is required, the Raspberry Pi can be used in combination with an SLR camera or other imaging sensors. The images acquired by the RPi setup can also be transmitted to the cloud for storage and analysis. Here we rely on the scientific cloud infrastructure offered by the iPlant Collaborative project [5], to deploy our plant image analysis software solution.

To ease configuration and monitoring, we deploy a web-based interface to operate the sensor remotely (cf. Figure 1). In the following, we describe how to set up the hardware and software components of our affordable sensing solution based on the Raspberry Pi.

1.1 RaspiCam

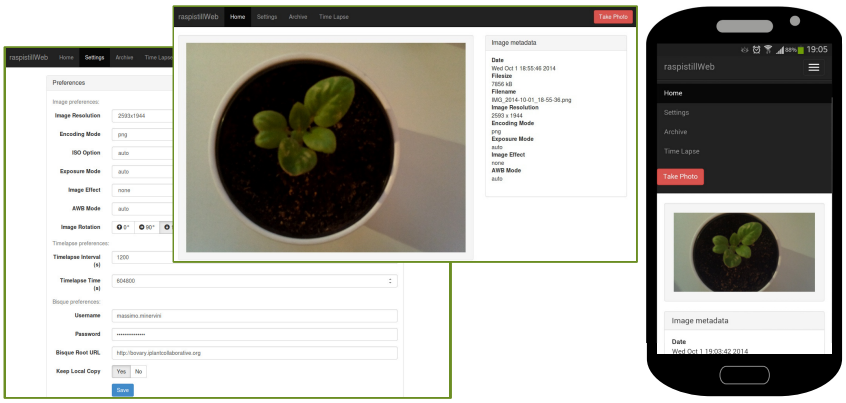
The "RaspiCam" camera module is a fixed-focus 8 megapixel CMOS image sensor produced by OmniVision Technologies.[2]

raspistill: Two command line utilities are available on the Raspberry Pi to operate the camera: `raspistill`, to capture still photos, and `raspivid`, to record HD video. For example, to acquire a picture and save it in the PNG format [6] we use the following command line options:

```
raspistill -n -e png -awb fluorescent -rot 180 -o filename
```

Notice that the camera module must be enabled in the Raspbian configuration (see Section 2.2).

[1] `http://www.raspberrypi.org`
[2] `http://www.ovt.com`



(a) Desktop interface (b) Mobile interface

Figure 2: Screen captures of our web-based interface to operate the Raspberry Pi camera, based on the raspistillWeb project.

The `raspistill` utility offers several options to configure image acquisition [7]. In particular, in our experimental setup we use neon lights for illumination, therefore we enable automatic white balance with the option *fluorescent*.

1.2 Easy device configuration via web interface

After initial setup, the Phenotiki device offers a web-based interface for easy configuration and operation from another computer (e. g., a laptop or even a smartphone, cf. Figure 2).

We implement our interface as a fork of the *raspistillWeb*[3] project (version 0.2), i. e. a web interface for the `raspistill` tool, implemented using the Python programming language and the Pyramid[4] web framework. We adapt the original software platform to the requirements of our application, adding the following key features:

(a) the user can select among different image file formats, including lossy (JPEG [8], GIF [9]) and lossless (BMP [10], PNG [6]) encoding standards;

(b) the user can start and interrupt time-lapse image acquisitions;

(c) more detailed information and metadata are displayed about acquired images;

(d) acquired images can be transmitted to the iPlant Collaborative [5] cloud infrastructure for storage and analysis; and

(e) the user can reboot or shut down the device.

As shown in Figure 2, graphical control elements of the user interface are intuitive and self-explanatory, thus rendering the web application easy-to-use. The 'Settings' page allows to configure parameters regarding image acquisition, time-lapse photography, and transmission to the iPlant. 'Home' and 'Time Lapse' pages allow to capture single still images and initiate a time-lapse acquisition, respectively. Detailed information about acquired images is displayed in 'Home' and 'Archive' pages. In the 'Archive' page the user can browse previously acquired images and

[3] `https://github.com/TimJuni/raspistillWeb`
[4] `http://www.pylonsproject.org`

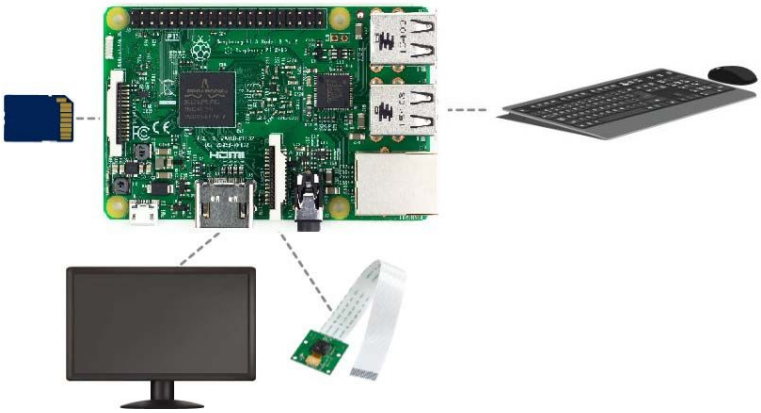


Figure 3: Schematic of our sensing solution at first configuration time, showing system setup and configuration. Note that after the initial setup, the system does not require input/output peripherals, and can be operated remotely (cf. Figure 1).

download time-lapse sequences as compressed ZIP archives. Our interface also adapts to small screens and can be displayed on mobile devices such as smartphones and tablets (cf. Figure 2).

By relying on the BisQue Python API, our software can transmit acquired images to the iPlant. This requires that valid credentials (username and password) for iPlant be specified in the Settings page. Optionally, the local copy of the image files can be deleted after transmission, to save storage space on the device.

2 Setting up the Phenotiki device and software

2.1 Hardware requirements

To replicate the Phenotiki device you will need the following hardware equipment:

- Raspberry Pi version 1 model B (or higher);
- "RaspiCam" camera module;
- USB micro power supply (recent versions of the Raspberry Pi necessitate a specialized power supply);
- 8GB (or larger) Secure Digital (SD) memory card or SD card with preinstalled Raspbian OS
- HDMI monitor and cable;
- USB keyboard and mouse;
- USB wireless dongle (only needed for RPi version 2B and older) ;
- self-powered USB hub to get a keyboard and a mouse connected (only needed for RPi version 1B and older) .

This equipment can be easily purchased from one of the several distributors selling worldwide.

Setting up the Raspberry Pi requires only few simple steps, described in detail in the official documentation [7].[5] In order to install the necessary software and perform initial configurations, the device must be attached

[5] `http://www.raspberrypi.org/documentation/setup/`

to input/output peripherals (monitor, keyboard, and mouse), as shown in Figure 3. To connect the camera module to the Raspberry Pi, proceed as shown in the instructional video available on YouTube.[6] Subsequently, the Raspberry Pi can be started headless and operated remotely from a computer connected to the same local network (cf. Figure 1).

2.2 Operating system and software setup

Operating system: This step depends on whether you purchased an SD card with preinstalled Raspbian or have an empty one.

In the first case (SD card with preinstalled Raspbian), just insert the SD card in your Raspberry Pi and run the device. If you have completed this step, you can immidiately proceed to the "Enable interfaces" section.

In the second case (empty SD card), you need to do the following steps to get the operating system installed on your Raspberry Pi:
Download and install the latest version of the Raspbian operating system, available at `http://downloads.raspberrypi.org/raspbian_latest`. Extract the system image from the ZIP archive and install it on the SD card (also used for local storage on the Phenotiki device). The steps to write a Raspbian image to an SD card depend on the user's system, and instructions for several operating systems can be found in the official documentation [7].[7] For example, on Linux, insert the SD card and run `df-h` to see the device name corresponding to the SD card, e.g., `/dev/mmcblk0p1`. Then, run the following commands with superuser privileges to first unmount the SD card and then install the Raspbian image `2016-03-18-raspbian-jessie.img`:

```
umount /dev/mmcblk0p1
dd bs=4M if=2016-03-18-raspbian-jessie.img of=/dev/mmcblk0
sync
```

The `dd` command may take several minutes to write the image to the SD card. `sync` will ensure that all buffered data are written to the SD card. Upon completion, unmount and remove the SD card from the card reader.

Enable interfaces: With the peripherals connected as in Figure 3, start the Raspberry Pi, that will automatically boot to the desktop environment. From Menu, Preferences, launch Raspberry Pi Configuration to perform some system-level configuration tasks. In the System tab, click on Expand Filesystem. In the Interfaces tab, enable Camera and SSH. In the Localisation tab set Locale and Timezone. After completing configuration reboot the device.

Wi-Fi setup: In the desktop environment, click on the network icon in the top menu bar. Select your local Internet connection and enter appropriate credentials if necessary. The device will remember network settings and connect on start-up.

Note that recent versions of the Raspberry Pi feature on board Wi-Fi, however, older versions require a USB wireless dongle.

Upon startup the Raspbian launches an SSH (secure shell) server, allowing remote access to the command line of the Raspberry Pi. For example, on a Linux system use the following command to connect to the device:

```
ssh -p 22 pi@<IP>
```

[6] `http://www.youtube.com/watch?v=GImeVqHQzsE`
[7] `http://www.raspberrypi.org/documentation/installation/`

where `<IP>` is the local (or remote) IP address of the Raspberry Pi (e. g., `192.168.1.5`). To find the IP address assigned to the Raspberry Pi in the local network, execute in a terminal on the device the command `hostname -I`. Default login user name and password are, respectively, 'pi' and 'raspberry'.

Packages upgrade: Once a Wi-Fi connection is established, open a terminal on the Raspberry Pi (either directly on the device or remotely via SSH) and upgrade the device firmware and all installed packages to the latest available version:

```
sudo rpi-update
sudo apt-get update
sudo apt-get -y dist-upgrade
sudo apt-get -y autoremove
sudo apt-get clean
```

This operation may take some time, depending on network speed.

Software dependencies: Install the software dependencies necessary to setup and run the Phenotiki software with BisQue support:

```
sudo apt-get install -y mercurial python2.7-dev \
       python-virtualenv python-setuptools python-picamera \
       libjpeg8-dev zlib1g-dev libxml2-dev libxslt1-dev
```

Install raspistillWeb: Download the Phenotiki fork of the raspistillWeb software by cloning our GitHub repository and install it on the device:

```
mkdir ~/phenotiki
cd ~/phenotiki
virtualenv --python=python2.7 env
cd env
git clone https://github.com/phenotiki/raspistillWeb.git
cd raspistillWeb
../bin/python setup.py develop
../bin/initialize_raspistillweb_db development.ini
```

Note that the software and all dependencies will be installed in the virtual environment without affecting global Python options and libraries. This step guarantees that the software will work even if the Raspbian operating system and its packages are updated.

Install BisQue support: Build and install the Python APIs for Bisque, running the following commands:

```
cd ~/phenotiki/env/
bin/pip install requests==2.7.0
bin/pip install BeautifulSoup4
CFLAGS="-O0" bin/pip install lxml
hg clone http://biodev.ece.ucsb.edu/hg/bisque/
cd bisque/bqapi/
../../bin/python setup.py build_py
../../bin/python setup.py install
```

The `hg` command will download the BisQue code, thus this step may take some time depending on network speed.

Subsequently, when a new version of the BisQue API is available, to update the BisQue API from the official repository to the latest release, execute:

```
cd bisque/bqapi/
hg pull -u
```

You may check also the official BisQue installation instructions[8] for a more extensive description of the installation process.

raspistillWeb: Start raspistillWeb by launching the service that will run on the device and will generate the web interface upon request:

```
cd ~/phenotiki/env/raspistillWeb/
../bin/pserve development.ini
```

Afterwards, it is possible to access the raspistillWeb interface from a web browser on a computer connected to the same local network. For example, if the IP address of the Raspberry Pi is `192.168.1.5`, in a browser on a laptop or a smartphone open `http://192.168.1.5:6543`. (By default the raspistillWeb service will be listening on port number 6543.)

By properly configuring the router of the local network, it is be possible to enable remote access to the raspistillWeb via the Internet. Notice however that this configuration step depends on the local network settings and you should seek support of your network administrator.

In order to start the raspistillWeb automatically on system boot, on the Raspberry Pi edit the `/etc/rc.local` file using a text editor with superuser privileges, for example:

```
sudo nano /etc/rc.local
```

Add the following lines at the bottom of the file before the line `exit 0`:

```
cd ~/phenotiki/env/raspistillWeb/
../bin/pserve development.ini
```

To save the changes and quit the editor, press `Ctrl+X`, then `Y`, and finally `Enter`.

Finally, if you are deploying multiple Phenotiki devices, you can assign different names by editing the `/etc/hostname` file:

```
sudo nano /etc/hostname
```

Change the default `raspberrypi` to a name of your choice, e.g., `phenotiki-device-01`. The name of a device is also displayed in Settings page of the web interface.

3 Operating the device

Coming soon!

References

[1] C. Severance, "Eben Upton: Raspberry Pi," *Computer*, vol. 46, no. 10, pp. 14–16, 2013.

[2] E. Upton and G. Halfacree, *Raspberry Pi User Guide*. Wiley, 2014.

[3] C. Andrews, "Easy as Pi," *Engineering Technology*, vol. 8, no. 3, pp. 34–37, 2013.

[8] `https://biodev.ece.ucsb.edu/projects/bisque/wiki/InstallationInstructions05`

[4] F. Cuomo, E. Mibuari, K. Weldemariam, and O. Stewart, "Leveraging Raspberry Pi for interactive education," in *Annual Symposium on Computing for Development*, ser. ACM DEV-4 '13, 2013.

[5] S. A. Goff, M. Vaughn, S. McKay, E. Lyons, A. E. Stapleton, D. Gessler, N. Matasci, L. Wang, M. Hanlon, A. Lenards, A. Muir, N. Merchant, S. Lowry, S. Mock, M. Helmke, A. Kubach, M. Narro, N. Hopkins, D. Micklos, U. Hilgert, M. Gonzales, C. Jordan, E. Skidmore, R. Dooley, J. Cazes, R. McLay, Z. Lu, S. Pasternak, L. Koesterke, W. H. Piel, R. Grene, C. Noutsos, K. Gendler, X. Feng, C. Tang, M. Lent, S.-J. Kim, K. Kvilekval, B. S. Manjunath, V. Tannen, A. Stamatakis, M. Sanderson, S. M. Welch, K. A. Cranston, P. Soltis, D. Soltis, B. O'Meara, C. Ane, T. Brutnell, D. J. Kleibenstein, J. W. White, J. Leebens-Mack, M. J. Donoghue, E. P. Spalding, T. J. Vision, C. R. Myers, D. Lowenthal, B. J. Enquist, B. Boyle, A. Akoglu, G. Andrews, S. Ram, D. Ware, L. Stein, and D. Stanzione, "The iPlant collaborative: Cyberinfrastructure for plant biology," *Frontiers in Plant Science*, vol. 2, no. 34, 2011.

[6] *Portable Network Graphics (PNG) Specification*, W3C Recommendation ISO/IEC 15 948:2003 (E), Rev. 2, 2003. [Online]. Available: http://www.w3.org/TR/PNG/

[7] Raspberry Pi documentation. Raspberry Pi Foundation. [Online]. Available: https://www.raspberrypi.org/documentation/

[8] *Information technology – Digital compression and coding of continuous-tone still images*, International Telecommunication Union (ITU) ITU-T Recommendation T.81, 1992.

[9] CompuServe Inc., "Graphics interchange format (GIF)," Jul. 1990, version 89a. [Online]. Available: http://www.w3.org/Graphics/GIF/spec-gif89a.txt

[10] J. D. Murray and W. VanRyper, *Encyclopedia of graphics file formats*. O'Reilly, 1994.