

## Software notes

### OS: Raspian 32-bit (Lite, Debian 11 bullseye version)

For IDS software (required to use the Ueye camera implemented in the system):  
from the official IDS website download the "IDS Software Suite" software for Linux ARM operating systems with 64-bit architecture, version v8 and "hard" floating point.

→ URL: [https://en.ids-imaging.com/download-details/AB.0010.1.55500.23.html?os=linux\\_arm&version=v8&bus=64&floatcalc=hard](https://en.ids-imaging.com/download-details/AB.0010.1.55500.23.html?os=linux_arm&version=v8&bus=64&floatcalc=hard)

It is recommended to download the archive version ('IDS Software Suite 4.96.1 for ARMv8 64-bit (hf) - archive file') and follow the instructions in the README file for proper installation.

In summary:

1) Install all the required IDS packages:

```
sudo apt-get install debconf cmake libc6 libomp5 libatomic1 libstdc++6  
libqt5opengl5 libqt5concurrent5 libqt5gui5 libqt5widgets5 build-  
essential libcap2 libusb-1.0-0 libqt5network5 libqt5xml5
```

2) Copy the .tgz / .tar files in the Home folder and extract them:

```
tar xvf ueye_<version>_<arch>.tar
```

3) Run the .run file and confirm any intermediate requests:

```
sudo sh ./ueye-<version>_<arch>.run
```

- Python packages:

1) Install Python3.9 and Pip (or verify their current installation)

2) Install the required packages by typing on the command line:

```
sudo pip3 install termcolor pyueye pillow opencv-python numpy
```

3) Connect the USB stick containing the .py files for interfacing with the Ueye camera.

If the USB is recognised immediately by the system, proceed with the following step (the USB is visible in the folder /media).

Otherwise, it is necessary to mount the USB stick and, after the files have been copied, unmount it before removing it from the Raspberry Pi3:

```
sudo blkid (to find the name under which the USB stick is read, i.e. /dev/sda)
```

```
sudo mkdir /media/usb
```

```
sudo mount /dev/sda /media/usb
```

... (various file copying operations as in point 4)

```
sudo umount /media/usb
```

4) Copy the .py files from the USB stick or external hard disk into the PyCamera\_control folder (to be created within the directory /home)

```
sudo mkdir /home/PyCamera_control
```

```
sudo cp <nome del file da copiare> /home/PyCamera_control
```

5) The .py files are set to save all results within a specially created folder on a foreign disk named 'Vektor'; if you do not want to connect an external disk, or if you want to do it under another name, edit the first lines of the PyCamera.py file.

```
sudo nano PyCamera.py (when finished editing, press Ctrl+O to save and then Ctrl+X to exit)
```

- Bash files:  
Installare il pacchetto screen: `sudo apt-get screen`
- RaspController configuration through RaspAp application:
  - 1) Download the RaspController application to your smartphone or tablet
  - 2) Installing RaspAp on Raspberry Pi3: <https://raspap.com/>

```
sudo apt-get update
sudo apt-get full-upgrade
sudo reboot
```

```
sudo raspi-config (from which to set WiFi location options)
```

```
curl -sL https://install.raspap.com | bash
```

```
sudo reboot
```

During installation, confirm all the choices proposed by the Quick installer and ensure that the Internet connection is never lost. After the last system reboot, the access point (AP) will present the default RaspAp settings:

**IP address:** 10.3.141.1

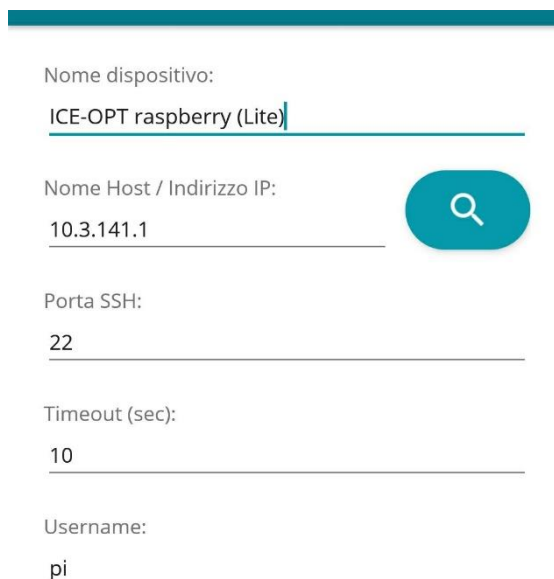
**Username:** admin

**Password:** secret

**DHCP range:** 10.3.141.50 — 10.3.141.255

**SSID:** raspi-webgui

**Password:** ChangeMe



The screenshot shows the RaspAp configuration interface with the following fields and values:

- Nome dispositivo: ICE-OPT raspberry (Lite)
- Nome Host / Indirizzo IP: 10.3.141.1
- Porta SSH: 22
- Timeout (sec): 10
- Username: pi

3) Add the device to the list in the RaspController application; the image above shows the settings currently entered. For authentication, enter the RaspberryPi password.

In order to connect to the Raspberry Pi3 via RaspController, the device on which the application is installed must be connected to the local network generated by the Raspberry Pi3, named 'raspi-webgui' by default.

4) For headless systems like this the simplest thing is to put a file in the 'boot' partition that is also seen by Windows.

In the case of the network, a text file named 'wpa\_supplicant.conf' should be created:

```
sudo nano wpa_supplicant.conf
```

with the following content:

```
ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1
country=IT
network={
    ssid="raspi-webgui"
    psk="ChangeMe"
    key_mgmt=WPA-PSK
}
```

**NOTE:** if the WiFi location settings selected via `raspi-config` are incorrect, the wireless connection via RaspAp may not work.

- If necessary, the first time the Ueye camera capture programme is launched, set (from the command line) the camera ID number and the corresponding IP address:

```
ueyesetid
  o ID: 1
```

```
ueyesetip
  o IP: 192.168.0.12 (example)
  o Netmask: 255.255.0.0 (example)
```