Raspberry Pi for Computer Vision
Case: Vein Visualization

PiLab Tutorial

Shu Isaka[*1], Takumi Kitajima[*1], Atsushi Inoue[*2]
[*1]Mie University, Tsu, Mie, Japan
[*2]Eastern Washington University, Spokane, WA, USA

Shu Isaka

- Graduate Student @ Mie University
- **Major:**
  Electrical & Electronic Engineering
- **Lab:**
  Information Processing Lab
- **Field of Study:**
  Biomedical Informatics.

Takumi Kitajima

- Graduate Student @ Mie University.
- **Major:**

  Public Health, Epidemiology
  (Prev.) Electrical & Electronic Engineering.

- **Lab:**

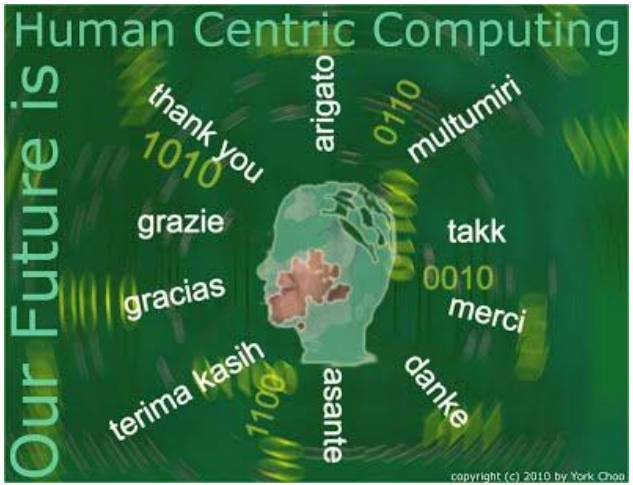
  Dept. of Public Health and Occupational Medicine

- **Field of Study:**

  Public Health, Epidemiology

Atsushi Inoue

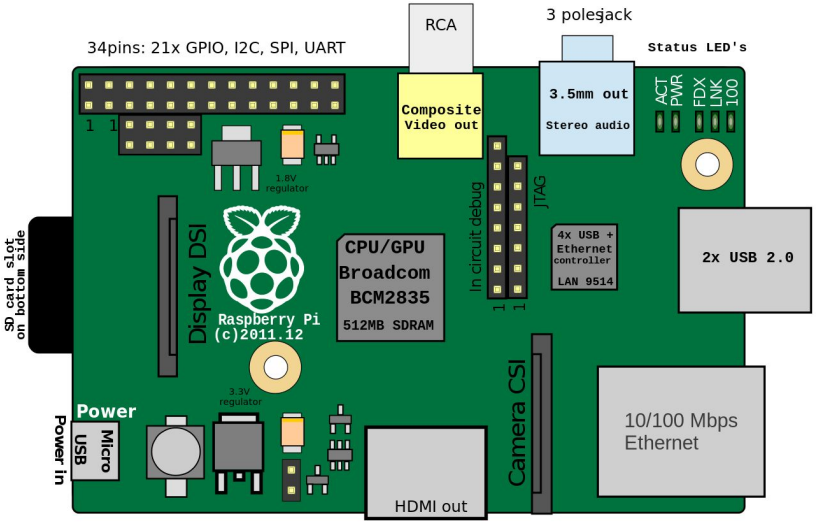
- Human Centric Computing→Perceptual Information Processing
  - **Perceptual Logic Programming (AI platform) & Machine Learning**
  - **Fuzzy Logic**
- Intelligent Informatics Initiative (I3)
  - Digital Entrepreneurship & E-Business
  - Information Security Management
  - Medical Informatics
- PiLab@EWU
  - Raspberry Pi, LINUX



Web: http://www.inoueatsushi.net/

Raspberry Pi

- Open source
- Small and lean
- Low cost
- Versatile
- Similar CPU as Android and iPhone
- Easy to use



RASPBERRY PI 3 MODEL B+

The latest revision of our third-generation single-board computer

BUY NOW >

or buy for business

RASPBERRY PI 3 MODEL B

Our third-generation single-board computer

BUY NOW >

or buy for business

RASPBERRY PI 2 MODEL B

The Raspberry Pi 2 Model B is the second-generation Raspberry Pi

BUY NOW >

or buy for business

RASPBERRY PI 1 MODEL B+

The Model B+ is the final revision of the original Raspberry Pi

BUY NOW >

or buy for business

RASPBERRY PI 1 MODEL A+

The Model A+ is the low-cost variant of the Raspberry Pi

BUY NOW >

or buy for business

RASPBERRY PI ZERO W

Single-board computer with wireless and Bluetooth connectivity

BUY NOW >

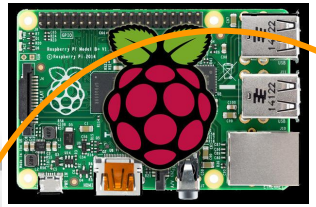
RASPBERRY PI ZERO

Our lowest-cost single-board computer

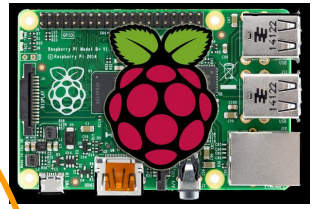
BUY NOW >

https://www.raspberrypi.org/products/

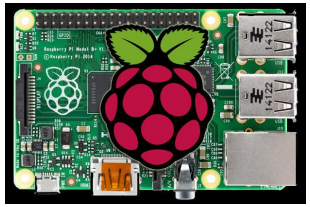
Raspberry Pi (LINUX) Networking & IoT @ iπLab



LINUX Workstation



Router, Hotspot



Database, WordPress, etc.

Clients (web, apps, etc.)

Sensor stations

Networking

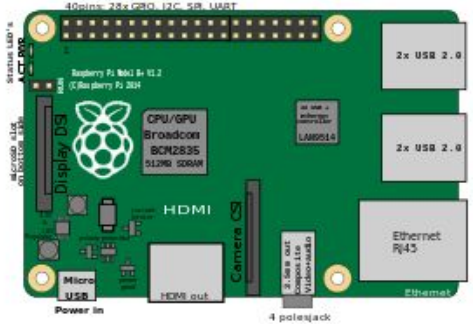
Servers (database, etc.)

IoT stations

Sidenote: PiLab Setting for DIY -- a variation of headless Pi

ICS connection via Ethernet

USB Ether adapter works when no Ether ports.



BYOL or cart laptop

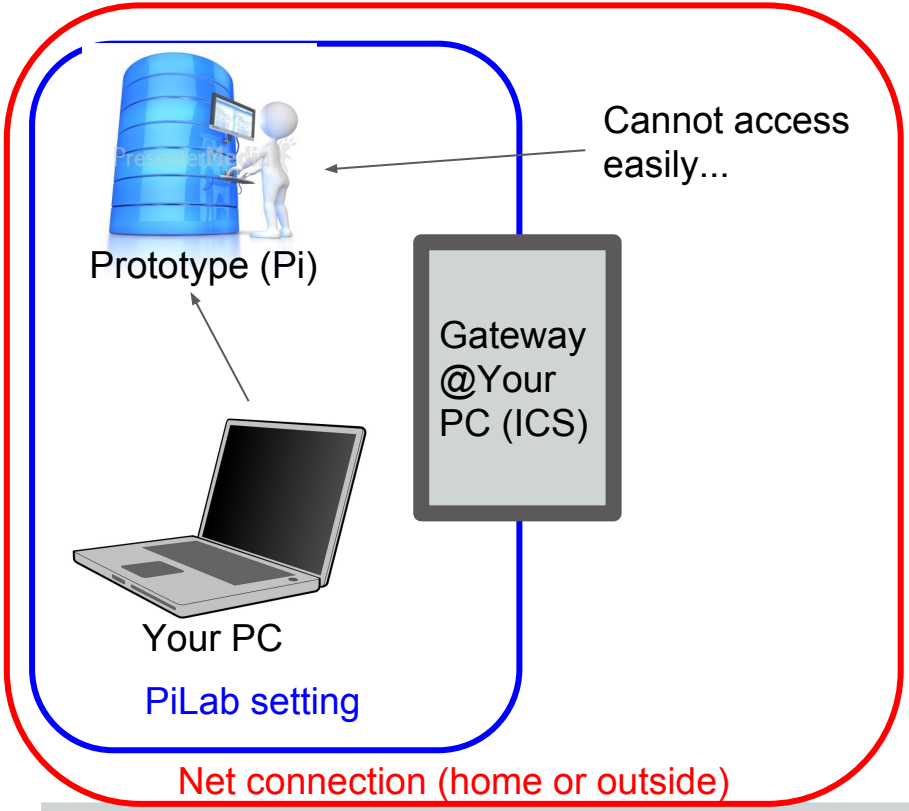
Ras Pi System Disk
On micro SD card

Power from the laptop USB

Convenient to have
USB card reader &
USB memory stick

CLI: ssh on laptop + sshd on Pi
GUI: X server or RDP client on laptop + sshd and/or xrdp on Pi

Sidenote: Prototyping individually with high mobility



[1] Download Raspbian OS image

Please start downloading the OS image for Raspberry Pi. It takes 1-30min (faster or slower depending on speed).

Raspbian Stretch With Desktop (with GUI)

ZIP -- 1.7GB (to be downloaded)

Unzip to extract .img file -- a few GB (to be extracted from the downloaded ZIP)

Note for PIXEL: unzipping a file >4GB has issues with an old unzipper. File Explore unfortunately cannot handle this. Use 7ZIP for Windows 7.

SD card imaging (vs file copying)

Imaging -- duplication of the entire drive/disk.

Sudo privilege is required.

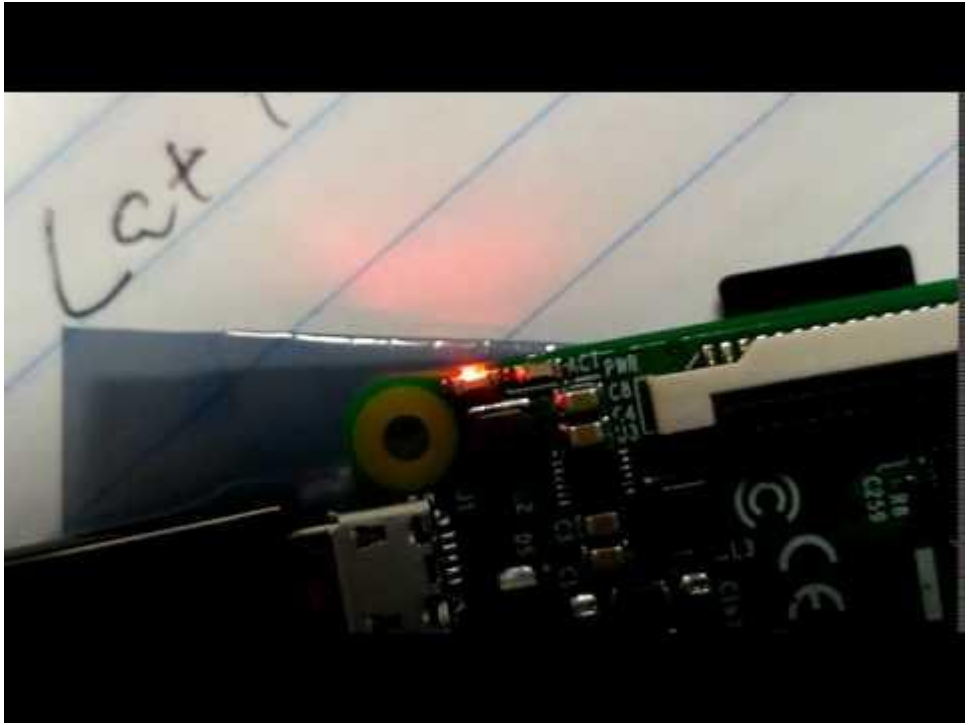
Password needs to be entered in many cases.

[2] SD card imaging: Etcher

Simply double click to execute Etcher (unzip is necessary for LINUX). NO INSTALLATION PROCESS.

[5] Booting Raspberry Pi

- Remember: NO POWER switch on Pi.
  Plugging in and out the micro USB power cable.
- Standard procedure
  1. Insert the micro SD card
  2. Plug in the Ether cable to the laptop and the Pi.
  3. Plug in the micro USB power cable (to power on)
  4. Observe the red LED (power indicator)
     This should be steady.
  5. Observe the green LED (disk access indicator).
     This should be actively blinking.


PWR

Raspbian OS Housekeeping

- Housekeeping after installation
  - **`'raspi-config'`** (clock, keyboard, overclocking, fs exp, etc.)
  - Update the OS
    - **`sudo apt-get update`**
    - **`sudo apt-get upgrade -y`**
    - **`sudo apt-get autoremove -y`**
    - **`sudo apt-get clean`**
  - Full upgrade (inplace of upgrade) -- takes longer
    - **`sudo apt-get dist-upgrade -y`**

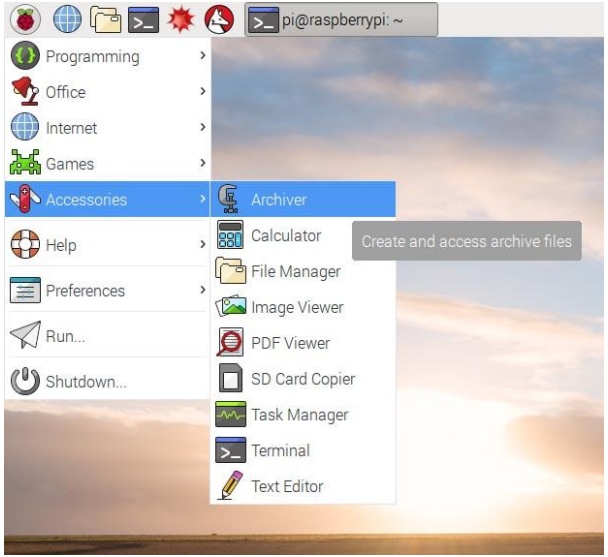
Update after the first installation may take very long -- over 1 hour easily.

Shutdown

- In Bash, SSH

  ```
  sudo shutdown -h now
  ```

- On desktop, In RDP
  Go to the menu->Shutdown (Shutdown)



Reboot

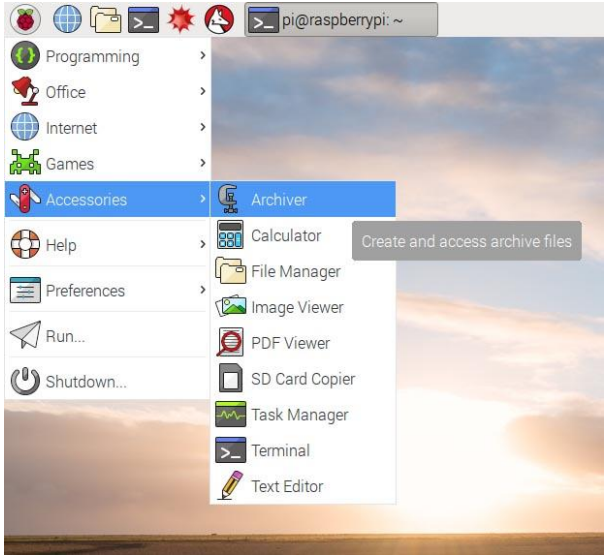
- In Bash, SSH

  ```
  sudo shutdown -r now
  ```

  OR

  ```
  sudo reboot
  ```

- On desktop, In RDP
  Go to the menu->Shutdown (Reboot)



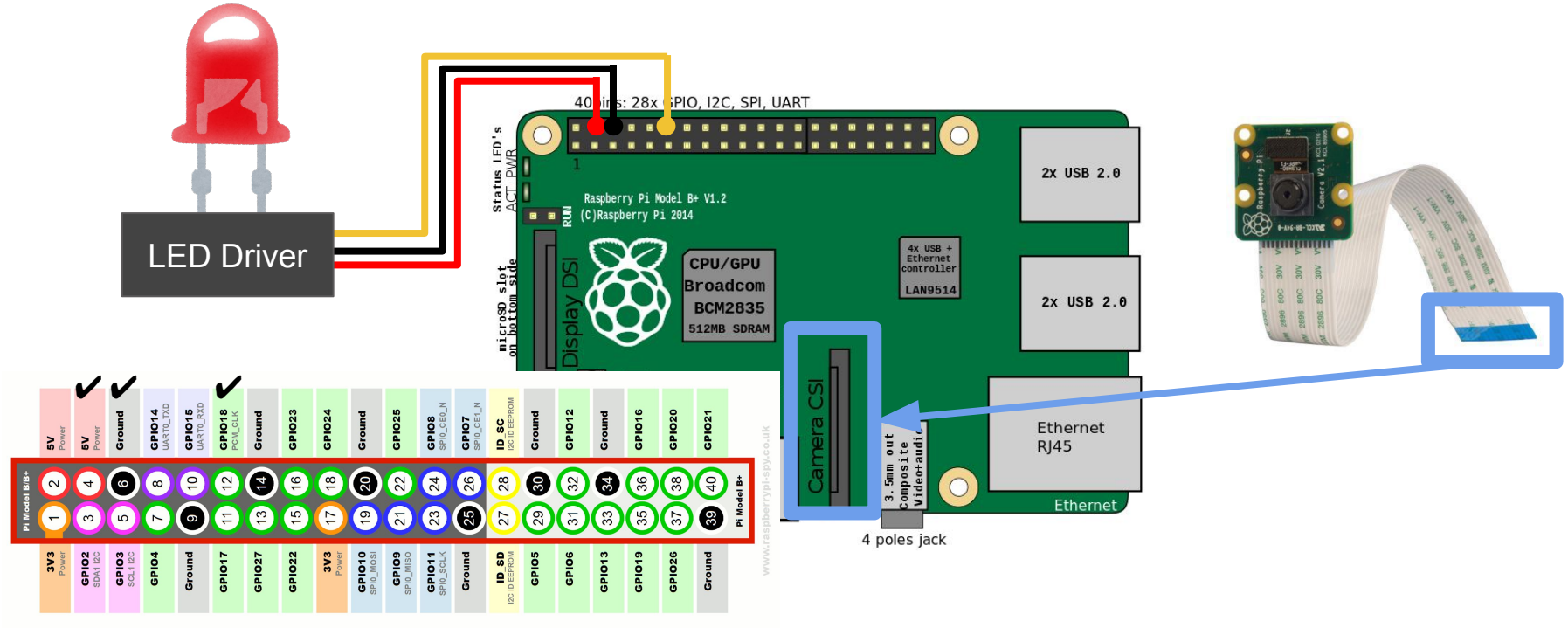
Raspbian is ready!
Enjoy Pi!!

Follow task specific instructions from this point and on.

Image Processing

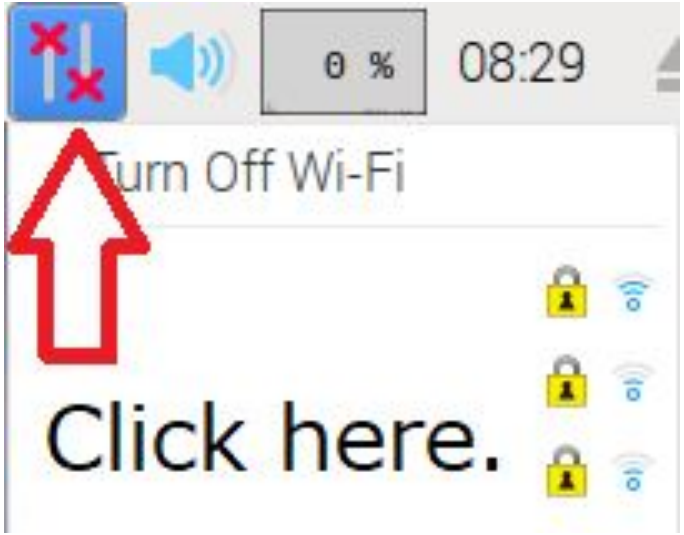
How to install & configure OpenCV on Raspberry Pi 3 Model B

Connect Devices



Establish WiFi Connection

- Select the SSID of Kitakyuhsu International Conference Center and input password.



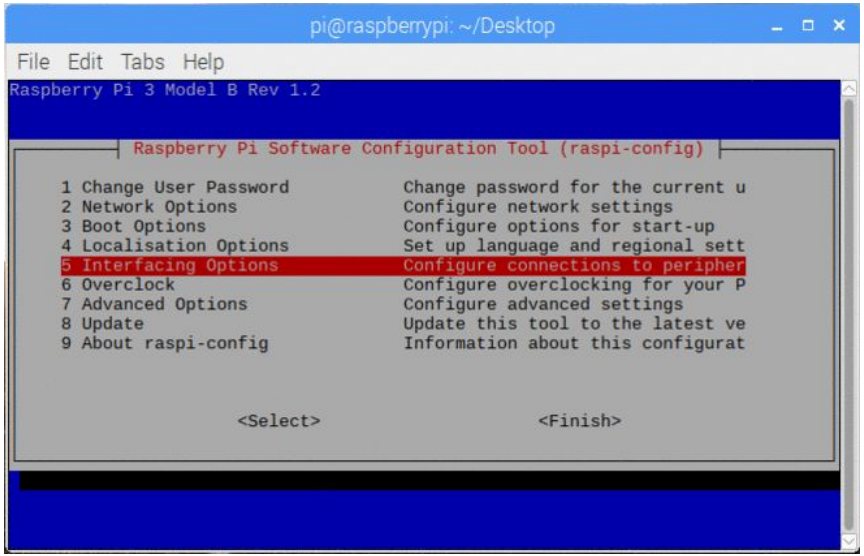
Raspberry Pi Configuration 1/2

- Open Raspberry Pi Configuration Window with this command.

```
$ sudo raspi-config
```

Raspberry Pi Configuration 2/2

- Raspi Camera should be enabled.



- After Rebooting, pls check WiFi connection

Install Packages to Raspberry Pi

- OpenCV (Image Processing Library)
- Raspberry Pi GPIO Library

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install libopencv-dev
$ sudo apt-get install python-opencv
$ sudo apt-get install python-rpi.gpio
```

Clone our project from GitHub

- https://github.com/shu-ramen/VeinVisualizationPi.git

```
$ cd [somewhere to put project files]
$ git clone
https://github.com/shu-ramen/VeinVisualizationPi.git
$ cd VeinVisualizationPi
```

Run our Vein Visualization System

- Run our project with these commands.

```
$ cd VeinVisualizationPi
$ python VeinVisualizer.py
```

Demonstration

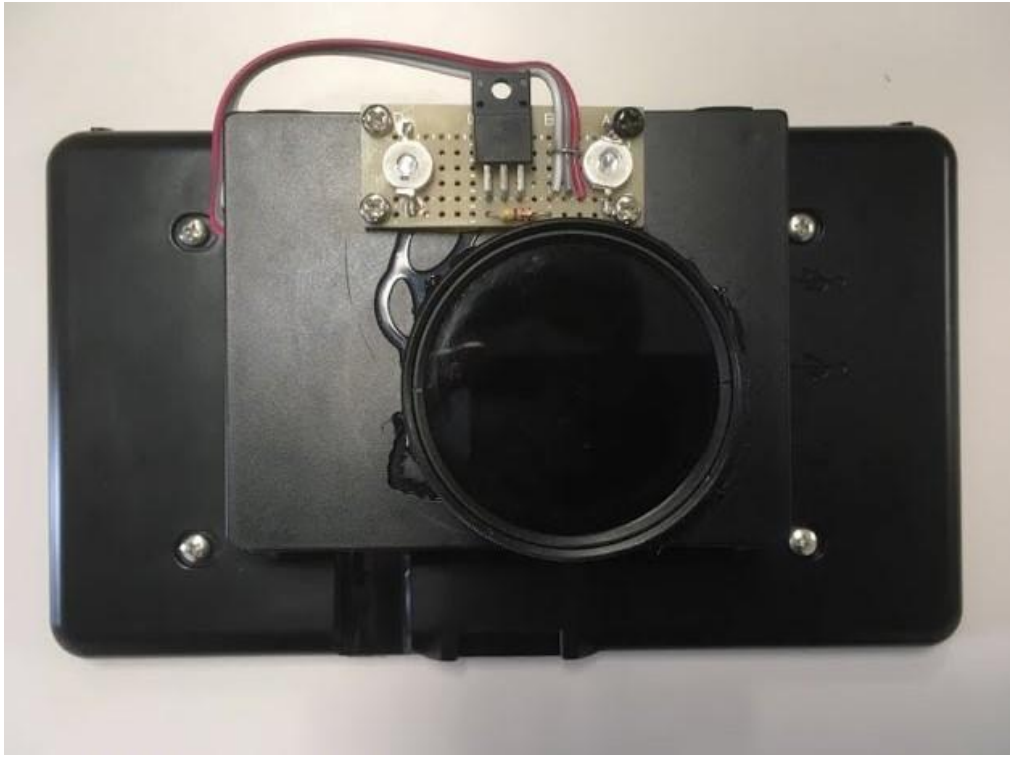


Image Processing Programming 1/7

Python Program

```
2   from picamera.array import PiRGBArray
3   from picamera import PiCamera
4   from datetime import datetime
5   import RPi.GPIO as gpio
6   import numpy as np
7   import time
8   import math
9   import cv2
10  import sys
11  import os
```

Comments

```
# PiCamera Control =
Image Obtaining

# GPIO COntrol = LED
Driver


# OpenCV = Image
Processing
```

Image Processing Programming 2/7

Python Program

```
22  # PWM
23  self.pwmPort = pwmPort
24  self.pwmFrequency = pwmFrequency
25  self.pwm = 0
26
27  # camera
28  self.frameWidth = frameWidth
29  self.frameHeight = frameHeight
30  self.framerate = framerate
31  self.camera = 0
32  self.rawCapture = 0
```

Comments

```
# GPIO Port = 12
# f = 1kHz




# 320 * 240 pixels

# 16 FPS
```

Image Processing Programming 3/7

Python Program

```
212 for frame in self.camera.capture_continuous(self.rawCapture,
                                                 format="bgr",
                                                 use_video_port=True):
...
215     img_original = frame.array
```

Comments

```
# Image Capturing. ⇒ Very Very Easy!!
```

Image Processing Programming 3/7

Python Program

```
212 for frame in self.camera.capture_continuous(self.rawCapture,
                                                 format="bgr",
                                                 use_video_port=True):
...
215     img_original = frame.array
```

Comments

```
# Image Capturing. ⇒ Very Very Easy!!
```

Original Image

Image Processing Programming 4/7

Python Program

```
99   # create contrast image
100 def __contrast(self, img):
101    gray = cv2.cvtColor
       (img,cv2.COLOR_BGR2GRAY)

102    resultImg = np.array
       ( [ self.lut[value] for value in gray.flat],
         dtype=np.uint8 )

103    resultImg = resultImg.reshape(gray.shape)
104    return resultImg
```

Comments

Color Conversion
(BGR to Gray)

Contrast Adjustment
With Sigmoid Function

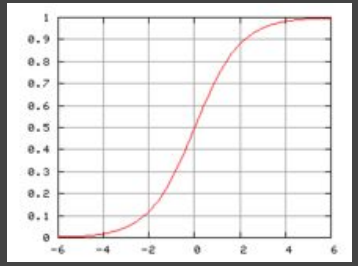


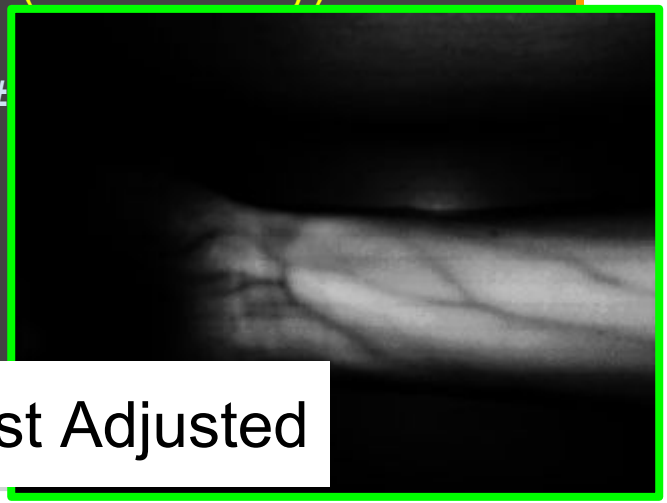
Image Processing Programming 4/7

Python Program

```
99   # create contrast image
100 def __contrast(self, img):
101    gray = cv2.cvtColor
       (img,cv2.COLOR_BGR2GRAY)

102    resultImg = np.array
       ( [ self.lut[value] for value in gray.flat],
         dtype=np.uint8 )

103    resultImg = resultImg.reshape(gray.shape)
104    return resultImg
```

Comments

```
# Color Conversion
   (BGR to Gray)

#
```



Contrast Adjusted

Image Processing Programming 5/7

Python Program

```
106     # get hand mask
107     def __getHandMaskAndOutline(self, contrast):
108         ret, thresh = cv2.threshold(contrast, 0, 255, cv2.THRESH_BINARY +
                                        cv2.THRESH_OTSU)
109         contours, hierarchy = cv2.findContours(thresh, cv2.RETR_TREE,
                                                   cv2.CHAIN_APPROX_SIMPLE)
110
111         # Get maximum size convex
112         maxIdx = -1
113         maxLen = -1
114         for i in range(len(contours)):
115             if (maxLen < len(contours[i])):
116                 maxIdx = i
117                 maxLen = len(contours[i])
118
119         # Create Mask
120         mask = np.zeros(contrast.shape, dtype=np.uint8)
121         mask.fill(0)
122         outline = -1
123         if maxIdx > 0:
124             outline = [ contours[maxIdx] ]
125             cv2.fillPoly(mask, outline, (255))
126
127         return mask, outline
```

Comments

Binarization

Get Contours of White Area

Get Maximum Contour
⇒ Hand Area

Create Mask of Hand Area

Image Processing Programming 5/7

Python Program

```
106     # get hand mask
107     def __getHandMaskAndOutline(self, contrast):
108         ret, thresh = cv2.threshold(contrast, 0, 255, cv2.THRESH_BINARY +
                                        cv2.THRESH_OTSU)
109         contours, hierarchy = cv2.findContours(thresh, cv2.RETR_TREE,
                                                   cv2.CHAIN_APPROX_SIMPLE)
110
111         # Get maximum size convex
112         maxIdx = -1
113         maxLen = -1
114         for i in range(len(contours)):
115             if (maxLen < len(contours[i])):
116                 maxIdx = i
117                 maxLen = len(contours[i])
118
119         # Create Mask
120         mask = np.zeros(contrast.shape, dtype=np.uint8)
121         mask.fill(0)
122         outline = -1
123         if maxIdx > 0:
124             outline = [ contours[maxIdx] ]
125             cv2.fillPoly(mask, outline, (255))
126
127         return mask, outline
```

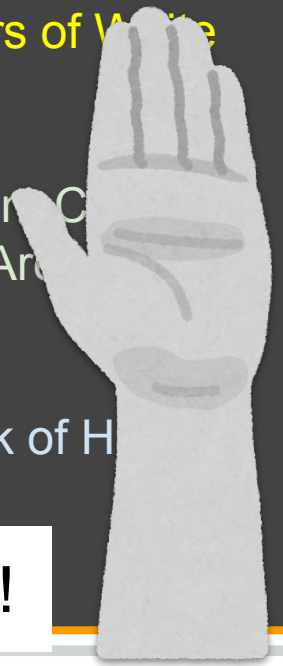
Comments

Binarization

Get Contours of W… Area

Get Maximu… C… ⇒ Hand Ar…

Create Mask of H… Area



Hand Detection!!

Image Processing Programming 6/7

Python Program

```
130 def __getVeinMask(self, contrast):
131     veinMask = cv2.Canny(contrast,
                self.lut_b_val-40, self.lut_b_val+5)
132     kernel = np.ones((3, 3), np.uint8)
133     veinMask = cv2.dilate(veinMask, kernel)
134     return veinMask
```

Comments

```
# Edge Detection

# Pixel Dilation
```

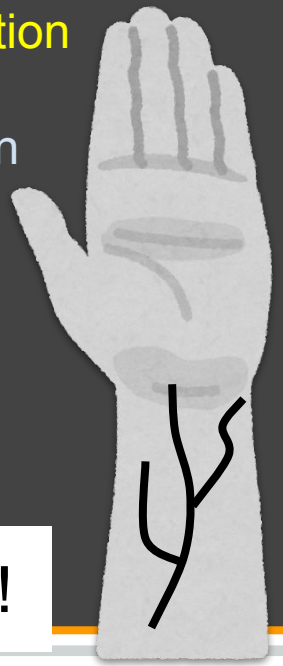
Image Processing Programming 6/7

Python Program

```
130 def __getVeinMask(self, contrast):
131     veinMask = cv2.Canny(contrast,
                self.lut_b_val-40, self.lut_b_val+5)
132     kernel = np.ones((3, 3), np.uint8)
133     veinMask = cv2.dilate(veinMask, kernel)
134     return veinMask
```

Comments

```
# Edge Detection

# Pixel Dilation
```



Edge Detection!!

Image Processing Programming 7/7

Python Program

```
157   def __getVein(self, handMask, outline, veinMask,
                                                    original):
158       if outline == -1:
159           return original
160
161       red = np.zeros(original.shape, np.uint8)
162       red[:] = (0, 0, 255)
...
166       vein = cv2.bitwise_and(red, red, mask=handMask)
167       vein = cv2.bitwise_and(vein, vein, mask=veinMask)
168       vein = cv2.bitwise_or(original, vein)
169
170       cv2.drawContours(vein, outline, 0, (0, 255, 255))
171
172       return vein
```

Comments

If Hand Area was not detected.

Overwrap Hand Area Mask and Vein Area Mask

⇒ Vein Emphasize

Image Processing Programming 7/7

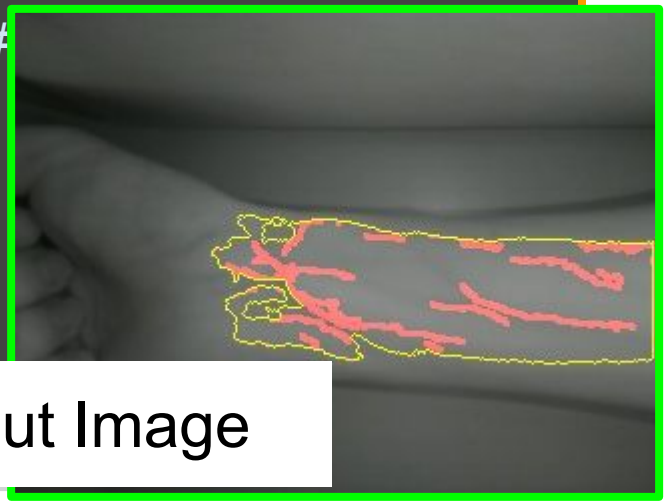
Python Program

```
157   def __getVein(self, handMask, outline, veinMask,
                                                    original):
158       if outline == -1:
159           return original
160
161       red = np.zeros(original.shape, np.uint8)
162       red[:] = (0, 0, 255)
...
166       vein = cv2.bitwise_and(red, red, mask=handMask)
167       vein = cv2.bitwise_and(vein, vein, mask=veinMask)
168       vein = cv2.bitwise_or(original, vein)
169
170       cv2.drawContours(vein, outline, 0, (0, 255, 255))
171
172       return vein
```

Comments

If Hand Area was not detected.

#



Output Image

Vein Visualization

I'll talk about...

1. About Vein Visualization
2. How to Visualize the Vein
3. Developing Hardware

I'll talk about...

1. About Vein Visualization
2. About Visualize the Vein
3. Developing Hardware

About Venipuncture

Venipuncture is medical practice.

Performed high frequency and high invasiveness.

Ex: Blood sampling , Injection of medicine

Therefore, importance of practical training is high.

Every year, caused some accidents.

Ready Made Products

http://www.noba.com/what-is-a-veinviewer/

“Vein Viewer”

Projection shape of Vein
On Human skin.

Reported increase performance for venipuncture on the practical affairs.

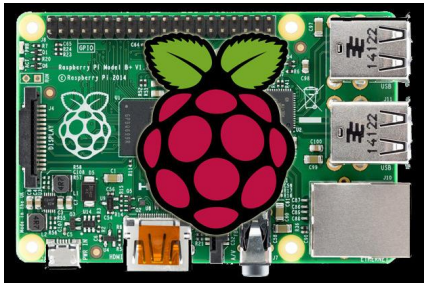
Vision

Ready-made-products are extremely expensive.

Our system's Concept:

- Easy to developing
- Economy
- Connect to medical IoT platform

  (this is a part of blood vessel information)

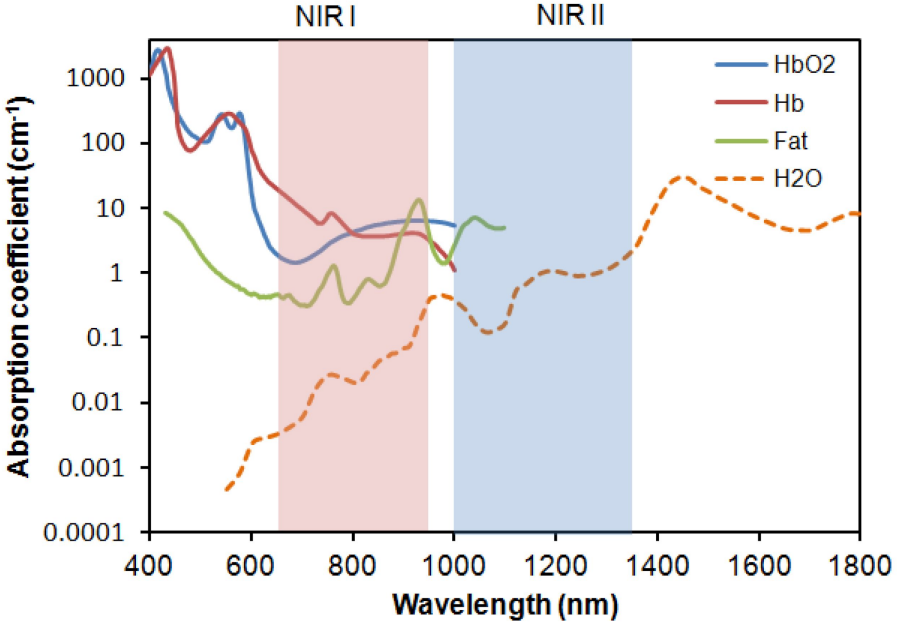


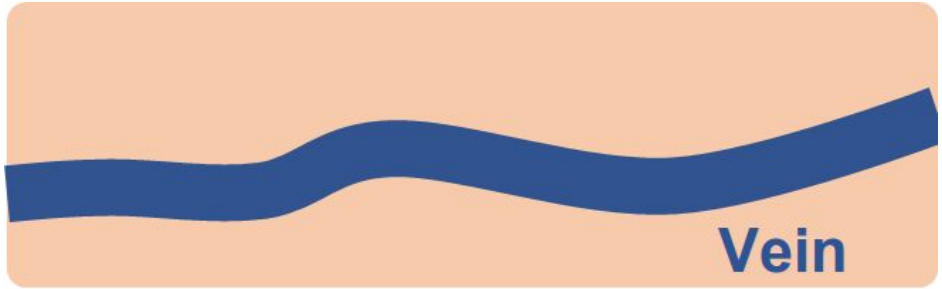
I'll talk about...

1. About Vein Visualization
2. How to Visualize the Vein
3. Developing Hardware

How to Visualize the Vein

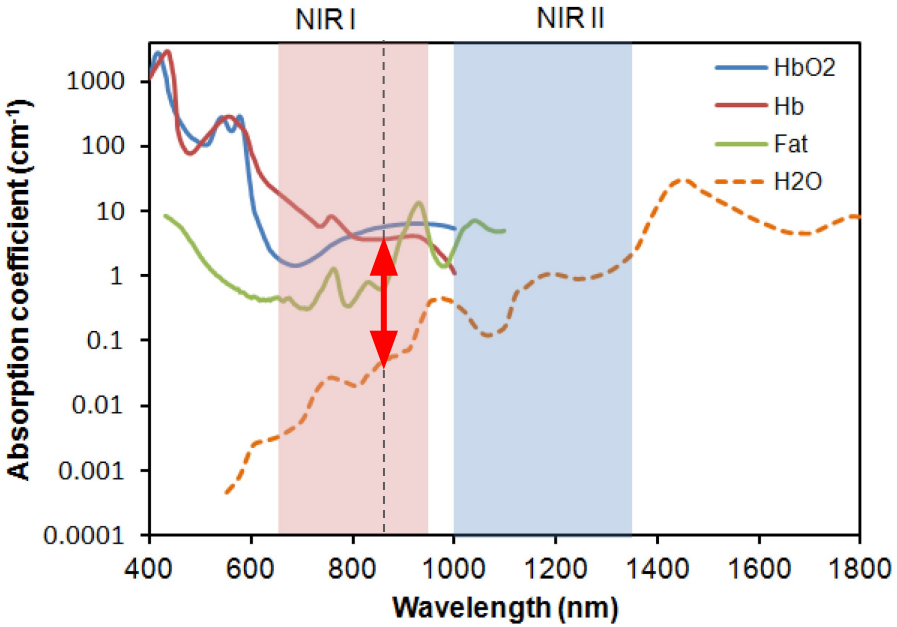
The Hb and HbO2 have high extinction coefficient on IR.

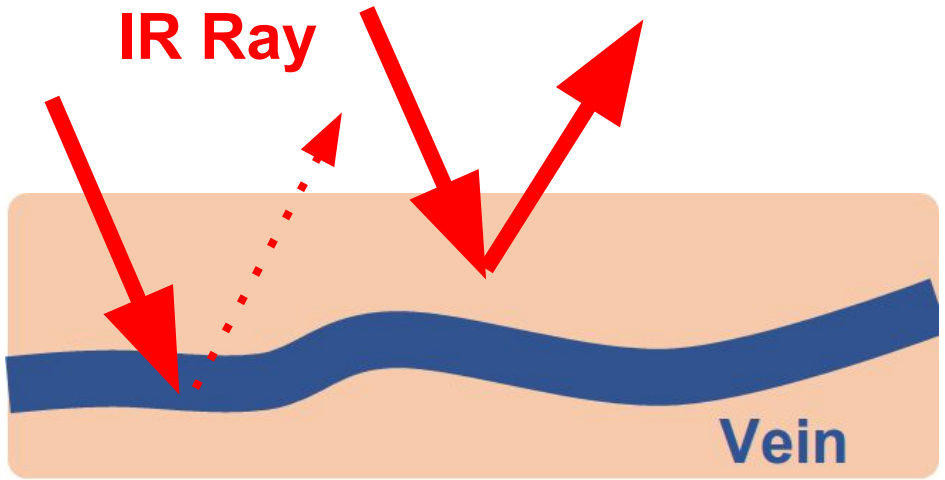




How to Visualize the Vein

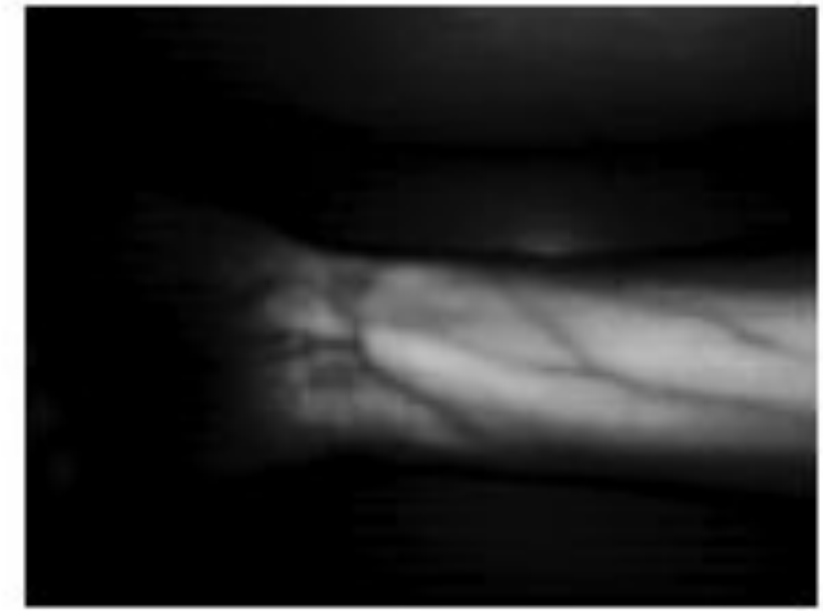
The Hb and HbO2 have high extinction coefficient on IR.

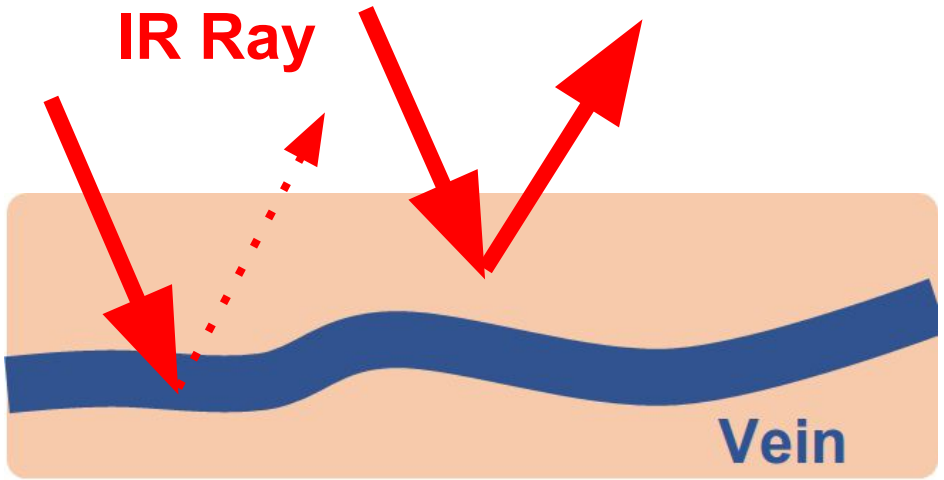




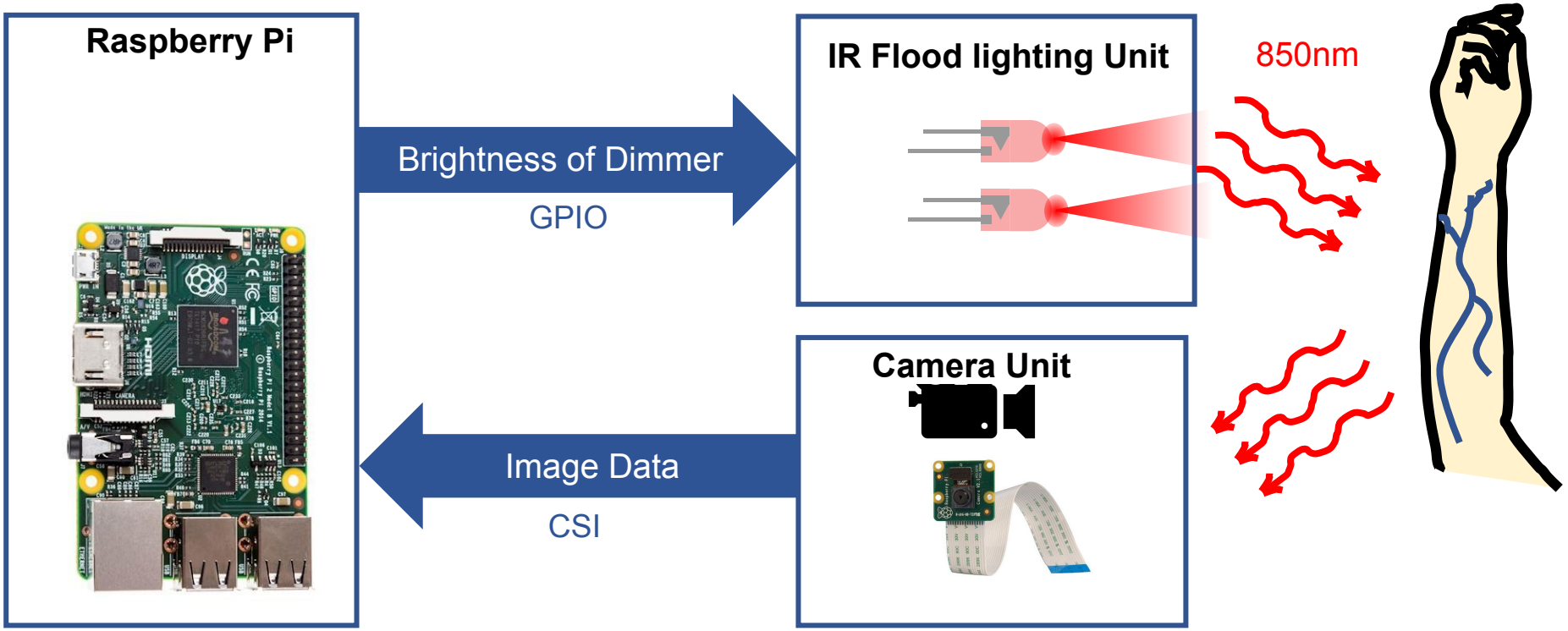
How to Visualize the Vein

The Hb and HbO2 have high extinction coefficient on IR.

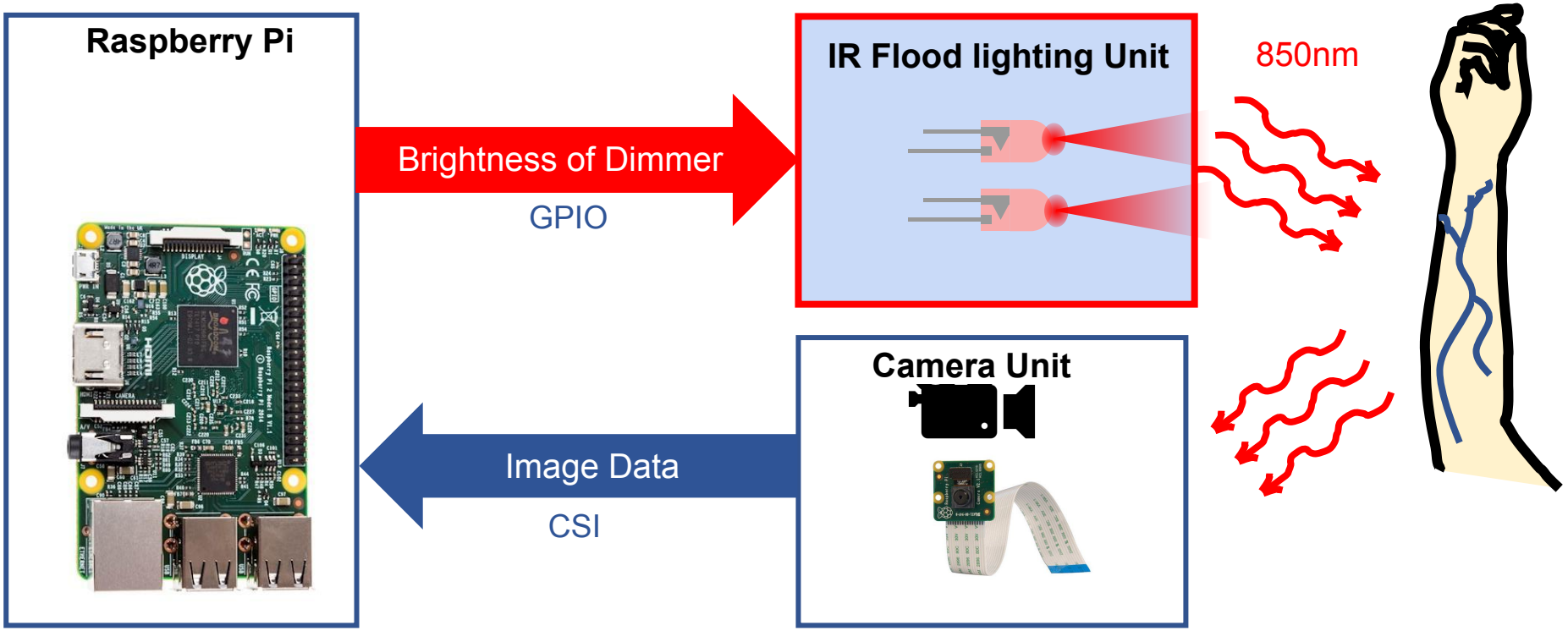




System Diagram



System Diagram

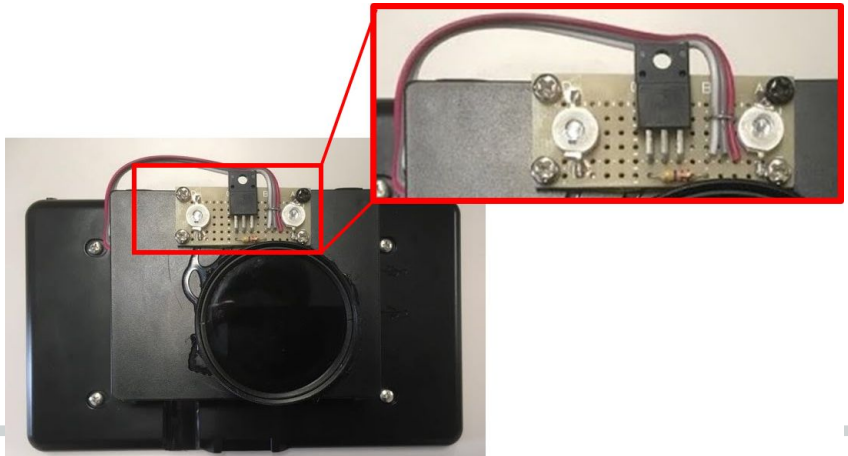


IR Flood light Unit

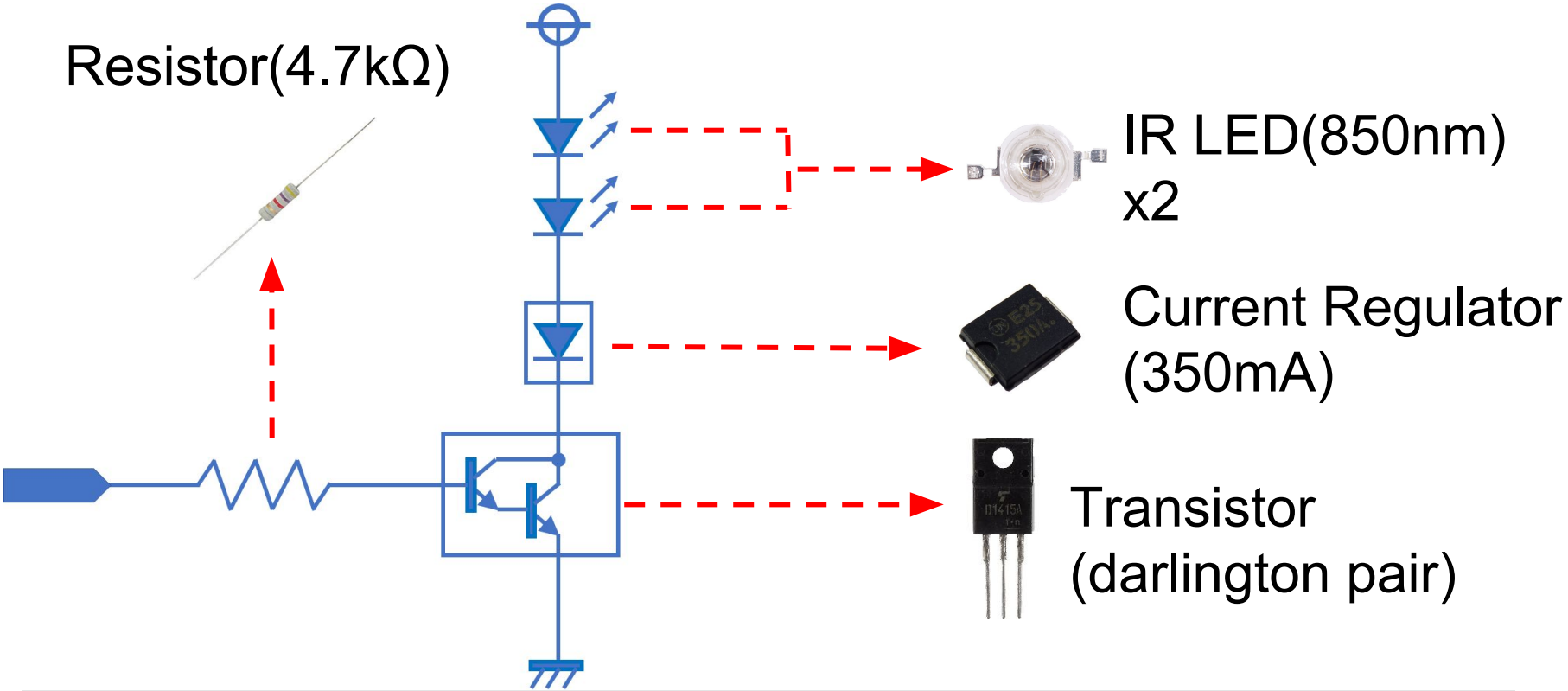
Function:

Flood lighting: Irradiate IR Ray.

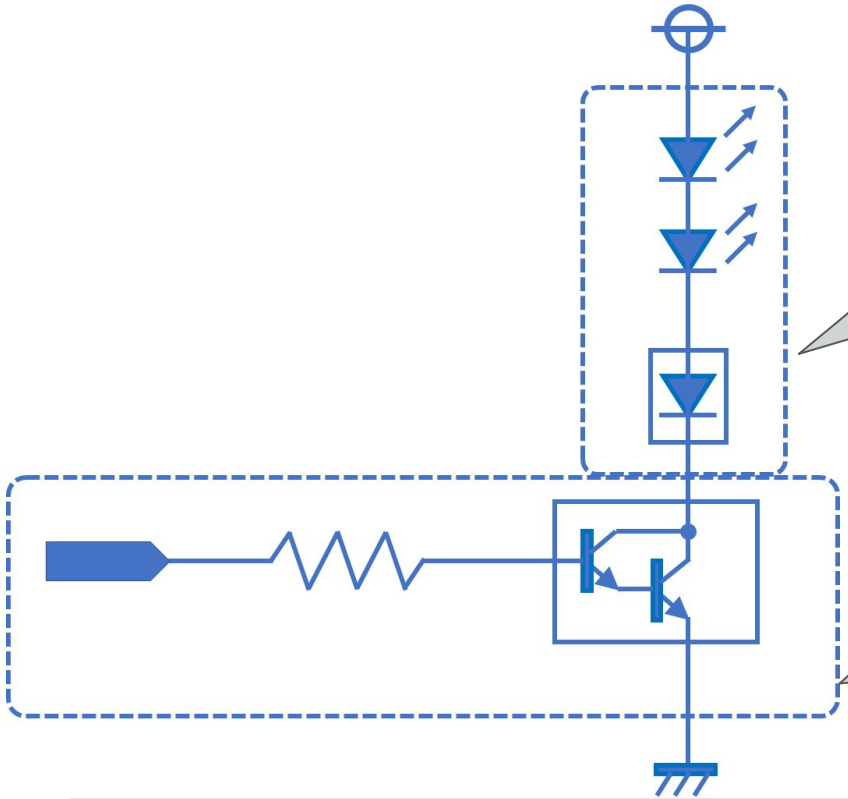
Control: Adjusting brightness of IR Ray.



Circuit Diagram (Flooding Light Unit)



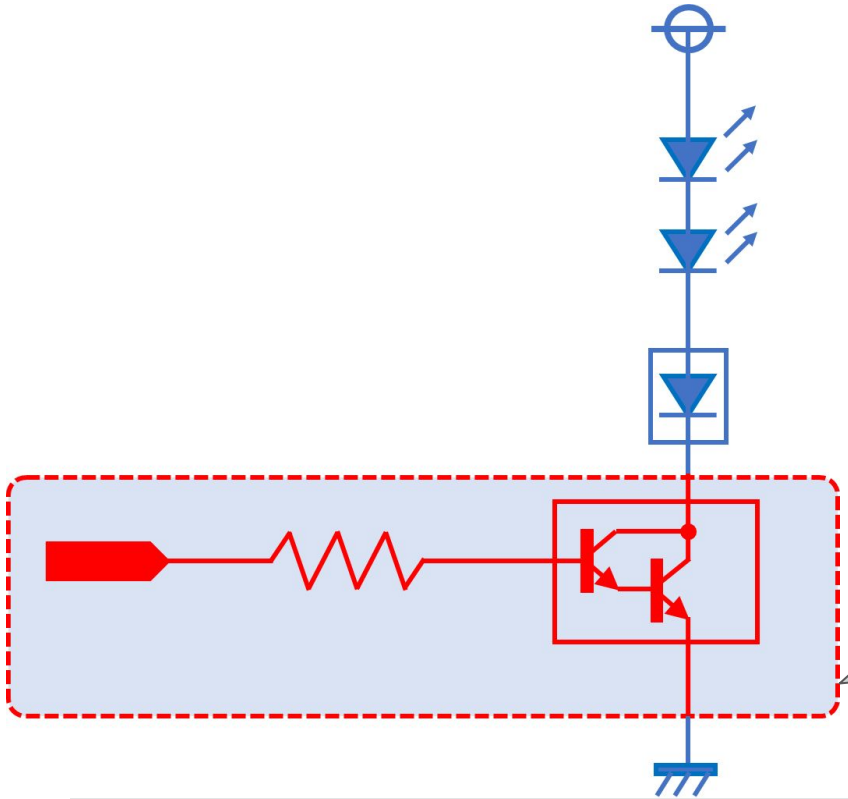
Circuit Diagram (Flooding Light Unit)



Lighting Circuit:
Lighting LED
Current Regulator

LED Drive Circuit:
Switching

Block 1 -LED Drive Circuit



LED Drive Circuit:
Switching for
High-Power LED

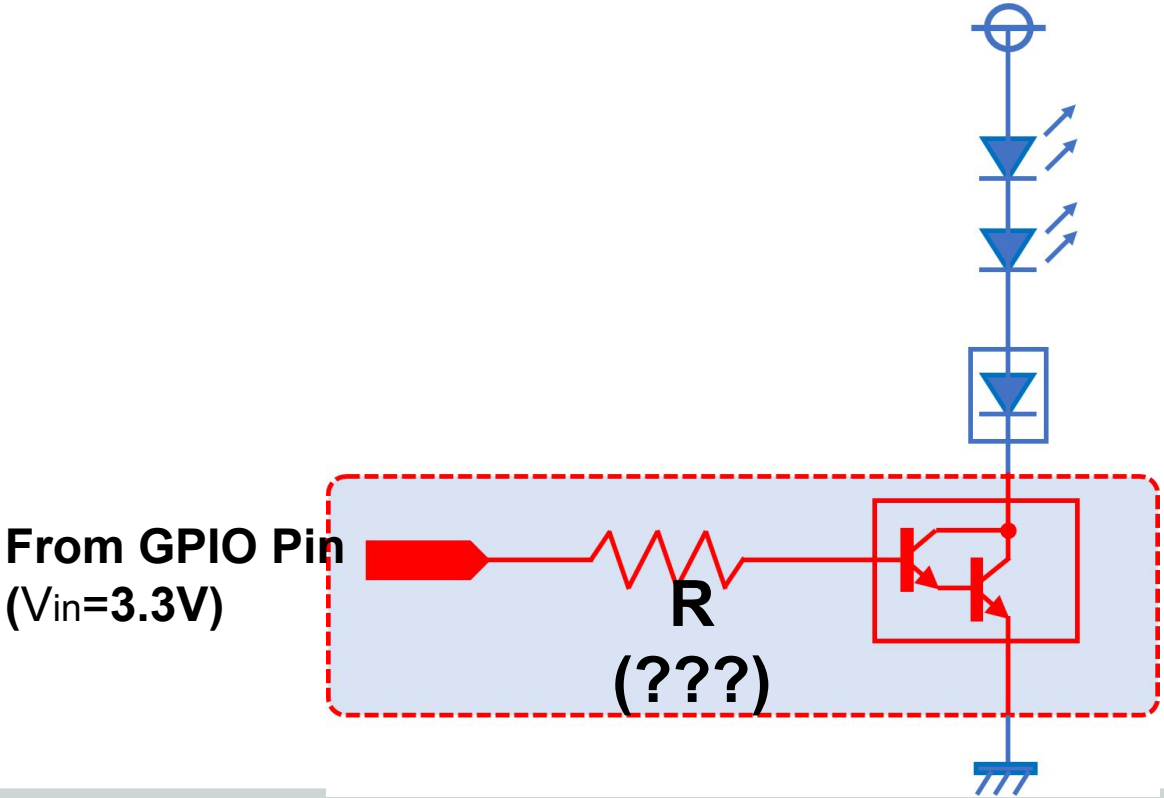
Current value from GPIO Pin is not enough for LED drive

How much the current value?

- Maximum LED Lightning current is **700mA**
- However, GPIO Pin's current limit is **50mA**

In this Case, Employing Switching Circuit using Transistor.

How to Design -LED Drive Circuit



Decide Base Resistor's value

V_{in}=3.3[V] (GPIO outputs Voltage),

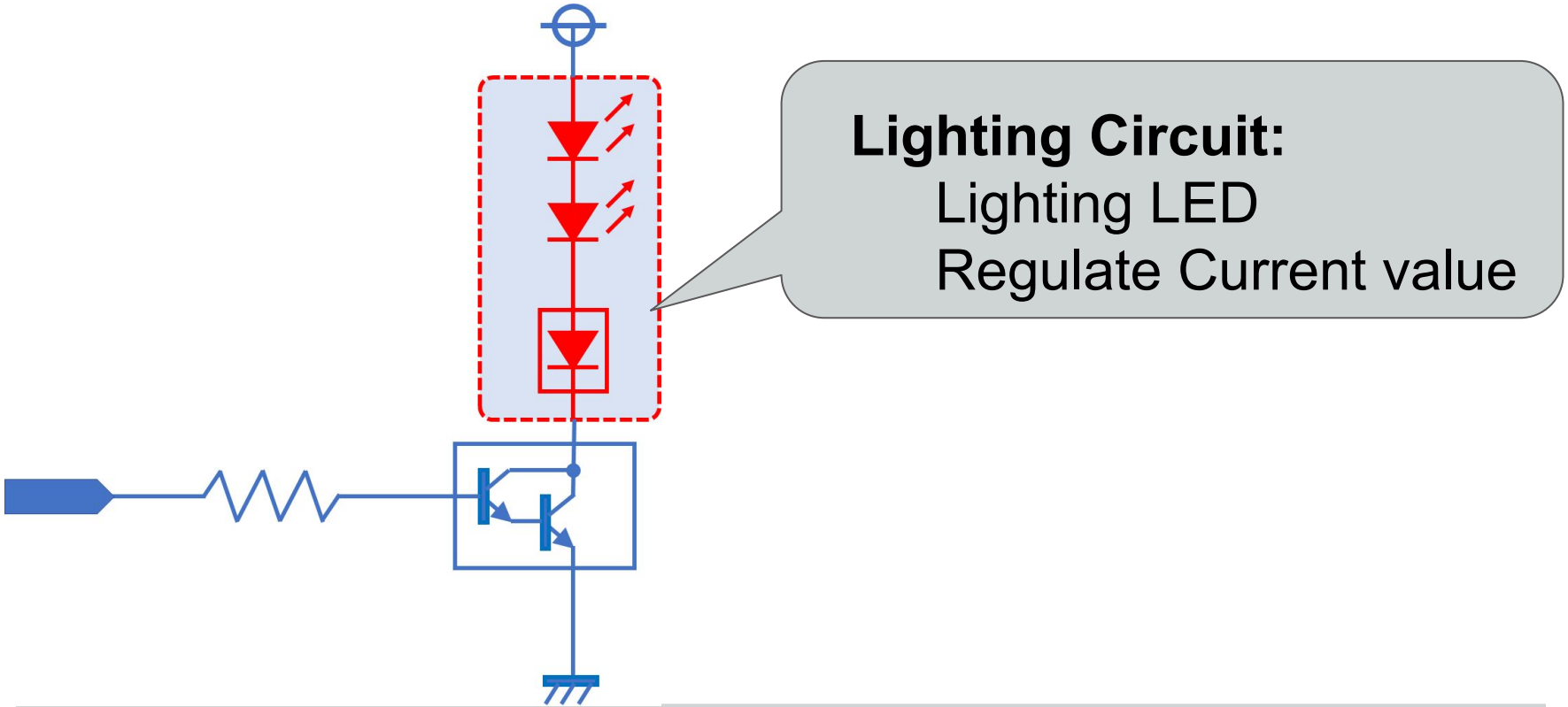
h_{fe}=1000 (Transistor's parameter),

Maximum Current for Drive LED=700 [mA]

$$R = \frac{V_{in}}{i_c/h_{fe}} = \frac{3.3}{0.7/1000} = 4714.28$$

R=4.7kΩ

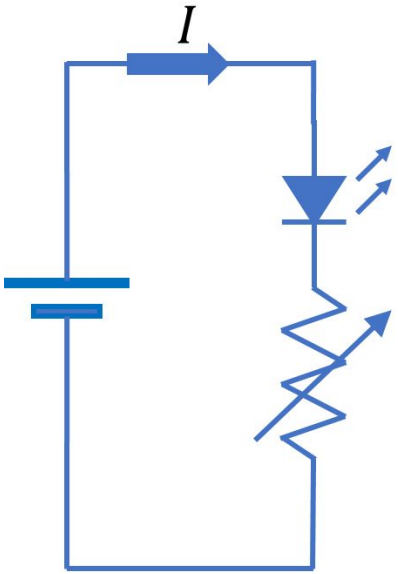
Block 2-LED Circuit:

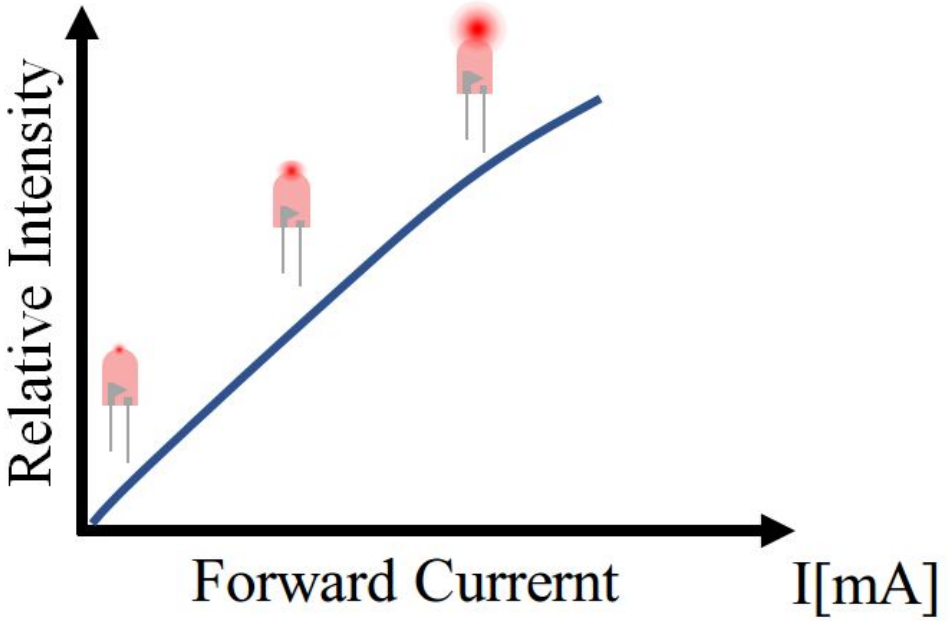


Lighting Circuit:
Lighting LED
Regulate Current value

LED's Brightness is decided by Current Value

The brightness of the light of an LED is proportional to an electric current.





Heat of LED Cause Changing Current Value

When the almost semiconductor devices has heat,
The current value increase.

Hi-Power LEDs self heating caused changing current value (= brightness).

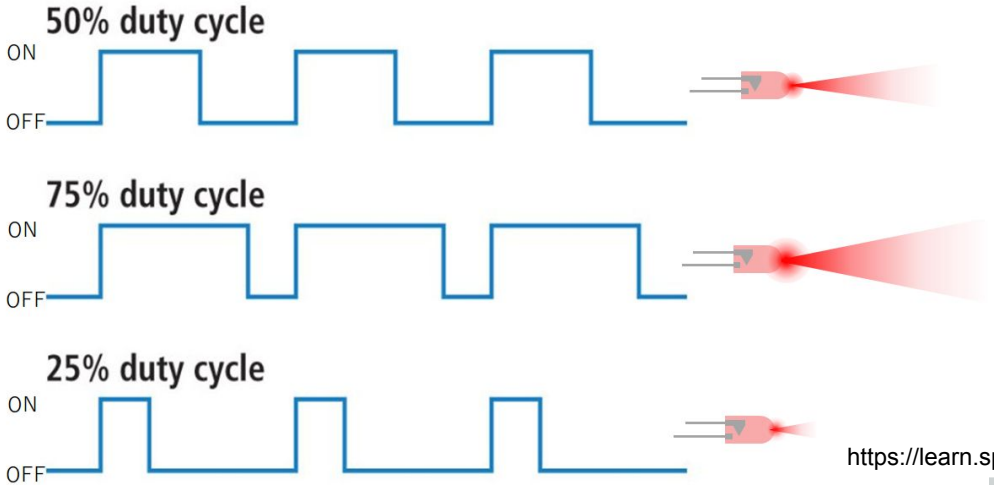
Therefore, We insert current regulator (350mA) to lightning circuit to get constant current value.

How to Adjusting Brightness

GPIO Pin outputs are only HIGH(3.3V) or LOW(0V).

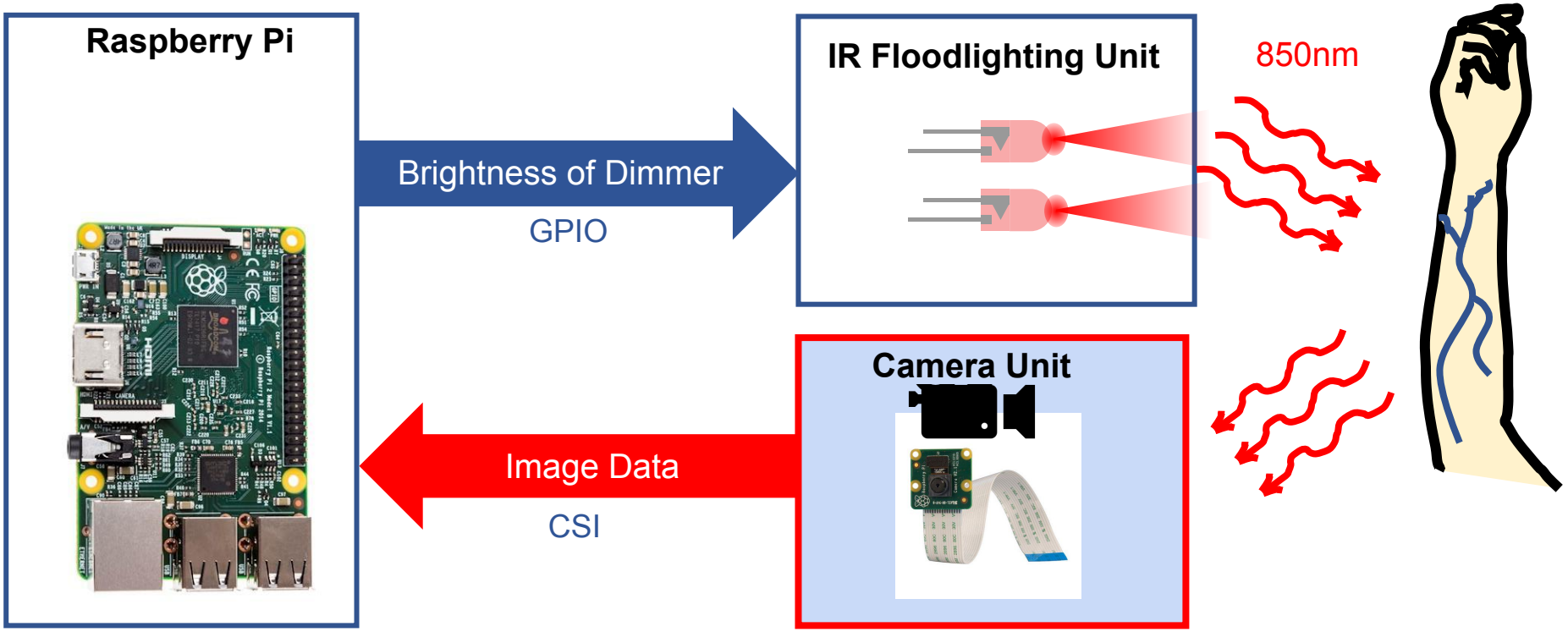
PWM(Pulse Width Modulation):

High frequency switching for control power.



https://learn.sparkfun.com/tutorials/pulse-width-modulation

System Diagram

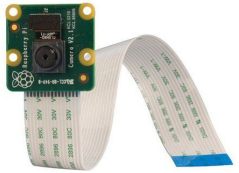


Camera Unit

Function

Camera:

Get pixel Information by Image sensor

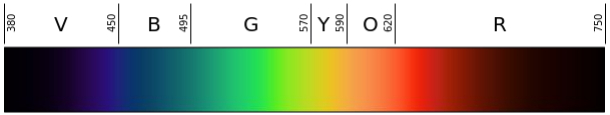


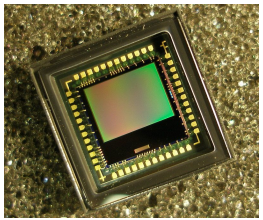
Filter:

Cut Visible light and get only IR Ray.

Almost Camera can’t take IR Ray

The CMOS Image sensor can convert invisible Ray(IR , UV) to electrical signal too.



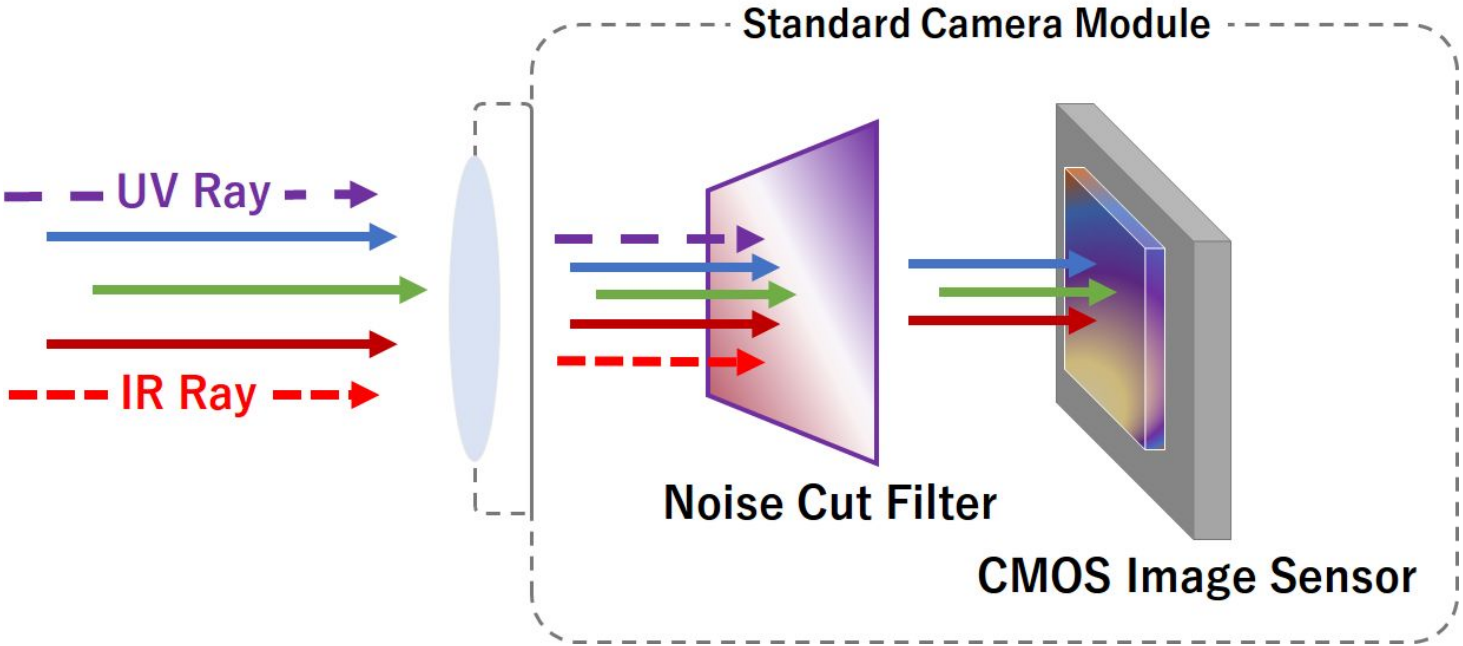


However, Invisible color informations on a Picture is noise for human.

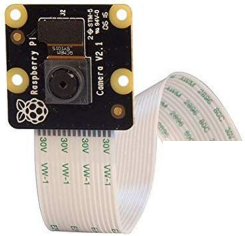
Almost digital camera include Noise Cut Filter.

Standard Camera Module

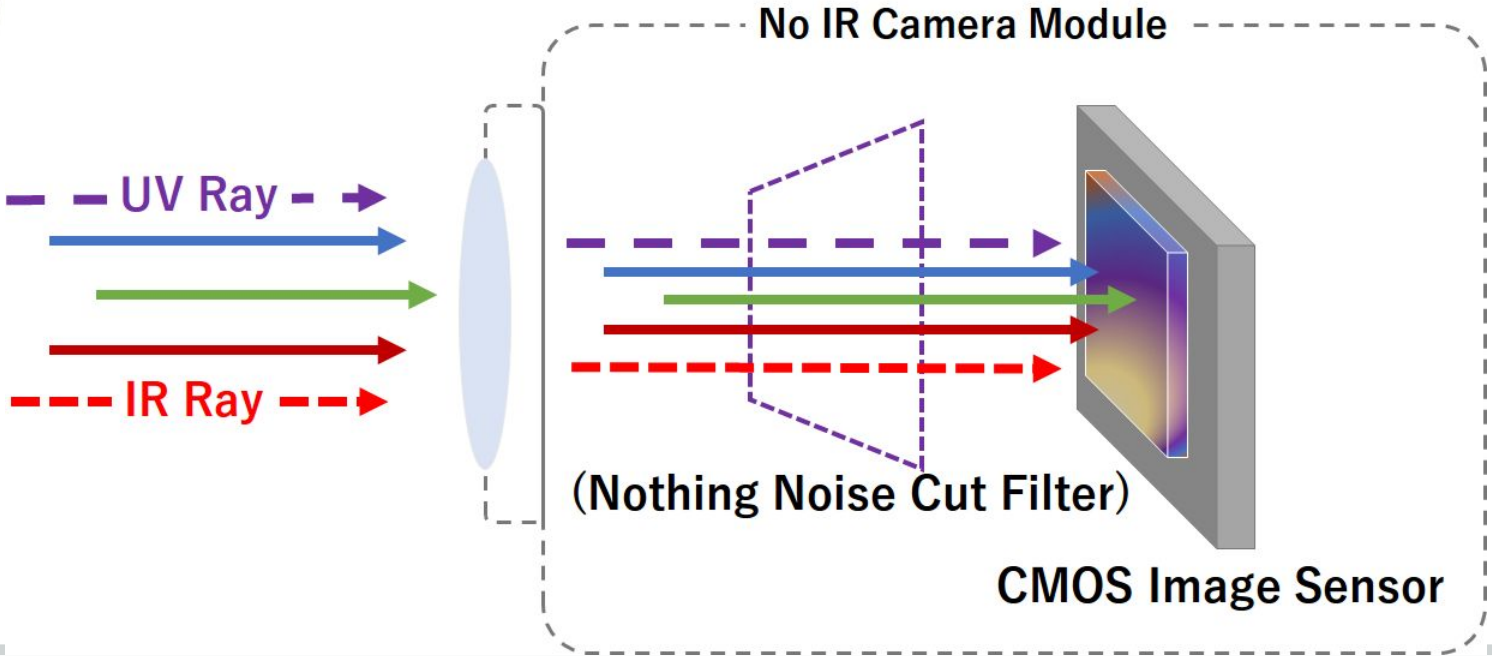
Almost digital camera include Noise Cut Filter.



We want to IR Ray information



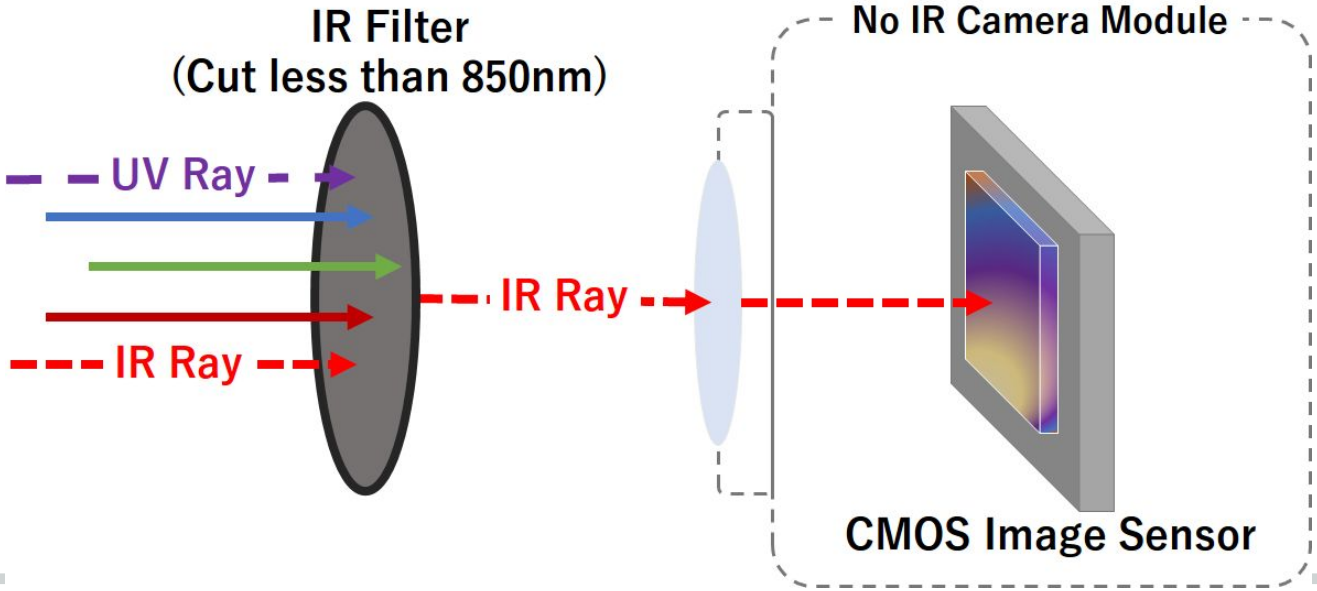
In No IR Camera, nothing Noise Cut Filter.



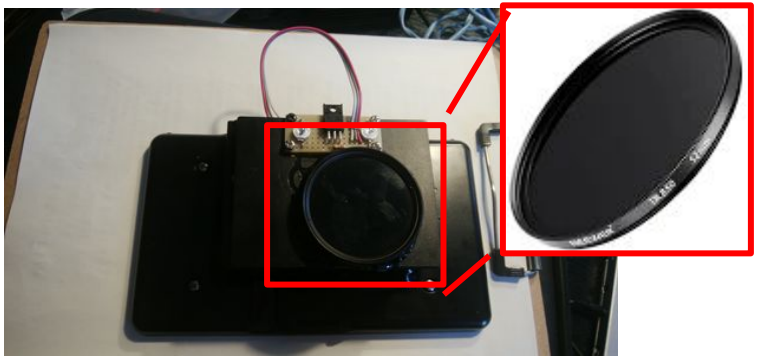
We only need IR light information!

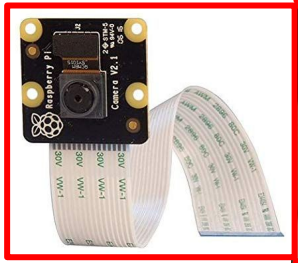
The IR Filter cut less than specific wavelength rays.

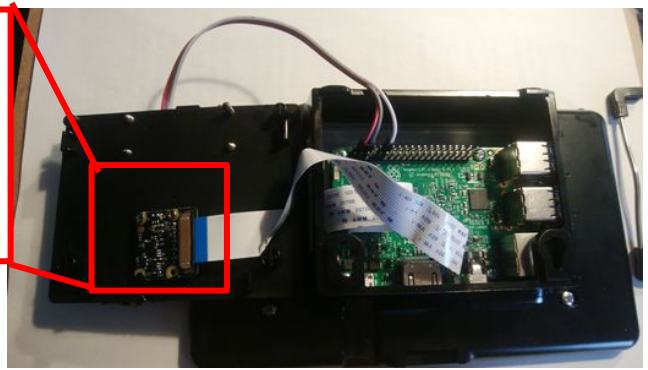
To get crearly vein shape... To simplify algolism...

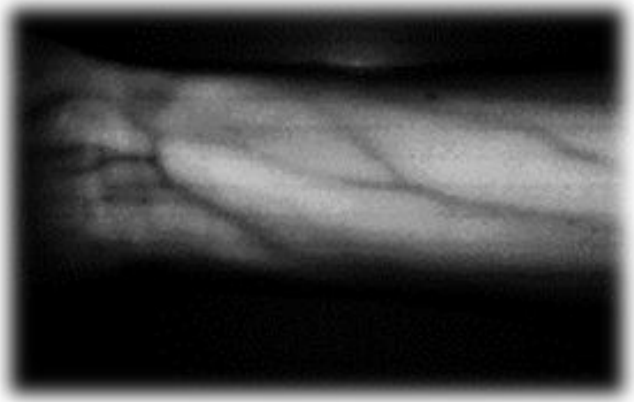


Enjoy!









Question?
Demonstration

Contact Atsushi Inoue (Business Card: https://goo.gl/Ltho3I, Web: http://www.inoueatsushi.net/)

This tutorial is based on the R&D to be presented at ICIEV 2018.

13:20-14:50 on Wednesday, June 27 @Room 22
Regular session (7-B): IoT, Control and Automation III
Paper 282 -- the last presentation.

A Modern, IoT Platform for Medical Engineering: Vein Visualization

Shu Isaka*, Takumi Kitajima†, Hiroharu Kawanaka*, and Atsushi Inoue‡

* Department of Electrical and Electronic Engineering, Mie University, Mie, Japan
Tel : +81-59-231-9396; E-mail: shu.h7.research@gmail.com
† Department of Public Health and Occupational Medicine, Mie University, Mie, Japan
Tel : +81-59-232-5576; E-mail: takukitaji@gmail.com
‡ Department of Information Systems and Business Analytics, Eastern Washington University, USA
Tel : +1-509-828-1228; E-mail: ainoue@ewu.edu

You are welcome to join us to discuss further.