## RaspberyPi4の準備 OS/電源など

講演時間15分 (株)人工知能ロボット研究所

西村輝一





## ㈱人工知能ロボット研究所

高齢化が進む中、ロボットへの期待は膨らむが・・・・安全性、信頼性、ロバスト性、ランニングコストは?



3K職場ロボット 17 ・定行アシリ

農業ロボット

物流ロボット

3K職場ロボット
 農業ロボット
 物流ロボット
 介護ロボット
 医療ロボット

## RasPi4への期待

RasPi3



RasPi4



### ASUS Tinker Board S



Intel Z8350



¥23,930

### Jetson Nano



¥12,540

# ¥7,400

### ロボット用CPU要求仕様

- -Ubuntu LTS/ROS1/ROS2搭載
- -Stereo Camera搭載

(embedded MultiMediaCard)

-on Board Disk

1.CPUの高速化/メモリ4GB 2.USBポートの高速化 3.boot用EEPROM搭載 4.低価格

## RaspberyPi4の準備

- RasPi4 4Gbyte memory
  - memoryは4GByteのものを。
- ◆ SDカード
  - ◆ 高速のもの/32Gbyte以上。同じものを使用すると良い、
- ◆ 5V電源
  - ◆ **5.3Vから5.5V** 3A以上。雷マークが出ない。
  - ◆ DC-DCコンバータは出力可変のものを使用。3A以上。
- ◆ Heat sink+FANの装着
  - ◆ CPU温度は60度以下を目安に。温度計マーク80℃
- HDMI
  - ◆ シールドがしっかりしているもの。WiFi接続トラブルの原因
- キーボード・マウス
- オーディオアンプ
  - ◆ シールド線使用の事











LinkinPerk ハイスピード HDMI ケーブル Micro HDMI to HDMI変換 オス・オス,イーサネット 4K|1080p | 3D,金メッキコネクタ搭載 プレミアムアルミケース

## RaspberyPi4のOS

- Raspbianのinstall
  - SD Formatter
     FAT32で再フォーマット



Win32DiskImager



https://www.raspberrypi.org/downloads/



◆ EEPROMのアップデート

https://www.raspberrypi.org/documentation/hardware/raspberrypi/booteeprom.md

- \$ sudo rpi-eeprom-update
- OpenCV4

https://www.pyimagesearch.com/2019/09/16/install-opencv-4-on-raspberry-pi-4-and-raspbian-buster/







## Ubuntu/ROSのインストール

UbuntuはLinuxのディストリビューションの一つ. UbuntuはDebianというLinuxディストリ ビューションから派生

• Ubuntu

Ubuntu 14.04 LTS ⇒Ubuntu 16.04 LTS⇒Ubuntu 18.04 LTS(LTS: Long Term support)

- (Ubuntu Trusty) (Ubuntu Xenial) (Ubuntu Bionic)
- ROS1 indigo Igloo
- ROS2

kinetic Kame Eloquent Elusor melodic Morenia Dashing Diademata

- Ubuntu 18.04 install
   <u>https://jamesachambers.com/raspberry-pi-4-ubuntu-server-desktop-18-04-3-image-unofficial/</u>
- OpenCV4のinstall

https://opencv.org/

http://milq.github.io/install-opencv-ubuntu-debian/

ROS2/Python3の install

https://index.ros.org/doc/ros2/Installation/

http://lunablanca.dip.jp/ros2-install/

## Ubuntu/ROS2動作確認

#Stereo Cameraの確認プログラム #!

### 実演: \$ ros2 run arm 2cam



#### <参考> #CPU温度の測定プログラム

import subprocess import time from time import sleep cmd="cat /sys/class/thermal/thermal\_zone0/temp" while(1): aa=subprocess.check\_output(cmd.split()) print ("CPU temp.=",(int(aa))/1000,"deg.C") sleep(6) #!/usr/bin/env python import time from time import sleep import subprocess import cv2 import sys import numpy as np #ros2 import rclpy from rclpy.node import Node import sensor msgs.msg from cv bridge import CvBridge # def main(): capH = cv2.VideoCapture(0) rclpy.init() node = rclpy.create node("Source") pub\_img = node.create\_publisher(sensor\_msgs.msg.lmage, "img\_topic") time.sleep(1.0) cvb = CvBridge()print ('start') while True: ret, dataH = capH.read(-1) image1=cv2.cvtColor(dataH,cv2.COLOR\_BGR2RGB) pub\_img.publish(cvb.cv2\_to\_imgmsg(image1,"rgb8")) cv2.imshow('frameH',dataH) if cv2.waitKey(1) & 0xFF == ord('q'): break #the end print ("finished!!!") capH.release() cv2.destroyAllWindows() if \_\_name\_\_ == "\_\_main\_\_": main()

## RasPi4+OpenCV による色認識



講演時間15分 (株)人工知能ロボット研究所 西村輝一

## 画像処理の活用

- 色認識
- ・テンプレートマッチング
- Key-pointマッチング
- 機械学習
  - HAAR
  - HOG
- 深層学習
  - 分類/Classification
  - 物体認識
  - ・セグメンテーション
- Depth Point cloud

## 色について/HSV

<u>(参考)ウィキペディアより</u>

・色相(hue)は、赤、黄、緑、青、紫といった色の様相の相違である。特定の波長が際立っていることによる変化であり、際立った波長の範囲によって、定性的に記述できる。ただし、常に同じ波長が同じ色に見える訳ではない。 赤から、黄、緑、青を経て、紫までは、スペクトル上の色であると言える。

・彩度、明度と併せて、色の三属性と言う。色から彩度と明度または輝度の要素を取り除いた残りであるということもできる。



ツール: 色数値化プログラムの実演紹介

hsv[x, y, 0]=y hsv[x, y, 1]=255 htt[x, y, 2]=255 htt[x, y

for y in range(0,320):

### 実演:screen\_color\_set.py

```
def find color(img):
  global xx,yy
  hsv = cv2.cvtColor(img, cv2.COLOR_BGR2HSV_FULL)
  h = hsv[:, :, 0]
  s = hsv[:, :, 1]
  v = hsv[:, :, 2]
  mask = np.zeros(h.shape, dtype=np.uint8)
  mask[((h>xx) & (h<yy)) & (s>50)] = 255
  kernel = np.ones((10,10),np.uint8)
  mask = cv2.morphologyEx(mask, cv2.MORPH_OPEN, kernel)
  contour, hierarchy= cv2.findContours(mask, cv2.RETR EXTERNAL, cv2.CHAIN APPROX NONE)
  mmm=cv2.cvtColor(mask, cv2.COLOR GRAY2BGR)
  cv2.imshow('mask2',mmm)
  rects = []
  print (len(contour))
  for contour in contour:
    approx = cv2.convexHull(contour)
    rect = cv2.boundingRect(approx)
    rects.append(np.array(rect))
    #find top1
    nn=0
    for rect in rects:
       nn=nn+1
       #cv2.circle(img, (int(rect[0]+(rect[2]/2)),int(rect[1]+rect[3]/2)),int(rect[2]/2), (0, 0, 255), thickness=1)
       cv2.rectangle(img, tuple(rect[0:2]), tuple(rect[0:2] + rect[2:4]), (0, 0, 255), thickness=1)
       cv2.putText(img, str(nn), tuple(rect[0:2]), cv2.FONT_HERSHEY_SIMPLEX, 1.0, (255, 255, 255), thickness=1)
  return img
```

#

def draw\_sq(event,x,y,flags,param):



out points - points[mask]

## RasPi4+Dlibによる機械学習



講演時間15分 (株)人工知能ロボット研究所 西村輝一





## 比較的簡単な機械学習

- 色認識
- テンプレートマッチング
- Key-pointマッチング
- 機械学習
  - OpenCV3 HAAR
  - Dlib HOG
- 深層学習
  - 分類/Classification
  - 物体認識
  - セグメンテーション
- Depth Point cloud

## **OpenCV4 HAAR特徵量**

### カスケード分類器 Haar特徴:輝度の勾配方向の分布

### 実演:haar.py

```
import cv2
#
def main():
  cap = cv2.VideoCapture(0)
  cascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
  while(cap.isOpened()):
    img = cap.read()[1]
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    face = cascade.detectMultiScale(gray, scaleFactor=1.2, minNeighbors=1, minSize=(30, 30))
    for (x, y, w, h) in face:
       cv2.rectangle(img, (x, y), (x + w, y+h), (0,0,300), 4)
    cv2.imshow("frame", img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
       break
  cap.release()
  cv2.destroyAllWindows()
#
if name == ' main ':
  main()
```

## 画像処理ライブラリーdlib

### OpenCV3になって消えたHog特徴量。dlibがすごいらしい。

#### SwapFile作成

\$ git clone https://github.com/JetsonHacksNano/installSwapfile

\$ cd installSwapfile

\$ ./installSwapfile.sh

### dlibのインストール

http://dlib.net/compile.html
\$ git clone https://github.com/davisking/dlib.git
\$ cd dlib
\$ python setup.py install

### Python dlib examples

実演:hogsvm.py 実演:hogStereo.py 実演: video\_facial\_landmarks.py pimagesearch

## Hog特徴量の学習

### 1.画像収集プログラム

\$ pip3 install icrawler https://qiita.com/NakaokaRei/items/8c7e7b1f2c0c7ef8b3a3

### 2.Annotation

delib/tools/imglab readme.txt 参照

3.学習

delib/python\_examples/train\_object\_detector.py

4.推論

delib/python\_examples/



## RasPi4+深層学習

講演時間10分 (株)人工知能ロボット研究所 西村輝一

### 深層学習の活用

- 色認識
- テンプレートマッチング
- Key-pointマッチング
- 機械学習
  - OpenCV2 HOG
  - OpenCV3 HAAR
  - Dlib HOG
- 深層学習
  - 分類/Classification
  - 物体認識
  - セグメンテーション
- Depth Point cloud

## **OpenCV**



### https://opencv.org/

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library.

### https://docs.opencv.org/master/

\* OpenCV - 4.3.0

2020-04-06

Docs	Main modules: core. Core functionality imgproc. Image Processing
H Windows	videoio. Video I/O highgui. High-level GUI
Release Notes	video. Video Analysis calib3d. Camera Calibration and 3D Reconstruction features2d. 2D Features Framework objdetect. Object Detection dnn. Deep Neural Network module ml. Machine Learning flann. Clustering and Search in Multi-Dimensional Spaces photo. Computational Photography stitching. Images stitching gapi. Graph API

### OpenCV4 DNN(Deep Neural Network module)

OpenCV4 DNNは、OpenCVの関数で、Caffe / Tensorflow / Darknet / ONNX / Torch が利用できます。OpenCV3.4.1よりCaffeやTensorFlow等のモデルを読み込みできました が、OpenCV4.0ではONNXフォーマットもサポートされました。

https://docs.opencv.org/master/d6/d0f/group\_\_dnn.html readNetFromCaffe() [1/3] Net cv::dnn::readNetFromCaffe ( const String & prototxt const String & caffeModel = string() Python: retval = cv.dnn.readNetFromCaffe( prototxt[, caffeModel] retval = cv.dnn.readNetFromCaffe( bufferProto[, bufferModel] ) #include <opencv2/dnn/dnn.hpp> Reads a network model stored in Caffe framework's format. Parameters path to the .prototxt file with text description of the network architecture. prototxt caffeModel path to the .caffemodel file with learned network. Returns Net object Examples: samples/dnn/colorization.cpp. \*畳み込みニューラルネットワーク(Convolution Neural Network)とは、AI が画像分析を行うための学習手法の1つです。

## Caffe / Tensorflow / Darknet / ONNX

Caffe(カフェ)とは、オープンソースのディープラーニングライブラリです。画像認識に特化しており、高速処理が可能です。カリフォルニア大学バークレー校の「コンピュータビジョンおよび機械学習に関する研究センター(Berkeley Vision and Learning Center)」が中心となって開発しています。

TensorFlow(テンソルフロー)とは、Googleが開発しオープンソースで公開している、機械学習に用いるための ソフトウェアライブラリです。

**Darknet** は C で書かれた機械学習フレームワークです。darknetを用いて構築されたreal-time object detection systemがYOLOです。

Torch(トーチ)とは、「機械学習ライブラリ」「科学計算フレームワーク」です。GPUを活用する機械学習アルゴリズムを幅広くサポートしています。Facebook が開発している。http://torch.ch/

ONNX(Open Neural Network Exchange)はニューラルネットワークのモデルを定義するためのオープン フォーマットの一つで、様々なDNNフレームワークで作成したモデルをONNXフォーマットで出力することで相互利用す ることができます。

https://github.com/onnx/models

### 画像分類/物体認識/セグメンテーション

### Classification

画像分類するアルゴリズムです。

### **Object detection**

- 1: 物体領域候補の抽出
- 2: 物体領域候補の物体認識
- 3: 検出領域の絞り込み

### **SSD (Single Shot Multibox Detector)**

SSD は深層学習ベースの物体検出アルゴリズム,物体候補領域の生成と分類を同時に学習・実行することが可能です。

### YOLO(You only look once)

YOLO では領域候補の抽出するのではなく直接的に物体検出をしようというものです。

### **Segmentation**

画像の領域を分割するタスクをSegmentation(領域分割)と呼び、Semantic Segmentationは「何が写っているか」で画像領域を分割するタスクのことを指す。



• 分類

- 物体検出
- ・セグメンテーション

おまけ

### **OpenCV AI Kits**

#### What are OpenCV AI Kits?

Now, to be honest, we are not ready to share a whole lot about our AI kits at this point. But because you an OpenCV fan, we will share a few secrets!

The kits consist of a board with an embedded camera(s). The kits will come in a few different configurations. The biggest one will be about the size of a Raspberry Pi. The smallest one will be less than half of it.

The AI board will be able to run Deep Learning models for image classification, object detection, segmentation, human pose estimation, and many more in real time. Some configurations will have a depth camera in addition to a regular RGB camera. We will make it really easy for you to train your own models and bring it to the device with awesome tutorials and support.





## MATLABからサーボを コントロール

講演時間10分 (株)人工知能ロボット研究所 西村輝一

## **MATLABの活用方法**

- MATLAB Simulink
  - ◆ 制御プログラム開発
- MATLAB Script
  - ◆ プロトの開発
  - ◆ 機械学習のツール充実
  - ♦ 簡単に開発可能
- ♦ ROSと連携

#### 二足歩行ロボット協会データベース http://biped-robot.or.jp/dbreports/

二足歩行ロボット協会データベース		会員登録 会員ログイン サイトTOP
TOP ABOUT		
ログインするとすべての機能をご利用いただけます。		
アクセスできません。		
全て ニュース イベント 競技規則	」 技術情報 その他	
第17回ROBO-ONEコンファレンス講演資料	第16回ROBO-ONEコンファレンス講演資料	ROBO-剣参加者ガイド 2018.4.9
2019.6.26 第17回ROBO-ONEコンファレンス議演資料を掲載しま す。	2018.7.6 2018年7月7日開催の第16回ROBO-ONEコンファレンス講 演員料を掲載します。 ご参加される方は事前にダウンロードの上ご出席くださ	第7回ROBO-剣参加者ガイドを掲載します。20180409 第8回ROBO-剣参加者ガイドを掲載します。20180824 第9回ROBO-剣参加者ガイドを掲載します。20190509
技術情報 Attached: 1_RaspberryPiでロボットを動かす.pdf 2_autoの為の画像処理基本プログラム.pdf 3_ROSをは じめよう.pdf 4_予選1位のロボット.pdf 5_Bトーナメ ント優勝ロボ.pdf 6_ランキング_一位のロボット.pdf 著選1位のロボット.pdf を調整したの方の	い。 <b>技術情報</b> Attached: 1_Arduinoでサーボを動かす.pdf 2_画像処 理をはじめよう.pdf 4_ROBO_剣ロボット、速巡.pdf	競技規則 Attached: 参加者ガイド7thROBO_剣.pdf 参加者ガイ ド_8thROBO_剣.pdf 参加者ガイド9thROBO_剣.pdf Comment: 0
12度 11版の日本9 152 C目入版,pdf autoのための画面のE 理_差し替え版,pdf Comment: 0	ンプロロンプロ RCBOONE_adl0ランキング1位_コビス_20180707_最終,pdf Comment: 0	参加者ガイド 2018.2.17

### Simulink連携サーボ駆動プログラム robo-one server

http://biped-

robot.or.jp/upload/dbmaterials/389 bca292de0a969fc91793281ec 5d73780original.pdf

http://biped-

robot.or.jp/upload/dbmaterials/566 bdd990b6575556054f928bc44 061159foriginal.pdf



### ポジション設定コマンド

角度の指定によりサーボを動作させることができます。

TX	1	2	3	
	CMD	POS_H	POS_L	

CMD ポジション設定コマンド POS\_H / POSL サーボの設定舵角

### CMD ポジション設定

MSB	CMD						LSB
7bit	6bit	5bit	4bit	3bit	2bit	1bit	Obit
1	0	0	х	x	х	х	х
ポジション語	定コマンド:	#100xxxxxb	ッ サーボのID 0x00(0) ~0x1F(31)				

( x x x x x はID番号)

### サーボの設定舵角

MSB	POS_H							
7bit	6bit	5bit	4bit	3bit	2bit	1bit	Obit	
0	x	х	x	x	х	х	х	
0固定		設定舵角( 上位7bit)						

MSB	POS_L							
7bit	6bit	5bit	4bit	3bit	2bit	1bit	Obit	
0	x	x	x	x	х	х	x	
0固定	設定舵角(下位7bit)							



## MATLAB script M-fileプログラム

### 実演:M-fileでサーボを動かそう。

```
s = serialport("/dev/ttyUSB0",115200);
s.Parity="even";
sid=0;
t_angle=3500;
while t_angle<11500
  t_angle=t_angle+5
  s_angle=set_angle(s,sid,t_angle)
end
clear s
function s_angle = set_angle(s,sid,t_angle)
  buff(1)=0x80+bitand(sid,0x1f);
  b2=uint16(bitshift(t_angle,-7));
  buff(2)=uint8(bitand(b2,0x007f));
  buff(3)=uint8(bitand(t_angle,0x007f));
  write(s,buff,"uint8");
  data = read(s,3,"uint8");
  s_angle=bitshift(data(2),7)+data(3);
end
```

### USBアダプターのセッティング

\$ sudo modprobe ftdi\_sio \$ sudo su root@\$ echo 165C 0008 > /sys/bus/usbserial/drivers/ftdi\_sio/new\_id root@\$ chmod 666 /dev/ttyUSB0 root@\$ exit \$ ls /dev/ttyU\*



## ご清聴ありがとうございました。