センシングシステム構築実習

・Lobe | Machine Learning Made Easy

■機械学習

・TensorFlow のモデルファイル .pb を .tflite に変換するときの注意点 - 株式会社ライトコード

■

略語	意味
LLM	大規模言語モデル
LVM	大規模視覚モデル

■アノテーション

・アノテーションツールおすすめ 5 選を比較〜ツールを選ぶ 3 つのポイントとは〜 | AI・アノテーションブログ | 株式会社ヒューマンサイエンス

・GitHub - HumanSignal/labelImg: LabelImg is now part of the Label Studio community. The popular image annotation tool created by Tzutalin is no longer actively being developed, but you can check out Label Studio, the open source data labeling tool for images, text, hypertext, audio, video and time-series data.

■ netron

・GitHub - lutzroeder/netron: Visualizer for neural network, deep learning and machine learning models

■従来の画像処理と AI 画像処理の比較

・画像処理の速度向上！最新技術と手法 | Hakky Handbook
・AI カメラと画像処理が変える工場：外観検査と自律ロボット活用
・従来の画像処理と AI 画像処理を併用し、フレキシブルな外観検査を実現 - ヴィスコ・テクノロジーズ株式会社
・画像処理装置〜 AI を使った検査 | ヴィスコ・テクノロジーズ株式会社

物体検出

・物体検出 SSD（Single Shot multibox Detector） #Python - Qiita
・【物体検出】SSD(Single Shot MultiBox Detector) の解説
・物体検出の５つの代表的なアーキテクチャの特徴まとめ | AI 研究所
・SSD: Single Shot MultiBox Detector 高速リアルタイム物体検出デモを Keras で試す #Ubuntu - Qiita

NVIDIA Jetson Xavier NX

・AVerMedia NVIDIA Jetson Xavier NX スタンダードボックス PC(新古品 バルク) / NX211B
・NX211B / NX211B-16G - 株式会社 エルザ ジャパン
・NX211B_NO111B - AVerMedia- AI-based solutions | AVerMedia

OKdo Nano C100

・OKdo Nano C100 開発キット - IoT 本舗 オンラインストア
・Nano C100 の初期設定
・【レビュー】 OKdo Nano C100 開発キット：もうひとひねり欲しい、Jetson Nano B01 の互換キット | がじぇっとりっぷ
・Jetson-Nano 互換機 OKdo Nano C100 の使い方（特に eMMC 起動） | 竹のしんのテック日記
・References for OKdo Nano C100 Developer Kit (a NVIDIA Jetson Nano Developer Kit B01 - The Jetson Nano Computing Module Compatible Board) · GitHub
・LetsOKdo/c100-bootupd: Custom bootloader for OKdo Nano C100 to enable microSD boot

C1000eR　**画像処理**

・OpenCV の実用的で便利なコード集 - カメラ調整編 (Python) - フードテックブログ
・ロジクール Web カメラ Brio C1000s と C1000eR の 1 つの違い。 – カタログクリップ

グレースケール

・カラーカメラとモノクロカメラの違い｜産業用 UVC カメラのすすめ｜株式会社アルゴ

・Python でグレースケール (grayscale) 化 - Qiita
・グレースケール画像のうんちく - Qiita
・OpenCV 画像の二値化 - Qiita

二値化・ラベリング・面積＆重心計算

OpenCV

インストール

```
pip install opencv-python
pip install opencv-contrib-python
```

・画像処理をマスターしよう！ Python で OpenCV を使う方法を紹介！ | TechTeacher Blog
・OpenCV-Python チュートリアル文書のページへようこそ！ — OpenCV-Python Tutorials 1 documentation

■カメラの設定

・【Python × OpenCV】カメラの設定を確認・変更する方法（解像度、明るさなど） - ケイエルブイ

■カメラ

```
while True:
    if cv2.waitKey(1)&0xFF == ord('q'):
        break
```

NumPy

インストール

```
 pip  install  https://download.lfd.uci.edu/pythonlibs/archived/numpy-1.22.4+mkl-cp39-cp39-win_amd64
.whl
```

・AMD 製 CPU の Ryzen で Numpy が遅い問題を解決する方法
・Anaconda の NumPy が高速みたいなので試してみた - Morikatron Engineer Blog
・Numpy を高速化する最も簡単な方法【Python on Windows】| ジコログ
・Anaconda の商用利用有償化に際して : pip で Intel Math Kernel Library (MKL) をサクッと入れる (Windows) - Qiita
    ・conda Numpy のように scikit-learn も高速化する方法 - Qiita

```
pip install https://github.com/cgohlke/numpy-mkl-wheels/releases/download/v2024.1.3/numpy-1.26.3-cp311-cp311-win_amd64.whl
```

・Christoph Gohlke

MKL がリンクされているかを確認

```
python -c "import numpy; numpy.show_config()"
```

・MKL 版 NumPy, SciPy をビルドする方法 (お手軽バージョン) #Python - Qiita

・[NumPy いかたこのたこつぼ]

・「Python は遅い」など、滅びゆく時代の神話 | HackerNoon
・【Numba】NumPy + for 文の高速化 | 西住工房（改）
・Python プログラムが遅い！高速化したい！そんな時は ... #Python - Qiita
・あなたの Python を爆速にする 7 つの方法
・GitHub - numpy/numpy: The fundamental package for scientific computing with Python.

CUDA

・GitHub - cudawarped/opencv-python-cuda-wheels: Automated CI toolchain to produce precompiled opencv-python, opencv-python-headless, opencv-contrib-python and opencv-contrib-python-headless packages.

JupyterLab

インストール

```
pip install --upgrade jupyterlab
pip install --upgrade jupyterlab-language-pack-ja-JP
```

起動

```
jupyter lab
```

または

```
python -m jupyter lab
```

・Jupyter Notebook ではなく JupyterLab を使いましょう | ジコログ
・図解！ Jupyter Lab を徹底解説！ (インストール・使い方・拡張機能) - AI-inter の Python3 入門
・JupyterLab を日本語化する | せぴろぐ
・【Python】Jupyter で「.py」を実行するには？ | ANDoblog

・Windows10 コマンドラインから規定のアプリを変更する方法 – IT Learning
・SetDefaultBrowser – Set the Default Browser per command line or script – the kolbicz blog

```
jupyter lab --generate-config
```

画像処理教材

```
pip3 install opencv-python==4.4.0.46
```

・pip install poencv-python で入れた openCV のバージョンを変更する - Qiita
・Opencv4 を Opencv3 にダウングレードする。 - Kinacon の技術ブログ

```
pip install -U matplotlib
```

```
pip install -U ipywidgets
```

プログラム中の、

```
tuple(dst[1][0])
```

を、

```
(int(dst[1][0][0]),int(dst[1][0][1]))
```

または、

```
tuple(map(int,dst[1][0]))
```

に替える。

・【OpenCV】cv2.putText 関数の使い方【文字を描画する】| 資格マフィア
・【Python】map の正しい使い方 - Qiita
・【OpenCV-Python】findContours による輪郭検出 | イメージングソリューション
・OpenCV - findContours で画像から輪郭を抽出する方法 - pystyle

```
pip install -U tensorflow
```

プログラム中の

```
acc
```

を

```
accuracy
```

に、

```
val_acc
```

を

```
val_accuracy
```

に替える。

■ matplotlib

```
import matplotlib
matplotlib.use('tkAgg')
import matplotlib.pyplot as plt
```

- python - "UserWarning: Matplotlib is currently using agg, which is a non-GUI backend, so cannot show the figure." when plotting figure with pyplot on Pycharm - Stack Overflow
- [SOLVED] Python3 and Matplotlib - no luck - Raspberry Pi Forums
- 【Python】matplotlib で問題が発生したときの解決方法 | シラベルノート

OpenCV GUI

- Widgets — OpenCV tutorial 2019 documentation
- nb.o の日記 : OpenCV Viz の Widget のサンプル（その１）
- GitHub - nngogol/All-GUI-in-OpenCV-python: Examples for ALL GUI elements in OpenCV in python.

Tkinter

- Python で Tkinter を使ってめちゃくちゃ簡単に GUI アプリを作る | だえうホームページ
- tkinter --- Tcl/Tk の Python インターフェース — Python 3.10.6 ドキュメント
- お気楽 Python3/Tkinter 超入門

Python

- Python チュートリアル — Python 3.9.4 ドキュメント
- 深堀り Python|Python で扱うことができる整数の桁数
- 一緒に Python を始めよう - Qiita
- Python の計算機イプシロン - Qiita
- 計算機イプシロンのこと - 再帰の反復 blog
- decimal --- 十進固定及び浮動小数点数の算術演算 — Python 3.9.4 ドキュメント
- iPromenade: ２の平方根を 1 万桁まで求める

- Python のリストと配列と numpy.ndarray の違いと使い分け | note.nkmk.me
- array --- 効率のよい数値アレイ — Python 3.9.4 ドキュメント
- Python のシーケンス型の演算子の使い方【初心者向け】 | TechAcademy マガジン
- 5. データ構造 — Python 3.9.4 ドキュメント
- NumPy 配列の基礎 — 機械学習の Python との出会い

- Python で型を極める【Python 3.9 対応】 - Qiita
- GitHub - python/mypy: Optional static typing for Python
- mypy 0.990 documentation
- 「型」のお気持ち　〜 「型なし」と「動的型」って同じ？違う？ - Qiita

・Python ♪用語集：モジュール、パッケージ、ライブラリの違い | Snow Tree in June
・自作ライブラリを作成してみよう | StartLab –Python 特化のプログラミングスクール –
・Python の　init　.py とは何なのか - Qiita
・Python の datetime でタイムゾーンを扱う
・Python/ 数値入力と文字入力と出力表示 - Wikibooks

・GitHub - mpi4py/mpi4py: Python bindings for MPI

■フィルター

```
cnts = list(filter(lambda x: cv2.contourArea(x) > 1000 , cnts))
```

■ OpenCV

・Python でグレースケール (grayscale) 化 - Qiita
・HSV 色空間 - Wikipedia
・ガウシンアンフィルタ - ノイズの除去｜ MiVLog（ミブログ）
・【画像処理】ガウシアンフィルタの原理・特徴・計算式 | 西住工房
・OpenCV - モルフォロジー演算 (膨張、収縮、オープニング、クロージング) - pystyle
・Python+opencv でウェブカメラの画像からインターホンモニタ部分を抽出・射影変換する - West Gate Laboratory
・OpenCV を利用して動画（カメラ）から動体検知をする方法について | CyberAgent Developers Blog
・Python, OpenCV, NumPy でカラー画像を白黒（グレースケール）に変換 | note.nkmk.me
・描画関数 — opencv 2.2 documentation
・Python, OpenCV で BGR と RGB を変換する cvtColor | note.nkmk.me
・取得画像のコピー - TB-code
・輪郭 : 初めの一歩 — OpenCV-Python Tutorials 1 documentation
・Python, OpenCV で動画を読み込み（ファイル・カメラ映像） | note.nkmk.me
・OpenCV の座標系と画素値【OpenCV】 | BioTech ラボ・ノート

■顔検出 (顔認識)

・【入門者向け解説】openCV 顔検出の仕組と実践 (detectMultiScale) - Qiita
・Python, OpenCV で顔検出と瞳検出（顔認識、瞳認識） | note.nkmk.me
・Haar Cascades を使った顔検出 — OpenCV-Python Tutorials 1 documentation
・顔認識・顔検出とは？ 6 つの顔認識 API の機能・精度・価格を比較！ | Ledge.ai
・[4 種類の顔検出を動かしてみた　Haar+Cascade/ HOG+SVM/ CNN/ MTCNN] - Atom's tech blog
・dlib_compiled/dlib-19.22.99-cp39-cp39-win_amd64.whl at main · shashankx86/dlib_compiled

■色認識

・OpenCV + Python での色認識・色検出 – S-Analysis
・OpenCV - inRange で画像を 2 値化する方法について - pystyle

■形状の認識

・四角形と三角形の中から三角形だけ抽出する (特定形状の抽出) - Python でいろいろやってみる
・領域 (輪郭) の特徴 — OpenCV-Python Tutorials 1 documentation

・python+opencv で画像処理の勉強 8 パターン・図形・特徴の検出とマッチング - Qiita

■カレントディレクトリの取得

```
import os
cwd = os.getcwd()
print(cwd)
```

・Python で現在のディレクトリを取得する - Python でパスの取得・操作 - Python の基本 - Python 入門
・【Python】絶対・相対パス取得・カレントディレクトリのアクセス方法｜ os・path モジュールによるファイル操作入門

■画像サイズの取得

・Python + OpenCV での画像サイズ取得方法 - Qiita

■画像処理 照明補正

・不均一な照度の補正と前景オブジェクトの解析 - MATLAB & Simulink Example - MathWorks 日本
・画像処理のあれこれ 4 輝度補正処理｜伊藤　桂一 (アダコテック) ｜ note
・シェーディング補正 | 表面検査 情報館 | 表面欠陥検査装置 SCANTEC（スキャンテック）| ナガセテクノエンジニアリング株式会社
・画像処理による織物表面検査 画像処理による染めむら検出

■寸法測定

・寸法測定 | 画像処理 .com | キーエンス
・キュウリの曲がり具合を画像処理でみてみる (2) 幅と長さを測る - Emotion Explorer
・Jetson Nano + USB カメラ + OpenCV + Scikit-image でサブピクセル精度の計測系を構築する - Qiita
・カメラ・ソフトウェア L-837 | Zoom ＆ Focus | 【HOZAN】 ホーザン株式会社

■ナンバープレートのフォント

・フォントプロジェクト

■ナンバープレート

・画像処理を用いた自動車のナンバープレート（自動車登録番号表）の解析 - AICam

■メモ

・画像を二値化して輪郭抽出、輪郭の個数を表示する - Python でいろいろやってみる
・関数の定義 : ゼロからの Python 入門講座 - python.jp
・Python でリストとタプルを相互に変換する list(), tuple() | note.nkmk.me
・OpenCV で描画してみよう #Python - Qiita
・【Python・OpenCV】画素値にアクセスする方法 - codevace
・Python+OpenCV で重心を求める
・画像の台形補正をしてみた #Python - Qiita
・Python の文法メモ : 【OpenCV】warpAffine によるアフィン変換
・一定面積以下のオブジェクトを削除する | バイオハッカー - 見習いバイオ研究生の備忘録
・OpenCV 画像の一部を切り出す : 座標指定・スライシング #Python - Qiita

・OpenCV - cv2.HoughCircles で円を検出する方法 - pystyle
・OpenCV の findContours 関数を使った画像の輪郭検出　画像処理｜ OpenCV オープンソースのすすめ｜株式会社アルゴ
・HSV 色空間での赤色・緑色の検出 | TomoSoft
・OpenCV - inRange で画像を 2 値化する方法について - pystyle
・Python と OpenCV を使用した画像上の点の描画 – qython.net
・Python, OpenCV で図形描画（線、長方形、円、矢印、文字など） | note.nkmk.me

■台形補正

・PerspectiveImageCorrection download | SourceForge.net
・pixFront
・pixFront の詳細情報 : Vector ソフトを探す！
・OpenCV による台形補正・射影変換を解説【Python】 | ジコログ

画像処理

■階調変換

■ヒストグラム

■フィルタリング

■幾何学変換

■二値化処理

画像認識

■テンプレートマッチング

・OpenCV – matchTemplate でテンプレートマッチングを行う方法 | pystyle

■特徴抽出

■物体追跡

・`ultralytics 8.0.232` OpenCV 4.9.0 readonly np.array fix · Pull Request #7256 · Ultralytics/ultralytics

・OpenCV にある追跡関係を調査中 # 画像認識 - Qiita
・Python+OpenCV で Web カメラの画像中のものをトラッキングする話 - EnsekiTT Blog
・Python で物体認識 AI の YOLOv8 を試してみた！ 〜 応用編 〜 # 画像認識 - Qiita
・OpenCV の物体追跡 – S-Analysis

■物体検出

・【AI】YOLOv8 を python openCV でちゃちゃっと物体検出してみる #Python - Qiita
・【やってみた】YOLOv8 の機能試す＆ Web カメラでリアルタイム推論
・Python でやってみた（画像処理編３）：物体検出 (ultralytics ／ YOLOv8) ｜ KIYO
・YOLOv8 で PC カメラからリアルタイムにマスク付けてる・付けていないを判定する #AI - Qiita

```
from ultralytics.yolo.data.augment import LetterBox
from ultralytics.yolo.utils.plotting import Annotator, colors
from ultralytics.yolo.utils import ops
```

を、

```
from ultralytics.data.augment import LetterBox
from ultralytics.utils.plotting import Annotator, colors
from ultralytics.utils import ops
```

に変更する。

魚眼

- OpenCV ＆ Python で、簡単に魚眼レンズの歪み補正（Calibration）を行う方法 | スマートホーム× DIY
- 魚眼レンズの補正 #Python - Qiita
- 【Windows】【Python】 OpenCV で魚眼レンズのカメラキャリブレーション #Python - Qiita
- DIYer Python で魚眼レンズの歪み補正
- GitHub - Kazuhito00/OpenCV-CameraCalibration-Example: OpenCV を用いたカメラキャリブレーションのサンプルです。2021/06/21 時点で Python 実装のある 3 種類 (通常カメラ向け、魚眼レンズ向け (fisheye モジュール)、全方位カメラ向け (omnidir モジュール)) について用意しています。
- OpenCV の魚眼カメラの表現を図解する - いわゆる備忘録
- 魚眼カメラのキャリブレーション　〜　アルゴリズム - Daily Tech Blog
- OpenCV で魚眼レンズのキャリブレーション | tetro

YOLO

- Real Time Inference on Raspberry Pi 4 (30 fps!) — PyTorch Tutorials 2.1.1+cu121 documentation
- Deep learning examples on Raspberry 32/64 OS - Q-engineering

- シウマイ弁当と君 ... ラズパイと Yolov5 | himanossan(肥満おっさん) のブログ
- raspberry pi4 YOLOv5, Segmentation fault - Raspberry Pi Forums
- Error: Yolov5 Openvino model run on Raspberry Pi 4 with NCS2 - Intel Community

- Raspberry Pi 4 で YOLOv5 v7.0 を動かす #RaspberryPi4 - Qiita
- RaspyerryPi4 で Yolov5 を動かす #Python3 - Qiita
- Raspberry Pi 4 で Docker コンテナから YOLOv5 v7.0 のサンプルアプリケーションを実行する方法 (2023 年 6 月時点) ｜ echo
- Raspberry Pi でやってみた３：YOLOv5 を用いたリアルタイム物体検出｜ KIYO
- GitHub - Qengineering/YoloV5-ncnn-Raspberry-Pi-4: YoloV5 for a bare Raspberry Pi 4

- ぬいぐるみを検出するモデルを YOLOv5 で作成し、ONNX 形式に変換して RaspberryPI で使用してみました | DevelopersIO
- RaspberryPi3 + yolo 5 物体検知（Python3.7.3 → 3.9.18 へアップデートも）→ numpy でエラー格闘 → OS を buster から bullseye へ - min117 の日記
- YOLOv5 benchmark on Raspberry Pi 4B (Arm Cortex A-72) | Download Scientific Diagram
- Deploy YOLOv5 Object Detection Models to Raspberry Pi

- Raspberry Pi - Ultralytics YOLOv8 Docs

- labelme+YOLOX で自作データセットの学習！

OpenPose

- Raspberry Pi でカメラ映像の骨格検出をする方法 Part 2 | Murasan Lab
- Raspberry Pi でカメラ映像の骨格検出をする方法 Part 1 | Murasan Lab
- GitHub - ferdavid1/PiPose--under-dev: Use Raspberry Pi camera and OpenPose library to track pose in real-time
- openpose · GitHub Topics · GitHub

- （ラズパイで）USB カメラの映像を Python の OpenCV で高速に表示する #Python - Qiita
- 【Raspberry Pi】GPU メモリーを拡張して描画速度をあげてみた | Murasan Lab
- 最新研究 Raspberry Pi 4 の GPU プログラミング
- Raspberry Pi の VideoCore GPU を用いた Deep Learning ...
- [検出レート約 30FPS] RaspberryPi3 Model B(プラスなし) で TX2 " より少し遅い " MobilenetSSD の物体検出レートを獲得しつつ MultiModel (VOC+WIDER FACE) に対応する #Python - Qiita
- [150 FPS ++] Coral Edge TPU Accelerator を 3 本突き刺して並列推論し超速の Posenet 性能を獲得する ー無駄な高性能の極みへー #Python - Qiita