

# 自作セキュリティカメラ



# アジェンダ

- ①自己紹介（バックグラウンド）
- ②このプロダクトの概要
- ③プロダクト製作の経緯・背景
- ④プロダクト自体の具体的な説明
- ⑤将来への展望

# 自己紹介

<2020・1月 機械学習コース>

浅田 竜太郎 30歳

家族:妻、子

福岡県出身 東京在住

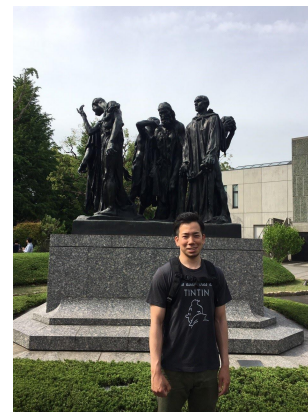
経歴 久留米工業高等専門学校 機械工学科 卒業  
(過去) <車とかバイクなどの構造に興味があった>



造船所に就職 現場の施工管理(横浜)8年ぐらい  
<大きい建造物に魅了を感じた>  
造船不況、、、、



エアコンメーカー アフターサービス 1年ぐらい



現状

通学:機械学習について勉強中

家庭:主に家事育児

# 一言で言うと : セキュリティーカメラ

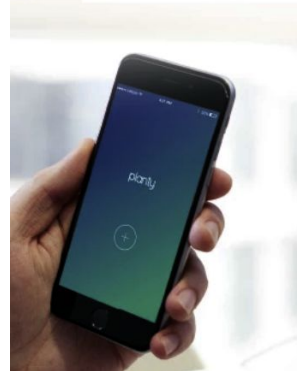
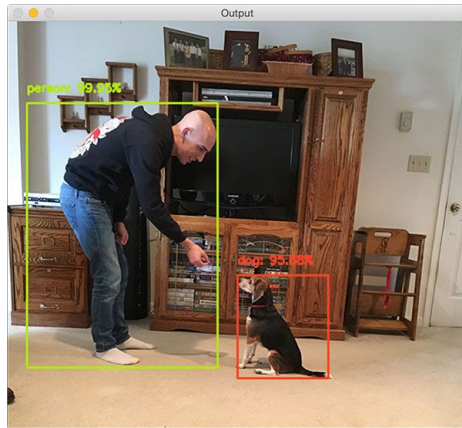
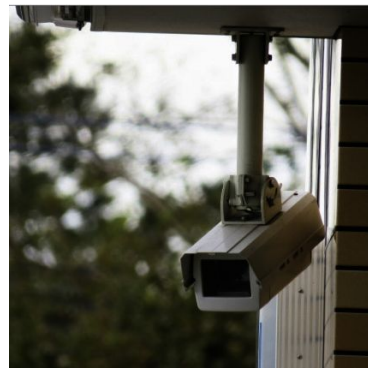
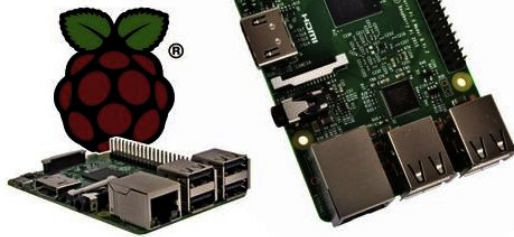
玄関にカメラ設置

→人を認識すると写真を撮る。

(機械学習)

→スマホに知らせる。

## Raspberry Pi 3



これにした理由

1、現実世界での需要

2、身近な問題

3、学んだことの理解

# 1、現実世界での需要

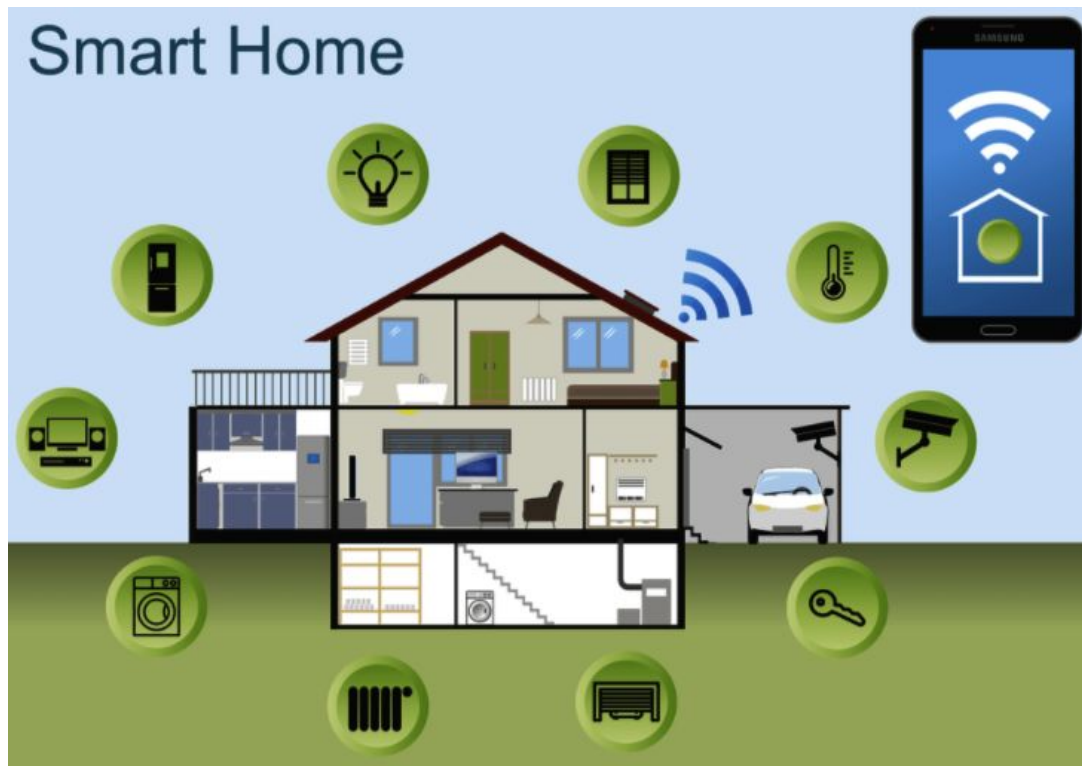


Figure 1. MobileNet models can be applied to various recognition tasks for efficient on device intelligence.

## 2、身近な問題

### スマートホームとは

IoT(もののインターネット)やAIなどの技術を駆使して、住む人にとってより安全・安心で快適な暮らしを実現する住宅のこと。



ささやかな困りごと

家に泥棒入ってないか、

家族は家にいるか、

カギ閉め忘れてないか





### 3、学んだことの理解

学校で学んだことを応用して実際の物にしたい。

百聞は一見に如かず 百見は一考に如かず 百考は一行に如かず 百行は一果(効)に如かず 百果(効)は一幸に如かず 百幸は一皇に如かず

聞くだけでなく、実際に見てみないとわからない 見るだけでなく、考えないと意味がない 考えるだけでなく、行動すべきである 行動するだけでなく、成果を出さなければならない 成果をあげるだけでなく、それが幸せや喜びにつながらなければならない 自分だけでなく、みんなの幸せを考えることが大事

# その他 制約

製作期間:5日間 → 情報収集3日

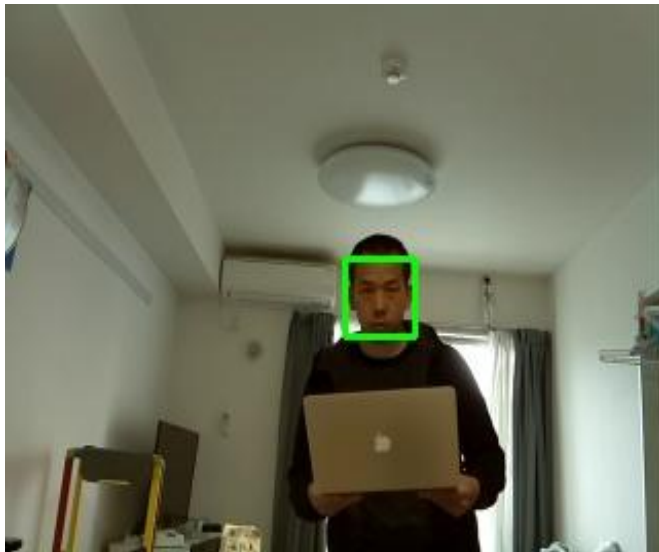
実験&修正2日

費用:一万以内 → ラズパイとカメラのみ

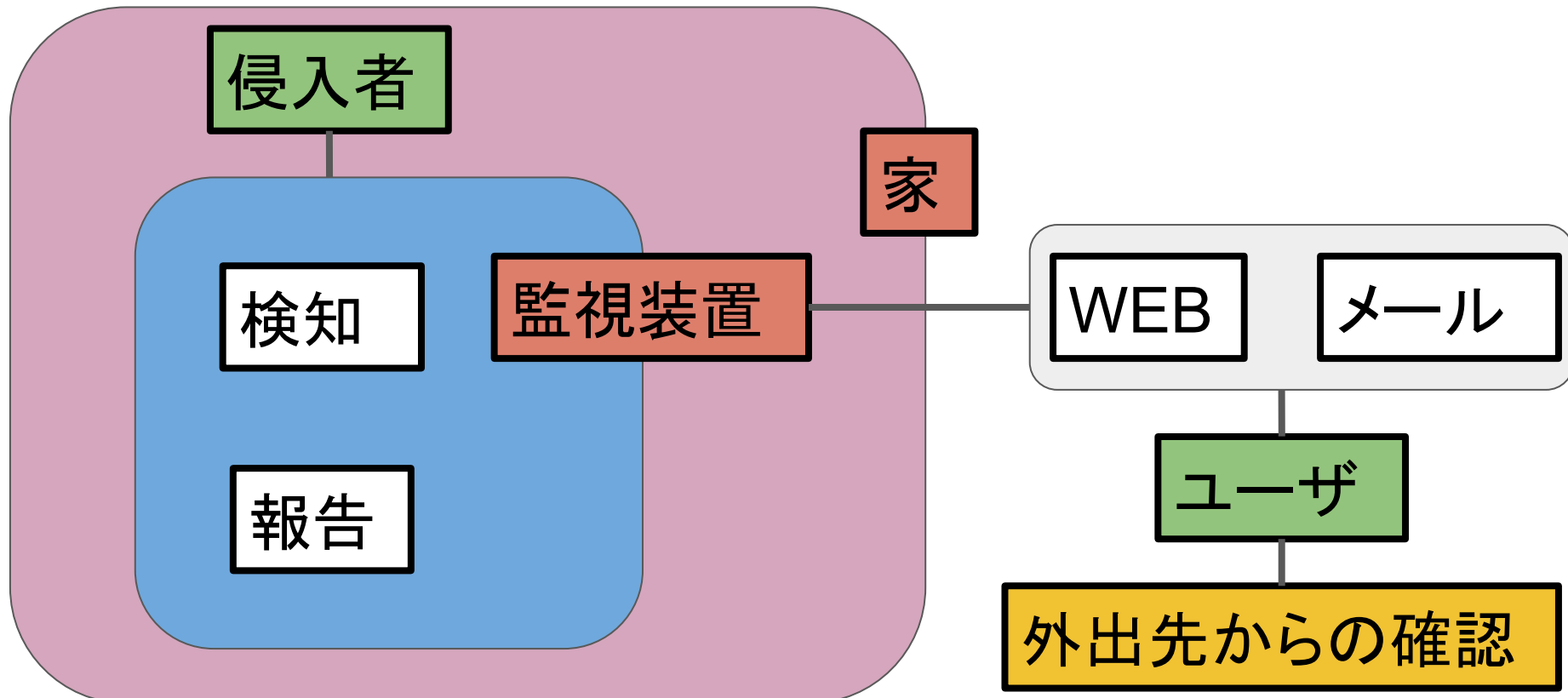
使いたい技術:機械学習を使った技術

→ 物体検出(画像処理)

在るもの使うベース



# システム構成



# 使ったコード

#スマートセキュリティカメラ

<https://github.com/HackerShackOfficial/Smart-Security-Camera>



A diagram showing a light purple rounded rectangle containing three white boxes. At the top center is a box labeled 'main.py(分類)'. Below it, on the left, is a box labeled 'mail.py' with a blue border. On the right is a box labeled 'camera.py' with a yellow border.

main.py(分類)

mail.py

camera.py

# 使用したモデルの特徴

Opencv: (初心者でも使いやすいと言われている)

**OpenCV**(オープンシーヴィ、英語: Open Source Computer Vision Library)とは **インテル**が開発・公開した **オープンソース**の **コンピュータビジョン** 向け **ライブラリ**

以下の分類モデルを使用(顔、体、上半身用など種類を選べる)

```
object_classifier =
```

```
cv2.CascadeClassifier("upperbody_recognition_model.xml")
```

Haar\_features と呼ばれる 特徴量を使用

# Haar\_features

明暗(明るさの移り変わり)具合を使った特徴量

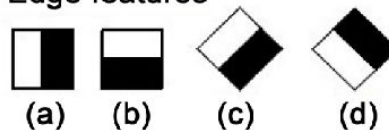
以下のようなバッチを使って畳み込み  
バッチ白と黒領域の合計の差をだす。

参考ビデオ: <https://vimeo.com/12774628>

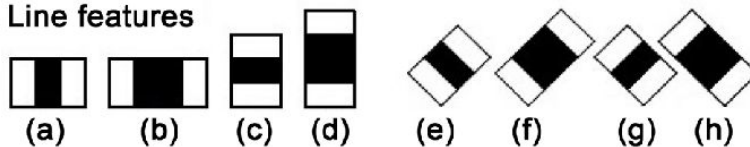
グレースケールで処理  
=高速(ニューラルに比べて)

しかし、光の加減  
にかなり作用される

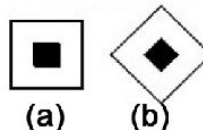
## 1. Edge features



## 2. Line features



## 3. Center-surround features



# 専用ケース

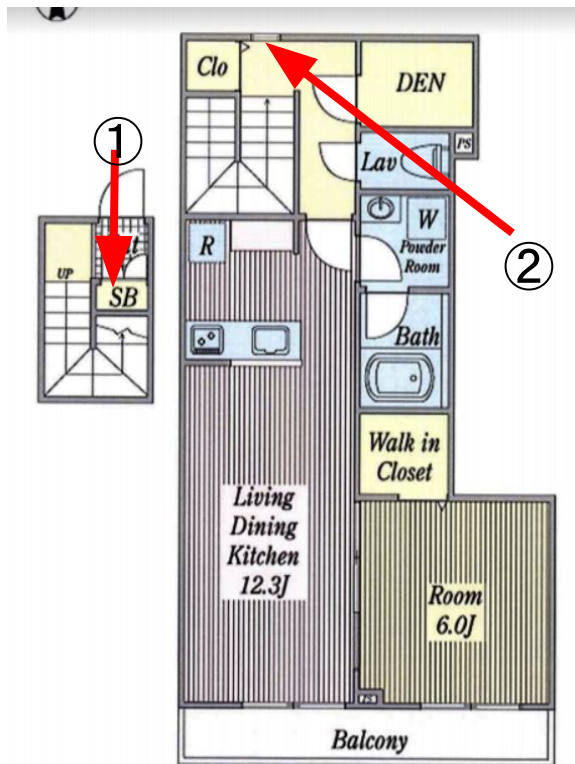


ローコスト





## 設置位置 2箇所で実験



## 玄関(門番型)



## 階段上（家のあかりが使える）



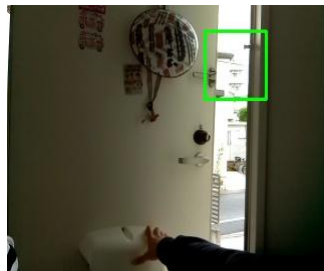
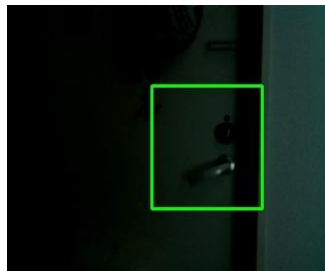
# 映像放映：実験風景

1\_撮影一回目

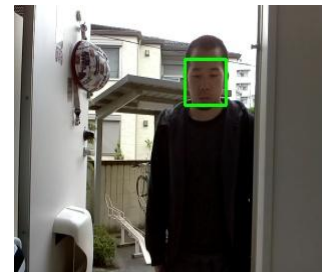
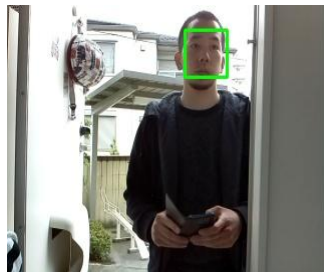
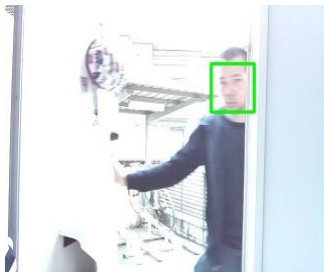
2\_カメラ向き修正後

3\_mailの確認

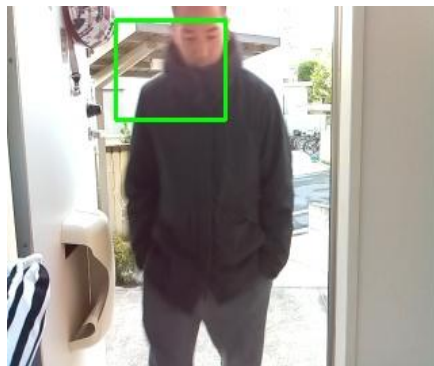
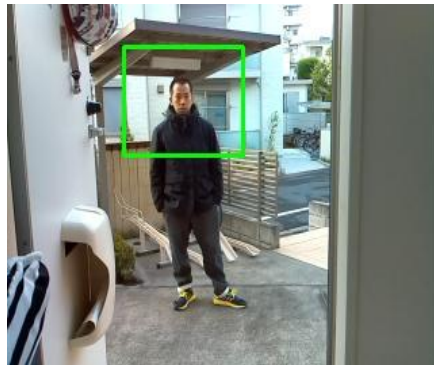
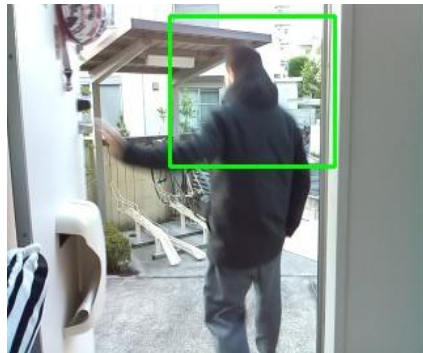
# 顔検出モデル



結構認識しにくい



# 上半身検出モデル



すぐ認識

たまに違うところも、、、

わかったこと

光の変化が激しい

向いてない

# 映像放映：実験風景

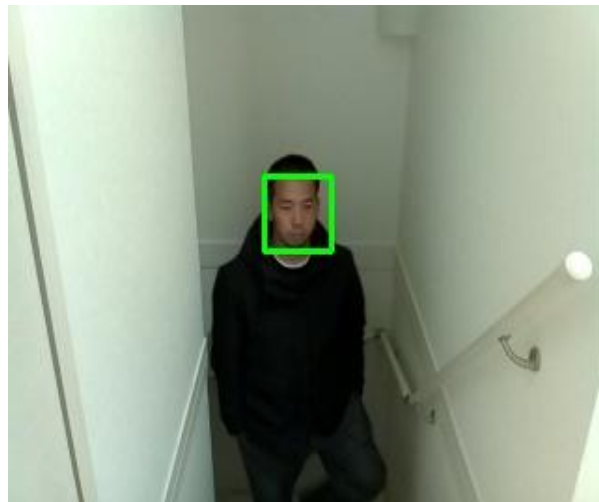
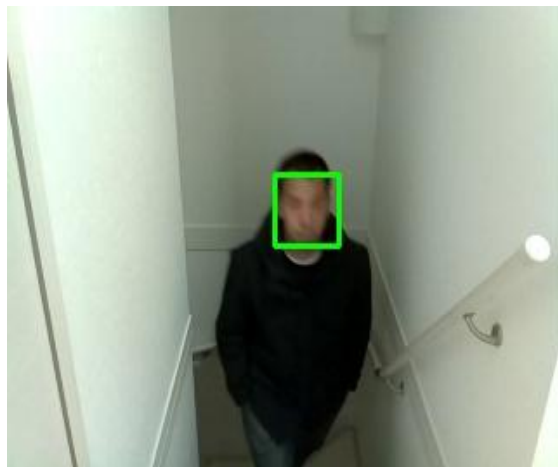
4\_二階に設置

5\_カオナシ

# 顔検出モデル

さっきより

だいぶいい



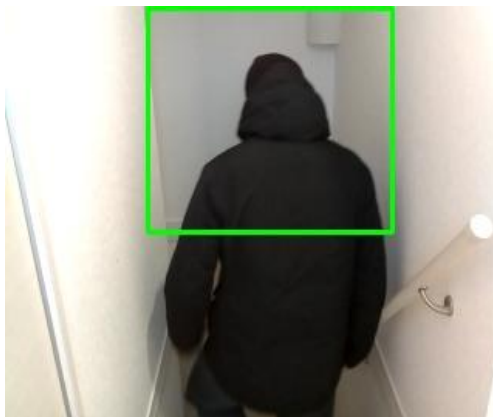


カオナシの場合は？



# 上半身検出モデル

カオナシ  
でも  
OK



# 結論

やりたいこと: 人かどうかを判断して報告する。

比較対象: Opencvの学習済みモデルの比較(全身、上半身、顔)  
設置場所の比較(玄関、階段上)

結論:

モデルは上半身用がベスト。

設置場所は階段上が最適。

使えそう。

# 懸念事項

## 階段のあかり(電気代)

電気をセンサー式にするか

暗くても検知可能か

## できれば分類したい。(家族か 他人か)

顔を学習して使えるか(ラズパイでニューラルネットを使えるか)

# 今後

Tensor\_flow\_liteが早いらしい。

COCO SSD MobileNet v1 model.と言うので試したい。

[https://www.tensorflow.org/lite/models/object\\_detection/overview](https://www.tensorflow.org/lite/models/object_detection/overview)

時間があれば実演