

Raspberry Pi を使用した防犯カメラの製作

遠藤 将仁 野瀬 聖太
平井 健大 山口 皓平

1. 研究概要

(1) 研究の目的

私たちは、近年「闇バイト」などによる高齢者を狙った犯罪や空き巣被害が増加している日本において、社会の役に立つものを開発したいと考えた。そこで、3年間の学習で身についた知識を生かし、犯罪防止に役立つ技術は何かについて話し合いを重ねた結果、AIを用いた監視カメラシステムを作成することに決定した。

(2) 年間計画

表 1 年間計画

4月	研究内容の決定
6月～8月	人物検知システムの作成
9月～10月	Web サーバの作成
11月～1月	双方向通信の作成

2. 研究の具体的な内容

(1) 使用機器

- Raspberry pi 5 (サーバ機)
人物検知機能、Web サーバ、双方向通信機能の実装のために使用した。(図 1)



図 1 Raspberry pi 5

- Raspberry pi 4 (クライアント機)
人物検知機能、双方向通信機能の実装のために使用した。(図 2)



図 2 Raspberry pi 4

- Web カメラ
画像の撮影、人物検知のための映像取得のために使用した。(図 3)



図 3 Web カメラ

- 3D プリンタ
カメラの機体の 3D データの印刷に使用した。(図 4)



図4 3Dプリンタ

(2) 開発環境

- Geany

各種プログラムを記述するのに使用した。(図5)

```

Untitled.py - (home) - Geany
ファイル[?] 新規[?] 検索[?] 表示[?] 文書[?] プロジェクト[?] ビルド[?] ツール[?] ヘルプ[?]
新規[?] 閉じる[?] 保存[?] すべて保存[?] 元に戻す[?] 戻る[?] 戻し[?] 削除[?] 編集[?] コンパイル[?] ビルド[?] 実行[?] 他の選択[?]
検索[?] シンボル[?] unitled.py*
ソース[?] 3.1
4 # unitled.py
5 #
6 # Copyright 2020 user <user@DESKTOP-AQARBD5>
7 #
8 # This program is free software; you can redistribute it and/or modify
9 # it under the terms of the GNU General Public License as published by
10 # the Free Software Foundation; either version 2 of the License, or
11 # (at your option) any later version.
12 #
13 # This program is distributed in the hope that it will be useful,
14 # but WITHOUT ANY WARRANTY; without even the implied warranty of
15 # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
16 # GNU General Public License for more details.
17 #
18 # You should have received a copy of the GNU General Public License
19 # along with this program; if not, write to the Free Software
20 # Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston,
21 # MA 02110-1301, USA.
22 #
23 #
24 #
25 #
26 def main(args):
27     return 0
28
29 if __name__ == '__main__':
30     import sys
31     sys.exit(main(sys.argv))
32

```

図5 Geany

- 123Design

カメラの機体の3Dデータの作成に使用した。(図6)

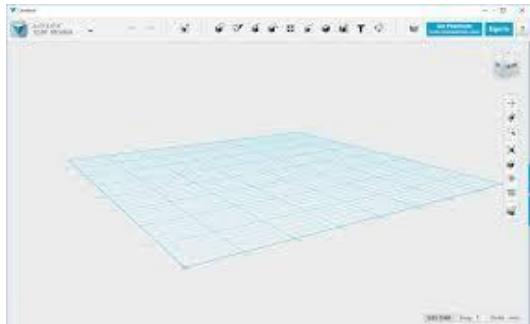


図6 123Design

(3) システムの設計

システムは監視カメラ・双方向通信システム、クライアント側(図7)とサーバ側(図8)、画像閲覧システム(図9)の3つに分けて設計した。

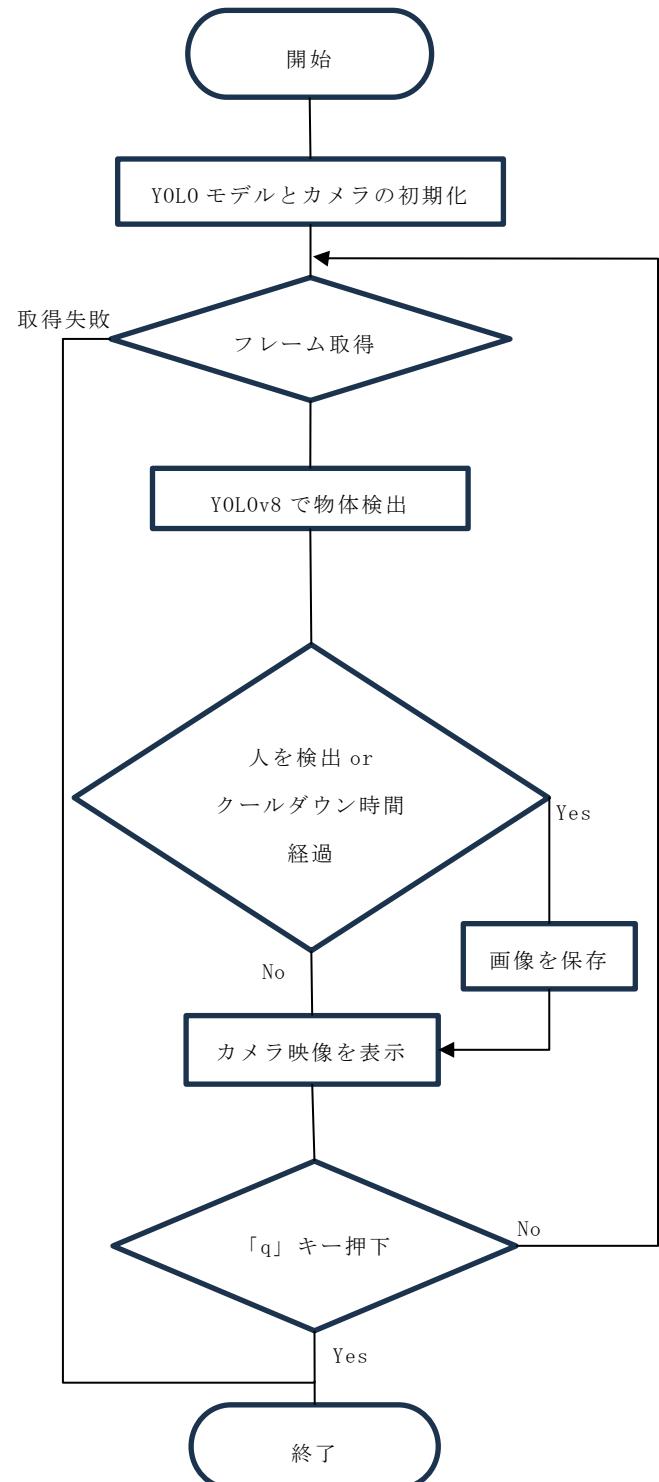


図7 監視カメラ・双方向通信システム
クライアント側

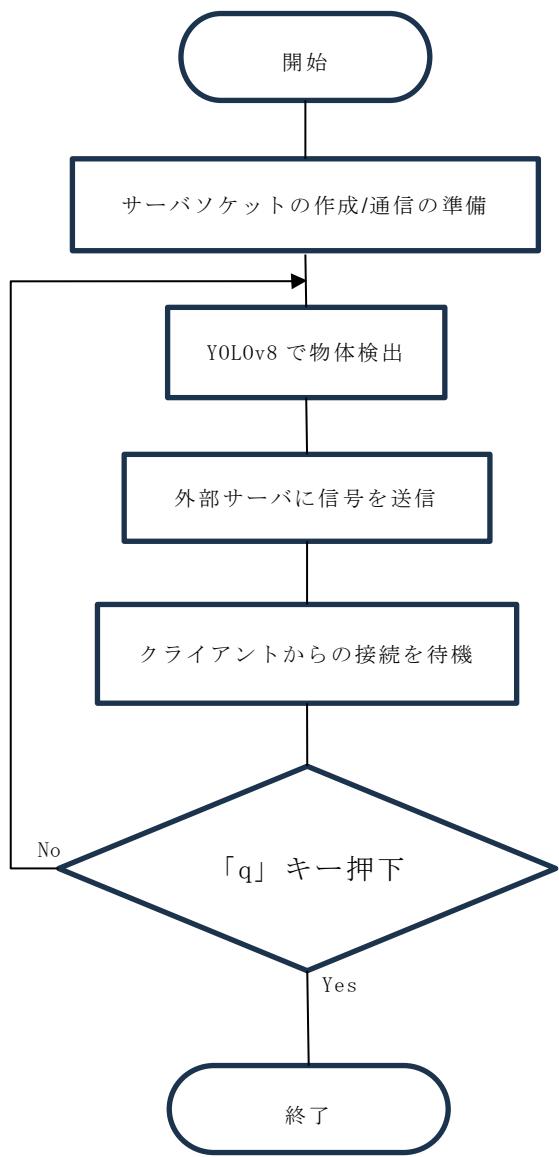


図8 監視カメラ・双方向通信システム
サーバ側

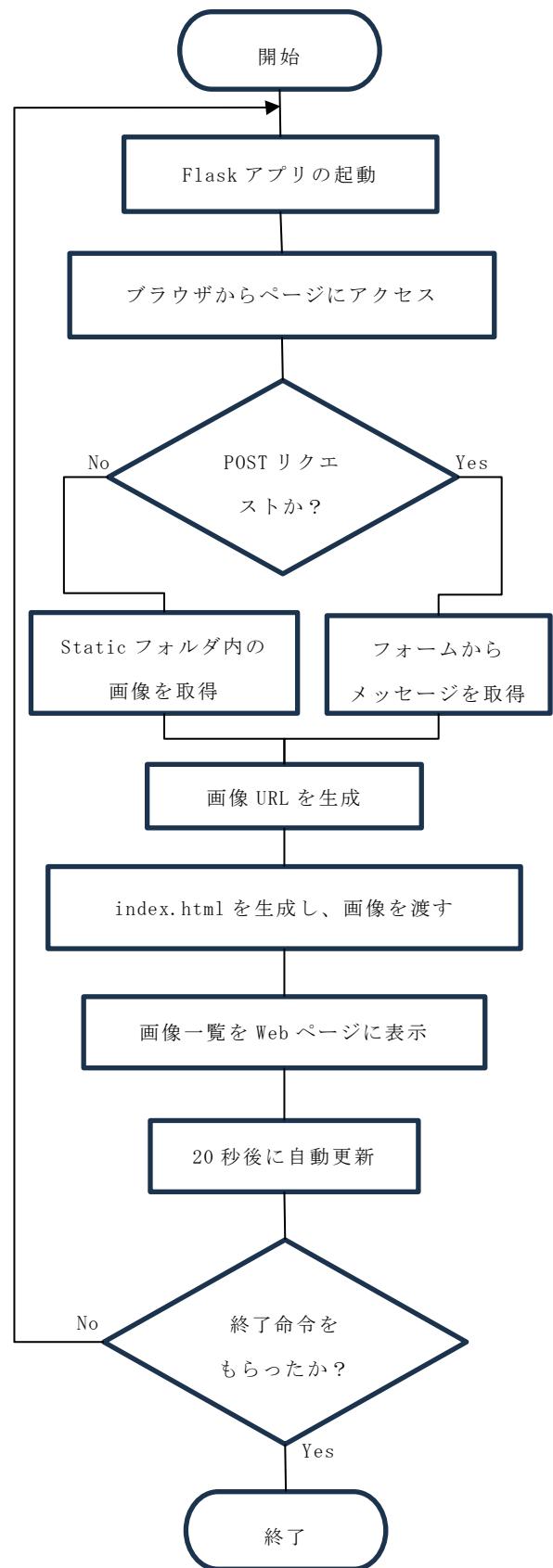


図9 画像閲覧システム

(4) 監視カメラ・双方向通信システムの実際の動作

人物が検知されると画像が撮影され、名前を付けられ保存される。(図 10)

```
保存しました: static/person_20251216_135708.jpg
^CERROR
保存しました: static/person_20251216_135728.jpg
^CERROR
保存しました: static/person_20251216_135748.jpg
```

図 10 保存通知

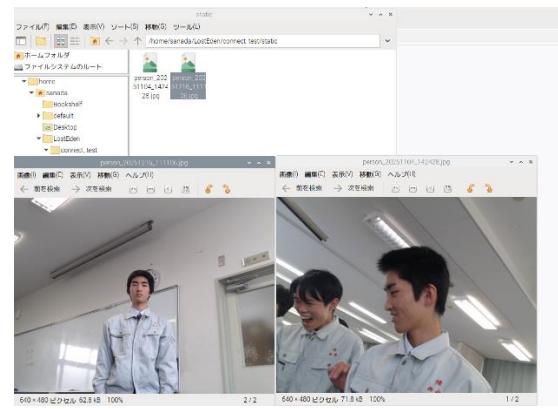


図 12 static フォルダ内

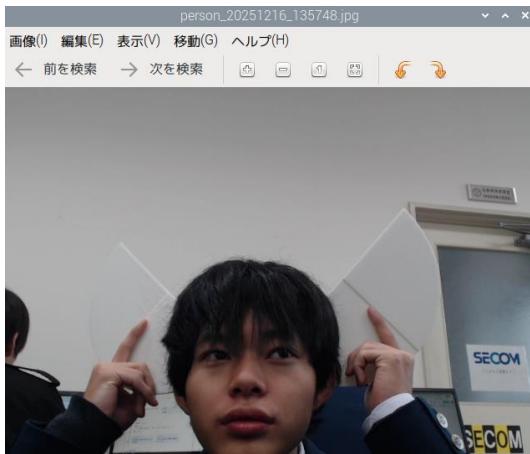


図 11 実際に保存された画像

(5) 画像閲覧システムの実際の動作

下記に示すプログラムにより、監視カメラシステムによって撮影された画像が保存されている static フォルダ内の画像（図 12）が、Web ブラウザから閲覧できるようにした。（図 13）

```
img_urls = [url_for('static', filename=f)
            for f in img_files]
```

画像の URL 生成プログラム

```
{% for img in img_urls %}
    
{% endfor %}
```

画像表示プログラム



図 13 Web ブラウザ上の画面

また、常に最新の画像を表示させるために、下記のプログラムを実装した。

```
<script>
    setTimeout(function(){
        location.reload();
    }, 20000);
</script>
```

20 秒ごとにリロードするプログラム

(6) 外装の作成

123design を用いて外装を設計し、そのデータから 3D プリンタで出力した。Raspberry pi4 が収まること。カメラを固定できることにする。この 2 つのことを行なうまで再設計を繰り返した。

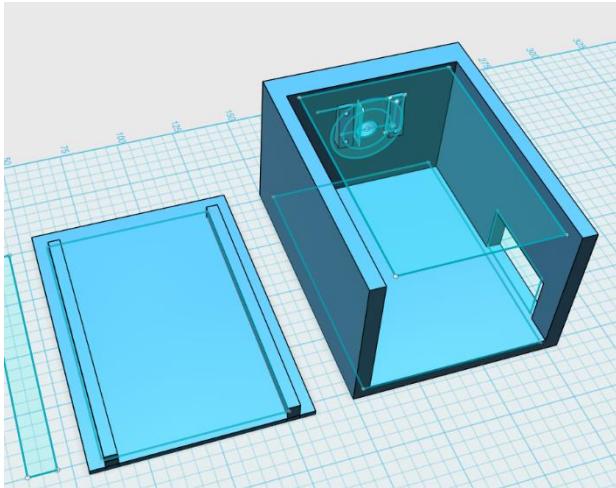


図 14 作成した 3D データ

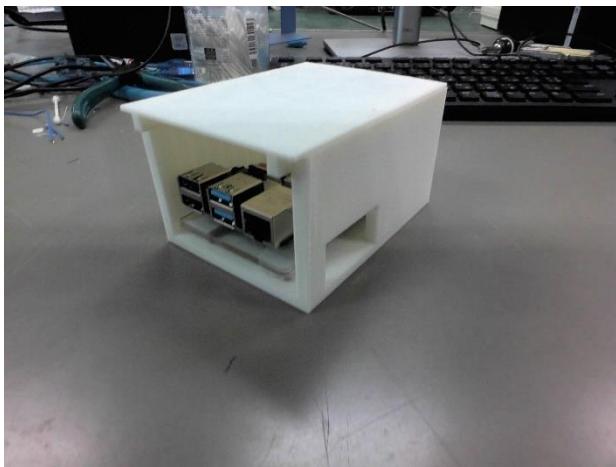


図 15 実際に作成したカメラの機体

3. 研究のまとめ

今回の課題研究を通して、Raspberry Pi の通信の仕組みや python でのプログラム、3DCAD での 3D データ作成などの IoT 機器の制作に関する幅広い知識を学ぶことができた。また、実際に研究を行ってみて日常で使われている小さな IoT 機器は様々な技術が複雑に組み合わせられて作られていることに気がつくことができた。

一方で、開発の初期段階で行った年間計画や役割分担の見通しが甘く、思っていた通りに開発を行えないことが多かった。そのため、研究目標であった社会の役に立つものを開発したいという目標に対して、利便性が高いものを作成することができず、目標を完全に達

成することができなかった。しかし、今回 IoT 機器の開発を行った経験を活かして、進路先では見通しの立った計画を建てられる人間になりたいと思った。

今後、機会があれば、画像閲覧アプリの開発や、人物検知 AI の作成など、今回の課題研究で実装することができなかった機能を追加したり、他の IoT 機器を開発してみたいと思った。

参考文献

参考 URL

- <https://jp.rs-online.com/web/>
- <https://stellasympho.free.makeshop.jp/>
- <https://www.logicool.co.jp/ja-jp>
- <https://my-best.com/>
- <https://www.kkaneko.jp/index.html>
- <http://easylabo.com/>
- <https://ja.wikipedia.org/wiki/%E3%83%A1%E3%82%A4%E3%83%B3%E3%83%9A%E3%83%BC%E3%82%B8>
- <https://www.raspberrypi.com/>
- https://www.realvnc.com/en/connect/download/viewer/?lai_vid=vgJLzMlkEfD6E&lai_sr=10-14&lai_sl=1&lai_p=1&lai_na=921110
- ラズパイに OpenCV をインストールして、画像・カメラ表示、文字・図形描画
<https://monomonotech.jp/kurage/raspberry/opencv.html>

- ・ラズパイ上の Python+Flask ローカル Web サーバにネットから ngrok 経由でアクセス
<https://zenn.dev/karaage0703/articles/0267c64d5d857d>

- ・【ラズベリーパイ】監視カメラの作り方 |
Python でカメラモジュール V2 を自在に操作
作

https://sozorablog.com/camera_shooting

- ・ラズベリーパイで WEB サーバーを構築しよう ! <Apache 編>
<https://www.fabshop.jp/webserver-apache/>

- ・1 2 3 D Design で 3 D のモデリングをしてみよう
<https://fabkura.gitbooks.io/fab-docs/content/article1-2.html>

- ・IT エンジニア もりしーの BLOG
<https://kotamorishita.com/>