

画像認識による個人判別と自動検温 IoT システムの開発

芝田 浩*

Development of an IoT system for automatic body temperature measurement with personal identification using image recognition

Hiroshi Shibata*

要旨: 新型コロナウイルス感染拡大防止の観点から、組織内での日々の健康チェックの徹底が求められている。しかし、計測するのが手間であることや、組織で管理する必要があることなどの課題がある。本研究では、カメラの画像から自動的に顔の検出と個人を判別すると共に、その体温の記録と組織内でのデータを一元管理する IoT システムを開発した。これにより、体温を自動的に記録することができると共に、組織内での管理工数についても低減することができる。

キーワード: AI, IoT, Python, Raspberry Pi, 個人判別, 体温計測

1. はじめに

新型コロナウイルスやインフルエンザウイルスへの感染抑制と拡大防止の観点から、個人の体調管理の徹底が重要視されるようになってきている。特に新型コロナウイルスの感染拡大機においては、組織としての把握が必須とされていた。

これらの感染対策としては、検温による体調管理、手洗い、咳エチケットなどの励行が挙げられる。それら対策の中でも、検温については、感染して体温が上がっているのに普通の行動をとってしまうと、周囲への感染拡大につながる恐れがある。そのため、有効な対策として、体温測定を徹底し健康を管理することが求められている。しかし、組織内で徹底して体温を管理するためには、報告する方も報告される方も関連した作業が大変であり、その作業を軽減する対策が求められている。ワクチン接種の普及と、5類への移行に伴い、新型コロナウイルスやインフルエンザウイルスへの感染についての社会的な危機意識が薄まりつつあるが、いつ感染が拡大するかわからない状況にある。

一方、AI (Artificial Intelligence) と IoT (Internet of Thing) については、多くの情報システムや情報機器への適用が進みつつあり、その機能の社会的なニーズや重要性が増している。これらは、様々な社会課題を解決し、これらの社会現

のために、科学技術やイノベーション等の研究開発・社会への実装と活用が期待されている。

2. 目的と解決目標

本研究では、AI と IoT の技術を活用することにより、人間がカメラの前に立つだけで、個人を判別し体温を自動計測し、その結果をデータベースで管理する IoT システムの開発を目的とする。本システムで計測されたデータは、Web ブラウザでアクセスすることにより、個人のユーザーでは日々の状態をグラフで可視化することができ、組織を管理するユーザーでは、組織内の体温計測状況を一覧で確認できるようにした。これにより、体調管理の手間を軽減させることを目指した。¹⁾

3. システム概要

3.1 システムの全体像

課題解決目標を達成するためのシステムの全体イメージを図 1 に示す。本システムは、カメラで検出した顔に対して、AI で人物を自動で判別する

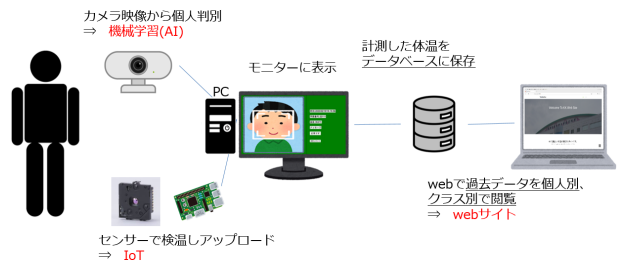


図 1 システムイメージ

* 山形県立産業技術短期大学校庄内校
〒998-0102 山形県酒田市京田 3 丁目 57 番 4 号

* Shonai College of Industry & Technology
3-57-4, Kyoden, Sakata, Yamagata, 998-0102, Japan

と同時に、温度センサーで検温する。さらに、それら取得したデータを DB で管理すると共に、Web ブラウザでのデータ閲覧を可能にする。システムの全体イメージを図 1 に示す。全体を大きく 3 つのサブシステムに分けて実現した (図 2)。

3.2 体温計測サブシステム

温度センサーを搭載した RaspberryPi 端末を使用して、IoT により体温を計測し送信する。温度センサーとしては、放射温度センサーを利用し、計測範囲の最も高い部分の温度を無線通信で個人判別サブシステムへ送信する。

3.3 個人判別サブシステム

AI を利用することにより、Web カメラからの画像から人物を自動で認識すると共に、その個人を登録されている人物かを判別する。本サブシステムでは、個人の登録機能と、個人判別機能および、計測したデータの送信機能を有する。

判別する人物のデータについては、個人の登録機能を、事前に実行し AI に学習させておく必要がある。登録されている個人と判定されない場合は、ゲストユーザーとして判定される。

個人判別の完了した後で、計測サブシステムからの検温データをもとにして検温結果と個人判別結果をモニタに表示すると共に、体温が高い場合は注意喚起のメッセージを表示する。さらに、これらの計測結果を Web サブシステムへ送信する。

3.4 Web サブシステム

IoT を用いて取得した個人の体温データに対する、DB での管理機能と閲覧機能を有する。過去データを個人、組織別に許可されたユーザーのみ閲覧できるように実現した。



図 3 温度センサーによる温度測定

4. 体温計測とデータの送信

4.1 赤外線放射温度センサーの適用

温度センサーとしては、赤外線放射温度センサーの Lepton3.5 を利用した。このセンサーは長波長赤外線を用いて測定する。本センサーは、インターフェースボードを使用して、USB 接続により RaspberryPi にデータを送る。センサー部分は、Web カメラと同じような位置に設置し、計測範囲内の一番高い温度の値を計測するようにした。センサーによる温度測定結果におけるサンプル画面を図 3 に示す。赤い字で示してあるのが、計測結果であり、対象範囲で一番高い温度を表示している。

4.2 IoT で取得したデータの送信

RaspberryPi では、温度センサーで計測したデータを個人判別サブシステムに無線で送信する。OS は、RaspberryPi OS で動作し、ソフトウェアは、Python を使用しプログラミングした。データを送信する無線通信では、MQTT(Message Queue Telemetry Transport)プロトコルを使用し、通信コストをできる限り低減するようにした。MQTT とは、パブリッシュ・サブスクライブ型でデータ伝送するプロトコルであり、軽量で効率のよいデータ伝送ができるという特徴がある。そのため、デバイスやマシンを接続する IoT システムで使われることが多い。

5. AI による個人判別

5.1 個人判別の処理手順

本システムでは、リアルタイムに顔検出と個人を判別するために、AI による画像処理技術を適用した。方法としては、PC に接続した Web カメラ

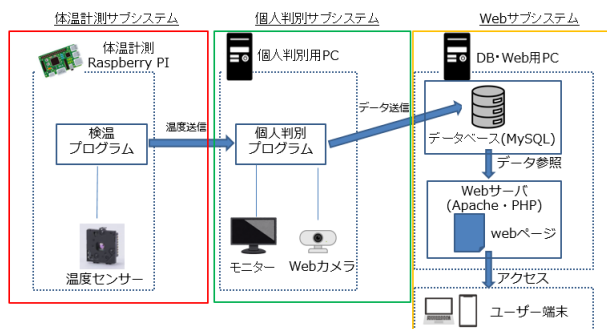


図 2 システム構成

からの画像に対して、顔の検出と個人の判別を実施する。その処理手順としては、得られた画像に対して、最初に OpenCV で実装されているカスケード分類器²⁾を使用し、顔の検出を行う。次に、検出した顔画像に対して、学習済みの顔であるかを判別する。これにより、顔の検出から判別に至るまでの処理量を抑えることができ、短時間で判別を可能にした。

5.2 個人判別のための学習データの取得

個人判別に際して、システムに予め個人を学習させる必要があり、事前に登録の処理を実施するようにした。通常は、個人の画像を何枚も取得する必要があるが、今回は、OpenCV のカスケード分類器を使用し、数 10 秒程度の時間で自動的に 50 枚取得するようにした。これにより、学習データの取得における人手によるマニュアルでの作業をできる限り排除することができ、個人の登録を容易に実現することができた。

5.3 個人判別の学習アルゴリズム

顔判別については、前節で取得した個人の顔画像を学習データとして、特徴を抽出する学習アルゴリズムの一つである“LBPHFaceRecognizer”³⁾を使用して学習させた。本学習アルゴリズムは、画像の局所的な表現とその近傍の相対値から特徴量を抽出し、それをヒストグラムとして比較するアルゴリズムである。本学習アルゴリズムは、画像の局所的な表現とその近傍の相対値から特徴量を抽出し、それをヒストグラムとして比較するアルゴリズムである。本学習アルゴリズムは、対象物の画像に対する照明変化の影響を受けづらく、計算コストが低く比較的精度よく判別できるという特徴がある。結果として、照明、天候や季節等に対して細かく設定を調整することなく個人判別の機能を実装することができた。

6. 個人判別と検温データの画面表示

個人判別した後、判別した個人データと共に、計測した体温のデータをモニタに表示する。その際、体温が高い場合には、背景を変えると共に、注意喚起するメッセージを表示した(図 4)。体温が 37.5℃未満の場合は、背景を緑色にした。体温が 37.5℃以上の場合は背景を赤色にする共に、注意喚起のメッセージを表示させた。

7. データの管理と閲覧

7.1 データの管理と閲覧

個人判別と計測した体温のデータについて、サーバー内に構築した DB で管理すると共に Web ブラウザでアクセスし閲覧できるようにした。サーバーは、ubuntu で構築した。採用したソフトウェアとしては、DB は MySQL、Web サーバーは Apache をそれぞれ採用し、Docker を使用しコンテナ型の仮想環境で構築した。

7.2 計測データの閲覧とアクセス制御

DB で管理している各種データについては、Web ブラウザでアクセスすることにより、閲覧できるようにした。その際、アカウントによるアクセスを制御することで、閲覧できる範囲を限定させた。各学生であれば、個人の体温の履歴に対してアクセスすることができ、教員であれば関係する学生の体温を一覧表示できるようにした(図 4)。

アクセス制御の方法としては、セキュアなデータへのアクセスを実現しつつ、アカウント管理のオペレーションを容易に実現するために Google Identity Services⁴⁾を使用した認証機能を導入し

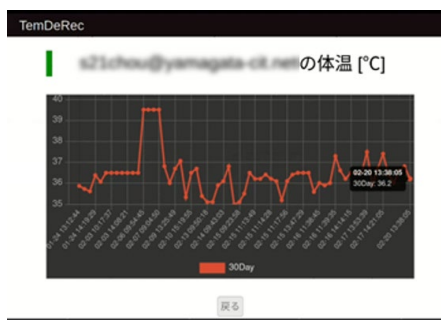


(a) 温度が 37.5℃未満の場合



(b) 温度が 37.5℃以上の場合

図 4 モニタへの計測結果とメッセージの表示



(a) 個人の体温のグラフ表示

ID	名前	時間	体温
22014		2023-02-20 08:32:1636.5	
22102		2023-02-20 08:30:1236.6	
22103		2023-02-20 08:45:0036.6	
22104		2023-02-20 08:40:4236.5	
22105		2023-02-20 13:28:3435.5	
22106		2023-02-20 08:36:5136.2	
22108		2023-02-20 08:31:2336.4	
22109		2023-02-20 08:46:2336.5	
22111		2023-02-20 13:38:0536.2	
22112		2023-02-20 08:43:1336.6	
22113		2023-02-20 08:48:1336.2	
22114		2023-02-20 08:25:2736.3	
22115		2023-02-20 08:34:5736.5	
22117		2023-02-20 08:33:4236.6	

(b) 担当する学生の一覧表示

図 4 Web ブラウザでのデータ表示



図 5 Web ブラウザでのユーザー認証画面

た。これは、Google のアカウントを使用したユーザー認証を Web サイトのアクセス制御に使用するサービスである。本サービスを利用することにより、作成したシステムに対して Web ブラウザでアクセスする際に、Google のアカウントによるユーザー認証機能を利用することができる (図 5)。

7.3 データの閲覧

Web ブラウザによるデータ閲覧画面は、レスポンスデザインによるデバイスによらないページレイアウトを実現した。Web のプログラミングとしては、PHP と JavaScript を使用した。レスポ

ンスデザインを採用するにあたり、フレームワークとして Bootstrap⁵⁾を使用した。

8. 結果と評価

AI と IoT の技術を活用することにより、人間がカメラの前に立つだけで、個人を判別し体温を自動計測し、その結果をデータベースで管理する IoT システムを実現することができた。しかし、体温の計測について実際よりも高めに計測される場合があることと、個人を誤判別する場合があります。各精度を高めていく必要がある。

9. おわりに

本研究では、人間がカメラの前に立つだけで個人を判別し体温を自動計測し、その結果をデータベースで管理する IoT システムを開発することができた。本システムにより、個人のユーザーでは日々の状態をグラフで可視化することができ、組織を管理するユーザーでは、組織内の体温計測状況を一覧で確認する機能を実現できた。新型コロナウイルス感染拡大防止の観点で本システムを使用し、組織のメンバーに対して体温計測し、それを把握することが有効であると考えられる。今後は、本システムでの顔の判別と検温の精度を向上させることで、実用性を高める必要がある。

謝辞

本テーマに卒業研究として取り組んでくれた山形県立産業技術短期大学校庄内校情報通信システム科卒業生の小林琉生さん、西塚天音さん、長南弥真斗さんに感謝いたします。

文献

- 1) 小林琉生, 西塚天音, 長南弥真斗:「画像認識による個人判別と自動検温~KK~」, 山形県立産業技術短期大学校庄内校 情報通信システム科 令和 4 年度卒業研究論文(2023)
- 2) OpenCV Foundation, “opencv/data/haarcascades”, <https://github.com/opencv/opencv/tree/master/data/haarcascades>
- 3) OpenCV Foundation, “OpenCV modules Face Analysis”, https://docs.opencv.org/4.8.0/db/d7c/group_face.html
- 4) Google LLC, “Goolge”, <https://www.google.com/>
- 5) X Corp, “Bootstrap で高速でレスポンスなサイトを構築しよう”, <https://getbootstrap.jp/>