

QRコードと顔認証を用いた遅刻管理システムの開発

石井奏翔^{*1}

指導教員：小山浩二^{*2} 後田航平^{*2}

Email: k.ushiroda@hokkaido-c.ed.jp

*1: 北海道函館中部高等学校普通科3年

*2: 北海道函館中部高等学校

◎Key Words Python, 顔認証, QRコード, 遅刻管理

1. はじめに

現在、函館市近郊の多くの学校では遅刻してきた生徒の対応や遅刻情報の管理に生徒と教員の手間がかかっている。本研究では、プログラミング言語Pythonを用いたQRコードと顔認証による遅刻管理システムを開発し、生徒及び教員の遅刻情報の記録・管理の効率化及び自動化を試みた。

2. システム設計

2.1 システムの概要

本システムは、顔認証を行うGUIアプリケーションと遅刻管理をするWEBアプリケーションの2つのアプリケーションで構成されている。GUIアプリケーションはアプリケーション作成ツールであるPythonライブラリCustomTkinterを用いて作成し、学校現場においてパソコン端末上で遅刻情報の記録を行うことを可能とした。Webアプリケーションはアプリケーション作成ツールであるPythonライブラリDjangoを用いて作成し、教員がWEBサイト上で遅刻情報の閲覧・編集を行うことが可能となっている。図1は遅刻情報の記録の流れをフローチャートにより表したものである。

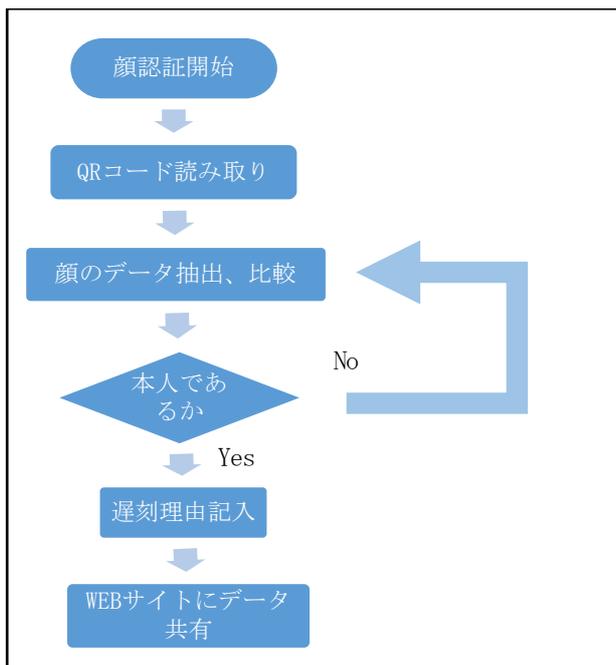


図1 遅刻情報記録の流れ

遅刻記録を行う際には、初めにQRコードを読み取り、システムが生徒を識別した後、顔認証による本人確認、遅刻理由の入力を行う方法を採用した。本システムで使用するQRコードには「学年」「クラス」「出席番号」「名前」の情報が記録されており、データの暗号化を行うことができるPythonライブラリCryptographyを用いて暗号化したデータを学生証に記載し、使用することを想定している。QRコードを使用することによって顔が類似している双子を識別して顔認証を行うことを可能とした。また、顔認証が実行されると、遅刻情報（生徒情報、登校時間、遅刻理由）がリアルタイムでWEBアプリケーションに共有されるように設計した。

2.2 顔のデータ化の手法

本システムでは、3Dカメラ(OAK-D S2, Luxonis)から取得した画像に対して、顔認識を実行することのできるpythonライブラリInsightFaceを用いて顔の特徴を表すデータを生成し、生成したデータを人工知能や機械学習が処理しやすい数値ベクトル表現に変換する手法であるEmbeddingを用いて顔の特徴を数値ベクトル表現に変換したものを使用した。本システムでは、光の影響によって顔をデータ化する際の精度が低下することを防ぐため、複数の画像から顔の特徴点を抽出し、その平均値を使用した。Embeddingによって生成されたデータは512個の要素からなるリスト型のデータで次のように表せる。

$$a_n = [v_1, v_2, v_3, v_4, \dots, v_{512}]$$
$$b_n = [v_1, v_2, v_3, v_4, \dots, v_{512}]$$

また、データを正確に収集するために空間の距離測定とAI処理を一体化した組み込みプラットフォームであるDepthAIに備わっているニューラルネットワークを用いて、顔の向きを推論を行い、顔の向きが正面の時のみ、データを取得する手法を採用した。

2.3 顔認証の手法

本研究では、コサイン類似度を用いて顔の類似度を算出・比較することで顔認証を行う方法を採用した。コサイン類似度とはベクトル空間における2つのベクトルがなす角のコサイン値のことであり、以下の(1)式により算出した。コサイン類似度の値域は-1から1の範囲の実数であり、1.0に近いほど顔の類似度が高いことを表す。

$$\cos \theta = \frac{\sum_{k=1}^n a_k b_k}{\sqrt{\sum_{k=1}^n a_k^2} \sqrt{\sum_{k=1}^n b_k^2}} \quad (1)$$

n: データの次元数(本システムの場合 n=512)

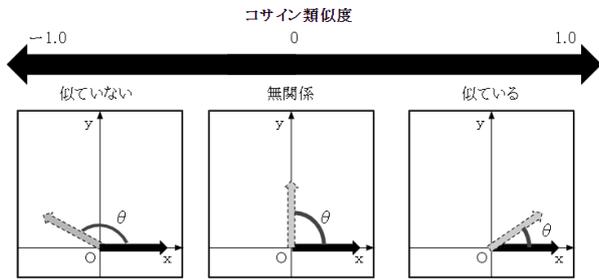


図2 コサイン類似度について

2.4 本人とみなす基準

本研究では顔認証において本人とみなす基準を定めるため、同一人物の場合と同一人物でない場合のコサイン類似度を比較した。実際に学校現場での使用を想定し、同一人物の場合において事前に登録済の顔データの撮影環境と比較するデータの撮影環境は異なる環境に設定した。下記の図3は同一人物の場合とそうでない場合のコサイン類似度を箱ひげ図で表している。

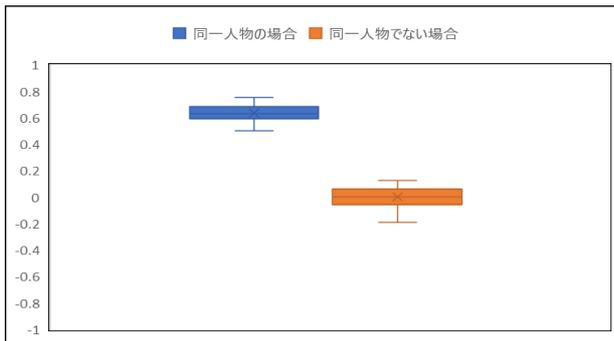


図3 コサイン類似度の比較

表1 コサイン類似度の平均値

項目	平均値
同一人物の場合	0.658
同一人物でない場合	0.014

表1は各場合におけるコサイン類似度の平均値を表している。同一人物とそうでない場合のコサイン類似度についてt検定を行ったところ、有意差が認められた ($p < 0.05$)。多様な撮影環境によって同一人物であってもコサイン類似度が小さく出てしまうことや、本システムではQRコードによる個人識別を採用していることを考慮し、コサイン類似度 ≥ 0.6 の場合、認証対象を本人とみなすように基準を設定した。

2.5 瞬き検知による不正認証防止策

顔認証は画像から顔を検出し、抽出されたデータを比較している。この特性上、写真に映っている顔でも顔認証ができてしまうという問題点もある。そこで、本シス

テムではMediapipeを用いて目付近のランドマークの座標から目のアスペクト比(EAR)を計算することによって瞬きを検知する機能を実装し、認証対象が本物の人間であるかを判別することで、セキュリティの向上を目指した。MediapipeとはGoogleが開発した機械学習と画像処理の機能が搭載されたPythonライブラリであり、眉毛の始点、鼻の先端、口の両端などの顔の特徴点の抽出をリアルタイムでおこなうことができる。Mediapipeによって抽出された特徴点が記載されている図4に示した6点の座標を以下のように表した。

$$p_1=(x_1, y_1), p_2=(x_2, y_2), \dots, p_6=(x_6, y_6)$$

目のアスペクト比の値は以下の(2)式によって算出される。アスペクト比の値は目が開いている状態で通常0.2~0.3の範囲にあり、目が完全に閉じている状態では通常0.1以下になることが知られている。そこで、本システムではアスペクト比が0.2以下から0.2以上に切り替わったときに瞬きをしたとみなしている。

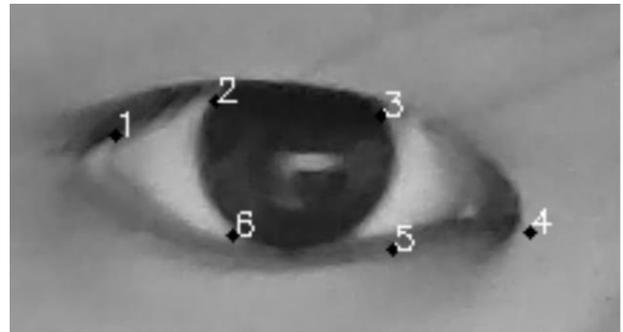


図4 目付近のランドマークをプロットした画像

$$\text{アスペクト比} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|} \quad (2)$$

3 おわりに

本研究では、プログラミング言語 Python を用いた QR コードと顔認証による遅刻管理システムを開発した。本システムにより、生徒及び教員の遅刻情報の記録・管理の効率化及び自動化が可能となった。

今後の課題としては、顔認証の処理速度向上とユーザビリティの向上の2点が挙げられる。高速処理が可能なライブラリの使用やプログラムの改良により、さらなる処理速度の実現が見込まれる。また、ユニバーサルデザインに配慮したアプリケーションデザインを設計し、ユーザビリティの向上に努めたい。

また、新たな顔認証の手法として3Dカメラによって計測される深度情報を用いた顔認証や不正認証防止策の開発についても併せて研究を行っていきたい。

謝辞

本研究は、一般社団法人未来教育サポートをはじめ、2023年に北海道函館市で行われた「第1回高校生探究リサーチフォーラム in 函館」の関係者の皆様の支援を受けて実施しました。また、本研究にあたり、数多くのご助言をいただいた小山先生に感謝申し上げます。