

画像認識を用いた自動打刻の勤怠管理システム

19C1004 池田泉海

Attendance management system using image recognition

Motomi Ikeda

We propose an automated the attendance management system. Conventional attendance management systems require manual time stamping. leading to problems such as the effort involved in stamping for each entry and exit, and the possibility of forgetting to stamp. To avoid this problem, we attempt to use image recognition to automate the time stamping process. The proposed method detects attendance and departure by employing image recognition to detect person. Automating the time stamping process has resulted in a reduction in effort for each entry and exit, and an improvement in the objectivity of the data.

Key Words: Attendance management system, Automated time stamping, Real-time object detection

1. 緒 言

近年、労働環境の変化に伴い、効率的で正確な勤怠管理の重要性が増している。そのため、勤怠管理システムでは、打刻に生体認証や顔認証を導入するなどして、データの客観性が高められている。しかし、ほとんどの勤怠管理システムは手動での打刻が必要であり、手動打刻には出退勤毎の打刻労力や打刻忘れ等の問題が存在する。

本研究では、これらの問題に対処するために YOLOv8⁽¹⁾ による画像認識で打刻を自動化した勤怠管理システムの提案を行う。YOLOv8 は物体検出アルゴリズムの一つで、リアルタイムでの物体検出によく用いられている。これは YOLO が画像を一度だけ見て物体の位置とクラスを同時に予測することで高速な物体検出ができるためである。

これらの特徴を利用して、リアルタイムでカメラ画像から人物を検出することで打刻を自動化する。打刻を自動化することで、従来の手動で打刻をする手法に比べ、利用者にとって手間を軽減し、打刻忘れを防止することができると考えられる。

2. 構築するシステム

本研究では、図 1 に示した構成で勤怠管理システムを作成する。システムの詳細について次項、次々項で説明する。

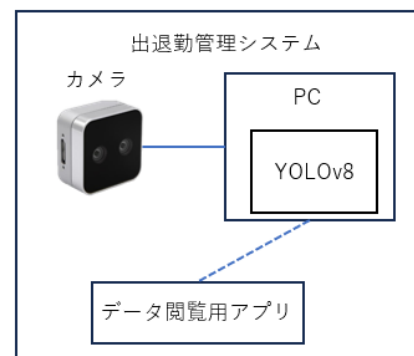


Fig. 1 System configuration

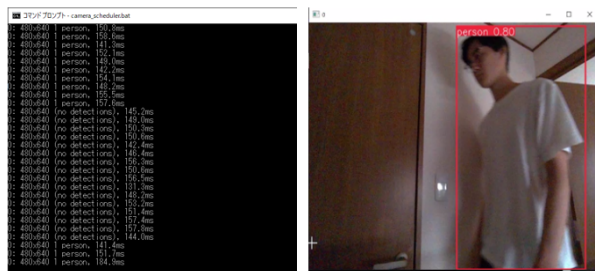
2・1 入退室する人物の検出 設置されたカメラを指定した時間だけ起動しておき、カメラ映像を約 0.2 秒毎に画像で取得する。取得した画像から YOLOv8 を用いて人物の検出を行う。画像から人物が検出された場合のみ画像を保存する。その日保存された最初の画像と最後の画像が保存された時間をそれぞれ出勤時間、退勤時間とする。

2・2 データのアップロード データ閲覧用の web アプリに入退勤時の画像データと日時データをアップロードする。毎日指定した時間になると、その日に保存されたデータが自動でアプリにアップロードされる。

3. 実験

実機を用いて図 1 のシステムを構築し、人物の検出からデータのアップロードまで動作可能か検証する。検証に使用するカメラは Intel Realsense Depth

図2からカメラからの映像を約0.2秒毎に画像で取得し、人物が画像内に写った場合、人物検出ができていることが分かる。



3・2 データのアップロード 指定した時間に自動でアプリ上にデータをアップロードすることができるか実験する。3・1 で保存された画像データとそれに紐づく日時データを指定した時間にアプリにHTTP リクエストを送り、データベースに保存する。図3(追加予定)のように、アプリ上でデータを確認することができ、データがアップロードできていることが分かる。

本研究では、YOLOv8で人物検出をすることで、画像認識を用いた自動打刻の勤怠管理システムを提案した。実験より、システムが動作可能なことを確認した。

[1] Ultralytics (2023) ultralytics [Source code]. <https://github.com/ultralytics/ultralytics>.