

# 顔画像の深層学習による オンライン試験システムの替え玉受験の検出

山下 義行<sup>1,a)</sup> 米満 潔<sup>1</sup> 古賀 崇朗<sup>1</sup> 高\*崎\* 光浩<sup>1</sup> 堀 良彰<sup>2</sup>

**概要:** 佐賀大学の Moodle システムにはオンライン試験時にノート PC の内蔵カメラで顔画像を撮影する機能が拡張実装されている。本研究ではその顔画像の深層学習によって個々の学生の顔を識別し、替え玉受験を検出する。まず替え玉の可能性の高い画像を自動検出し、次に教員はその画像を目視確認し、替え玉を認定する。佐賀大学の 2022 年度前期のひとつの科目で収集した画像 1 万 3 千枚についてこれを適用し、3 件の替え玉受験を発見した。

**キーワード:** 2110204 Web ベーストレーニング、2150205 画像認識・理解、2130102 機械学習

## Detection of Proxy Examination Takers on an Online Examination System by Deep Learning of Face Images

**Abstract:** The Moodle system in Saga University has the extended implementation for capturing face images of each student examinee with their own laptop PC's built-in camera during online examination. In this study, in order to detect the cheats of proxy examination taking, the faces of individual students are identified by deep learning of the captured face images. At the first step of the procedure, the recognition program automatically detects the images with higher suspicion of the cheats and then the teacher visually checks them and certifies the cheats. This method was applied to 13 thousand of face images collected from the students in a class in the first semester of 2022 at Saga University, from which three cases of proxy examination taking were detected.

**Keywords:** 2110204 Web based training, 2130102 Machine learning, 2150205 Image recognition/understanding

### 1. はじめに

コロナ禍が始まって以来、様々な新しい教育手法が模索されている。佐賀大学ではそのひとつの試みとしてオンライン試験時の不正の発見と抑止を目指し、学習管理システム Moodle [1] に受験時の学生の詳細な挙動を取得・保存する拡張機能を実装し [2]<sup>\*1</sup>、2022 年度からは本格運用を開始した。

システムの拡張機能のひとつにノート PC に内蔵された

カメラを用いて受験中の学生を撮影し、システムに画像ファイルとして保存する機能がある。これによって受験学生を画像として特定できるため、不正の発見、特に替え玉受験の発見が可能になる。不正が発見できるならば、結果として不正の抑止にもつながると期待できる。しかしシステムに保存される画像数は膨大である。佐賀大学の 2022 年度前期に保存された画像数は約 2 万 5 千枚であった。ひとつの科目で 1 万枚を超える場合もある。不正のチェックを教員が行おうと考えても 1 万枚を超える画像をひとつひとつ目視確認することは困難である。

本研究ではこの替え玉発見を大幅に省力化するソフトウェアツールを開発する。そのための基盤技術として機械学習ライブラリ TensorFlow [3] を用いる。主な処理は以下の二つである。

<sup>1</sup> 佐賀大学全学教育機構  
Organization for General Education, Saga University

<sup>2</sup> 佐賀大学総合情報基盤センター  
Saga University Computer and Network Center

a) yaman@cc.saga-u.ac.jp

<sup>\*1</sup> このシステムは、令和 3 年度国立大学改革強化推進補助金対象事業：「ポストコロナにおける ICT 活用学修評価環境の構築」によって構築された。

- (1) 画像からの顔の抽出
- (2) 顔画像の個人識別

まず全てのカメラ画像から顔が写った領域を抽出する。もし画像に顔が映っていない、あるいは複数の顔が映っているならば、それも不正と疑われる。顔画像は学習データと評価データに二分され、学習データの顔画像は学籍番号毎に深層学習へ入力する。学習後、評価データの顔画像からは学籍番号を求める。その学籍番号が想定される番号と異なるならば、替え玉が強く疑われる候補となる。そのような画像は、本研究で開発する目視確認ツールも用いて簡単な操作で効率的にチェックする。

オンライン試験を用いた受験の不正を発見する試みは既に様々な方法で行われている。本研究と同様に顔画像の比較で替え玉を発見する商用システムも既にある [4]。本研究とそれらとの大きな違いは以下である。

- (1) 大学の授業の中で学生がノート PC で Moodle のオンライン試験を受験することのみを想定しており、特殊な環境、特殊な機材を必要としない。
- (2) 試験を行う毎に累積されていく画像を用いたインクリメンタルな処理を行う。事前に学生の顔画像を登録する必要がない。
- (3) 替え玉の疑惑の度合い（疑惑度）を測る独自の指標を導入し、替え玉の目視確認作業を軽減している。

本稿の構成は以下の通りである。2 節では佐賀大学版オンライン試験システムの機能について紹介する。3 節では替え玉受験の検出方法の概要を述べる。4 節では 2022 年度前期科目についての検出実験の結果を述べる。5 節では深層学習による個人識別の結果を詳しく分析する。

## 2. 佐賀大学版オンライン試験システム

本研究のオンライン試験システムは、Moodle の小テスト機能に以下の機能を付加したものである [2]。

**静止画撮影** オンライン試験受験中の問題画面遷移の度に学生の様子をノート PC 内蔵のカメラで静止画として撮影する。

**受験時の不正操作記録** オンライン試験受験中の学生の以下の操作を検出し、記録する。

- (1) 問題表示ウィンドウがフルスクリーンでない。
- (2) コピー&ペースト操作を試みた（実際には実行できない）。
- (3) マウスの右クリックを行なった。
- (4) 他ウィンドウへのアクティブ画面遷移を行なった。

**位置情報、IP アドレスの記録** オンライン試験受験中の PC の緯度・経度、接続 IP アドレスを記録する。

上記の個々のデータには記録日時が秒単位で付随する。

## 3. 替え玉の検出の方法

本研究では前節のカメラ画像を自動解析し、替え玉受験

が疑われる画像を検出する。図 1 は検出手順の全体をイラスト化したものである。この節ではこれを概説する。

### 3.1 カメラ画像

カメラ画像には (1) 学籍番号、(2) 科目名、(3) 撮影日時 (秒単位) のタグが付随する。

画像はオンライン試験の問題画面を遷移する度に自動的に撮影されるから、たとえば 2 問からなる小テストを毎週、実施する場合、学生は解答内容の見直しを行うために数度の画面遷移を行う。よって週当たり 3~4 枚、15 週で 50 枚程度のカメラ画像が収集できるだろう。期末テストでは、たとえば 20 問の設問を用意した場合、設問ページ間の前後移動が自由ならば、30~40 枚の画像が集まるだろう。

実際、佐賀大学の前期の、ひとつの教養系の科目 (受講学生数 147 名、放棄学生は除く) では小テストにおいて計 9,188 枚の画像が収集されており、学生当たり約 63 枚であった。期末テスト (設問数 15) では計 3656 枚、学生当たり約 25 枚が集まった。

### 3.2 学習データと評価データの切り分け

カメラ画像から抽出された顔画像は深層学習のため、学習データと評価データに分ける。ここでは以下の二つの日時  $T$ 、 $T'$  を定めて、データ収集を行う。

**学習データ収集期限  $T$  :** 授業開講日 (本研究では 2022 年 4 月 1 日 0 時 0 分とする) から日時  $T$  までに収集された画像を深層学習の学習データに用いる。

**評価データ収集期限  $T'$  :**  $T$  以降、 $T'$  以前に収集された画像は深層学習の評価データに用いる。

### 3.3 顔画像の切り出し

深層学習のための前処理としてカメラ画像中から人の顔を抽出し、顔画像として切り出す (図 1 の左側を参照)。

この処理は個人の識別ではなく単に人の顔を抽出するだけであるから、すでに多くの技術開発が行われており、顔画像認識のための学習済みのモデルも種々、入手可能である。ただし学生は大学内で受験する場合にはマスクを着用し、自宅で受験する場合にはマスクを着用していない場合が多い。よってその両方の場合の顔を認識し、切り出す必要がある。本研究では Tensorflow 版の既学習データ [5] を用いた。なお、画像中の顔と認識された領域には確信度 (confidence)  $\in [0, 1]$  が付随する。本研究では確信度 0.98 以上のものを顔と認識する設定を行なった。この設定の場合、画像の隅に映り込んだ小さなサイズの顔は認識から除外される。

また、マスクを着用している顔画像を深層学習の学習データとしてマスクを着用していない顔画像を識別する場合 (その逆の場合も同様) の認識率を上げるため、切り出された顔画像に加え、その顔画像の上半分の画像も学習デー

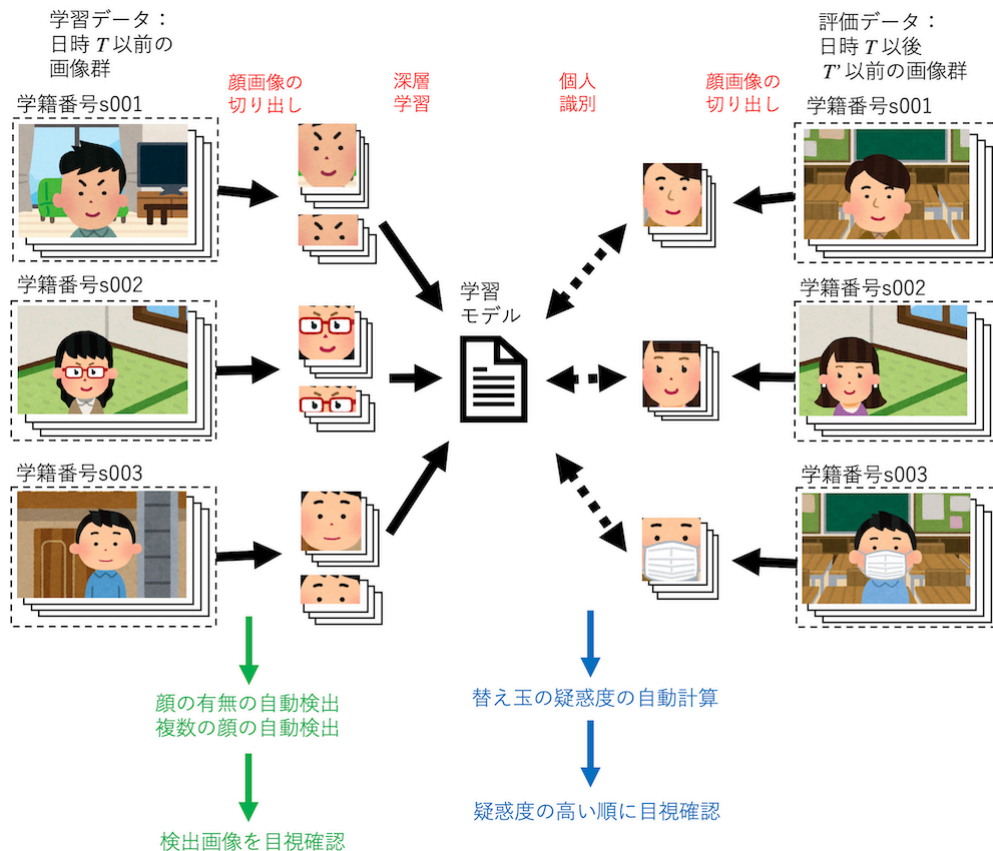


図 1 不正検出の処理の流れ：学籍番号 s001 の学生は学習データと評価データで顔が異なる。s002 の学生は学習データではメガネを着用しているが、評価データでは着用していないため、個人識別が困難。s003 の学生は学習データではマスクを着用していないが、評価データでは着用しており、顔の上半分画像によって識別は可能。

タとして用いる（図 1 左側の「顔画像」を参照）。簡単な工夫にすぎないが、個々人の識別率が上がると期待される。

顔認識の技法として Tensorflow の他に Haar Cascade 識別器の学習済みデータを用いる方法 [6] も試みた。しかし後者は前者に比べて認識率が約 2 割低い結果となったため、本研究では前者のみを用いる。

### 3.4 不正検出（顔の有無の検出）

本項で述べる内容は替え玉検出とは直接関係しないが、不正の検出という意味では重要である。

不用意な盗撮防止のため、PC 内蔵のカメラをマスキングテープなどで目隠しする場合がある。同様の理由でカメラ画像を別の画像に差し替える設定を PC に行なっている場合がある。このような行為の下でオンライン試験システムを利用してもカメラ画像に顔が映っておらず、替え玉検出に支障がある。

そこで前項の処理の一部として顔画像の有無を自動検出する（図 1 の左下の緑色の部分を参照）。具体的には以下の通りである。

(1) 全てのカメラ画像について顔（ここでは個人は特定しない）が映っているか否か自動検出する。

表 1 全ての画像に顔が写っていない学生の自動検出

システム登録の全学生数	387 名 (画像数 25,770 枚)
画像に顔が全く検出されなかった学生数	37 名

表 2 表 1 の 37 名の詳細

カメラ画像に顔を映さない細工を行なっていると判断できる者	25 名
過失によると判断できる者	6 名
上のいずれとも判断できない者	6 名

(2) ある学生の全ての画像に顔が映っていないと検出されたならば、教員がそれら画像を目視確認した後、該当学生に「オンライン試験受験時にはカメラ撮影に協力するように」指導する。

上の処理を佐賀大学のオンライン試験システムに保存されている 2022 年度前期の全ての画像について行なった結果が表 1 である。その表中の 37 名の学生の画像を目視によって調査した結果は表 2 である。ここに「過失」とは、たとえば以下のような場合である。

(1) 受験する際の学生の姿勢に習慣的な癖があり、図 2 の左の画像、中央の画像のように、カメラ画像に顔全体



図 2 顔検出を妨げるカメラ画像 (左: 顔が画像から見切れている、中央: 顔がカメラに近すぎる、右: 輝度値が極端に低い)

表 3 1 枚の画像に複数人が映っている画像の自動検出

全画像数	25,770 枚
複数人が映っていた画像数	3 枚

表 4 表 3 の 3 枚の目視による詳細確認

プログラムの誤認識	1 枚
不正	2 枚

が写っていない。図 2 右のように暗い画像 (多くの場合、自室で部屋の照明を落として受験した場合の画像) は人目には顔を判別できないが、検出プログラムは輝度補正を自動的に行うため、極端に暗い場合を除き、顔の検出にほぼ支障がない。

(2) PC に複数台のカメラが内蔵/接続されており、顔の映るカメラが選択されていない。

### 3.5 不正検出 (複数人の顔の検出)

ほとんどの試験では問題を複数人で協力して解答することを禁じている。よって、1 枚のカメラ画像に複数人の顔が認識されたならば不正である。

前項同様に、3.3 項の顔画像の切り出し処理の一部としてこれを半自動検出する (図 1 の左下の緑色の部分を参照)。手順は以下である。

- (1) 全てのカメラ画像について複数人が認識される画像を自動検出する。
- (2) その画像の背景に (a) 通行人、または後ろの机で受験中の学生がたまたま映り込んだ等の状況か否か、あるいは (b) プログラムの誤認識によるものか否かを目視で判断する。これに該当しないならば不正である。

前項と同じく 2022 年度前期に取得されたカメラ画像について調べた結果が表 3 である。表中の 3 枚 (いずれも 1 枚の画像に 2 名が映っているとの認識結果であった) を目視によって確認した結果が表 4 である。

### 3.6 顔画像の深層学習

3.2 項の方法で学習データに分類したカメラ画像について 3.3 項の方法で顔画像を切り出し、それら顔画像を学生毎にまとめて深層学習を行う。ここでは Tensorflow-hub[7] による転移学習を用い、学習時間を短縮する。

画像枚数が少ない場合には Keras ライブラリの ImageDataGenerator[8] を用いて画像の水増し処理を行う。本研究では画像枚数が少なくとも 100 枚以上になるように水増しを行う。学習繰り返し回数は次節で述べる実験では

8 万回とした。

### 3.7 不正検出 (替え玉の検出)

この項では替え玉を検出する方法について述べる。実際の実験での設定、結果、考察は次節で述べる。

#### 3.7.1 疑惑度の計算

3.2 項の方法で評価データに分類した画像についても 3.3 項の方法で顔画像を作成しておく (図 1 の右側を参照)。カメラ画像をそのまま評価に用いることも可能だが、顔画像を用いる方が誤認識が減る。

評価データの各顔画像  $F$  の個人識別を行うことを考える。

Tensorflow-Hub の画像識別では、顔画像  $F$  が学籍番号  $n$  の学生の顔であることの確信度 (confidence)  $C[F, n] \in [0, 1]$  を全ての  $n$  について知ることができる。確信度  $C[F, n]$  の値が大きいほど  $F$  が学籍番号  $n$  の学生の顔画像である可能性が高い。ここで画像  $F$  にタグ付けされた学籍番号を  $N(F)$  と表すこととする。そして、その画像  $F$  の疑惑度 (suspicion) を

$$S[F] = 1 - C[F, N(F)]$$

と定義する。そうすると、疑惑度が大きいほど  $F$  が  $N(F)$  である可能性は低く、替え玉受験が疑われる。

替え玉受験検出のため、評価データに含まれる全画像  $F_1, F_2, \dots$  についてそれぞれ疑惑度  $S[F_1], S[F_2], \dots$  を計算し、疑惑度が大きいものから順に並べた画像の並びを  $G_1, G_2, \dots$  とする。よって以下が成り立つ。

$$S[G_1] \geq S[G_2] \geq \dots$$

以下では簡単のため、疑惑度の大きい顔画像  $G_i$  を単に疑惑画像と呼ぶ。

#### 3.7.2 目視確認の方法

疑惑画像の順位が求められたならば、次はその目視確認である。疑惑の大きい順に確認を行い、疑惑が十分小さいと判断された時点で確認作業を中断すればよい。

替え玉の有無の目視確認は、ここでは以下の 5 種類の画像を目視で相互比較して行う。

- (1) 疑惑画像  $G_i$ 。
- (2) 疑惑画像と同じ学籍番号でタグ付けされた評価データ中の画像の並び  $G'_1, G'_2, \dots$ 。同じ学籍番号であるから以下が成り立つ。

$$N(G_i) = N(G'_1) = N(G'_2) = \dots$$

さらに以下の条件で疑惑度順に並べておくものとする。

$$S[G_i] > S[G'_1] > S[G'_2] > \dots$$

疑惑画像  $G_i$  だけで替え玉の有無を判断できない場合

には、それよりも疑惑度は小さいものの、同じ受験環境下で撮影されたこれらの画像が判断の助けとなる。 $G'_1, G'_2, \dots$  の中に替え玉が発見できる場合もある(後に述べる表6の  $T=7$ 月23日がそれである)。

- (3) 疑惑画像について最も高い確信度を与える学籍番号  $M$  でタグ付けされた学習データ中の画像の並び  $P_1, P_2, \dots$ 。つまり、全ての学籍番号  $s001, s002, \dots$  に関する疑惑画像  $G_i$  の確信度を  $C[G_i, 1], C[G_i, 2], \dots$  とするとき、それらの中の最大確信度  $C[G_i, M]$  を与える学籍番号が  $M$  である。 $M$  は疑惑画像に映っている人物の学籍番号の第1候補である。
- (4) 疑惑画像について2番目に高い確信度を与える学籍番号  $M'$  でタグ付けされた学習データ中の画像の並び  $P'_1, P'_2, \dots$ 。 $M'$  は疑惑画像に映っている人物の学籍番号の第2候補である。
- (5) 疑惑画像と同じ学籍番号でタグ付けされた学習データ中の画像の並び  $Q_1, Q_2, \dots$ 。もし  $Q_1, Q_2, \dots$  が上記(3)または(4)の画像の並びに等しいならば、代わりに疑惑画像に3番目に高い確信度を与える学習データ中の画像群を用いる。

上の定義の下、以下の判定を行う。

- (1) もし(1)、(2)と(5)が同一人物であるならば、疑惑は晴れる。
- (2) もし(1)、(2)と(3)、あるいは(1)、(2)と(4)が同一人物であるならば、替え玉を請け負った人物を特定した上で替え玉は確定である。
- (3) もし(1)、(2)が(3)、(4)、(5)とは異なる人物であるならば、替え玉を請け負った人物は特定できないが、替え玉は確定である。

### 3.7.3 目視確認を支援するアプリケーションプログラムの開発

上の5種類の画像をコンピュータディスプレイ上に自動的に表示し、対話的に目視確認を行うアプリケーションプログラムをPythonを用いて開発した。その実行中のスナップショットが図3である。なお、プライバシー保護の観点から学生の顔画像は全て胸像のイラストに置き換えた。実装を簡単にするため、個々の顔画像はOpenCVの個別のウィンドウに表示する[9]。また、顔画像を原画像をそのまま表示した場合、図2の右のような輝度が極端に低い画像では目視での顔の比較が困難であるから、ヒストグラム平滑化[10]した顔画像の表示を行う。

図3と、先に示した(1)~(5)の画像との関係は以下の通りである。

- 左上の赤枠の画像が(1)の疑惑画像である。なお、類似した疑惑画像の繰り返し表示を避けるために、各学籍番号タグについて最大疑惑度の画像1枚のみを目視確認対象とする。

- 左上の赤枠の右隣の画像の並びが(2)の、疑惑画像と同じ学籍番号タグを持つ評価データ画像である。左側の画像ほど疑惑度が高い。
- 2行目が(3)の画像の並びである。ディスプレイ上には水平方向に最大8枚の画像しか表示できないから、画像数が9枚以上の場合には画像を0.5秒毎に巡回的に上書き表示していく。緑の枠は画像の並びの先頭の位置の画像を表す。
- 3行目が(4)の画像の並びである。この行も画像を0.5秒毎に巡回的に上書き表示していく。
- 最下行が(5)の、疑惑画像と同じ学籍番号タグを持つ学習データ画像の並びである。青枠は疑惑画像と同じ学生番号タグを持つことを表している。この行も画像を0.5秒毎に巡回的に上書き表示していく。

プログラム実行ではスペースキー、'b'キー、'h'キーによる以下の対話的な操作ができる。

**プログラム開始直後** 疑惑度が最も大きい顔画像  $G_1$  について図3の表示を行う。

**スペースキー打鍵** 疑惑度がひとつ小さい顔画像  $G_{i+1}$  の表示へ進む。

**'b'キーの打鍵** 疑惑度がひとつ大きい顔画像  $G_{i-1}$  の表示に戻る。

**'h'キーの打鍵** 顔画像の表示についてヒストグラム平滑化の有無をスイッチする。ヒストグラム平滑化は、画像の輝度、コントラストが低いときでも顔のホクロなどの微細な特徴を鮮明に表示できるため、人物を同定するときに有効である。しかし学生を自然な素顔と比較したいときには原画像を表示する。その2種類の表示を随時切り替えて確実な目視を実施する。デフォルトではヒストグラム平滑化画像を表示する。

## 4. 替え玉検出実験

前節の方法に従い、3.1項で触れた佐賀大学の前期の教養系の科目について替え玉検出の実験を行なった。この科目は15回の授業をオンデマンドと小テストで行い、期末テストを対面で実施した。

この授業で取得した画像から3件の替え玉事案を発見した。以下ではそれについて報告する。

### 4.1 学習データと評価データの仕分け

学習データと評価データは、適切に切り分けることが重要である。図4のように、学習データに本来の学生と替え玉の学生の両方が含まれる場合、評価データの替え玉学生は高い確信度で本人と識別されてしまい、替え玉発見は困難になる。このように学習データ画像に替え玉画像が混入することを以下では学習データの**汚染**(contamination)と呼ぶこととする。

汚染を避けるために、3.2項の  $T$  と  $T'$  を表5のように



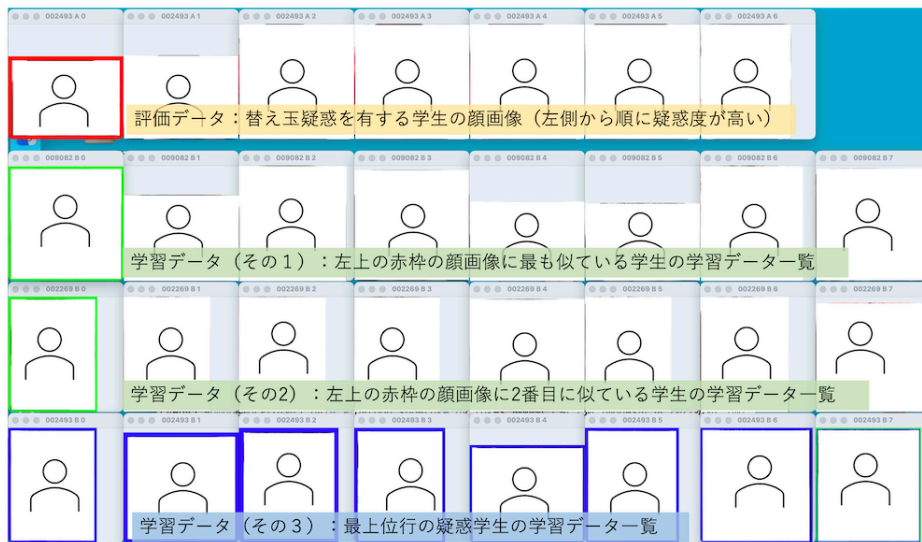


図 3 替え玉疑惑の目視による確認を行う Python プログラムの実行の様子：各ウィンドウには実際にはヒストグラム平滑化された学生の顔画像が表示される。各ウィンドウのラベルの数字 (002493 など) はシステム内部の学生管理番号であり、実際の学籍番号とは無関係である。

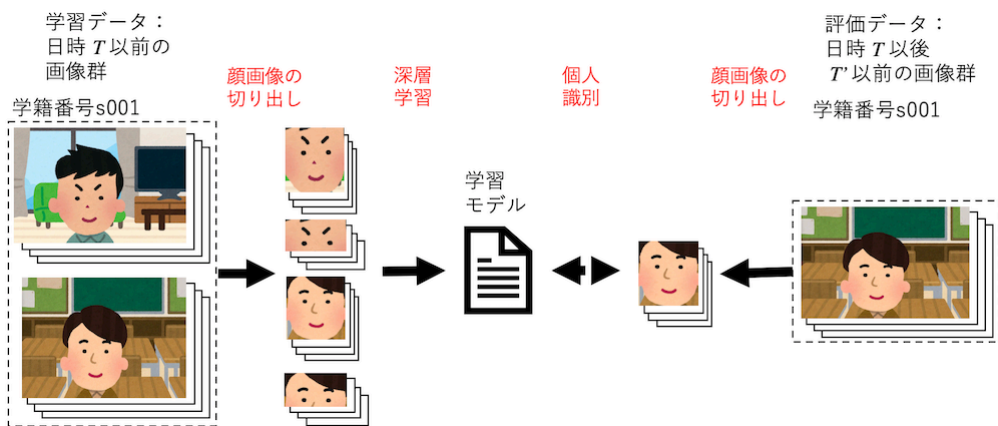


図 4 学習データの汚染：図 1 の学籍番号 s001 の学習データに替え玉画像が混入すると替え玉検出ができなくなる。

表 5 学習データと評価データを切り分ける日時  $T$  と  $T'$

#	$T$	$T'$	期間中のテストの種類
1	5月14日	5月21日	小テスト
2	5月21日	5月28日	小テスト
...	...	...	...
11	7月23日	7月30日	小テスト
12	7月30日	8月6日	小テストと期末テスト

設定した。つまり、#1の実験では2022年4月1日以降、5月13日23:59までに取得された画像を学習データとし、5月14日00:00以降、5月20日23:59までに取得された画像を評価データとする。もしこの実験で替え玉が見つからなければ、#2の実験では2022年4月1日以降、5月20日23:59までに取得された画像を学習データとし、5月21日00:00以降、5月27日23:59までに取得された画像を評価データとする。もし#1の実験で替え玉が見つかったならば、学習データの汚染を避けるため、その画像を除外し

た後に#2の実験へ進む。

なお、5月14日までの期間はデータ収集のみとし、評価は行わない。授業序盤であることから、この期間にデータの汚染は起きていないことを前提とする。

#### 4.2 替え玉の検出

上の方法に従い、学習データを深層学習し、その学習モデルを用いて各評価データ画像について疑惑度を求める。そして疑惑度が高い画像順に3.7.3項のアプリケーションプログラムで人手で目視確認を行う。

結果、表6に示すように、画像に付けられた学籍番号タグの学生とは異なる学生が画像に映っていた替え玉を3件発見した。いずれも小テストでの替え玉受験である。3件とも替え玉を請け負った学生を特定できなかった。つまり識別プログラムが確信度の上位2名(3.7.2項の項目(3)と

表 6 発見された 3 件の替え玉事案の概要：なお 3.5 項に述べた 1 枚の画像に複数人が映っていたケースはこれに含まない。

$T$	順位	疑惑度	正規化疑惑度
6 月 18 日	5/94	0.999915	0.993543
7 月 2 日	1/85	0.999994	0.999570
7 月 23 日	5/77	0.998650	0.897388

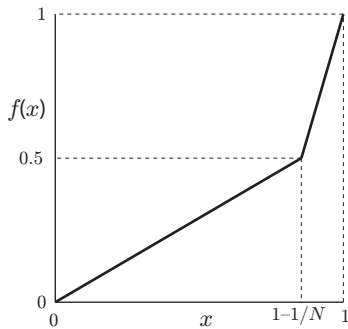


図 5 疑惑度を正規化する関数  $f(x)$

(4) に挙げた学生とは異なる学生であった。なお、期末テスト ( $T=7$  月 30 日、 $T'=8$  月 6 日) については特に丁寧に確認を行ったが、替え玉受験は無かった。

表 6 の順位  $m/M$  は、全疑惑画像（各学籍番号について 1 枚）の数  $M$  の中の当該替え玉画像の疑惑度順位  $m$  を表す。順位が小さいほど少ない手間で発見できる。表 6 の  $T=7$  月 2 日の場合、疑惑の最上位であるため、プログラム起動直後に替え玉受験を発見できた。 $T=6$  月 18 日、7 月 23 日の場合はスペースキー 5 回の操作であった。

表 6 の疑惑度は 1 に極めて近い数であるが、これは不自然ではない。というのも、学生数が  $N$  であるとき、仮に識別プログラムが個人識別を全くできず、全ての確信度が同じになるならば、すなわち  $C[G_i, n] = 1/N$  ならば、疑惑度は  $S[G_i] = 1 - 1/N$  となる。もし  $N = 100$  ならば、 $S[G_i] = 0.99$  となり、疑惑度上位の画像では 0.99 以上になるからである。

しかしこれでは疑惑度が学生数に依存し、疑惑度を分析する際の判断材料として使いにくい。そこで以下では、値が 0.5 のときに「疑惑があるとも無いとも言えない」解釈になるように疑惑度値を正規化する。そのために、ある関数  $f$  が定義域  $[0, 1]$  で単調増加であって、かつ

$$f(0) = 0, \quad f(1 - 1/N) = 0.5, \quad f(1) = 1,$$

を満たすように用意し、 $\bar{S}[F] = f(S[F])$  を正規化疑惑度 (normalized suspicion) と呼ぶこととする。本研究では  $f$  として単純な折れ線 (図 5 参照) を用いる。表 6 には正規化値も載せた。次章では正規化疑惑度の統計分布を議論する。

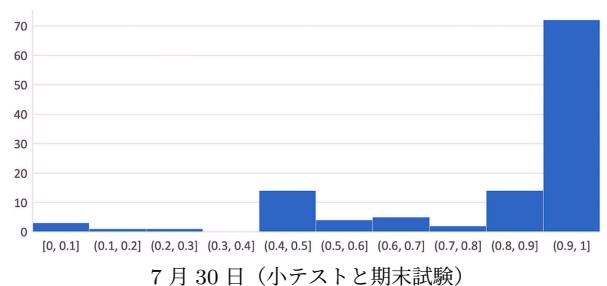
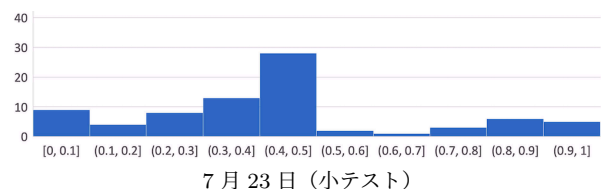
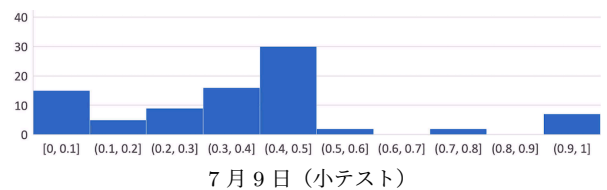
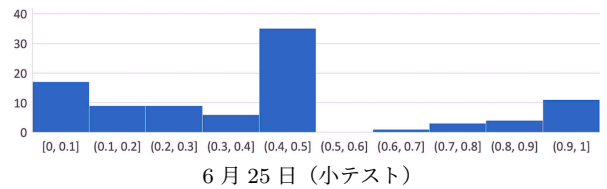
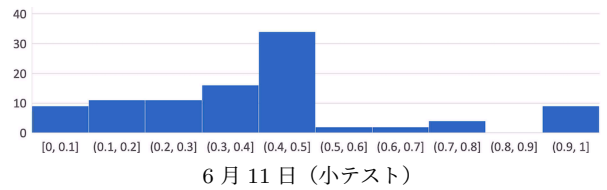
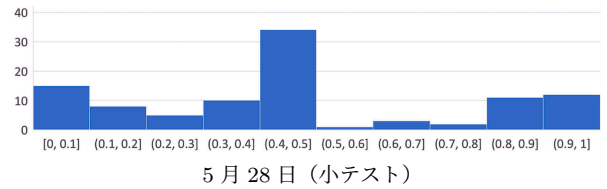
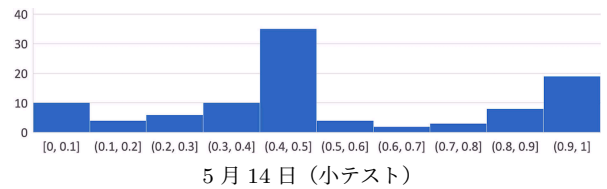


図 6 2 週間毎の正規化疑惑度の分布の変化：期末テスト画像を含む  $T=7$  月 30 日の分布のみ大きく異なる。

## 5. 実験結果の分析

### 5.1 疑惑度の統計分布

深層学習の効果を見るため、図 6 に、評価データ中の画像（ただし各学籍番号につき疑惑度が最も大きい 1 枚を採

表 7 正規化疑惑度の平均値の時間変化

$T$	標本数 (学生数)	正規化疑惑 度の平均値
5月14日	99	0.54
5月28日	99	0.47
6月11日	96	0.41
6月25日	93	0.42
7月9日	84	0.38
7月23日	77	0.44
7月30日	114	0.84

用)について2週間毎の正規化疑惑度の分布(各学籍番号タグの付いた複数の画像の中で疑惑度が最も大きい画像の疑惑度の分布)を示す。これについて以下が見てとれる。

- (1) 正規化疑惑度が0.5以上の画像の比率が時間と共に減り、逆に0.5以下の比率が時間と共に増えている。表7はそれを裏付ける正規化疑惑度の平均値の時間推移である。学習データが時間と共に蓄積され、識別の成功率が上がっていると言えよう。
- (2) 正規化疑惑度0.5は疑惑があるとも無いとも言えない中間値である。 $T=7$ 月23日以前の分布において疑惑度が中間値よりも小さい画像が多いのは、多くの画像で個人識別にある程度成功しているためである。
- (3)  $T=7$ 月30日の分布のみ異なる分布をなし、表7の平均値もここだけ非常に大きい。これについては次項で考察する。

## 5.2 個人識別を妨げる要因

オンライン試験システムにおける受験学生の個人識別を妨げる要因は以下の通りである。

**メガネの有無** メガネ着用の顔画像で学習したモデルではメガネ非着用の顔画像を認識できない(図1の学籍番号S002の学生を参照)。逆もそうである。これは識別を妨げる最も大きな原因である。

**マスクの有無** マスク非着用の顔画像で学習したモデルでは着用の顔画像を認識できない(図1の学籍番号S003の学生を参照)。逆もそうである。

3.3項ではこれを改善するために顔画像の上半分を切り取った画像も学習に用いる簡易な改善策を導入している。これによって識別率は向上する(定量的な議論は今後の課題)が、より精密な手法が望まれる。

**身だしなみの有無** 自宅でリラックスして受験する場合の顔画像で学習したモデルでは、身だしなみを整えた顔画像の認識率が下がる。

**姿勢の変化** 自宅で受験する場合には自由な姿勢で受験するため、大学教室での受験と顔の映り方が異なる場合がある。同一人物でも上向き(アオリ)と下向き(フカン)の顔ではかなり見え方が異なる。

$T=7$ 月30日の識別実験は、小テストの画像を学習デー

タとし、主に期末テストの画像を評価データとする場合である。学生の行動として小テストを自宅でマスクを着用せず受験し、期末テストを大学教室でマスクを着用して受験するケースが多いのだが、その場合、本人識別は容易ではない。さらに小テストを自宅でメガネを着用して受験し、期末テストを大学教室でメガネを着用せず(たぶんコンタクトレンズを装着して)受験した場合、プログラムによる本人識別はほとんど不可能である。これらの要因から、 $T=7$ 月30日のみ疑惑度が極端に高くなっている。

## 6. おわりに

佐賀大学版オンライン試験システムで取得された受験時の学生の顔画像を深層学習で識別し、替え玉受験を検出する方法を提案した。本方式で3件の替え玉受験を発見することができた。

今後はまず3.4項、3.5項で述べた不正検出を運用に取り入れる。これらは個人識別を必要としないため深層学習が不要であり、また認識精度が十分に高いため、すぐにも運用が可能である。

個人識別を伴う替え玉検出では、メガネ、マスクの着用の有無での識別精度向上について既存研究を参考に検討する。また深層学習時のパラメータを最適化し、識別精度を底上げする。授業においては対面での小テストを増やすなどして様々な場面でのデータの取得を行う。

## 謝辞

このオンライン試験システムの開発にあたりご協力いただいた株式会社SRA 西日本様に対して感謝の意を表す。

## 参考文献

- [1] Moodle: "Moodle - Open-source learning platform" <https://moodle.org/> (accessed 2023.1.18)
- [2] 米満 潔, 古賀 崇朗, 永溪 晃二, 山下 義行, 堀 良彰, 西郡 大: オンライン試験における不正行為抑止機能の開発、火の国情報シンポジウム2022論文集、A22-1, pp. 1-7, 情報処理学会九州支部(2022).
- [3] TensorFlow, <https://www.tensorflow.org/>.
- [4] デジタル庁: 講習・試験のデジタル化を実現する製品に関する公募結果, <https://www.digital.go.jp/policies/digital-extraordinary-administrative-research-committee/online-training-public-offers-result> (2022).
- [5] Chandrika Deb: Face Mask Detection, <https://github.com/chandrikadeb7/Face-Mask-Detection>
- [6] Paul Viola and Michael Jeffrey Jones: Rapid Object Detection using a Boosted Cascade of Simple Features, Inter. Conf. on Computer Vision and Pattern Recognition (2001).
- [7] TensorFlow Hub, <https://www.tensor-flow.org/hub>.
- [8] Keras: the Python deep learning API, <https://keras.io>.
- [9] OpenCV: High-Level GUI, [https://docs.opencv.org/4.x/d7/dfc/group\\_highgui.html](https://docs.opencv.org/4.x/d7/dfc/group_highgui.html).
- [10] OpenCV: Image Processing in OpenCV, [https://docs.opencv.org/4.x/d7/dbd/group\\_imgproc.html](https://docs.opencv.org/4.x/d7/dbd/group_imgproc.html).