令和 6 年度　卒業論文

VGG16 を利用した講義中の学生の受講状況の識別

北海道情報大学　経営情報学部
システム情報学科　長尾光悦ゼミ

2112601
篠原啓希

VGG16 を利用した講義中の学生の受講状況の識別

2112601

篠原啓希

概要

本研究では，VGG16 を利用した講義中の学生の受講状況の識別を提案する．ここでは，教室後方から撮影した学生の状態画像から 7 状態を識別する．また，VGG16 が画像のどこに注目し，識別を行っているのか可視化する．これにより，学生の受講状況を識別し，目視で受講状況を確認する講師の負担を軽減可能とする．

1. はじめに

近年，大学をはじめとする高等教育機関では，学生の学習態度や講義への参加状況を適切に把握し，学習支援を強化することが求められている．また，稲葉らの調査によるとパソコンを使用した e-ラーニングの教材を配信，管理するシステムである学習管理システム（Learning Management System: LMS）の大学での 2020 年度の全学導入率は 86.5%に上った[1]．そのため，e-ラーニングの普及とそれを対面授業と組み合わせた講義の浸透など，講義形態の多様化が進んでいることから，各講義での学生の受講状況を的確かつ自動的に識別し，学習態度の改善につなげる技術の開発が注目されている．

従来，大学の授業では，出席の確認や学生の理解度を把握するために，出席管理システムやアンケート調査，教員の観察といった方法が用いられてきた．これらの手法は，一定の効果を持つものの，いくつかの課題が指摘されている．例えば，手動での出席確認は時間がかかり，学生が代理出席を行う可能性がある．アンケートによる理解度確認は，学生の主観的な回答に依存するため，正確性に欠ける場合がある．また，教員が授業中に学生の態度を観察することも，教員の見落としや主観が入り込むリスクがあり，客観的なデータに基づく評価が難しいという課題がある．

さらに，近年の教育現場では，学習アナリティクス（Learning Analytics）の導入が進められており，データを活用して学生の学習行動を可視化し，教育の質を向上させる取り組みが行われている．特に，人工知能（AI）や機械学習技術の発展により，画像認識技術を活用して，講義中の学生の行動や態度を自動的に分析する手法が注目されている．例として，授業の開始から終了まで，10 分ごとに送信される教室のスナップショットから顔認識を用いて学生を認識し，授業の終了後，学生が管理画面にログインしてその授業の出席状況をすぐに確認できる AttenFace というシステムが提案されている[2]．このシステムは LMS である Moodle に統合し出席データを

Moodle 内で利用可能にしている．このような技術の導入により，教育機関は従来の手作業に頼らず，より客観的で効率的な方法で学生の受講状況を把握し，適切な支援を提供することが可能となる．

本研究では，畳み込みニューラルネットワーク（Convolutional Neural Network: CNN）の一つである VGG16 を用いて，講義中の学生の受講状況を識別する手法を提案する．VGG16 は，高い画像認識性能を有する事前学習済みモデルであり，比較的少量のデータで高精度な識別が可能である．本手法では，教室後方から撮影した学生の状態画像から正常(姿勢を正し聞いている)，机の下でスマホを操作，脇を開けてのスマホ操作，脇を閉じてのスマホ操作，突っ伏し，俯いて寝ている，上向きで寝ている状態の 7 状態を識別する．また，VGG16 が画像のどこに注目し，識別を行っているのか可視化することでシステムの識別根拠を理解でき，識別精度の向上に向けての対策を容易に立てることを可能とする．

2. 関連研究

2.1. 関連研究

関連研究として，吉武は ImageNet の学習済みモデルをファインチューニングして，学生の正面を向く，下を向く，寝ている，スマホを操作の 4 種類の受講態度の識別を行っている[3]．ファインチューニングとは，既存の学習済みモデルの重みのうち一部を再学習して，新しいデータセットに対応し，最終出力層を付け替えることで識別できるようにする手法である．
その結果，約 90%の認識率であったことが報告されている．しかしながら，この研究では正面から撮影した画像に基づき推定を行っており，講義で PC を使う場合など手元が隠れる写真などに対応できない問題やスマホを操作しているクラスが 1 つのため，様々な姿勢でスマホを操作している場合，識別できない可能性がある．本研究では，後方から受講状況を撮影した画像に基づき識別を行う．これにより，手元や正面の顔を撮影せず受講状況を識別する．

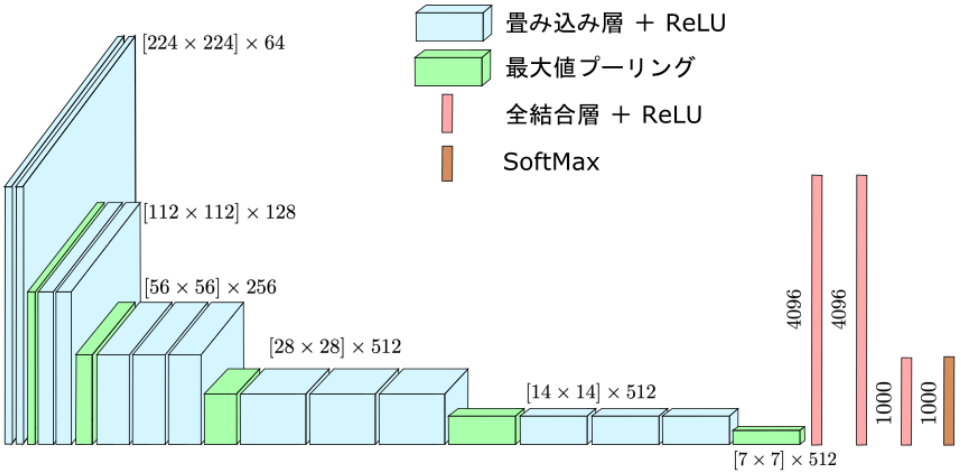
2.2. VGG16

本研究では，畳み込みニューラルネットワークの学習済みモデルである VGG16 を用いて講義中の学生の受講状況の識別を行う．具体的には受講状況を，正常(姿勢を正し聞いている)，机の下でスマホを操作，脇を開けてのスマホ操作，脇を閉じてのスマホ操作，突っ伏し，俯いて寝ている，上向きで寝ている状態の 7 種類に分け，VGG16 をファインチューニングしたモデルを用いて識別する．VGG16 とは，2014 年にオックスフォード大学の Visual Geometry Group が発表した学習済みモデルであり，図 1 に示す通り，畳み込み層 13 層，全結合層 3 層の深さ 16 層で構成されたモデルである．ImageNet と呼ばれる大規模データベースで学習され，出力層のカテゴリが 1000 種類あることから高い判別精度と特徴抽出が可能であり，2014 年の ILSVRC で準優勝を獲得している．

VGG16 の特徴として，3×3 の畳み込みフィルタを使用している．図 2 に示すとおり，3×3 のフィルタを何層も積み重ねることで，7×7，5×5 の畳み込み層の受容野と同じ受容野を持つことができる．また，畳み込み層を何層も積み重ね，学習することにより，細かい特徴を抽出しやすく，小さい畳み込みフィルタを重ねることで学生の体の姿勢の小さな違いを特徴として抽出す

るのに優れていると考える．

このモデルを使用しファインチューニングを行うことで，VGG16 の特徴抽出を他の学習に応用させることが出来る．具体的には，VGG16 の 15 層以降のみの再学習を行い，浅い層の重みを固定する．VGG16 は，浅い層では画像の輪郭や線などのおおよその特徴を抽出し，深い層は画像特有の特徴を抽出する．したがって，浅い層の特徴抽出力を再利用し，新たに受講状況の特徴を抽出させるために，15 層以降を再学習する．



出典：VGGNet: 初期の定番 CNN, CVML エキスパートガイド

https://cvml-expertguide.net/terms/dl/cnn/cnn-backbone/vggnet/

図 1　VGG16

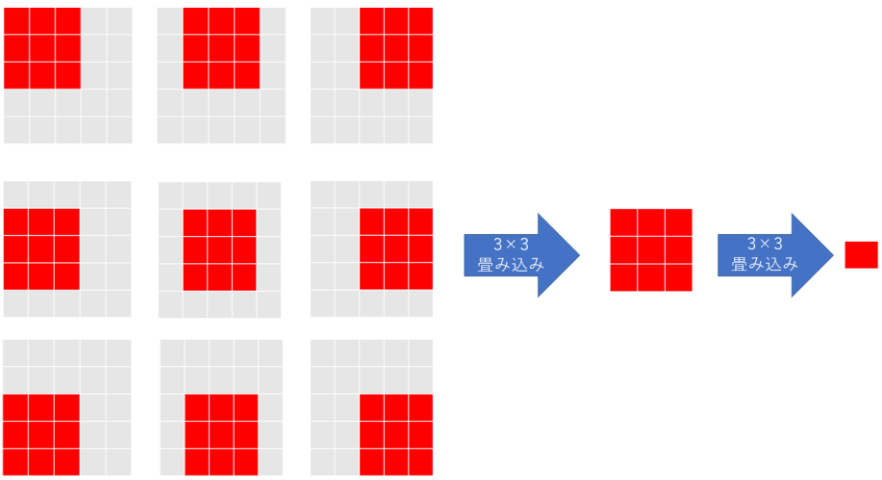


図 2　畳み込み層

3. VGG16 を利用した講義中の学生の受講状況の識別

3.1. モデル構成

本モデルの概要を図 3 に示す．図において，赤文字は VGG16 の層，水色の背景は学習させない層を表す．ここでは，VGG16 の入力層を改良し，最初の畳み込み層の前にデータ拡張層と正規化層を挿入する．また，正規化層では画素値を 0～1 の範囲にスケーリングする．更に，VGG16 の最後の畳込み層のみを再学習させ，その層からの出力を全結合層に入力するため，平坦化層を挿入する．過学習を防止するため，50%の割合でノードを不活性化させるドロップアウト層と，その後に全結合層を挿入する．最後に，再度ドロップアウト層と出力層を挿入したネットワーク構築とする．ここで，データ拡張とドロップアウト層がどのようなものであるか説明する．

3.1.1. データ拡張

本実験では訓練データが少ないため，データ拡張を行う．本実験ではデータ拡張層で，図 4 のような拡張率 1.1，縮小率 0.5 の間で拡大縮小，図 5 のような半分の確率で左右反転，図 6 のような縦方向画素数の 10%横方向画素数の 10%を移動の最大値とした平行移動，図 7 のような±45°の範囲で回転をランダムに行い，データセットを拡張する．各図の左上は拡張元のデータである．

3.1.2. ドロップアウト層

ドロップアウト層の説明に先立って過学習について説明する．過学習とは，学習時に利用したデータのみに過剰に適合してしまうために，汎化性能が失われてしまう現象である．過学習が発生してしまうと，学習プロセスで取得したデータに過剰にフィットしてしまうため，未知のデータに対する識別精度が低下してしまう．

この問題の解決策の一つとして，ドロップアウト層が存在する．ドロップアウトとは，学習時に一定割合のノードを無効化させて学習を行い，次の更新では別のノードを無効化して学習することを繰り返す．これによって学習時にネットワーク構造の自由度を強制的に小さくして汎化性能を向上させ過学習を抑制する．

3.2. 使用データ

本研究では，大学生 6 人を基にデータを収集した．図 8 の番号順にそれぞれの席で 7 種類の受講状況を撮影した．撮影した動画から，図 9 のような各受講状況を手動で切り出した画像データと，図 10 のような Yolo の姿勢推定モデルを用いて推定した骨格をプロットした画像を用いて，Yolo の姿勢推定モデルによって切り出した画像データの 2 パターンを用いる．手動で切り出した写真データを学習用 244 枚，テスト用 87 枚に分けて用いた．Yolo モデルを用いた写真データを学習用 228 枚，テスト用 83 枚に分けて用いた．カメラから番号 2 の席までの距離は 4m で高さ 3m の地点から撮影した．使用したカメラは iPhone14ProMAX である．

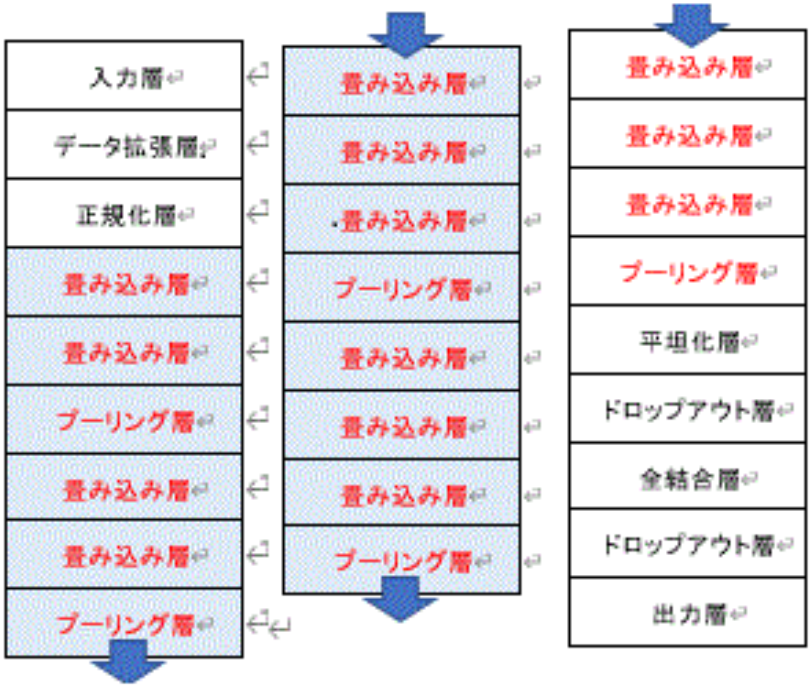


図 3　本実験のモデル構造

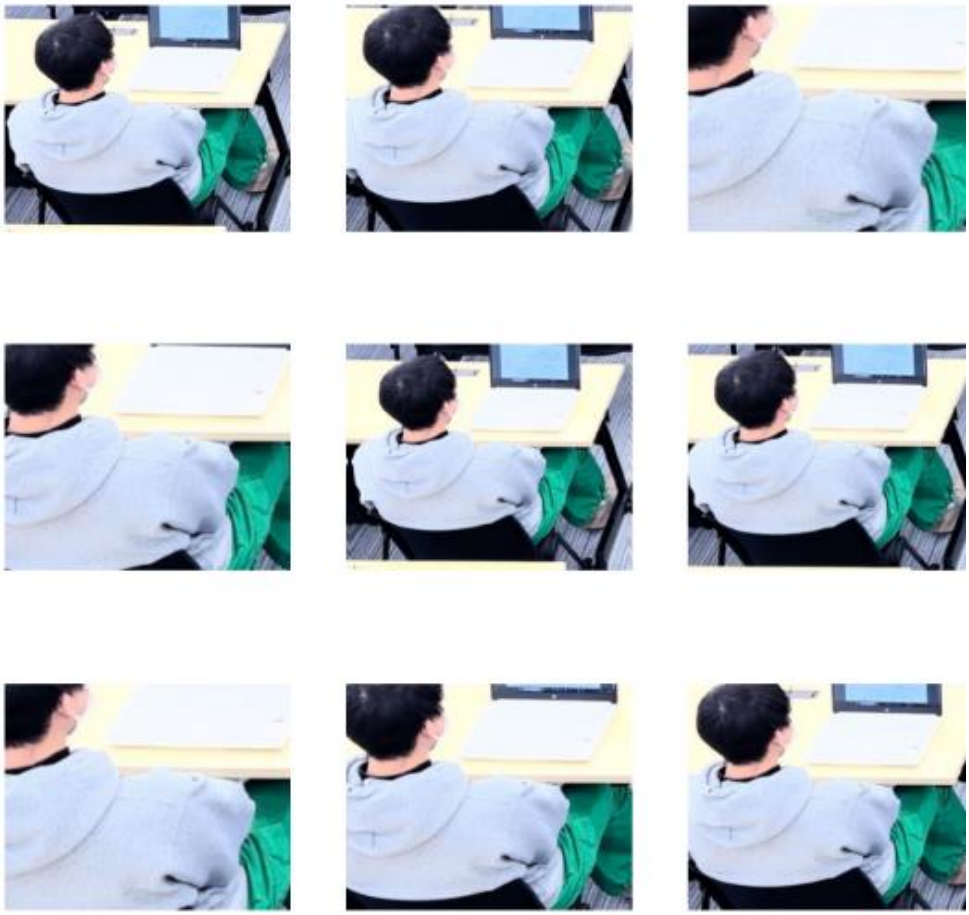


図 4　拡大縮小後のデータ例

図 5　左右反転後のデータ例

図 6　平行移動後のデータ例

図 7　回転後のデータ例

図 8　撮影時の状況

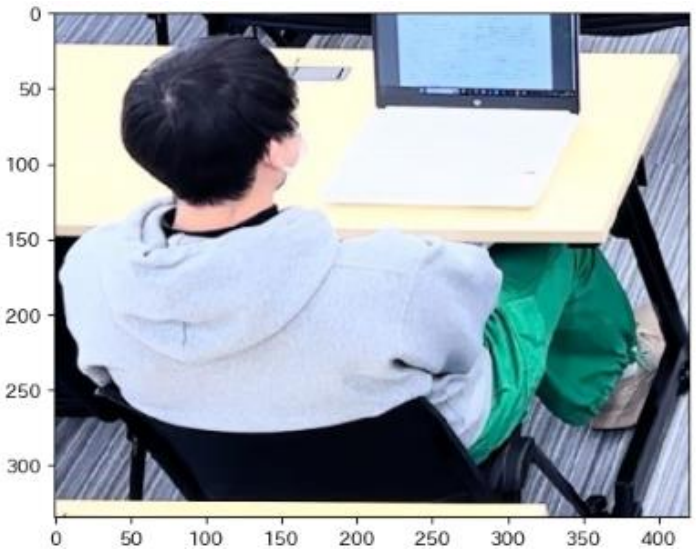


図 9　手動でのデータ例

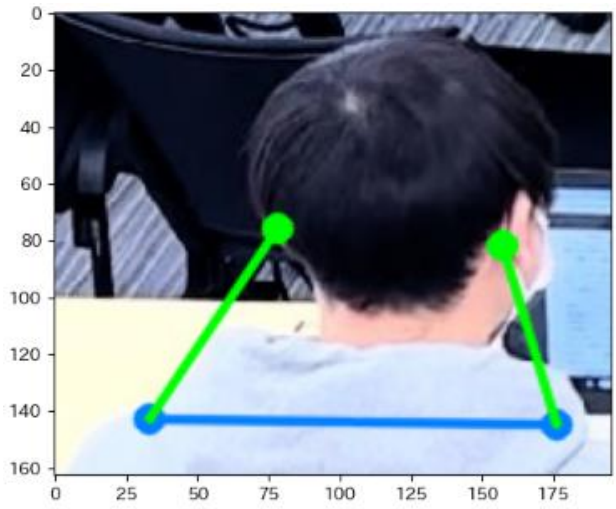
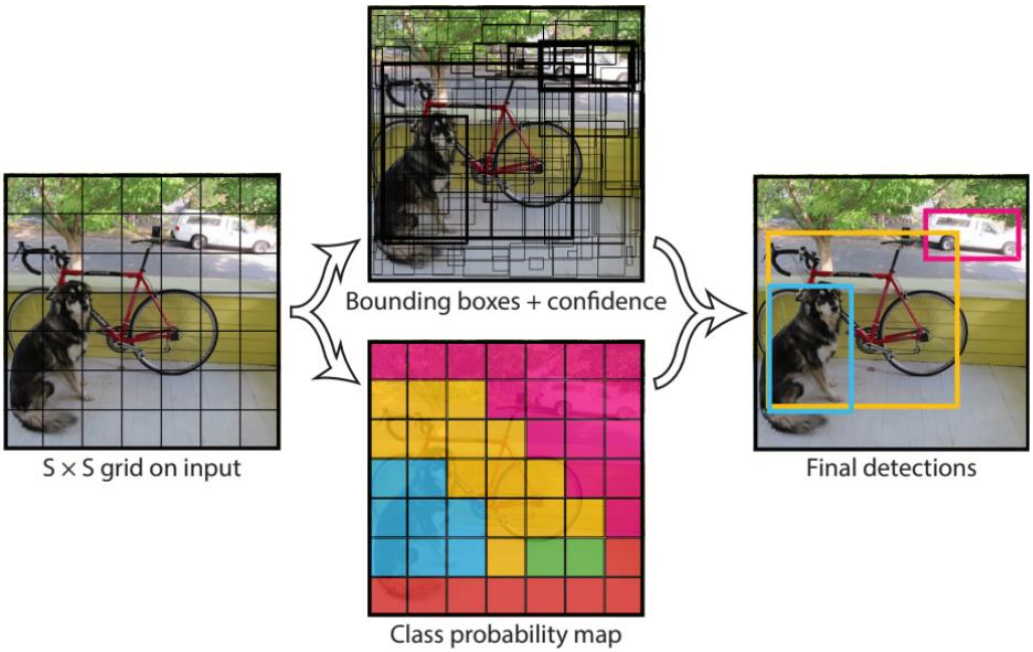


図 10　Yolo を用いたデータ例

3.2.1. Yolo

YOLO(You Only Look Once)は，画像や動画などに映っている人や車などの物体を検出するアルゴリズムの一つであり，他の物体検出アルゴリズムより比べ識別する処理が早いという特徴を持っている．Yolo は図 11 のような物体検出を行っている．画像を 7×7 個で正方形に分割し，各正方形の中で物体か背景かの識別する処理を行い，枠線が太いほど物体であるという確率が高くなるという処理と各正方形がどの物体に所属しているかの判断をして，得られた結果をもとに物体認識を行っている．

本実験では Ultralytics 社が公開した YOLOv8 の物体検出モデルをベースに，物体とそのキーポイントの推定を行うことができるようにしたモデルの yolov8-pose を用いる．また，将来的に複数人を安価かつリアルタイムで識別することを目指すため，表 1 から比較的処理速度が速い yolov8s-pose を用いた．骨格を画像の切り出しに yolo のモデルを用いるのも同様の理由である．



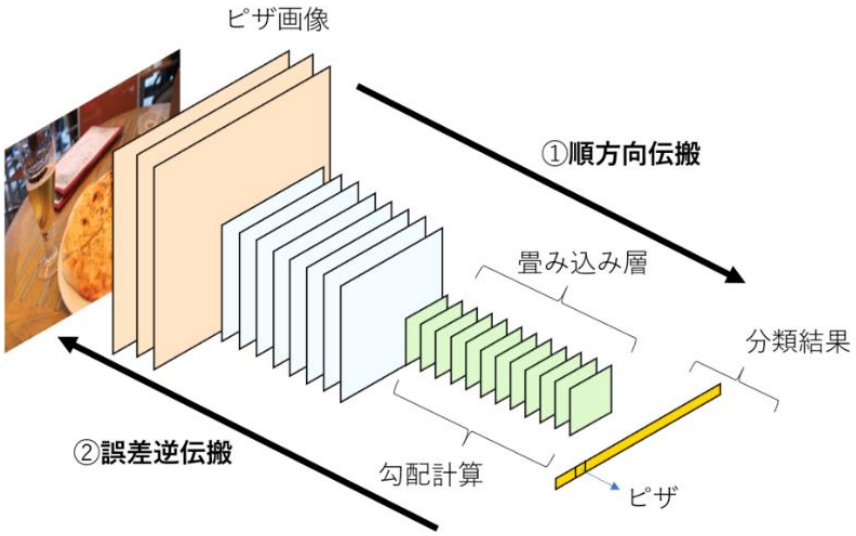
出典：株式会社マクニカ，物体検出と Deep Learning ～ 入門から応用まで ～

https://www.cvfoundation.org/openaccess/content_cvpr_2016/html/Redmon_You_Only_Look_CVPR_2016_paper.html

図 11　YOLO の物体検出

3.3. Grad-CAM

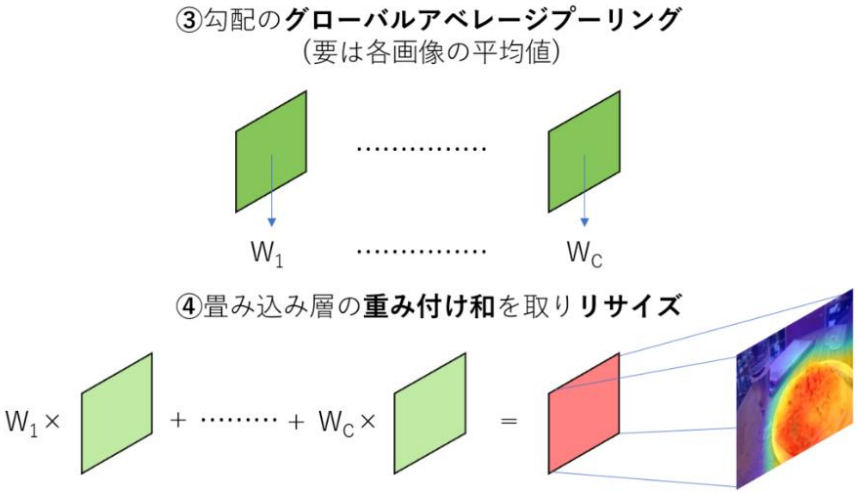
本研究では，Grad-CAM(Gradient-weighted Class Activation Mapping)を適用し，モデルが識別の際どこに注目しているか可視化する[4]．Grad-CAM は予測値に対する勾配を重み付けすることで，重要なピクセルを可視化する技術で，図 12,13 のようにする[5]．まずは画像を入力し，CNN を通してクラス分類を行う．その際に得られる畳み込み層の出力と，クラス分類の出力を取り出す．また，順方向計算の後に誤差逆伝搬を行い，畳み込み層の各要素に対するクラス分類出力の勾配を計算する．次にそれぞれの畳み込み層に対して各畳み込み層の勾配の平均値のグローバルアベレージプーリング(GAP)を取る．得られる値はスカラー値なので，これを元の順方向で得られた畳み込み層の出力に重み付し，全ての畳み込み層で加算することで 1 枚の画像が得る．ここに活性化関数 Relu を挟み，小さな畳み込み層の出力をリサイズすることで Grad-CAM の画像が得られる．ただし，本研究では上記モデルの畳み込み層の出力を得られなかったため，上記モデルの学習時の重みを保存し，同じ条件のモデルを作り，それに重みを読み込ませたものの出力を使用する．



出典:ころがる狸のデータ解析ブログ，“【CNN+Grad-CAM】仕組みの解説と画像の予測根拠可視化”

https://dajiro.com/entry/2020/06/26/234720#google_vignette

図 12　Grad-CAM①



出典:ころがる狸のデータ解析ブログ，“【CNN+Grad-CAM】仕組みの解説と画像の予測根拠可視化”

https://dajiro.com/entry/2020/06/26/234720#google_vignette

図 13　Grad-CAM②

表 1　YOLOv8-pose モデル一覧

Model	size (pixels)	mAP^{pose} 50-95	mAP^{pose} 50	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n-pose	640	50.4	80.1	131.8	1.18	3.3	9.2
YOLOv8s-pose	640	60.0	86.2	233.2	1.42	11.6	30.2
YOLOv8m-pose	640	65.0	88.8	456.3	2.00	26.4	81.0
YOLOv8l-pose	640	67.6	90.0	784.5	2.59	44.4	168.6
YOLOv8x-pose	640	69.2	90.2	1607.1	3.73	69.4	263.2
YOLOv8x-pose-p6	1280	71.6	91.2	4088.7	10.04	99.1	1066.4

出典:Qiita，【Python】最新物体検知 AI YOLOv8 の Python ライブラリ ultralytics がすごすぎる！

https://qiita.com/Mikeinu/items/530bdb2ddeeedc32eb58

4. モデルの評価実験

4.1. 実験方法

実験においてテスト用データに対する予測を行う．実験では，モデル評価では学習率を 0.001，エポック数を 800 とし，入力画像サイズは画像によってさまざまだが，入力層に入力する際に 224×224pixel に変形する．各データでの学習，テスト用や各状態の枚数の内訳を表 2，3 に示す．損失関数は多クラス分類なので Sparse Categorical Crossentropy を用いた．各データの正解ラベルだと予測された割合の自然対数にマイナスをかけたものの平均である．

4.2. 手動で切り出した画像の識別

手動で切り出した画像の識別において，正解率は，学習用データでは 99%に達したが，テストデータでは約 83.9%であった．図 14 と図 15 に学習時の損失と識別精度を示す．エポック数の増加に伴い，トレーニングとテストの識別精度が増加している．一方，損失はドロップアウトを使用したがトレーニングデータの損失は減少しているが，検証データの損失は増加しており，学習過程において過学習が起きていることが示された．

エポック数の増加に伴い，トレーニングとテストの識別精度が上昇し，損失は減少している．一方，エポック数が 60 を過ぎたあたりからトレーニングの正解率がテストのものを上回り，逆に損失は下回ることが続き，十分に学習されたことがわかった．

4.3. Yolo モデルを用いて切り出した画像の識別

Yolo モデルを用いて切り出した画像の識別において，正解率は，学習用データでは 99%に達したが，テストデータでは約 81.9%であり，手動で切り出した画像よりも性能が低下した．これは Yolo モデルが人と判別して切り出す範囲が手動と比べて幅があることで予測が難化したことが考えられる．図 16 と図 17 に学習時の損失と識別精度を示す．エポック数の増加に伴い，トレーニングと検証データの識別精度が増加している．一方，損失はドロップアウトを使用したがトレーニングデータの損失は減少しているが，検証データの損失は増加しており，学習過程において過学習が起きていることが示された．

表 2　手動で切り出したデータ数

	学習用	テスト用
正常	36 枚	12 枚
机の下でスマホを操作	36 枚	12 枚
脇を開けてのスマホ操作	36 枚	12 枚
脇を閉じてのスマホ操作	36 枚	12 枚
突っ伏し	36 枚	12 枚
俯いて寝ている	34 枚	12 枚
上向きで寝ている	30 枚	15 枚

表 3　Yolo を用いたデータ数

	学習用	テスト用
正常	36 枚	12 枚
机の下でスマホを操作	35 枚	12 枚
脇を開けてのスマホ操作	36 枚	12 枚
脇を閉じてのスマホ操作	36 枚	12 枚
突っ伏し	29 枚	11 枚
俯いて寝ている	34 枚	12 枚
上向きで寝ている	22 枚	13 枚

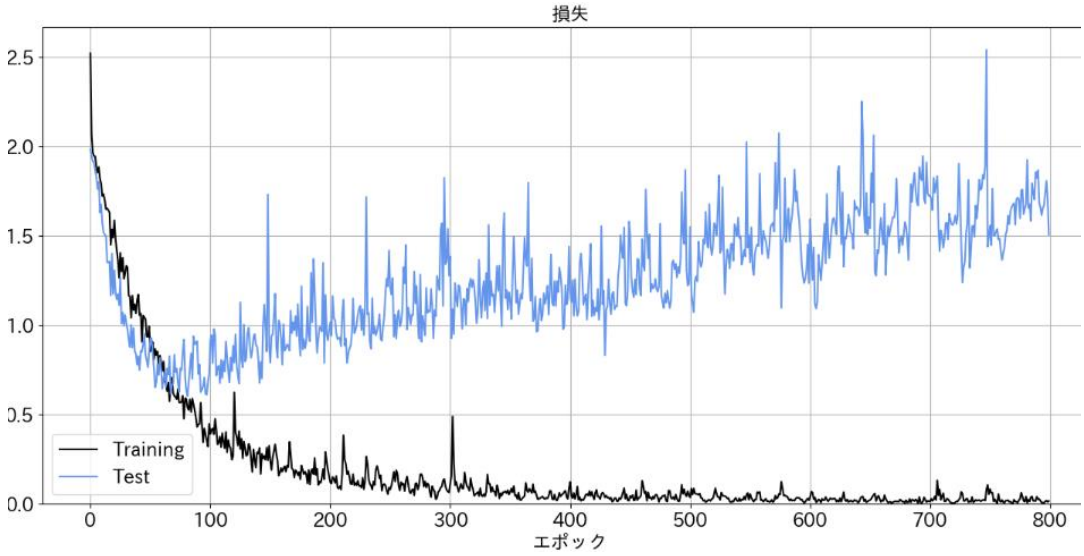


図 14　学習時の損失

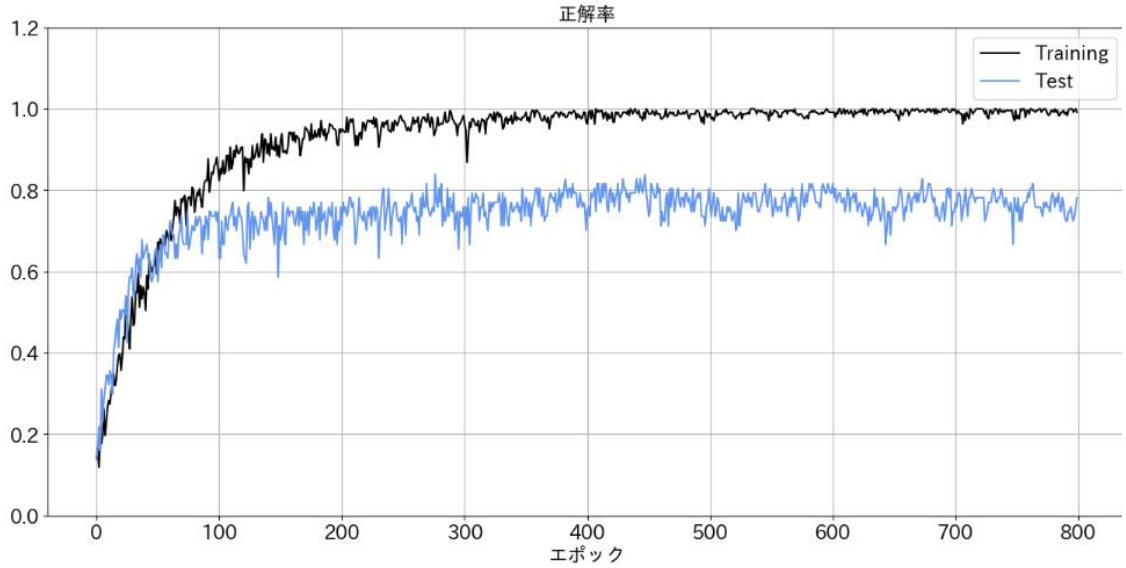


図 15　学習時の正解率

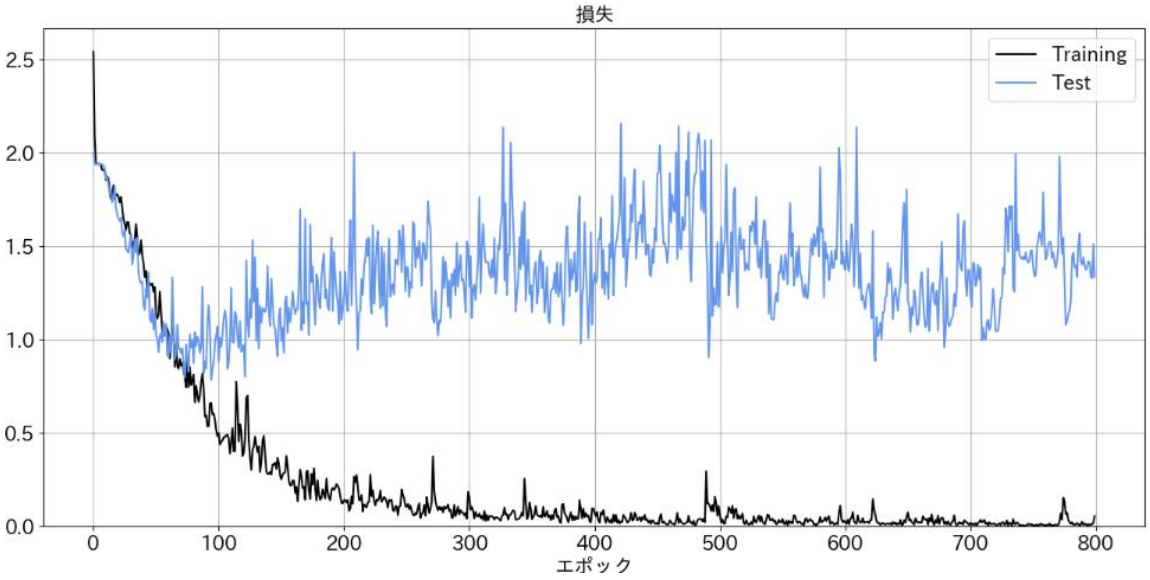


図 16　学習時の損失

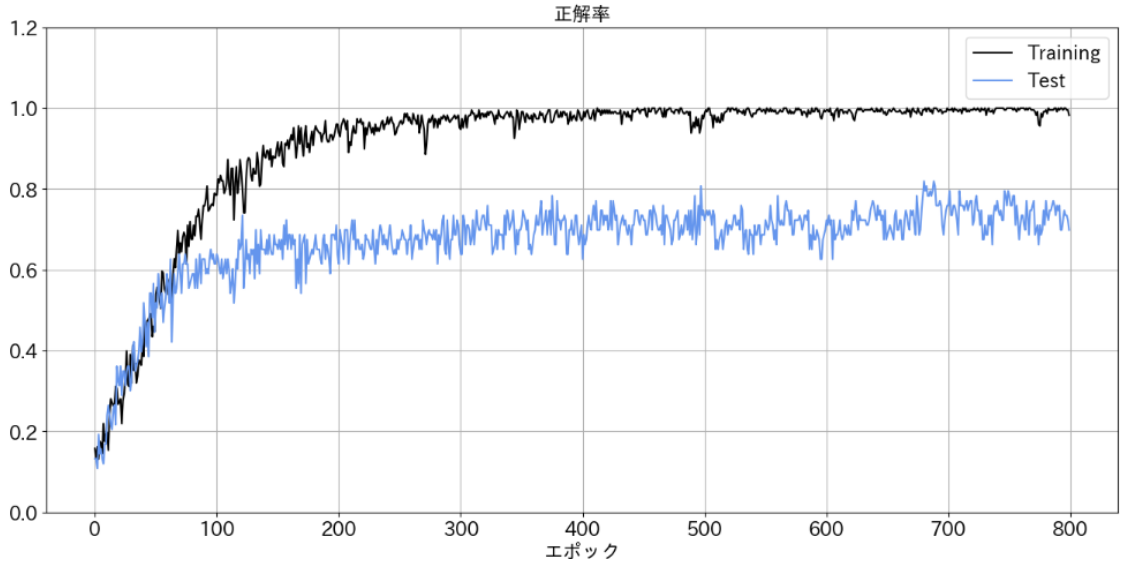


図 17　学習時の正解率

4.4.　手動で切り出した画像の識別での Grad-CAM の適用結果

手動で切り出した画像の識別について，学習率を 0.001，エポック数は過学習を起こしたとみられる直前の 80 としたとき，テストデータの正解率は約 71.3%，損失は約 0.63 であった．図 18 と図 19 に学習時の損失と識別精度を示す．

受講態度別にみると，正常(姿勢を正し聞いている）や突っ伏し，上向きで寝ている状態は概ね正しく識別できた．しかしながら，机の下でスマホを操作している状態と俯いて寝ている状態，脇を開けてのスマホ操作か脇を閉じてのスマホ操作は姿勢が類似しているため，誤判別される場合が確認された．表 4 に受講態度別の正解率を示す．

その学習時の重みを読み込ませた同じ条件のモデルの出力をもとにした Grad-CAM の適用結果例を図 20 に示す．図における画像の配置は図 8 の座席の位置と同じである．ラベル 0 が正常(姿勢を正し聞いている)，1 が机の下でスマホを操作，2 が脇を開けてのスマホ操作，3 が脇を閉じてのスマホ操作，4 が突っ伏し，5 が俯いて寝ている，6 が上向きで寝ている状態である．図において色が赤に近い程注視している範囲を示している．概ね左に座る人は体の右，右に座る人は体の左に注目していることが分かった．しかし，背景に注目したものもあり，修正が必要である．また，正解率と損失，受講態度別の正解率はクラス分類の出力から手入力で集計した．正解率は約 71.3%，損失は約 0.61 であった．表 5 に受講態度別の正解率を示す．

受講態度別にみると，正常は頭，机の下でスマホを操作している状態は頭や首，脇を開けてのスマホ操作はわき腹や頭の下の方，脇を閉じてのスマホ操作は脇や背中，突っ伏しは脇腹や首，俯いて寝ている状態は頭や首，上向きで寝ている状態はつむじに注目していることが分かった．表 3 に受講態度別の正解率を示す．

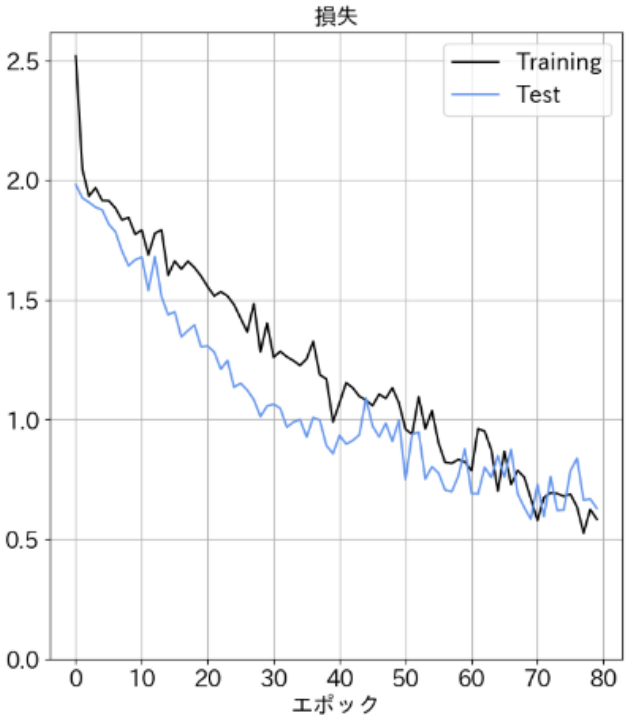


図 18　学習時の損失

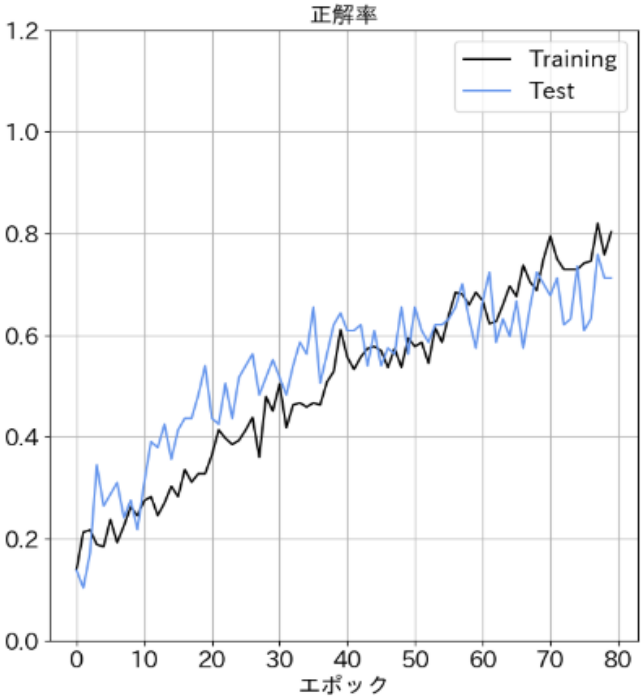


図 19　学習時の識別精度

表 4　正解率

	正解率
正常	83.3%
机の下でスマホを操作	41.7%
脇を開けてのスマホ操作	50.0%
脇を閉じてのスマホ操作	75.0%
突っ伏し	100%
俯いて寝ている	50.0%
上向きで寝ている	93.3%

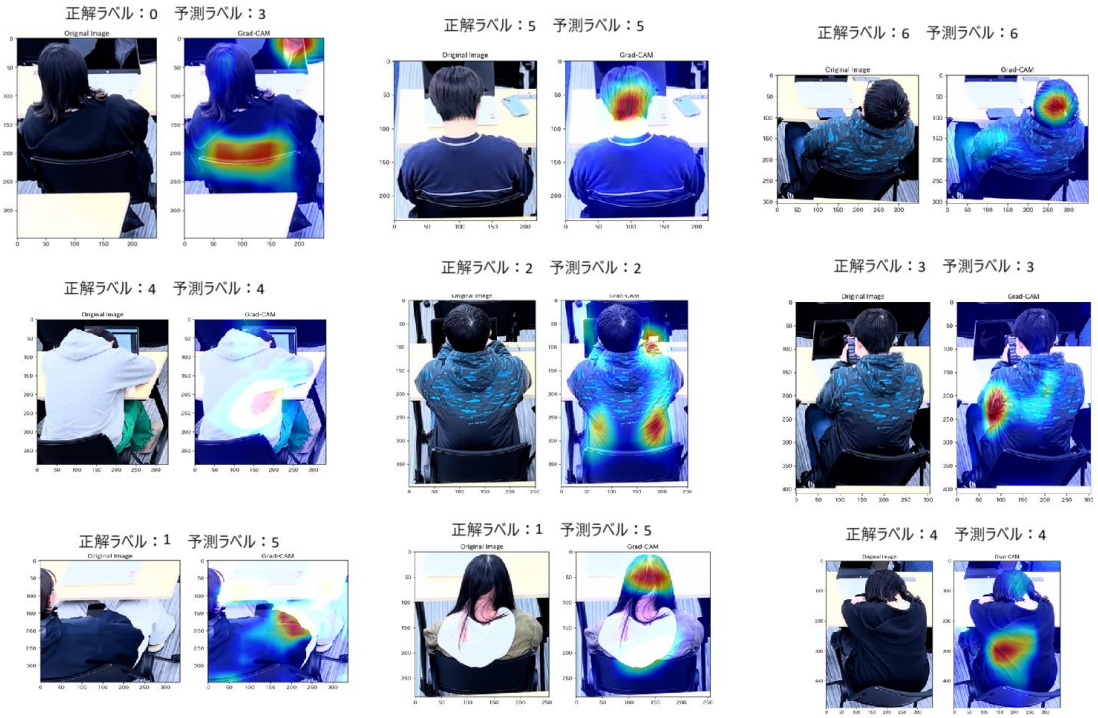


図 20　Grad-CAM の結果例

表 5　正解率

	正解率
正常	83.3%
机の下でスマホを操作	41.7%
脇を開けてのスマホ操作	50.0%
脇を閉じてのスマホ操作	75.0%
突っ伏し	100%
俯いて寝ている	50.0%
上向きで寝ている	93.3%

4.5.　Yolo モデルを用いて切り出した画像の識別の Grad-CAM の評価

Yolo モデルを用いて切り出した画像の識別について，学習率を 0.001，エポック数は過学習を

起こしたとみられる直前の 80 としたとき，テストデータの正解率は約 59.0%，損失は約 0.92 であった．図 21 と図 22 に学習時の損失と識別精度を示す．

受講態度別にみると，突っ伏し，机の下でスマホを操作している状態，上向きで寝ている状態は概ね正しく，正常はある程度識別できた．しかしながら，俯いて寝ている状態が机の下でスマホを操作している状態と，また，脇を閉じてのスマホ操作は脇を開けてのスマホ操作とが同じ姿勢だと誤判別される場合が多く確認された．表 6 に受講態度別の正解率を示す．

その学習時の重みを読み込ませた同じ条件のモデルの出力をもとにした Grad-CAM の適用結果例を図 23 に示す．図における画像の配置は図 8 の座席の位置と同じである．ラベル 0 が正常(姿勢を正し聞いている)，1 が机の下でスマホを操作，2 が脇を開けてのスマホ操作，3 が脇を閉じてのスマホ操作，4 が突っ伏し，5 が俯いて寝ている，6 が上向きで寝ている状態である．図において色が赤に近い程注視している範囲を示している．また，正解率と損失，受講態度別の正解率はクラス分類の出力から手入力で集計した．正解率は約 63.9%，損失は約 0.92 であった．表 7 に受講態度別の正解率を示す．手動と同じく概ね左に座る人は体の右，右に座る人は体の左に注目していることが分かった．しかし，背景に注目したものもあり，修正が必要である．

受講態度別にみると，正常は頭の付け根，机の下でスマホを操作している状態は首回り，脇を開けてのスマホ操作は背中からわき腹や首，脇を閉じてのスマホ操作は背中からわき腹，突っ伏しは背中や頭，俯いて寝ている状態は首回り，上向きで寝ている状態は頭の前の方に注目していることが分かった． 手動のものと比べて骨格の線で囲まれた背中や両耳と肩にひかれた線の中の頭や首に注目するものが増えたが，それがかえって偏った識別を招いていることも考えられる．

図 21　学習時の損失

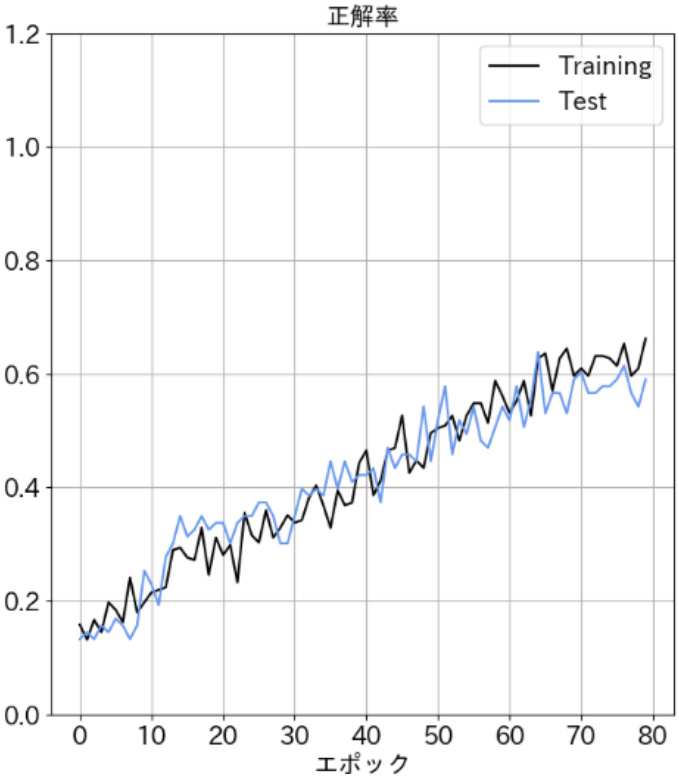


図 22　学習時の正解率

表 6　正解率

	正解率
正常	66.7%
机の下でスマホを操作	81.8%
脇を開けてのスマホ操作	41.7%
脇を閉じてのスマホ操作	16.7%
突っ伏し	100%
俯いて寝ている	16.7%
上向きで寝ている	92.3%

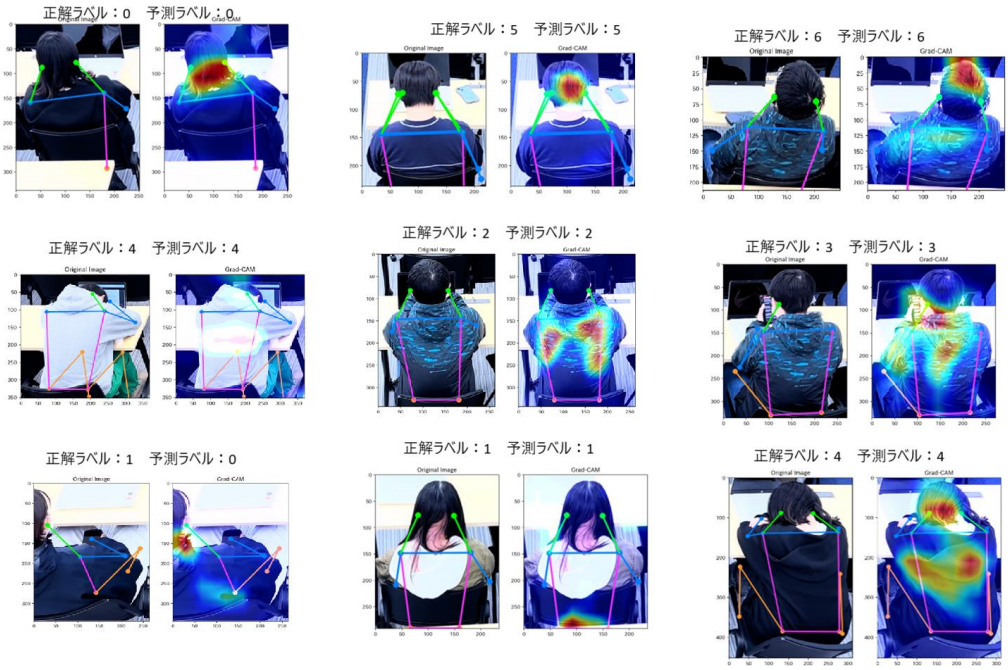


図 23　Grad-CAM の結果例

表 7　正解率

	正解率
正常	83.3%
机の下でスマホを操作	81.8%
脇を開けてのスマホ操作	41.7%
脇を閉じてのスマホ操作	25.0%
突っ伏し	100%
俯いて寝ている	25.0%
上向きで寝ている	92.3%

5. おわりに

本論文では，VGG16 を利用した講義中の学生の受講状況の識別を提案した．ここでは，教室後方から撮影した学生の状態画像から 7 状態を識別して，VGG16 が画像のどこに注目し，識別を行っているのか可視化した．その結果，今回の 7 種類の講義状況の識別においては手動で切り出した画像での識別のほうが YOLO を用いたものより有効であることを確認した．また，Grad-CAM により，頭や脇の周辺などそれぞれの講義情報で特徴的な箇所に注目していることを確認した．

今後の課題として，今回は高さ 3m から撮影したため，ほかの高さではどういった結果になるのか，背景にモデルが注目することがあり，データ拡張の内容や背景の削除などの検討，複数の学生が近接したり重なったりする場合の識別方法をどうするか考えることが挙げられる．

さらに，将来的に実際の教室にカメラを設置する際，推論をエッジデバイスで行うことが考えられる．そのため，本実験のような大きいモデルの予測結果と正解データを教師データとして小さいモデルの学習に利用する蒸留という手法を用いて，大きいモデルに匹敵する精度を持つ小さいモデルを作ることでモデルを軽量化し，高性能な GPU を使わずエッジデバイスのみで識別可能にする必要があると考える．

参考文献

[1] 稲葉利江子，酒井博之，辻靖彦，平岡斉士，重田勝介：“大学における ICT 環境の規模別導入状況の現状と経年変化”，大学 ICT 推進協議会 2021 年次大会論文集，pp. 307-312 (2021)

[2] Ashwin，R.：“AttenFace: A Real Time Attendance System using Face Recognition”，2022 IEEE 6th Conference on Information and Communication Technology (CICT)，pp. 549-553 (2022)

[3] 吉武春光：“Fine Tuning による学生の受講態度の推定”， 西南学院大学商学論集，pp. 199-211 (2023)

[4] Selvarajul，R. R.，Cogswell，M.，Das，A.，Vedantam，R.，Parikh，D.， and Batra，D.：“Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization”，2017 IEEE International Conference on Computer Vision，pp. 618-626 (2017)

[5] ころがる狸のデータ解析ブログ “【CNN+Grad-CAM】仕組みの解説と画像の予測根拠可視化” https://www.jaic-g.com/news/pressrelease/news-2380/（参照:2025 年 2 月 3 日）