Vol.2023-CVIM-232 No.27
2023/1/27

点群時系列データからの老化に伴う運動機能低下の検知

槌道 慎也[1,a)]　青木 工太[1,b)]　槇原 靖[1,c)]　八木 康史[1,d)]

概要：高齢化が進む社会では介護を必要とする人の増加が問題となっており，骨折・転倒は高齢者が介護を必要とする大きな要因の一つである．転倒事故は老化に伴う運動機能の低下によって発生しやすくなり，より重大なケガに繋がるようになることから，運動機能の改善によって健康寿命を延ばすことが重要である．そのため，転倒リスクの早期検出を含めた，高齢者見守りのためのセンサー機器等を用いた情報通信技術の開発が期待されている．一方で，一般的なカメラを用いた見守りは嫌悪感を持たれる場合が多く，プライバシーに配慮する必要がある．本研究では，3 次元点群データから転倒リスク推定を行う機械学習手法を提案する．転倒リスクは TUG (Timed up & Go Test) に基づいて定義し，TUG を真横から深度カメラで撮影したデータを用いる．転倒リスクを推定する分類器への入力は，時系列点群から切り出した起立中もしくは着席中の 1 秒間の動作データである．70 歳以上の高齢者 320 名からデータを収集し，提案手法の有効性を検証するための実験を行った．

キーワード：高齢者見守りシステム, 転倒リスク推定, 深度画像, 深層学習

1. はじめに

近年，医療の発展により平均寿命が延び続けると共に，健康上の問題で日常生活が制限されることなく生活できる期間である健康寿命を延ばすことについて関心が高まっている．高齢社会白書 [1] によると，現在，日本の平均寿命は男性 81.56 年，女性 87.71 年であるのに対して，健康寿命は男性 72.68 年，女性 75.38 年と 10 年ほどの差がある．また，高齢者が介護を必要とするようになった主な原因は認知症，脳卒中，高齢による衰弱，骨折・転倒，関節疾患であり，衰弱，骨折・転倒，関節疾患が原因で介護を必要とするようになった人の割合は合計で 37.3%であった．この衰弱，骨折・転倒，関節疾患はいずれも老化による筋力やバランス感覚といった運動機能の低下に伴って発症しやすくなり，加えて，転倒事故は年を重ねるにつれて筋力が衰えることでケガが重症化しやすく，けがによる入院生活や安静は筋力や身体機能のさらなる衰えを引き起こし，転倒事故の危険性はより高まる [2], [3]．このような負のループに入らないためには転倒リスクの高まりをいち早く検知して，運動やリハビリを行うことで健康寿命を延ばすことが重要である．これによって介護を必要とする人の数が減ることで，社会問題の 1 つである介護の負担を減らすことにつながると考えられる．

高齢者の転倒事故を事前に予測するものとして転倒アセスメントスコアシートが利用されている [4], [5]．これは，対象者の状態がスコアシートの各項目に当てはまるかをチェックすることでリスクの推定を行うものである．しかし，このスコアシートは対象者の状態をよく知る医療従事者や介護者でなければ評価を行うことができず，定期的な評価の更新は業務の増加につながると考えられる．

こうした問題に対して，自動で転倒リスクを推定する研究が行われている．深層学習を用いて転倒リスクの高い患者とそうでない患者を識別し，院内の環境から転倒リスクを高めている要因をリアルタイムで通知することにより，病院の人員といったリソースを効率よく運用するためのシステムが提案されている [6]．また，人工知能を用いて電子カルテからリスクの推測を行う転倒転落予測システムのサービス [7] が提供されており，このサービスの AI は医療従事者が診断した場合と同程度の精度でスコアを算出し，業務の効率化や定期的なスコアの更新を目指している．さらに，運動機能を評価するためのテストとして定められた動作を行い，それにかかる時間を計測する手法 [8] や，身に着けた加速度センサからの情報をもとに転倒リスクを推定する手法 [9] などが提案されている．しかし，これらの手法は定期的な通院や，数十秒かかるテスト，センサの着脱など，生活への負担に繋がることが懸念される．

[1] 大阪大学
University of Osaka, Osaka, Japan
[a)] tsuchimichi@am.sanken.osaka-u.ac.jp
[b)] aoki.k@am.sanken.osaka-u.ac.jp
[c)] makihara@am.sanken.osaka-u.ac.jp
[d)] yagi@am.sanken.osaka-u.ac.jp

ⓒ 2023 Information Processing Society of Japan

そこで本研究では，深度カメラを用いて日常生活を撮影し，その映像から自動で転倒リスクを推定するための手法を提案する．カメラが取得した画像による推定はカメラの設置以外の特別な操作は必要ない．また，深度画像は被写体までの距離を撮影するため，RGB 画像と比較して被写体の顔の特定が難しいことから，プライバシーの観点で対象者が感じる嫌悪感が少ない．日常生活では様々な動作が行われるが，特に起立と着席を対象とする．これらの動作は椅子やベッドといったある程度位置が決まっている家具に関連して行われるため，カメラに写る対象者の位置や向きの変動が少ない．70 歳以上の高齢者を対象としてデータ収集を行い，時系列点群データの動作認識で用いられる 3D Dynamic Voxel (3DV) と点群に対する学習ネットワークの PointNet++を組み合わせた手法 [10] を用いて，起立・着席の動作からの転倒リスク推定の有効性を検証する．

2. 関連研究

2.1 高齢者の転倒リスク推定

高齢者の増加に伴って高齢者の転倒事故も増加しており，転倒事故を検知する，あるいは転倒リスクを推定する研究が数多く行われている．カメラからの画像を基に転倒事故を検知し，通知するサービス [11], [12] はすでに実用化されているが，転倒事故の自動検知は素早い救護に繋がるものの，転倒事故の減少には繋がりにくい．転倒リスクの推定手法として，加速度センサを用いた手法 [9] や，医療カルテやバイタルサイン [13] を用いた手法が存在する．しかし，センサの着脱を日常的に行うことは被験者にとって負担であることや，機器の付け忘れも起こりうることに加えて，医療カルテを基にする場合，病院での診察を受けない限り情報が更新されず，リアルタイム性に欠けることからリスクの高まりを見逃す可能性がある．自動かつ定期的に転倒リスクの推定が可能で，対象者に負担がかからない転倒リスクの推定手法として，家に設置されたセンサやデバイスから対象の行動を識別し，動作の種類，動き方，位置をもとに転倒リスクを推定する手法が提案されている [14]．しかし，この手法は予め複数の機器を家に設置する必要があることや，対象者が加速度センサのデバイスを装着する必要がある．また，日常生活動作をセンサから取得していることからプライバシーへの配慮が足りないことも考えられる．よって本研究では，1 つの深度センサを用いて特定の位置での起立・着座のみに注目し，その情報のみから転倒リスクの推定を行う．

2.2 3 次元空間上の動作認識

動作認識とは撮影した映像に写る被写体が行っている動作を推測するクラス分類問題であり，3 次元データでの動作認識は主に骨格ベースの手法と深度ベースの手法がある．骨格データは各関節の座標と，それらを繋ぐノードから構成されており，Graph Convolutional Networks を用いた推定手法が多く提案されている [15], [16], [17]．骨格ベースの動作認識は精度が高い傾向がある一方で，特に人を横視点から撮影した際，左右の手足が入れ替わって推定されたり，遮蔽された関節が推定できないなど、安定した関節座標値の取得が難しい場合がある．そのため，より実用的な手法として，深度センサから得られるデータを 3 次元動作認識に直接利用する深度ベースの動作認識の研究が行われている．深度センサを用いてカメラから被写体までの距離を撮影することで得られる深度画像は，被写体の形状を点群という形で 3 次元空間上への再構成が可能である．深度データをそのまま利用する例として，深度画像をそのまま用いる手法 [18], [19]，点群に変換する手法，点群からボクセルに変換する手法 [10] が存在する．特に 3DV と PointNet++を用いた Wang らの手法は，深度ベースでありながらも高い動作認識の精度を実現している．そのため，本研究では Wang らの手法を用いて，同一の動作から個人間の特徴を抽出し，転倒リスクの推定を行った．

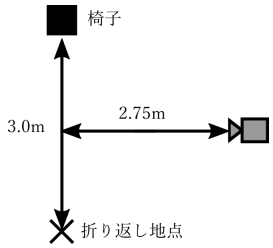


図 1: 撮影環境

3. データセット

本研究で用いるデータセットは被験者が TUG を行う様子を深度カメラで撮影した深度画像列である．本節では実験に用いたデータとアノテーションについて説明する．

3.1 TUG の概要

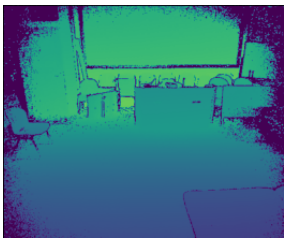
TUG とは運動機能を測るためのテストで，下肢筋力，バランス，歩行能力，易転倒性を評価することができる．椅子に座った姿勢に始まり，立ち上がって 3 m 先まで歩いて折り返し，再度椅子に座るまでの時間を計測する．これを 2 回行い，速かった方のタイムを採用する．TUG にかかった時間が 11.0 秒以上の場合，運動機能の低下による転倒の危険性が高い運動器不安定症の診断根拠の一つとなる [20]．

3.2 撮影環境

今回の実験では深度画像を解像度 512×424 で 30fps，RGB 画像を解像度 1920×1080 で 30fps で撮影できる Microsoft Kinect v2 を用いて撮影を行った．**図 1** はカメラと TUG を行う場所の位置関係を表したもので，カメラは高さ 1.6m の地点に設置されている．**図 2** は撮影された深度画像と RGB 画像の一例である．

ⓒ 2023 Information Processing Society of Japan

a RGB image



b Depth image

図 2: 撮影された画像

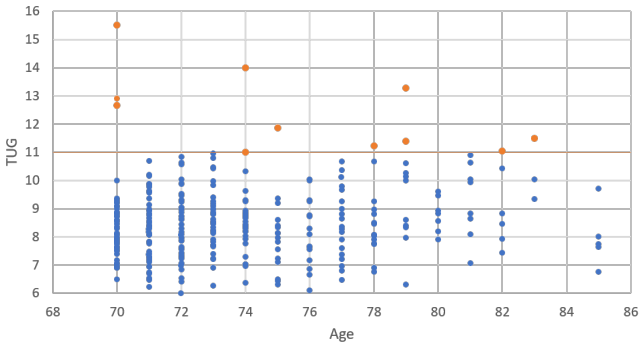


図 3: 年齢と TUG の時間の分布

3.3 アノテーション

撮影された映像全体から TUG が行われたフレームとそうでないフレームを分ける必要がある．テストの開始を被験者が立ち上がるために動き始めた瞬間，終了は被験者が座り終えて静止した瞬間と定義し，RGB 画像を目視で確認することでフレームの分類を行った．開始のフレームと終了のフレームの撮影された時間の差が TUG のタイムとなる．本研究では TUG が 11 秒以上の運動器不安定症の可能性がある状態を Positive，11 秒未満の比較的運動機能が高い状態を Negative と定義した．

図 3 は横軸に被験者の年齢，縦軸に被験者の TUG のタイムを表したものであり，各被験者ごとに 2 回計測して速かった方の TUG のデータのみをグラフにしたところ，Positive の被験者は 11 名，Negative の被験者は 309 名の計 320 名であった．また，今回は起立・着席の動作に注目して実験を行うため，その動作の部分だけのフレームを抜き出す．起立の終了は一歩目を踏み出す足が地面から離れる直前，着席の開始は最後の一歩が終わった瞬間と定義し，TUG のフレームを分類した時と同様に起立・着席のフレームとそうでないフレームを分けた．

4. 提案手法

本研究では深度画像列から，その画像列の正解ラベル (Positive, Negative) を推定する．深度画像の動作認識に用いられる 3DV と PointNet++の手法を用いて実験を行った．本章ではこれらの手法に加えて，データの前処理についての説明を行う．

4.1 前処理

撮影は複数日にまたがって行われており，カメラは完全に固定されていないため，時間によってカメラと水平面の位置関係がずれる可能性がある．そのため，すべてのデータを同じ条件で前処理を行えるように，各被験者の TUG の直前のフレームを用いて平面の位置合わせによって床と水平ベクトルを平行にした．また，**図 2b** のように撮影した深度画像には床や壁などの不要な部分が含まれており，起立と着席の動作は椅子付近でしか行われないため，椅子近辺以外の不要な点を削除する必要がある．以下に，前処理の手順を示す．

(1) TUG 開始直前のフレームから床のみの領域を切り出す．
(2) 床の点群から平面ベクトルを近似することでカメラから見た床の平面ベクトルを求める．
(3) 鉛直方向のベクトルと垂直な平面ベクトルと，2. で求めた平面ベクトルとの間にある関係（回転行列）を求める．
(4) 使用するフレームの点群と回転行列の積を計算することで，床に水平な向きから撮影した点群と同等の点群を取得する．
(5) 椅子の周囲に 8 つの基準点を定義し，その基準点から作られる直方体の外の点群は動作に関係ない点としてデータから削除する．

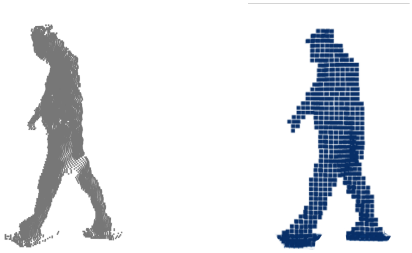
上記の手法によって，床や壁などが含まれたデータから人物と椅子のみが映った深度画像を取得した．

4.2 3D Dynamc Voxel

RGB 画像での動作認識の手法の一つに Temporal rank pooling がある．これはフレームごとに異なる重みづけを行うことでビデオごとに特徴を持たせる手法であり，これを 3 次元の点群に応用したものが 3DV である．3DV は各フレームの値に重みをつけ，一つの 3 次元空間に値をまとめることで構成されており，そのため学習の実行速度やメモリ使用量の観点から見て効率的である．以下で深度画像から 3DV への変換方法について説明する．

4.2.1 深度画像から Point cloud への変換

深度画像 $(depth\_im)$ の各ピクセル (u, v) は，カメラの焦点距離 (f_x, f_y) とセンサの中心 (c_x, c_y) のカメラ内部パラメータを用いることで，センサを原点とした 3 次元空間座



a Point cloud　　b Voxel

図 4: 点群とそれを変換したボクセル

標 (x, y, z) に変換することができる．

$$z = depth\_im(u, v)$$
$$x = (u - c_x) * z / f_x$$
$$y = (v - c_y) * z / f_y$$

4.2.2 Point cloud から Voxel への変換

3 次元空間を 35mm 四方で区切り，もし内部に点が一つでもあれば，そのブロックの持つ値を 1，それ以外のブロックの値を 0 とする．これにより，体積がない点群はある一定の体積があるボクセルに変換できる（**図 4**）．

4.2.3 Voxel から 3DV への変換

フレーム総数 N の深度画像列から 3DV を作成することを考える．t フレーム目のボクセル V の座標 (x, y, z) を $V_t(x, y, z)$ とすると $V_t(x, y, z)$ の値は 1 か 0 であり，3DV は各フレームの値に重みを付けた値の加算である．t フレーム目の重みは $\frac{t}{N}$ であり，作成される 3DV を D, その座標 (x, y, z) を $D(x, y, z)$ としたとき, D の各座標の値は

$$D(x, y, z) = \sum_{t=1}^{N} \frac{V_t(x, y, z) * t}{N}$$

となる．

4.2.4 Temporal split

3DV は時系列ボクセルデータを 1 つのボクセルデータとして表現するため，古いフレームが上書きされてしまうなど，動作の情報の一部が失われる可能性がある．そのため，フレームの一部分から追加で 3DV を作成した．本実験では，$\frac{2N}{5}$ フレームずつスライド幅 $\frac{N}{5}$ で切り出して 4 つの 3DV (m_1, m_2, m_3, m_4) とした．

4.2.5 標準化

機械学習による特徴抽出の性能を上げるため，座標や各座標の値に対して標準化を行う．座標がボクセルのインデックスのままでは正の整数しかとれないため，座標を各座標の値を 1 組にまとめる．これにより，作成されるデータ P の座標 $D(x, y, z)$ が持つ値は

$$P_{D(x,y,z)} = (\overbrace{x', y', z'}^{Spatial}, \overbrace{m_G, m_1, m_2, m_3, m_4}^{Motion})$$

となる．データ P に対して以下のように標準化を行う．

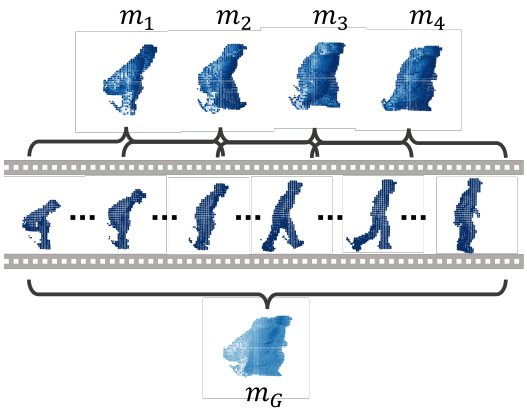


図 5: 時間で分割した映像からの 3DV の作成

Spatial　y' の最大値が 0.5，最小値が-0.5 になるように正規化し, x' と z' はそれぞれ y に対して行われた正規化の比率に応じてスケーリングする

Motion　最大値が 0.5, 最小値が-0.5 になるように正規化する

また，m に値を持つ座標の数は元の深度画像によって異なる．ネットワークに入力するデータのサイズを一様にするため，データ P の点の上限を 2048 個にし，データの数が足りない場合は 0 埋めを行う．これによって 2048×8 のデータを学習ネットワークに入力する．

4.2.6 データ拡張

多様性のあるデータセットの構築のため，データ拡張を行った．上記の方法で作成したデータに対して，鉛直方向を中心に左右に 5° 刻みで 15° まで回転させることでデータ数を 7 倍に増やした．

4.3 PointNet++

PointNet++は点群に対して高精度で学習を行えるネットワーク [21] で，PointNet[22] を拡張したものである．本研究では Positive と Negative を分けるためにクラス分類問題を PointNet++を用いて行った．

4.3.1 PointNet++の概要

点群は近傍の点同士は強い関係性（局所性）を持つことから，点群のクラスタリングには近傍の点の位置関係などの関係性を学習する必要がある．また，ネットワークへの点の入力順が異なるとしても，各点の座標が同じなら同一の出力をする必要がある（順不同性）．これらの条件を満たしつつ点群に対して学習を行える手法が PointNet であり，それを繰り返し実行することで，ある程度離れた点同士の関係性も学習を行える手法が PointNet++である（**図 6**）．図 6 の水色の点を中心に近傍の点の情報を集め，近傍の点の関係性を表す 1 つの点に情報を圧縮している．

4.3.2 PointNet++の実装

図 7 はネットワークの詳細な実装を表した図である．k 近傍法を用いてランダムに選出した点から一定距離にある点の情報を集め，その点同士の関係性を畳み込み層を用い

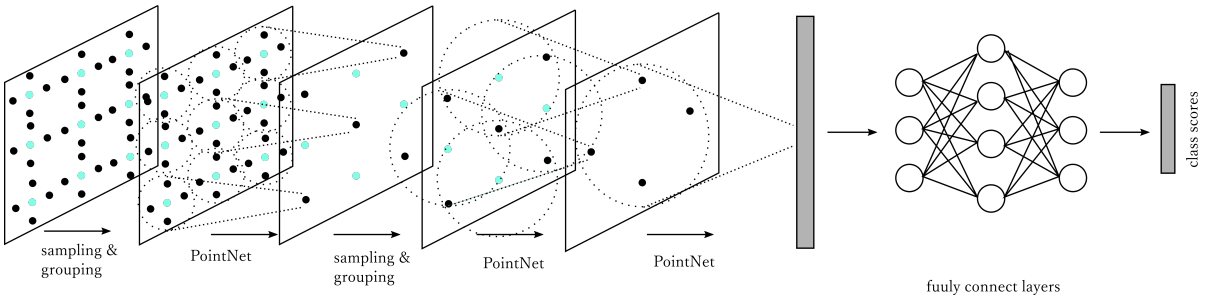


図 6: PointNet++の概念図

表 1: 実験設定の組み合わせ

	クラス数	Label	Loss	sigma
①	2	One-hot	Cross Entropy	–
②	5	One-hot	Cross Entropy	–
③	5	LDL	Cross Entropy	1.0
④	5	LDL	Cross Entropy	0.5
⑤	5	LDL	MSE	–
⑥	5	LDL+TUG	Cross Entropy + MSE	1.0
⑦	5	LDL+TUG	Cross Entropy + MSE	0.5

て学習し，Pooling 層を通して最も大きな関係性のみを残すことで学習を行っている．この処理を 2 回行うことである程度離れた点同士の関係性を学習しつつ情報の圧縮を行い，得られた情報を全結合層を通してクラスごとのスコアを算出している．

5. 実験

5.1 実験設定の説明

5.1.1 共通の実験設定

本実験では学習ネットワークの出力とロス，学習に使用する被験者の TUG のスコアの条件を変更して実験を行った．共通の条件として，11 秒以上のデータの少なさを補う交差検証のためにデータを 11 分割し，各グループの内訳は各グループに 11 秒以上の被験者を 1 人ずつ，11 秒未満の被験者はグループ間で TUG の秒数に偏りが出ないように分ける．また，訓練時には速かった方の TUG のデータだけでなく遅かった方の TUG も使用し，epoch あたりの各クラスの訓練データの数を等しくした．検証データには 11 秒以上の被験者から 3 名の被験者と 11 秒未満の被験者を 4 名選択し，計 7 名の 2 回分の TUG を検証データとした．訓練及び検証データには計測した 2 回分のデータとそれらをデータ拡張したデータも使用するが，テストデータにはデータ拡張をしておらず，かつ速かった方の TUG のデータのみを用いて実験を行った．

5.1.2 実験設定の組み合わせとその説明

クラス数はネットワークの出力の数を示しており，2 クラスは 11 秒以上 (Positive) か否 (Negative) かの 2 値分類を，5 クラスは TUG の秒数によってより細かくクラスを分けたものになる．5 クラスの内訳は 8 秒未満，8 秒以上 9 秒未満，9 秒以上 10 秒未満，10 秒以上 11 秒未満，11 秒以上である．Label は正解ラベルを意味しており，LDL は Label Distribution Learning の略で One-hot と異なり，近傍のラベルにも σ で定められた一定の影響を考慮する (**図 8**)．

表 2: 動作ごとの精度

実験設定	動作	感度	特異度
①	起立	0.8224	0.8709
①	着席	0.7552	0.8315
②	起立	0.6831	0.8692
②	着席	0.814	0.5386
⑤	起立	0.875	0.8378
⑤	着席	0.8207	0.7559

TUG は実際の TUG のスコアの秒数をそのまま正解ラベルに使用する．Loss は学習時に使用するロス関数を示しており，Cross Entropy Loss と Mean Squared Error (MSE) Loss を使用した．④はネットワークの出力と各クラスの代表値 [7.5, 8.5, 9.5, 10.5, 11.5] との内積をとることで，正解ラベルと出力のサイズを揃えることで MSE ロスを適用した．σ は LDL の正解ラベルの近傍のラベルへの影響度を示しており，σ が大きいほど近傍のラベルへの影響が大きくなる．⑤と⑥は LDL と TUG それぞれでロスをとり，その和をネットワーク全体のロスとして利用する．

5.2 実験設定ごとの精度

2 クラス分類の場合，ネットワークの出力は Positive らしさを意味する確信度を p_1 とすると $[1-p_1, p_1]$ となる．ある一定の p_1 以上の時，推定結果を Positive，それ以外を Negative とし，11 グループ全体で最も精度が高くなる閾値 p_1 のときの精度を算出した．5 クラス分類の場合，11 秒未満のクラスを 1 つのクラスと見なして 2 クラス分類の精度を算出する．

5.2.1 動作ごとの精度

実験設定ごとの起立と着席の精度の表が表**表 2** である．すべての実験設定ではなく代表的な実験設定の結果のみを記載しており，実験設定①のときの起立の閾値 p_1 は 0.07 で着席は 0.19 であった．

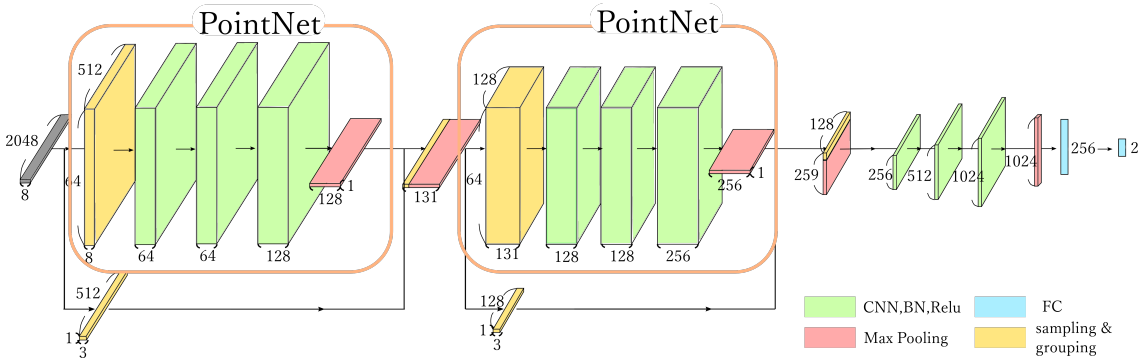


図 7: PointNet++のネットワーク構造

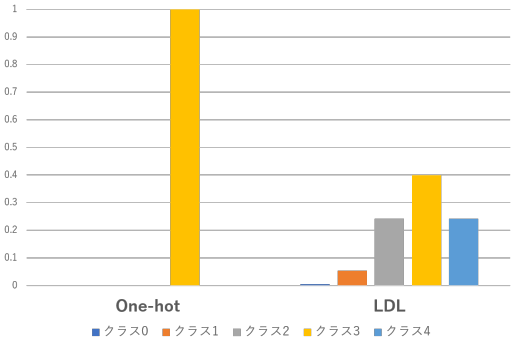


図 8: One-hot と LDL の違い

表 3: 実験設定ごとの精度

実験設定	感度	特異度
①	0.8224	0.8709
②	0.6831	0.8692
③	0.4884	0.9393
④	0.3198	0.8033
⑤	0.8750	0.8378
⑥	0.1017	0.9142
⑦	0.7820	0.8682

表 4: 学習に使用する TUG の秒数ごとの精度

実験設定	感度	特異度
⑦	0.7820	0.8682
⑧	0.5029	0.6047
⑨	0.8140	0.8711

5.2.2 実験設定ごとの精度

表 2 より，起立の動作の方が精度が高くなる傾向にあることが分かったため，以下の実験は起立のデータのみで実験を行った．実験設定ごとの感度・特異度を示したものが**表 3** である．

また，11 秒の境界に近いデータは推定が難しく，学習に使用する上でこれらのデータはノイズになり得ると考えた．そのため，実験設定⑦に加えて，10.5 秒〜11 秒（実験設定⑧），10.75〜11 秒（実験設定⑨）のデータを学習に使用せずに実験を行った．その結果が**表 4** である．

表 5: 5 クラス分類の精度

	0	1	2	3	4
0(2364)	0.6366	0.1992	0.0812	0.0444	0.0384
1(2783)	0.3057	0.3097	0.1928	0.1209	0.0709
2(1463)	0.0813	0.2064	0.2433	0.1907	0.2782
3(810)	0.0741	0.1284	0.1901	0.2889	0.3185
4(344)	0.0174	0.0291	0.0407	0.0988	0.8139

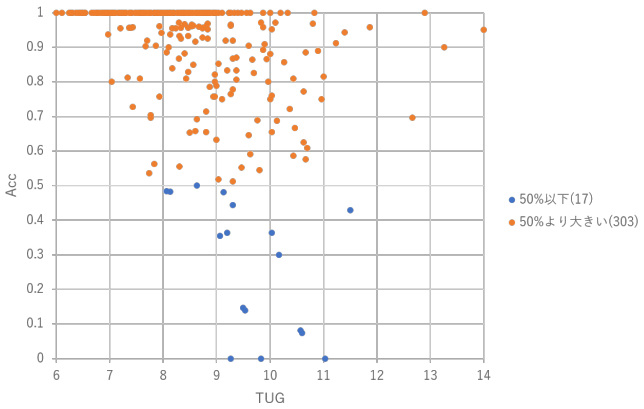


図 9: 2 クラス分類の被験者ごとの精度

5.2.3 最も精度の良かった実験設定の詳細な結果

感度・特異度の値だけ見れば実験設定⑤の精度も高いが，**図 9** のように被験者ごとの精度を出力したところ，多くの被験者で精度が 100%か 0%に近い値であった．汎化性能という観点で見ると実験設定⑨が最も優れていたため，実験設定⑨を最も精度の高い実験設定とした．その詳細な実験結果が**表 5** と図 9 である．

表 5 は 5 クラス分類の精度を示したものであり，縦軸が正解ラベルで括弧内はテストデータの数，横軸は推測したラベルとなっている．クラス 3 とクラス 4 の境界が 2 値分類における Positive と Negative の境界であり，クラス 0 から 3 を同一のクラスとみなして感度・特異度を求めたものが表 3 および表 4 である．表 5 を 2 値分類した時の被験者ごとの精度を示したものが図 9 である．図 9 は縦軸が被験者ごとの精度，横軸がその被験者の TUG の秒数となっており，色は 50 %以下か，それ以外で分けられている．

ⓒ 2023 Information Processing Society of Japan

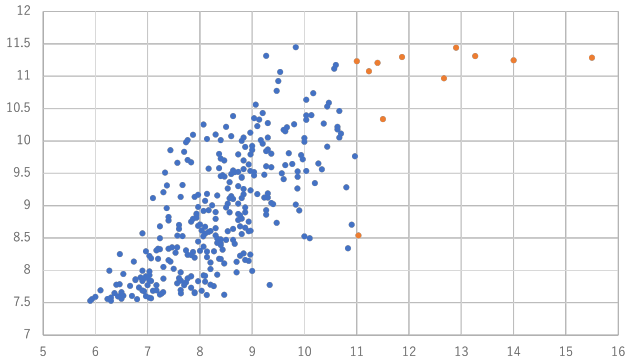


図 10: 5 クラス分類の出力から算出した TUG の期待値

図 10 は 5 クラス分類の出力から算出した TUG の期待値である．

6. 考察

動作ごとの結果（表 2）から，起立の方が精度が高いことがわかった．3DV と PointNet++を用いた手法は動作認識の分野で使用されている手法であり，動き方についてより良く学習することから，今回の推定でも動き方が 1 つの大きな判断根拠となり得る．元のデータを確認したところ，着席の動作は起立に比べて足の置き方や体を回転させる方向など人によって様々な動き方をしていたため，推定の根拠が大きく動き方に依存してしまっていた可能性がある．その一方で，起立の動作は完全に立ち上がってから一歩目を出すか，立ち上がりながら一歩目を出すかの中間にあり，着席よりもデータに一貫性があったため精度が高くなったと考えられる．

ネットワークの出力の次元数に関する結果（表 3）から，Positive と Negative の 2 値分類で学習するよりも，TUG の秒数で細かく区切った 5 クラス分類から 2 値分類した方が精度が高い．これは，同じ Negative のデータでも 6 秒付近のデータと 11 秒付近とで，間違えた時のロスの大きさを変えるべきであることを意味している．11 秒付近のデータは動き方は Positive に近いため間違える可能性が高いが，このようなデータに対して同様のロスを適用してしまうと特異度は高くなるが感度は低くなり，その結果として全体の精度は低くなってしまう．

ただし，出力を 5 クラス分類とするだけでは不十分であり，クラス 0 の 8 秒未満のデータとクラス 4 の 11 秒以上のそれぞれのクラス内で，同じクラスにも関わらず TUG の秒数に大きな差が存在している．TUG が 6 秒のデータと 8 秒付近のデータとでは，上記の場合と同様に同じロスを与えてしまうとクラス 0 の精度が上がり，クラス 1 の精度が下がる可能性がある．そのため，表 3 から実験設定③よりも実験設定⑦の方が良い結果となっていると考えられる．

TUG の秒数と精度の関係を図 9 より確認すると，11 秒に近いデータほど精度が低くなる傾向があるが，これはクラス間の閾値に近いデータほど隣のクラスの特徴も多く含むためである．表 4 から分かる通り，10.75〜11.0 秒の被験者のデータを学習に使用しない方が精度が高かった．感度が大きく向上していることから，境界付近の Negative のデータは Positive の特徴を多く含んでおり，一部の Positive のデータを推定する際に Negative と判定するように学習してしまっていたことが分かる．また，図 9 の各被験者ごとの精度が 50%以下のデータを確認すると Negative の被験者では，椅子から立ち上がり歩き始める際の最初の 1 歩の歩き出し方で，完全に立ち上がってから 1 歩目を歩き出していた．逆に，Positive の被験者で精度が著しく低かった被験者や，Negative の被験者で精度が 100%に近い被験者は立ち上がりながら 1 歩目を歩き出していた．しかし，完全に立ち上がってから歩き出しているにもかかわらず，精度が 50%を大きく上回っている被験者も少なくなかったため，動き方のみでネットワークが判定しているわけではないことも分かった．

7. 結論と課題

本研究では 3DV と PointNet++を用いることで深度映像の任意の 1 秒から運動機能が低下している人とそうでない人を分類できる可能性を示した．11 秒以上か否かの 2 値分類よりも，クラスを秒数に応じて細かく分けた 5 クラス分類の方が精度が高いことが分かった．これからの課題として，より柔軟な判断を行うためにクラス分類ではなく回帰による直接的に秒数を求められる手法の方が望ましい．また，今回は椅子や床の一部など余計な点群が入力されていたり，動き方により焦点が当たった推定手法となっている．このことから，より精度を上げるためには動き方以外の判定基準を持つような学習手法や，静止物体の影響を小さくした学習データの生成が求められる．

参考文献

[1] 内閣府. 令和 2 年版高齢社会白書（全体版）. https://www8.cao.go.jp/kourei/whitepaper/w-2020/html/zenbun/index.html, 2020.
[2] 国立長寿医療研究センター. 転びやすくなる原因は？ https://www.ncgg.go.jp/hospital/navi/22.html, 2022.
[3] 東京都福祉保健局. 廃用症候群. https://https://www.fukushihoken.metro.tokyo.lg.jp/iryo/sonota/riha_iryo/kyougi01/rehabiri24.files/siryou242.pdf, 2015.
[4] 久保和子大杉博美. アセス メントスコアを用いた効果的な転倒転落防止への取り組み. *Tokushima Red Cross Hospital Medical Journal 8*, 2003.
[5] 平井さよ子 賀沢弥貴 安西由美子森田恵美子. 転倒アセスメントスコアシートの改訂と看護師の評定者間一致性の検討. 日看管会誌, 2010.
[6] 難波孝彰, 山田陽滋. 病院内の患者の転倒転落事故防止のための深層学習を用いた一次スクリーニング自動化およびリアルタイム・リスクアセスメント方法. 日本機械学会論文集, 2019.
[7] 株式会社 FRONTEO. 転倒転落予測 ai システム

"coroban®". https://www.fronteo.com/.

[8] D PODSIADLO. The timed "up & go" : a test of basic functional mobility for frail elderly persons. *J Am Geriatr Soc*, Vol. 39, pp. 142–148, 1991.

[9] Deep learning to predict falls in older adults based on daily-life trunk accelerometry. *Pattern Recognition Letters*, 2018.

[10] Fu Xiong Wenxiang Jiang Zhiguo Cao Joey Tianyi Zhou Yancheng Wang, Yang Xiao and Junsong Yuan. 3dv: 3d dynamic voxel for action recognition in depth video. 2020.

[11] Hangzhou Hikvision Digital Technology Co. Ltd. Hikvision falling down detection. https://www.hikvision.com/uk/solutions/solutions-by-function/falling-down-detection/.

[12] Irisity AB. Iris™ fall detection. https://irisity.com/technology/fall-detection/.

[13] H. GholamHosseini, M. M. Baig, M. J. Connolly, and M. Lindén. A multifactorial falls risk prediction model for hospitalized older adults. In *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3484–3487, 2014.

[14] S. Craw Glenn Forbes, Stewart Massie. Fall prediction using behavioural modelling from sensor data in smart homes. *Artificial Intelligence Review*, 2020.

[15] Maosen Li, Siheng Chen, Xu Chen, Ya Zhang, Yanfeng Wang, and Qi Tian. Actional-structural graph convolutional networks for skeleton-based action recognition. *CVPR*, 2019.

[16] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Two stream adaptive graph convolutional networks for skeletonbased action recognition. *CVPR*, 2019.

[17] Lei Shi, Yifan Zhang, Jian Cheng, and Hanqing Lu. Skeleton-based action recognition with directed graph neural networks. *CVPR*, 2019.

[18] Pichao Wang, Wanqing Li, Zhimin Gao, Chang Tang, and Philip O. Ogunbona. Depth pooling based large-scale 3-d action recognition with convolutional neural networks. *IEEE Trans Multimedia*, 2018.

[19] Yang Xiao, Jun Chen, Yancheng Wang, Zhiguo Cao, Joey Tianyi Zhou, and Xiang Bai. Action recognition for depth video using multi-view dynamic images. *Inf. Sci.*, 2019.

[20] 公益財団法人 日本整形外科学会. https://www.joa.or.jp/index.html.

[21] Hao Su Charles Ruizhongtai Qi, Li Yi and Leonidas J Guibas Tuytelaars. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. 2017.

[22] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 77–85, 2017.

ⓒ 2023 Information Processing Society of Japan