情報処理学会論文誌 Vol.61 No.2 486–499 (Feb. 2020)

Kinectを用いた行動座標によるピッキング行為の検知

白石 将貴[1,a)] 宇田 隆哉[1,b)] 藤川 真樹[2,c)]

受付日 2018年12月26日, 採録日 2019年11月7日

概要：本論文は，家人による解錠操作と，ピッキングによる解錠操作を区別することを目的とするものである．住宅のセキュリティを考える際，出入り口以外は一般的に人が出入りしないため，簡易なセンサによって不正な侵入を検知することが可能であるが，出入り口は家人も出入りするため，単純なセンサによる検知はできない．監視カメラの設置は一般的であるが，侵入があったことを後日知ることはできても，侵入時にその侵入を検知するには有人監視を常時行うしかない．そこで，本論文では，Kinect を用いて不正な侵入を自動的に検知するシステムの提案を行う．Kinect を用いた動作識別の研究はすでに行われており，不正な侵入のなかでも，通常の出入りとは異なる，ドアを破壊するような行為の検知は既存技術でも対応可能である．そこで，本論文では家人による解錠操作と，ピッキングによる不正な解錠操作を区別する点に重点を置いた．本研究では機械学習を用いている．大量にデータを集めて学習しさえすれば，従来の単純な機械学習でも家人による解錠操作とピッキングを区別できる可能性はあるが，訓練にかかる時間や計算機のメモリが問題となる．そこで，本研究では，少人数の被験者のデータで効果的に訓練できる方法を模索した．具体的には，体格差の統一や行動周期の統一，擬似的サンプル数の増加である．Kinect を使用することにより，撮影された画像を保存する必要がないため，被撮影者のプライバシも保護される．実験の結果，線形補間により生成したサンプルを追加した場合の CNN を用いたピッキングの区別は F 値の平均 57%であったが，被験者によっては 94%の精度で区別ができ，類似の特徴を持つ他人がいる場合に精度が大きく上昇する可能性があることが分かった．

キーワード：Kinect，骨格座標，機械学習

Detection of Lock-picking Action with Kinect by Action Coordinate

Masaki Shiraishi[1,a)] Ryuya Uda[1,b)] Masaki Fujikawa[2,c)]

Received: December 26, 2018, Accepted: November 7, 2019

Abstract: The objective of this paper is to distinguish opening a door by the key from opening it by picking. In terms of security of a house, windows can be monitored by small sensors since no one goes through windows in usual. On the other hand, doors cannot be monitored by the sensors since not only thieves but also residents go through the doors. CCTV (closed-circuit television) camera is one of the solutions. However, invasion can be detected not during but after the invasion, or the movie by the camera must be always watched by a person. Therefore, in this paper, we propose the system which automatically detects invasion by Kinect. Of course, there are some researches in which Kinect is used to distinguish human actions, and big actions such as breaking doors can be detected with methods in the researches. In contrast, we put the stress on to distinguish a small difference such as difference between opening a door by the key and opening it by picking. We used machine learning in this paper. If big data of opening a door can be collected, the small difference would be able to be distinguished by usual way of machine learning. However, time for training and memory space on computers are not ignorable. Therefore, in this paper, we searched for methods which enable effective training with few examinees. Specifically, unifying the physiques and action cycles of examinees and creating pseudo samples. Using Kinect also protects the privacy of people in the movie since only coordinates of joints of people are stored. The average of F-measure of detecting picking by CNN with generated samples by linear interpolation was 57%, but the highest mark of an examinee was 94%. It shows that examinees with similar behavior may increase the percentage.

Keywords: Kinect, coordinates of joints, machine learning

[1] 東京工科大学
Tokyo University of Technology, Hachioji, Tokyo 192–0982, Japan
[2] 工学院大学
Kogakuin University, Shinjuku, Tokyo 163–8677, Japan

[a)] g2117031c3@edu.teu.ac.jp
[b)] uda@stf.teu.ac.jp
[c)] fujikawa@cc.kogakuin.ac.jp

1. はじめに

一般家屋への侵入検知を考える場合，家の窓に安価なセンサを取り付ける方法が考えられるが，家人が常時出入りする家の扉にこのようなセンサを導入することはできない．監視カメラの設置は有効ではあるが，犯行後に侵入者の特定や時刻の特定はできても，侵入時にこれを検出するには常時有人監視する必要があり，非常に人的コストが掛かる．

近年では，三菱電機による「深層学習を使い，商業施設にいる不審者や社会的弱者を監視カメラでとらえるシステム」[1] のように，監視カメラを人工知能により監視するという技術も登場している．このシステムでは，杖や乳母車などのものを属性として定義することで，深層学習を利用して映像を分析し，不審な行動をとっている人や手助けが必要な人を識別している．

監視カメラと深層学習を組み合わせれば，常時有人監視しなくても扉からの不正な侵入を検知できるように思えるが，問題は単純ではない．監視カメラによる映像のデータ量は大きく，これを各家庭からセンターに常時転送して集約することは容易ではない．そこで，本論文では，Kinect を用いて扉から不正に侵入する行為を検知するシステムを提案する．Kinect から得られた骨格座標のみを行動識別に利用すれば，センターに転送するデータ量を削減できる．

なお，映像ではなく，Kinect や他のセンサデバイスからの情報を用いて行動判別を行う研究は他にも存在する．これらの研究は，パンチやキック，ウォーキングなどの大きく異なる行動を対象にした識別手法であり，これらの技術を用いれば，扉を破壊して侵入するなどの行動は検出可能と思われる．そこで，本論文では，ピッキングと通常の鍵開けのように，動作の差が小さいものを高精度で区別できるのかという点に絞り，様々な実験を行った．さらに，少ない被験者数で訓練を行い，それが分類精度にどのように影響するのかを調査した．なお，本論文で用いる精度という用語は，どれくらいの確率で分類が行えるかという一般的な意味で用いており，深層学習における Accuracy とは区別している．

2. 関連研究

2.1 行動識別手法

Pang らは，Kinect を用いて，骨格座標の位置と速度から，パンチやキックの行為を検知するシステムの提案をしている [2]．実験の結果，パンチは 95.83%，キックは 100%の精度で検知することができている．彼らは，機械学習を用いずに，骨格座標の数値を区分するだけで行動を判別しているため，ピッキングと鍵開けといった，骨格座標の位置と速度が類似している行動の区別には不向きである．

Horiuchi らは，機械学習を用いて 0.5 秒後の人体の動きをリアルタイムで推定し出力するシステムの提案をしている [3]．実験の結果，ジャンプ時の体の重心は実際の動作の 0.5 秒前後に 7.0 cm の誤差で推定できる．しかし，これは被験者が途中で意図的に行動を変えることができない動作に限定して予測を行うものであり，ドアの前に立った人間が次にピッキングをするか鍵開けをするかというような，意図的に行動を選択できるものの予測は行えない．

中原らは，掃除ロボットなどに 3D センシング機器を付加し，家庭内や施設内で高齢者を認識・追跡して日常行動を把握する手法を提案している [4]．彼らの手法では，簡易ロボットに Kinect センサを搭載し，深度画像を用いて対象の行動認識を行っている．実験の結果，歩行のような特徴ある行動に関しては高精度での識別が可能であるが，ピッキングと鍵開けといった類似の行動が識別できるかは不明である．

中島らは，人物画像から検出した骨格画像を用いて，人体に遮蔽が起こる状況でも骨格座標を検出し追跡する手法を提案している [5]．実験の結果，安定して検出される首関節を追跡することで，遮蔽された残りの骨格座標が推定できている．しかし，これは大まかな推定に過ぎず，ピッキングと鍵開けといった類似の行動における骨格座標の位置の違いを予測することは不可能である．

渡邊らは，人物を上方から撮影した距離画像から，人物の姿勢を推定するための手法を提案している [6]．彼らは，点群をボクセルリスト化し人体の部位の長さで探索する逐次探索型のルールベース手法とランダムフォレストによる機械学習手法を用いることで，推定処理やトラッキング処理の精度向上や，上方視点から特徴量を抽出することで処理高速化を目指している．彼らの実験では，逐次探索手法では TOF による正解率は 75%で，機械学習手法では 60%でリアルタイムでの姿勢推定が可能であった．しかし，大まかな姿勢推定でもこの精度であり，ピッキングと鍵開けといった類似の行動が識別できるかは不明である．

2.2 個人識別手法

森らは，歩容情報を用いて個人識別を行う手法を提案している [7]．彼らの手法では，Kinect v2 を用いて取得した足首間の距離から 1 周期分の歩行動作を抽出している．そして，静的距離，動的距離，関節角度に関しての特徴量を平均値，中央値，最大値から求めている．実験の結果，体格差から生じる動的距離が識別に有効としており，我々の研究に適用する場合，家人か不審者かを区別することには使用可能と思われる．しかし，家人と体格が類似している不審者は区別できず，1 度ドアの前に立たれてしまうと，ピッキングを検出することはできない．

Dehbandi らは，Kinect と機械学習を用いて，手や腕の障害や上肢障害のレベルを部類するための手法を提案している [8]．彼らの実験では，健康な被験者が，健康な個体，軽度障害，中度障害をエミュレートとして評価を行ってお

表 1 関連研究における行動識別の比較一覧
Table 1 Comparative list of action detection in related researches.

研究者	識別・推定するもの	我々の目的
Pang ら	パンチとキック	類似の座標は不可能
Horiuchi ら	ジャンプ中の 0.5 秒後の動作	意図的に変更可能な動作は不可能
中原ら	歩行しているか	類似行動には非言及
中原ら	遮蔽された骨格座標	詳細な予測は不可能
渡邊ら	人物の姿勢	類似の座標は不明
森ら	歩行中の人物	体格のみのため不可能
Dehbandi ら	障害のレベル	他人の識別に適用不可

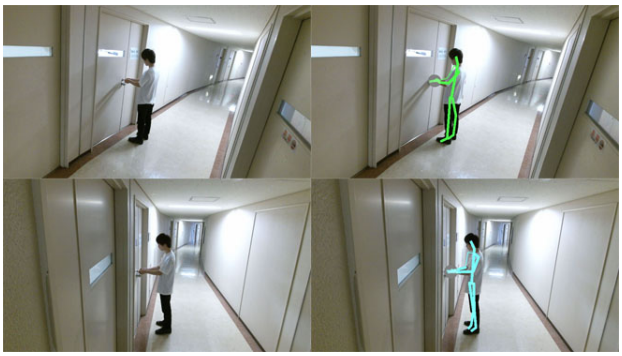

図 1 Kinect で取得したピッキング時の RGB 画像と骨格座標
Fig. 1 RGB and joints images by Kinect on picking.

り，健康な個体は 100%，軽度障害は 83.3%，中度障害は 91.7%，平均では 91.7%で障害を識別可能としている．しかし，Dehbandi らの評価では訓練とテストの被験者が同一人物であり，我々の研究に適用する場合，ピッキングを行う不審者本人の行動を学習することになり，状況としてあり得ない．

以上の関連研究と，鍵開けとピッキングを区別するという我々の目的との比較を**表 1** にまとめる．

3. 提案手法

3.1 提案概要

本論文では，Kinect と機械学習を用いてピッキングと鍵開けを区別する手法を扱う．ピッキングとは，針金などの工具を用いてドアを開錠することであり，本研究の実験においては，棒状のものを鍵穴に沿って上下する動作としており，右手で棒状のものを持ち鍵穴付近を上下に動かす．一方，鍵開けとは，正しい鍵を鍵穴に差し込み鍵を回す動作としており，右手で鍵穴に鍵を入れて鍵の開閉を行う．いずれの動作においても，左手はドアノブに添えておく．

本研究で区別する内容を画像で示す．**図 1** は，ピッキング時の RGB 画像と骨格座標をそれに重畳した画像を，本論文の評価と同じ位置から Kinect で撮影したものである．同様に，**図 2** は，鍵開け時のものである．

ピッキングを行う者は基本的に家人ではないが，本研究

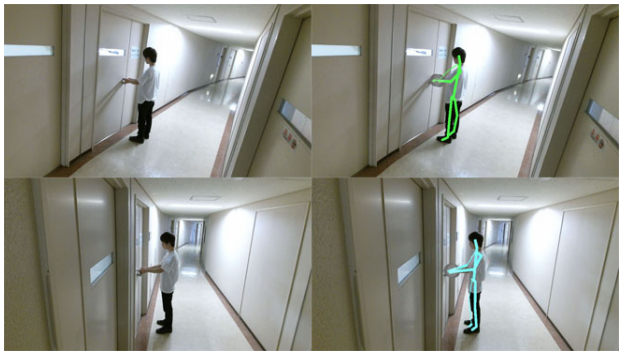

図 2 Kinect で取得した鍵開け時の RGB 画像と骨格座標
Fig. 2 RGB and joints images by Kinect on unlocking with key.

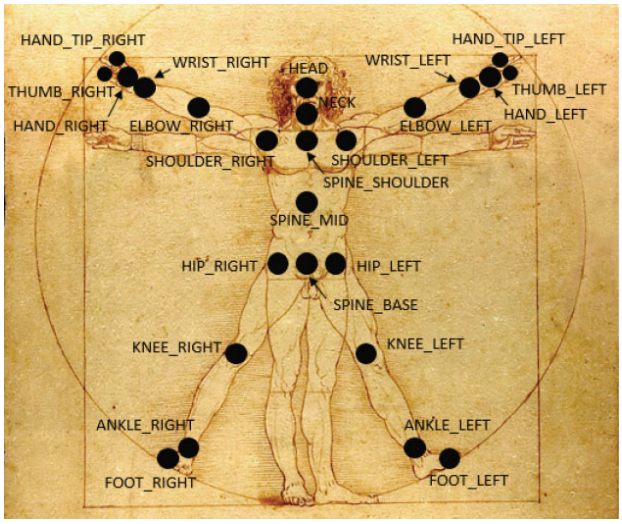

図 3 全骨格座標の位置と名称
Fig. 3 Positions and names of all joints[*1].

においては行動の区別のみを扱い，被験者の区別は行わない．ただし，現実世界においてピッキングを行う者の行動を事前に集めて訓練に使用することは困難であるため，テストに使用する被験者とは別人物で訓練を行っている．

なお，防犯性を高めるだけであれば，ドアをツーロックにしたり，指紋やカードを使った電子的な鍵にするという手法も有効であるが，本研究の目的は，家屋への侵入時にそれを検知することである．

3.2 骨格座標のデータ

Kinect のスケルトントラッキングによって骨格座標を取得するが，本研究で扱う Kinect v2 は人体を**図 3** に示す 25 個の骨格座標として表しており，1 つの座標を X，Y，Z 軸の座標で表現できる．本研究では，1 つの骨格座標の X，Y，Z 軸の座標をカンマ区切りにして表記し，さらにそれを 25 個の骨格座標の分だけカンマ区切りにして 1 行に記録している．これを 1 フレームと呼ぶ．1 フレームを 40 コ

[*1] https://adinora.com/2018/03/14/die-liebe-der-tod-und-die-zeit-danach/

マ，つまり 40 行取得したものを，機械学習における 1 サンプルとしている．

3.3 骨格座標データの整形

訓練とテストに使用する被験者間の体格差を補正するため，骨格座標データの整形を行った．ドアノブに接している骨格座標と床に接している骨格座標の値を固定し，それ以外の骨格座標の値を，被験者がおよそ平均的な身長になるように線形に変更している．ドアノブに接している骨格座標を基準としたのは，ドアノブの位置は被験者にかかわらず同一であるためであり，床に接している骨格座標の値を基準としたのは，足の座標が地面にめり込んだり宙に浮いたりするのを防ぐためである．なお，基準とする平均身長は，総務省が発表している 2018 年度日本人男性の平均身長を参考に 170 cm とした [9], [10]．

3.4 被験者データを元にしたデータ生成

本研究では，サンプル数が少ないため，擬似的にサンプル数を増やすために，被験者データを元にしたデータ生成を行う．まず，2 名の被験者の同じフレーム，同じ骨格座標の値の平均値をとる．次に，両被験者の骨格座標の値からこの平均値を引き，2 で割る．この値を両被験者の骨格座標の値に足したもの，および先ほど求めた平均値の値を新たな骨格座標のサンプルとして追加する．

3.5 行動周期を一致させるためのフレームの線形補間

ピッキングと鍵開けの速度は被験者によって異なるため，その際の周期が一致すれば精度が上がる可能性がある．そこで，線形補間によりフレームの時間間隔を伸縮し，サンプルを作成する．その際の処理について説明する．たとえば，40 フレームのサンプルがあるとして，これを 41 フレーム以上に線形補間して 40 フレームまでを使用する場合，41 フレーム以降のデータは削除される．逆に，39 フレーム以下に線形補間して 40 フレームまでを使用する場合，不足するフレームは 1 フレーム目から不足するフレーム数だけコピーして使用する．線形補間して 37 フレームになったのであれば，38〜40 フレームは 1〜3 フレームのコピーとなる．

3.6 骨格座標のスライド

本手法のシステムを実際に利用する場合，ピッキングを行う不審者自身の行動を訓練して分類に利用することはできない．本論文の評価においても，訓練に利用する被験者とテストに利用する被験者は異なる．そこで，被験者の骨格座標の値をスライドさせ，全被験者の骨格座標の値が近いものになるようにした場合の評価も行った．

各フレームの基準となる元の座標を被験者 1〜10 の 10 名分の基準となる座標の平均値で引き，スライドさせるための座標値を求めた．その後，各フレームの元の値と先ほど求めたスライドさせるための座標値を足し，保存した．なお，被験者 1〜10 から求めた値は，被験者 1〜20 に適用している．

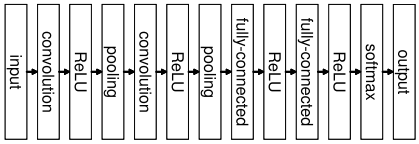

図 4 CNN のネットワーク構造
Fig. 4 CNN network structure.

スライドを行う際，基準となる骨格座標が必要となる．そこで，動作を行っているときに最も座標の値がほぼ一定である足のかかととして，左のかかとを基準とした．なお，Kinect の設置場所により骨格座標の値に差があるため，2 地点の撮影場所から取得したそれぞれの平均値を使用した．

3.7 機械学習

本論文では，畳み込みニューラルネットワーク（CNN）およびサポートベクターマシン（SVM）により評価を行う．ただし，主たる評価に使用するものは CNN であり，SVM は CNN の値の変化を確認する際にのみ使用している．

CNN においては，1 次元の畳み込みを行う．このネットワークの構造は，**図 4** に示すように，畳み込み層，ReLU 層，プーリング層，畳み込み層，ReLU 層，プーリング層，全結合層，ReLU 層，全結合層，ReLU 層，ソフトマックス層とした．1 サンプルはカンマ区切りの数値 40 行で構成されているため入力ユニット数は 1 であり，またソフトマックス層によって出力されるユニット数は，ピッキングか鍵開けかの 2 値分類を行うので 2 である．このときに使用するバッチサイズは 32 から 1,024 までの値の間で 2 を n 乗した値または 1,800 の値とし，エポック数は 50 から 50 ずつ増加させていき，F 値が 90%を超えたものから選択する．本論文の評価においては，予備実験にて F 値が 90%以上であり，かつ標準偏差の値が小さいエポック数とバッチサイズの組合せを選択している．

評価の際には，10 分割交差検証を用いている．これに関しては，行動識別とは関係ないが，機械学習を用いた評価を行う場合に，訓練用サンプルとテスト用サンプルの比率を調査したところ，山口らの研究 [11] および川村らの研究 [12], [13] では 9 割を訓練に 1 割をテストに使用していたことから，これらに合わせた．

SVM の構造は，入力層，カーネル層，出力層とする．精度評価には F 値を使用する．

なお，CNN および SVM の評価においては，precision（P）と recall（R）と F 値を示す．F 値は $2\times P\times R/(R+P)$ で求まるが，本論文の評価で示す値は，鍵開けを class 0，ピッキングを class 1 と分類する際の P，R，F 値の 2 クラ

ス平均である．よって，たとえば，P が 0.76 で R が 0.52 の場合，F 値が 0.38 となると計算式と合わないように思うかもしれないが，class 0 が P：1.00，R：0.04，F 値：0.08 で，class 1 が P：0.51，R：1.00，F 値：0.68 であれば，平均値では前述の値となる．

4. 実装

4.1 スケルトントラッキング

3.2 節で述べた Kinect v2 によるスケルトントラッキングの環境は次のとおりである．OS は Windows 8.1，IDE は Visual Studio 2015，プラットフォームは x64，OpenCV のバージョンは 3.1.0，プログラミング言語は C++を用いた．Kinect v2 のデータ取得におけるメソッドの流れを**図 5** に示す．

図 5 に示すように，Sensor，Source，Reader，Frame，Data を順に取得していく必要がある [14]．始めに Sensor で，Kinect を扱うための Sensor インタフェースを定義し，Sensor を取得し，Sensor を開く．次に Source で，Color フレームのための Source インタフェースを定義し，その後 Sensor から Source を取得する．さらに Reader で，Color フレームのための Reader インタフェースを定義し，Source から Reader を開く．Frame と Data では，Color 画像のサイズ，データサイズの設定をし，Color 画像を扱うために Open CV の準備を行う．また，Color 画像を取得するための Frame インタフェースを定義し，Reader から Frame を取得し，Frame から Color 画像を取得する．そして，ファイルに動画を書き出す．この動画ファイルは被験者が Kinect 内に収まっているかを確認するためのものである．Color 画像を表示するにはサイズが大きいので，サイズを半分にし，最後に Frame を解放した．

次に，被験者の骨格座標を取得する．まず，Body のための Frame インタフェースを定義し，Reader から Frame を取得する．そして Frame から Body を取得することにより，被験者から骨格座標を取得していく．Kinect v2 では，同時に 6 人まで骨格座標を取得可能なので，場合分けによってそれぞれのテキストファイルに骨格座標を記述するようにしている．そして，3.2 節で述べた 1 サンプル単位でファイルに出力し，異なる被験者のサンプルは異なるファイルに出力されるようにした．フレームごとの骨格座標の取得間隔は 0.025 秒とし，1 サンプルは 2 秒間のデータで構成されている．これは，Kinect v2 の性能上，骨格座標の値が更新されてから取得される最小のタイミングを考慮して決定したものである．1 サンプルのファイルサイズは 26 KB から 28 KB である．

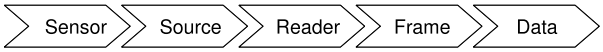

図 5 Kinect v2 におけるデータ取得の流れ
Fig. 5 Flow of data acquisition by Kinect v2.

4.2 提案手法の実装

提案手法の実装を次に示す環境で行った．OS は Windows 8.1，IDE は NetBeans 8.0.1，プログラミング言語は Java である．

最初に，3.3 節で述べた骨格座標データの整形の実装について述べる．まず，1 サンプル分の値である 3,000 個の骨格座標を格納する配列を入力と出力の 2 種類で定義した．また，1 サンプル分の行数を格納するための配列を定義した．その後，テキストデータを読み込み，カンマ区切りの値を抽出し，各行が 1 つの配列に格納されるように数値を代入した．

さらに，被験者の身長が 170 cm になるように拡大縮小した．この拡大縮小では，HEAD と，FOOT RIGHT および FOOT LEFT 間の平均を求め，それが 170 cm になるように特定の値で割っている．この値を出力する側の配列に代入し，テキストファイルとして出力した．被験者 1 名あたりのデータは 100 × 4 通りであるため，被験者ごとにファイル名を 4 回変更しながらこれを 100 回繰り返している．

このデータ整形過程を具体的に説明すると，WRIST RIGHT，HAND RIGHT，WRIST LEFT，HAND LEFT，THUMB RIGHT，HAND TIP RIGHT，THUMB LEFT，HAND TIP LEFT，ANKLE RIGHT，FOOT RIGHT，ANKLE LEFT，FOOT LEFT の座標を固定している．足のサイズに関しては，身長と足のサイズの間には必ずしも相関がないため本実験では固定した．ELBOW RIGHT と ELBOW LEFT に関しては，HAND RIGHT と HAND LEFT の骨格座標を基準にし，それ以外は FOOT RIGHT と FOOT LEFT の位置を基準にして，鉛直方向の Y 座標のみを変更した．このデータ整形を行った後の 1 サンプルのファイルサイズは 27 KB から 29 KB となった．

3.4 節で述べた，被験者データを元にしたデータ生成の実装についての配列への格納およびテキストファイルの処理に関しては，骨格座標データの整形の実装と同様である．この処理を行った結果，1 サンプルのファイルサイズは 30 KB から 32 KB となった．

3.5 節で述べた，行動周期を一致させるためのフレームの線形補間の実装について説明する．配列への格納およびテキストファイルの処理に関しては，骨格座標データの整形の実装と同様である．

3.6 節で述べた，骨格座標のスライドの実装についての配列への格納およびテキストファイルの処理に関しては，骨格座標データの整形の実装と同様である．

4.3 機械学習の実装

4.3.1 CNN の実装

3.7 節で述べた CNN の実装について説明する．畳み込

みには chainer.links の ConvolutionND を用いた．1 層目の畳み込み層では，入力チャネルはサンプル分なので 1，出力チャネルは 1 サンプル分の行数である 40，フィルタサイズは 3，ストライドを 3，パッドを 1 とした．2 層目は ReLu 層である．3 層目は平均プーリング層であり，プーリングサイズを 2 とした．4 層目の畳み込み層では入力チャネルを 40，出力チャネルを 20，フィルタサイズを 3，ストライドを 1，パッドを 1 とした．5 層目は ReLu 層である．6 層目は平均プーリング層であり，プーリングサイズを 2 とした．7 層目は全結合層であり，入力ユニット数 360 として，出力ユニット数を 1 サンプル分の数値の個数である 1,000 とした．8 層目は ReLu 層である．9 層目は全結合層であり，入力ユニット数 1,000 として出力ユニット数を 1,000 とした．10 層目は ReLu 層である．11 層目はソフトマックス層であり，入力ユニット数が 1,000 であり，2 値分類を行うので出力ユニット数は 2 である．

次に，全サンプルが存在するディレクトリを指定し，そこから全ファイルのリスト名を取得する．さらに，テスト用サンプルが全体の 1 割となるように選択し，そのリスト名を取得する．そして，残りを訓練用サンプルとして，そのリスト名を取得する．その後，それぞれのサンプルのデータを float 型の 32 ビットの配列に格納する．それぞれのサンプルのリスト名は int 型の 32 ビットの配列に格納する．そして，それぞれのサンプルのデータ配列の次元数を，$1 \times 40 \times 75$ の 3 次元に変更する．さらに，データセットを作成し，データを順番に取り出す．また，epoch と，学習もしくはテスト時の誤差関数 precision を表示し，学習の最後にそれぞれのディレクトリの precision，recall，F 値を表示する．

すべての被験者のサンプルを訓練に使用する際には，10 分割交差検証のために，各サンプルに番号を振り，その番号を 10 で割った余りによって，その交差検証に使用するテスト用サンプルを区別している．一方，テスト用サンプルに 1 名の被験者のデータを使用し，残りの被験者のデータを訓練用サンプルとする場合には，被験者ごと入れ替えてテスト用サンプルを区別している．

4.3.2 SVM の実装

3.7 節で述べた SVM の実装について説明する．SVM の実装には sklearn を用いており，LinearSVC を使用している．パラメータはすべてデフォルト値である．よって，線形 SVM が精度評価に用いられている．

最初に，全サンプルが存在するディレクトリを指定し，そこから全ファイルのリスト名を取得する．次に，テスト用サンプルが被験者 1〜10 のうち 1 人となるように選択し，そのリスト名を取得する．そして，被験者 11〜20 のサンプルすべてを訓練用サンプルとして選択し，そのリスト名を取得する．

それぞれのサンプルのデータを float 型の 32 ビットの配列に格納し，それぞれのサンプルのリスト名を int 型の 32 ビットの配列に格納する．次に，線形 SVM の分類器にパラメータを与えるが，すべてデフォルト値のままとした．最後に，ピッキングと鍵開けにおけるそれぞれのディレクトリの precision，recall，F 値を表示した．

5. 評価手法

5.1 ドアと被験者の位置関係

被験者がピッキングと鍵開けを行う際に，自由にどのように行ってもよいことにしてしまうと，極端な例としては右手と左手の動作を逆にしたり，不自然な体勢で鍵を開けることもでき，正しく評価できない．そこで，鍵開けを行う際に，自然に行うとどうなるのか，足のかかとの開き幅やドアとかかととの距離を調査した．かかとを基準に測定した理由は，足は最も体重がかかる場所であり体を支える場所であることと，つま先は足を開く幅に個人差があるのではないかと考えたためである．土踏まずも，足の中央にあるため，足が開く場合にかかとよりも影響を受ける．また，被験者により足のサイズも異なるため，脚が足と接続されているかかとがこれらの影響を一番受けない箇所である．そこで，本研究では，鍵開けを行う被験者 1 から 10 の 10 名に対して，本人が最も自然に鍵開けを行う際の足の開き幅とドアと被験者との距離の関係を測定し，その平均を一般的な人物のドアを開錠するときの立ち位置とした．**表 2** にドアと被験者の位置関係を示す．その結果から，被験者 10 名におけるかかとの開き幅の平均は約 14 cm となり，ドアからかかとの距離の平均は約 56 cm となった．よって，Kinect を使用した本研究の評価実験においては，すべての被験者をこれらの数値の位置に配置した．

5.2 骨格座標データの複数連続取得

評価実験においては，被験者が継続してピッキングや鍵開けの動作を行っている際に，連続して複数のサンプルを取得している．本実装においては，Kinect を用いたプログ

表 2 ドアと被験者の位置関係
Table 2 Positions of examinees against door.

被験者番号	かかとの開き幅 [cm]	ドアとかかとの距離 [cm]
被験者 1	14	55
被験者 2	19	51
被験者 3	17	65
被験者 4	7	56
被験者 5	15	48
被験者 6	16	61
被験者 7	12	60
被験者 8	12	52
被験者 9	12	55
被験者 10	15	59
平均	14	56

ラムを実行すると，サンプル番号と取得するサンプル数を入力する画面が表示されるので，数値を入力と Kinect が動作し，骨格座標が取得され始める．その際，骨格座標が認識されていない場合があると実験に支障があるため，被験者は体を Kinect に向けてからピッキングや鍵開けの動作に移るようにした．そのため，データ取得直後のサンプルは評価に用いず，骨格座標が取得され始めてから約 5 秒後のサンプルから評価に使用している．これは，5 秒あれば，被験者が扉のほうに向き直り，指定された位置でピッキングや鍵開けの動作を行い始めるのに十分だからである．ここで，サンプルの取得間隔は 2 秒，1 サンプル分の取得に必要時間は 1 秒間強である．それをふまえて本研究では，1 回あたりの取得サンプル数を 25 とし，それを 2 回行い 50 サンプル分のデータを取得した．したがって，Kinect による骨格座標データの取得回数を 27 とし，始めの 2 サンプルは評価に使用せず，それ以降の 25 サンプルを評価に使用している．

5.3 被験者データを元にした具体的なデータ生成方法

3.4 節で示した被験者データを元にしたデータ生成手法の具体的な方法について説明する．まず，被験者 1〜10 から，重複のないようにすべての組合せでペアを作成する．たとえば，被験者 1 は被験者 2〜10 とペアになり，被験者 2 は被験者 3〜10 とペアになる．被験者のすべての組合せで 45 通りのペアが作成され，これに対して被験者データを元にしたデータ生成を行った．生成されるサンプルは，オリジナルの被験者データである 2,000 サンプルに対し，27,000 サンプルである．

5.4 行動周期を一致させるためのフレームの具体的な線形補間方法

3.5 節で示した行動周期を一致させるためのフレームの線形補間の具体的な方法について説明する．周期変更に関しては，周期を 0.7 倍したものと 1.3 倍にしたサンプルを生成した．その理由を，鍵による開錠動作における行動周期を示した**表 3** と，ピッキングによる開錠動作における行動周期を示した**表 4** を用いて説明する．参考までに被験者 20 名の値を掲載しているが，本研究の評価で基本的に用いているものは，被験者 1〜10 の数値である．表 3 から，鍵による開錠動作における周期の最小値は 0.35 [回/秒] であり，最大値は 1.15 [回/秒]，平均値は 0.71 [回/秒] であった．また，表 4 から，ピッキングによる開錠動作における周期の最小値は 0.5 [回/秒] であり，最大値は 1.55 [回/秒]，平均値は 0.93 [回/秒] であった．以上の結果から，それぞれのサンプルの周期を 0.7 倍もしくは 1.3 倍すれば，他のサンプルの元々の周期に一致もしくは非常に近い値になると考え，0.7 倍と 1.3 倍の周期のサンプルを生成することとした．生成されるサンプルは，オリジナルの被験者デー

表 3 鍵による開錠動作における行動周期
Table 3 Action cycle in key unlocking.

被験者番号	回数 [回数/秒]					標準偏差
	1	2	3	4	5	
被験者 1	0.70	0.70	0.75	0.75	0.70	0.0245
被験者 2	0.65	0.80	0.70	0.75	0.80	0.0245
被験者 3	0.70	0.65	0.75	0.70	0.70	0.0583
被験者 4	1.10	1.10	1.05	1.10	1.15	0.036
被験者 5	0.45	0.35	0.50	0.35	0.35	0.0632
被験者 6	0.70	0.65	0.70	0.65	0.65	0.0245
被験者 7	0.40	0.35	0.35	0.35	0.40	0.0245
被験者 8	1.10	1.00	1.10	1.05	1.00	0.0447
被験者 9	0.50	0.60	0.50	0.50	0.45	0.0490
被験者 10	0.85	0.85	0.80	0.75	0.85	0.0400
被験者 11	0.60	0.60	0.80	0.85	0.85	0.116
被験者 12	0.45	0.50	0.50	0.30	0.25	0.105
被験者 13	0.40	0.55	0.50	0.50	0.60	0.0663
被験者 14	0.30	0.40	0.55	0.65	0.60	0.130
被験者 15	0.35	0.40	0.90	0.85	0.65	0.105
被験者 16	0.60	0.65	0.75	0.85	0.90	0.114
被験者 17	0.30	0.45	0.55	0.55	0.55	0.0980
被験者 18	0.75	0.85	0.75	0.85	0.80	0.0447
被験者 19	0.40	0.55	0.70	0.85	0.90	0.186
被験者 20	0.80	0.85	0.85	0.70	0.85	0.0583
被験者 1〜10			0.71			0.235
被験者全体			0.66			0.215

表 4 ピッキングによる開錠動作における行動周期
Table 4 Action cycle in unlocking by picking.

被験者番号	回数 [回数/秒]					標準偏差
	1	2	3	4	5	
被験者 1	1.15	1.20	1.20	1.40	1.35	0.0245
被験者 2	1.15	1.45	1.10	1.35	1.40	0.0583
被験者 3	0.55	0.55	0.55	0.55	0.60	0.0316
被験者 4	1.15	1.15	0.75	1.05	1.10	0.0316
被験者 5	0.50	0.50	0.90	0.85	0.65	0.0632
被験者 6	0.55	0.80	0.90	0.75	0.85	0.0245
被験者 7	0.55	0.50	0.55	0.60	0.55	0.0245
被験者 8	0.85	0.85	1.00	1.05	1.20	0.0447
被験者 9	0.65	0.60	0.75	0.70	0.80	0.0490
被験者 10	1.35	1.30	1.55	1.60	1.55	0.0400
被験者 11	1.95	1.75	2.20	2.25	1.95	0.183
被験者 12	0.45	0.40	0.35	0.35	0.40	0.0374
被験者 13	0.75	0.70	0.65	0.65	0.70	0.0374
被験者 14	0.80	1.20	1.45	1.40	1.40	0.241
被験者 15	0.40	0.50	0.55	0.65	0.60	0.0860
被験者 16	0.85	1.50	1.55	1.95	1.95	0.403
被験者 17	0.65	0.65	0.65	0.60	0.60	0.0245
被験者 18	1.00	1.15	1.00	1.40	1.55	0.220
被験者 19	0.60	0.70	0.60	0.70	0.70	0.0490
被験者 20	0.55	0.50	0.50	0.70	0.65	0.0812
被験者 1〜10			0.93			0.332
被験者全体			0.94			0.446

タである 2,000 サンプルに対し，108,000 サンプルである．

5.5 骨格座標の具体的なスライド方法

3.6 節で示した骨格座標のスライドの具体的な方法について説明する．鍵開けとピッキングを被験者 1〜10 が行った際の，左かかとの座標値を**表 5** に示す．表のスペースの都合により，被験者番号を「No.」，全被験者平均を「Ave」と記した．また，表中の X，Y，Z は，それぞれの座標の値のことを指す．なお，スペースの都合で小数点第 3 位まで表中に記したが，実際に Kinect から取得可能な精度は小数点第 4 位までであり，その数値を使用している．

表 5 より，斜め後ろから撮影した場合の全被験者平均は，X 座標 0.3698 [m]，Y 座標 −0.9475 [m]，Z 座標 3.1412 [m] であった．また，真横から撮影した場合の全被験者平均は，X 座標 −0.0068 [m]，Y 座標 −1.0054 [m]，Z 座標 2.8940 [m] であった．これらの値を，Kinect の設置場所に応じて，それぞれの骨格座標からスライドした．

5.6 実験環境

本研究で，ピッキングと鍵開けを行う際の環境としての，人物とドアと Kinect の位置関係を**図 6** に示す．

被験者は 5.1 節で示した位置に立つ．そして，背筋を伸ばし，頭は少し下を向くようにする．Kinect の設置位置としては，ドアがあるため被験者の真正面からの関節座標の取得は不可能なので，斜め背後と真横に設置することを考えた．

表 5 被験者の立ち位置
Table 5 Standing position of examinee.

動作		Kinect 設置：斜め後ろ			Kinect 設置：真横		
	No.	X	Y	Z	X	Y	Z
鍵開け	1	0.925	−1.067	2.976	0.910	−1.079	2.966
	2	0.429	−1.211	2.895	0.480	−1.201	2.919
	3	0.081	−0.874	2.864	0.120	−0.299	2.715
	4	0.041	−0.925	3.199	−0.039	−0.978	3.269
	5	0.352	−0.921	3.287	0.296	−0.934	3.236
	6	0.281	−0.902	3.267	0.296	−0.894	3.263
	7	0.284	−0.931	3.294	0.290	−0.908	3.286
	8	0.311	−0.982	3.224	0.282	−0.947	3.233
	9	0.320	−0.979	3.219	0.305	−0.974	3.242
	10	0.723	−0.976	3.224	0.710	−0.971	3.248
ピッキング	1	0.910	−1.079	2.966	−0.049	−0.978	2.782
	2	0.480	−1.201	2.919	−0.282	−1.309	2.951
	3	0.119	−0.299	2.715	−0.241	−0.565	2.895
	4	−0.039	−0.978	3.269	−0.208	−1.105	2.667
	5	0.296	−0.934	3.236	0.072	−0.995	2.936
	6	0.296	−0.894	3.263	0.034	−1.008	2.872
	7	0.290	−0.908	3.286	0.140	−0.995	2.947
	8	0.282	−0.947	3.233	0.072	−1.057	2.805
	9	0.305	−0.974	3.242	0.054	−0.977	2.967
	10	0.710	−0.971	3.248	0.263	−1.040	2.915
	Ave	0.370	−0.948	3.141	−0.007	−1.005	2.894

まず，Kinect の高さは 185 cm とした．Kinect は被験者に正面に設置するのが推奨されており，Dehbandi らの研究などでも Kinect を正面の最適な距離に設置して評価を行っているが，本研究の環境では，被験者の正面にはつねに扉がありそこには設置できない．そこで，一般的な監視カメラを設置するのと同様に，常識的に考えて，扉の斜め横から斜め下向きに撮影されるようにした．高さを 185 cm としたのは，推奨では被験者の胸の位置くらいになる高さが望ましいが，その位置は出入り時に被験者とぶつかるため，常識的に考えて設置が困難としたためである．位置が高ければ高いほど推奨の高さから外れていくため，極端に高身長の人でなければぎりぎり頭が接触しない程度の高さとした．

次に，Kinect の水平方向の角度は，被験者の斜め右後ろでドアに対して 30 度の位置と，ドアに対して被験者の真横の 2 つとした．被験者の正面から撮影できないのであれば，被験者の真後ろが次に望ましいが，扉からアームを伸ばしてそのように監視カメラを設置するのは常識的に考えて不自然である．さらに，本研究では，体の前にあるドアノブを操作するため，真後ろでは手元が胴体の陰になってしまう．そこで，なるべく被験者の右手側と左手側を同時にとらえられる位置で，ドアからそれほどアームを伸ばさなくてもよい範囲として 30 度の位置を選択した．ただし，家の構造によっては，ドアの前にそれほどアームを伸ばせない場合もある．そこで，被験者の真横の場合も選択している．なお，被験者の真横から撮影する場合には，被験者の左手側は胴体の陰になり撮影できない．被験者の右手側から撮影するのは，人間は右利きが多く，ドアは基本的に右手で開けることが想定されており，右手側を映す必要が

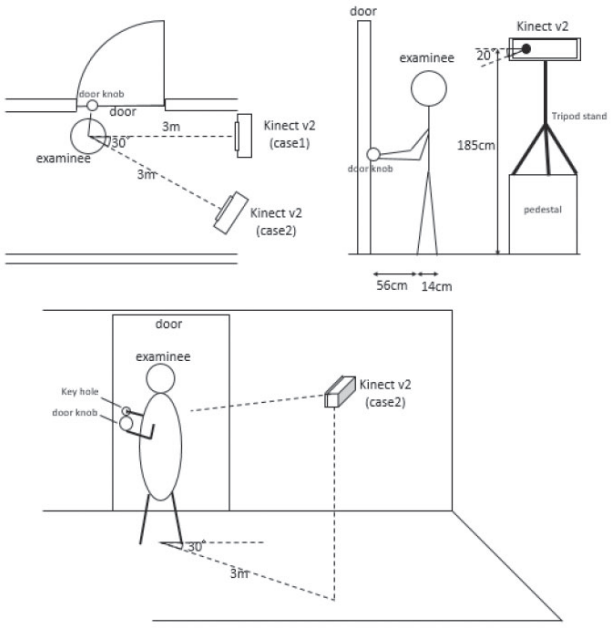

図 6 Kinect により骨格座標を取得する実験環境
Fig. 6 Experimental environment in acquisition of coordinates by Kinect.

あるためである．

仕様上，Kinect v2 の人物の検出範囲は 0.5 m～4.5 m であり，近すぎたり遠すぎたりすると骨格座標が取得できなくなる．そこで，本研究では，Kinect と人物の水平方向の距離が約 300 cm となるようにしている．詳細に説明すると，人間には身長があるため，たとえば身長 160 cm で骨格の中心が上から 80 cm の位置とすると，真上から骨格の中心までが 300 cm の場合，頭の位置は 220 cm（300 − 80）となり，足の位置は 380 cm（300 + 80）となる．実際には真上ではなく斜め上から撮影しているが，手をあげれば頭より上に手がくることもあり，ドアの前で行動する人間の全体が Kincet v2 の検出範囲の中央付近に入るようにすることを考えてこの距離としている．水平方向の角度について補足しておくと，実験場所の廊下の幅が 192 cm であり，被験者から 300 cm の距離をとるとこれ以上の角度を付けることはできない．以上より，この 2 カ所に Kinect を設置することとした．

6. 評価

6.1 被験者情報

本研究の被験者は 20 名存在するが，主として評価に使用する被験者は 10 名であり，残りの 10 名はこの評価を補うために一部の実験に使用している．被験者 20 名の身長を**表 6** に示す．

6.2 マシンスペック

本研究で使用する機械学習用サーバのスペックを以下に示す．CPU の型番は Intel® Core™ i5-7600 CPU である．クロック周波数は 3.50 GHz，コア数は 4 となっている．メモリにおいて，メインメモリのサイズは 64 GB，メモリ使用量は 4.4 GB，メモリの空き容量は 21 GB，共有メモリは 0.1 GB，バッファのキャッシュは 39 GB，ページのキャッシュは 58 GB である．また，ビデオカードは Intel Corporation Device 5912（rev 04）である．本研究で使用した最大のデータ数，データ容量はそれぞれ 5.4 節における線形補間データを生成した際であり，約 2.11 GB である．データの容量上，著者らの環境では約 3GB を超えてしまうと機械学習を行えなくなってしまう．そのため，被験者 20 名すべてを主として評価に使用せず，10 名としている．

表 6 被験者の身長
Table 6 Height of examinees.

被験者番号	身長 [cm]	被験者番号	身長 [cm]
被験者 1	169	被験者 11	177
被験者 2	165	被験者 12	165
被験者 3	163	被験者 13	171
被験者 4	172	被験者 14	166
被験者 5	161	被験者 15	171
被験者 6	171	被験者 16	166
被験者 7	170	被験者 17	164
被験者 8	170	被験者 18	182
被験者 9	170	被験者 19	172
被験者 10	163	被験者 20	168

6.3 エポック数とバッチサイズの決定

3.7 節で述べた CNN を使用する際に，エポック数とバッチサイズを決定する必要がある．そこで，被験者 1～10 のサンプルを用いて，10 分割交差検証により最適なエポック数とバッチサイズを調べた．なお，テスト用サンプルのピッキングと鍵開けのサンプル数は同数となるようにした．また，訓練用サンプルにもテスト用サンプルにも被験者 1～10 の全員を含んでいる．これは，ピッキングを鍵開けと見分ける場合とは異なり，訓練したモデルが最適解に近づくようにするには何エポックを計算する必要があるかを求めるものであるため，訓練用サンプルにテスト用サンプルと同一の被験者が混在していても関係ない．

エポック数は 50 から 50 ずつ増加させ，バッチサイズは 32 から倍々にしていき，1,800 になるまでの値について精度評価を行った．本研究で用いるエポック数とバッチサイズは F 値が 90%を超えたときに標準偏差が 5%以下のものを用いるものとした．**表 7** に，エポック数 50 のときにバッチサイズを変更した場合の精度の結果を示した．また，**表 8** に，エポック数 100 のときにバッチサイズを変更した場合の精度の結果を示した．表のスペースの都合上，表中のバッチサイズを「bat」で表した．また，表中の数値が平均値，括弧書き内の数値が標準偏差を示す．たとえば，0.85 (0.0779) であれば，平均 0.85 で標準偏差が 0.0779 である．

表 7 エポック 50 とバッチサイズとの関係
Table 7 Relationship of 50 epochs to batchsizes.

bat	precision	recall	F 値
32	0.85 (0.0779)	0.83 (0.0962)	0.83 (0.115)
64	0.84 (0.0669)	0.82 (0.0771)	0.82 (0.0853)
128	0.84 (0.0611)	0.83 (0.0670)	0.83 (0.0677)
256	0.80 (0.0558)	0.79 (0.0674)	0.78 (0.0727)
512	0.74 (0.0496)	0.72 (0.0495)	0.72 (0.0540)
1,024	0.70 (0.0328)	0.67 (0.0387)	0.65 (0.0528)
1,800	0.65 (0.0546)	0.62 (0.0426)	0.60 (0.0541)

表 8 エポック 100 とバッチサイズとの関係
Table 8 Relationship of 100 epochs to batchsizes.

bat	precision	recall	F 値
32	0.90 (0.121)	0.89 (0.122)	0.88 (0.152)
64	0.90 (0.0624)	0.89 (0.0749)	0.89 (0.0791)
128	0.91 (0.0485)	0.91 (0.0538)	0.90 (0.0549)
256	0.91 (0.0381)	0.91 (0.0418)	0.90 (0.0416)
512	0.84 (0.0547)	0.83 (0.0589)	0.82 (0.0597)
1,024	0.77 (0.0523)	0.76 (0.0555)	0.75 (0.0578)
1,800	0.72 (0.0397)	0.70 (0.0428)	0.69 (0.0492)

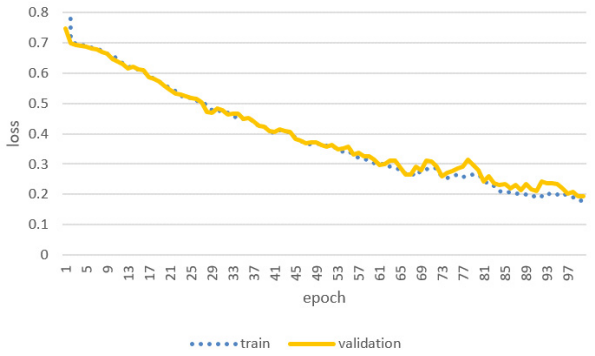

図 7 エポック数とバッチサイズを決定する実験における loss
Fig. 7 Loss at experiments for the best epochs and batchsizes.

表 7 の結果から，エポック数が 50 のときにバッチサイズを変更しても F 値が 90%を超える場合は存在しなかった．表 8 の結果から，エポック数が 100 のとき，バッチサイズが 128，256 の場合に F 値が 90%超を示した．そのなかで，標準偏差が 5%以下となったのはバッチサイズが 256 の場合で，その標準偏差は 4.16%であった．以上の結果から，本研究ではエポック数を 100，バッチサイズを 256 とし，以降の評価を行っていくこととした．

なお，本実験では訓練用サンプルが少ないために過学習を起こしている可能性がある．そこで，訓練（train）と検証（validation）の loss を取得し，**図 7** に示した．その結果，訓練においては，85〜96 エポックで loss がおよそ 0.20 となり，100 エポックで 0.179 まで下がった．検証においては，84〜96 エポックで loss が 0.21 から 0.24 の間を行き来し，100 エポックで 0.194 まで下がった．loss は減少し続けているが，過学習は起きていないことが確認できた．

6.4 サンプルを無加工で用いた場合の精度評価

被験者 1〜10 のサンプルをそのまま使用して CNN の 10 分割交差検証で評価を行った．訓練用サンプルとテスト用サンプルの被験者は異なる．結果を**表 9** に示す．表のスペースの都合により，被験者番号を「No.」，全被験者平均を「Ave」と記した．また，表中の数値が平均値，括弧書き内の数値が標準偏差を示す．

表 9 の結果から，被験者全体の平均の F 値は 56%と低く，その標準偏差も被験者平均で 14.9%とばらつきがあることが分かる．被験者ごとに平均の F 値を見ると，50%台のものから 80%近くのものまであり，一部の被験者に関してはある程度高い精度で分類が行えているが，ほとんどの被験者の分類精度が低い．

6.5 被験者データを元に生成したサンプルを追加した場合の被験者ごとの精度評価

3.4 節の方法で擬似的に被験者数を増やし，どのように精度が変化するか調査した．被験者 1〜10 のサンプルを使用して CNN の 10 分割交差検証で評価を行った．訓練用サンプルとテスト用サンプルの被験者は異なる．なお，テスト用サンプルの被験者のデータは擬似的な訓練用サンプルの生成に使用していない．結果を**表 10** に示す．表記の省略方法は表 9 と同様である．

表 10 の結果から，被験者 10 名の平均 F 値は 59%で分類はうまく行えていない．なお，F 値の平均標準偏差も 18.2%と，表 9 の 14.9%より大きくなっている．しかし，被験者ごとに F 値を見ると，被験者 6 と被験者 10 においてそれぞれ 91%と 82%となっており，うまく分類できている被験者もいることが分かる．これは，55 名程度の被験者を集めて実験を行えば，一部の高精度を示した被験者に関しては精度をより向上させることができるが，工夫をすれば 10 名の被験者でも同程度の精度を出せるということを示唆している．

6.6 フレームの線形補間データを追加した場合の被験者ごとの精度評価

3.4 節の方法で，行動周期が一致もしくは近いサンプルが追加された場合に，どのように精度が変化するか調査し

表 9 被験者ごとの精度評価
Table 9 Evaluation by each examinee.

No.	precision	recall	F 値
1	0.64 (0.128)	0.58 (0.0857)	0.54 (0.0954)
2	0.68 (0.100)	0.62 (0.0673)	0.60 (0.0673)
3	0.56 (0.149)	0.55 (0.127)	0.54 (0.132)
4	0.64 (0.201)	0.53 (0.0736)	0.44 (0.0766)
5	0.76 (0.013)	0.54 (0.041)	0.42 (0.0811)
6	0.85 (0.0623)	0.79 (0.114)	0.76 (0.148)
7	0.56 (0.155)	0.50 (0.0518)	0.42 (0.0891)
8	0.59 (0.0724)	0.58 (0.0619)	0.58 (0.0604)
9	0.60 (0.0949)	0.56 (0.0321)	0.54 (0.0264)
10	0.80 (0.102)	0.76 (0.0978)	0.75 (0.0968)
Ave	0.67 (0.154)	0.60 (0.122)	0.56 (0.149)

表 10 生成したサンプルを追加した場合の被験者ごとの精度評価
Table 10 Evaluation by each examinee with generated samples.

No.	precision	recall	F 値
1	0.58 (0.109)	0.51 (0.172)	0.46 (0.0543)
2	0.66 (0.136)	0.62 (0.115)	0.61 (0.115)
3	0.72 (0.0814)	0.60 (0.0617)	0.55 (0.0926)
4	0.59 (0.215)	0.50 (0.124)	0.44 (0.116)
5	0.67 (0.129)	0.53 (0.0318)	0.41 (0.0730)
6	0.93 (0.0377)	0.91 (0.0545)	0.91 (0.0572)
7	0.61 (0.125)	0.57 (0.0754)	0.53 (0.0851)
8	0.77 (0.0605)	0.66 (0.104)	0.62 (0.140)
9	0.74 (0.105)	0.65 (0.114)	0.61 (0.155)
10	0.84 (0.0453)	0.82 (0.0564)	0.82 (0.0599)
Ave	0.71 (0.158)	0.64 (0.152)	0.59 (0.182)

表 11 線形補間により生成したサンプルを追加した場合の被験者ごとの精度評価
Table 11 Evaluation by linear interpolation for each examinee with generated samples.

No.	precision	recall	F 値
1	0.64 (0.114)	0.55 (0.0593)	0.48 (0.0969)
2	0.59 (0.0868)	0.57 (0.0707)	0.55 (0.0701)
3	0.61 (0.166)	0.53 (0.0981)	0.46 (0.109)
4	0.35 (0.0502)	0.39 (0.0422)	0.35 (0.0473)
5	0.61 (0.152)	0.51 (0.0435)	0.39 (0.0522)
6	0.95 (0.0437)	0.94 (0.0582)	0.94 (0.0641)
7	0.65 (0.125)	0.56 (0.0756)	0.51 (0.103)
8	0.73 (0.0731)	0.62 (0.0707)	0.57 (0.109)
9	0.70 (0.125)	0.62 (0.117)	0.58 (0.151)
10	0.87 (0.0492)	0.83 (0.0809)	0.83 (0.0861)
Ave	0.67 (0.188)	0.61 (0.170)	0.57 (0.199)

た．被験者 1〜10 のサンプルを使用して CNN の 10 分割交差検証で評価を行った．訓練用サンプルとテスト用サンプルの被験者は異なる．なお，テスト用サンプルの被験者のデータは追加される訓練用サンプルの生成に使用していない．結果を**表 11** に示す．表記の省略方法は表 9 と同様である．

表 11 の結果から，被験者 10 名の平均 F 値は 57%で分類はうまく行えていない．なお，F 値の平均標準偏差も 19.9%と，表 9 の 14.9%，表 10 の 18.2%と比較して増加している．しかし，被験者ごとに F 値を見ると，被験者 6 と被験者 10 においてそれぞれ 94%と 83%となっており，うまく分類されている被験者もいることが分かる．なお，表 10 の値よりもこれらは上昇しているが，標準偏差を考慮するとばらつきの範囲内である．よって周期による差は CNN の分類には影響がないことが判明した．

6.7 訓練用サンプルの被験者を変更した場合の被験者ごとの精度評価

6.5 節と 6.6 節の評価方法で高い F 値となった被験者 6 と被験者 10 は，互いの行動が類似している可能性がある．そこで，訓練用サンプルの被験者を変更した場合に，どのように精度が変化するか調査した．被験者 11〜20 のサンプルを訓練用サンプルに，被験者 1〜10 のいずれか 1 名のサンプルをテスト用サンプルに使用して，CNN で評価を行った．これまでの 10 分割交差検証と合わせるため，被験者 1 名のテスト用サンプルを 10 分割して 10 回テストしている．結果を**表 12** に示す．また，SVM の場合にはどのように精度が変化するかも調べるために，同様の条件で SVM においても評価を行った．結果を**表 13** に示す．表記の省略方法は表 9 と同様である．

表 12 の結果から，CNN による被験者 10 名の平均 F 値は 52%で分類はうまく行えていない．なお，その標準偏差も被験者平均で 9.63%とばらつきがある．被験者 6 と被験者 10 の F 値はそれぞれ 52%と 51%になっており，類似の特徴を持つ被験者が訓練用サンプルに含まれていないと，CNN では精度が高くならないことが示された．

表 12 CNN による被験者ごとの追加精度評価
Table 12 Additional CNN evaluation by each examinee.

No.	precision	recall	F 値
1	0.63 (0.113)	0.53 (0.0225)	0.48 (0.0535)
2	0.52 (0.0374)	0.52 (0.0347)	0.51 (0.0332)
3	0.68 (0.126)	0.59 (0.0931)	0.54 (0.122)
4	0.72 (0.111)	0.57 (0.0545)	0.48 (0.0752)
5	0.63 (0.0889)	0.60 (0.0638)	0.57 (0.0776)
6	0.70 (0.134)	0.59 (0.115)	0.52 (0.160)
7	0.53 (0.0490)	0.52 (0.0452)	0.52 (0.0435)
8	0.66 (0.109)	0.56 (0.0681)	0.49 (0.115)
9	0.58 (0.0454)	0.57 (0.0424)	0.57 (0.0406)
10	0.65 (0.100)	0.56 (0.0471)	0.51 (0.0986)
Ave	0.63 (0.117)	0.56 (0.0967)	0.52 (0.0963)

表 13 SVM による被験者ごとの追加精度評価
Table 13 Additional SVM evaluation by each examinee.

No.	precision	recall	F 値
1	0.86 (0.0316)	0.80 (0.0591)	0.79 (0.0688)
2	0.95 (0.0316)	0.94 (0.0441)	0.94 (0.0472)
3	0.82 (0.0367)	0.73 (0.0805)	0.70 (0.101)
4	0.74 (0.0432)	0.51 (0.00490)	0.37 (0.0128)
5	0.82 (0.0125)	0.75 (0.0500)	0.73 (0.0654)
6	0.93 (0.0530)	0.91 (0.0712)	0.91 (0.0741)
7	0.76 (0.0161)	0.71 (0.0624)	0.69 (0.0815)
8	0.86 (0.0512)	0.78 (0.106)	0.76 (0.126)
9	0.86 (0.0324)	0.83 (0.0394)	0.82 (0.0408)
10	0.90 (0.0502)	0.88 (0.111)	0.86 (0.142)
Ave	0.85 (0.0743)	0.78 (0.135)	0.76 (0.174)

一方で，表 13 の結果から，SVM による被験者 10 名の平均 F 値は 76%となっている．標準偏差は被験者平均で 17.4%とばらつきがある．これは，被験者 4 の F 値の平均が 37%となっているせいもあるが，一方で，F 値が 60%台が 1 名，70%台が 4 名，80%台が 2 名，90%台が 2 名となっており，CNN でうまく分類できなかった被験者を SVM ではうまく分類できる可能性があることが示された．

6.8 座標をスライドさせた場合の精度評価

6.7 節において，被験者 6 と被験者 10 の F 値が低下したのは，類似の特徴を持つ被験者が訓練用サンプルに含まれていないからと推測されるが，この類似の特徴とは，行動的特徴ではなく，体格による場合も考えられる．そこで，3.6 節で述べた骨格座標のスライドにより，被験者の体格差をなくした場合に，どのように精度が変化するか調査した．被験者 11〜20 のサンプルを訓練用サンプルに，被験者 1〜10 のいずれか 1 名のサンプルをテスト用サンプルに使用して，CNN で評価を行った．これまでの 10 分割交差

表 14 骨格座標をスライドした場合の CNN による被験者ごとの追加精度評価
Table 14 Additional CNN evaluation by each examinee in slide of action coordinate.

No.	precision	recall	F 値
1	0.62 (0.126)	0.54 (0.0388)	0.51 (0.0386)
2	0.51 (0.0102)	0.50 (0.00800)	0.50 (0.0111)
3	0.60 (0.141)	0.55 (0.0940)	0.52 (0.0923)
4	0.67 (0.0689)	0.56 (0.0252)	0.48 (0.0618)
5	0.45 (0.188)	0.44 (0.0911)	0.38 (0.0954)
6	0.79 (0.0804)	0.67 (0.125)	0.62 (0.167)
7	0.60 (0.0742)	0.56 (0.0550)	0.50 (0.0819)
8	0.73 (0.0498)	0.59 (0.0623)	0.51 (0.119)
9	0.70 (0.124)	0.61 (0.0703)	0.57 (0.0785)
10	0.61 (0.122)	0.56 (0.0735)	0.53 (0.0914)
Ave	0.63 (0.145)	0.56 (0.0934)	0.51 (0.110)

表 15 骨格座標をスライドした場合の SVM による被験者ごとの追加精度評価
Table 15 Additional SVM evaluation by each examinee in slide of action coordinate.

No.	precision	recall	F 値
1	0.76 (0.0323)	0.60 (0.0346)	0.53 (0.0643)
2	0.95 (0.0341)	0.94 (0.0385)	0.94 (0.0406)
3	0.80 (0.0349)	0.66 (0.0887)	0.60 (0.115)
4	0.42 (0.0356)	0.48 (0.0150)	0.36 (0.00831)
5	0.79 (0.0686)	0.71 (0.0402)	0.69 (0.0356)
6	0.95 (0.0422)	0.94 (0.0558)	0.94 (0.0567)
7	0.69 (0.0429)	0.67 (0.0196)	0.65 (0.0210)
8	0.92 (0.0276)	0.91 (0.0385)	0.91 (0.0410)
9	0.85 (0.0199)	0.82 (0.00917)	0.81 (0.0118)
10	0.88 (0.0310)	0.84 (0.0633)	0.83 (0.0682)
Ave	0.80 (0.155)	0.75 (0.157)	0.73 (0.192)

検証と合わせるため，被験者 1 名のテスト用サンプルを 10 分割して 10 回テストしている．結果を**表 14** に示す．また，SVM の場合にはどのように精度が変化するかも調べるために，同様の条件で SVM においても評価を行った．結果を**表 15** に示す．表記の省略方法は表 9 と同様である．

表 14 の結果から，骨格座標のスライドにおける CNN による被験者 10 名の平均 F 値は 51%で分類はうまく行えていない．標準偏差も被験者平均で 11.0%とばらつきがある．表 15 の結果から，SVM の場合の平均 F 値は 73%，標準偏差は被験者平均で 19.2%であった．SVM においては F 値が向上しているように見えるが，標準偏差を考慮すると誤差の範囲内といえる．また，CNN には影響は見られなかった．

7. 考察

提案手法を実運用システムで用いる場合の条件や限界について考察する．

まず，Kinect は太陽光の影響を受けやすいため，屋外での使用が可能かという疑問点について考察する．現状では，Kinect を一般的に利用する場合，太陽光が直接差し込まないように注意が必要であるが，O'Toole らは，画像処理アルゴリズムや追加の処理をすることなく，センサのみで明るい太陽光の下で Kinect の 3D カメラを使えるようにする技術を提案している [15]．また，藤川らは蛍光体粉末を誘導物として利用する歩行支援システムの研究のなかで，光信号の変調・復調，および信号のパルス化によって，受光ユニットが屋外環境下でも影響を受けにくいことを述べている [16]．よって，これらの技術が Kinect で一般的に利用できるようになれば，提案手法を実運用システムで用いる際に，太陽光の影響を受けにくくなると考えられる．

次に，Kinect による骨格座標の位置精度が，人物を正面から撮影した場合には高く，横や後ろから撮影した場合は低いのではないかという疑問点について考察する．Otte らは，研究のなかで Kinect v2 の精度について評価している [17]．そのなかで，それぞれの方向の SN 比を評価しているが，たとえば，Head であれば anterio-posterior（AP）方向[*2]で 31.69，medio-lateral（ML）方向[*3]で 7.90，vertical（V）方向で 4.70[*4]となっており，どの座標でも総じて SN 比が高いのは AP 方向である．よって，人物をどの向きから撮影するかではなく，その人物の動きが AP 方向であれば精度は高いことになる．

また，提案手法において，X，Y，Z 座標を扱う際，行方向は時間的変化であるため隣接する値に近接性があるが，列方向はそれぞれの骨格座標の値であるため近接性がない．この場合，CNN を用いても意味がないのではないかという疑問点について考察する．列方向のそれぞれの骨格座標の値には相対的な距離があるため，CNN を用いる際に意味がある．たとえば，頭と右手の距離であれば，畳み込みに影響を与える．

最後に，ファイルサイズについて考察する．4.2 節で，提案手法を実装して新たに擬似的なサンプルを生成する際に，1 サンプルあたりのファイルサイズが大きくなっている場合がある．これは，ファイルが CSV 形式であるためであり，小数点以下のどこまでの数字を保持するかや，マイナス記号の有無などによる影響である．たとえば，0.1 と 0.2 の平均は 0.15 であり，元の値より 1 文字分増加するためにファイルサイズが大きくなる．サンプルを機械学習で扱う際には，数値は float 型の配列に格納されるため，1 サンプルあたりのファイルサイズはメモリ容量には影響しない．

なお，使用するライブラリの都合上，我々の実験では訓練に使用可能なデータ容量には制約がある．5.4 節で，フレームの線形補間によりサンプルを生成しているが，これ

[*2] Kinect に対して近づいたり遠ざかったりする方向
[*3] Kinect に対峙した場合の左右方向
[*4] 鉛直方向

が我々が訓練で扱えるデータ量の限界である．サンプルを擬似的に増加させずに訓練を行うのであれば，同時に 20 名の被験者のサンプルを扱うことも可能であるが，それでは他の方法と精度の比較ができなくなるため今回は行っていない．また，3.4 節で述べた被験者データを元にしたデータ生成手法において，被験者間の座標の値を 2 で割った値を用いているが，これは，2 を用いるのが，増加するサンプル数が最も少ないためである．この値があまり大きくなると，我々の環境では訓練が行えなくなる．なお，3 で割った値を用いてもデータ量の限界は超えないが，2 で割った値を用いてサンプル数を擬似的に増加させた結果が芳しくなかったため，今回は試していない．

提案手法を実用化する際には，訓練の際にデータ量の制約を受けたくないのであれば，より大きなデータを扱えるライブラリを使用するという方法がある．我々の実装のまま実用化するのであれば，たとえば，10 人ずつ訓練に使用したフィルタを用意し，一定数のフィルタでピッキングと判断された場合には，警備員が監視カメラの映像を確認するという方法も考えられる．

8. まとめ

監視カメラは犯罪の抑止や事件後の犯人の特定には役立つが，家に侵入している犯人をその場で捕らえるためには，常時有人監視しなければならない．そこで本論文では，Kinect の骨格座標のみを用いて，家のドアから不正に侵入する行為を検知するシステムを提案した．そのなかで，従来研究では高精度で分類できるかどうか示されていない，ピッキングと鍵開けという類似の動作に注目し，少ない被験者数で分類精度について評価した．SVM を用いた評価では，平均 F 値は 76%や 73%となったが，一部の被験者 10 名のうち 3 名が 91%，94%，94%と高精度で分類された．CNN においても，類似の行動的特徴を持つ被験者のデータが訓練用サンプルに含まれている場合に，高い精度が出ることが判明した．

本システムでは，映像と比較して，Kinect の骨格座標のデータは小さいため，大量のデータを各家庭からセンタに送信しても，データ転送がトラヒックに与える影響は映像よりも小さい．さらに，全家庭がセンタに映像を送信する必要もなく，センタにデータを提供しない家庭は，センタへの通信設備も不要である．以上より，提案手法およびシステムは，家庭の防犯対策に有用であるといえる．

謝辞 本研究を進めるにあたり，ディスカッションに参加いただいた長名優子博士に，心より謝意を表する．

参考文献

[1] NIKKEI STYLE：五輪控え AI で不審者検出　三菱電機，高齢者ら支援も，入手先 ⟨http://style.nikkei.com/article/DGXMZO10563380S6A211C1000000?channel=DF220420167276⟩（参照 2017-01-27）．

[2] Pang, J.M., Yap, V.V. and Soh, C.S.: Human Behavioral Analytic System for Video Surveillance, *Proc. IEEE International Conference on Control System, Computing and Engineering* (*ICCSCE 2014*), pp.23–28 (2014).

[3] Horiuchi, Y., Makino, Y. and Shinoda, H.: Computational Foresight: Forecasting Human Body Motion in Real-time for reducing Delays in Interactive System, *Proc. 2017 ACM International Conference on Interactive Surfaces and Spaces* (*ISS '17*), pp.312–317 (2017).

[4] 中原啓太，山口弘純，東野輝夫：移動型センサと kinect を用いた家庭内の行動ロギング手法，情報処理学会関西支部支部大会，G-05 (2016).

[5] 中島雅貴，小篠裕子，斎藤英雄：時系列情報を考慮した人体骨格追跡と評価，電子情報通信学会技術研究報告，Vol.117, No.392, pp.267–270 (2018).

[6] 渡邊昭信，味松康行，村上俊夫，中村克行：上方視点距離画像を用いた人物姿勢推定手法の検討，情報処理学会研究報告，Vol.2017-CVIM-209, No.29, pp.1–13 (2017).

[7] 森　駿文，菊池浩明：深度センサによる歩容特徴量を用いた個人識別・追跡方式の提案，情報処理学会コンピュータセキュリティシンポジウム（CSS2017），pp.972–979 (2017).

[8] Dehbandi, B., Barachant, A., Harary, D., et al.: Using Data From the Microsoft Kinect 2 to Quantify Upper Limb Behavior: A Feasibility Study, *IEEE Journal of Biomedical and Health Informatics*, Vol.21, No.5, pp.1386–1392 (2017).

[9] kamomako: Smart Life Net, available from ⟨http://kamomako.hatenablog.jp/entry/kenkoutisiki/taijyuu-heikin-nenreibetu⟩ (accessed 2018-11-22).

[10] Statistics Japan: PHYSIQUE OF YOUTHS BY AGE (2000-14), available from ⟨https://www.stat.go.jp/data/nenkan/65nenkan/zuhyou/y652402000.xls⟩ (accessed 2018-11-24).

[11] 山口弘純，安本慶一：エッジコンピューティング環境における知的分散データ処理の実現，電子情報通信学会論文誌，Vol.J101-B, No.5, pp.298–309 (2018).

[12] 川村隆浩，ワ コラ，中川博之，田原康之，大須賀昭彦：インタラクションシーケンスに着目した商品検索目的抽出エージェントの開発，電子情報通信学会論文誌，Vol.J94-D, No.11, pp.1783–1790 (2011).

[13] 川村隆浩，越川兼地，中川博之，清　雄一，田原康之，大須賀昭彦：メディア情報の Linked Data 化と活用事例の提案，電子情報通信学会論文誌，Vol.J96-D, No.12, pp.2987–2999 (2013).

[14] 杉浦　司：Kinect for Windows v2 入門―C++プログラマー向け連載，入手先 ⟨https://www.buildinsider.net/small/kinectv2cpp⟩（参照 2018-11-22）．

[15] O'Toole, M., Achar, S., Narasimhan, S. and Kutulakos, K.: Homogeneous Codes for Energy-Efficient Illumination and Imaging, *ACM SIGGRAPH* (2015).

[16] 藤川真樹，橋本　就，渕　真悟：蛍光体粉末を誘導物として利用する歩行支援システムの研究，産業応用工学会論文誌，Vol.5, No.2, pp.63–74 (2017).

[17] Otte, K., Kayser, B., Mansow-Model, S., Verrel, J., Paul, F., Brandt, A.U. and Schmitz-Hübsch, T.: Accuracy and Reliability of the Kinect Version 2 for Clinical Measurement of Motor Function, *PLoS One*, Vol.11, No.11 (2016).

白石 将貴

2017 年東京工科大学コンピュータサイエンス学部コンピュータサイエンス学科卒業．2019 年同大学大学院バイオ・情報メディア研究科コンピュータサイエンス専攻博士前期課程修了．現在，CTC テクノロジー株式会社勤務．

宇田 隆哉 （正会員）

1998 年慶應義塾大学理工学部計測工学科卒業．2000 年同大学大学院理工学研究科計測工学専攻前期博士課程修了．2002 年同大学院理工学研究科開放環境科学専攻後期博士課程修了．博士（工学）．現在，東京工科大学コンピュータサイエンス学部講師．ネットワークセキュリティの研究に従事．2002 年 IFIP/SEC 2002 Best Student Paper Award 受賞．電子情報通信学会会員．

藤川 真樹 （正会員）

1994 年詫間電波工業高等専門学校情報工学科卒業．1996 年徳島大学工学部知能情報工学科卒業．1998 年同大学大学院工学研究科知能情報工学専攻博士前期課程修了．2004 年中央大学大学院理工学研究科情報工学専攻博士後期修了．博士（工学）．1998～2016 年綜合警備保障株式会社勤務．2016 年 4 月より工学院大学情報学部コンピュータ科学科准教授．情報セキュリティ，認証，セーフティ，人工物メトリクスの研究に従事．