Vol.2022-DPS-190 No.51
Vol.2022-CSEC-96 No.51
2022/3/11

ゲームで得られたPC上のマウスの軌跡を用いた個人識別の検討

山田 裕貴[1]　佐藤 聡[3]　新城 靖[2]　星野 厚[1,4]

概要：マウスからの通信が自分の意図したものでないと，コンピュータは意図しない不正な制御が行われ，安全性が低下してしまうため，マウスからの通信が自分自身の操作によるものかを調査することは重要である．そこで，マウスからの通信データを用いて機械学習を行い，入力したデータが自分自身の操作であるかどうかを判定する識別器の作成を検討している．本検討では，上記識別器を作成するにあたって，マウスからの通信データに合う適切な機械学習手法やそれに入力するデータへの加工方法を調べるために行った予備実験の結果を報告する．予備実験では，著者が 3 種類のゲームの軌跡をプレイした際のマウスからの通信データ用いて，入力するデータが特定のゲームをプレイした時のマウスの軌跡であるかどうかという識別器を作成し，精度を測った．

1. はじめに

サイバー空間で安心安全に生活するためには，利用者は自分が使用しているコンピュータが，自分の意図通りに動作しているかを意識することが重要になってきている．現在利用されているコンピュータの多くは，入力デバイスとは別の機器となっていて，その間の通信は，USB などの企画で定められたプロトコルに従っている．この時，マウスからの通信が，自分の意図したものでないと，コンピュータは意図しない不正な制御が行われ，安全性が低下してしまう．例として，マウスが他人によって操作されている場合や，マウスが不正に改造されている場合が考えられる．従って，マウスからの通信が自分自身の操作によるものかを調査することは重要である．

マウスの操作を用いたバイオメトリクス認証として，マウスダイナミクス認証と呼ばれる認証方式が研究されている [1]．また，入力機器の一つであるキーボードを用いたバイオメトリクス認証は，記録されるデータにユーザのパスワードなどが含まれているため，プライバシーの問題に発展する可能性があるが，マウスダイナミクス認証方式は，プライバシーに関わる記録されるデータが少ないことで注目を集めている．さらに，認証後も継続的にそのマウスの操作がその人のものかを再認証する継続的なマウスダイナミクス認証方式の研究が行われている [2][3]．こういったマウスダイナミクス認証方式では，マウスの操作データを取得する際には，頻繁にマウスを操作するゲームアプリケーションの利用が好ましい [4]．しかし，これらの研究では，コンピュータ内部でプログラムやロガーを動かし，マウスの軌跡を得ているが，実際には不正な制御が行われている可能性があるコンピュータ側からマウスの操作データを信用することは難しい．

そのため，マウスとコンピュータが USB 規格のプロトコル上で通信していることから，コンピュータ上で使用しているゲームアプリケーション毎で，自分自身によるマウスの操作によって発生した通信であるかを判定する識別器の作成を検討している．本検討では，上記識別器を作成するにあたって，マウスからの通信データに合う適切な機械学習手法やそれに入力するデータへの加工方法を調べるために行った予備実験の結果を報告する．

2. 先行研究

我々の研究グループでは，マルウェアの感染やコンピュータの乗っ取りなどで普段とは異なる”自己らしくない”通信を行う，サイバー空間での脅威をいち早く発見するために，アプリケーション識別機能を有するファイアーウォールのログから識別対象者の通信の振る舞いを学習し，ある通信が自己らしい通信であるか否かを識別する手法を提案した [5]．先行研究では，ネットワークとコンピュータ間に設置されているファイアウォールにて判別したアプリケーション名を利用して自己らしい通信であるかどうかを判

1 筑波大学大学院理工情報生命学術院システム情報工学研究群情報理工学位プログラム
2 筑波大学システム情報系
3 筑波大学学術情報メディアセンター
4 株式会社チノウ

定しているため，アプリケーション名を判定するファイアウォールが設置されていない場合には自己らしい通信であるかどうかを判別できない．

3. 提案手法

コンピュータ上で使用しているゲームアプリケーション毎のマウスの操作によって発生した通信を入力として，それが自分自身によるものであるかを判定し，その結果を出力する識別器の作成を行う．**図 1** に提案手法の概要を示す．マウスの軌跡データは，マウスからの通信データから取得できる．次にそのマウスの軌跡データを入力データとして加工する．学習フェーズでは，その入力データを用いて機械学習を行うことで，マウスを操作した対象者用の識別器を作成する．判定フェーズでは，この識別器に，学習フェーズと同様のデータを入力すると，その入力データが，対象者が行なったマウスの操作によるものであるかどうかを判定する．本検討では，アプリケーションには，頻繁にマウスを操作するゲームを用い，USB 規格で通信するマウスを利用する．

3.1 マウスからのデータの取得方法

USB マウスとコンピュータの通信を監視し，USB マウスからの通信データを取得するために，USBPcap[*1]を用いる．USBPcap を用いると，マウスのクリックやボタンの押下状態，水平方向のマウスの移動量，垂直方向のマウスの変化量，ホイールの変化量を含む USB マウスからの通信データを取得することができる．本検討では，水平方向を X 方向，垂直方向を Y 方向と記す．

また，マウスが操作されていない時にはマウスからデータが送信されないため，得られたマウスからのデータを各機械学習手法の入力に加工する前に，マウスが操作されていない時，X・Y 方向のマウスの移動量をどちらも 0 として埋めた．その例を，**図 2** に示す．この際，マウスから一定のポーリングレートでマウスからデータが送信されるため，USBPcap で観測されたマウスからのデータの受信時間を信用して，ある時点でマウスからのデータが送られているか送られていないのかを判断した．このポーリングレートは，一般的に 1000Hz，つまり，1ms に 1 回であるマウスが多いが，マウスごとにこの値が調整できたり，1000Hz でないものもある．また，この工程で作成されたデータをマウスの軌跡データと呼ぶ．

3.2 学習フェーズの課題

学習フェーズの課題として，マウスの軌跡データを用いて対象者用の識別器を作成する際の機械学習手法と，マウスの軌跡データから機械学習に入力するデータへの加工方法の中から，最も良い識別精度となる組み合わせを選択することが挙げられる．また，機械学習手法とこの入力への加工方法の組み合わせは，いくつか存在する．

現在検討している機械学習手法は，以下の通りである．

(1) ディープニューラルネットワーク
(2) リカレントニューラルネットワーク
(3) サポートベクタマシン
(4) ランダムフォレスト

特に，リカレントニューラルネットワークは，先行研究 [5] で用いられた手法で，時系列データへの識別精度が高い．また，ランダムフォレストや，サポートベクタマシンは，マウスダイナミクス認証の研究 [2][3][4] において使用されていた機械学習手法である．

次にデータの加工方法は次の 3 つで構成される．

Step 1 マウスの軌跡データのうち，どれを用いるか．

Step 2 マウスの軌跡データをどのような組で分割するか．

Step 3 分割された組の要素をそのまま使うか，統計処理をするか．

Step 1 では，マウスの軌跡データのうち，X 方向のマウスの移動量，Y 方向のマウスの移動量，マウスのクリックやボタンの押下状態，ホイールの変化量のどれを用いるかを検討する必要がある．また，X と Y との移動量を**図 3** を用いて角度と移動距離に変換する方法も考えられる．

Step 2 では，マウスの軌跡データのうち，1 組の入力データとして使用するデータの開始位置と終了位置（範囲）を検討する必要がある．現在は，以下の 5 つの方法を検討している．

(1) 連続するマウスの軌跡データのうち，N 個を 1 組の入力データとする．
(2) クリックされた位置を開始位置とし，それから N 個目のデータを終了位置する．
(3) マウスが動き始めたと判定された位置を開始位置とし，それから N 個目のデータを終了位置とする．
(4) クリックされた位置を開始位置とし，もう一度クリックされた位置を終了位置とする．
(5) マウスが動き始めたと判定された位置を開始位置とし，それからマウスが止まったと判定された位置を終了位置とする．

(1)，(2)，(3) については，1 組に含まれる要素数が N 個と固定長となる．一方，(4)，(5) についてはマウスの使われ方によって 1 組に含まれる要素数が可変となる．

Step 3 では，使用する 1 組の入力データを，得られたデータのまま使用するのか，それらの統計量を使用するのかを検討する必要がある．特に，統計量を用いる場合，どういった統計量を使用するかをも検討する必要がある．例

[*1] https://desowin.org/usbpcap/

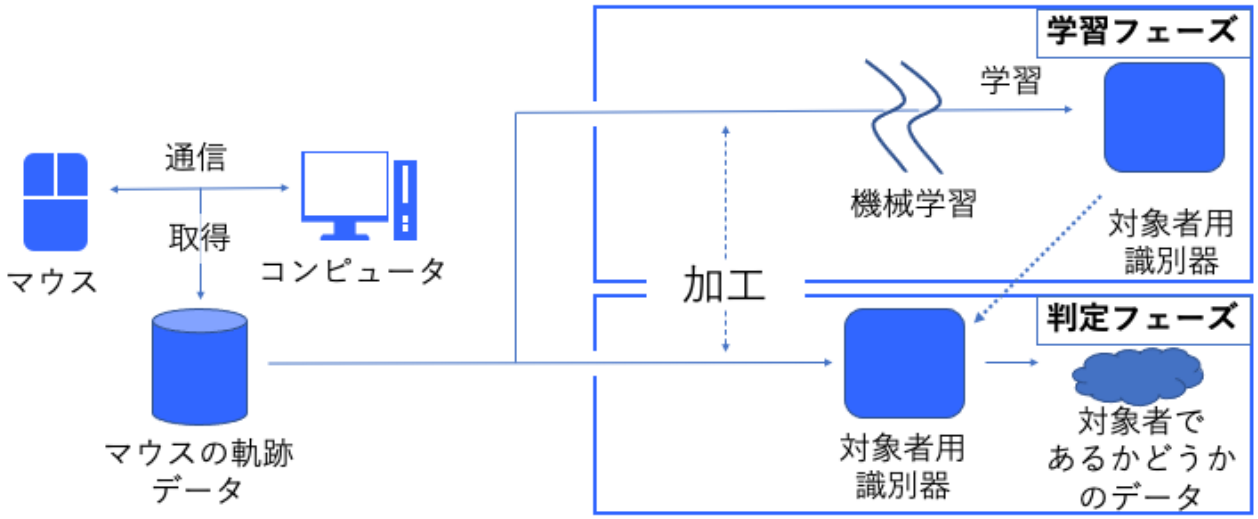

図 1 提案手法

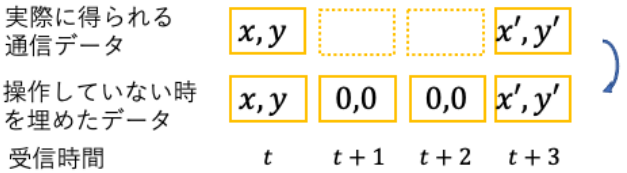

図 2 マウスの軌跡データ

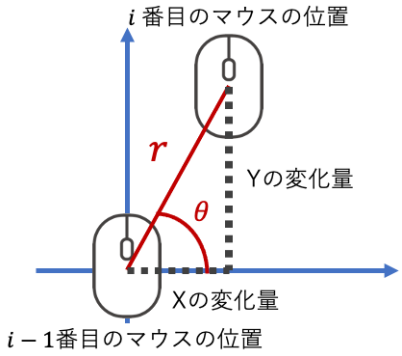

図 3 角度 θ と移動距離 r

えば，継続的なマウスダイナミクスの研究 [2] では，マウスが動いた距離，角度，速度の平均，分散，歪度を使用している．このような統計量に変換する利点として，可変長データを固定長データにすることができる．例えば，1 組の入力データがマウスが動いていると判定される間のデータである場合，これは可変長となるが，統計量は固定長になる．検討している 4 つの機械学習手法では，1 組の入力の長さが固定長でないと学習できないため，**Step 2** における (4),(5) については，統計量に変換する必要がある．

4. 予備実験

本検討の予備実験では，ディープニューラルネットワークとランダムフォレストを使用した．また，マウスの軌跡データのうち，X 方向のマウスの移動量，Y 方向のマウスの移動量のみを使用した．また，データの範囲として，連続するマウスの軌跡データのうち，$N = 1000, 2000, 5000, 10000, 20000$ 個を 1 組の入力データとする．ディープニューラルネットワークでは，通信データから得られた X 方向,Y 方向の移動量を正規化したものを，ランダムフォレストでは，X 方向，Y 方向の移動量を移動距離 r と角度 θ に変換し，r と θ の N 個の平均，分散，最大，最小を用いている．

実験内容は，著者のうち 1 名が 3 種類のゲームを操作し，それから得られたマウスの軌跡データを加工して得られる入力データから，その入力が，あるゲームであるかどうかを判別する識別器を作成し，評価することである．この予備実験に対して，本検討で提案する手法では，対象者があるアプリケーションを操作した際のマウスからの通信データから得られたマウスの軌跡データを加工して得られる入力データから，対象者が行なったマウスの操作によるものであるかどうかを判別する識別器を作成することである．これらの違いは，著者によってあるゲームを操作したマウスからの通信によって，入力データがあるゲームかどうかを判別する識別器なのか，対象者があるアプリケーションを操作したマウスからの通信から，入力データが対象者かどうかを判別する識別器なのかという点であるが，ゲームごとにマウスの操作が異なるため，ゲームごとにそれぞれ違う人が操作していると仮定し，予備実験を行なった．また，3 種類のゲームを操作して得られたマウスの軌跡データを全てひとまとまりにし，あるゲームであるもの，あるゲームでないものと 2 種類のラベルをつけ，そのうち，8 割を識別器を作成するのに使用し，残りの 2 割を識別器の評価に用いた．また，X・Y 方向の変化量は，マウスごとに単位が変わってくるため，本検討では，G703（Logicool 社製）を使用した．このマウスのポーリングレートは，個人で設定可能であるが，初期設定である 1000Hz を使用する．

4.1 対象としたゲーム

対象としたゲームは，非対戦ゲームの Minecraft[*2], 5 対 5 の対戦シューティングゲームの Valorant[*3], 1 対 4 の対戦ゲームで鬼ごっこのような Dead by Daylight[*4]である．なお，Dead by Daylight は，陣営の違いによる操作の違いがあるため，1 人側の陣営のみを対象とした．また，これらのゲームで共通する操作は，視点移動にマウスの操作を用いる点である．視点は，ゲーム内での利用者の自身の見え方を表している．マウスを水平，垂直方向に移動させると，ゲーム上での視点も水平，垂直方向に移動する．水平方向の視点移動は制限はないが，垂直方向の移動は，視点方向の正面部分でしか移動できない．

4.2 予備実験の評価方法

予備実験は，マウスの軌跡のデータを入力した際の識別結果から，精度 acc は，全入力データの個数を n, 同じラベルに分類された入力データの個数を $\hat{n}$ とすると，以下の式で定義される．

$$acc = \frac{\hat{n}}{n}$$

4.3 ディープニューラルネットワーク

ニューラルネットワークは，入力層，隠れ層，出力層で構成され，各層にはノードと，各層間のノードをつなぐ重みで構成されている．入力層のノード数は，入力するデータの次元と同じ個数必要で，出力層のノード数は分類するクラスの数作成する．隠れ層のノードや隠れ層の数は自由に設定可能である．特に，隠れ層の数が 4 つ以上のものをディープニューラルネットワークと呼んでいる．今回は，それぞれのゲームごとに，隠れ層を 10 層作成し，各層のノードの数を 100 とする．

次に入力データについて説明する．連続するマウスの軌跡データのうち，$N = 1000, 2000, 5000, 10000, 20000$ 個を 1 組の入力データとし，その値を正規化したものを用いているが，1 組の入力データのうち，N 個全てが 0 ならば，それは使用しないことにした．また，N 個に満たないデータも使用しない．（**図 4**）

表 1 に，ディープニューラルネットワークで作成された各識別器の精度を示す．

4.4 ランダムフォレスト

ランダムフォレストは，まず，入力データを，いくつかのグループに分ける．その際，ある入力データが複数のグループに属する可能性もある．その後，各グループごとに決定木を作成する．分類の方法は，それぞれの決定木で予測を行い，最も多い結果を採用する．今回は，それぞれのゲームごとに，ランダムフォレスト内で作成される決定木の数を 1000，各木の最大の深さを 5 とする．

次に，入力データについて説明する．ディープニューラルネットワークで使用した入力データのうち，1 組の X, Y の変化量から，1 組の角度 θ, 移動距離 r に変換する．例えば，ニューラルネットワークで使用された入力データが $N = 1000$ 個の連続したマウスの軌跡データを用いている場合，角度 θ, 移動距離 r はそれぞれ 1000 個に変換される．次に，角度 θ, 移動距離 r の平均値，分散，最大値，最小値を求め，計 8 個の統計量を 1 組の入力として用いた．角度 θ, 移動距離 r と，X・Y 方向の変化量の関係を図 3 に示す．

表 2 に，ランダムフォレストで作成された各識別器の精度を示す．

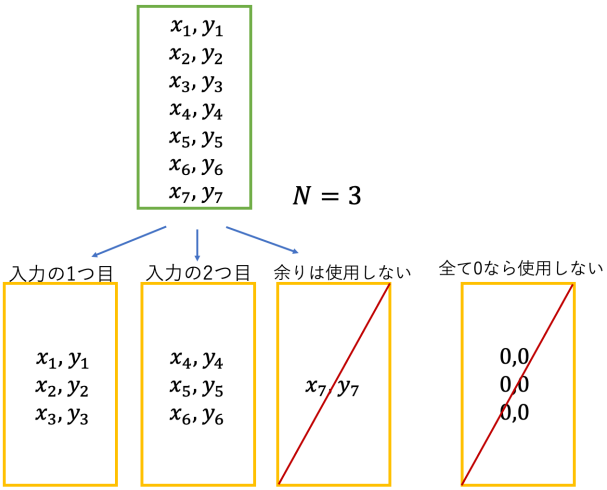

図 4 ディープニューラルネットワークで用いた入力データの加工方法（例 $N = 3$ の場合）

表 1 ニューラルネットワークにおける，N ごとの各ゲームであるかどうかの識別器の精度

N	Minecraft	Valorant	Dead by Daylight
1000	0.601	0.836	0.766
2000	0.601	0.833	0.768
5000	0.600	0.829	0.771
10000	0.599	0.825	0.774
20000	0.402	0.775	0.823

表 2 ランダムフォレストにおける，N ごとの各ゲームであるかどうかの識別器の精度

N	Minecraft	Valorant	Dead by Daylight
1000	0.638	0.849	0.766
2000	0.649	0.856	0.768
5000	0.666	0.859	0.772
10000	0.739	0.868	0.783
20000	0.748	0.866	0.806

4.5 考察

ディープニューラルネットワーク，ランダムフォレストでどちらにおいても，非対戦型ゲームの Minecraft であるか

[*2] https://www.minecraft.net/
[*3] https://playvalorant.com/
[*4] https://deadbydaylight.com/

の識別器の精度と対戦型ゲームの Valorant，Dead by Daylight であるかの識別器の精度に対して低かった．これは，対戦型のゲームでは，ゲームでの勝利のために適応した同じような行動をすることが考えられ，逆に非対戦ゲームでは，自分の好きなように行動できることが考えられるため，精度に違いが出たと考察している．

ランダムフォレストにおいて，Minecraft であるかの識別器の精度は $N = 20000$ のとき，0.748 であり，ディープニューラルネットワークにおける精度と違いが見られた．この理由として，ディープニューラルネットワークのパラメータ設定が不適切であることも考えられる．また，Valorant，Dead by Daylight であるかの識別器の精度 0.866,0.806 とほぼ同等であり，非対戦ゲームと対戦ゲームの区別なく同程度の精度で識別できた．

また，マウスを動かし始めた始点やマウスを動かすのをやめた終点などの特徴量や，マウスのクリックなどのマウスの水平方向，垂直方向の移動量以外の特徴量が，全体的に低かった Minecraft で重要であるのではないかと考察している．

4.6 今後の課題

予備実験のとして，ディープニューラルネットワークと 1 種類の入力データの加工方法，ランダムフォレストと 1 種類の入力データの加工方法の 2 種類のみしか実験ができていない．そのため，他の機械学習手法と入力データの加工方法の組み合わせで実験することを課題とする．今回，ディープニューラルネットワークで用いた入力の加工方法として，図 4 の方法を試した．この方法では，ある入力が全て 0 だった場合は使用しないことにしたが，これだと一定時間以上の”マウスを動かしていない状態”を捨てていることになる．また，N ごとにデータを区切っているため，マウスを動かし始めた点，動きを止めた点が分からず，入力データごとに一貫性がない．これらを改善して，一定時間以上の”マウスが動いていない状態”を用いたり，マウスの動き始めから止まるまでのデータを用いることで，より精度が上がると考察しているため，より良いデータの加工方法を検討したい．また，ランダムフォレストで用いた入力の加工方法として，図 3 の r と θ の平均値，分散，最大値，最小値を求めた．これらは，あるポーリングレートで得られたマウスの X 方向の移動量とマウスの Y 方向の移動量から求まるので，r,θ のポーリングレート当たりの変化量，つまり r,θ の速度の統計量であることがわかる．しかし，r,θ のポーリングレート当たりの変化量がどう変化していくのか，つまり，r,θ の加速度の統計量はない．この加速度以外でも，統計量の追加を行うことで，統計量を用いた時の１つの入力データの特徴量が増え，より精度が上がると考察しているため，さまざまな統計量を用いたい．さらに，特徴量として，マウスの X,Y 方向の移動量だけでなく，マウスのクリック，ホイールの変化量などの統計量を用いての実験を検討している．現在，機械学習手法として，時系列データに特化したリカレントニューラルネットワークなどの手法を用いて，実験を行なっている．さらに，今回用いた精度は，テストデータの精度を表しているが，先行研究では，ROC 曲線を用いた評価が行われている．さらに，マウスダイナミクス認証の研究 [2][3][4] では，識別器の誤認率が重要であり，等価エラー率などによる評価を行なっていた．これらを参考に，今後の評価方法を定めていきたい．

5. 関連研究

Zi Chu らは，web ページに埋め込まれたマウスやキー操作のロガーを使用して，ブログにコメントを投稿するスパムボットを検出する方法を定義した [6]． 本検討では，ボットであるかどうかではなく，自分かどうかを識別することで，プログラムに操作される場合だけでなく，他者によって操作される場合も検出を目指している．

継続したマウスダイナミクス認証についての研究を行なった 3 つの研究を紹介する．Maja Pusara らは，Windows の標準ブラウザである Internet Explorer を作業する，18 人のユーザのマウスの操作ログを用いて，この認証のための識別器を作成した [2]．この操作ログは，各ユーザごとに周波数 k（k はユーザごとに任意）でマウス操作を記録し，記録されたログ間で，距離、角度、および速度を計算する．その後，M 個の距離、角度、および速度から，その平均，分散，歪度を求めて，それを識別器の入力とし，機械学習手法として，サポートベクタマシン用いて識別器を作成した．次に Nan Zheng らは，30 人の各ユーザに様々なタスクを行なってもらい，ロガーを用いてマウス操作の記録を取得し，入力データとして，マウスを動かしてクリックするまでの一連の動作のうち，マウスを動かす方向やマウスが動いた軌跡の曲率を用い，機械学習手法としてサポートベクタマシンを利用して，識別器の作成を行なった [3]．また，Nyle Siddiqui らは，10 人が Minecraft を 20 分プレイした時のマウス操作のログを python プログラムで取得し，マウスの移動量，クリック，ドラッグアンドドロップ，ホイール操作の速度、加速度、および移動方向を入力とし，機械学習手法として，ランダムフォレストを用いて識別器を作成した [4]．これらのマウスダイナミクス認証の研究では，評価方法に等価エラー率を用いて評価していた．

6. おわりに

本検討では，コンピュータ上で使用しているゲームアプリケーション毎で，自分自身によるマウスの操作によって発生した通信であるかを判定する識別器を作成するための機械学習手法と，入力データ作成方法を議論した．機械学

習手法としてはディープニューラルネットワークなどの 4 つを検討している．また，入力データの作成方法は，マウスの軌跡データのうちのどれを用いるか，マウスの軌跡データをどのような組みで分割するか，分割された組みの要素をそのまま使うか，統計処理されたものを使うかを検討する必要がある．そのため，予備実験として，著者のうち 1 名が 3 種類のゲームを操作し，それから得られたマウスの軌跡データを加工して得られる入力データから，その入力が，あるゲームであるかどうかを判別する識別器を作成し，評価した．この予備実験では，この機械学習手法と入力データの作成方法の組み合わせのうち，2 つを行った．機械学習手法として，ディープニューラルネットワークとランダムフォレストを用いた．また，入力データへの加工方法として，ディーニューラルネットワークに使用したものは，N 個の連続したマウスの軌跡データの X・Y の変化量 x, y を 1 組として用い，ランダムフォレストで用いたものは，ディープニューラルネットワークで使用した入力データの X・Y の変化量 x, y を移動距離 r と角度 θ に変換し，それの平均，分散，最大，最小を用いた．結果として，ディープニューラルネットワークとその入力の組み合わせよりもランダムフォレストとその入力の組み合わせの方が全体的に精度が良いことがわかった．1 つの入力データをそのまま用いるより，それを統計量に変換したものを入力データに変換した方が良い可能性があるが，同じ入力データと異なる機械学習手法の実験を行なっていないため，今後検証する必要がある．また，ゲームごとに，そのゲームであるかの識別精度に差が出ていたことがわかり，Valorant，Dead by Daylight，Minecraft の順で識別精度がよかった．

今後の課題として，予備実験で行えなかった機械学習手法と入力データ作成方法の組み合わせで実験を行い，考察することだ．その中でも，入力データの作成方法は，予備実験で欠落していた”マウスが動いていない状態”を上手く組み込むこと，X・Y の変化量だけでなくクリックやホイールの変化量を加味した入力データを作成することも課題である．

参考文献

[1] Kenneth Revett, Hamid Jahankhani, Sérgio Tenreiro de Magalhães, and Henrique M. D. Santos. A survey of user authentication based on mouse dynamics. In Hamid Jahankhani, Kenneth Revett, and Dominic Palmer-Brown, editors, *Global E-Security*, pp. 210–219, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[2] Maja Pusara and Carla E. Brodley. User re-authentication via mouse movements. In *Proceedings of the 2004 ACM Workshop on Visualization and Data Mining for Computer Security*, VizSEC/DMSEC '04, p. 1–8, New York, NY, USA, 2004. Association for Computing Machinery.

[3] Nan Zheng, Aaron Paloski, and Haining Wang. An efficient user verification system via mouse movements. Vol. 139–150, pp. 139–150, 10 2011.

[4] Nyle Siddiqui, Rushit Dave, and Naeem Seliya. Continuous user authentication using mouse dynamics, machine learning, and minecraft. In *2021 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, pp. 1–6, 2021.

[5] 市之瀬樹生, 佐藤聡, 新城靖, 三宮秀次, 星野厚. アプリケーション識別機能付きファイアウォールのログを対象とした機械学習による自己らしい通信の識別手法. 情報処理学会論文誌, Vol. 62, No. 3, pp. 838–847, mar 2021.

[6] Zi Chu, Steven Gianvecchio, Aaron Koehl, Haining Wang, and Sushil Jajodia. Blog or block: Detecting blog bots through behavioral biometrics. *Comput. Networks*, Vol. 57, pp. 634–646, 2013.