

# Myo.Nish:筋電位センサと深層学習を用いた小鼻動作追跡によるウェアラブルハンズフリー入力システムの提案と実装

中山 裕貴<sup>1</sup> 濱川 礼<sup>2</sup>

**概要:** 本論文では、筋電位センサと深層学習を用いた小鼻の動作解析手法の提案と、提案手法を利用したハンズフリーのための小鼻動作入力システム「Myo.Nish」の実装について述べる。電子機器への入力手法として、音声や視線の動きなど、体の動きに基づいたハンズフリー入力は長年研究されてきた。これらは、身体障害者の電子機器の操作、ウェアラブルデバイスの操作などに利用されてきた。小鼻の拡張・収縮動作は、所要時間が短く随意的なためハンズフリー入力へ十分利用することが可能である。本論文では、市販の筋電位センサを用いて小鼻の拡張・収縮動作中の鼻筋の筋電位を計測し、深層学習を用いて小鼻の動作解析を行う手法を提案する。また、解析結果を用いて実際にパソコンやスマートフォン、ウェアラブルデバイスといった電子機器を操作するシステム「Myo.Nish」を実装し、評価を行った。Myo.Nishでは、小鼻の拡張・収縮動作を認識し、クリックやショートカット呼び出し、キーボード入力などの機能を実行することでハンズフリー操作を実現している。

## 1. 背景

電子機器への入力手法としてハンズフリー入力は古くから研究が行われてきた。特に近年では、視線や音声を用いたものは数多く、誰でも利用可能なシステムとして受け入れられ始めている [1][2]。これらは、IoT やウェアラブルデバイスの発展、PC やスマートフォンなどの情報通信端末の普及率上昇により、生活様式が今までと大きく変化していることが原因と考えられる [3]。実際に、NTT データ経営研究所による調査では、「AI による音声認識・コミュニケーションについては、3 割程度の人々が既に利用したことがある、10 代では 6 割の人の利用経験がある。」という結果となっており、ハンズフリー操作が広く普及していることが分かる [4]。またこれらの発展に伴い、現在では音声や視線を用いた手法以外の様々なハンズフリー手法の研究が盛んに行われている [5]。

我々は、小鼻の随意動作に関するハンズフリーの有用性を検証してきた [6][7][8]。本論文において小鼻の随意動作とは、鼻孔を広げる拡張動作と鼻孔を萎める収縮動作である。小鼻の随意動作に着目した理由は、小鼻の随意動作を用いた研究は十分に行われていない反面、小鼻動作の随意性、独立性、即時性からハンズフリー利用において、小鼻の動作は有用性が高いと仮説を立てたからである。実際に、

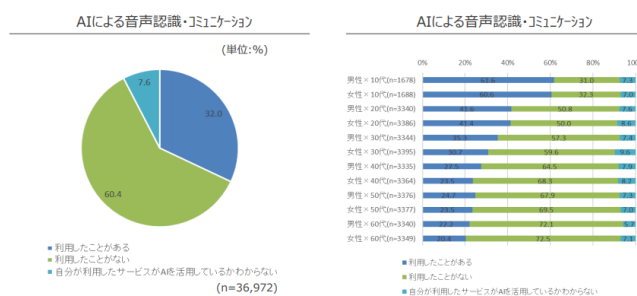


図 1 第 2 回消費者アンケート結果速報

本研究の事前研究にあたる圧力センサを用いた小鼻動作解析による入力システム「iNose」では、小鼻の動作入力、クリックやキーボード操作などのタスクをこなし、カーソル操作を行う視線追跡入力と十分な相乗効果を生み出し、小鼻の動作を用いたハンズフリーには、入力システムとして十分な価値があることが判明した [6]。また、身体障害者就労施設「わだちコンピュータハウス」にて上肢障害を持つ 10 名に実施した Nish に関する需要調査では、Nish を今すぐ使用したいかという項目において 50%の方が今すぐ使いたいと回答した [8]。圧力センサを利用することによる問題点を解決するため、我々は鼻筋運動に着目し、筋電位センサを利用することで、より堅牢性の高いシステム「Myo.Nish」を作成した [7]。しかし、動的タイムワーピング法と最近傍法を用いて小鼻動作解析を行う「Myo.Nish」では、システムの可用性という点に対して以下の問題点が

<sup>1</sup> 中京大学 工学研究科  
<sup>2</sup> 中京大学 工学部 情報工学科



図 2 鼻孔の拡張動作（左）と収縮動作（右）

であると分かった。

- 作業的要因：利用ごとに電極の配置位置が僅かに異なることが原因となり、取得する筋電位データが同種の動作においても大幅に異なってしまい、システムがうまく動作しないことがある。そのため、電極の配置が変更されるごとに教師データを取得しなおす必要がある。
- 時間的要因：上述の通り、システムを利用するためにはまず各動作種別の教師データを取得する必要がある、動作の収集に時間を要する。

上述の問題を解決するため、電極の配置箇所に関わらず正しく小鼻動作の認識を行う必要がある。これには、様々な設置個所での筋電位データを教師データに用いる必要がある。しかし、機械学習を用いた手法では、教師データの増加に伴い、解析時間が増加することが問題となる。そのため、我々は、学習データ数による処理速度低下が発生しないアプローチとして深層学習を用いた解析を行いこの問題の解決を試みた。また、深層学習を取り入れることにより、機械学習を用いていた時に、鼻筋のデータだけでは判別が不可能だった、口を窄めることに起因する鼻孔の収縮運動と、随意的な鼻孔の収縮運動を分類可能かどうか評価を行った。

## 2. 関連研究・関連システム

ハンズフリー入力に関する研究は今日まで様々なものが行われている。本章では、「ハンズフリー入力」「鼻を用いた入力手法」に関する研究を紹介する。

身体追跡、特に顔の部位に関する分野には様々な研究が存在する。Takuro Nakao らの研究では、顔の下半分の筋肉、特に舌や口、頬に関する筋電位データを解析しその動きに基づいてハンズフリー入力を実現している [9]。自作したデバイスを顔に装着し、取得した複数の筋電位データから特徴量とロジスティック回帰アルゴリズムなどを用いて各部位の動作認識を達成している。この自作したデバイスには2つの電極があり、口の上部と下部に接するように設計されている。

鼻の動作追跡において、眼鏡型ウェアラブルデバイス JINs meme を使用した鼻と指を用いたジェスチャー入力システムとして、Juyoung Lee らの研究がある [10]。Scikit-learn を用いた機械学習分類により、指で鼻をこするなど

のジェスチャーを認識している。この研究では、本研究と異なりハンズフリーシステムを目的としていない。解析する動作に、鼻の随意運動は含まれていない。また、本来眼の動きに関する眼電位データを取得するデバイスを使用しており、電極の設置位置・目的とする筋も異なる。

## 3. 提案システム

本章では、BLSTM を用いた鼻筋電位解析、それに伴う小鼻動作追跡の手法について述べる。本提案手法のデータフローを図3に示す。深層学習部では、BLSTM による鼻筋電位分類モデルの作成を行う。利用者は最初に利用する前に一度だけデータセットを作成し、BLSTM による分類モデルを作成する。これにより、次回以降の利用において、機械学習でのアプローチのように毎回使用時にデータ収集を行う必要がなくなり、システムの可用性を高めることができる。本手法では、鼻筋のみで口を窄める動作を判定している。そのため、BLSTM の動作種別分類結果は、鼻孔の拡張動作、鼻孔の収縮動作、口を窄めることで発生する鼻孔の収縮動作の3クラスとなる。

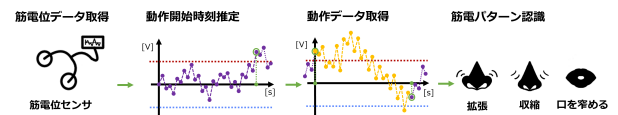


図 3 データフロー

### 3.1 筋電位センサの配置

本節では、筋電位センサの配置手法について述べる。筋電位センサを用いたシステムにおいて、センサの配置は非常に重要な問題である。正しく配置できていない場合、目的としている筋の動きを正確に計測することが不可能になってしまうからである。非侵襲型の筋電位センサでは、データを取得したい筋の中心部位の皮膚表面上に設置することが一般的とされている。また、周辺の筋活動を包括的に収集したい場合は、周辺の筋肉に跨るように設置するとよい。本研究では、解剖学用語の国際標準規格である Terminologia Anatomica を参考に目的とする筋を決定した。小鼻を拡張する作用のある筋は鼻筋に属する鼻孔拡張筋と上唇鼻翼挙筋であり、小鼻を収縮する筋は、鼻筋に属する鼻孔圧迫筋とされる [11]。図解解剖学事典によるこの筋の位置を図4(図内番号 16:鼻孔圧迫筋, 17:鼻孔開大筋, 96.7:上唇鼻翼挙筋)に示す [12]。顔の筋肉は細かく密集しているため、目的の筋のみをセンシングすることは困難であるため、鼻筋全体の活動を記録できるよう、鼻梁上部から鼻翼下部にかけて電極を配置した。この配置の場合、小鼻の拡張動作・収縮動作ともに筋電位データを取得することが可能となる。また、実際にセンサを配置した様子を図5に示す。

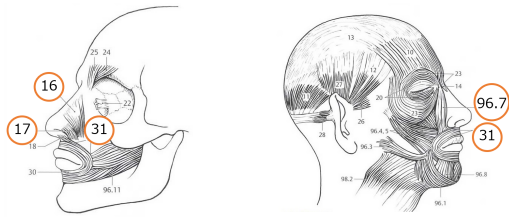


図 4 筋の位置

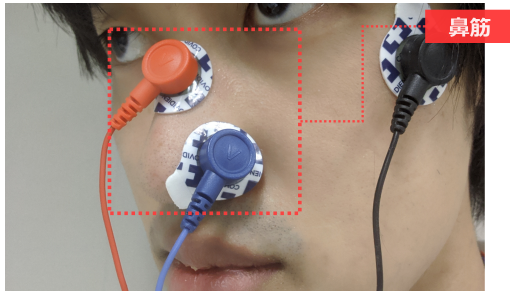


図 5 実際の電極配置

### 3.2 動作データの定義

本節では、筋電位識別を行う鼻筋運動に関する筋電位データについて述べる。本研究では、識別対象とする鼻筋の筋電位データを以下のように定義する。未動作時と比較し、何らかの値の変化が発生した時刻を動作開始推定時刻  $T$  とし、それ以降の動作に関連する筋電位データを十分に収集可能である期間  $I[s]$  とする。その期間  $I$  内にて観測された筋電位データを一連の系列データ  $E_{t(t=T, \dots, T+I)}[V]$  を小鼻の動作に関する 1 つの動作データ  $D$  として扱う。このため動作データ  $D$  は固定長の時系列データである。この時、筋電位データのサンプリング周波数を  $f[\text{Hz}]$  とした場合、動作データ  $D$  に含まれる筋電位データの総数は  $k = If$  となり、動作データ  $D$  は (1) 式で表せる。動作データの例を図 6 に示す。

$$D = \{E_T, E_{T+\frac{1}{f}}, \dots, E_{T+\frac{k-1}{f}}, E_{T+\frac{k}{f}}\} \quad (1)$$

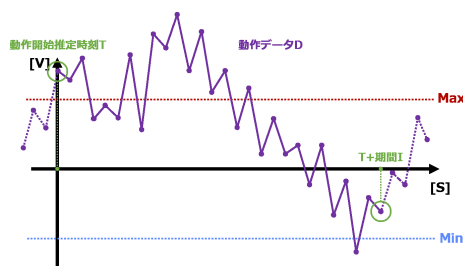


図 6 動作データの例

### 3.3 動作開始時刻推定

本節では、前節で述べた動作データの取得に用いる小鼻の動作開始時刻  $T$  を推定する手法について述べる。本研究では、筋電位データを波形として扱う。鼻筋運動が発生し

た場合、センサによって取得される波形の振幅は未動作時と比較して、大きくなる。そのため、2 つ閾値  $Max, Min$  を用いることで、小鼻に関する何らかの動作が行われた動作開始時刻  $T$  を推定することができる。推定のタイミングは、筋電位センサの値が閾値  $Max$  を上回った場合、または閾値  $Min$  を下回った場合である。上述のことから 2 つの閾値は、随意的な小鼻動作を行わない一定期間内に計測された筋電位の値の最大値を  $Max$ 、最小値を  $Min$  と設定することで、キャリブレーションを行う。一定期間内に取得した筋電位データの総数  $n$ 、取得した筋電位データを  $E_{i(i=1,2,\dots,n)}[V]$  とした場合、以下の (2), (3) 式によって表現できる。

$$Max = \max\{E_0, E_1, \dots, E_{n-1}, E_n\} \quad (2)$$

$$Min = \min\{E_0, E_1, \dots, E_{n-1}, E_n\} \quad (3)$$

順次取得するデータを  $E_t[V]$  とし、以下の (4) または (5) 式を満たすとき、その時刻を動作開始推定時刻  $T$  とする。図 7 に (4) 式を満たす場合の例を示す。

$$E_t > Max \quad (4)$$

$$E_t < Min \quad (5)$$

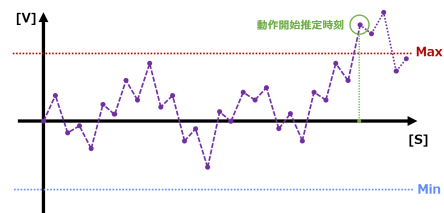


図 7 動作時刻推定の例 (Max の時)

### 3.4 データセット

本節では、BLSTM の学習に必要なデータセットについて述べる。BLSTM による鼻筋電位分類モデルの作成のためデータセットには、

- 鼻孔の拡張動作
- 鼻孔の収縮動作
- 口を窄める動作

を行ったときに取得できる鼻筋の動作データを用いる。電極の配置は、第 3.1 節の通り、鼻孔拡張筋と上唇鼻翼挙筋、鼻孔圧迫筋を網羅できるように鼻梁上部から鼻翼下部にかけて電極を配置する。データセット収集時、配置による筋電位の変化に対応するため、各動作につき 100 回分の動作データを収集後、再配置を行うようにした。データセット内の動作データの一例を図 8 に示す。

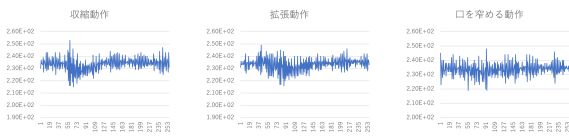


図 8 データセット一例

### 3.5 BLSTM による学習

BLSTM は時系列データに対応した LSTM モデルの一種である。通常の LSTM は、過去の入力から未来の予測を行うが、BLSTM は双方向のため、未来の入力から過去の出力の予測も同時に行う。適用例としては、文章の推敲や、機械翻訳、フレーム間の補完、音素・音声認識などが挙げられる。本研究で取り扱う鼻筋のデータは、固定長の時系列データのため、動作の開始時から終了時の筋電位データ・動作終了時から開始時の筋電位データどちらも用意することができる。そのため、筋電位の分類問題にも十分適用可能であると考えた。

## 4. システムの実装

本章では、前3章で提案した BLSTM を用いた Myo.Nish の実装について述べる。本研究では、操作対象端末を Windows としたが、python で実装を行っているため、Mac OS やその他の OS でも利用可能である。Myo.Nish のシステムフローを図9に示す。動作データ取得部では、市販品のマイコンボード、筋電位センサの Arduino UNO と MyoWare を用いて小鼻の動作データを取得する。これらは、入手の容易性や扱いやすさから選定した。鼻筋運動が発生した場合、Arduino UNO にて、取得した筋電位の値から小鼻の動作開始時刻を推定し、一連の動作データを取得する。その後、小鼻の動作データを Windows 側にシリアル通信によって送信する。動作解析部では、Arduino から受け取った動作データを生成した BLSTM モデルによって解析し、結果に応じたコマンド生成を実行する。

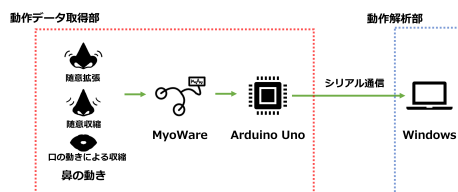


図 9 システムフロー

### 4.1 動作データ取得部

本節では、Arduino UNO と MyoWare を用いた動作データ取得について述べる。実装には Arduino IDE を用いた。

#### 4.1.1 A/D 変換と取得値

本項では、Arduino UNO での A/D 変換について述べる。MyoWare は、筋運動に伴う表皮の電位変化を計測するセンサである。そのため、計測される筋電位データは、

デジタルデータではなくアナログデータとなる。これらを Windows や Arduino で扱うためには、デジタルデータに変換する必要がある。Arduino UNO には、A/D コンバータが予め内蔵されているため、今回の実装では、その A/D コンバータを使用した。A/D コンバータの分解能は 10bit のため、MyoWare から Arduino に送信される取得値  $AD_{out}$  は 1024 段階の値 (0~1023) として扱われる。本実装において、Arduino Uno より供給される MyoWare の電源電圧は 5.0[V] のため、実際の筋電位  $E_{now}[V]$  は (6) 式により計算される。

$$E_{now} = \frac{5.0}{1023} AD_{out} \quad (6)$$

#### 4.1.2 取得サンプル数・サンプリングレート

Myo.Nish では、小鼻動作の開始から対象の端末に操作が行われるまでの時間を 300 ミリ秒以内とし、これを達成目標としている。この数値は、一般に筋電位の認識スピードとして十分に遅延がないとされる数値である [13]。そのため、サンプル数を大きく設定しすぎてしまった場合、データ収集自体に時間を要してしまい、解析にかかる時間が十分用意することができなくなってしまう。BLSTM の処理速度や、小鼻動作に関わる筋電位の変化を十分に取得できる時間を考慮し、本実装では、動作データに含まれる筋電位データの総数を 256 個のデータとし、サンプリングレートを 1000[Hz] と設定した。

#### 4.1.3 閾値の設定タイミング

本実装では、動作データの収集を行うための小鼻の動作開始時刻の推定に用いる閾値設定を Myo.Nish 起動時に行う。鼻筋は非常に細かい筋であるため、些細な配置の変動によって取得する値が変動してしまう。そのため、この値は、BLSTM を用いた手法でも、あらかじめ設定することは困難であり、設置ごとに再計測することが最良である。また、機械学習アプローチでの学習データ収集に比べ非常に短い時間でできるため、可用性を大きく損なうことはない。システムの可用性とのバランスを考慮し、5 秒間と設定した。

#### 4.1.4 データ送信

本項では、Arduino から Windows へのデータ送信について述べる。Myo.Nish では、Myoware の計測値が閾値  $Max$ ,  $Min$  の範囲を外れた場合、小鼻に関する筋活動が行われたと判定し、動作データを収集後、操作端末にその動作データを送信する。また、本実装では、動作開始推定時刻の 56 ミリ秒前から動作開始後の 200 ミリ秒後までのデータを一連の動作データとして扱う。これは、動作開始推定時刻にはすでに動作に関連する波形が表れており、その直前のデータを含めることがより自然な動作に関連するデータと考えたからである。データ送信はシリアル通信で行い、ボーレートは 11520[baud] に設定した。

## 4.2 動作解析部

本節では、Arduino から送信されてきたデータをもとに小鼻動作を識別し、Windows へのコマンド生成について述べる。実装は python で行い、BLSTM モデルの作成には keras ライブラリを使用した。

### 4.2.1 学習データ収集

本実装では、BLSTM モデルの作成のため、1つの動作種別毎に動作データをそれぞれ10セット収集した（各動作種別1000個計3000個）。個人差を考慮し、学習モデルの作成に利用するデータセットはその個人のデータのみとした。

### 4.2.2 BLSTM モデルの設定

本実装では、keras ライブラリを用いて BLSTM ネットワークを構築した。入力は次元の系列データとした。入力層、BLSTM 層、ドロップアウト層、全結合層、出力層の順で階層を構築した。BLSTM のタイムステップは動作データと同じく256とした。また、BLSTM 内の内部ユニット数は100とした。ドロップアウト層の不活性率は50%とした。全結合層のユニット数は100とし活性化関数には relu 関数を用いた。出力層では、3クラス分類のため、ユニット数を3とし、活性化関数には softmax 関数を使用した。損失関数はクロスエントロピー法を利用した。

### 4.2.3 コマンド生成

本項では、Windows 側のコマンド生成について述べる。受け取った動作データを BLSTM により分類後、分類した動作クラスに応じた入力を生成する。入力の生成には python ライブラリの pyautogui を使用した。動作クラスと生成する入力の対応表を表1に示す。

動作クラス	入力
小鼻の拡張動作	左クリック
小鼻の収縮動作	右クリック

## 5. 評価・考察

本章では、BLSTM を用いたアプローチでの Myo.Nish の評価について述べる。システムの適用デバイスを Windows OS 搭載の PC とし、2種の小鼻動作の入力精度に関する評価を行った。

### 5.1 評価項目

本節では、Myo.Nish による小鼻動作入力の入力精度に関する評価として、以下の項目についてそれぞれ評価実験を行った。

#### (1) 電極配置の堅牢性

BLSTM を用いることで電極の配置に関する堅牢性が向上したか評価を行う。

#### (2) 口を窄める動作の判定率

BLSTM によって鼻筋の動作データのみで口を窄める動作による鼻孔の収縮動作を除去できると仮定している。そのため、鼻筋データのみで口を窄める動作に起因する鼻孔の収縮動作を正しく認識できているか評価を行う。

#### (3) 各主動作の認識率

BLSTM を用いることで、入力精度がどう変化するか評価を行う。

#### (4) 認識スピード

BLSTM を用いた場合に、小鼻動作の認識速度が300ミリ秒以内に抑えられるかどうか評価を行う。

## 5.2 評価方法

本節では、前節で述べた評価項目に関する実際の評価方法について述べる。

#### (1) 電極配置の堅牢性

2種類の小鼻動作よりランダムに1種類の動作を被験者に提示する。被験者は提示された動作を行う。これを計10回繰り返したのち電極の再配置を行う。再配置後同様にテストを行い、再配置前との認識率を計測する。再配置は計5回行う。再配置箇所は目的の筋を網羅できる範囲内で再配置前と異なる位置に設置する。

#### (2) 口を窄める動作の判定率

被験者は口を窄める動作を100回行い、BLSTM による判定結果を計測し、正しく例外処理されているか小鼻の随意収縮として認識されているかを評価する。

#### (3) 各主動作の認識率

2種類の小鼻動作よりランダムに1種類の動作を被験者に提示する。被験者は提示された動作を行い、認識された結果が提示された動作と一致したか集計する。これを100回行い、動作の認識率を計測する。提示後、30秒間入力が無い場合、未入力として処理する。

#### (4) 動作の認識スピード

被験者は、それぞれ100回分2種類の小鼻の動作を行い、動作データを受け取り始めてから、出力を生成するまでの時間を計測する。

## 5.3 評価結果

各評価項目について、その結果を以下に示す。

#### (1) 電極配置の堅牢性

評価結果を表2に示す（各戸内は電極の配置別）。合計50回の試行のうち、小鼻の拡張の認識率は70.0%、収縮は66.0%であった。

#### (2) 口を窄める動作の判定率

評価結果を表3に示す。合計100回の試行の内、小鼻の収縮の認識率は57.0%、口を窄める動作は68.0%であった。

表 2 堅牢性の結果

提示動作	拡張	収縮	口を窄める	未入力
小鼻の拡張動作 (1)	6	3	1	0
小鼻の収縮動作 (1)	1	7	2	0
小鼻の拡張動作 (2)	8	2	0	0
小鼻の収縮動作 (2)	0	7	3	0
小鼻の拡張動作 (3)	7	3	0	0
小鼻の収縮動作 (3)	0	6	4	0
小鼻の拡張動作 (4)	8	2	0	0
小鼻の収縮動作 (4)	0	6	4	0
小鼻の拡張動作 (5)	6	2	2	0
小鼻の収縮動作 (5)	1	7	2	0

表 3 認識率の結果

提示動作	収縮	口を窄める	未入力
小鼻の収縮動作	57	43	0
口を窄める動作	32	68	0

### (3) ノイズ・誤認識率

評価結果を表 4 に示す。合計 100 回の試行の内、小鼻の拡張の認識率は 67.0%、収縮は 71.0%であった。

表 4 認識率の結果

提示動作	拡張	収縮	口を窄める	未入力
小鼻の拡張動作	67	20	13	0
小鼻の収縮動作	8	71	21	0

### (4) 認識スピード

評価結果を表 5 に示す。それぞれ合計 100 回ずつの試行での平均スピードは、拡張動作が 295 ミリ秒、収縮動作が 297 ミリ秒であった。

表 5 認識スピードの結果

種別	平均 [ms]	最速 [ms]	最遅 [ms]
小鼻の拡張動作	295.5	276.3	539.5
小鼻の収縮動作	297.5	275.0	453.0

## 5.4 考察

評価実験の結果から、BLSTM を利用したアプローチでは十分な認識結果を得ることは難しいということが判明した。評価項目のうち十分に達成できたといえる項目は、認識スピードのみとなった。認識スピードの最遅値に関して、収縮、拡張ともに学習モデルを読み込んだのち最初の予測で最遅値が確認された。追加で、100 回モデルロード直後の速度を計測した結果、類似した値が確認できたため、何らかの原因で初めの予測に時間が要することが判明した。堅牢性に関しては、評価のスコアは低下するが、機械学習では実現不可能だった、装着時のデータ収集を省くことが実現できることが分かった。モデルの精度を高めることができれば、より可用性が向上すると考えられる。口を窄める動作に関しては、随意収縮時の誤認識が多く、入

力が省かれてしまうという問題が残った。口を窄める動作の認識に関しても 68.0%と、実現できているが十分な精度とは言えない結果となった。総じて、十分な精度の学習モデルを作成することができなかったと言える。精度を得られなかった原因は、計測する筋が一種となっており、分類に用いる特徴量が少ないこと、収集した筋電位データをそのまま学習に用いているということ、同一設置箇所における学習データ数の不足などが考えられる。

## 6. おわりに

今回の実装では、Myo.Nish に対し深層学習による動作解析を適用した。その結果、機械学習による解析や圧力センサ利用時の認識率と比較して、大幅に認識率を損なうこととなってしまった。しかし、深層学習の利用は筋電位センサの電極配置による筋電位データの影響への対応策としてある程度の効力を発揮したと考えることもできる。学習モデルのもともとの認識精度を向上させることができれば、機械学習アプローチや圧力センサに比べて確実により価値の高いシステムになると考え、今後システムの改良・評価を進めていく。

## 参考文献

- [1] Meenatchi D., Aishwarya R., Shahina A.: A Voice Recognizing Elevator System; ICSCS , vol.397, pp.179-187,(2016).
- [2] Sengupta, K., Ke, M., Menges, R., et al.: Hands-Free Web Browsing: Enriching the User Experience with Gaze and Voice Modality;ETRA '18, No.88,(2018).
- [3] 総務省：情報通信白書, 平成 30 年版
- [4] NTT データ経営研究所, 『第二回消費者アンケート結果速報』[https://www.caa.go.jp/policies/policy/consumer\\_policy/meeting\\_materials/assets/consumer\\_policy\\_cms101\\_200525\\_06.pdf](https://www.caa.go.jp/policies/policy/consumer_policy/meeting_materials/assets/consumer_policy_cms101_200525_06.pdf)
- [5] Ofner, P., Schwarz, A., Pereira, J. et al.: Attempted Arm and Hand Movements can be Decoded from Low-Frequency EEG from Persons with Spinal Cord Injury;Sci Rep 9, Vol.7134, (2019).
- [6] 中山, 久野, 谷口, 濱川: iNose:圧力センサによる小鼻動作追跡と視線追跡を併用するウェアラブルハンズフリー入力システムの提案と試作;HCI, Vol.187, No.30, (2020) .
- [7] 中山, 濱川:Myo.Nish : 筋電位センサを用いたハンズフリーのための小鼻動作入力システムの提案と試作; ヒューマンインタフェース学会研究報告集 22, p79-84, (2020).
- [8] 中山, 濱川:NISH:加速度センサによる顔追跡を導入した福祉機器としての小鼻動作入力インタフェースの検討; 第 2 回神経刺激インタフェース研究会,(2020).
- [9] Nakao, T., Pai, Y., Isogai, M., et al.:Make-a-face: a hands-free, non-intrusive device for tongue/mouth/cheek input using EMG; SIGGRAPH'18, No.24,(2018).
- [10] Lee, J., Yeo, J.,Dhuliawala, M., et al.:Itchy Nose: Discreet Gesture Interaction using EOG Sensors in Smart Eyewear;ISWC'17,pp.94-97,(2017) .
- [11] FICAT: Terminologia Anatomica; Thieme,(1998).
- [12] 山田: 図解解剖学事典 第 3 版; 医学書院,(2018).
- [13] Mizuno, H., Tsujiuchi, N., and Koizumi, T.:Forearm motion discrimination technique using real-time emg signals;IEEE Eng Med Biol Soc., pp.4435-4438, (2011).