

非接触型センサを用いた歩行周期検出システムの開発

大 関 文 弥*・木 澤 悟

Detection system at walking cycle using non-contact sensor

Fumiya OZEKI* and Satoru KIZAWA

(平成21年11月27日受理)

This paper proposes detection system at walking cycle using non-contact sensor. Hemiplegic individuals with Drop foot can't raise toe or fall hemiplegic gait when they walk. Recently, Functional Electrical Stimulation (FES) system is developed to assist walking of hemiplegic patients by control of muscle using electrical stimulation. FES system for Drop foot need signal of swing phase at walking. Usually, Footswitch is used as detection system to judge whether swing phase or not. However this switch have issue that a feeling of wrongness and not much endurance. So, we use accelerometer and gyroscope wore the knee instead of Footswitch. And we use Neural Network as the way to presume swing phase.

1. 緒言

脳卒中の後遺症として片麻痺が残る場合、歩行時に足のつま先を上げることができない下垂足を生じることがある。この下垂足に対する歩行再建法として、機能的電気刺激（FES）を用いた筋制御によってつま先を引き上げ、歩行障害を改善する方法がある。従来、FESにおける遊脚期の識別は、足底に装着したフットスイッチからのON/OFF情報によって行われていた。しかしながら、この方法では足裏に違和感やフットスイッチの耐久性での問題がある。そこで、本研究では患者にとっての違和感や不快感を和らげるとともに、接触型であるフットスイッチの耐久性の問題を解決するため、非接触型センサを用いた遊脚期検出システムについて検討した⁽¹⁾。フットスイッチの代わりに遊脚期の情報を得るための非接触型センサとして3軸加速度センサおよび1軸ジャイロセンサを用い、それらのセンサ情報とパターン解析が得意なニューラルネットワーク(N.N.)を組み合わせることで、歩行中の遊脚期を推定した。

* 秋田高専専攻科学生

2. 実験装置

2.1 非接触型センサ

遊脚期を推定するための非接触型センサは、3軸加速度センサ（Hitachi Metals製 H48C）およびジャイロセンサ（村田製作所製 ジャイロスター）を利用した。フットスイッチおよび非接触型センサをFig.1に示す。



フットスイッチ
(接触型センサ)



3軸加速度+ジャイロセンサ
(非接触型センサ)

Fig. 1 非接触型センサ

2.2 実験装置および手順の概要

Fig.2に本研究で提案する遊脚期検出システムを示す。歩行データの計測・記録は、データ収集のサンプリング周期を10msecと設定したHIYOKI製のデータロガーを利用して行った。このとき、センサ出力から電氣的なノイズや振動によるノイズを除去するためにカットオフ周波数が10Hzであるローパスフィルタを用いた。遊脚期を推定するために行うN.N.の計算はPC上で学習後、算出されたN.N.のパラメータを秋月製H8マイコン（AKI-H8-3052F）

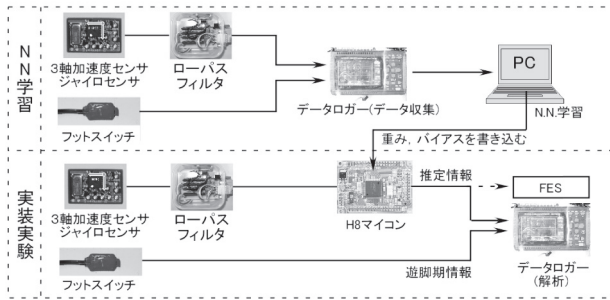


Fig. 2 遊脚期検出システムの概要

に書き込み実行した。なお、H8マイコンへのセンサ情報の入力にはサンプリング周期20msecに設定した。遊脚期検出システムを用いた実験は、次に示す(1)～(3)の手順で行った。

手順 (1)

N.N.学習を行うに当たり、歩行時の3軸加速度センサ・ジャイロセンサ、そしてフットスイッチの出力をデータロガーにより記録する。3軸加速度センサおよびジャイロセンサの装着位置は膝下約50mmとし、フットスイッチは同じ足の踵部とした。Fig. 2 に示した、N.N.学習に用いる歩行データの計測システムの装着概要をFig. 3 に示す。

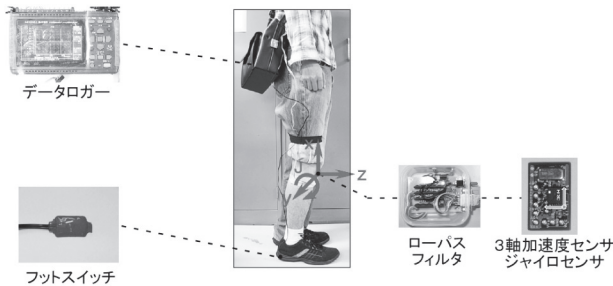


Fig. 3 装着した計測システム

手順 (2)

計測した各情報をもとにPC上でMATLABを用いてN.N.の学習を行い、ネットワークの重み・バイアス値を求め、N.N.のパラメータを含めた遊脚期推定プログラムをH8マイコンに書き込む。

手順 (3)

実装実験においては60sec間の直線歩行を行い、歩行時の加速度・ジャイロセンサ出力をH8マイコンで読み込んだ後、N.N.による遊脚期推定値をデータロガーにより記録する。同時に、確認用の正確な遊脚期情報を得るために、フットスイッチを装着しON・OFF情報をデータロガーにより記録す

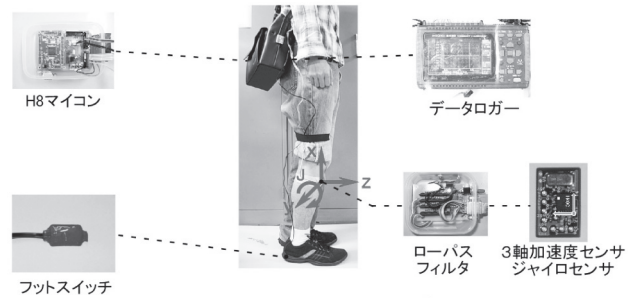


Fig. 4 装着した実装実験におけるシステム

る。Fig. 2 に示した実装実験の装着概要をFig. 4 に示す。

3. 実験方法

3.1 歩行データ測定

本研究の最終目的は下垂足患者に対する遊脚期の推定であるが、研究の前段階として健常者に装着して直線歩行させ、本研究で開発したシステムの有効性を確認し検証した。

はじめに前節の手順(1)のN.N.学習用データを構築するために、80sec間の直線歩行を行い、教師信号用としてのフットスイッチ出力、入力信号として3軸加速度センサそしてジャイロセンサの出力をデータロガーで計測・記録した。歩行時の速度は、より様々な状況を想定してN.N.に学習させるために、感覚的に速め・普通・遅めと変化させた。

3.2 ニューラルネットワークの学習

前節の手順(2)では、手順(1)で得たセンサ情報をもとに、ニューラルネットワークを用いた学習を行う。構築したN.N.は階層型で、入力層、中間層、出力層からなる3層構造である。入力センサは3軸加速度センサ(x軸、y軸、z軸)およびy軸周りジャイロセンサであるが、入力センサの組み合わせによって遊脚期推定に違いがあるかどうか確かめるために、2通りの場合で実験を行った。y軸方向の加速度センサを利用するか否かの違いは、健常者ならば外側に足をぶん回す様な歩行は、一般的にあり得ないのでy軸方向は無視できるのではないかと考えた。以下に2つのN.N.構造を示す。

- N.N.構造(I) 加速度センサ(x軸、z軸)、ジャイロセンサを利用した場合
- N.N.構造(II) 加速度センサ(x軸、y軸、z軸)、ジャイロセンサを利用した場合

N.N.構造(I)のN.N.学習は、各層のニューロン数を入力9、中間8、出力1とした。特に入力層

には、上下方向の加速度（x軸加速度）、前後方向の加速度（z軸加速度）およびy軸まわりのジャイロ情報で3時点分（現時点、1時点前、2時点前）を考慮して、計9入力を用い、時系列データはソフト的に20msec間隔でサンプリングした。出力層は遊脚期の推定用として1出力である。なお、中間層および出力層のニューロン関数は非線形のシグモイド関数を利用した。Fig. 5-1にN.N.構造（I）を示す。

N.N.構造（II）のN.N.学習は構造（I）と同じ3層構造であり、各層のニューロン数は入力12、中間8、出力1とした。入力層は、左右方向の加速度（y軸加速度）を含めた加速度・ジャイロ出力をそれぞれ3時点分用いた計12入力であり、時系列データはソフト的に20ms間隔でデータをサンプリングした。Fig. 5-2にN.N.構造（II）を示す。

ここで本研究におけるN.N.学習とは、センサ情

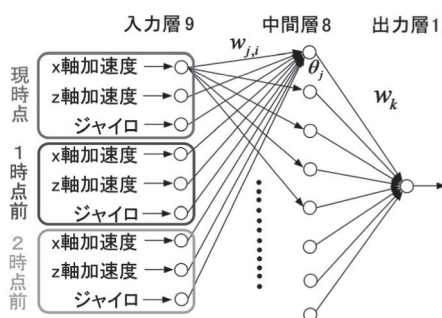


Fig. 5-1 N.N. 構造 (I)

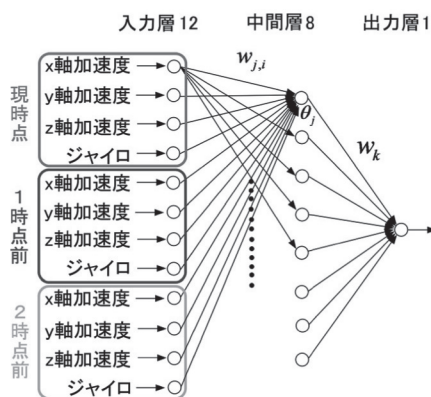


Fig. 5-2 N.N. 構造 (II)

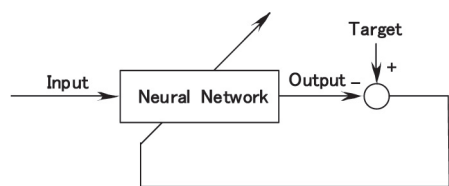


Fig. 6 学習の原理

報から遊脚期情報を推定するためにFig. 5-1およびFig. 5-2に示すネットワーク間の重み w とバイアス値 θ の適切な値を求めることである。学習原理の概念をFig. 6に示す。入力情報からN.N.の出力を求めて教師信号との比較を行い、その誤差に応じてN.N.の重みとバイアス値を調整することでネットワークの出力を教師信号へ近づけさせる。学習はMATLABを用いてオフラインで行い、学習アルゴリズムとしてバックプロパゲーションを利用し、教師信号としてフットスイッチのON・OFF情報を用いた。

N.N.を用いた学習を行わせるにあたり、学習データの長さがN.N.の出力結果に影響を及ぼすと考えられる。そこで、計測した80sec間の歩行データの長さを様々に変えて学習することを試みた。学習に用いるデータは、Fig. 7の学習用歩行データに示すように、10secから80secまで10sec毎にデータの長さを変化させ、Case1 (10sec間)~Case8 (80sec間)の8通りの学習データ長さを設定した。

学習終了後は実装実験を行うために、学習によって得られたN.N.の重みパラメータをH8マイコンに書き込み、実装実験によるN.N.の出力が遊脚期推定結果に影響があるかどうかを検討した。なお、H8マイコンに入力されるセンサ情報のデータサンプリングは20msecとした。これは、PC上のN.N.学習において時系列データをソフト的に20msec間隔でサンプリングしたことに対応するためである。

3.3 実装実験

前節の手順(3)では、手順(2)で学習させたN.N.を用いて、実際に歩行時の各センサ情報から遊脚期を正確に推定できるかどうかを確認するために、前節のFig. 4に示したシステムにより実装実験を行った。実装実験での歩行は60sec間、感覚的に一定速度の直線歩行とした。

実装実験では、歩行時の加速度センサおよびジャイロセンサの出力をH8マイコンで読み込み、マイコン上にプログラムされたN.N.により、遊脚期の推定を行う。H8マイコン上のN.N.による遊脚期推定結果は、足が上がった遊脚期情報として「0」、足が地面に着いた支持脚期情報として「1」が出力されるが、さらにあいまいな遊脚期の判定情報として0より大きく1より小さい値が出力される場合がある。また、本システムにおいて、ローパスフィルタで完全に除去することができなかったセンサ情報に含まれるノイズの影響で、遊脚期推定値が不安定になる場合があり、N.N.出力に対して移動平均法を

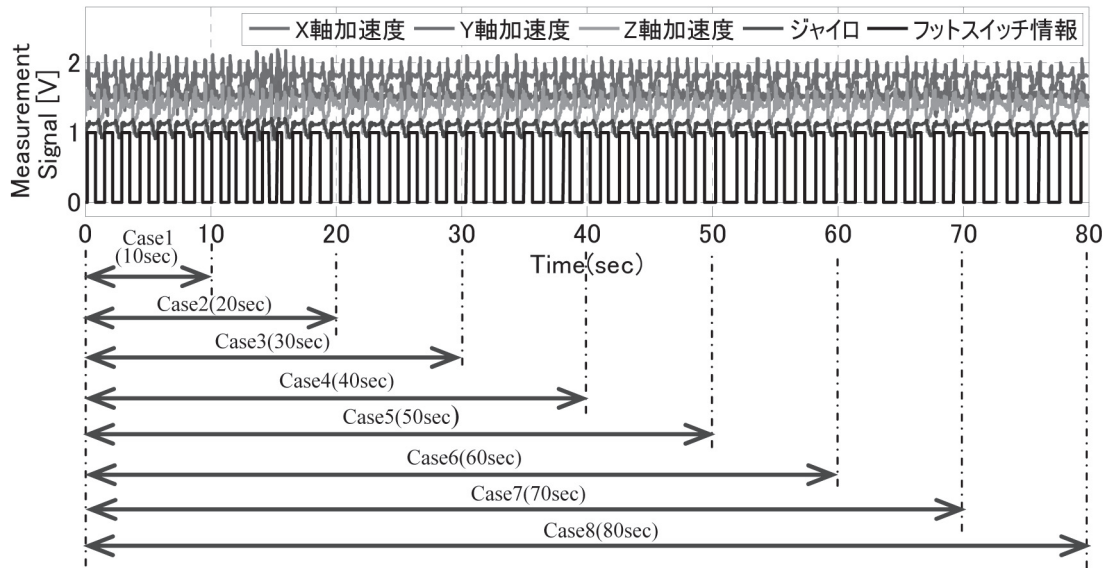


Fig. 7 学習用歩行データ長さ

用いることにした。移動平均法とは、現在の情報および過去の情報を平均することにより、有限のインパルス応答をフィルタリングする手法である。本実験では、現時点および過去4時点分のN.N.出力を平均し、Fig. 8に示すように遊脚期推定値を平滑化した。破線は移動平均前、実線は移動平均後である。しかしながら、移動平均化した遊脚期情報が0より大きく1より小さい場合は、遊脚期と支持脚期を判断できない。そのため、Fig. 9に示すように、閾値を設定し、平均した値が0.85以上の場合は支持脚期判定「1」とし、0.25以下の場合には遊脚期判定「0」とし、それ以外は1時点前と同様の判定をさせることにより推定結果を2値化させる。N.N.による遊脚期の推定結果はデータロガーに記録した。また、こ

のとき、正確な遊脚期を知るために装着したフットスイッチの出力を同時にデータロガーにより記録した。

4. 評価方法

N.N.によって推定した遊脚期出力は、推定エラーあるいは、フットスイッチの出力に対してタイミングのずれが生じる。そこで、Fig. 10に示すとおり、フットスイッチの出力と実装実験でのN.N.出力を重ね合わせ、Case1~Case8の学習用データ長さの違いによってどの程度エラー並びに遊脚期推定のタイミングの遅れが生じるかを比較検討することにした。推定エラーは、Fig. 10に示す微少な時間の誤判断とし、エラーの時間の長さではなく、エラー回数で考える。遊脚期推定値のタイミングの遅れの評価

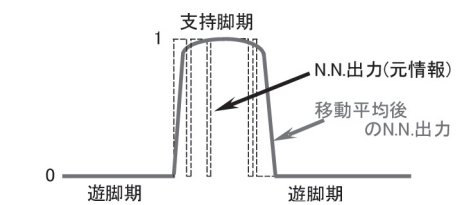


Fig. 8 移動平均法

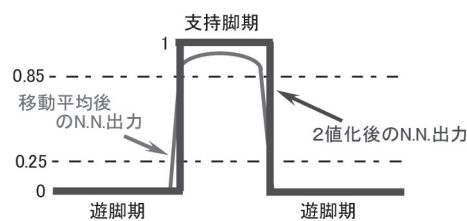


Fig. 9 出力情報の2値化

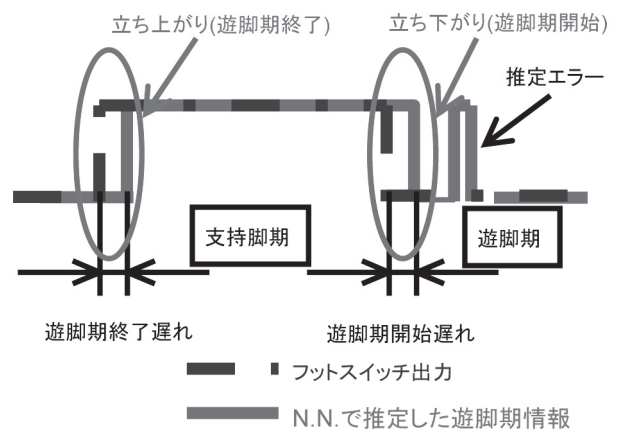


Fig. 10 評価方法

は、Fig. 10に示すように立ち上がり（遊脚期終了）と立ち下がり（遊脚期開始）に分け、フットスイッチ出力に対するタイミングの遅れ時間をヒストグラム化することにより行った。

さらに、推定エラー回数とタイミングの遅れの比較は、N.N.の構造がN.N.構造（I）とN.N.構造（II）に関してそれぞれ行い、また、（I）と（II）とでどちらが優れた遊脚期検出のためのN.N.構造であるかを検討した。つまり、加速度センサの方向成分の入力数（y方向成分の有無）の違いによって、遊脚期推定結果にどのような影響を与えるか検証した。

5. 実験結果

5.1 学習用計測歩行データ

学習用のデータを構築するために、一例として学習用データ長さCase1に対する歩行データ（x軸加速度、y軸加速度、z軸加速度、ジャイロ、フットスイッチ）をFig. 11に示す。図は学習用に歩行時間80secから10sec長のデータを抽出したものである。フットスイッチの情報に関しては、スイッチがOFFのとき遊脚期情報として0 [V]、スイッチがONのとき支持脚期情報として1 [V]で表している。図に示していないが、同様にCase2～Case8の場合は、歩行時間80secの歩行データから20sec～80secのデータを抽出したものである。

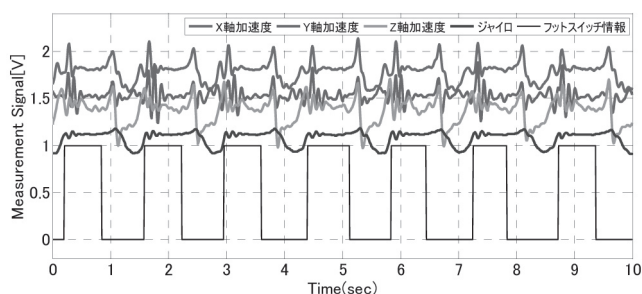


Fig. 11 歩行データ (Case1 の場合)

5.2 学習結果

Fig. 11の歩行データを用いてCase1 に対するN.N.学習をPC上でMATLABにより行った。また、学習で求められたネットワーク間の重みとバイアス値を利用して、設計したN.N.による遊脚期推定値が教師信号であるフットスイッチの遊脚期に近似しているか確認するためにPC上でシミュレーションする。なお、シミュレーションの入力には、Fig. 11に示したセンサ情報を用いた。N.N.の構造の違いであるN.N.構造（I）とN.N.構造（II）に対するシミュレーション結果をそれぞれFig. 12-1, Fig. 12-2に示す。

レシジョン結果をそれぞれFig. 12-1, Fig. 12-2に示す。

Fig. 12-1 と Fig. 12-2のいずれもフットスイッチ出力と遊脚期推定値がほぼ完全に一致するほど学習結果は良好であり、センサ入力数の違いの影響は無いと言える。シミュレーション結果ではN.N.構造（I）とN.N.構造（II）には大きな違いがなかったため、どちらのN.N.構造が遊脚期の推定に有効であるかどうか実装実験によって検証する。図には示さないがCase2～Case8でもそれぞれ学習を行ったところ、同様なシミュレーション結果が得られた。

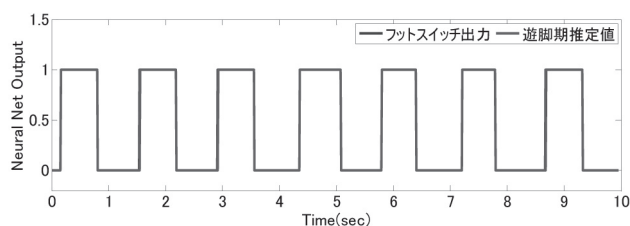


Fig. 12-1 N.N.構造（I）のシミュレーション結果

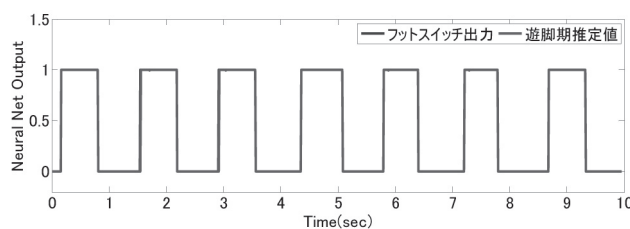


Fig. 12-2 N.N.構造（II）のシミュレーション結果

5.3 実装実験の結果

前節で述べたように、シミュレーション上では、良好な遊脚期の推定結果が得られたので次にN.N.構造の違いによる実装実験を行った。

□N.N.構造（I）の場合

学習用データが最も短いCase1 の場合の実装実験の出力結果をFig. 13に示す。図で、破線（FOOT）はフットスイッチ出力、実線（NNET）はニューラルネットによる遊脚期推定値を表している。Case1では、N.N.による遊脚期推定値は非常に不安定であり推定エラーが多数発生している。そのため、推定した遊脚期のタイミング的な遅れ時間に関するヒストグラムの作成は不可能であった。

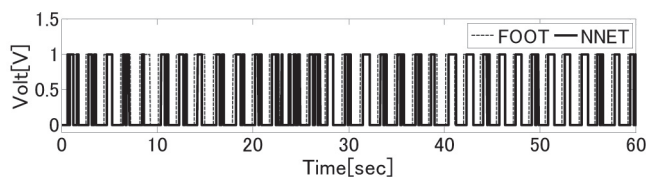


Fig. 13 N.N.およびフットスイッチ出力の比較

次に学習用データ長さが40secであるCase4の実装実験の結果をFig. 14-1~Fig. 14-3に示す。Fig. 14-1は実装実験における遊脚期の推定結果とフットスイッチの出力結果を比較した図であり、Fig. 14-2とFig. 14-3は、それぞれ遊脚期推定結果のFig. 14-1における立ち下がり（遊脚期開始）と立ち上がり（遊脚期終了）でのフットスイッチとのタイミングの遅れを比較したヒストグラムである。図は横軸に遅れ時間、縦軸に遅れ時間に対する頻度をとっている。タイミングの遅れに関しては、フットスイッチ出力に対する遊脚期推定値の遅れ時間を正值としている。学習用データとしてCase4を用いた場合の実装実験結果は、Case1の場合と比べて1度も推定エラーが生じていないため、遊脚期の推定が適切に行われていると言える。また、Case4のヒストグラムを見ると、遊脚期推定結果はフットスイッチの出力よりも常にタイミング的に時間が遅れていることがわかる。

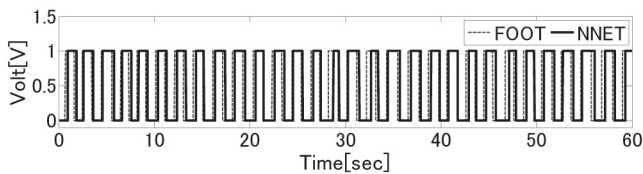


Fig. 14-1 N.N.およびフットスイッチ出力の比較

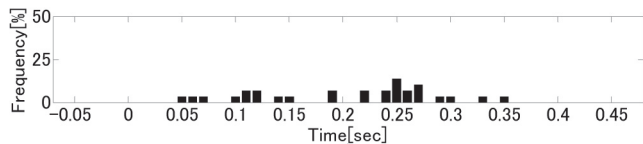


Fig. 14-2 遊脚期開始時の遅れに関するヒストグラム

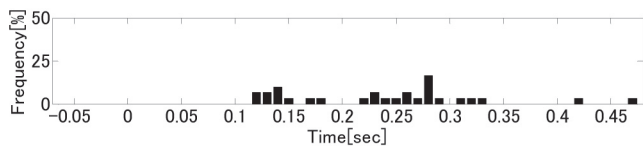


Fig. 14-3 遊脚期終了時の遅れに関するヒストグラム

次に学習データが最も長いCase8での実装実験結果をFig. 15-1~Fig. 15-3に示す。Fig. 15-1を見るとCase4と同様、推定エラーの発生はなく遊脚期を適切に推定していると言える。N.N.構造（I）の場合、遊脚期終了時のタイミング遅れに関するヒストグラムであるCase4のFig. 14-3とCase8のFig. 15-3を比較すると、学習データの長いCase8の方が遅れ時間が短いこと、また、遅れ時間のばらつきが小さく安定した遊脚期を推定していることがわかる。

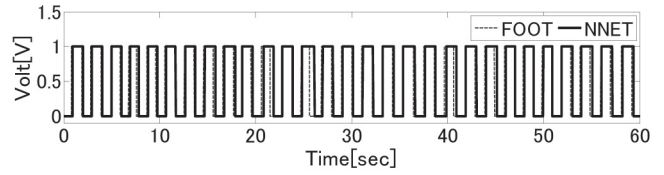


Fig. 15-1 N.N.およびフットスイッチ出力の比較

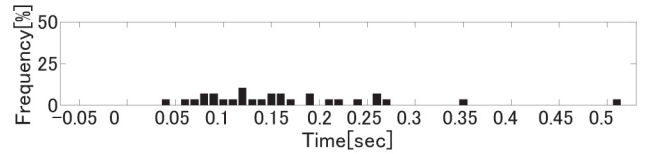


Fig. 15-2 遊脚期開始時の遅れに関するヒストグラム

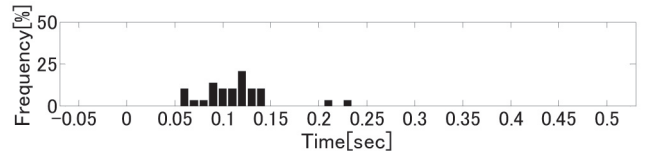


Fig. 15-3 遊脚期終了時の遅れに関するヒストグラム

□ N.N.構造（II）の場合

学習用データの最も短いCase1の実装実験結果をFig. 16-1~Fig. 16-3に示す。Fig. 16-1より、遊脚期推定値にはエラーが生じているが、エラー回数はN.N.構造（I）におけるCase1の実装実験結果Fig. 13と比べると少なく、ヒストグラムは作成可能で、このヒストグラムFig. 16-1およびFig. 16-2を見ると、N.N.構造（II）の場合でもN.N.構造（I）と同様、遊脚期の推定値がフットスイッチ出力に比べて遅れていることがわかる。

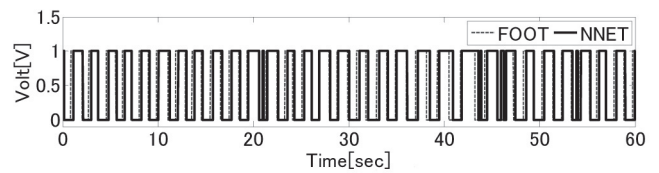


Fig. 16-1 N.N.およびフットスイッチ出力の比較

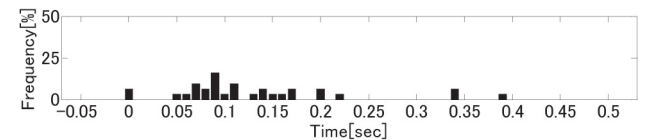


Fig. 16-2 遊脚期開始時の遅れに関するヒストグラム

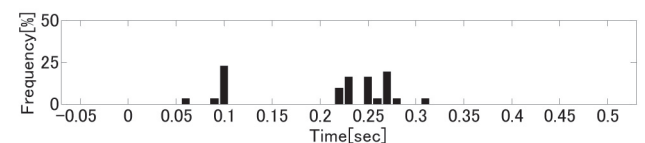


Fig. 16-3 遊脚期終了時の遅れに関するヒストグラム

Case1よりも学習データが長いCase4での実装実験結果をFig. 17-1～Fig. 17-3に示す。Fig. 17-1より、Case4では推定エラーは一度もないが、他のCaseと同様、Fig. 17-2・Fig. 17-3から、遊脚期推定値はフットスイッチ出力より遅れていることがわかる。

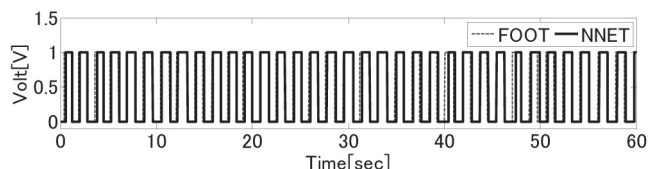


Fig. 17-1 N.N.およびフットスイッチ出力の比較

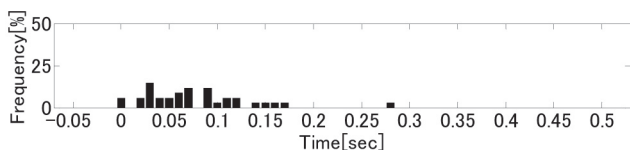


Fig. 17-2 遊脚期開始時の遅れに関するヒストグラム

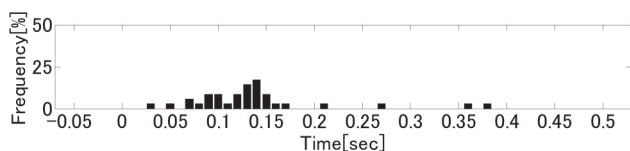


Fig. 17-3 遊脚期終了時の遅れに関するヒストグラム

次に学習データが最も長いCase8での実装実験結果をFig.18-1～Fig.18-3に示す。Fig.18-1より、Case4と同様にCase8では推定エラーは一度もない。他のCaseでの実験と同様に、遊脚期推定値はフットスイッチ出力よりも遅れている。つまり、N.N.構

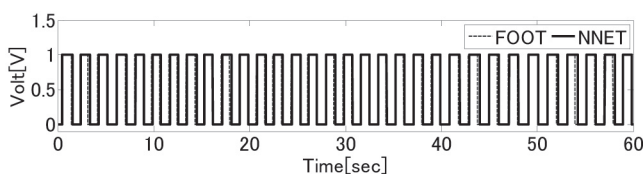


Fig. 18-1 N.N.およびフットスイッチ出力の比較

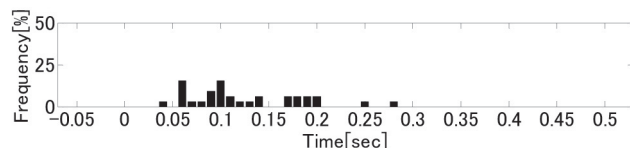


Fig. 18-2 遊脚期開始時の遅れに関するヒストグラム

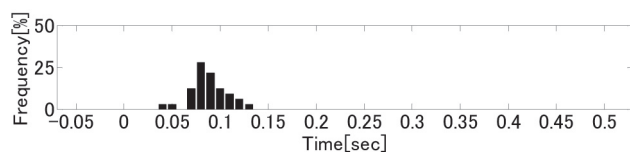


Fig. 18-3 遊脚期終了時の遅れに関するヒストグラム

造(Ⅱ)の場合においてはCase1, Case4そしてCase8の実験結果と比較すると、学習時間が長くなるにつれて、遊脚期推定値の時間遅れの遅れ頻度のばらつきが小さくなる傾向にあり、ある程度一定な遅れで遊脚期を推定していることがわかる。

以上、上記では学習用データ長さCase1, Case4そしてCase8より検討しなかったが、N.N.構造(Ⅰ)とN.N.構造(Ⅱ)の場合の実装実験結果をまとめる。学習データ長さが遊脚期推定結果に与える影響をグラフ化するために、N.N.構造(Ⅰ)の場合に対しての学習データ長さ(Case1～Case8)と遊脚期遅れ時間との関係をFig.19-1に示す。Fig.19-1は、「平均遊脚期終了遅れ時間」、「平均遊脚期開始遅れ時間」、そして「平均遊脚期遅れ時間」について学習時間長さ毎に遅れ時間をプロットしたものである。ここでいう「平均遊脚期終了遅れ時間」とは実装実験60sec間の歩行において、前節で説明した「遊脚期終了遅れ時間」を足し合わせ、遅れの回数で除したものである。同様に「平均遊脚期開始遅れ時間」とは「遊脚期開始遅れ時間」を足し合わせ、遅れ

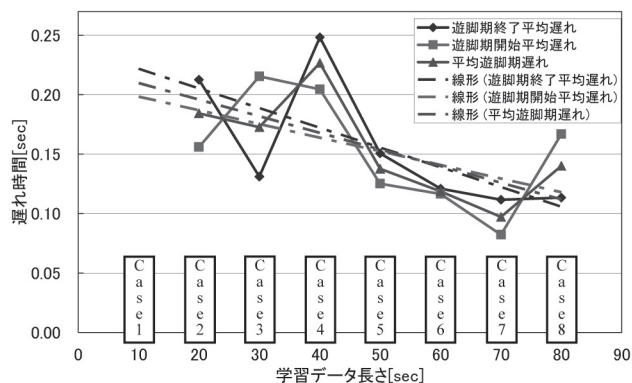


Fig. 19-1 学習データ長さと推定情報の遅れ時間との関係・N.N.構造(Ⅰ)の場合

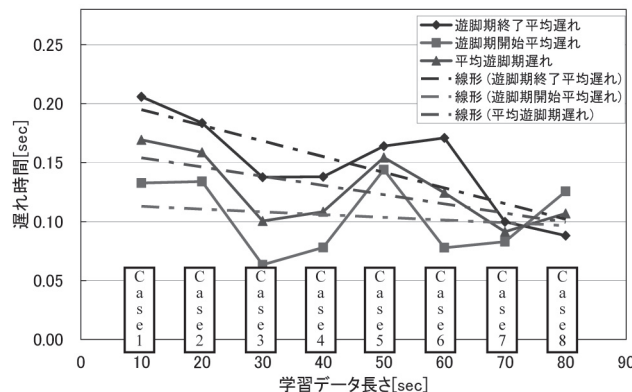


Fig. 19-2 学習データ長さと推定情報の遅れ時間との関係・N.N.構造(Ⅱ)の場合

の回数で除したものである。また、「平均遊脚期遅れ」は遊脚期終了遅れと遊脚期開始遅れを足し合わせ、その遅れ回数で除したものである。さらに、一点鎖線は「平均遊脚期終了遅れ時間」、「平均遊脚期開始遅れ時間」、「平均遊脚期遅れ時間」に対して線形近似を行ったものである。Fig.19-1より、学習用データが長いほど遊脚期推定値の遅れ時間が短くなる傾向にあることがわかる。Fig.19-2はFig.19-1と同様に、N.N.構造（Ⅱ）の場合に対しての学習データ長さ（Case1～Case8）と遊脚期遅れ時間との関係を示す。Fig.19-2より、N.N.構造（Ⅱ）ではN.N.構造（Ⅰ）と同じように、学習用データが長いほど遊脚期推定値の遅れ時間が短くなる傾向にあることがわかる。ここで、Fig.19-1に示すN.N.構造（Ⅰ）の場合とFig.19-2に示すN.N.構造（Ⅱ）の場合の遅れ時間を比較すると、N.N.構造（Ⅱ）での遅れ時間はN.N.構造（Ⅰ）の遅れ時間よりも短いことがわかる。つまり、実験を行う前は、健常者では足を大きく外側にぶん回すことがないためy軸加速度情報が不必要であると予測していたが、実際にはy軸加速度情報も利用したほうが、より正確に遊脚期を推定することが可能であった。

6. 結 言

本研究では、開発した遊脚期検出システムが健常者に対して、遊脚期の推定が可能であることが確かめられた。しかしながら、遊脚期の推定値はフットスイッチ出力と比べて、時間的な遅れが生じている。

推定時間遅れの理由としては、マイコンでの処理時間による遅れ、また、移動平均法によって生じる遅れの影響が大きいと考えられる。また、N.N.の学習において、学習用データが長いほど推定エラーが少なく、また、推定時間の遅れも短くなる傾向にあることが確かめられた。本研究では健常者の直線歩行を想定して行った。予想としては、歩行の進路方向に対して水平方向、つまりy軸方向は不必要な加速度センサと考えていたが、実際は、y軸方向の加速度センサもN.N.の入力情報として用いることによって、より正確に遊脚期判定を行うことが可能であることも実験によって確認できた。これらの考察から下垂足患者は、足を外側に振り回すように歩くため、左右軸加速度には大きな特徴があると考えられる。そのため、y軸加速度を用いることは、より正確な遊脚期推定に役立つと思われる。

今後は、本研究で提案した非接触型センサがフットスイッチに取って代わるFES装置のための遊脚期検出システムとして利用できるように、遊脚期判定の遅れ時間を0.1sec以内に抑えられるようにシステムの改良を行っていきたい。また、本研究で開発したシステムが実際の患者に対してどの程度有効であるかを確かめていきたい。

参考文献

- (1) 齋藤, 他5名, 日本機械学会ジョイントシンポジウム2008講演論文集, No.08-23, 2008, 365-367