

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3610455号  
(P3610455)

(45) 発行日 平成17年1月12日(2005.1.12)

(24) 登録日 平成16年10月29日(2004.10.29)

(51) Int. Cl.<sup>7</sup>

A61B 5/0488  
H03M 7/16

F I

A61B 5/04 330  
H03M 7/16

請求項の数 1 (全 13 頁)

(21) 出願番号	特願2001-20880 (P2001-20880)	(73) 特許権者	301021533 独立行政法人産業技術総合研究所 東京都千代田区霞が関1-3-1
(22) 出願日	平成13年1月30日(2001.1.30)	(73) 特許権者	501039237 梶谷 勇 茨城県つくば市梅園1丁目1番1 中央第 2 独立行政法人産業技術総合研究所内
(65) 公開番号	特開2002-224071 (P2002-224071A)	(73) 特許権者	597073531 樋口 哲也 茨城県つくば市梅園1丁目1番1 中央第 2 独立行政法人産業技術総合研究所内
(43) 公開日	平成14年8月13日(2002.8.13)	(72) 発明者	梶谷 勇 茨城県つくば市梅園1丁目1番4 経済産 業省産業技術総合研究所電子技術総合研 究所内
審査請求日	平成13年1月30日(2001.1.30)		最終頁に続く

(54) 【発明の名称】 筋電パターン識別装置

(57) 【特許請求の範囲】

【請求項1】

筋肉の活動電位である筋電パターンから特徴パターンを抽出し、その特徴パターンをビットパターンに符号化し、そのビットパターンのパターン識別を行うことによって出力制御信号を生成する筋電パターン識別装置において、ビットパターンへの符号化に冗長符号を用いる筋電パターン識別装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、筋肉の活動電位（筋電パターン）をインタフェースとする方法、装置における筋電パターン識別方法及び装置に関する。

【0002】

【従来の技術】

図2は、従来技術の筋電パターンから特徴量を抽出して、制御対象を操作する装置を説明するための図である。図中、1は筋電パターン、2は表面電極群、3は増幅、平滑化装置、4は特徴パターン抽出装置（FFT、逆写像など）、5は符号化装置（バイナリ符号・グレイ符号）、6はパターン識別器、7は制御対象（例：モータ、ロボット、福祉機器、リハビリテーション機器、筋電義手、ゲームなど）をそれぞれ示している。

【0003】

図示したように、複数の筋肉の協調活動から発生する活動電位である筋電パターン 1 は、皮膚表面上の一つ、または複数個の電極群 2 (表面電極群) で測定される。ここで測定されるのは、複数の筋肉から発生した活動電位の和である。次に、それらは、増幅、平滑化装置 3 によって増幅、平滑化を施される。特徴パターン抽出装置 4 では、特徴パターンを抽出する。符号化装置(バイナリ符号・グレイ符号) 5 では、それらを、バイナリ符号やグレイ符号を用いて、ビット列に符号化する。パターン識別器 6 では、それらの識別を行い、制御対象 7 の制御信号を生成する。ここでは、論理回路などの論理値フィルタが用いられる。

【0004】

このような従来技術は、符号化にバイナリ符号、グレイ符号(表1)を用いるために、パターン識別器の設計に時間がかかる場合や、複雑なパターン識別器が必要な場合がある。このために、小型化、低価格化が妨げられ、筋電パターンをインタフェースとする装置の普及が妨げられてきた。

【表1】

特徴量	バイナリ符号 $X_1 X_2 X_3 X_4$	グレイ符号 $X_1 X_2 X_3 X_4$
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

10

20

30

また、バイナリ符号、グレイ符号の場合は、自発的でない筋張力変化による筋電パターン 40  
 の変化によって、正確なパターン識別が困難になる場合があり、このために、適用可能な  
 応用が限られてきた。

【0005】

【発明が解決しようとする課題】

そこで、本発明は、かかる問題点を解決して、冗長符号化法を用いることで、単純な演算  
 処理装置や、表引き装置によって実現可能にして、小型化、低価格化を図ることを目的と  
 している。

さらに、これを用いることによって識別精度を改善し、筋電をインタフェースとする装置  
 の普及を促進することを目的としている。

【0006】

50

【課題を解決するための手段】

本発明の筋電パターン識別方法は、複数の筋肉の協調活動から発生する活動電位である筋電パターンを皮膚表面上で表面電極群により測定し、測定された筋電パターンから特徴パターンを抽出し、抽出された特徴パターンを冗長符号によってビットパターンに符号化し、符号化されたビットパターンのパターン識別を行って出力制御信号を生成することから成る。

【0007】

また、本発明の筋電パターン識別装置は、複数の筋肉の協調活動から発生する活動電位である筋電パターンを皮膚表面上で測定する表面電極群と、測定された筋電パターンから特徴パターンを抽出する特徴パターン抽出装置と、抽出された特徴パターンを冗長符号によ

10

【0008】

【発明の実施の形態】

図1は、本発明の筋電パターンから特徴量を抽出して、制御対象を操作する装置を説明するための図である。図中、1は筋電パターン、2は表面電極群、3は増幅、平滑化装置、4は特徴パターン抽出装置、8は符号化装置（冗長符号）、6はパターン識別器、7はモータ群をそれぞれ示している。

【0009】

冗長符号の符号化装置8を除いて、前述した従来技術と同じである。図示したように、複数の筋肉の協調活動から発生する活動電位である筋電パターン1は、皮膚表面上の一つ、または複数個の電極群2（表面電極群）で測定される。ここで測定されるのは、複数の筋肉から発生した活動電位の和である。次に、それらは、増幅、平滑化装置3によって増幅、平滑化を施される。（ただし、平滑化を行わない場合もある。）。特徴パターン抽出装置4では、特徴パターンを抽出する。パターン識別器6では、それらの識別を行い、制御対象7の制御信号を生成する。ここでは、論理回路などの論理値フィルタが用いられる。

20

【表2】

特徴量	冗長符号 $X_1 X_2 X_3 X_4$
0	0000
1	0001
2	0011
3	0111
4	1111
5	1110
6	1100
7	1000

30

40

【0010】

【実施例】

ここでは、本手法の効果を示すために、筋電義手の動作決定に本手法を用いることによって、識別精度を改善した例を示す。

筋電は数 $\mu$ V～数mV程度の微弱な信号であるため、その測定においては、一般的に、2個の電極で測定した信号を差動増幅することによって、電源やその他の機器から発生するノイズの影響を小さくする必要がある。そこで、ここでは、図3に示す8個（4組）の電極を前腕周りの4ヶ所に張り付けて筋電を測定する。本装置は、これらの他に、1個のアース電極、測定した筋電を差動増幅する増幅器、ノート型パーソナルコンピュータ（ノー

50

トPC)、AD変換器からなり、AD変換器とノートPCはSCSIインタフェースで接続する。このAD変換器では、4ヶ所で測定した筋電を、それぞれ1000Hzで離散化し、12ビットのデジタル信号に変換してノートPCに取り込む。また、この測定においては、商用電源から発生するノイズを取り除くために、商用電源周波数(50Hz)を遮断するアナログフィルタを用いた。

【0011】

以下の説明では、時刻  $t$  で測定した信号の組を、数1であらわすことにする。

【数1】

$$f(t) = (f_1(t), f_2(t), f_3(t), f_4(t))$$

10

ここでは、6つの動作(前腕の回内、回外、手首の屈曲、伸展、手先の開閉)を意図しながら筋肉を収縮させたときに、筋電を測定する。つまり、図4(a)に示す筋収縮(2秒間の脱力の後に3秒間筋収縮を維持する)を6つの動作毎に20回(合計120回)繰り返しているときに筋電を測定する。ただし、前腕切断者の場合は、切断前の動作イメージをもとに筋肉を収縮したときの筋電を測定する。図4(b)に、手首を屈曲した時に前腕の屈筋群上の皮膚表面で測定した筋電の例を示す。

【0012】

測定した筋電信号は、高周波成分を取り除くために平滑化処理を施す。一般的に、この平滑化処理にはアナログローパスフィルタを用いることが多いが、本測定では、情報を失わないためにローパスフィルタをかけずにPCに取り込んだので、PC上で平滑化処理を施す。具体的には、時刻  $t$  における平滑整流値を数2に示す計算によって求める。図4(c)に平滑整流した筋電の例を示す。

20

【数2】

$$g(t) = \left( \sum_{i=1}^{100} |f_1(t-i)|, \sum_{i=1}^{100} |f_2(t-i)|, \sum_{i=1}^{100} |f_3(t-i)|, \sum_{i=1}^{100} |f_4(t-i)| \right)$$

【0013】

義手の動作決定に用いる筋電の特徴パターンの抽出では、一般的に、(1)筋収縮開始時に測定した筋電、(2)筋収縮を維持した状態(以下では定常状態と呼ぶ)で測定した筋電から抽出する方式が用いられる。(1)の場合、筋肉を収縮しはじめてから義手が動き出すまでの遅れが小さい利点がある。ところが、動作識別率に関しては(1)よりも、(2)の方式の方が良いことが知られているので、ここでは、定常状態の筋電から特徴パターンを抽出する。

30

抽出する特徴パターンとしては、定常状態の筋電の平滑整流値を用いる。この平滑化においても、一般的にはアナログローパスフィルタを用いるが、ここでは、PC上で数2で求めた値の1秒間の平均値を計算し、これを特徴パターンとする。

【0014】

この特徴パターンの抽出においては、筋電測定時の被験者の疲労を小さくするために、1回の筋収縮から10個の平均値を求める。つまり、図5に示すように、1秒毎の平均値を、動作開始1秒後から100ミリ秒づつずらして、数3に示す計算によって求める。ここで  $i$  は動作を開始した時間、つまり数4で求めた値が、あらかじめ設定された閾値を越えた時間とする。

40

【数3】

$$ave(n) = \frac{\sum_{j=1}^{1000} g(i+1000+j+n*100)}{1000}, (n=0, 1, \dots, 9)$$

【数4】

$$G(t) = g_1(t) + g_2(t) + g_3(t) + g_4(t)$$

50

## 【 0 0 1 5 】

しかしながら、実際には、筋肉の収縮の仕方によっては数 4 の値が閾値を越えない場合もある。このため、動作毎 20 回の筋収縮のすべてから数 3 を計算できるわけではない。そこで、パターン識別器を学習させるために用いるトレーニングパターンとしては、数 4 の値が敷居値を越えた筋収縮のうち 5 回から抽出した数 3 の値 ( 5 [ 回の筋収縮 ] × 10 [ パターン ] × 6 [ 種類の動作 ] = 300 [ パターン ] ) を用いる。その他の 5 回の筋収縮から抽出した数 3 の値は、パターン識別器を評価するためのテストパターンとして用いる。以下の説明では、作成したトレーニングパターンを、数 5 で表し、作成したテストパターンを、数 6 で表す。

## 【 数 5 】

$$\mathbf{Tr}_{org}(n) = (Tr_{org,1}(n), Tr_{org,2}(n), Tr_{org,3}(n), Tr_{org,4}(n)), (n = 0, 1, \dots, 299)$$

10

## 【 数 6 】

$$\mathbf{Te}_{org}(n) = (Te_{org,1}(n), Te_{org,2}(n), Te_{org,3}(n), Te_{org,4}(n)), (n = 0, 1, \dots, 299)$$

## 【 0 0 1 6 】

図 6 に、作成したトレーニングパターンの第 1 成分を X 軸、第 2 成分を Y 軸としてプロットした例を示す。

20

ここでは、さらに、別出願した対数変換を用いて筋電パターンから特徴パターンを抽出する。具体的には、以下の数 7、数 8 を用いて対数変換を行う。

## 【 数 7 】

$$\begin{aligned} \mathbf{Tr}_{log}(n) &= (Tr_{log,1}(n), Tr_{log,2}(n), Tr_{log,3}(n), Tr_{log,4}(n)) \\ &= (-\log(Tr_{org,1}(n)), -\log(Tr_{org,2}(n)), -\log(Tr_{org,3}(n)), -\log(Tr_{org,4}(n))) \\ &(n = 0, 1, \dots, 299) \end{aligned}$$

## 【 数 8 】

$$\begin{aligned} \mathbf{Te}_{log}(n) &= (Te_{log,1}(n), Te_{log,2}(n), Te_{log,3}(n), Te_{log,4}(n)) \\ &= (-\log(Te_{org,1}(n)), -\log(Te_{org,2}(n)), -\log(Te_{org,3}(n)), -\log(Te_{org,4}(n))) \\ &(n = 0, 1, \dots, 299) \end{aligned}$$

30

## 【 0 0 1 7 】

以下では、数 5、数 7 で求めたトレーニングパターン、それぞれを用いてパターン識別器の訓練を行い、数 6、数 8 で求めたテストパターンを用いて識別器の評価を行う。また、以下では、説明を簡単にするために、トレーニングパターン、テストパターンを、数 9 を用いてあらわす。

## 【 数 9 】

$$\mathbf{Tr} = (Tr_1(n), Tr_2(n), Tr_3(n), Tr_4(n)), \mathbf{Te} = (Te_1(n), Te_2(n), Te_3(n), Te_4(n))$$

40

## 【 0 0 1 8 】

ここでは、論理回路によって筋電パターンの識別を行うので、識別に用いるパターン識別器としては、進化型チップを用いる。この進化型チップの特徴は、従来のハードウェアは回路構成がその設計プロセスによって固定的であるのに対し、その回路構成を動的に、かつ、自律的に変更可能な能力によって、仕様の変更や動作環境の変化に対して適応できるように設計されている。

## 【 0 0 1 9 】

50

進化型チップは、回路構成を何度でも変更可能なハードウェア（再構成可能ハードウェア）と、その回路構成を適応的に変更する回路から構成される。この再構成可能ハードウェアの回路構成は、コンフィグレーションビット列と呼ばれるソフトウェアビット列をダウンロードすることによって、何度でも変更できる。また、回路構成の適応的な書き換えには、遺伝的アルゴリズム（Genetic Algorithm；以下ではGAと略す）と呼ばれる探索手法を高速に実行する回路を用いる。GAは、生物の進化に発想を得た探索手法で、複数の解の候補をそれぞれ0、1のビット列に符号化することによって、並列に最適解を探索する。つまり、これらのビット列を染色体と見なし、交叉や突然変異と呼ばれる操作を施して変化させることで、新しい解の候補を作る。そこで、あらかじめ決められた評価関数によって、各染色体（＝解の候補）がどれだけ解に近いかを評価する。この評価結果に基づき、解から遠いものを削除し、解に近いものを残す操作を繰り返すことによって、解を探索する。

10

## 【0020】

進化型チップでは、再構成可能ハードウェアの回路構成を指定するコンフィグレーションビット列を染色体と見なす。そこでGAを適用して染色体を進化させることによって、評価関数が適切に設定されていれば、最適な回路構成を指定するコンフィグレーションビット列を得ることができる。

この進化型チップは、（1）GA操作用回路、（2）再構成可能ハードウェア、（3）染色体を保存するメモリ（染色体用メモリ）、（4）トレーニングパターンを保存するメモリ（トレーニングパターン用メモリ）、（5）入出力インタフェースからなる

20

## 【0021】

再構成可能ハードウェアとしては、Programmable Logic Array（以下ではPLAと略す）が用いられている。PLAは、図7に示す例のように、ANDアレイ（接続している入力信号の論理積を作る）とORアレイ（接続しているANDアレイの出力信号の論理和を作る）からなる。この図では、黒丸と白丸が、それぞれ、入力と出力間の接続を決定するスイッチを示しており（黒丸が接続を意味する）、このスイッチのON/OFFをコンフィグレーションビット列で指定することによって、任意の論理回路を構成できる。

## 【0022】

進化型チップによるパターン識別では、論理回路によって識別を行うため、識別パターンを0、1のビット列に変換しなければならない。そこで、ここでは、筋電の特徴パターンを数10、数11に示す計算で0～15の実数に変換し、それらを整数化したものを4ビットのビット列に符号化する。

30

## 【数10】

$$\begin{aligned}
 sTr(n) &= (sTr_1(n), sTr_2(n), sTr_3(n), sTr_4(n)) \\
 &= 15 \times \left( \frac{Tr_1(n) - Tr_{1_{min}}}{Tr_{1_{max}} - Tr_{1_{min}}}, \frac{Tr_2(n) - Tr_{2_{min}}}{Tr_{2_{max}} - Tr_{2_{min}}}, \frac{Tr_3(n) - Tr_{3_{min}}}{Tr_{3_{max}} - Tr_{3_{min}}}, \frac{Tr_4(n) - Tr_{4_{min}}}{Tr_{4_{max}} - Tr_{4_{min}}} \right) \\
 &(n = 0, 1, \dots, 299) \\
 &(Tr_{1_{min}}, Tr_{2_{min}}, Tr_{3_{min}}, Tr_{4_{min}}) : Tr_1(n), Tr_2(n), Tr_3(n), Tr_4(n) \text{の最小値} \\
 &(Tr_{1_{max}}, Tr_{2_{max}}, Tr_{3_{max}}, Tr_{4_{max}}) : Tr_1(n), Tr_2(n), Tr_3(n), Tr_4(n) \text{の最大値}
 \end{aligned}$$

40

## 【数11】

$$\begin{aligned}
 sTe(n) &= (sTe_1(n), sTe_2(n), sTe_3(n), sTe_4(n)) \\
 &= 15 \times \left( \frac{Te_1(n) - Tr_{1min}}{Tr_{1max} - Tr_{1min}}, \frac{Te_2(n) - Tr_{2min}}{Tr_{2max} - Tr_{2min}}, \frac{Te_3(n) - Tr_{3min}}{Tr_{3max} - Tr_{3min}}, \frac{Te_4(n) - Tr_{4min}}{Tr_{4max} - Tr_{4min}} \right) \\
 &\quad (n = 0, 1, \dots, 299) \\
 &\quad (Tr_{1min}, Tr_{2min}, Tr_{3min}, Tr_{4min}) : Tr_1(n), Tr_2(n), Tr_3(n), Tr_4(n) \text{ の最小値} \\
 &\quad (Tr_{1max}, Tr_{2max}, Tr_{3max}, Tr_{4max}) : Tr_1(n), Tr_2(n), Tr_3(n), Tr_4(n) \text{ の最大値}
 \end{aligned}$$

## 【 0 0 2 3 】

従来、このビット列への符号化では表 1 に示すバイナリ符号や、グレイ符号が用いられてきた。ところが、この符号化法を用いる場合、次に示す問題がある。つまり、筋電の特徴パターンは、図 6 に示した例のように、動作毎に任意の連続的な領域に分布していることが多い。

10

そのため、この特徴パターンを論理回路によって識別するには、任意の連続した特徴量の値に対して 1 を出力する回路が必要である。しかしながら、グレイ符号の場合は、連続した値に対して 1 を出力するために、複雑な回路が必要な場合がある。以下に示す例では、符号化したビット列、つまり回路の入力信号を数 1 2 と表すことにする。

## 【 数 1 2 】

$$X_1 X_2 X_3 X_4$$

20

## 【 0 0 2 4 】

例えば、連続区間 0 ~ 7 に対してのみ 1 を出力する場合は、数 1 3 で表される回路によって、この区間の値に対して 1 を出力できる。

## 【 数 1 3 】

$$\overline{X_1}$$

ところが、連続区間 5 ~ 12 の時だけ 1 を出力する場合は、3 つの積項（入力信号の論理積）から成る回路（数 1 4）が必要である。進化型チップによる回路合成では、一般的に、構成する入力信号の数が多い積項（例：4 つの入力信号からなり、数 1 2 で表される積項や、上の例に示した積項（数 1 5）や積項（数 1 6）など）を多く必要とする回路の合成に時間がかかることが知られている。

30

## 【 数 1 4 】

$$X_2 X_4 + X_2 \overline{X_3 X_4} + X_1 X_3 \overline{X_4}$$

40

## 【 数 1 5 】

$$X_2 \overline{X_3 X_4}$$

## 【 数 1 6 】

$$\overline{X_1 X_3 X_4}$$

## 【0025】

このため、グレイ符号を用いた場合、識別回路の合成に時間がかかる場合があり、これが、識別精度を下げる一因になっていると考えられる。これに対して、冗長符号を用いて符号化を行うことにより、識別回路を単純な論理回路で実現できるため、進化型チップによる識別回路の合成を高速化でき、識別精度を改善できる。

そこで、作成したトレーニングパターンを用いて進化型チップの回路を合成し、トレーニングパターンによって識別精度の評価を行ったところ、冗長符号を用いることによって、識別精度を、13人の平均で3.1%、最大で10.5%改善できることを実証した。

## 【0026】

さらに、回路合成に要した時間として、トレーニングパターンを用いて回路を評価した回数、13人の平均値を比較すると、冗長符号を用いない場合に451582.71回必要であったのに対し、冗長符号を用いた場合には314333.91回であった。つまり、提案手法によって69.6%にまで短縮した。

本発明の特徴とする符号化装置（冗長符号）8では、特徴パターンを表2に示す符号を用いて、ビットパターンに符号化する。この符号はジョンソンカウンタの出力パターンとして用いられている符号であるが、冗長な符号化法であるため、4ビットの場合は0~7までしか符号化できない。

## 【0027】

この冗長符号の特徴は、任意の連続的な値に対して、単純な回路だけで1を出力できることである。つまり、この符号では、2つの値の差が1である場合（区間長2）に、それらの符号は常に1ビットだけ異なり、差が2の場合は2ビット、差が3の場合は3ビットだけ異なるように設計されている。

例えば、1の冗長符号は0001、2の冗長符号は0011であるので、これらは1ビットしか異ならない。また、1と3の場合は、それらの符号が0001と0111であるので2ビットだけ異なる。このため、区間長2の任意の連続的な値に対して1を出力する場合は、3つの入力信号で構成される積項を持つ回路が必要である。例えば、区間1~2の場合は数17で表される積項を持つ回路によって、これらの区間に対して1を出力可能である。

## 【数17】

$$\overline{X_1 X_2 X_4}$$

## 【0028】

同様に、区間長3の場合は2つ、区間長4の場合は1つの入力信号で構成される積項だけで、それらの区間の値に対して1を出力する識別回路を実現できる。さらに、区間長が5~7の場合は、区間長4の場合に必要な積項を2つ組み合わせることによって実現できる。

例えば、区間1~6（区間長6）の場合は区間1~4に対して1を出力する積項（数18）と、区間3~6に対して1を出力する積項（数19）の論理和（数20）によって、この区間の値に対して1を出力する回路を実現できる。

## 【数18】

10

20

30

40

$$X_4$$

【数 19】

$$X_2$$

10

【数 20】

$$X_2 + X_4$$

【0029】

【発明の効果】

このように、本発明は、連続的な領域に分布する筋電パターンを識別する識別器を簡単な回路で実現できる。 20

また、本発明は、冗長符号化法を用いることで、単純な演算処理装置や、表引き装置によって実現できるため、小型化、低価格化が可能である。

これによって、さらに、制御可能な制御対象の数を増やすことが可能となり、このため、筋電をインタフェースとする装置の普及を促進することが可能となる。

【図面の簡単な説明】

【図1】本発明を説明するための図。

【図2】従来技術を説明するための図。

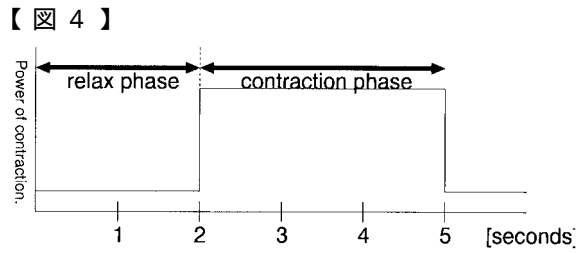
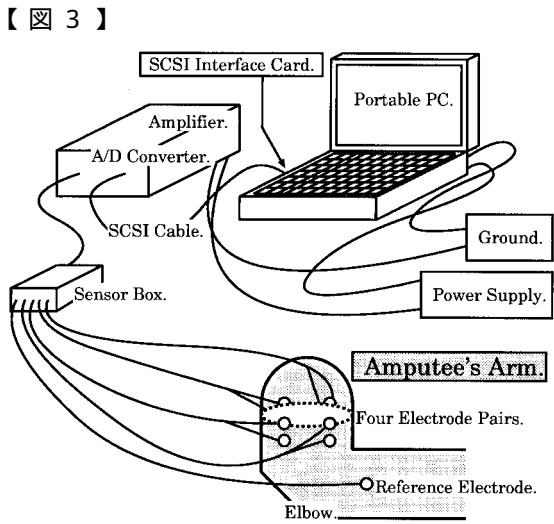
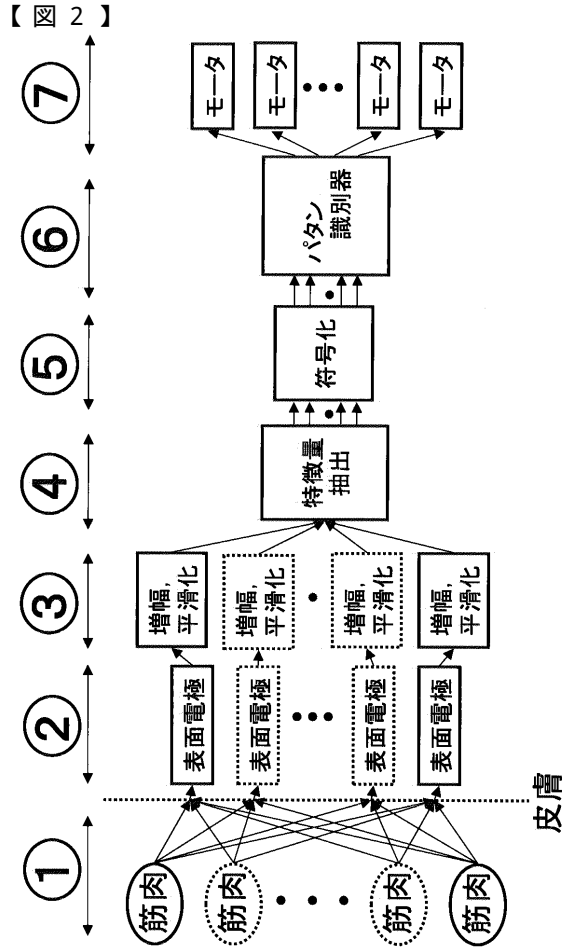
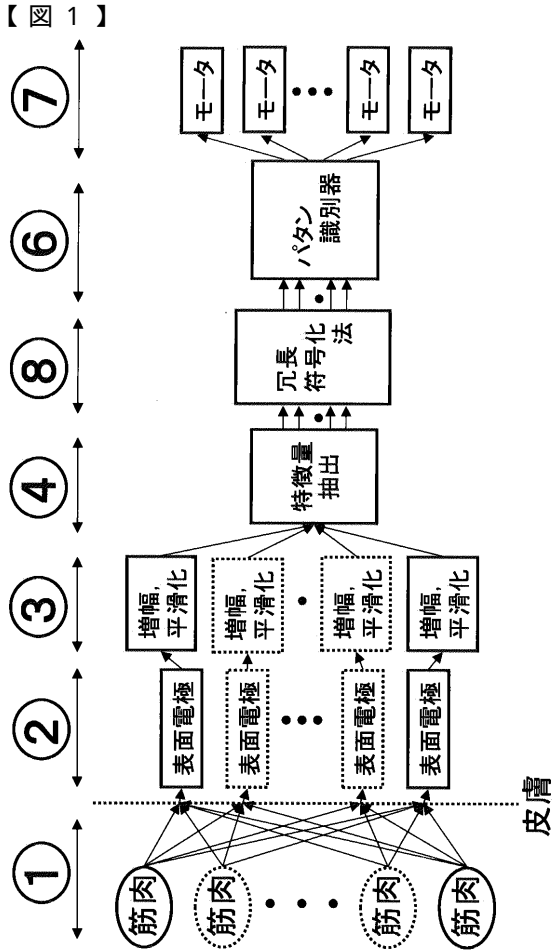
【図3】筋電測定装置を説明するための図である。

【図4】測定した筋電の例。 30

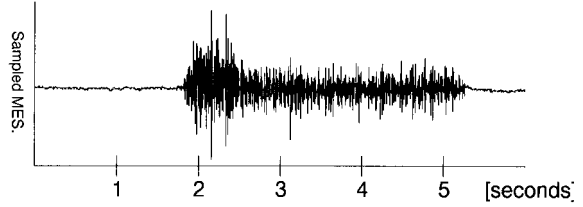
【図5】特徴パターンの抽出方法を説明するための図。

【図6】筋電の特徴パターンの分布図。

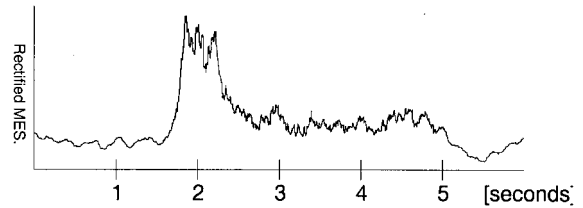
【図7】PLAの構造を説明するための図。



(a) The muscle contraction scheme.

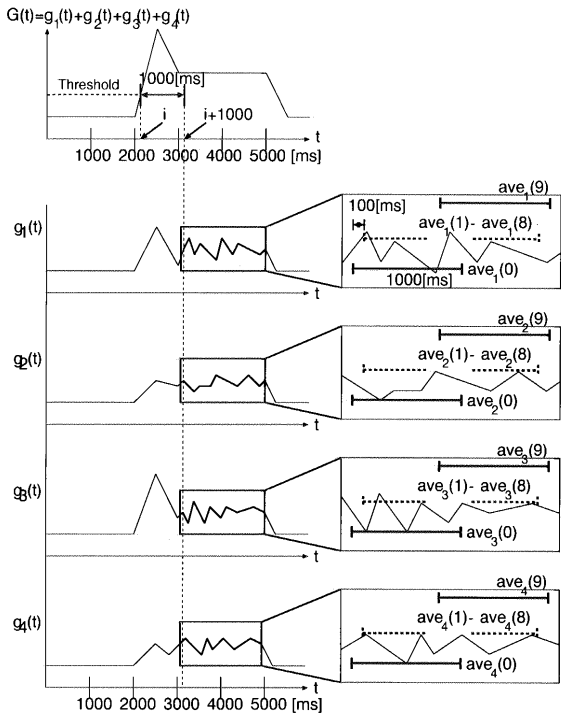


(b) An example of sampled myoelectric signals.

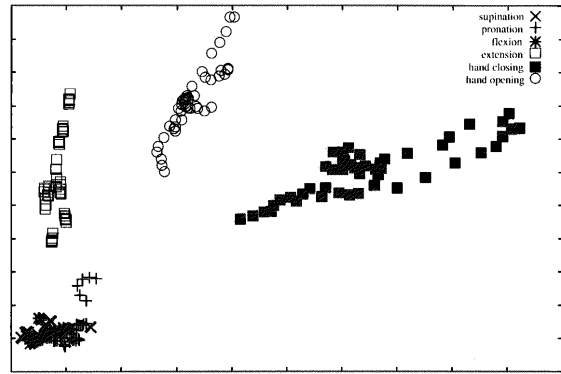


(c) An example of the rectified myoelectric signals.

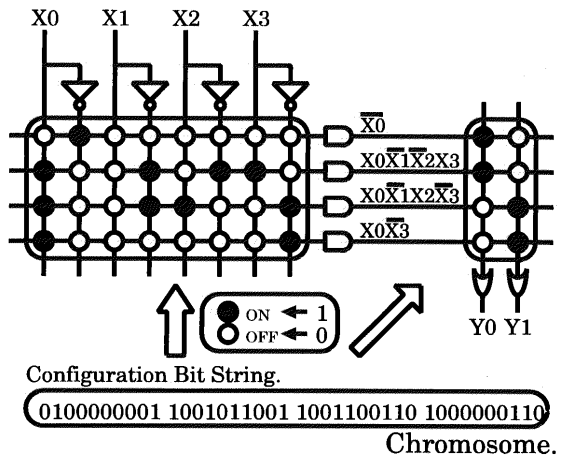
【 図 5 】



【 図 6 】



【 図 7 】



---

フロントページの続き

(72)発明者 樋口 哲也

茨城県つくば市梅園1丁目1番4 経済産業省産業技術総合研究所電子技術総合研究所内

審査官 門田 宏

(56)参考文献 梶谷勇 他, 障害者に適応可能な筋電操作型義手の開発, 電子情報通信学会技術研究報告, 日本  
 , 社団法人電子情報通信学会, 1998年 9月18日, Vol.98、No.275, P.9  
 - 16

(58)調査した分野(Int.Cl.<sup>7</sup>, DB名)

A61B 5/0488

JICSTファイル(JOIS)