フリック動作による移動ロボットの操作
(ニューラルネットワークを用いた動作モードの切り替え)山本 智規^{*1}, 柴田 諭^{*1}Operation of the mobile robot using flick motion
Switching of the control method using the neural networkTomonori YAMAMOTO^{*1} and Satoru SHIBATA^{*1}^{*1} Ehime Univ. Graduate School of Science and Engineering
3 Bunkyo-cyo, Matsuyama-shi, Ehime 790-8577, Japan

Received 19 January 2015

Abstract

In this research, a mobile robot system moved by intuitive gesture instructions using the flick motion is suggested. The flick motion is commonly used in operating smart phones and involves sudden and rapid motion of the hand. One of the aim of this research is to move the robot quickly using this flick motion. At first, a flick motion is measured and the characteristic of the movement is analyzed. Next, a model to move a mobile robot in response to flick movement is proposed. In the model, the velocity of the hand is converted into virtual force, and the virtual force moves an object with virtual mass and viscosity. Another aim is introduction of a mechanism automatically switching the flick instruction and the following instruction. For the purpose, the switcher using the neural network is learned in advance to distinguish between the movement of the flick instruction and the following instruction. The experiment was conducted to examine the usefulness of the proposed system. As the results, flick instructions and following instructions were changed automatically and were able to operate a robot to the aim position comfortably.

Key words : Flick motion, Robot, Human-machine interface, Neural network, Kinect, Gesture, Instruction

1. 緒 言

これまで、ロボットは主に産業分野で利用され、工場内の人間と隔離された作業空間で専門のオペレータにより操作されてきた。しかしながら近年、医療、福祉、生活分野において人間をサポートするロボットの導入が検討され、そのための研究が盛んに行われている(例えば、小俣, 2011, 松日楽, 2006)。このように、人間をサポートするロボットは、人間の近傍で動作し、またロボットに対する専門的な知識を有しない一般のユーザが使用することに注意しなければならない。すなわち、人間と隔離された空間で動作する産業用ロボットに求められるような作業効率や信頼性のほかに、使用する人間や周囲の人間に対して恐怖感や違和感を与えない、人間の心理・感性に受けいられらるような動きを生成する必要がある(大西他, 2006)。また、サポートロボットに対する指示は、直感的で簡潔なものであることが望ましい(細田, 2006)。

そのようなインタフェースの一つとして、手を使ってロボットを操作するシステムが考えられる。手先を用いたジェスチャ指示に関する研究は種々報告されている。鈴木ら(鈴木他, 2003)は、手先の動作を全方位カメラを使って認識し、移動ロボットを操作するシステムを提案している。また、長谷部ら(長谷部他, 2010)はコンピュータ

にクリック動作を認識させる際、カメラで捉えた指差し動作と指の位置をジェスチャとして提案している。谷ら(谷他, 2003)は、テープセンサを利用して、手振り動作によってロボットを動かすシステムを提案しており、浅野ら(浅野他, 2010)は、部屋の中に複数台のカメラを設置して、操作者の手振りを検出するシステムを提案している。これらの研究をはじめとして、人間の手の動きを用いたジェスチャ指示は、コンピュータやロボットに直感的に指示を与えることが出来る手法として注目されている。

一方、これらの研究では、直感的な操作でロボットに指示を与えることができるが、指示によって、ある定められた動作目的のみが与えられ、その動作目的を達成するに至るロボットの運動特性にはあまり重点が置かれていない。人間の身近な環境で活躍するロボットを考えた場合、動作目的を達成する過程でのロボットの運動特性は、人間心理に強く影響することが知られている(柴田他, 2009)。このことから、動作目的に至るまでのロボットの運動特性を考慮した指示が必要といえる。

ところで近年、スマートフォンやタブレット端末の普及率が全人口の30%を超えるなど、その利用者数が増加している。これらを利用するとき、画面を指ではじくような動作、いわゆる“フリック動作”を用いて画面に表示されている内容を高速にスクロールして操作を行う。この操作、および画面のスクロールの動きは、これらの情報機器を利用するのに慣れた人にとってはもちろんのこと、初心者にとっても直感的に理解しやすく、扱いやすいものである。そこでこのフリック動作に着目し、ロボットを操作するときの指示方法として応用することを考える。すなわち、このフリック動作をロボットを弾いて動かすようなジェスチャとみなし、ロボットを素早く動かすシステムの構築をまず試みる。本研究は、フリック指示をロボットへの指示とする基礎的な研究のため、スマートフォンやタブレット端末でフリック動作を利用したことのあるユーザーを対象として検討を行っていく。

ここで、サポートロボットは、家庭などでの使用を考慮すると、システム自体が安価であるのが望ましい。一般的に、手先の動きを読み取るのに使われるセンサおよびカメラは高価であり、構築されるロボットシステム全体が高価となる傾向にある。また、人間の手先にセンサやマーカーをつける場合、その影響で自然な動きがなくなる可能性が否定できず、センサやマーカー装着の手間を考えても現実的ではない。非接触で手先の動きを計測でき、安価であるセンサの1つとして、最近、Kinect センサ (Microsoft) が注目されている。Kinect センサは、ゲームデバイスとして販売されており、一般的なコンピュータに接続してセンサから対象物までの距離データを得ることが可能である。さらに、得られたセンサからの距離情報から、人体モデルとのフィッティングを行い、ジェスチャの抽出などを行うプログラムなども用意されており (Microsoft Kinect for Windows SDK, OpenNI 等)、容易に人間の手先位置などを取得することが可能である。そこで本研究では、人間の手先指示運動を計測するセンサとして、この Kinect センサの適用を試みる。

フリック動作を用いた指示によりロボットを操作するシステムでは、手先の小さな運動からロボットの大きな運動を生成することが期待される。一方、細かい軌道や速度をロボットに指定可能な手法として、筆者ら(柴田他, 2009)は、操作者が指先を移動させることにより指示運動を生成し、その指示に追従してロボットをある位置から別の位置へ移動させるシステムを報告している。これは、人間の動作の特徴である滑らかな軌道や速度変化をロボットに持たせつつ、その軌道や速度を人間の意図通りに動かすことを可能にしている。

これらのことから、フリック動作を用いた指示と、人間の手先運動にロボットを追従させる指示の両者を自由に操作者が選択し、利用できるシステムを構築する。これにより、操作者が小さな手先の動きでロボットを大きく動作させたい場合はフリック動作による指示(以下、フリック指示と称する)を用い、位置決め精度が必要な細かい指示を行いたい場合は手先の動きに追従するようにロボットを操作する指示(以下、追従指示と呼称する)を用いることが可能となる。

このシステムを実現するためには、Kinect センサより得られた手先運動(手先位置情報の時系列データ)からフリック指示、追従指示という2つの異なった目的を指示途中に認識し、指示に適した運動をロボットに生成させる必要がある。その認識には、学習性に優れ、任意の入出力関係を同定できることで知られているニューラルネットワークを用いることを提案する。そして、ニューラルネットワークを用いて、入力された手先運動がどちらの指示であるかを指示動作中に判別し、その結果に基づいてロボット運動を生成するシステムを構築する。

Fig. 1 Measurement system using Kinect sensor

2. フリック指示運動の計測

フリック動作は、スマートフォンなどの画面を指ではじくような動作であり、表示されているリストなどの内容を高速にスクロールさせる動作として知られている。この弾くような動作でロボットを操作することが出来れば、単純な移動の場合などでは指示回数が少なく済み、負担低減が出来ると考えられる。ここで、スマートフォンなどでのフリック動作では、動かす対象は画面に表示されている内容であり、画面に指で触れ、その指を弾くような動きをする。一方、ロボットへのジェスチャ指示としてのフリック動作を考えた場合、スマートフォンなどのように、画面への指の接触を検知する手法は使用できない。このため、非接触で人体の動きを検出することが出来るデバイスである Kinect センサを使用し、手先の動きの計測を行う。Kinect センサでは、センサからの距離情報を取得し、その情報から人間の形状を時々刻々と追跡することで、身体の一部に計測器具をつけることなしに体の動きを計測することができる。

2.1 フリック指示動作の計測

実際に人間がフリック指示を行っているときの手の動きを計測する。ここでは、室内でロボットシステムを使用する場面を想定し、8 畳程度の室内 (3.4 m × 3.4 m) において指示を出すものとした。計測システムの構成を図 1 に示す。図のように被験者の足下を原点とする座標系において、Kinect センサを Y 軸方向に 1700 mm, Z 軸方向に 1200 mm の地点に配置する。

人間の右肩と右手先の位置を、Kinect センサを用いて取得する。そして、肩の位置を基準とした右手先の位置の変化をサンプリング時間ごとに計測する。

実験では、まず被験者に体の正面に手を伸ばした状態で待機してもらう。実験開始の合図とともに、被験者の正面を基準にして手前から奥 (Y 軸正方向)、奥から手前 (Y 軸負方向)、左から右 (X 軸正方向)、右から左 (X 軸負方向) の 4 方向に対して、フリック指示および追従指示を各 5 回ずつ計 20 回行ってもらい、計測を行う。ここで、追従指示は任意の二点間で右手を水平に自然に移動させるイメージで、フリック指示はロボットをはじくように操作することをイメージして動かすように指示した。この計測を、日ごろスマートフォンなどでフリック動作を利用している被験者 21 名に対して行った。なお、計測のサンプリングタイムは 50 msec である。

2.2 計測結果

図 2(a) に、フリック指示を行った際の手先の速度変化、図 2(b) に追従指示を行った際の手先の速度変化の代表的な一例をそれぞれ示す。図 2(a), 2(b) ともに手前から奥 (Y 軸正方向) に動作させた場合のものである。動作区間は、手先の速度が Y 軸正方向に 0 mm/s を超えた地点から、再び 0 mm/s になる間とした。また、点線はそれぞれの動作において同じ時間を躍度最小モデルで動かした場合の速度を示している。これは、人間の自然な手先 2 点間運動は躍度最小モデルで表現できることが知られており (Flash and Hogan, 1985)、人間が無意識のうちにやっている自然な上肢運動と、物体を弾くイメージで行うフリック動作を比較するためである。

(a) When the right hand is moved by the flick motion

(b) When the right hand is moved by the normal motion

Fig. 2 Features of the velocity pattern

図 2(a) より、フリック指示では短時間で速度を上昇させ、最大速度が大きい特徴が見て取れる．躍度最小モデルと比較すると、躍度最小モデルは左右対称の釣鐘型であるのに対し、フリック指示では速度ピークが動作前半に位置していることが分かる．一方、図 2(b) より、追従指示では、速度波形が左右対称の釣鐘型となっており、躍度最小モデルでの波形とほぼ同様の特徴を示している．これらの特徴は、被験者が行った 210 試行の内、210 試行に観測された (100%)．

これらのことから、フリック指示と追従指示は明らかに特徴の異なる動作であり、その特徴を判別できれば、フリック指示と追従指示の切り替えが可能になる．

3. ニューラルネットワークを利用した判別システムの構築

ロボットを直感的に操作する方法として、フリック指示と追従指示の 2 つのジェスチャ指示を組み合わせる操作方法を提案する．このシステムでは、ロボットが手先の動作 (位置や速度など) をもとに、その動作がフリック指示であるか追従指示であるかを判断し、それに応じてロボットの運動を生成する．このとき、位置や速度に対して、あらかじめ設定した値と比較することによる判断ではなく、動作の特徴から総合的に判断する必要がある．これは、人間の動作は、同じ意図を持った動作であっても全く同じ動きをするわけではないことや、様々な人が行う動作の違いなどに対応する必要があるためである．このような判断を行うため、学習性に優れ、任意の入出力関係を同定できるニューラルネットワークを用いる．

ニューラルネットワークを利用して、人間のジェスチャ指示から、ロボットの移動制御を行うときの制御系の概略図を図 3 に示す．この制御系では、ニューラルネットワークにより、手先位置の動きが追従指示またはフリック指示のどちらのものであるかを指示動作の初期に認識し、それに基づきロボットの目標運動軌道を生成する．

Fig. 3 Block diagram of the system

3.1 ニューラルネットワークを用いた判別システム

ニューラルネットワークを利用したフリック指示および追従指示の判別においては、右肩を基準とする右手先の相対座標（以下、手先座標）の時間的な変化を用いる。手先座標を、全被験者の腕長（手先を前方に伸ばした際の長さ）の最大値で割り-1.0 から 1.0 の範囲に正規化したものを入力 $Input(i)$ とする。 $Input(i)$ を式 (1) に示す。

$$\begin{cases} Input(k \times 3 + 1) = \frac{Righthand_x(k) - Rightshoulder_x(k)}{ArmLength} \\ Input(k \times 3 + 2) = \frac{Righthand_y(k) - Rightshoulder_y(k)}{ArmLength} \\ Input(k \times 3 + 3) = \frac{Righthand_z(k) - Rightshoulder_z(k)}{ArmLength} \end{cases} \quad (1)$$

ここで、 k はサンプリング時刻を表し、 $k=0$ が現在、 $k=1$ が 1 サンプリング前、 $k=2$ が 2 サンプリング前とする。 $Righthand$ は Kinect センサから取得された右手先の 3 次元座標データ、 $Rightshoulder$ は右肩の 3 次元座標データである。 $ArmLength$ は全被験者の腕長の最大値で、計測結果から、 $ArmLength=850.0$ mm とした。

続いて、ニューラルネットワークには、図 5 のような、3 層の階層型ニューラルネットワークを用いる。入力層のノード数は、式 (1) を用いて、現時刻 ($k=0$) から 7 サンプリング前 ($k=7$) までの計 8 サンプリング分の $input(k \times 3 + 1)$, $input(k \times 3 + 2)$, $input(k \times 3 + 3)$, 計 24 個とし、サンプリングタイム 50msec ごとに入力データを更新していくものとする。また、中間層のノード数を 40 個とする。さらに、出力層は -1.0 から 1.0 までの数値を出力し、操作者が手前から奥へ、奥から手前へ、左から右、右から左の方向にそれぞれ結果を算出するため、出力層のノード数を 4 個とする。入力層の出力 $I(i)$ は、入力データ $Input(i)$ をそのまま出力する (式 (2))。

$$I(i) = Input(i) \quad (2)$$

中間層の出力 H_j は式 (3) で表される。

$$h_j = f\left(\sum_{i=1}^{24} W_{ij} I(i)\right) \quad (3)$$

ここで、添字 j は中間層 j 番目の意味、 W_{ij} は入力層 i 番目-中間層 j 番目間の重み係数、 f は活性化関数である。活性化関数 f には、-1.0 から 1.0 の値を出力する式 (4) に示される非線形な写像を実現可能なシグモイド関数を使用する。

$$f(x) = \frac{1 - \exp(-2x)}{1 + \exp(-2x)} \quad (4)$$

出力層の出力 O_k は式 (5) で表される。

$$O_k = f\left(\sum_{j=1}^{40} W_{jk} H_j\right) \quad (5)$$

ここで、 W_{jk} は中間層 j 番目-出力層 k 番目間の重み係数である。

このとき、ニューラルネットワークへの入力データが、追従動作指示ならば出力が -1.0、フリック動作指示ならば 1.0 に近づけるよう、ニューラルネットワークの学習を行う。具体的には、式 (6) のような出力 O_k の教師データ (-1.0 か 1.0) に対する 2 乗誤差 E を考え、この E を逆伝播させることで各層間の重み係数を更新し、学習を繰り返す毎に E が小さくなるように学習を行う。

$$E_k = \frac{1}{2} \sum_{k=1}^4 (t_k - O_k)^2 \quad (6)$$

最急降下法により、各重み係数の更新量は式 (8) となる。

$$\Delta W'_{jk} = \alpha (O_k - t_k) O_k (1 - O_k) H_j \quad (7)$$

$$\Delta W_{ij} = \alpha ((O_k - t_k) O_k (1 - O_k) W'_j) H_j (1 - H_j) I(i) \quad (8)$$

ここで α は学習係数であり、 $\alpha = 0.001$ とする。なお α は、学習時間が短く、過学習を起こさない値を試行錯誤的に決定した。

出力された O_0 を O_{Right} , O_1 を O_{Left} , O_2 を O_{Front} , O_3 を O_{Back} とする。

Fig. 4 Coordinate system of the neural network learning

Fig. 5 Three layered neural network in which output layer has four nodes

以上の構成のニューラルネットワークを、2章の計測実験で得られた、 X , Y 軸正負の4方向で、各5回ずつ行った追従指示およびフリック指示のデータを用い、それぞれのデータに対して、学習サイクル数 5000 回で学習させた。これを被験者 21 名分繰り返した。ここで、学習に使用するデータは、右肩と右手先の座標データのうち、実際に手先が指示動作をしている区間を抜き出して使用する。具体的には、手先が動作を開始し、速度が 0 mm/s から上昇した地点から、手先が動作を終了し、速度が再び 0 mm/s に到達した地点までを学習区間とする。

3.2 指示動作における判別システムの学習結果

学習後のニューラルネットワークに対して、被験者 21 名に対し、4 方向にそれぞれ各 5 回計測したフリック指示、追従指示を入力し、その出力について検証する。右方向にフリック指示を行った試行データ 105 個に対して、その出力について 0.1 ごとに区切ったヒストグラムを図 6(a) に、右方向の追従指示を行った試行データ 105 個の結果を図 6(b) に示す。同様に、左方向にフリック指示を行ったものを図 7(a)、左方向に追従指示を行ったものを図 7(b)、前方向にフリック指示を行ったものを図 8(a)、前方向に追従指示を行ったものを図 8(b)、後方向にフリック指示を行ったものを図 9(a)、後方向に追従指示を行ったものを図 9(b) にそれぞれ示す。

図 6(a) の右方向、図 7(a) の左方向、図 8(a) の前方向、および図 9(a) の後ろ方向のフリック指示では、すべての出力が正に分布していた。このことから、ニューラルネットワークが正の出力をした場合にフリック動作である

と判断できる．次に，図 6(b) の右方向，図 7(b) の左方向，図 8(b) の前方向，および図 9(b) の左方向の追従指示では，すべての場合において負の値を出力している．このことから，ニューラルネットワークが負の値を出力した場合に追従指示であると判断できる．

4. ロボットシステムの構築

4.1 ロボットシステムの構築

第 3.1 章で構築した判別システムを利用し，操作者の指示がフリック指示であるか，それとも追従指示であるかが自動的に判別され，それぞれの指示方法に対するロボットの運動生成は，以下のように実現される．

4.1.1 追従指示モード

追従指示モードでは，ロボットは操作者の手先の動きに追従するように運動生成を行う．それは，基本的には手先の位置に比例定数を乗じたものをロボットの目標位置とし，それに高精度に追従すれば実現できる．ただし，人間の指示運動に遅れなく正確に追従することは，操作者にとってはあまり好ましくなく，適度な遅れや滑らかさを付加することにより人間心理に受け入れられるものとなることが明らかになっている（柴田他，2009）．

そこで，感性伝達関数（小壁他，2006）を導入する．感性伝達関数を施すことにより，適度な遅れと滑らかさが付加され，より操作者にとって心理的，感性的に受け入れられる動きとなることが期待される．ここでは，感性

伝達関数として次式の 2 次遅れ要素を採用する.

$$G_{Following}(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (9)$$

なお, 式中にある ω_n は固有周波数, ζ は減衰係数であり, 本研究ではロボットの動きを振動的にさせないため, $\zeta = 1.0$ とする. 図 10 に, 追従モードのブロック線図を示す.

Fig. 10 Block diagram of the following mode

Fig. 11 Block diagram of the flick mode

ここで, 本研究で使用するロボットの運動特性は, 以下に示す一次遅れ要素に近似されることが予備実験により分かっている.

$$G_R(s) = \frac{K_R}{1 + T_R s} \quad (10)$$

なお, 近似した結果, 各定数はそれぞれ $K_R = 1.0$, $T_R = 1.0$ であった. 図 10 で, 操作者の手先位置 $H(s)$ とロボットの実際の位置 $Y(s)$ との関係が式 (9) になるために, $H(s)$ とロボットの目標位置 $U(s)$ の間の伝達関数 $G_K(s)$ は, $G_{Following}(s)/G_R(s)$ となり, 目標位置 $U(s)$ は式 (11) で算出される.

$$\begin{aligned} U(s) &= G_K(s)H(s) \\ &= \frac{G_{Following}(s)}{G_R(s)}H(s) \end{aligned} \quad (11)$$

これにサンプリング時間 T のサンプラー, 零次ホルダーを付加し z 変換, 逆 z 変換を行うことで, 次式を得る. ただし, 式中の $H(k)$ は入力 (手先位置) であり, $U(k)$ は出力 (ロボットの目標位置) を表し, k は離散時間における現在時刻を表している.

$$\begin{aligned} U(k) &= 2e^{-\omega_n T} U(k-1) - e^{-2\omega_n T} U(k-2) \\ &\quad + \frac{1}{K_R} (1 - e^{-\omega_n T} + (T_R - \frac{1}{\omega_n}) T \omega_n^2 e^{-\omega_n T}) H(k-1) \\ &\quad - \frac{1}{K_R} (1 - e^{-\omega_n T} + (T_R - \frac{1}{\omega_n}) T \omega_n^2) e^{-\omega_n T} H(k-2) \end{aligned} \quad (12)$$

以上の手順により得られた $U(k)$ を, 追従指示でのロボットの目標値とする.

4.1.2 フリック指示モード

フリック指示モードでは, ロボットは手先の動きに対して弾かれたような挙動を示すことを目指し, 慣性を受けてある速度で速くまで進み, その後動きが収束するように停止する運動を考える. この動きを実現するために, 式 (13) に示される質量と粘性を持つ質点の運動系を利用した運動モデルによる目標位置生成を提案する. ただし, 式中の $F(t)$ は手先の速度に比例する仮想的な力であり, フリック指示における手先の速度 $V(t)$ に比例定数 K_f を乗ずることにより得られる. ここで, t は現在時刻を示している. また, M は仮想質量, C は仮想粘性で, ロボットの運動に影響を与えるパラメータである. 例えば, M を一定とした場合, C を小さくすることで, 粘性による減速が抑えられ, ロボットは速く, 速くまで移動し, C を大きくすることで, 粘性による減速が顕著となり, ゆっくりと, わずかな距離の移動となると思われる.

$$F(t) = K_f V(t) \quad (13)$$

$$M\ddot{U}(t) + C\dot{U}(t) = F(t)$$

図 11 に, フリックモードのブロック線図を示す. 図 11 において, $G_{Flick}(s)$ は式 (13) より

$$G_{Flick}(s) = \frac{K_f}{Ms^2 + Cs} \quad (14)$$

となる．4.1.1 節で述べたロボットの動特性 $G_R(s)$ を考慮すると，ロボットの目標値を算出する式が得られる．

$$\begin{aligned} U(s) &= \frac{G_{Flick}(s)}{G_R(s)} V(s) \\ &= \frac{K_f(1+T_R s)}{K_R(Ms^2 + Cs)} V(s) \end{aligned} \quad (15)$$

これを離散化すると，以下の差分方程式が得られる．

$$\begin{aligned} U(k) &= (1 + e^{-\frac{C}{M}T})U(k-1) - (e^{-\frac{C}{M}T})U(k-2) \\ &\quad + \frac{K_f M}{K_R C^2} \left(\frac{C}{M}T + \frac{T_R C - M}{M} - \frac{T_R C - M}{M} e^{-\frac{C}{M}T} \right) V(k-1) \\ &\quad - \frac{K_f M}{K_R C^2} \left(\frac{C}{M}T e^{-\frac{C}{M}T} + \frac{T_R C - M}{M} - \frac{T_R C - M}{M} e^{-\frac{C}{M}T} \right) V(k-2) \end{aligned} \quad (16)$$

なお， $V(k) = H(k) - H(k-1)$ である．式 (16) により算出された位置座標 $U(k)$ をフリック指示でのロボットの目標位置とする．

4.1.3 ニューラルネットワークを利用したモード切り替え

前章において学習したニューラルネットワークを利用して，追従指示モードとフリック指示モードが自動的に切り替わるシステムを構築する．

3.2 章の学習結果より，ニューラルネットワークの出力値が 0 未満の場合は追従指示モード，出力値が 0 以上の場合はフリック指示モードと判別する．また，ニューラルネットワークの出力値は，前述のように 50msec 毎に更新されており，手先が動作を開始してから正確な結果が出力するまでには，8 サンプル分の手先座標が必要となる．このことから，モード切り替えに使用するニューラルネットワークの出力値は，手先が移動し始めてからサンプル数 8 サンプル後，つまり 0.4 秒後の値を採用し，手先の動作およびロボットの動作が終わるまで，この指示モードのままロボットを操作する．

4.2 全方位移動ロボット

本研究では，操作対象のロボットとして，移動時に位置と姿勢方向を同時に変化させることができるホロノミック特性を有する全方位移動ロボットを用いる．これは，人間の任意の方向への指示に対応するためである．全方位移動ロボットの外観図を図 12 に示す．全方位移動ロボットの足回りは，車輪型移動機構で全方位移動を実現するべく，3 輪のオムニホイール (土佐電子社製) を用いる．オムニホイールは並進運動と回転運動を同時に行うことができ，並進運動で 2 自由度，回転運動で 1 自由度の合計 3 自由度の全方位移動が可能である．

ロボットの位置は，それぞれのモータに取り付けられたエンコーダからのパルス信号を，32 ビットカウンタボード (インタフェース社製，PCI-6201E) を介してコンピュータに取り込むことにより得られる．また，コンピュータで算出されたロボットの操作量を，12 ビット D/A 変換器 (インタフェース社，PCI-3523A)，サーボアンプ (マクソン社製，ADS 50/10) を介して DC サーボモータ (マクソン社製，RE-40) へ送り，位置制御を行う．

ロボット制御用コンピュータの OS には，リアルタイム性を持つ ART-Linux (Debian GNU/Linux) (ART-Linux) を用いる．ロボットは，実際に屋内で物などを載せて移動することにより支援を行うことを想定し，一般的な椅子などと同程度の高さ 690mm，直径 400mm とする．

また，ロボットの上部に，身体 of 3 次元座標を取得するためのセンサシステムである Kinect センサを取り付ける．Kinect センサを用いて全方位移動ロボットを動かすときの様子を図 13 に示す．ロボットに取り付けられた Kinect センサから得られた距離データを，Kinect センサ専用のドライバソフトウェアを介して Kinect センサ制御用のコンピュータに取り込む．そして，取り込まれたデータをもとに OpenNI ライブラリ (OpenNI) により各体の部位の 3 次元座標を算出し，その中から手先など必要な位置座標を，位置計測ソフトウェアに取り込む．位置計測ソフトウェアは，TCP/IP 通信を介してロボット制御用コンピュータに，手先位置座標をはじめロボットの操作に必要な情報を送信する．ロボット制御用コンピュータは，得られた情報をもとに目標軌道を算出し，それに基づいてロボットを位置制御する．

Fig. 12 Omni-directional mobile robot

Fig. 13 Experimental setup

5. ジェスチャ指示による全方位置移動の検証実験

構築したロボットシステムを利用して、実際にロボットシステムが、操作者の意図したとおりに動くかどうかを確認めるべく、以下に示す検証実験を行った。

実験の様子を図 13 に示す。実験は、操作者がロボットに対して、少し離れた場所にあるものを別の場所へ運ぶ場面を想定し、以下のように行った。ロボットは 1500mm 離れた位置に配置する。そして、被験者に Y 軸の負方向に追従指示およびフリック指示を行ってもらい、ロボットを Y 軸の負方向に移動させる。また、日常において障害物がある中でロボットを操作してもらうという状況を想定し、図 13 にあるようにロボットの到達目標エリアから操作者に向かって前方に障害物を設置した。このとき、フリック指示における仮想質量 $M=1.0$ 、仮想粘性係数 C は、0.1, 0.2, 0.3 および 0.4 を用いた。仮想質量に関しては予備実験により 1.0 という値が望ましいことを確認している。これらの仮想粘性係数について、被験者がフリック動作に対して抱くイメージとの関連について調べるため、アンケート形式の心理評価を行った。調査する項目は、「フリック指示で動いた距離はあなたのイメージに対して大きい／小さい」「フリック指示で動いた速さはあなたのイメージに対して速い／遅い」そして「今の動きは、フリック指示としてイメージ通りである／そう思わない」で、それぞれに対し 7 段階の尺度で評価してもらった。

5.1 実験結果

年齢 22～25 歳の成年男子 10 人に対して実験を行った。なお被験者には、2 章で計測し、ニューラルネットワークの学習対象である被験者以外も含まれている。

このとき、実験により計測されたロボットの軌跡について示したものの一例を、各粘性係数ごとに図 14(a) から図 14(d) に示す。なお、図中にある正方形の枠は目標エリアを示しており、白丸はロボットがフリック指示を認識して動き始めた地点であり、黒丸はロボットが追従指示をうけて動き始めた地点である。

図のロボットの軌跡から、被験者は最初フリック指示でロボットを大まかに動かした後、追従指示でロボットを手先に追従して動作させ、目標エリアに接近させていることが分かる。 C が 0.1 のとき、ロボットは一回のフリック指示で約 2.5m 移動している。このとき、目標地点からの行き過ぎが約 1m あり、その後、被験者は追従指示に移行するものの、目標地点の四角内にロボットを進入させることが出来ていない。 C が 0.2 および 0.3 のとき、一回のフリック動作で約 1.5m 移動させ、その後、追従指示で目標地点の四角内にロボットを進入させている。 C が 0.4 のとき、フリック指示によってロボットが動く距離が短くなっているため、図 14(d) のように 2 回フリック指示をしてロボットを動かしていることがわかる。なお、ほぼすべての被験者が、フリック指示と追従指示の自動的な切り替えに成功しており、学習後のニューラルネットワークは、学習した対象以外の被験者に対しても有効に判別結果を出力できるといえる。

また、これらの動きに対して評定尺度法に基づく心理評価を行った。心理評価は、フリック動作を行った時のロボットの移動距離、速さ、イメージに基づくものであるかどうかについて 7 段階評価で実施した。その時の結果

Fig. 14 Results of the proposed system

を図 15 に示す。ただし、図では両極には相反する形容詞対が配置され、7段階のそれぞれに 1 ずつの得点を与えたときの平均と分散が示されている。なお、アンケート結果に対して、Tukey-Kramer 検定 (有意水準 5%) を行っ

Fig. 15 Result of the psychological evaluation

ている。

図 15 の距離に関する項目では、 C が 0.1 に対して 0.3 および 0.4 に有意差が存在した。このことから、 C を変化させることで、距離に関する印象を変化させられることが分かった。 $C=0.2, 0.3$ において、評価が中央付近に位置しており、被験者は距離が大きすぎず、小さすぎないと感じていることが分かる。 $C=0.1$ のとき、被験者は $C=0.3$ より「距離が大きい」と感じていることが分かる。また、速度に関する項目については、 C が 0.1 に対して 0.3 および 0.4 に有意差が存在した。 $C=0.2, 0.3$ において、評価が中央付近に位置しており、フリック指示に対する速度がちょうど良いものであると評価しているといえる。 $C=0.1$ のとき、被験者は指示に対して動きが速く、 $C=0.4$ のとき、動きが遅いと感じていることが分かる。全体的な印象については、 $C=0.2, 0.3, 0.4$ のとき、評価の平均値が中央よりポジティブよりで、イメージ通りの動きであると評価している。 $C=0.1$ については、有意差はないものの、評価がネガティブよりになっている。

これらのことから、被験者のイメージ通りの距離や速度を生成できるパラメータとして、 $C=0.2, 0.3$ が候補としてあげられる。

6. 結 言

本研究では、ジェスチャ指示として物体を弾く動作でロボットを素早く移動できるフリック指示をまず提案した。そしてフリック指示と、手先の動きに追従させる形でロボットを操作する筆者らがすでに提案した追従指示の両者を任意に利用できるシステムを提案した。それら 2 種類の指示方法を動的に判別するためにニューラルネットワークを採用し、それぞれの指示動作に対する学習を行った。そして、学習したニューラルネットワークを利用して指示動作を判別する検証実験を行い、その有効性を確認した。

次に、学習に使用したニューラルネットワークをロボットシステムに導入した。このとき、フリック指示は、人間がイメージするロボット運動であること、およびロボットを素早く動かすことができるという観点から、仮想的な質量と粘性を持つ質点の運動モデルによるロボット運動を提案した。また追従指示では、操作者の手先の動きに対して感性伝達関数を施したロボット動作を構築しシステムに導入した。そして、実際にそのシステムを用いてロボットを初期位置から目標位置へと移動させる指示動作実験を被験者により行った。その結果、システムは 2 種類の指示を正確に判別する事ができ、操作者の意図するロボット運動を実現できた。さらに、フリック動作、追従動作を組み合わせることにより、効果的にロボットを目標位置へと導くことが可能であることを確認した。また、心理評価を行った結果、被験者がちょうど良いと感じる距離や速度を生成できる仮想粘性の値が存在することが分かった。

本研究では、スマートフォンやタブレット端末などでフリック指示を行ったことのある被験者を対象に実験を行った。これらの被験者に対して、提案した手法、すなわち空中で物体を弾くようなフリック動作が、デバイスの画面を弾く動作と比較してどのような印象を与えるか、また、これらのデバイスを使用したことのないユーザーに対して、本システムが有効かどうかを検証することは興味深い課題である。これらのことは、今後の課題として明らかにしていきたい。また、個人差や、ユーザーの習熟度合いによる動作の変化に対応するため、ニューラルネットワークを随時チューニングしていく仕組みを導入していきたい。さらに、本システムをドローンなどへの操作に用い、長距離を素早く移動させることに対する検討を行っていく予定である。

文 献

- ART-Linux, available from <http://www.dh.aist.go.jp/jp/research/assist/ART-Linux/> (参照日 2015 年 1 月 18 日).
- 浅野秀胤, 織茂達也, 高橋真人, 寺林賢司, 太田睦, 梅田和昇, フーリエ変換を用いた小さな手振りの検出, ViEW2010 ビジョン技術の実利用ワークショップ講演論文集 (2010), pp.264–267.
- Flash, T. and Hogan, N., The coordination of arm movements: an experimentally confirmed mathematical model, The Journal of Neuroscience, Vol.5, No.7(1985), pp.1688–1703.
- 長谷部尚志, 芦澤寛樹, 山本昇志, ジェスチャ認識を備えた直感的な画像インタフェース, 社団法人映像情報メディア学会技術報告 ITE Technical Report Vol.34, No.10(2010), pp.29–32.
- 細田祐司, 人間共生ロボットの開発への取り組み, 日本ロボット学会誌, Vol.24, No.3(2006), pp.296–299.

- 小壁正義, 山本智規, 柴田論, 感性伝達関数を用いた腕型ロボットから人間への手渡し運動, 感性工学研究論文集 Vol.6, No.2(2006), pp.79–86.
- 河内まき子, 持丸正明, 岩澤洋, 三谷誠二, 日本人人体寸法データベース 1997-98, 通商産業省工業技術院くらしと JIS センター (2000).
- 松日楽信人, 生活を変えるロボットを目指して, 日本ロボット学会誌, Vol.24, No.3(2006), pp.292–295.
- Microsoft, Kinect for Windows, available from <http://www.microsoft.com/en-us/kinectforwindows/> (参照日 2015 年 1 月 18 日).
- 小俣透, 人と接するロボットの研究を発展させるために, 日本ロボット学会誌, Vol.29, No.3(2011), p.224.
- 大西清文, 神代充, 柴田論, 山本智規, 大西敏也, 音声とジェスチャ指示に基づく人に優しいヒューマン・ロボット・インタフェースに関する研究, 福祉工学シンポジウム講演論文集 2006(2006), pp.193–196.
- OpenNI, Open-source SDK for 3D sensors, available from <https://github.com/OpenNI/OpenNI> (参照日 2015 年 1 月 18 日).
- 柴田論, 山本智規, 神代充, 指先指示運動に協調するロボットシステムに関する考察, 日本福祉工学会誌, Vol.11, No.2 (2009), pp.30–36.
- 鈴木敬弘, 大矢晃久, 油田信一, 手招きに応じる移動ロボット, 日本機械学会ロボティクス・メカトロニクス講演会'03 講演論文集 (2003), 1P1-1F-A7(1)–(2).
- 谷和男, 平尾江利, 加藤佳男, 川村拓也, 手振り動作によるロボットの運動制御, 日本機械学会ロボティクス・メカトロニクス講演会'03 講演論文集 (2003), 2AI-1F-E5(1)–(2).

References

- ART-Linux, available from <http://www.dh.aist.go.jp/jp/research/assist/ART-Linux/>, (accessed on 18 January, 2015).
- Asano, H., Orimo, T., Takahashi, M., Terabayashi, K., Oota, M. and Umeda, K., Detection of small hand-waving movements using discrete fourier transform, Proceedings of ViEW2010 (2010), pp.264–267 (in Japanese).
- Flash, T. and Hogan, N., The coordination of arm movements: an experimentally confirmed mathematical model, The Journal of Neuroscience, Vol.5, No.7(1985), pp.1688–1703.
- Hasebe, H., Ashizawa, H. and Yamamoto, S., Projector-camera interface with gesture recognition, The Institute of Image Information and Television Engineers, ITE Technical Report Vol.34, No.10(2010), pp.29–32 (in Japanese).
- Hosoda, Y., An approach to development of a human-symbiotic robot, Journal of Robotics Society of Japan, Vol.24, No.3(2006), pp.296–299 (in Japanese).
- Kokabe, M., Yamamoto, T. and Shibata, S., Design of acceptable handing motion of arm-Robots utilizing the “KANSEI” transfer function, Journal of Japan Society of Kansei Engineering, Vol.6, No.2(2006), pp.79–86 (in Japanese).
- Kouchi, M., Mochimaru, M., Iwasawa, H. and Mitani, S., Anthropometric database for Japanese population 1997–98, Japanese Industrial Standards Center (AIST, MITI)(2000)(in Japanese).
- Matsuhira, N., Towards the robots to change our life style, Journal of Robotics Society of Japan, Vol.24, No.3(2006), pp.292–295 (in Japanese).
- Microsoft, Kinect for Windows, available from <http://www.microsoft.com/en-us/kinectforwindows/>, (accessed on 18 January, 2015).
- Omata, T., For the development of robotics research involving humans, Journal of Robotics Society of Japan, Vol.29, No.3(2011), p.224 (in Japanese).
- Ohnishi, K., Jindai, M., Shibata, S., Yamamoto, T. and Ohnishi, T., Research on a gentle human-robot interface based on instructions by voice and gesture, Proceedings of the Welfare Engineering Symposium 2006 (2006), pp.193–196 (in Japanese).
- OpenNI, Open-source SDK for 3D sensors, available from <https://github.com/OpenNI/OpenNI>, (accessed on 18 January, 2015).
- Shibata, S., Yamamoto, T. and Jindai, M., A study on robot system cooperating with human instruction using fingertip movement, Journal of The Japan Society for Welfare Engineering, Vol.11, No.2(2009), pp.30–36 (in Japanese).

Suzuki, T., Ohya, A. and Yuta, S., Mobile robot replying to human's beckoning action, Proceedings 2003 JSME Conference on Robotics and Mechatronics (2003), 1P1-1F-A7(1)-(2) (in Japanese).

Tani, K., Hirao, E., Kato, Y. and Kawamura, T., Control of robot motion by hand waving, Proceedings 2003 JSME Conference on Robotics and Mechatronics (2003), 2AI-1F-E5(1)-(2) (in Japanese).