(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2008-254074
(P2008-254074A)
(43)公開日　平成20年10月23日(2008.10.23)

(51)Int.Cl.		F I		テーマコード（参考）
B25J 9/22	(2006.01)	B25J 9/22	A	3C007
G05B 19/42	(2006.01)	G05B 19/42	J	3C269
G06T 15/70	(2006.01)	G05B 19/42	P	5B050
		G06T 15/70	B	

審査請求 未請求　請求項の数 10　ＯＬ　（全 29 頁）

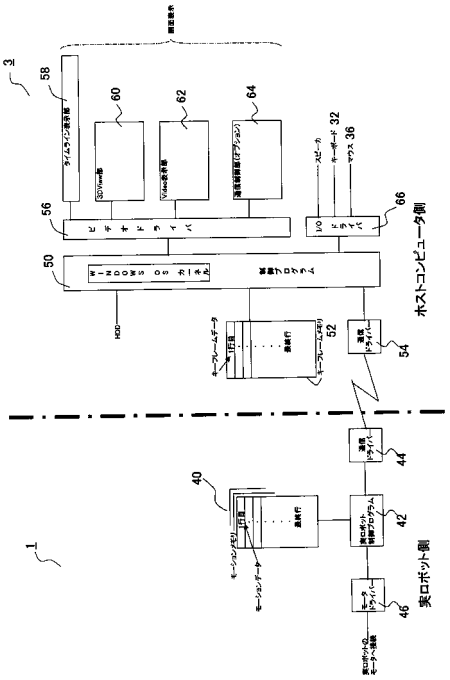
(21)出願番号　特願2007-95291(P2007-95291)
(22)出願日　平成19年3月30日(2007.3.30)

(71)出願人　000132471
株式会社セガ
東京都大田区羽田１丁目２番１２号
(74)代理人　100094514
弁理士　林　恒徳
(74)代理人　100094525
弁理士　土井　健二
(72)発明者　星野　建二
東京都大田区羽田１丁目２番１２号　株式会社セガ内
(72)発明者　肥谷　宇人
東京都大田区羽田１丁目２番１２号　株式会社セガ内

最終頁に続く

(54)【発明の名称】ロボットのモーション編集装置及びそのプログラム

(57)【要約】
【課題】多関節型ロボットのモーションデータを編集する編集装置に関し、作成したモーションデータを再利用して、新たなモーションデータを作成する。
【解決手段】編集装置（３）は、操作ユニット（３６）の操作に応じて、３次元擬似空間（６０）でのロボットのモデルの各部位の位置、傾きを変更して、操作ユニット（３６）のデータ作成指示に応じて、変更されたロボットのモデルの各部位の位置、傾きから、各関節の角度を計算し、キーフレームデータを作成し、３次元表示のための擬似空間内におけるロボットの状態データを付して、多関節型ロボット（１）に送信する。３次元擬似空間で作成されたデータを再利用して、他のモーションデータを３次元擬似空間で作成できる。
【選択図】図２



【特許請求の範囲】
【請求項１】
実空間で関節を駆動する多関節型ロボットのモーションデータを編集するモーション編集装置において、
擬似空間内における前記ロボットのモデルを３次元表示する３次元表示部を有する表示ユニットと、
前記表示ユニットで表示された前記ロボットのモデルの各部位の位置、傾きを指示するための操作ユニットと、
前記操作ユニットの操作に応じて、前記ロボットのモデルの各部位の位置、傾きを変更して、前記３次元表示部に表示するとともに、前記操作ユニットのデータ作成指示に応じて、前記変更されたロボットのモデルの各部位の位置、傾きから、各関節の角度を計算し、キーフレームデータを作成する処理ユニットとを有し、
前記処理ユニットは、前記キーフレームデータに前記３次元表示のための前記擬似空間内におけるロボットの状態データを付して、前記多関節型ロボットに送信する
ことを特徴とするモーション編集装置。
【請求項２】
前記処理ユニットは、前記操作ユニットで指示された前記ロボットのモデルの部位を、前記３次元表示空間の基準面に接するように、前記ロボットのモデルの座標を変換し、前記キーフレームデータを作成する
ことを特徴とする請求項１のモーション編集装置。
【請求項３】
前記処理ユニットは、前記操作ユニットの指示により、前記作成されたキーフレームデータから、前記ロボットのモデルの部位間の衝突の有無を判定し、前記３次元表示空間の前記ロボットのモデルの前記衝突した部位の表示色を変更する
ことを特徴とする請求項１のモーション編集装置。
【請求項４】
前記処理ユニットは、前記表示ユニットに、前記操作ユニットで操作し、前記ロボットのモデルの前記キーフレーム間の時間を指定するためのタイムライン表示部を表示し、前記処理ユニットは、前記タイムライン表示部で指定された時間により、前記キーフレームデータの時間情報を作成する
ことを特徴とする請求項１のモーション編集装置。
【請求項５】
前記処理ユニットは、前記キーフレームデータを前記多関節型ロボットに送信するとともに、送信処理の遅延を検出し、前記キーフレームデータの時間情報を変更して、次のキーフレームデータを送信する
ことを特徴とする請求項１のモーション編集装置。
【請求項６】
前記処理ユニットは、ビデオデータに含まれる再生時間情報から、前記タイムライン表示部で指定された時間と同期させたビデオ画像を表示する
ことを特徴とする請求項４のモーション編集装置。
【請求項７】
前記処理ユニットは、ビデオデータに含まれる再生時間情報から、前記タイムライン表示部で指定された時間と同期させ、前記表示ユニットに、ビデオ画像を表示し、且つ前記多関節型ロボットに前記作成したキーフレームデータを送信し、前記ビデオ画像と前記多関節型ロボットの動作を同期する
ことを特徴とする請求項４のモーション編集装置。
【請求項８】
前記処理ユニットは、前記多関節型ロボットのメモリに記憶されたモーションデータを読み込み、前記キーフレームデータの形式に変更し、前記表示ユニットの前記ロボットのモデルのポーズを表示し、

前記操作ユニットの操作に応じて、表示された前記ロボットのモデルの各部位の位置、傾きを変更して、前記操作ユニットのデータ作成指示に応じて、前記変更されたロボットのモデルの各部位の位置、傾きから、各関節の角度を計算し、キーフレームデータを作成する

ことを特徴とする請求項１のモーション編集装置。

【請求項９】

前記処理ユニットは、前記モーションデータに含まれる制御コードを解析し、前記キーフレームデータに変換する

ことを特徴とする請求項８のモーション編集装置。

【請求項１０】

実空間で関節を駆動する多関節型ロボットのモーションデータを編集するためのプログラムにおいて、

擬似空間内における前記ロボットのモデルを表示ユニットに３次元表示するステップと、

前記表示ユニットで表示された前記ロボットのモデルの各部位を位置、傾きを指示するための操作ユニットの操作に応じて、前記ロボットのモデルの各部位の位置、傾きを変更して、前記３次元表示部に表示するステップと、

前記操作ユニットのデータ作成指示に応じて、前記変更されたロボットのモデルの各部位の位置、傾きから、各関節の角度を計算し、キーフレームデータを作成し、前記キーフレームデータに前記３次元表示のための前記擬似空間内におけるロボットの状態データを付して、前記多関節型ロボットのためのモーションデータを作成するステップとを、コンピュータに実行させるプログラム。

【発明の詳細な説明】

【技術分野】

【０００１】

本発明は、多関節型ロボットの動作パターンを規定するロボットのモーション編集処理を行うロボットのモーション編集装置及びそのプログラムに関し、特に、多数の関節を有する多関節型ロボットの動作パターンを、擬似空間でのＣＧモデルの操作により作成するロボットのモーション編集装置及びそのプログラムに関する。

【背景技術】

【０００２】

近年のロボット技術の発展により、歩行可能なロボットが出現している。このようなロボットでは、一般に多関節型ロボットが利用されており、関節数が多い程、多彩な動作（モーション）ができる。又、産業界のみならず、一般家庭や日常生活にも、ロボットが浸透している。

【０００３】

関節数が多いロボットでは、多種多彩の動作（モーション）が可能であるが、そのモーションを規定する位置や速度（関節角や角速度）を入力し、モーションデータを作成する必要がある。

【０００４】

この方法として、従来、画面の擬似空間に、ロボットのＣＧ（Ｃｏｍｐｕｔｅｒ Ｇｒａｐｈｉｃｓ）モデルを表示し、マウス等の入力により、ＣＧモデルを操作し、モーションデータを編集するシステムが提案されている（例えば、特許文献１，２参照）。

【０００５】

この従来技術では、３次元擬似空間に、ロボットのＣＧモデルを表示し、ロボットに、所望のモーションを行わせるため、マウスやキー操作で、ＣＧモデルを、所望のポーズに動作させる。そして、そのポーズの位置から、実ロボットの関節角データを得て、モーションデータを編集するものであった。

【０００６】

更に、このモーションデータを再生して、擬似空間のＣＧモデルを動作させ、モーショ

ンデータによって、ロボットが所望のモーションを実行するかを確認する。確認後は、モーションデータは、ロボットに転送され、ロボットのメモリに格納される。
【特許文献１】特開２００２－１０３２５８号公報（図２５、図２６、段落０３２３～段落０３３２）
【特許文献２】特開２００５－１２５４６０号公報（段落０１２７～段落０１３２、図５、図１２）
【発明の開示】
【発明が解決しようとする課題】
【０００７】
近年のロボットのエンターティメント分野での利用に伴い、ロボットに多種、多様の操作形態や、多種、多様の動作を行わせたいとの要望がある。特に、個人がロボットを所有した場合には、その個人の趣向や使用方法により、予め決められた動作以外の動作を、自身で、プログラムしたいとの要望がある。
【０００８】
このためには、あるモーションデータを格納したロボットは、そのモーションデータに従う動作しかできないが、他の動作を所望する場合には、前述の擬似空間で、ＣＧモデルを操作して、新たにモーションデータを作成する必要があった。
【０００９】
又、モーションデータは、複数のポーズからなり、このポーズ間の変化の時間を設定することにより、ポーズを連続するモーションを実現できる。しかしながら、従来技術では、ポーズの作成は可能であるが、この時間の設定が困難であった。
【００１０】
更に、作成したモーションデータで、実ロボットを、擬似空間のＣＧモデルの動作や表示画像と同期して、動作させたいとの要望もある。従来技術では、このような使用形態には、対応するのが、困難である。
【００１１】
従って、本発明の目的は、ロボットの多種のモーションデータを、より容易に作成するためのロボットのモーション編集装置及びそのプログラムを提供することにある。
【００１２】
又、本発明の他の目的は、作成したモーションデータで、実ロボットを、擬似空間のＣＧモデルの動作や表示画像と同期して、動作するためのロボットのモーション編集装置及びそのプログラムを提供することにある。
【００１３】
更に、本発明の他の目的は、既存のロボットのモーションデータを利用して、他のモーションのモーションデータを作成するためのロボットのモーション編集装置及びそのプログラムを提供することにある。
【００１４】
更に、本発明の他の目的は、処理装置の能力に関わらず、作成したモーションデータで、実ロボットを、擬似空間のＣＧモデルの動作や表示画像と同期して、動作するためのロボットのモーション編集装置及びそのプログラムを提供することにある。
【課題を解決するための手段】
【００１５】
この目的の達成のため、本発明のモーション編集装置は、実空間で関節を駆動する多関節型ロボットのモーションデータを編集するモーション編集装置において、擬似空間内における前記ロボットのモデルを３次元表示する３次元表示部を有する表示ユニットと、前記表示ユニットで表示された前記ロボットのモデルの各部位の位置、傾きを指示するための操作ユニットと、前記操作ユニットの操作に応じて、前記ロボットのモデルの各部位の位置、傾きを変更して、前記３次元表示部に表示するとともに、前記操作ユニットのデータ作成指示に応じて、前記変更されたロボットのモデルの各部位を位置、傾きから、各関節の角度を計算し、キーフレームデータを作成する処理ユニットとを有し、前記処理ユニ

ットは、前記キーフレームデータに前記３次元表示のための前記擬似空間内におけるロボットの状態データを付して、前記多関節型ロボットに送信する。

【００１６】

又、本発明のプログラムは、実空間で関節を駆動する多関節型ロボットのモーションデータを編集するためのプログラムにおいて、擬似空間内における前記ロボットのモデルを表示ユニットに３次元表示するステップと、前記表示ユニットで表示された前記ロボットのモデルの各部位の位置、傾きを指示するための操作ユニットの操作に応じて、前記ロボットのモデルの各部位の位置、傾きを変更して、前記３次元表示部に表示するステップと、前記操作ユニットのデータ作成指示に応じて、前記変更されたロボットのモデルの各部位の位置、傾きから、各関節の角度を計算し、キーフレームデータを作成し、前記キーフレームデータに前記３次元表示のための前記擬似空間内におけるロボットの状態データを付して、前記多関節型ロボットのためのモーションデータを作成するステップとを、コンピュータに実行させる。

【００１７】

又、本発明では、好ましくは、前記処理ユニットは、前記操作ユニットで指示された前記ロボットのモデルの部位を、前記３次元表示空間の基準面に接するように、前記ロボットのモデルの座標を変換し、前記キーフレームデータを作成する。

【００１８】

又、本発明では、好ましくは、前記処理ユニットは、前記操作ユニットの指示により、前記作成されたキーフレームデータから、前記ロボットのモデルの部位間の衝突の有無を判定し、前記３次元表示空間の前記ロボットのモデルの前記衝突した部位の表示色を変更する。

【００１９】

又、本発明では、好ましくは、前記処理ユニットは、前記表示ユニットに、前記操作ユニットで操作し、前記ロボットのモデルの前記キーフレーム間の時間を指定するためのタイムライン表示部を表示し、前記処理ユニットは、前記タイムライン表示部で指定された時間により、前記キーフレームデータの時間情報を作成する。

【００２０】

又、本発明では、好ましくは、前記処理ユニットは、前記キーフレームデータを前記多関節型ロボットに送信するとともに、送信処理の遅延を検出し、前記キーフレームデータの時間情報を変更して、次のキーフレームデータを送信する。

【００２１】

又、本発明では、好ましくは、前記処理ユニットは、ビデオデータに含まれる再生時間情報から、前記タイムライン表示部で指定された時間と同期させたビデオ画像を表示する。

【００２２】

又、本発明では、好ましくは、前記処理ユニットは、ビデオデータに含まれる再生時間情報から、前記タイムライン表示部で指定された時間と同期させて、前記表示ユニットに、ビデオ画像を表示し、且つ前記多関節型ロボットに前記作成したキーフレームデータを送信し、前記ビデオ画像と前記多関節型ロボットの動作を同期する。

【００２３】

又、本発明では、好ましくは、前記処理ユニットは、前記多関節型ロボットのメモリに記憶されたモーションデータを読み込み、前記キーフレームデータの形式に変更し、前記表示ユニットの前記ロボットのモデルのポーズを表示し、前記操作ユニットの操作に応じて、表示された前記ロボットのモデルの各部位を位置、傾きを変更して、前記操作ユニットのデータ作成指示に応じて、前記変更されたロボットのモデルの各部位を位置、傾きから、各関節の角度を計算し、キーフレームデータを作成する。

【００２４】

又、本発明では、好ましくは、前記処理ユニットは、前記モーションデータに含まれる制御コードを解析し、前記キーフレームデータに変換する。

【発明の効果】
【００２５】
操作ユニットの操作に応じて、３次元擬似空間でのロボットのモデルの各部位を位置、傾きを変更して、操作ユニットのデータ作成指示に応じて、変更されたロボットのモデルの各部位を位置、傾きから、各関節の角度を計算し、キーフレームデータを作成し、３次元表示のための擬似空間内におけるロボットの状態データを付して、多関節型ロボットに送信するので、３次元擬似空間で作成されたデータを再利用して、他のモーションデータを３次元擬似空間で作成できる。
【発明を実施するための最良の形態】
【００２６】
以下、本発明の実施の形態を、ロボットのモーション編集装置、モーション編集処理機能、モーション編集処理、ロボットとの通信処理、ロボットとの同期処理、他の実施の形態の順で説明するが、本発明は、この実施の形態に限られない。
【００２７】
（ロボットのモーション編集装置）
図１は、本発明のロボットのモーション編集装置の一実施の形態の構成図、図２は、図１の構成のブロック図であり、ロボットとして、人型の多関節ロボットを例に示す。
【００２８】
図１に示すように、ロボット１は、胴体１０と、顔２８と、４関節の右腕１２，１３－１，１５，１７と、右手１９と、４関節の左腕１１，１３－２，１４，１６と、左手１８、４関節の右足２１，２３，２５，２７と、４関節の左足２０，２２，２４，２６とを有する。
【００２９】
顔２８は、関節軸ｒ５を中心に、胴体１０に対し、回転できる。右腕は、関節１２、右上腕１３－１、関節１５、右下腕１７、右手１９を持ち、４つの関節軸ｒ１～ｒ４を中心に、回転する。同様に、左腕は、関節１１、左上腕１３－２、関節１４、左下腕１６、左手１８を持ち、４つの関節軸ｒ６～ｒ９を中心に、回転する。
【００３０】
右足は、関節２１、右大腿２３、右下腿２５、右足部２７を持ち、４つの関節軸ｒ１０、ｒ１２～ｒ１４を中心に、回転する。左足は、関節２０、左大腿２２、左下腿２４、左足部２６を持ち、４つの関節軸ｒ１１、ｒ１５～ｒ１７を中心に、回転する。従って、このロボット１は、胴体１０に対し、１７関節の自由度がある人型ロボットである。
【００３１】
一方、モーション編集装置３は、パーソナルコンピュータで構成されており、ロボット１と、図示しないケーブル、無線で通信する。この装置３は、ＣＰＵ，メモリ、ハードディスクドライブを搭載したベース３０と、ベース３０に設けられたキーボード３２と、ベース３０に折りたたみ可能な表示部３４と、外部マウス３６とを有する。
【００３２】
この装置３に、図２以下で説明するモーション編集プログラムをロードすることにより、モーション編集装置として、動作する。図２に示すように、モーション編集プログラム５０は、カーネル（例えば、WINDOWS(登録商標)）の制御の元に動作する表示制御を行うとともに、ロボット通信制御処理を行う。
【００３３】
表示制御処理は、Ｉ／Ｏドライバ６６を介するキーボード３２、マウス３６からの入力により、表示部３４の表示を制御するとともに、モーションデータを編集する。ロボット通信制御処理は、キーフレームメモリ５２、通信ドライバ５４を使用して、ロボット１との通信制御を行う。
【００３４】
ロボット１は、モーションデータを格納するモーションメモリ４０と、編集装置１と通信するための通信ドライバ４４と、ロボット１の各関節のモータを駆動するモータドライ

バ４６と、ＣＰＵを有する。ＣＰＵは、実ロボット制御プログラム４２を実行し、ロボット１の動作を制御する。

【００３５】

表示部３４は、タイムラインウインドウ（以下、タイムライン表示部）５８、３ＤＶｉｅｗウインドウ（以下、３ＤＶｉｅｗ表示部）６０、オプションウインドウ（以下、オプション表示部）６４、Ｖｉｄｅｏウインドウ（以下、Video表示部）６２からなる。これらの表示部は、プログラム５０の制御により、ビデオドライバ５６を介し、駆動される。

【００３６】

図３は、表示部３４の表示画面、図４は、表示部３４のタイムライン表示部の説明図を示す。タイムライン表示部５８は、ウインドウズ（登録商標）のプルダウンメニュー（ファイル、表示等）と、２つのタイムラインとから構成され、上部を広域タイムライン、下部を作業用タイムラインと言う。

【００３７】

図４（Ｂ）の拡大図に示すように、図４（Ａ）の上部の広域タイムライン５８－１は、横軸に時間軸スケール（例えば、図では０秒から１８，０００秒付近）を表示し、その上に、例えば、黄色のバーで、後述の作業用タイムライン５８－２の幅（時間）を示す。また、後述の作業用タイムライン５８－２の再生範囲を、広域タイムライン５８－１の下に、小さな横バーで示す。

【００３８】

図４（Ｃ）の拡大図に示すように、作業用タイムライン５８－２は、横軸に、時間軸スケール（例えば、図では０秒から２，０１０秒付近まで）を表示し、時間軸スケールの１つの枠は、１５ミリ秒刻みに設定している。この１５ミリ秒時間単位で、ポーズを作成する事が出来る。この時間軸に、作業用タイムライン５８－２の再生範囲を、例えば、緑色でバー表示する。

【００３９】

又、広域タイムライン５８－１、作業用タイムライン５８－２の両方の時間軸スケールに、太い縦のマーク（例えば、赤色のマーク）で、キーフレームが設定されている事を表示する。

【００４０】

作業用タイムライン５８－２の時間軸スケールの下には、例えば、オレンジマークで、現在の作業点を示し、この時点（フレーム）のロボットのポーズは、図３の３ＤＶｉｅｗ表示部６０に表示される。同様に、図３の右部のＶｉｄｅｏ表示部６２には、この時点での、動画映像が映し出される。

【００４１】

更に、図４（Ｃ）に示すように、再生制御ボタンが、再生範囲を示すバーの上部に表示され、左から「頭だし」、「再生・停止」、「最後部へ」の３つのボタンからなる。

【００４２】

次に、図５により、３ＤＶｉｅｗ表示部６０を説明する。図５に示すように、３ＤＶｉｅｗ表示部６０は、ロボットの形状が直感的に分るように、ロボットを３次元的に表示する。３面図的に、正面、側面、上面の方向から見ることも出来るし、カメラの向きを切り替えることも出来る。又、３ＤＶｉｅｗ表示部６０は、同時に複数の表示を行う事が出来る。

【００４３】

３ＤＶｉｅｗ表示部６０の下段には、以下のボタンが設けられている。即ち、左から、表示されたロボットの現ポーズを、キーフレームとして登録するキーフレームキー６０－１、ロボットのポーズを、インバースキネマティクスにて、動かすインバースキネマティクス（ＩＫ）キー６０－２、ロボットのポーズを、フォワードキネマティク（ＦＫ）にて動かすフォワードキネマテイクスキー６０－３、マウスの動作を、ポーズ付けモードか、モーション付けモードかを切り替えるポーズモード/モーションモード切替ボタン６０－４、過去の動作を取消すアンドゥボタン６０－５、アンドゥで取消した動作を再度設定す

るリドゥボタン６０－６を設けている。

【００４４】

更に、その右に、画面を俯瞰で見た３Ｄ（３次元）表示とするカメラボタン６０－７、画面を正面から見た表示状態にする（表裏を順次切り替える）正面ボタン６０－８、画面を側面から見た表示状態にする（左右を順次切り替える）側面ボタン６０－９、画面を上面から見た表示状態にする（上下を順次切り替える）上面ボタン６０－１０、ロボットを初期姿勢に設定するホームポジションボタン６０－１１、ロボットを垂直移動し、地面へ着ける（接地）ボタン６０－１２、ロボットの重心点を地面上に表示するボタン６０－１３、前後のキーフレームの画像を重ねて半透明に表示するゴースト表示ボタン６０－１４、地面の表示をON/OFFするボタン６０－１５とを設けている。

【００４５】

これらのボタンを、マウス３６の操作で、クリックすることにより、前述のロボットに対する表示処理を行う。

【００４６】

また、ロボットの手足各部には、「ここから先を対称にする」、「この部分を固定させる」や、足先には、それ以外に「足を着地させる」等の機能を指定するプルダウンメニュー６０－１６を表示する。

【００４７】

次に、図３のＶｉｄｅｏ表示部６２は、ロボット動作と同期して再生する（場合の）映像を表示する。この映像は、ロボット動作再生中でも、ロボット動作作成中でも、タイムライン５８－２上のオレンジ色のマークの現在時刻（カレントフレーム）に同期し、その時刻の映像を再現する。画面サイズは、マウス３６で任意の大きさに出来、フルスクリーン表示も可能である。

【００４８】

図６は、ロボット通信制御機能の説明図である。ロボット通信制御機能は、実際のロボット１と通信を行い、実動作等を行うための制御機能であり、図６のような表示で確認できる。通信制御の為の表示は、オプション表示部６４中のロボット通信タブ６４－１のクリックにより実行される。

【００４９】

ロボット通信タブ６４－１にチェックを入れると、キーフレームメモリ５２から、各モータの角度データが、通信制御（ドライバ）５４を介して、実際のロボット１に送信され、３ＤＶｉｅｗ６０上のロボットの動作と，実際のロボット１の動作は同期する。実際には、これに先立ち、通信ポート６４－２を、実際のロボット１の接続されているポートを合わせる。

【００５０】

また、ロボット１は、モーション用メモリ４０（図２参照）を有しており、ロボット１独自のモーションデータを持っている。図６では、そのメモリのモーションデータＭ１からＭ１０を表示している。たとえば、Ｍ１のモーションデータを、ロボット１から読み込むために、図６のＭ１をクリックした後、モーションモードキー６４－３の読み込みボタンをクリックする。これにより、ロボット１からモーションデータを読み込み、キーフレームメモリ５２に格納し、図４（Ａ）のタイムライン５８上に、モーションを展開する。

【００５１】

同様に、タイムライン５８上のモーション（図２のキーフレームメモリ５２のモーションデータ）を、ロボット１のモーション用メモリ４０へ送る場合には、モーションモードキー６４－３の書き込みボタンをクリックする。

【００５２】

更に、ロボット１の内蔵するモーション用メモリ４０の内容に従って、そのモーションを実行（再生）することを指示する場合には、たとえば、Ｍ１のモーションデータ枠をクリックした後、モーションモードキー６４－３の選択モーション再生ボタンを押す。これにより、ロボット１は、モーションデータＭ１で、実動作をする。

【００５３】
教示機能モードキー６４－４は、ロボット１からポーズのデータ（関節角）を取得するためのキーである。
【００５４】
（モーション編集処理機能）
図７は、ロボット編集処理（操作）と、キー、マウス操作の関係図、図８乃至図９は、その衝突判定処理の説明図、図１０乃至図１５は、その接地処理の説明図、図１６乃至図１８は、その部分位置合わせ処理の説明図である。
【００５５】
図７により、基本操作を説明する。図４（Ｂ）の広域タイムライン表示部５８－１の作業用タイムライン（黄色表示）の選択位置を変更するには、マウス３６により、ポインタを作業用タイムラインに位置付け、マウス３６を押して、左右にドラッグする。図４（Ｃ）の作業用タイムライン５８－２の再生範囲（緑色表示）を移動するには、マウス３６により、ポインタを再生範囲に位置付け、マウス３６を押して、左右にドラッグする。図４（Ｃ）の作業用タイムライン５８－２の再生範囲（緑色表示）を変更するには、マウス３６により、ポインタを再生範囲の端部に位置付け、マウス３６を押して、左右にドラッグする。
【００５６】
図４（Ｃ）の作業用タイムライン５８－２のキーフレーム全体を再生範囲として、選択するには、マウス３６により、ポインタを再生範囲に位置付け、キーボード３２のＣｔｒｌキーと「Ａ」キーを押す。図４（Ｃ）の作業用タイムライン５８－２の再生範囲の任意の範囲を選択するには、マウス３６により、ポインタを再生範囲に位置付け、マウス３６の中心を押し（始点指示）、終点としたい位置までドラッグして離す（終点指示）。
【００５７】
図４（Ｃ）の作業用タイムライン５８－２の再生範囲のキーフレームを移動するには、マウス３６により、ポインタをそのキーフレームに位置付け、マウス３６を押して、左右にドラッグする。図４（Ｃ）の作業用タイムライン５８－２の再生範囲のキーフレームを削除するには、マウス３６により、ポインタをそのキーフレームに位置付け、マウス３６を押して、上のごみ箱へドラッグする。
【００５８】
このようにして、以下で説明するように、３ＤＶＩＥＷ表示部６０に表示されたロボットのポーズの時間情報を設定し、且つ確認のため、再生する。
【００５９】
次に、ロボットのポーズを設定する機能を説明する。胴体１０を除くロボット各部位の位置の変更を行うには、３ＤＶＩＥＷ表示部６０に表示されたロボットの対象部位に、マウス３６により、ポインタを位置付け、マウス３６を押して、上下左右にドラッグする。これにより、ＩＫ時には、ロボットの関節単位でなく、部位（例えば、下腿部や足部）単位に、ＦＫ時には、ロボットの関節単位に、位置、姿勢を変更できる。
【００６０】
ロボットの全体の位置の変更を行うには、ＩＫ時に、３ＤＶＩＥＷ表示部６０に表示されたロボットの胴体部位に、マウス３６により、ポインタを位置付け、マウス３６を押して、上下左右にドラッグする。ロボットの全体の傾きの変更を行うには、ＦＫ時に、３ＤＶＩＥＷ表示部６０に表示されたロボットの胴体部位に、マウス３６により、ポインタを位置付け、キーボード３２のＣｔｒｌキーとマウス３６の中心を押して、上下左右にドラッグする。キーボード３２のＣｔｒｌキーを押さずに、マウス３６の中心を押して、上下左右にドラッグすると、それに合わせて、ロボットの表示位置が移動する。
【００６１】
カメラアングルで、ロボットの表示角を変更するには、３ＤＶＩＥＷ表示部６０に、マウス３６により、ポインタを位置付け、マウス３６の右を押して、上下左右にドラッグする。ロボット各部位に対する機能の一覧（手足を対称にする場合や、手足を固定する場合

、足先を着地させる場合）を表示するには、３ＤＶＩＥＷ表示部６０のロボット対称部位に、マウス３６により、ポインタを位置付け、キーボード３２のＳｈｉｆｔキーを押し、マウス３６の右を押す。ロボットを最適サイズを表示するには、３ＤＶＩＥＷ表示部６０に、マウス３６により、ポインタを位置付け、マウス３６の左をダブルクリックする。

【００６２】

このように、３ＤＶＩＥＷ表示部６０のロボットに対するマウスの操作で、ポーズを任意に設定する。従って、３ＤＶＩＥＷ部６０で、ロボットのポーズを設定し、タイムライン表示部５８で、ポーズ間の時間を設定して、実ロボットの関節角度と、関節速度を指定できる。そして、設定されたポーズの間の時間のポーズは、自動で補完される。又、マウス操作で設定できるため、極めて、簡単に、時間間隔とポーズを設定できる。このため、ポーズの連続動作であるモーションを、容易に作成できる。更に、再生指定により、３ＤＶＩＥＷ表示部６０で、そのモーションを確認できる。

【００６３】

次に、衝突判定処理を説明する。図６のオプションタブ６４－１にある衝突判定メニューを選択すると、３ＤＶｉｅｗ上でロボットの衝突判定をする事が出来る。衝突判定は、メニューをオンにする（図示せず）と、リアルタイムに、ロボットの部位間の衝突部位を、ロボット以外の色（例えば、オレンジ色）に変える。これにより、衝突が確認できる。

【００６４】

図９を参照して、図８の衝突判定処理を説明する。

【００６５】

（Ｓ１０）衝突判定をオンとすると、処理を開始する。

【００６６】

（Ｓ１２）ロボットの各部分ごとの空間データを２つずつ順番に対比し、重なるかどうかを判定する。即ち、ロボットの各関節（モータ）の角度を設定する。次に、この角度から、ロボットの各部分をバウンディングボックス（最小包含箱）で被い、そのボックスの値（位置座標）を取得する。尚、バウンディングボックスとは、物体（オブジェクト）をすっぽりと包み込む最小の箱を言う。バウンディングボックスを２つずつ比較する。

【００６７】

（Ｓ１４）予め、重なる可能性がないものや、最初から常につながっている部分は判定の対象としない。即ち、２つのバウンディングボックスが、最初から常に接している場合には、ステップＳ１６に進む。ここで、最初から常に接して場合とは、例えば、ロボットでは、胴体と首など、直接連結し、関節を形成している部分である。又、２つのバウンディングボックスが接する可能性がない場合にも、ステップＳ１６に進む。ここで、接する可能性がない場合とは、例えば、左右の肩などは、ロボットの自由度を考えると、接する可能性がない。

【００６８】

（Ｓ１６）比較するバウンディングボックスが残っているかを判定する。残っていれば、ステップＳ１２の比較処理に進む。一方、比較するバウンディングボックスが残っていなければ、終了する。

【００６９】

（Ｓ１８）ステップＳ１４で、判定対象とされた２つのバウンディングボックスの値（位置座標）を比較して、接しているかを判定する。接していれば、接している（衝突している）部分の表示の色を変更し、ステップＳ１６に進む。図９に示すように、ロボットの左手と、左大腿部とが、この判定により、接しているため、その部位の表示色を、変更する。逆に、接していないと、ステップＳ１６に進む。

【００７０】

ここでは、計算を簡略化するために、各部分は、近似の直方体（バウンディングボックス）に置き換えている。又、球（バウンディングスフィア）などに置き換えて計算する方法もある。尚、バウンディングスフィア（最小包含球）とは、物体（オブジェクト）をすっぽりと包み込む最小の球を意味する。

【００７１】
このように、実在のロボットを正しく動作させるために、３ＤＶｉｅｗ上に表示されるロボットの各部分の衝突判定を行い、衝突した場合には、その部分の色を変えることで、ユーザーに知らせる。ユーザーは、その情報を元に、ポーズやモーション（動作）の修正を行う。
【００７２】
このような技術は、従来のコンピュータゲームなどでも実用されているが、このロボットのモーション編集での目的は、実在のロボットを正しく動作させること、そして正しくない動作による故障などからロボットを守ることである。
【００７３】
また、衝突判定を行う各部分の設定（例えば、バウンディングボックスのサイズなど）を調整することで、ロボットの個体差（機体ごとの誤差）や、より余裕を持たせた判定（ある程度の距離に近づいたら知らせる、など）を行うことができる。
【００７４】
次に、図５の接地ボタンをクリックした場合の接地処理を、図１０乃至図１５で説明する。この処理は、ロボットや動物・人体モデルなどのＣＧモデルの任意に指定した足の裏を、地面と接地するように補正する処理である。
【００７５】
図１３乃至図１５を用いて、図１０の処理フローを説明する。
【００７６】
（Ｓ２０）マウスで足裏を指定し、図１１で後述する指定部位を基準とした、モデルデータの傾き補正処理により、足の裏が地面と平行になるようにＣＧモデル全体の傾きを補正する。図１３のように、ロボットの接地すべき足を指定する（図１３の丸印）と、図１４のように、ロボットを回転し、足の裏が地面と平行になるようにＣＧモデル全体の傾きを補正する。
【００７７】
（Ｓ２２）次に、図１２で後述するモデルデータの接地処理を行う。即ち、図１４のように、足の裏が地面と平行になるようにＣＧモデルを、図１５のように、地面に、ロボットの足裏を接地させる。
【００７８】
次に、図１１により、指定部位を基準とした、モデルデータの傾き補正処理を説明する。
【００７９】
（Ｓ３０）指定される可能性のある部位（「足裏接地処理」では、足裏を含む部位）に関しては、基準となる傾き（足裏の場合は、地面と平行になる傾き）ｒ０（例えば、傾き「０」）を予め設定しておく。
【００８０】
（Ｓ３２）ＣＧモデル全体の基準点（ここでは、胴体）を「Ａ」、基準とする部位（足裏）を「ｂ」と指定する。
【００８１】
（Ｓ３４）「ｂ」の傾きｒ１を計算する。
【００８２】
（Ｓ３６）「ｂ」に関して、傾きの差ｒｄ＝ｒ１－ｒ０を計算する。
【００８３】
（Ｓ３８）モデル全体の基準点Ａの傾きを、Ａの傾き－ｒｄに変更する。傾き補正を行い、基準点Ａの座標データを保存する。
【００８４】
次に、図１２により、モデルデータの接地処理を説明する。
【００８５】
（Ｓ４０）対象モデルデータ全体で、一番低い部分の高さ座標Ｙ（地面からの）を求め

る。
【００８６】
（Ｓ４２）モデルデータの基準点Ａの高さ座標をＡｙとし、（Ａｙ－Ｙ）を計算する。これにより、図１５のように、接地する。
【００８７】
（Ｓ４４）このＡの座標データを保存する。
【００８８】
傾き補正した結果で、指定した「足裏」より低い位置に、他の部位「Ｃ」が存在する場合は、その部位「Ｃ」が接地し、「足裏」は宙に浮いてしまう。強制的に「足裏」を接地させることも可能ではあるが、その場合、「Ｃ」が地面より低い位置になってしまい不自然である。このような場合に、「足裏」が接地しないことで、ポーズの不都合を確認させ、ポーズの修正を促すことができる。
【００８９】
又、足裏接地処理では、足の裏が地面と平行になるように、モデルデータ全体の傾きを補正するが、基準となる傾きが設定されていれば、足裏でなくて頭・手・背中などの部分でもかまわない。
【００９０】
次に、図１６、図１７、図１８により、部分位置合わせ処理を説明する。部分位置合わせ処理は、予め作成されたポーズやモーションを組み合わせて、新しいモーションを作成する場合に、自然な（連続した）モーションにする処理である。従来は、目測による手作業などで行っていたが、この処理により、基準とする部分（図１７、図１８では、右足）を指定することにより、自動で行うことができる。
【００９１】
図１７、図１８の例で説明すると、図１７（Ａ）のロボットの両足が揃ったポーズから、図１７（Ｂ）の右足を１歩踏み出すモーションと、図１８（Ａ）の左足を１歩踏み出すモーションをつなげて、２歩歩くモーションを作成する場合に、そのまま、つなげても、同じ位置で踏み出す足が左右入れ替わるだけで、歩くモーションにはならない。即ち、移動しない。
【００９２】
そこで、歩くモーションにするため、図１７（Ｂ）の右足を１歩踏み出すモーションで踏み出した位置や向きに、図１８（Ａ）の左足を一歩踏み出すモーションの軸足（右足）の位置や向きを合わせ、図１８（Ａ）を、図１８（Ｂ）のように、座標変換する。この処理は、図５と同様に、プルダウンメニュー６０－１６により、指定する。
【００９３】
図１６により、部分位置合わせ処理を説明する。
【００９４】
（Ｓ５０）先ず、部分位置合わせを行うモーションＬ（図１８（Ａ））を選択する。
【００９５】
（Ｓ５２）合わせる基準とする部分（図１７、図１８では、右軸足）Ａを、マウスで選択する。
【００９６】
（Ｓ５４）モーションＬの直前のポーズＰ１（図１７（Ｂ））における基準部分Ａ（右軸足）の位置、向きデータＤ１を取得する。
【００９７】
（Ｓ５６）モーションＬのポーズＰ２（図１８（Ａ））において、基準部分Ａ（右軸足）の位置と、向きデータが、ステップＳ５４のデータＤ１と同じとなるように、ロボット全体の位置、向きを動かし、その時の位置、向きの変更を、モーションＬのデータに対し、行う。モーションがポーズの連続の場合には、モーションＬに含まれる全てのポーズに対して、同じ変化量を与える。
【００９８】

このようにして、最初のモーションの最後のポーズ（図１７（Ｂ））と、２つ目のモーションの最初のポーズ（図１８（Ａ））の基準となる部分の位置や向きを合わせることができる。

【００９９】

（モーション編集処理）

次に、前述のモーション編集機能を使用したモーション編集処理動作を説明する。

【０１００】

（１）モーション編集ソフトウエアを起動すると、表示部３４に、タイムライン表示部５８と、３ＤＶｉｅｗ表示部６０が表示される。

【０１０１】

（２）新たにモーションを作る場合には、マウス３６の操作で、作業用タイムライン５８－２の現在時刻オレンジマークを0.000秒（左端）に合わる。３ＤＶｉｅｗのホームポジションボタン６０－１１（図５参照）を押し、ロボットを初期姿勢（ポーズ）にする。

【０１０２】

（３）３ＤＶｉｅｗ表示部６０のキーフレーム設定ボタン６０－１を押し、第一番目のキーフレームを登録します。次に、第二番目のキーフレームを置きたい時間（例えば、０．５５５秒）に、マウス３６の操作で、作業用タイムライン５８－２のオレンジマークを移動する。

【０１０３】

（４）３Ｄｖｉｅｗ表示部６０内のロボットの部位を，マウス３６の操作で、掴み、ドラックすると、その部位は移動する。部位を任意の位置に移動し、ポーズを作成する。このとき、インバースキネマティクス（ＩＫ）を選ぶか、フォワードキネマティクス（ＦＫ）を選ぶかは、キー６０－２，６０－３で設定でき、任意である。ポーズが完成すると、３ＤＶｉｅｗ表示部６０のキーフレーム設定ボタン６０－１を押し、第二番目のキーフレームを登録する。第三番目のキーフレーム以降は、二番目と同様に作業する。

【０１０４】

（５）必要なキーフレーム（ポーズ）数が完成すると、1つのモーションは完成する。作業用タイムライン５８－２の再生範囲指定の緑帯を、第一番目のキーフレームから最終キーフレームまで設定し、再生ボタン（図４（Ｃ）参照）を押すと、一連のモーションが、３ＤＶｉｅｗ表示部６０上で再現される。

【０１０５】

（６）実際のロボット１が通信ポートに接続されている場合には、オプション表示部６４中のロボット通信タブ６４－１内のロボット通信にチェックを入れておけば、実際のロボット１と、３ＤＶｉｅｗ６０のロボットは、完全に同期して動く。

【０１０６】

（７）更に、Ｖｉｄｅｏ表示部６２をＯＮにし、ハードデイスクドライブから動画ファイルを読込んで、この動画に合わせて、上述の「キーフレームを置きたい時間（例えば、0.555秒）」等を選定することにより、動画と同期したロボットモーションを簡単に作成できる。

【０１０７】

（８）完成したモーション（タイムシート）は、ファイルの保存機能で、キーフレームメモリ６０に保存する。

【０１０８】

（９）又、既存のモーションを修正して、新しいモーションを作る場合には、ファイルの保存機能で保存したモーション（タイムシート）を開き、タイムライン５８－１，５８－２に乗せる。別のモーションを追加したい時は、アイコン「モーションを追加で開く」で、追加出来る。

【０１０９】

（１０）アイコン「開く」「モーションを追加で開く」代わりに、実際のロボット１内のモーションメモリ４０上にあるモーションデータを使うことも出来る。この場合は、図

６で説明したように、オプション表示部６４中のロボット通信タブ６４－３の「読み込み」「追加読み込み」をクリックする。

【０１１０】

（１１）タイムライン上に乗ったモーションデータは、ポーズの変更や、キーフレームを、前述の（４）、（５）により、変更できる。Ｖｉｄｅｏとの、同期も同様である。

【０１１１】

（１２）変更が終わったら、完成したモーション（タイムシート）を、ファイルの保存機能で保存する。同様に、実際のロボット１内のモーション用メモリ４０上に保存したい場合は、オプション表示部６４中のロボット通信タブ６４－３の「書き込み」が対応し、ロボット通信にチェックを入れ、「書き込み」ボタンをクリックして、保存する。

【０１１２】

（ロボットとの通信処理）

次に、ロボットとの通信処理を説明する。モーション編集装置３と、実ロボット１の間では、データを変換して相互に通信を行う。図１９は、編集装置３から実ロボット１への通信動作の説明図である。図２０は、実ロボット１から編集装置３への通信動作の説明図である。

【０１１３】

図１９において、編集装置３のキーフレームメモリ５２のキーフレームデータを、制御プログラム５０が読み込み、全てのポーズ（モーション）を一括で変換し、通信ドライバ５４より、ロボット１へ通信する。この場合に、ロボット１は、通信ドライバ４４を介し受けたモーションデータを、モーション用メモリ４０に記憶し、ロボット１が、後で実行する方法と、図１９のように、時間経過に合わせて、リアルタイムに、キーフレームごとに、変換し、通信し、ロボット１が実行する方法がある。

【０１１４】

又、図２０に示すように、実ロボット１のモーション用メモリ４０の全てのポーズ（モーション）を、一括して変換し、通信し、編集装置３のキーフレームメモリ５２に格納し、タイムラインにデータを展開する。

【０１１５】

以下、ポーズとは、ある時刻における、ロボットの全関節の状態（角度）またはそのデータを言い、モーションとは、一定時間範囲における複数の連続したポーズまたはそのデータを言う。

【０１１６】

又、編集装置は、図４で説明したように、時間表示に、「フレーム」という単位（基本設定では「１フレーム＝１５ミリ秒」）を使用しており、その時間ごとのポーズデータを「フレームデータ」と呼ぶ。

【０１１７】

図２１は、フレームデータの構成図であり、フレーム（モーション開始からの時間）と、ロボットの位置（仮想空間におけるロボットの基準点の位置）と、ロボットの傾き（仮想空間内におけるロボットの基準点の傾き）と、各関節の角度とからなる。

【０１１８】

更に、キーフレームは、ユーザーが任意に指定した、モーションの基準となるポーズが設定されたフレームのことを言い、そのフレームデータをキーフレームデータと呼ぶ。編集装置３では、ユーザーが、キーフレームのポーズを作成・指定し、キーフレーム間のフレームのポーズを自動補完することで、モーションを作成する。

【０１１９】

次に、実ロボット１のデータ形式を図２２で説明する。図２２に示すように、実ロボット１は、各モーター（関節）の角度と、その角度に到達するまでの時間の連続したものを、モーションデータとして扱う。

【０１２０】

上述のように、仮想空間のロボットと、実ロボット１とでは、データ形式が異なるため

、これらの間で通信するには、データ変換を必要とする。本実施の形態の通信処理では、編集装置３のキーフレームと、実ロボット１におけるモーションデータの１行を対応させている。これは全ての通信において基本的に同じである。

【０１２１】

図２３は、編集装置３から実ロボット１へ通信する場合のデータ変換の説明図である。図２３に示すように、編集装置３は、図２１のフレームを、図２２の時間（スピード）に、図２１の各関節の角度を、図２２の各モーターの角度に、図２１のロボットの位置・傾きを、追加データとして変換し、ロボット１に送信する。

【０１２２】

この変換において、フレームを時間（スピード）に変換するため、キーフレームの間隔を、時間に変換する。ここでは、１フレーム＝１５ミリ秒で計算する。又、各関節の角度を、各モーターの角度に変換するため、各関節の角度をそのまま（必要な場合は単位変換を行い）設定する。更に、擬似空間におけるロボットの基準点の位置・傾きを追加データとするため、実ロボット１のモーションデータには、初期設定で存在しない項目なので、項目ごと追加して設定する。具体的には、位置データ３項目（Ｘ，Ｙ，Ｚ座標）＋傾きデータ３項目（ヨー、ピッチ、ロー）＝６項目である。

【０１２３】

次に、図２４は、実ロボット１から編集装置３へのデータ変換の説明図である。編集装置３は、ロボット１の時間（スピード）をフレームに変換し、ロボット１の各モーターの角度を各関節の角度に、ロボット１の追加データを、擬似空間におけるロボットの基準点の位置・傾きに設定する。又、直接教示機能を使用する場合には、ロボット１の各モーターの角度を各関節の角度に設定して、データ通信を行う。

【０１２４】

この変換において、ロボット１の時間（スピード）データは、キーフレームの間隔に変換する。１フレーム＝１５ミリ秒で、フレーム単位に変換する。各モーターの角度は、そのまま（必要な場合は単位変換を行い）、各関節の角度に設定する。更に、仮想空間におけるロボットの基準点の位置・傾きのデータ項目は、実ロボット１のモーションデータには、初期設定で存在しない項目なので、ロボット１に追加データとして存在する場合（即ち、編集装置３で作成して、ロボット１に記録されたデータ）は、そのまま設定し、存在しない場合はデータ処理を行わない。

【０１２５】

次に、図１９に示したように、編集装置３が、時間経過に合わせて、リアルタイムに、キーフレームごとに、変換し、通信し、ロボット１が実行する場合に、編集装置３が実行する通信遅延対応処理を説明する。

【０１２６】

図２５は、通信遅延補償処理フロー図である。通常、ＣＧの描画やゲームを実行中に処理落ち（処理の実行時間が、プログラムで想定された予定時間を超えてしまう状況）が起きた場合、処理落ちを認識した時点で、未実行の処理を全て削除して、その時点で実行すべき処理から順次処理を行うなどの対応処理を行う。

【０１２７】

しかし、実際のロボット１に対して、リアルタイム通信でモーション（動作）操作を行う場合、処理できない部分を単純に削除してしまうと動作が連続せず、場合によっては動作が実行不能になったり、ロボットが大幅にバランスを崩してしまったりする。このような結果を回避するために、処理が間に合わなかった部分を削除することなく、処理速度を変えることで対応する処理を行う。

【０１２８】

（Ｓ６０）先ず、次に、ロボット１に送信するデータＤ１と、そのデータの実行完了時間Ｔ１（図２１のフレームによる時間）を取得する。

【０１２９】

（Ｓ６２）次に送信するデータが有る場合には、通信処理の前に、現在時刻Ｔ０を取得

し、実行完了時刻Ｔ１までの時間Ｔｄ（＝Ｔ１－Ｔ０）を計算し、ステップＳ６４に進む。

【０１３０】

（Ｓ６４）一方、次に送信するデータがない場合には、通信以外の処理（描画等）を実行する。

【０１３１】

（Ｓ６６）通信以外の処理の実行後、ロボット通信タブがＯＮかを判定し、オンなら、ステップＳ６０に戻り、オンでないなら、ステップＳ６４に戻る。

【０１３２】

（Ｓ６８）そして、時間Ｔｄ（若しくは、Ｔｄから計算される速度）とデータＤ１を、ロボット１に送信し、ステップＳ６４に進む。

【０１３３】

このように、各通信において、現在時刻と実行完了時刻とから修正時刻を計算するので、前回の通信処理や他の処理（描画など）の影響で、予定通りでない時刻に、通信処理が行われる場合でも、遅れた時間に間に合わせるように動作を早くすることができる。このため、実際のロボットの動作完了時刻が変わらないようにすることができる。

【０１３４】

またこの方法は、描画なども含む全ての処理を一連の流れで行う仕様に大変有効であるが、通信部分だけが独立して機能するような仕様（並列処理、マルチタスクなど）でも有効である。

【０１３５】

次に、図２０で説明した編集装置３に、実ロボット１からモーションデータを読み込み、キーフレームとしてタイムライン上に登録する処理を説明する。図２６乃至図２９は、モーションデータを、キーフレームデータに変換するための説明図である。

【０１３６】

図２６に示すように、実ロボット１のモーションデータは、一般的にモータの動作時間と、関節のモータ角度を、時系列に並べたものである。ここでは、制御可能関節数は最大２４個（図の横方向の番号）であるが、この実施の形態の実ロボット１（図１）の実際の関節数は１７個である。

【０１３７】

図２６は、１７個の関節を持つロボットの動作が、１３個のポーズ（図の縦方向の番号）からなるモーションデータの例である。即ち、図２６の第一列目は、１３個のポーズを上から順につけた行番号である。又、第14行は、終了を意味する。

【０１３８】

第二列目は、モータの動作時間を、１６進数で示している。この実ロボット１は、動作の時間単位が１５ｍｓであるから、１行目の「ａ」と言う動作時間は、１５ｍｓの（１０進数の）１０倍の１５０ｍｓを示している。各モータは、目的のポーズへ、１５０ｍｓで達するだけのスピードで動作することを示す。

【０１３９】

第三列目は、関節番号１のモータの角度（ポーズ）を示す。図２６の「４０００」と言う値は、中心角度を示す（相対値で示す）。第四列目から第二十四列目までは、第三列目と同様に、各関節番号のモータの角度を示す。尚、５、９、10、16、22、23、24の関節は未使用なので、数値に意味はない。

【０１４０】

このモーション動作の説明をすると、開始ポーズから１行目のポーズに１５０ｍｓで到達し、続けて２行目のポーズに（２０*１５＝）300mS後に達するモーションであり、以下同様な繰り返しを行う。

【０１４１】

上述の例では、モーションデータのモータ角度は、全てロボットのポーズを示す。このようなモーションデータは一般的で、特開平10-302084の図５でも記載されている。

【０１４２】
本実施の形態では、このようにモーションデータから、キーフレームデータを作成し、タイムライン上に登録する処理である。図２６のモーションデータのモータ角度枠のデータが、全てロボットポーズならば、キーフレームメモリのキーフレームデータに変換することができる。そして、キーフレームデータにより、タイムライン上のキー位置が時間に対応し、モータ角度は、３ＤＶｉｅｗ上のロボットのポーズで示される。
【０１４３】
しかしながら、モーションデータのモータ角度枠に、モータ角度以外のデータが記述される場合もある。この場合に、これをそのままキーフレームデータに変換しても、３ＤＶｉｅｗ上に表示するロボットのポーズは、どのような形になるか保障できない。
【０１４４】
このように、従来から知られるモーションデータ（のモータ角度枠）に、モータ角度以外のデータが記述される場合でも、正しいポーズを取ることの出来る処理である。図２７は、モーションデータ（のモータ角度枠）に、モータ角度以外のデータが記述されたモーションデータの一例の説明図である。
【０１４５】
図２７では、３行目と６行目に、モータ角度枠に、モータ角度ではない数値が示されている。即ち、このロボットでは、主に８０００番台の数値がこれらの対象である。３行目の「８２０ａ」は、「ループカウンタを３にセットする」と言う意味で、６行目の「８３０ａ」は、「ループカウンタをデクリメントして、０で無ければ、指定した行にジャンプする」と言う意味である。
【０１４６】
このようなループ制御、条件ジャンプコマンド（「８２０a」や「８３０a」）は、実ロボット内の制御プログラムが、解釈判断（インタープリター）するコードである。このモーションデータを、編集装置３が読み出しても、キーフレームデータに変換出来ない。
【０１４７】
このため、まず、１行目、２行目をタイムライン上のキーフレーム位置にポーズに登録する（キーフレームメモリ５２に登録する）。ついで、３行目は、モーションデータであるにもかかわらず、タイムライン上のキーフレーム位置にポーズを登録しない。タイムライン上のキーフレーム位置にポーズを登録しないで、実ロボット１内の制御プログラムが解釈すると同じ解釈を行う。この場合は、「３」と言う値をループカウンタに登録する。
【０１４８】
続けて、４行目、５行目は再び、タイムライン上のキーフレーム位置にポーズを登録する。６行目は、これを登録せず、ループカウンタの値を１つ減らす。即ち、「２」とする。
【０１４９】
ループ処理のため、続けて、４行目、５行目をタイムライン上のキーフレーム位置にポーズを登録する。さらに、４行目、５行目をタイムライン上のキーフレーム位置にポーズを登録する。この段階では、ループカウンタの値は「０」なので、７行目、８行目、９行目とつづけてタイムライン上のキーフレーム位置にポーズを登録する。以上のような方法でタイムラインを完成させる。
【０１５０】
タイムライン上のキーフレームは、全てモータ角度で構成されており、実働可能なものとなる。
【０１５１】
図２８は、その変換処理フロー図である。
【０１５２】
（Ｓ７０）編集装置３は、モーションメモリの行数Ｍと、キーフレームメモリの行数Ｋを、「１」に初期化する。
【０１５３】

（Ｓ７２）Ｍ行目のモーションデータを調べる。

【０１５４】

（Ｓ７４）モーションデータが終了（「ｆｆｆｆ」）であるかを判定し、モーションデータが終了であれば、終了のキーフレームデータを、メモリ５２に書き込み、終了する。

【０１５５】

（Ｓ７６）逆に、モーションデータが終了でなければ、モータ角度以外のデータ（データが８０００番台）かを判定する。モータ角度以外のデータなら、ステップＳ８０のループ処理に進む。逆に、モータ角度のデータであれば、モーションデータを、キーフレームに割り付ける（キーフレームメモリ５２に書き込む）。そして、モーション用メモリの行数Ｍと、キーフレームメモリの行数Ｋを、「１」インクリメントし、ステップＳ７２に戻る。

【０１５６】

次に、ループ処理を説明する。

【０１５７】

（Ｓ７８）データが何番であるかを判定する。

【０１５８】

（Ｓ８０）データが「８２０ａ」である場合には、ループ開始処理を行う。ここでは、例として、ループカウンタに、「３」をセットする。更に、ループ開始行Ｌは、Ｍを「１」インクリメントした値で保存する。

【０１５９】

（Ｓ８２）そして、モーション用メモリの行数Ｍを、「１」インクリメントする。

【０１６０】

（Ｓ８４）一方、データが「８３０ａ」である場合には、ループ終了判定処理を行う。ここでは、ループカウンタの値を、「－１」する。

【０１６１】

（Ｓ８６）そして、ループカウンタの値が、「０」であるかを判定する。「０」なら、モーション用メモリの行数Ｍを、「１」インクリメントして、ループを終了し、「０」でないなら、モーション用メモリのジャンプ先の行数Ｍを、ループ開始行Ｌとして、ループ開始部分へ戻る。

【０１６２】

このように、モーションデータ内のデータで、それがモータ角度ならば、そのままタイムライン上のキーフレーム位置に、ポーズを登録し、モータ角度以外のデータなら、タイムライン上のキーフレーム位置に、ポーズを登録せず、ロボット内のインタープリター動作と同様な動作に切り替え、タイムラインを完成する。

【０１６３】

次に、これらのデータを、編集装置３で運用する場合には、各行に、２５カラム目から３０カラム目に、ワールド座標・角度を加える。ここで、ワールド座標・角度とは、ロボットが存在する擬似空間の基準点・基準方向に対するロボットの基準点の位置座標・傾きの角度を言う。ロボットの基準点は、例えば、胴体部である。

【０１６４】

図２９は、その一例を示す。例えば、前述の足裏接地処理、本体移動処理などを行うことで、モータの角度データに、上記データが組み合わされ、新しいモーションデータが作成される。

【０１６５】

図２９において、第二四列目から第三十列目までは、追加したワールド座標・角度の値である。尚、５，９，２２，２３，２４番モータは使用していないので、「０」としてある。２５列目が、Ｘ座標、２６列目がＹ座標、２７列目がＺ座標、２８列目がヨー、２９列目がピッチ、３０列目がローである。この値は、上述のように、擬似空間の基準点・基準方向と、ロボットの基準点の位置座標・傾きの角度との相対値で示している。

【０１６６】

このように、編集装置３上で作成されたデータのロボットへの「書き込み」、ロボットからの「読み込み」は、上記の２５カラム目から３０カラム目を加えたデータで行う。即ち、上記データの２５カラム目から３０カラム目に当たるワールド座標・角度データを操作することによって、仮想空間内でのロボットの移動を自然に表現することができる。例えば、前述の部分位置合わせ処理や、足裏接地処理を実行した場合である。

【０１６７】

このように、モーションデータを、タームライン上に乗せるため、すべて角度データで構成されるよう変換する。これを、モーションデコーダと呼ぶ。

【０１６８】

（ロボットとの同期処理）

次に、ロボットとの同期処理を、図３０乃至図３３で説明する。図３０に示すように、ハードデイスクドライブＨＤＤ内には、音声を含むビデオデータが格納されている。ビデオデータは、ビデオドライバ５６に読み込まれ、ビデオ表示部６２に表示される。タイムライン表示部５８を操作して、このビデオデータと、キーフレームデータを作成し、更に、ビデオデータの再生と、キーフレームデータによる実ロボット動作、３ＤＶｉｅｗでの仮想ロボット動作を同期する。

【０１６９】

図３１乃至図３３の例で説明する。図３１乃至図３３は、表示部３４の３ＤＶｉｅｗ表示部６０と、ビデオ表示部６２を示す。図３１乃至図３３に示すように、タイムライン表示部５８の再生領域を操作すると、タイムライン表示部５８で指定された経過時間と、ビデオデータに含まれる再生時間情報から得たビデオの経過時間が同じになるようにすることで、ビデオ表示部６２に、タイムライン経過時間と同期したビデオ画像が表示される。図３１のように、ビデオ画像表示で、右側の人間Ｂが、手を水平に伸ばした状態で、再生を一時中断し、３ＤＶｉｅｗ６０を、マウス操作し、３ＤＶｉｅｗでのロボットのポーズを、手を水平に伸ばしたポーズに作成し、キーフレームに登録する。

【０１７０】

次に、ビデオ再生を再開し、図３２のように、ビデオ表示で、右側の人間Ｂが、手を左横に曲げた状態で、再生を一時中断し、３ＤＶｉｅｗ６０を、マウス操作し、３ＤＶｉｅｗでのロボットのポーズを、手を左横に曲げたポーズに作成し、キーフレームに登録する。

【０１７１】

更に、ビデオ再生を再開し、図３３のように、ビデオ表示で、右側の人間Ｂが、かがんだ状態で、再生を一時中断し、３ＤＶｉｅｗ６０を、マウス操作し、３ＤＶｉｅｗでのロボットのポーズを、かがんだポーズに作成し、キーフレームに登録する。

【０１７２】

このように、ビデオ画像のポーズと同じポーズのデータ作成が容易となる。

【０１７３】

又、このタイムライン表示部５８の再生を指示すると、キーフレームデータが、図３０のように、ビデオ表示と同期して、実ロボット１に送信される。このため、実ロボット１は、図３１～図３３のビデオ表示の人間Ｂと同じ動作を、ビデオ表示と同期して、実行する。この時、３ＤＶｅｗ６０のロボットモデルも、実ロボット１と同じ動作を行う。

【０１７４】

このため、ビデオに同期したコンテンツの作成が容易であり、且つビデオに同期して、実ロボットを動作でき、ロボットのエンタテイメント性を向上できる。この場合に、前述の通信遅延補償処理は、特に、有効となる。

【０１７５】

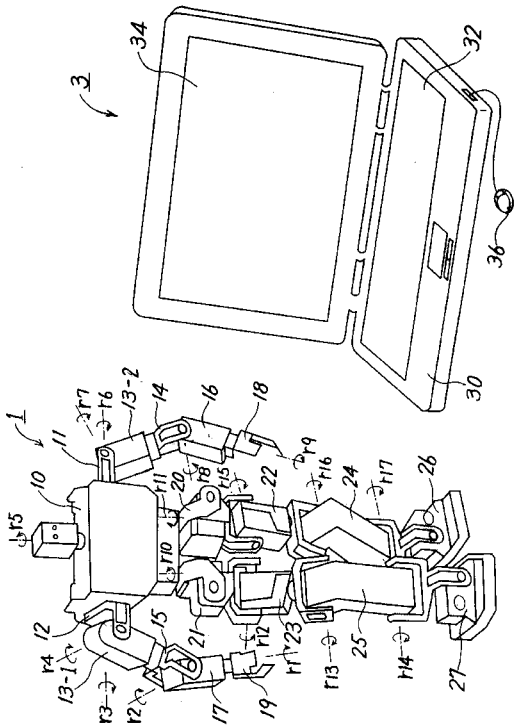
（他の実施の形態）

前述の実施の形態では、１７関節のロボットで説明したが、他の関節数のロボットにも適用できる。ロボットは、人型のみならず、４足歩行型等の他の形態をしたロボットに適用できる。

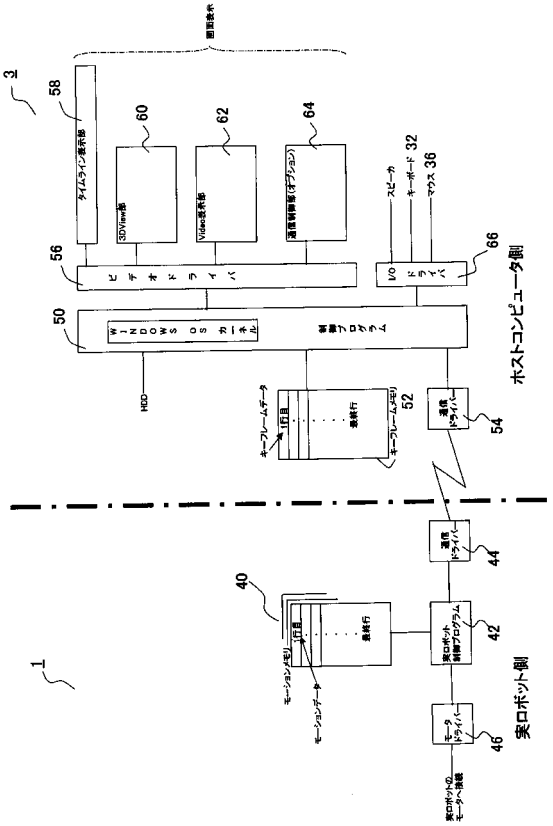
【０１７６】
以上、本発明を、実施の形態で説明したが、本発明は、その趣旨の範囲内で種々の変形が可能であり、これを本発明の範囲から排除するものではない。
【産業上の利用可能性】
【０１７７】
操作ユニットの操作に応じて、３次元擬似空間でのロボットのモデルの各部位を位置、傾きを変更して、操作ユニットのデータ作成指示に応じて、変更されたロボットのモデルの各部位を位置、傾きから、各関節の角度を計算し、キーフレームデータを作成し、３次元表示のための追加データを付して、多関節型ロボットに送信するので、３次元擬似空間で作成されたデータを再利用して、他のモーションデータを３次元擬似空間で作成できる。
【図面の簡単な説明】
【０１７８】
【図１】本発明のモーション編集装置の一実施形態の構成図である。
【図２】図１のモーション編集装置のブロック図である。
【図３】図１、図２の表示部の説明図である。
【図４】図３のタイムライン表示部の説明図である。
【図５】図３の３ＤＶｉｅｗ表示部の説明図である。
【図６】図３のオプション表示部の説明図である。
【図７】図１、図２、図３の処理と操作の関係図である。
【図８】図７の衝突判定処理フロー図である。
【図９】図９の衝突判定処理の説明図である。
【図１０】図７の足裏接地処理フロー図である。
【図１１】図１０の傾き補正処理フロー図である。
【図１２】図１０の接地処理フロー図である。
【図１３】図１０の足裏接地処理の第１の説明図である。
【図１４】図１０の足裏接地処理の第２の説明図である。
【図１５】図１０の足裏接地処理の第３の説明図である。
【図１６】図７の部分位置合わせ処理フロー図である。
【図１７】図１６の部分位置合わせ処理の第１の説明図である。
【図１８】図１６の部分位置合わせ処理の第２の説明図である。
【図１９】図２のロボットへの通信処理の説明図である。
【図２０】図２のロボットからの通信処理の説明図である。
【図２１】図１９、図２０のフレームデータの説明図である。
【図２２】図１９、図２０のモーションデータの説明図である。
【図２３】図１９のデータ変換処理の説明図である。
【図２４】図２０のデータ変換処理の説明図である。
【図２５】図１９の通信遅延補償処理フロー図である。
【図２６】図１９のデータ変換処理のためのモーションデータの説明図である。
【図２７】図１９のデータ変換処理のための他のモーションデータの説明図である。
【図２８】図２７のモーションデータの変換処理フロー図である。
【図２９】図２０のデータ変換のための送信データの説明図である、
【図３０】本発明の一実施の形態の動画との同期処理の説明図である。
【図３１】図３０の同期処理の第１の画面例の説明図である。
【図３２】図３０の同期処理の第２の画面例の説明図である。
【図３３】図３０の同期処理の第３の画面例の説明図である。
【符号の説明】
【０１７９】
１　多関節型ロボット
３　モーション編集装置

３２　キーボード
３４　表示部
３６　マウス
４０　モーションメモリ
４２　実ロボット制御プログラム
５０　制御プログラム
５２　キーフレームメモリ
５４　通信ドライバ
５８　タイムライン表示部
６０　３ＤＶｉｅｗ表示部
６２　オプション表示部
６４　ビデオ表示部

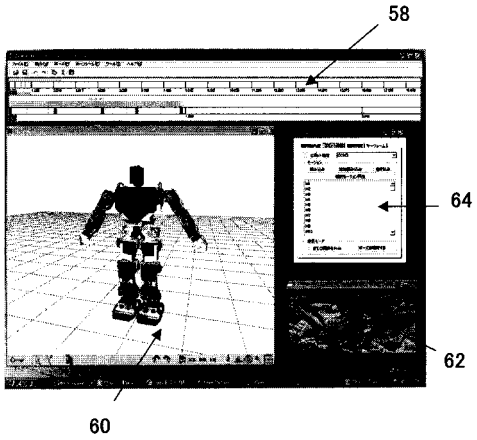
【図１】



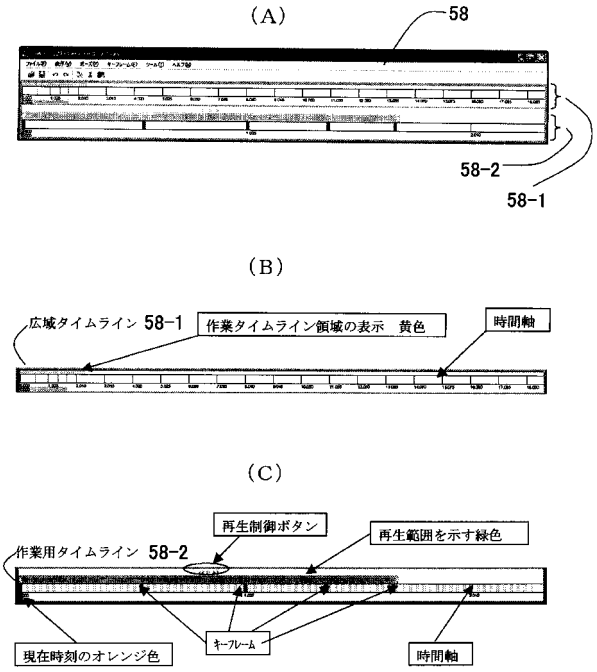
【図２】



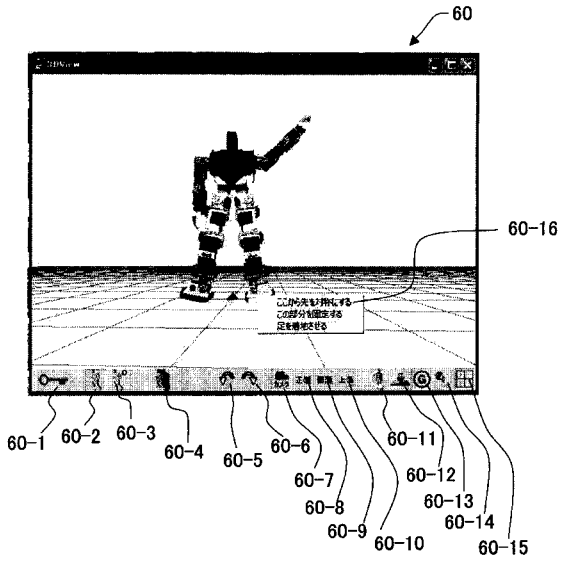
【図３】


58
64
62
60

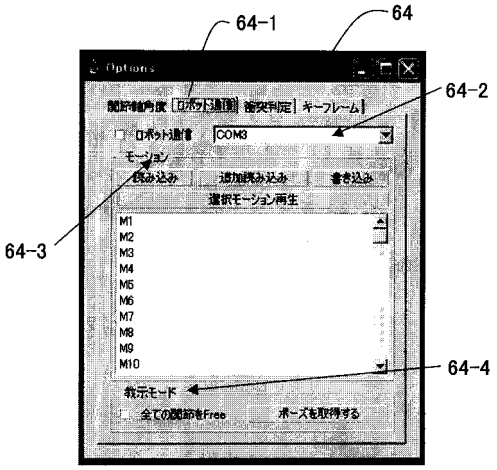
【図４】


(A)
58
58-2
58-1
(B)
広域タイムライン 58-1
作業タイムライン領域の表示　黄色
時間軸
(C)
作業用タイムライン 58-2
再生制御ボタン
再生範囲を示す緑色
キーフレーム
現在時刻のオレンジ色
時間軸

【図５】


60
60-16
60-1
60-2
60-3
60-4
60-5
60-6
60-7
60-8
60-9
60-10
60-11
60-12
60-13
60-14
60-15

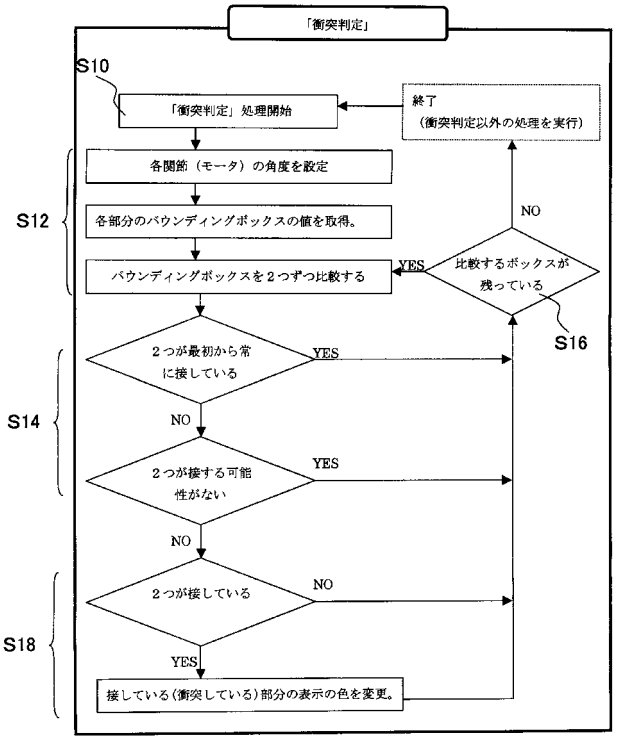
【図６】


64-1
64
64-2
64-3
64-4
Options
関節軸角度
ロボット通信
衝突判定
キーフレーム
COM3
モーション
読み込み
追加読み込み
書き込み
選択モーション再生
M1
M2
M3
M4
M5
M6
M7
M8
M9
M10
表示モード
全ての関節をFree
ポーズを取得する

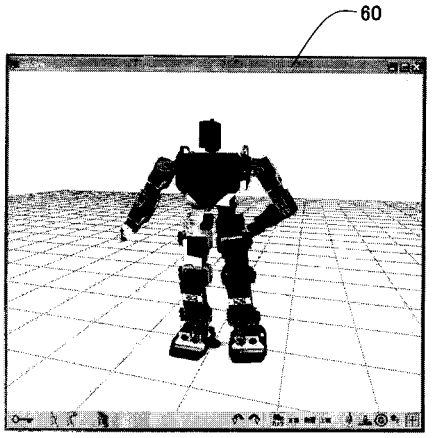
【図７】

操作したい事	マウス矢印位置	マウス 左	マウス 中心	マウス 右	キーボード Shift	キーボード Ctrl	キーボード キー
作業用ライトンの選択位置変更	黄色	押して左右へドラッグ					
表示範囲の移動	緑色	押して左右へドラッグ					
表示範囲の変更	緑色帯の左右矢印	押して左右へドラッグ				○	A
表示範囲：1フレーム全体を選択	緑色帯左右		押す（始点）、離す（終点）				
表示範囲：任意範囲の選択	緑色						
キーフレームの移動	赤色	押して左右へドラッグ					
キーフレームの削除	赤色	押して上（ゴミ箱）へドラッグ					
体ヶ各部位の位置変更(体：肩体)	3D View：体ヶ対象部位	押してドラッグ					
体ヶの各体位置変更(IK体)	3D View：体ヶ肩体部位	押してドラッグ					
体ヶの全体角度変更(FK体)	3D View：体ヶ肩体部位	押してドラッグ				○	
ズームイン／アウト	3D View		押してドラッグ				
パン：上下／左右	3D View		押してドラッグ				
カラダアクセ：体ヶ表示有無変更	3D View：体ヶ対象部位			押してドラッグ	○		
体ヶ各部位の色変更	3D View	ダブルクリック				○	
ロボットの最適サイズ表示	3D View：対象箇所				○	○	○
手足二つの体ヶを対称にする	3D View：対象箇所				○	○	○
手足二つの部分を固定する	3D View：対象箇所				○	○	○
足裏：足を接地させる	3D View：対象箇所						

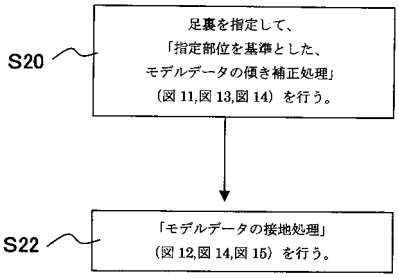
【図８】



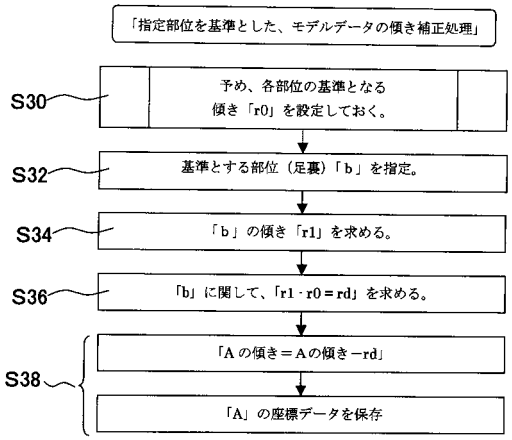
【図９】



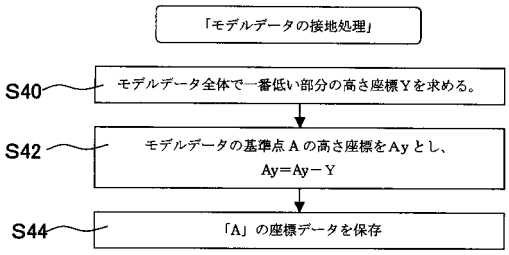
【図１０】



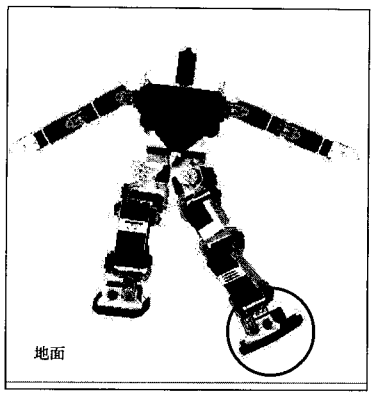
【図１１】



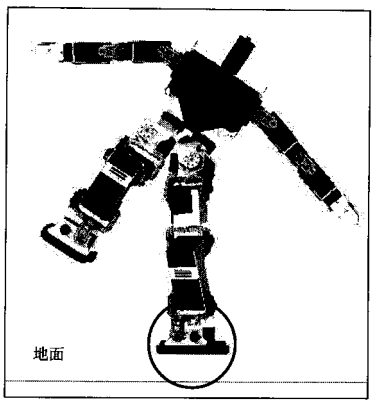
【図１２】



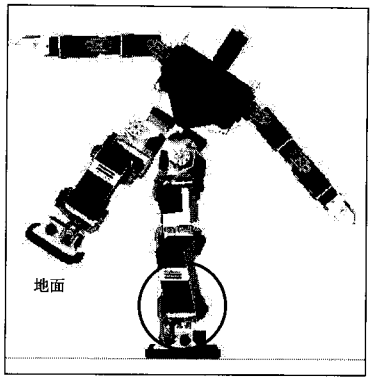
【図１３】



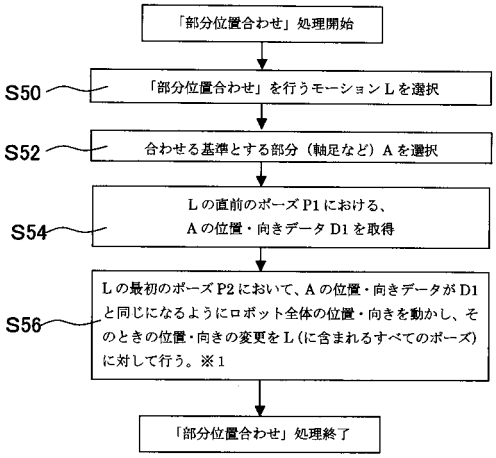
【図１４】



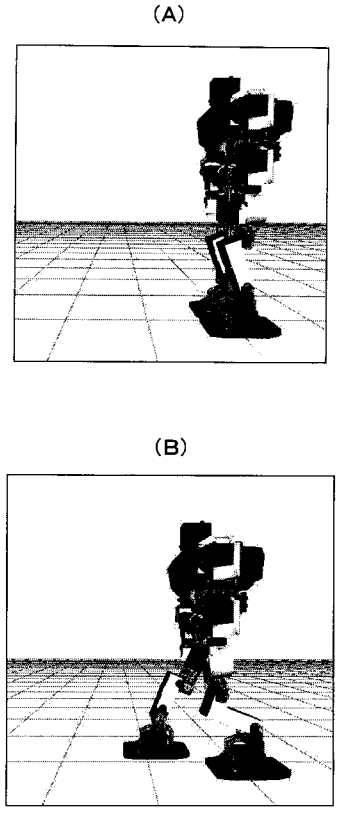
【図１５】



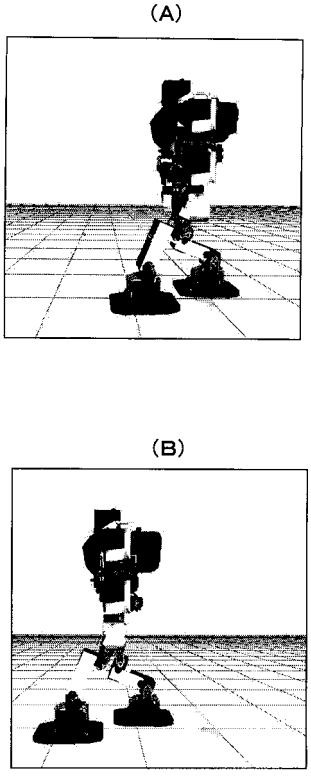
【図１６】



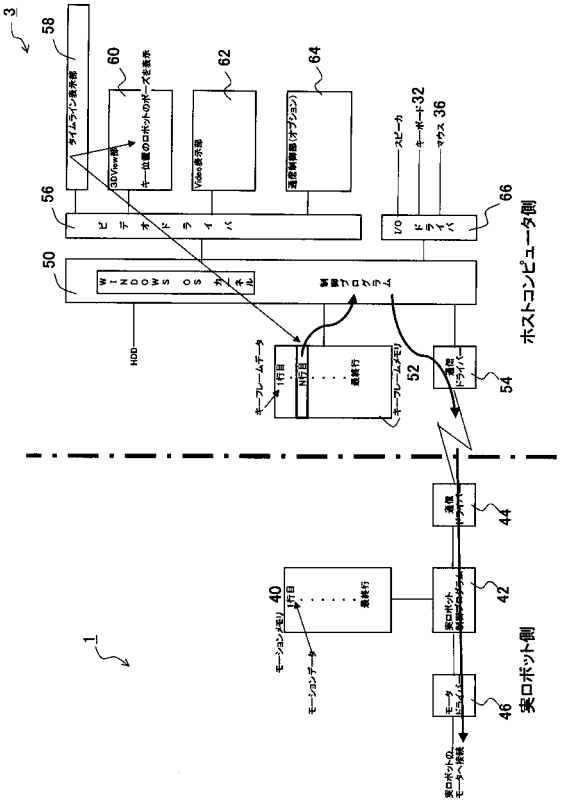
【図１７】



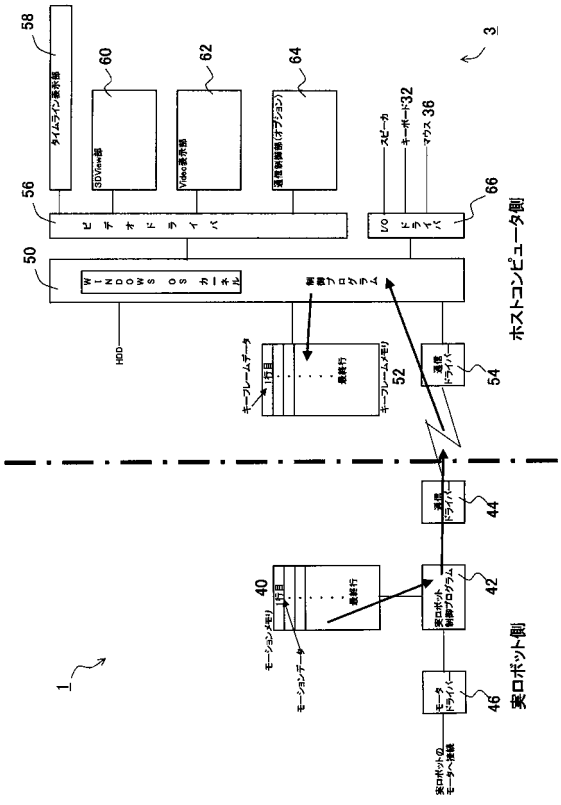
【図１８】



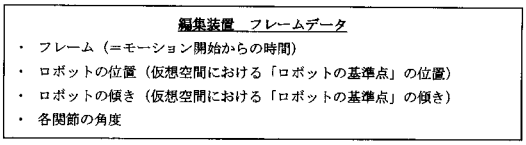
【図１９】



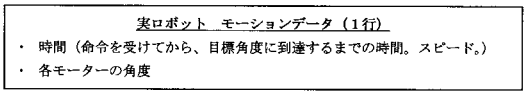
【図２０】



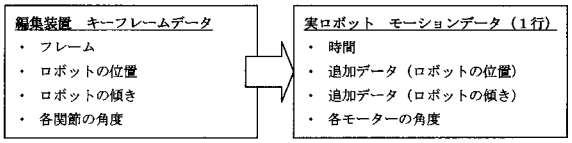
【図２１】



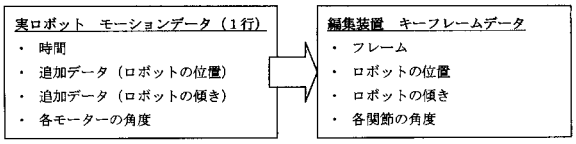
【図２２】



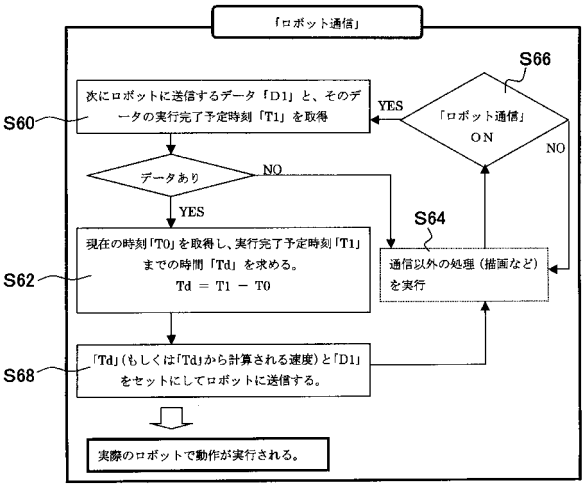
【図２３】



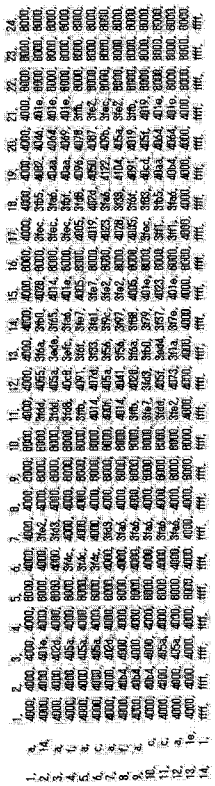
【図２４】



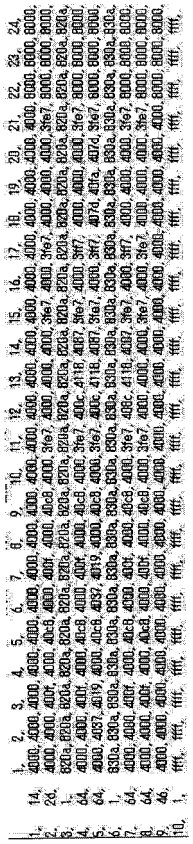
【図２５】



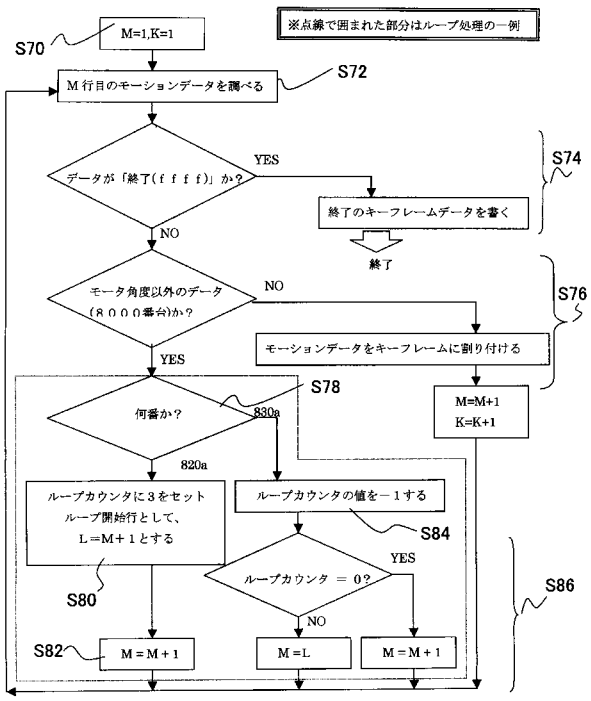
【図２６】



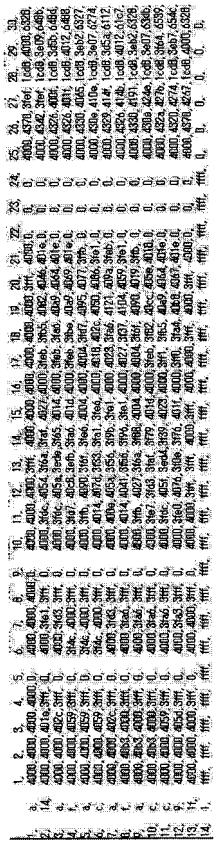
【図２７】



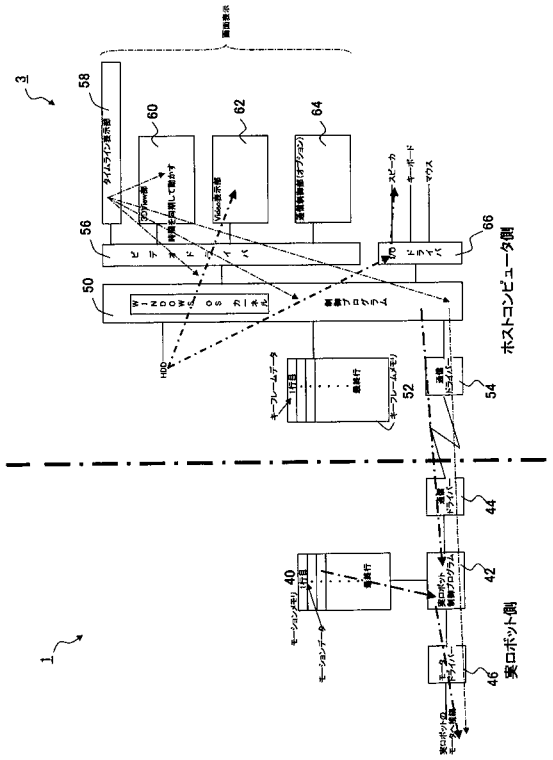
【図２８】



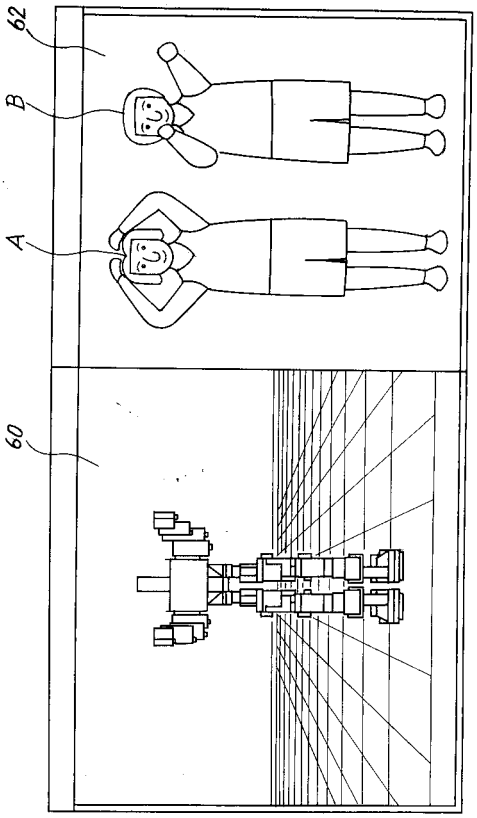
【図２９】



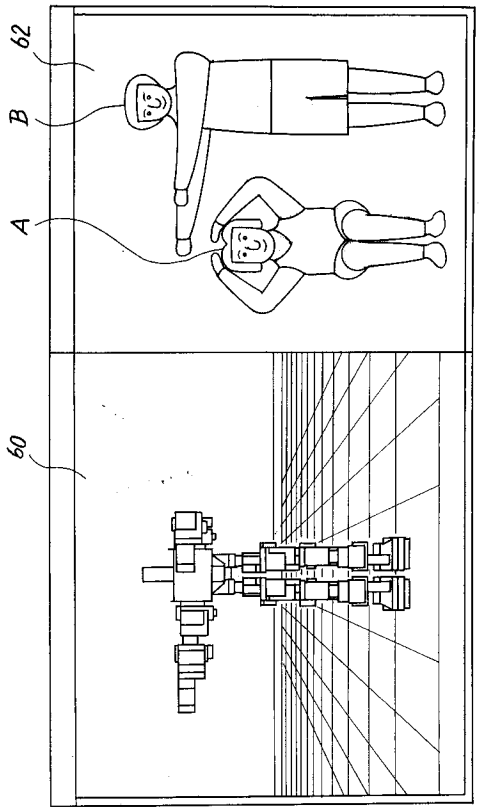
【図３０】



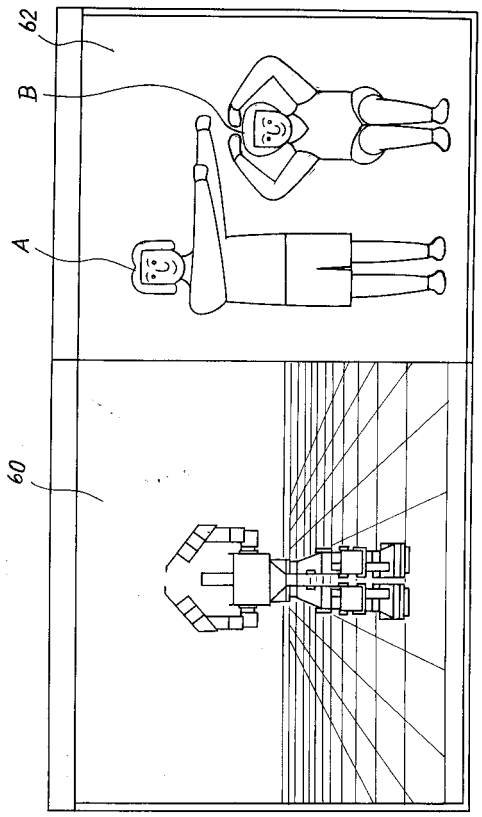
【図３１】



【図３２】



【図３３】



フロントページの続き

(72)発明者　片寄　雅之
東京都大田区羽田１丁目２番１２号　　株式会社セガ内

Ｆターム(参考) 3C007 AS36 CS08 JU17 LS05 LS09 LS10 LS11 LS14 WA03 WA13
WC03
3C269 AB33 BB09 EF69 MN09 MN42 MN48 QC01 QC03 QC10 QD02
QD03 QD10 QE02 QE21 SA11 SA15 SA17 SA33
5B050 BA08 BA09 CA07 EA24 EA28 FA02 FA09