

触覚デバイスを用いたバーチャルギターの開発

田村 真晴*・山下 英生**

(平成20年10月31日受理)

Development of a Virtual Guitar using Haptic Device

Masaharu TAMURA and Hideo YAMASHITA

(Received Oct. 31, 2008)

Abstract

In recent years, a haptic device that output power as one of the computer output devices has been developed. We can get the feeling that we really touch the material through a sensor of haptic device when we touch a material simulated in a computer.

In this research, a virtual guitar in which the feeling playing guitar and the sound volume are changed by adjusting power to input with a haptic device was developed. With the haptic device we feel as if we play a genuine guitar. Moreover, it seems that we can play various musical instruments by enhancing the system.

The purpose of this paper is to describe the composition of the virtual guitar system and the calculation methods of power and sounds playing virtual guitar.

Key Words: haptic device, virtual guitar, vibration of bowstring, repulsion power of bowstring

1. はじめに

コンピュータには様々な出力デバイスがある。映像や文字を出力するディスプレイ、音を出力するスピーカーなどが一般的な出力デバイスであるが、力を出力する触覚デバイスと呼ばれるデバイスがある。触覚デバイスは、コンピュータ内で計算・設定した力を、触覚デバイスのコントローラに出力することができる装置である。これを応用することにより、コンピュータ内で模擬された物質に触覚デバイスを用いて触ると、実際にその物質に触っているような触感を得ることができる。

この触覚デバイスを使い、医学の分野において人間の本物の身体を用いることなく、コンピュータを用いて手術の訓練をするシステムの研究開発が行われている¹⁾。対象を

モデリングした映像をディスプレイに表示し、対象の動きや、それに触ったときの弾力などを計算して触覚デバイスに力を出力することにより、視覚および触覚情報を提示できる。また、単純にある作業の力覚情報のみを記録し、その情報を元にその作業の訓練を行うシステムの研究開発も行われている²⁾。これは、ベテランの操作者がある作業を行った力覚情報を記録し、その力覚情報を再生することにより作業の力覚情報を実感することで、その作業の訓練を行うものである。

本研究では、この触覚デバイスを用いて、入力する力を加減することにより、ギターを弾く触感や音の大きさが変化するバーチャルギターシステムの開発を行った^{3), 4)}。触覚デバイスを用いてコンピュータ内で模擬されたギターを弾くと、本物のギターを弾いているような触感が得られ

* 広島工業大学大学院工学研究科情報システム工学専攻

** 広島工業大学情報学部知的情報システム学科

る。また、ギターを弾く力を加減することにより音の強弱をつけることができる。このようなバーチャル楽器は、他の楽器への拡張も可能で、さまざまな応用分野があると思われる。

2. バーチャルギターシステムの概要

バーチャルギターとは、コンピュータ内で模擬されたギターを、本物のギターを弾いているような感触で弾くことができるシステムを目指す。触覚デバイスのコントローラとギターを弾くピックを連動させて、ピックを用いてコンピュータディスプレイ上に表示されたギターを弾く。これにより、各弦に設定された振動周波数の音出力される。触覚デバイスは、バーチャルギターシステムにおいてギターの弦を弾く際の入力装置として使用され、ギターの弦を弾いたときの力の反応を感じることができる。

触覚デバイスを用いる利点としては、実際にギターを弾いているような感触が得られることである。弦の長さや線密度、弦とピックの距離などから弾力を計算して、触覚デバイスのコントローラに力を出力することにより、実際にギターを弾いているような感触が得られる。また、ギターを弾く強さによって音の強弱をつけることができる。強い力でギターを弾けば大きな音が、弱い力で弾けば小さな音出力される。

バーチャルギターの音階の変更は、キーボードのキーを押さえることにより実現する。ある一つのキーを押すと、そのキーに対応したコードの音階を各弦に設定し、キーを押している間にその弦を弾けば弦に設定された音階の音出力される。音出力されている最中にピックが弦に触れると、その弦が出力している音が止まる。

3. 触覚デバイス

触覚デバイスは、コンピュータ内で設定した力を出力することが出来るデバイスである。本研究で使用した触覚デバイスは「PHANTOM Omini」と呼ばれる触覚デバイスであり、本体から伸びたアームの先にペン型のコントローラが付いている。「PHANTOM Omini」は最大出力が3.3Nと小さいため、あまり大きな力を使うシステムは作れないが、ギターの反発力程度なら適当である。触覚デバイス「PHANTOM Omini」の外観を図1に示す。

触覚デバイスのコントローラは、ワークスペース内におけるコントローラの現在位置座標を知ることができる。図1のコントローラ部分を持って動かすことにより、コントローラの動きに連動して、ディスプレイ上の三次元空間内で表示されたカーソルが動く。このカーソルを操作して、三次元空間内のオブジェクトに接触すれば、ユーザはその

接触による反発力を手に感じることができる。あらかじめ設定された硬さや摩擦力などから力を計算し、その大きさに相当する力がコントローラにフィードバックされる。この結果、このオブジェクトが表す物体に触っているような触感を得ることができる。

オブジェクトへの接触だけではなく、コントローラに付いているボタンの入力、カーソルがオブジェクトの一定距離内に近づくなどのイベントに応じて力を出力することもできる。触覚デバイスとコンピュータの関係を図2に示す。また、単に入力装置や出力装置として使用することも可能である。つまり、三次元コントローラとして使用したり、キーボードなどの他の入力装置からの入力に応じて力を出力したりすることも可能である。



図1 触覚デバイス「PHANTOM Omini」

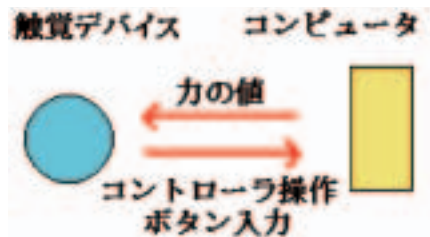


図2 触覚デバイスとコンピュータの関連

4. 力の計算

ピックを用いて弦を弾いたときの反発力は、次のように計算する。まず弦の中心に最も近いピック上の点を求め、この点と弦の中心位置から接触判定や力の計算を行う。そして、弦長から弦の反発係数を計算し、それらを元に弦の反発力を計算する。このようにして求めた力を触覚デバイスに出力する。

4.1 弦の中心に最も近いピック上の点の座標算出

図3にピックを使用して弦を弾くときの、ピックと弦の位置関係を示す。弦の中心点を点Oとし、点Oに最も近いピック上の点Pの座標を求める。この点の座標は、弦とピックの接触判定や反発力の計算に使用する。ピックの両端をそれぞれ点D、点Eとする。点OからDEを通る直線に下ろした垂線とその直線の交点が点Pとなる。点Oに最も近いピック上の点の座標を求めるには、まずピックの傾きからベクトル \vec{OP} の単位ベクトル \vec{N} を求めておく。点D、点Eの座標はピックの中心座標とピックの傾きから求めることができる。長さ $|\vec{OP}|$ を求めれば、点Pの座標が求まる。すなわち、長さ $|\vec{OP}|$ は(1)式により、点Pの座標は(2)式により求めることができる。

$$|\vec{OP}| = \vec{OD} \cdot \vec{N} \dots\dots\dots (1)$$

$$\vec{P} = \vec{O} + \vec{N} \cdot |\vec{OP}| \dots\dots\dots (2)$$

(2)式の \vec{P} 、 \vec{O} は点P、点Oの座標を表すベクトルである。点Pが線分DE上にあるとき、点Pが点Oに最も近いピック上の点となる。点Pが線分DEの外側にある場合、点Pのx座標がピックの中心点Cのx座標より小さければ点Eを、大きければ点Dを、点Oに最も近いピック上の点とする。点Pが線分DE外にある状態を図4に示す。上述のようにして求めた、点Oに最も近いピック上の点を使い、接触判定や反発力の計算を行う。

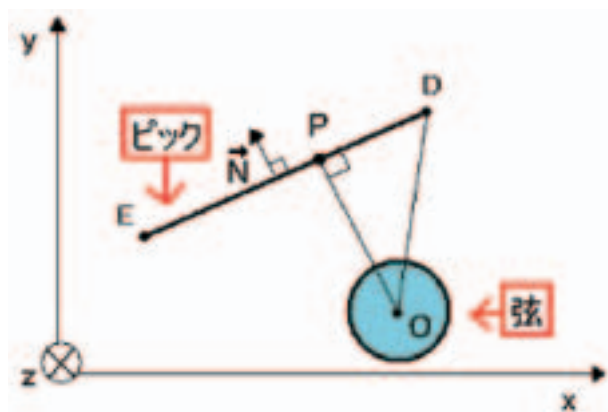


図3 弦とピックの断面図

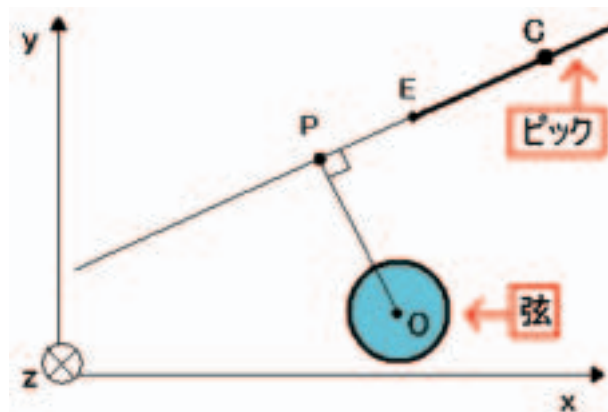


図4 点Pが線分DE外にある状態の弦とピックの断面図

4.2 弦の反発係数の算出

弦の反発係数kを算出する。弦の反発係数とは、弦を押さえたときに弦が元の位置に戻ろうとする力の係数である。弦の全長を ℓ_0 [m]、ピックで弾く弦の弦長を ℓ [m]、弦の線密度を ρ [kg/m]、弦の固有振動数を f [Hz]とおく。これらの値からピックで弦を弾く位置と弦の張力を求め、そこから弦の反発係数を求める。ただし、弦長と固有振動数は音階変更を行うと変化する。

一般にギターにおいて、弦をピックで弾く位置は、弦のブリッジから弦の全長の約1/8の距離pの位置となる。ここでは、弦を弾く位置を、ブリッジから弦の全長の1/8の距離の位置に固定している。ブリッジから弦を弾く位置までの距離pと弦の張力Tは、(3)、(4)式で与えられる。

$$p = \ell_0/8 \text{ [m]} \dots\dots\dots (3)$$

$$T = (2\ell f)^2 \rho \text{ [N]} \dots\dots\dots (4)$$

両端を固定された弦に力を加えるとき、弦に力を加える位置によって、弦を一定距離だけ動かすときに必要な力の大きさが変わってくる。力を加える位置から弦の両端までの距離を、それぞれp、 $\ell - p$ とすると、弦の反発係数kは(5)式で求められる。また、弦の変位距離をaとしたときの弦の状態を図5に示す。

$$k = T(1/p + 1/(\ell - p)) \dots\dots\dots (5)$$

ここで求めた弦の反発係数kを使って反発力を求める。

触覚デバイスのコントローラを操作する（ピックを動かす）ことにより、図5に示すように弦がピックに押さえ込まれることによって移動する距離aが常に変化するの、一定時間ごとにaを算出して反発力を求め、コントローラにその力をフィードバックさせる。

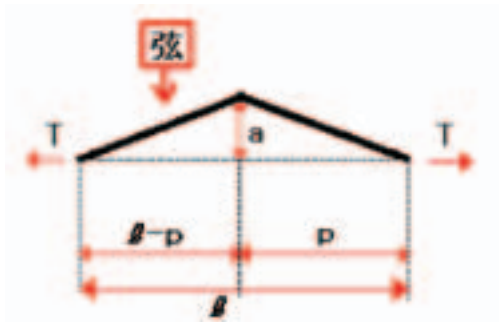


図5 ピックにより押さえ込まれた状態の弦

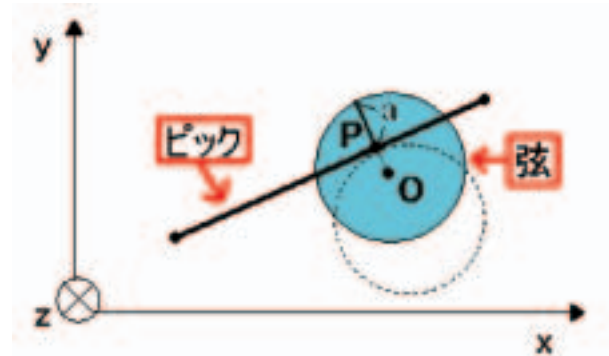


図6 弦とピックの接触時の断面図

4.3 反発力の計算

弦とピックの接触判定を行い、接触していると判定すると反発力の計算を行う。接触している状態とは、コンピュータ内では弦とピックが交わっている状態であり、その状態を図6に示す。反発力を \vec{F} 、弦の反発係数を k 、弦の中心点を点 O とし、点 O に最も近いピック上の点を点 P とする。ピックが弦と接触してから点 O に近づいた距離、つまり弦の変位距離を a とし、ベクトル \vec{OP} の単位ベクトルを \vec{N} とする。弦の半径を r としたとき、弦の変位距離 a は(6)式により求める。図6のような状態でのピックにかかる反発力 \vec{F} を(7)式により求める。

$$a = r - |\vec{OP}| \dots\dots\dots (6)$$

$$\vec{F} = (a \cdot k) \vec{N} \dots\dots\dots (7)$$

しかし、この計算方法ではピックが弦の中心を通過すると反発力が逆方向に働いてしまう。そこで、図7のような領域設定を行う。図7のように弦の中心を基準にA, B, C, Dの4つの領域を設定したとき、1フレーム前に点 P が位置していた領域の、点 O に対して対称の領域に点 P が入った場合、点 P がその領域を出るまで反発力の計算を(8)式に変更する。

$$\vec{F} = -((r + |\vec{OP}|) * k) \vec{N} \dots\dots\dots (8)$$

これにより、ピックが弦の中心を通過しても、反発力が逆方向に働くことはなくなる。このときの状態の断面図を図8に示す。

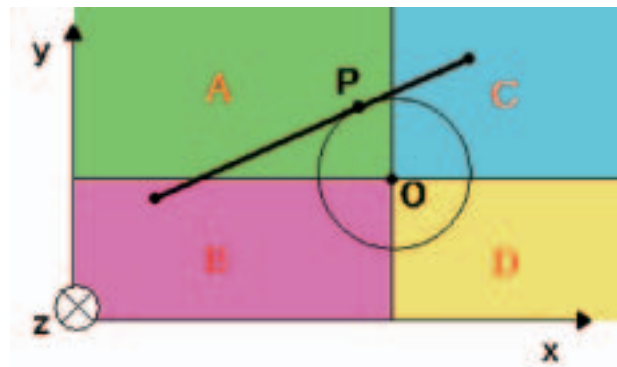


図7 領域設定

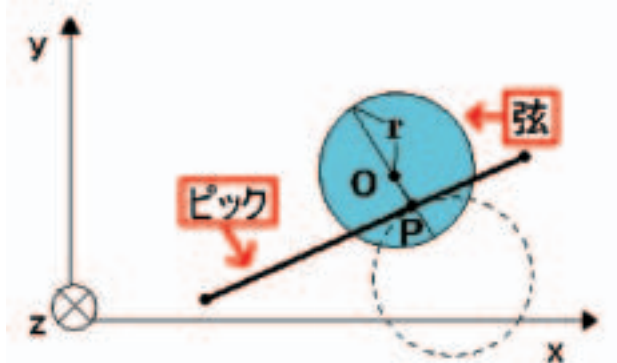


図8 ピックが弦の中心を通過したときの断面図

5. 音データの作成

弦をピックで弾いたとき、弦を弾く強さと各音階のギターの音波データから、出力する音波データを作成し、そのデータから WAVE 音源を作成して音をスピーカーに出力する。ギターを弾く強さによって音の強弱が決まり、ギターの音波データによって音階が決まる。ギターの音波データは、音階ごとに音波データファイルを作成して、そこからデータを読み込む。ギターを弾く強さと音の減衰率から再生時間を計算し、それを元に出力する音波データのサイズを求める。これらのデータから出力する音波データを作成する。

5.1 減衰振動

減衰振動とは時定数 α で音波の大きさが減少していく振動のことであり、ギターの音波も減衰振動波として扱う。つまり、ピックで弦を弾いたときに出力される音の大きさは徐々に小さくなっていく。実際には音の高さや弾き方により α は変化するが、バーチャルギターシステムでは減衰率をすべての音波に対して $\alpha = 1.0$ 秒とする。すなわち、最初の振幅の大きさをAとすると、t秒後の減衰した振幅Yは(9)式で表せる。縦軸が振幅、横軸が経過秒数とし、A = 1.0としたときの正弦波の減衰振動を図9に示す。青色の線が振幅の大きさを表し、紫の線が波形を表している。

$$Y = A \cdot \exp(-1.0 \cdot t) \dots\dots\dots (9)$$

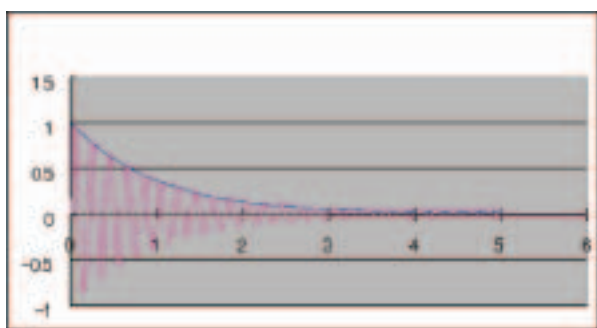


図9 減衰振動

5.2 再生時間の計算

音波データのデータサイズを計算するために音の再生時間を計算する必要がある。音波は減衰振動するので、音波の振幅が0になるまでの時間が再生時間となる。しかし、ある程度振幅が小さくなると人間の耳ではほとんど聞こえなくなる。そこで、振幅がある一定の大きさ A_0 以下になるまでの時間を再生時間 t_0 とした。

弦を弾く強さは、弦の変位距離aで表される。弦を弾く強さの最大値、つまり最大音量が出る弦の変位距離をV_MAXとする。弦を弾く強さは、量子化した量（ここでは16ビット）で表しているの、振幅の最大値は32767となる。再生開始からt秒後の弦の振幅をAとしたとき(10)式が成り立つ。

$$A = \exp(-t) \cdot (a/V_MAX) \cdot 32767 \dots\dots\dots (10)$$

tを1秒ずつ増やしながらかAを計算し、Aが閾値 A_0 以下になったときの時間tを再生時間とした。現在、振幅の閾値 A_0 は、実際に再生してみてほとんど音が聞こえなかった値の20としている。

5.3 データサイズ計算

音波データを作成するためにデータサイズを求める必要がある。データサイズは、音波データを記憶する配列のデータ領域確保と、WAVE音源の作成に必要となる。サンプリング周波数を f_s としたとき、データサイズは(11)式で求められる。

$$\text{Datasize} = t_0 \cdot f_s \cdot \text{チャンネル数} \cdot \text{量子化ビット数}/8 \dots\dots\dots (11)$$

チャンネル数とは音出力のチャンネル数のことで、モノラルだと1、ステレオだと2となる。今回はモノラル設定とした。

サンプリング周波数とは、音波などをデジタルデータにする際に、1秒間に採取する標本の頻度のことである。サンプリング周波数は採取対象音波の周波数の2倍以上でなければ、採取対象音波と異なる周波数の波形となる。現在バーチャルギターシステムでは1046Hzの下の音が最も高い周波数の音なので、サンプリング周波数は2092Hz以上である必要がある。サンプリング周波数が大きいほど、より高音質の音になるが、その分計算に時間がかかるためにシステムの動作が遅くなる。音楽CDなどのサンプリング周波数は44.1kHzとなっているが、このサンプリング周波数の音波データで計算を行うと、本システムの動作が遅くなって満足に機能しなかった。そこで、本システムの動作に影響がなく、なるべく高いサンプリング周波数を調べた結果12kHz位が妥当だということが分かった。本研究では計算のしやすさを考慮に入れてサンプリング周波数を11.025kHzとした。

5.4 音波データ作成

音を1秒間再生するのに必要なデータ数は、サンプリング周波数 f_s に等しくなる。すなわち、音波データは $1/f_s$ 秒間隔で、減衰を考慮した波形の大きさを配列に格納することになる。その配列の添え字番号をiとしたときの音波データは(12)式で与えられる。

$$W[i] = \exp(-1.0(i/f_s)) \cdot a/V_MAX \cdot w[i] \dots\dots\dots (12)$$

ここでw[i]は、ギターの減衰していない音波の元データを記憶している配列である。総データ数DNumは(13)式で求める。

$$DNum = t_0 \cdot f_s \dots\dots\dots (13)$$

(12)式の計算を、iが0からDNum-1になるまで行えば、配列Wに再生する音波データが作られる。これにより作

成した音波データから WAVE 音源を作成し、ギター音を出力する。

6. バーチャルギターシステムの機能

バーチャルギターシステムの機能、および実行画面における表示についての説明を行う。また、バーチャルギターシステムで使用するコード割当てファイルを作成するための、コード割当てプログラムについての説明も行う。

6.1 音階変更

ギターの音階はフレット部の弦を押さえることにより変更するが、弦を押さえる動作をキーボードのキーを押さえる動作に対応させた。しかし、6つの弦の各フレットを押さえる動作全てをキーボードのキーに割当てするにはキーの数が足りない。そこで、音階の変更をド、レ、ミなどの単音から C、C#、D などのコード単位で変更することにした。コードとは和音のことである。

ある一つのキーを押すと、そのキーに対応したコードの音階を各弦に設定する。キーを押している間にその弦を弾けば、弦に設定された音階の音が出力される。一般的に、一つの曲に使用するコードの数はあまり多くないので、使用するコードのみを複数キーに割当てて使用する。現在のシステムは、「z・x・c・v・a・s・d・f・q・w・e・r」の12個のキーにコードを設定できる。

6.2 システム実行画面

バーチャルギターシステムの実行画面を図10に示す。実行画面の画面構成は、(1)ギターを弾く部分を拡大表示する**メイン画面**、(2)ギターのコードとキーボードのキーの対応を表示する**コード対応表示画面**、(3)フレット部分を拡大表示している**フレット部表示画面**の3つの画面を一つのウィンドウに表示している。



図10 バーチャルギターシステム実行画面

(1) メイン画面

図10の右側がメイン画面で、弦を弾く部分を拡大して表示している。その拡大図を図11に示す。触覚デバイスのコントローラと連動させたピックで弦を弾くことによって、実際にギターを弾いているような触感が得られ、音出力される。



図11 メイン画面

(2) コード対応表示画面

図10の左上がコード対応表示画面で、音階変更の際に押すキーボードのキーとギターコードとの対応が表示されている。その拡大図を図12に示す。コード対応表示画面のキー配置はコンピュータのキーボードと同じキー配置で対応している。四角い枠が一つのキーを表し、その中に設定されたコードを赤い文字で表示している。そして、枠の左下に対応するキーのアルファベットが記述されている。画面下には現在設定されているコード割当てのスロット番号が表示されている。コード割り当ての設定を記録したファイルがスロット0～9にあり、これらはポップアップメニューから切り替えることができる。

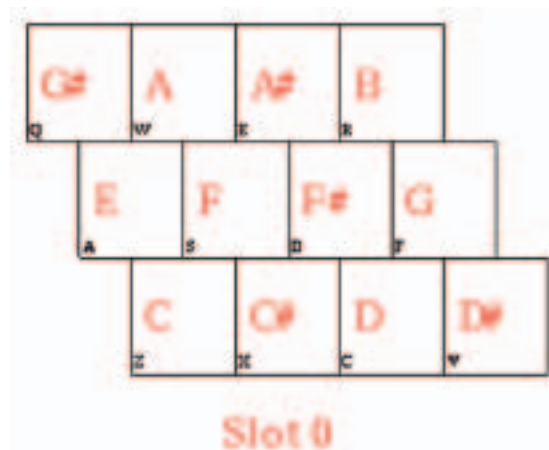


図12 コード対応表示画面

(3) フレット部表示画面

図 10 の左下がフレット部表示画面で、フレットの部分を拡大して表示している。その拡大図を図 13 に示す。弦の部分に表示された赤い球体オブジェクトは、音階変更のキーを押しているときに表示され、現在、弦のどの位置を押さえているかを表す。コードに対応したキーを押さえると、そのコードで弦を押さえる位置に赤い球体オブジェクトが表示される。また、この画面の左側に現在押されているコードの文字が表示される。

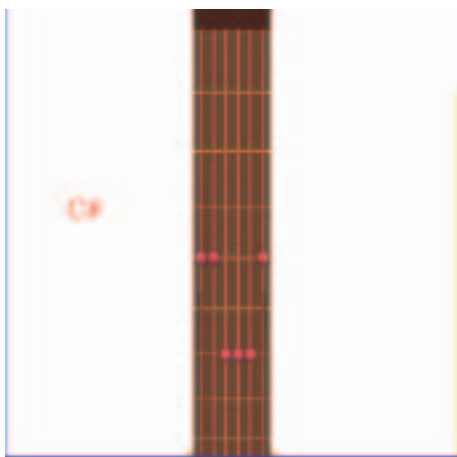


図 13 フレット部表示画面

6.3 コード割当て

さまざまな曲に対応するため、キーボードのキーに自由にコードを割当てた設定ファイルをいくつか作り、登録しておく。弾きたい曲に対応するコードの割当てを、登録したファイルから読み込むようにした。コード割当て設定ファイルの作成は、バーチャルギターシステムとは別のプログラムで行う。

コード割当てプログラムの実行画面を図 14 に示す。メニューの [ファイル] からコード割当て設定ファイルの読み込みと保存、プログラムの終了が行える。また、メニューの [コード] から [コード設定] を選ぶと、図 15 のようなコード割当て画面が表示される。ここでコードの割当てと、コード割当て設定の名前の入力を行う。これらの設定を、実行画面のメニューから任意のスロットに保存すれば、コード割当て設定ファイルが作成される。

図 15 のように各キーにコードを割当てた後、[OK] ボタンを押すか、または、図 14 の [ファイル] から [開く] を選んで既存のコード割当て設定ファイルを読み込むと、図 14 の実行画面中央に赤い文字でコード割当て情報が表示される。上部にある《設定名 []》という表示の括弧内に、このコード割当て設定ファイルの設定名が表示される。その下に、キーに割当てられたコードが、コード対応表示画

面と同じ配置で表示される。

コード割当て設定ファイルの変更は、バーチャルギターシステムのポップアップメニューから行える。バーチャルギターシステムの実行画面上で右クリックするとポップアップメニューが開き、コード割当て設定ファイル作成時に入力した設定名が表示される。そのときの状態を図 16 に示す。ここで、[曲名 0] を選べば [曲名 0] の、[曲名 2] を選べば [曲名 2] のコード割当て設定が読み込まれて、コード対応表示画面にそのコード割当てが表示されます。

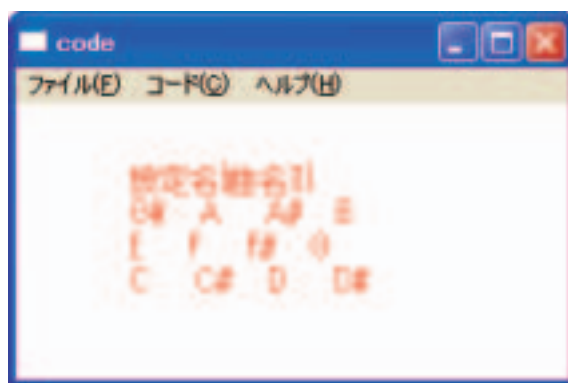


図 14 コード割当てプログラム実行画面

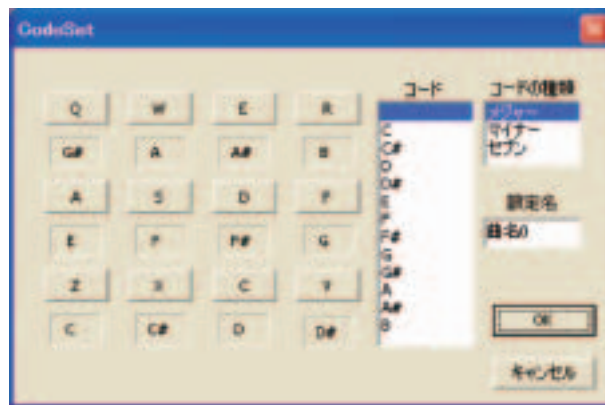


図 15 コード割当て画面

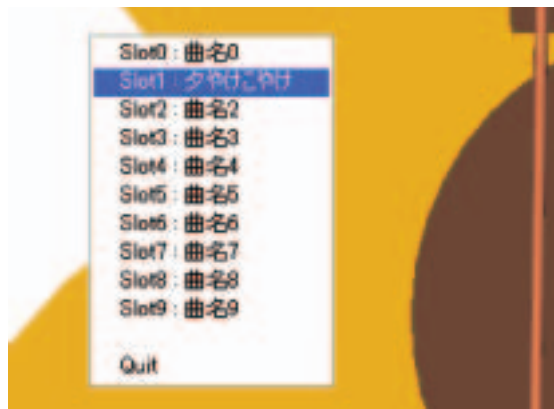


図 16 ポップアップメニュー

6.4 ピックの傾き変更

弦を弾きやすくするためにピックに傾きをつけているが、触覚デバイスのコントローラに付いている青いボタンを押すとピックの傾きが図3のとときは逆になる。ピックの傾きが逆になったときの弦とピックの状態を図17に示す。これは、ギターを弾く際に、普通の弾き方とは逆方向に弾くことがあるために実装した。

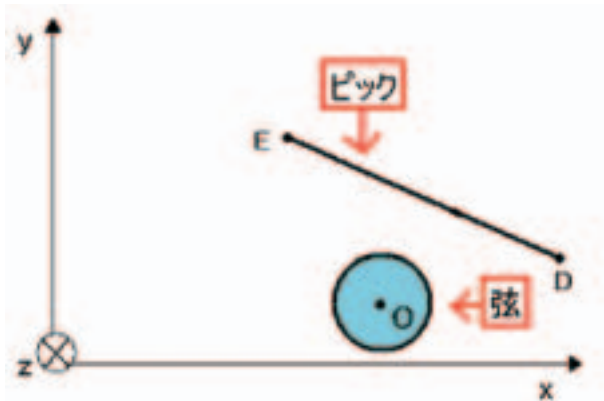


図17 ピックの傾きが逆になったときの断面図

7. おわりに

本論文では触覚デバイスを用いたバーチャルギターシステムについて述べた。ギターを弾いたときの触感とギター音の発生を実現した。触感を表す反発力はある程度再現できたが、細かい触感の再現までは至っていない。また、簡

単な曲なら演奏することが可能であるが、コードの割り当てや操作方法などの、使い易さの面でまだ改良を行っていく必要がある。

今後の課題としては、実行画面におけるメイン画面のカメラ位置を変更可能にすることである。これは、使用者が最も弾きやすい向きや角度に、自分でバーチャルギターの向きや傾きを調節できるようにするためである。また、ギターを弾く触感をよりリアルに再現するために、物質の質感の再現方法を考える必要がある。

文 献

- 1) 中尾 恵・小山 博史・小森 優・松田 哲也・高橋 隆, 「心臓手術シミュレーションに要する心拍動の視覚及び力覚提示に関する研究」, 日本バーチャルリアリティ学会論文誌, VOL.7 No.3, pp.413-420, (2002)
- 2) 嵯峨 智・梶本 裕之・館 日章, 「Haptic Video—インピーダンス情報を活用した作業環境記録再生手法」, 日本バーチャルリアリティ学会論文誌, VOL.7 No.3, pp.413-420, (2002)
- 3) 田村 真晴・山下 英生, 「触覚デバイスを用いたバーチャルギターの開発」, 平成19年度電気・情報関連学会中国支部第58回連合大会講演論文集, p.298, (2007)
- 4) 田村 真晴・山下 英生, 「触覚デバイスを用いたバーチャルギターの開発(2)」, 平成20年度電気・情報関連学会中国支部第59回連合大会講演論文集, p.483, (2008)