

液晶タブレットを用いた音と図形の生成に関する検討

出口幸子, 古田直也
近畿大学工学部

概要： 液晶タブレットをユーザインタフェースとして音と図形を生成するための基礎的な課題を研究した。筆圧の時間変化を計測する実験を行った結果、200～500[ms]の筆圧値を取ることが望ましいことが分かり、100[ms]時の筆圧値を用いるためには補正が必要なことが示された。一方、正弦波の合成により音を生成する方式について検討した。音の大きさ、音高、音色を、それぞれペンの筆圧、Y座標、X座標で制御し、音色については、ユーザが倍音や減衰を指定できるようにした。音の生成と同時に基本図形を描画する方式についても検討した。

A Study on the Generation of Sounds and Figures using LCD Tablet

Sachiko Deguchi, Naoya Furuta
School of Engineering, Kinki University

Abstract: This paper describes a research on the basic problem to generate sounds and figures using LCD tablet as a user interface. The result of the experiment on pen pressure shows that the appropriate time to get pen pressure is 200 to 500[ms], therefore, the adjustment is needed to use the pressure at 100[ms]. Next, this paper shows how to generate digital sounds by synthesizing sine waves. The pen pressure controls the volume, the y-coordinate of pen position controls the pitch, and the x-coordinate controls the timbre. Users can specify the ratio of harmonics and the decay function. This system can also display basic figures with sounds.

1. はじめに

近年普及しつつあるタブレットは、マウスとは異なり、絶対座標を用いた直感的な操作が可能である。特に液晶タブレットは画面に直接入力することで自然な操作ができる。そのため、タブレットや液晶タブレットは現在種々の分野で実用的に使用されている。一方、研究においては、文字や数式の入力方式の研究、ユーザインタフェースの研究[1]、等が行われている。また、音楽インタフェースとしても使われるようになって来ており、例えばMAX[2]を使った作品製作に用いられている。筆者らは、液晶タブレットで音を生成するための基礎的な課題として、筆圧の計測と基本的な音の制御について研究し、概要を報告した[3]。本稿では、その詳細と検討事項、および基本図形の描画について述べる。

2. 液晶ペンタブレット

本研究では、ワコムの液晶ペンタブレット CintiQ C-1500X を使用している。

2. 1. ペン情報

タブレットの座標は左下が(0, 0)になる。右上の座標は CintiQ の分解能から(6143, 4607)とする。CintiQ の筆圧は 512 レベルだが、タブレットドライバから情報が渡されたときの筆圧の最大値は 1023 (CintiQ の筆圧の 2 倍-1, 0 の場合は 0) である。今回は USB 接続としたので、93 点/秒の筆圧情報を得る。筆圧は、入力荷重そのままの値ではなく、設定が可能である。本研究ではデフォルトの設定 (入力荷重 8 を出力筆圧 0 とし、ほぼ比例の関係) を用いている。なお、本研究では傾き情報は使用していないが、CintiQ では、傾きは 900(ペン先スイッチ)と-900(テールスイッチ)

のみである。

2. 2. タブレットプログラミング

開発環境は Microsoft Windows XP 上で Microsoft Visual C++ 6.0 を使用している。タブレットインタフェースとして、米 LCS / Telegraphics 社が開発したタブレット用の API 群である WinTab を使う[4]。

入力を知るには、タブレットから WT_PACKET メッセージを送信させる方法を使用する。ウィンドウプロシージャが WT_PACKET メッセージを受信したら WTPacket 関数を使い、バッファを取り出す。本研究では、座標、筆圧、およびパケット発生時間を取得している。

本研究で使用した WinTab 関数を以下に記す。

- WTInfo 関数： タブレットの分解能やサポートされている機能などを取得する。
- WTOpen 関数： タブレットをオープンする。戻り値でタブレットを識別する。
- WTClose 関数： タブレットをクローズする。
- WTPacket 関数： バッファを一つ取り出す。戻り値はパケットが見つかった場合 0 以外の値となる。
- WTPacketsGet 関数： バッファを任意の数だけ取り出す。戻り値はバッファに複製されたパケットの数となる。

2. 3. 筆圧の計測

パケットの発生時間を取得する実験をした結果、正確に 1 秒間に 93 回サンプリングしていることが分かった。約 10[ms]に一回サンプリングしていることになるが、目的の筆圧に到達するまでに時間が掛かる。そこで、どの時点の筆圧を用いればよいかを検討するため、パケットの発生順に筆圧を取るプログラムを作成し、筆圧を計測する実験を行った。

2. 3. 1. 筆圧の平均値の時間変化

入力の目的筆圧を 512 (最大値の半分) となるように指定し、入力開始時点から 3 秒間、サンプリング時間毎に筆圧を計測する実験を行った。実験用プログラムは、筆圧値 512 のラインを表示しておき、入力中の筆圧値を順次表示することによ

り、ユーザが筆圧を制御して目的筆圧に合わせるようにしている。

図 1 は 400 回試行し、時間と平均筆圧の関係を示したグラフである。その内の 100 回試行のグラフを作成しても、ほぼ同じ形になる。100[ms] (97[ms], 10 点目のサンプリング), 200[ms] (204[ms], 20 点目), 300[ms] (301[ms], 29 点目), 400[ms] (398[ms], 38 点目), および 500[ms] (505[ms], 48 点目) 時点の筆圧の平均値を、400 回試行と 100 回試行で比較すると表 1 のように大きな違いはない。よって平均値を得るには 100 回試行で十分と考えられる。

同実験を異なる 3 名が 100 回試行し、筆圧の時間変化をグラフにしたところ、人によって入力の癖が若干あることが分かった。しかし、何れも約 500[ms]で目的の筆圧に達しており、また、約 200[ms]位からグラフの傾斜が緩やかになっている。このことから、理想的には 500[ms]の筆圧を取ればよいが、現実的にはアプリケーションに応じて 200[ms]~500[ms]の筆圧値を取ればよいことが分かる。しかし、対話的インタフェースにおいて、200[ms]は、ユーザにとって、遅れが知覚される時間である。

2. 3. 2. 筆圧の分布と分散・標準偏差

筆圧の分布を知るためにヒストグラムを作成した。400 回試行の 100[ms], 200[ms], 300[ms], 400[ms], および 500[ms]時点におけるヒストグラムを作成したところ、単峰で左右対称に近い形になった。

一方、ばらつきの尺度として分散と標準偏差を求めた。表 2 は 400 回試行、およびそれを 4 分割した 100 回試行の各々について、100[ms], 200[ms], 300[ms], 400[ms], および 500[ms]時点の筆圧の標準偏差を求めた結果である。

表 2 から分かるように、400 回試行も 100 回試行も同様に、100[ms]~300[ms]の範囲では標準偏差の違いがなく、400[ms]以上では標準偏差が減少して行く。別に行った 400 回試行でも同様の結果になった。これは、ヒストグラムからも読み取れる傾向である。

表 2 に示すように、ペン入力後のどの時点 (100, 200, 300, 400, 500[ms]) においても、標準偏差は、

最初の100回試行と次の100回はあまり違いがなく、その次の100回と最後の100回では順に減少している。これは、被験者がペン入力操作に慣れて行く過程を示している。しかし、同一被験者がこの実験の前に行った400回試行では、このような標準偏差の減少は見られなかった。この結果は、ユーザインタフェース実験における再現性の難しさを示している。

2. 3. 3. 筆圧の感覚

先に述べた実験では、時間と筆圧との関係を分析するために、目的の筆圧を指定（表示）してユーザがそれに合わせて入力する方法を取った。これは、不自然な入力方法であるし、実際にペンを使う場合、指定された筆圧を入力することは稀であろう。そこで、自分が弱い、中位、強いと思う筆圧で入力したときの時間と筆圧の関係を計測する実験を行った。3名がそれぞれ100回試行した筆圧の平均値をグラフにしたものを図2、3、4に示す。この結果から、目的筆圧を指定しない場合も、100回の平均値を取ると、一定の値に収束して行くことがわかった。

被験者Aについて、筆圧を指定した時の図1と指定しない時の図2を比較すると、図1同様、図2の3種類の筆圧は、それぞれ約200[ms]からグラフの傾きが緩やかになっている。また、他の2名についても同様の結果になっている。この結果は、目的の筆圧を指定しない場合も、指定する場合と同様、200[ms]以降の値を取るとよいことを示している。

3. 音の生成

ここでは、本研究において液晶タブレットを用いて音を生成する方式について述べる。

3. 1. 正弦波の合成

音を特徴付ける要素として大きさ、高さ、音色があり、それぞれ、物理的には振幅、基本周波数、波形に対応している。本研究では、正弦波の合成で音を生成する。合成の基本式を以下に示す。

$$W(t) = P \times k \times g(t) \times \\ \{ A_1 \times \sin(2\pi f t) + \dots \\ + A_j \times \sin(2\pi j f t) + \dots \}$$

ここで、Pは筆圧に補正値を足した値、kは倍音成分による合成波の振幅の増加を調整するための係数、g(t)は音の減衰を表す関数、A1は基本周波数の振幅、Ajはj倍音の振幅、およびfは基本周波数を表す。

3. 1. 1. 振幅

本研究では、振幅を筆圧に比例させるための検討を行った。音を鳴らす場合、初めの入力から200[ms]後の筆圧を用いると遅れが目立つので、100[ms]後の筆圧を取ることにした。但し100[ms]時点では前節に示したように、筆圧の平均値274に対し、標準偏差が98であり、筆圧が十分に取れない場合がある。そこで、100[ms]時の筆圧値の標準偏差の約2倍である200を計測値に足して補正し、その値をPとして音の生成に用いることにした。一方、倍音や他の成分の正弦波を足して行くと振幅が大きくなって行くので、音量を揃えるための係数kとして、 $C/\sum A_j$ を用いる（Cは定数）。また、ユーザインタフェース上で、音量調整の係数を入力できるようにしている。

上記のような補正値を足す方法では、以下の問題がある。

- (1) 筆圧が、本人が意図していたよりかなり小さい場合には、補正値を足した効果がない。
- (2) 補正値を大きくすると、筆圧に対応した強弱が付き難い。

そこで、現在、以下の方法を検討している。

- (a) 生成する音の大きさを、大、中、小の3レベルにする。ユーザに予め大、中、小と思う筆圧を入力してもらい、そのデータを基に、入力値が何れであるかを推定して音を生成する。しかし、図2、3、4から分かるように、100[ms]時点の入力値を3レベルに分類することは難しく、2レベルに限ることも検討する。
- (b) 筆圧値を音の大きさに用いず、音は一定の大きさとする。筆圧値は、ばらつきがあること活かし、音色を変化させること等に利用する。この場合、筆圧値を得る時間を100[ms]より小さくし、音楽演奏に適した時間とする。

3. 1. 2. 基本周波数

本研究では、画面のY軸方向に3オクターブ

の幅で音階を構成する音（ド，レ，ミ，ファ，ソ，ラ，シ）を並べている．ユーザが画面上の音をペンで指定することで基本周波数が決定する．音階上の各音の周波数は音律により規定される．本研究では，平均律とピタゴラス音律の何れかをユーザが選択できるようにしている．音階を変更する機能も追加する予定である．

3. 1. 3. 波形

(1) 倍音とその他の成分

倍音は 10 倍音まで合成でき，それぞれの音量（基本周波数に対する相対的な割合）を設定できる． i 倍音の i は整数のみでなく，小数第 4 位まで指定できる．例えば，基本周波数に $1.26(2^{4/12})$ 倍の音， $1.5(2^{7/12})$ 倍の音を合成すれば，3 和音が生成できる．また，基本周波数に加え 1.01 倍の音などを指定すれば，うなりを表現することができる．

(2) 減衰

合成した波形に減衰関数を掛けることにより減衰を実現できる．本研究では以下の関数を検討し，ユーザインタフェースで指定できるようにしている．

- (a) $g(t)=1$ 減衰なし．
- (b) $g(t)=\exp(-at)$ 指数関数的に減衰．
- (c) $g(t)=1$ [$0 \leq t \leq b$]， $g(t)=1-a(t-b)$ [$b \leq t \leq b+1/a$]
b まで減衰なしで，b から $-a$ の傾きで減衰．
- (d) $g(t)=1$ [$0 \leq t \leq b$]， $g(t)=\exp\{-a(t-b)\}$ [$b \leq t$]
b まで減衰なしで，b から指数関数的に減衰．
- (e) $g(t)=1$ [$0 \leq t \leq b$]， $g(t)=\sqrt{1-(t-b)/a}$ [$b \leq t \leq b+a$]
b まで減衰なしで，b から放物線のカーブで減衰．

3. 2. ユーザインタフェース

音を生成するために作成したユーザインタフェースを図 5 に，設定用ダイアログを図 6 に示す．音の大きさ（振幅），音高（基本周波数），音色（波形）は，タブレット入力ではそれぞれ筆圧，Y 座標，X 座標に対応している．図 5 に示す 22 行 5 列のセルをペンで指定することでそれぞれ違う音を出すことができる．行は音高，列は音色に対

応している．音高は，画面上に並んだ 7 音音階の音をペンで選択することで決まる．音色は各列の下に A, B, C, D, E のボタンを押してダイアログを表示して設定する．各ボタンの下のエディットボックスに音色を識別する名前を入れることができる．

図 6 に示すダイアログの領域 A では，音の長さをエディットボックス，音律をラジオボタン，減衰関数をコンボボックス，および減衰係数をエディットボックスで設定する．領域 C のエディットボックスで倍音あるいは他の成分を指定し，領域 B でそれらの振幅の割合を設定する．

3. 3. WAVE データ

(1) WAVE データの作成

本研究で試作したプログラムでは，入力がある度に計算して WAV ファイル形式のデータを作成し，メモリ上のサウンドイメージとして音を鳴らす[5]．WAV ファイルは，フォーマット情報を格納するフォーマットチャンク，および WAVE データを格納するデータチャンクから構成される．本研究の WAVE データは，前述した方式で正弦波を合成し，それをサンプリングして作成する．

入力に対して毎回 WAVE データを作成するので，できるだけ負荷を少なくしたい．本システムで扱う基本周波数の最大は 1047Hz で，10 倍音まで合成するので周波数の最大は 10470Hz である．その 2 倍は 20940Hz であるから，サンプリングレートを 24KHz とする．

(2) WAVE データの再生

WAVE 再生の手順を以下に示す．

- ① waveOutOpen 関数でデバイスをオープンする．
- ② waveOutPrepareHeader 関数でデバイスとバッファを結びつける．
- ③ waveOutWrite 関数でバッファを再生デバイスに書き込む．
- ④ waveOutUnprepareHeader 関数でヘッダを開放する．
- ⑤ 再生が終わっていれば waveOutClose 関数で再生デバイスをクローズする．

本研究では同時発音数（デバイス数）を 10 までとしている．なお，マルチバッファリングを使

えば、一つのバッファの音が鳴っている間に次のバッファを書き込むことで、音を途切れなく再生させることができる。またバッファを小さくすれば負荷を分散できる。そこで、本研究ではトリプルバッファリングを用いている。

4. 図形描画

ここでは、本研究における基本図形の描画について説明する。

4. 1. ユーザインタフェースの描画

図5に示すユーザインタフェースは、以下のように描画している。

(1) ウィンドウの表示： WinMain 関数内でウィンドウクラスの設定、ウィンドウの作成、および表示を行う。まず、WNDCLASSEX 構造体（ウィンドウクラス構造体）を設定し、RegisterClassEx 関数で Windows に登録する。次に CreateWindow 関数でウィンドウを作成し、ShowWindow 関数で非クライアント領域の表示、UpdateWindow 関数でクライアント領域の描画をする。

(2) 文字列、ボタン、エディットボックスの表示： CreateWindow 関数を用い、パラメータに、STATIC を指定して文字列、BUTTON を指定してボタン、EDIT を指定してエディットボックスを作成している。

(3) その他の文字列の表示： wsprintf 関数と TextOut 関数を用いて文字列を表示している。

(4) 色つきセルと枠の表示： FillRect 関数を用いて横に長い四角を縦に 22 個、色分けして描画し、その上から MoveToEx 関数、LineTo 関数を用いて線を引いている。その周りに FillRect 関数を用いて、背景色の四角を四つ描画している。

4. 2. 音と同時の図形表示

本研究では、音と同時に図形を変化させる方式を研究している。今回は、図5に示すようにペン入力に対して、入力座標を中心とし、筆圧に比例した半径の円を表示している。筆圧がかかっているセルと同じ色でソリッドブラシを作り、ペン入力の座標と筆圧（1/5 を半径とする）から、円に外接する正方形の左上と右下の座標を求め、

Ellipse 関数でそれらを指定して円を描画する。次のペン入力時に、色つきセルを再描画してから円を描画する。これは、最も基本的な描画であるが、図形表示がない場合に比べて、ユーザの受ける印象が強いことが分かった。

5. おわりに

液晶タブレットをユーザインタフェースとして音と図形を生成するための基礎的な課題を研究した。筆圧の時間変化を計測する実験を行い、次のことが明らかになった。

- (1) 200～500[ms]の筆圧値を取ることが望ましいが、この時間遅れは対話的インタフェースには適さない。
- (2) 筆圧の標準偏差は、100～300[ms]の範囲はあまり差がなく、400[ms]以上で減少して行く。
- (3) 筆圧の標準偏差は、100 回試行では実験の条件（主に被験者の慣れ）により大きく変わる。一方、正弦波の合成により音を生成する方式について、次のように検討した。
 - (i) 振幅はペンの筆圧に比例させる。100[ms]時点の筆圧を用いるために入力補正をする。振幅を3レベルに限ることも検討している。
 - (ii) 基本周波数は、画面上に Y 軸方向に並んだ音をペンで選択することで決める。
 - (iii) 波形は画面上の X 軸方向で変える。10 倍音までの成分、および倍音以外の成分も指定できる。また減衰も指定できる。

図形描画については、以下のように検討した。

- (I) 音を指定するセルを画面上に表示した。
- (II) ペン入力時に、ペンの位置を中心とし、筆圧に比例した半径の円を、セルの色で表示する。今後の主な課題は以下の通りである。
 - (a) ペンの筆圧情報の扱いとして、現在検討中の2点：予めユーザの筆圧データを登録して音の大きさを決める方法、および筆圧を音色変化に利用する方法を実現して評価する。
 - (b) ①図形と音の提示、②ペン入力、③図形と音の生成・変化、というサイクルを繰り返す対話的なインタフェースを検討する。なお、音の生成に MIDI も使用できるようにする。

参考文献

- [1] 加藤直樹, 中川 正樹: ペンユーザインタフェースの設計のためのペン操作性の検討, 情報処理学会論文誌, Vol.39, No.5, 1998.
- [2] IRCAM: <http://freesoftware.ircam.fr/>
- [3] 出口幸子, 高林明広: 液晶ペンタブレットを用いた音の生成システムの試作, 人工知能学会第 19 回全国大会, 2005.
- [4] LCS/Telegraphics:
<http://www.pointing.com/Resources.html>
- [5] 田辺義和: Windows サウンドプログラミング, 翔泳社, 2001.

表 1 : 筆圧の平均値(筆圧指定有:被験者 A)

	100[ms]	200[ms]	300[ms]	400[ms]	500[ms]
001-100 試行	284.7	416.0	450.4	476.6	492.4
001-400 試行	273.9	396.6	437.0	470.3	490.2

表 2 : 筆圧の標準偏差(筆圧指定有:被験者 A)

	100[ms]	200[ms]	300[ms]	400[ms]	500[ms]
001-100 試行	105.5	111.5	100.9	85.7	74.5
101-200 試行	111.8	121.7	109.3	94.8	83.8
201-300 試行	91.9	94.3	86.9	74.5	63.6
301-400 試行	76.5	73.4	67.1	57.7	46.6
001-400 試行	98.0	102.4	92.9	80.2	69.1

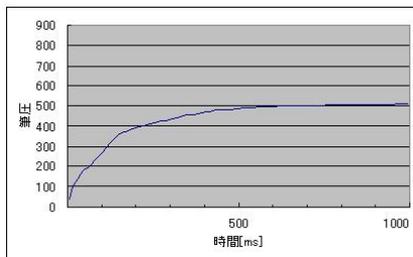


図 1 : 筆圧の時間変化(筆圧指定有:被験者 A)

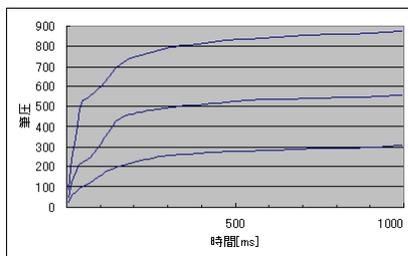


図 2 : 筆圧の時間変化(筆圧指定無:被験者 A)

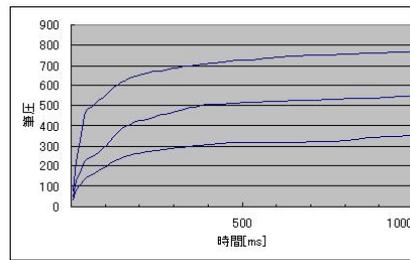


図 3 : 筆圧の時間変化(筆圧指定無:被験者 B)

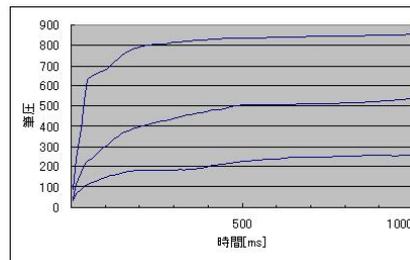


図 4 : 筆圧の時間変化(筆圧指定無:被験者 C)

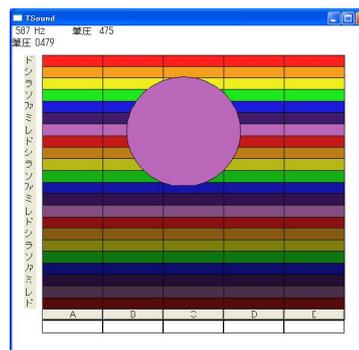


図 5 : 音の生成のユーザインタフェースおよび音と図形の同時生成



図 6 : 音色設定用ダイアログ