

平成24年度

筑波大学情報学群情報科学類

卒業研究論文

題目

キートップ上の音響センシングによる
引っ掻きジェスチャの認識

主専攻 ソフトウェアサイエンス主専攻

著者 黒澤敏文

指導教員 志築文太郎 高橋伸 田中二郎 三末和男

要 旨

本論文では、キートップ上の引っ掻き（引っ掻きジェスチャ）による入力手法と、その音響センシングについて述べる。本手法では、ユーザはキーボードのホームポジションに手を置いた状態から1本の指を用いて上下左右のいずれかの方向へ向けてキートップを引っ掻く。この動作により、ユーザがキーボード上のホームポジションから手の位置を動かさずに量と方向の入力を同時に行う事を可能とする。ユーザが引っ掻いた方向の認識に、ユーザが引っ掻きジェスチャを行う際に生じる音（操作音）を用いる事とした。また、入力量の認識のために、操作音の波形のピークの個数を数えた。操作音を取得するために、キーボードにピエゾピックアップを装着した。また、方向の認識に用いる特徴量として操作音の周波数スペクトルを用い、識別器としてサポートベクターマシンを用いた。

目次

第1章 序論	1
1.1 背景	1
1.2 目的	1
1.3 本論文の構成	1
第2章 関連研究と本研究の位置付け	3
2.1 キーボード上における入力を拡張する研究	3
2.2 ユーザの操作音を識別する事により入力を拡張する研究	3
2.3 キーボード上に発生する音を識別する研究	4
2.4 本研究の位置付け	4
第3章 Keyboard Clawing：キートップ上の引っ掻きジェスチャによる入力手法	6
3.1 キートップ上の引っ掻きジェスチャによる入力手法	6
3.2 本手法の特徴	6
3.3 認識方法のアイディア	9
第4章 実装	10
4.1 システム構成	10
4.2 操作音の取得	13
4.3 ノイズ処理	13
4.3.1 環境音のノイズ除去	13
4.3.2 打鍵音と操作音の区別	14
4.4 距離の識別	14
4.5 方向の識別	16
4.6 GUI ツール	17
第5章 応用例	21
5.1 音量調節機能	21
5.2 スクロール機能	23
5.3 フォントサイズ変更機能	24
第6章 評価実験	28
6.1 実験目的	28

6.2	実験環境	28
6.3	実験内容	28
6.4	実験結果と考察	30
6.4.1	距離の識別率	30
6.4.2	方向の識別率	31
6.4.3	操作音が検出されなかった割合	32
6.4.4	引っ掻きジェスチャ時における打鍵率	34
6.4.5	アンケート結果より	36
第7章	議論	38
7.1	距離の識別	38
7.2	方向の識別	38
7.3	ラップトップ型計算機に付属するキーボードへの適用	39
7.4	引っ掻きジェスチャに用いる指の本数の識別	41
7.5	机から受ける反射音の影響	42
第8章	結論と今後の課題	43
	謝辞	44
	参考文献	45
	付録 A 評価実験の際に用いた実験同意書及びアンケート用紙	48
	付録 B 評価実験のアンケートにて得られたコメント	51

図目次

3.1	各方向へのキートップの引っ掻き方	7
3.2	具体的なキートップの引っ掻き方の例	8
4.1	使用したピエゾピックアップ (SH710)	11
4.2	ピエゾピックアップを装着したキーボード	11
4.3	システムのフローチャート	12
4.4	ノイズ処理前後の波形	15
4.5	上：平滑化前の波形 下：平滑化後の波形	16
4.6	各方向に対する操作音の周波数スペクトル	17
4.7	距離の識別と方向の識別それぞれに用いる部分	17
4.8	GUI ツールの画面	20
5.1	GUI の操作による音量調節	22
5.2	WMP におけるショートカットキー (F8) による音量調節	22
5.3	Keyboard Clawing による音量調節	22
5.4	スライダのドラッグまたはホイールの操作によるスクロール	23
5.5	対応するキー (PageDown) によるスクロール	23
5.6	Keyboard Clawing によるスクロール	24
5.7	GUI の操作によるフォントサイズの変更	25
5.8	Word におけるショートカットキー (Shift + Ctrl + <) によるフォントサイズの変更	25
5.9	GUI の操作による文字の選択	25
5.10	ショートカットキー (Shift + 左キー) による文字の選択	25
5.11	Keyboard Clawing による文字の選択	26
5.12	Keyboard Clawing によるフォントサイズの変更	27
6.1	実験環境	29
6.2	本実験の際に被験者に提示される画面	30
6.3	指がキートップに触れる瞬間	31
6.4	下方向への引っ掻きジェスチャ (横から)	32
6.5	FN 率を距離別に表したグラフ	34
6.6	FK 率を距離別に表したグラフ	35

6.7	被験者が引っ掻きジェスチャの際に誤って打鍵したキーとその回数（方向別）	36
7.1	新たに提案する方向認識アルゴリズム	39
7.2	ラップトップ型計算機上にピエゾピックアップを配置した様子	40

表 目 次

3.1	各方向へ向けて引っ掻く際に触れる指とその部位（右手の場合）	9
4.1	実行する際に使用した計算機の仕様	10
5.1	各方向と距離の引っ掻きジェスチャに対応させる音量の調節量	22
5.2	各方向と距離の引っ掻きジェスチャに対応させるスクロールの向きと量 . . .	23
5.3	各方向と距離の引っ掻きジェスチャに対応させる機能	26
6.1	距離の識別率（%）	31
6.2	Per-User Test における方向の識別率（%）	33
6.3	Cross-User Test における方向の識別率（%）	33
6.4	<i>FN</i> 率の内訳（%）	33
6.5	<i>FK</i> 率の内訳（%）	35
7.1	本数識別の実験に用いる指	41
7.2	本数の識別率（1-4 本、%）	41

第1章 序論

本章では、まず研究の背景と目的を述べる。次に、本論文の構成を述べる。

1.1 背景

現在、キーボードは計算機の基本的な入力デバイスとして広く社会に浸透している。そのため、キーボード上における入力を拡張するための研究が盛んに行われている。研究の例として、キーボードの各キーに装着したタッチセンサを用いてキーボードをタッチパネルとして扱う研究 [BGV10]、カメラを用いてキーボード上に置かれたユーザの指の位置を検出し、キーボード上においてマルチタッチジェスチャを可能にする研究 [KM12]、ラップトップ型計算機に付属しているトラックパッドにユーザの指が触れている状態をショートカットのトリガとして用いる研究 [Rek03] 等が挙げられる。

また、音響センシングによりユーザのジェスチャを認識する研究が行われている。研究の例として、ガラス面に装着したピエゾマイクを用いてユーザがノックまたはタップした位置とその強さを識別する研究 [PLCH02]、壁または机に装着した聴診器を用いてユーザが机を引っ掻いたジェスチャを認識する研究 [HH08] 等が挙げられる。

本研究は、ユーザがキートップ上を引っ掻く操作（以降、引っ掻きジェスチャと呼ぶ）を行った際に生じる音（以降、操作音と呼ぶ）を解析することにより、キーボード上において上下左右のいずれかの方向と量の入力を可能とする。なお、ユーザが引っ掻きジェスチャを行う際、キーボードのホームポジションの位置周辺を始点とする。これにより、ユーザは引っ掻きジェスチャの際に、ホームポジションから手の位置を動かさずに素早く入力を行う事が出来る。

1.2 目的

本研究の目的は、ユーザがキーボードのホームポジションの位置から手を動かさずに素早く方向と大きさの入力を行う事が可能になる手法を提案する事である。

1.3 本論文の構成

第1章では、本研究の背景と目的を述べた。第2章では、関連研究と本研究の位置付けを述べる。第3章では、引っ掻きジェスチャによる入力手法である Keyboard Clawing を述べる。

第 4 章では、Keyboard Clawing の実装について述べる。第 5 章では、Keyboard Clawing の応用例を述べる。第 6 章では、Keyboard Clawing の評価実験を述べる。第 7 章では、Keyboard Clawing についての議論を行う。第 8 章では、本論文の結論と今後の課題を述べる。

第2章 関連研究と本研究の位置付け

本章では、本研究の関連研究と、本研究の位置付けを述べる。まず、キーボード上における入力を拡張する研究を挙げる。次に、ユーザが操作をする際に発生する音（操作音）を識別する事により入力を拡張する研究を挙げる。その後、キーボード上における操作音を識別する事によりユーザの入力を識別する研究を挙げる。最後に、関連研究に対して本研究を位置付ける。

2.1 キーボード上における入力を拡張する研究

キーボード上における入力を拡張する研究が行われている。

Rekimoto は、ラップトップ型計算機に付属しているトラックパッドにユーザの指が触れている状態を検出し、この状態をショートカットのトリガとして用いることにより、様々な入力を実現する手法である ThumbSense [Rek03] を示した。Kim らは、カメラを用いてキーボード上に置かれたユーザの指の位置を検出し、マルチタッチジェスチャを可能にする手法 Ghost Fingers [KM12] を示した。Fallot-Burghardt らは、キー上にタッチパッドを配置し、キーボード上においてトラックパッド操作を可能にする手法 Touch&Type[FBFS⁺06, FBKL⁺03] を提案した。

また、キーボードを加工する事により入力を拡張する研究も行われている。塚田らは、ポインティングを行う事が出来るキーボードである Pointing Keyboard[塚田 02] を開発した。Pointing Keyboard は、内部に格子状に赤外線を張った枠をキーボード上に重ねる事により、キーボード上におけるユーザの指の動きを認識している。Dietz らは、圧力により抵抗値が変わるピエゾ抵抗素子を各キーの内部に用いる事により、ユーザがキーを押した強さを検出するキーボードを開発した [DEWB09]。Block らは、キーボードの各キーの内部に装着したタッチセンサを用いて、キーボード上におけるユーザの指の動作を認識し、かつキーボード表面をスクリーンとして扱う事によりタッチパネルとして扱えるキーボード、Touch-Display Keyboard[BGV10] を開発した。

2.2 ユーザの操作音を識別する事により入力を拡張する研究

操作音を識別する事によりユーザの入力を拡張する研究が行われている。綾塚らは、複数の振動センサを用いる事により、壁または机などに対するユーザのノックによる個体音からノック

ク位置を検出する手法 [綾塚 01] を提案した。Paradiso らは、ガラス面に 4 つのピエゾマイクを装着し、ユーザがノックまたはタップした位置とその強さを識別する手法を示した [PLCH02]。

操作音を用いてユーザのタップ位置を検出するだけでなく、ユーザのジェスチャを認識する研究も行われている。堀らは、タッチパネル面に触れた指に同じ手の親指を押し付け、指の摩擦音を解析する事によりスポイトの吸い上げ動作と押し出し動作のメタファを入力に用いる手法である指スポ [堀 10] を提案した。Amento らは、リストバンドに装着したコンタクトマイクを用いて、ユーザの平面に対する指先ジェスチャを認識する手法を示した [AHT02]。Delye らは、手首またはかかとに装着した 2 つのピエゾピックアップによりユーザの骨を伝わる振動を検出し、ユーザのジェスチャを認識する手法 Hambone [DPPS07] を示した。Lopes らは、テーブルトップにマイクを装着し、ユーザがテーブルトップをタッチした際に生じた音を解析する事により、ユーザが行ったタッチの方法を識別する手法を提案した [LJJ11]。Murray-Smith らは、ユーザが爪で引っ掻いた際に生じた音を識別し、操作を行うデバイスである Stane [MSWHQ08] を開発した。

Harrison らは、操作音を用いる事により入力を拡張する研究をいくつか行なっている [HH08, HTM10, HSH11, HXH12]。Scratch Input [HH08] では、壁または机に聴診器を装着し、ユーザが机を引っ掻いた際に生じる振動パターンを解析する事によりユーザのジェスチャを識別している。Skinput [HTM10] では、生体音響信号を検出するウェアラブルセンサを用いる事により、ユーザがタップを行った身体の位置を検出している。TapSense [HSH11] では、サーフェス上におけるタップ音を解析する事により、ユーザがタップを行う際に使用した手の部位を識別している。AcousticBarcodes [HXH12] では、バーコード状に溝を刻み込んだ板をユーザが引っ掻く事により生じる音のパターンを解析する事により、バーコードの情報を読み取っている。

2.3 キーボード上に発生する音を識別する研究

Berger らは、攻撃の対象者が入力したパスワードを打鍵音から類推し、辞書攻撃に用いる手法を示した [BWY06]。Zhuang らは、打鍵音を用いてユーザがタイプしたキーを識別する手法を示した [ZZT09]。Kato らは、操作音を用いてユーザがキーボードの表面を大きく撫でた方向を識別する手法 Surfboard [KSI10] を示した。Surfboard は音の識別のために、操作音の周波数スペクトルの時系列データを用いてベイズ推定を行なっている。Surfboard が識別できるのは左右方向の 2 通りのみである。

2.4 本研究の位置付け

本研究は 2.1 節にて示した関連研究と同様に、手をホームポジションから動かす事なく、キーボード上における入力を拡張する。本研究はこれらの研究とは異なり、キーボード上における操作音を用いてユーザの入力を識別する。

また、本研究は 2.2 節にて示した関連研究と同様に、ユーザの操作の識別に操作音を用いる。本研究はこれらの研究とは異なり、ユーザの操作の識別に、ユーザがキートップを引っ掻

いた際に生じる音を用いる。

さらに、本研究は 2.3 節にて示した関連研究と同様に、キーボード上において生じる操作音を用いる事により、ユーザの入力を識別する。本研究は [BWY06, ZZT09] とは異なり、打鍵時でなくキートップを引っ掻いた際に生じた音を用いてユーザの入力を識別する。また、[KSI10] とは異なり、上下左右の 4 方向に対して入力量を伴った入力を行う事が出来る。

第3章 Keyboard Clawing：キートップ上の引っ掻きジェスチャによる入力手法

本章では、キートップ上の引っ掻きジェスチャによる入力手法である Keyboard Clawing を述べる。まず本手法の概要を述べた後、本手法の特徴を述べる。

3.1 キートップ上の引っ掻きジェスチャによる入力手法

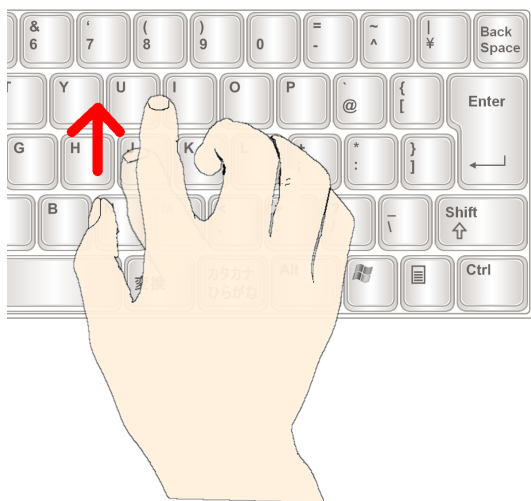
キートップ上の引っ掻きジェスチャによる入力手法を、具体的な操作手順とともに説明する。まず、ユーザはキーボードのホームポジションに手を置く。この状態から、ユーザは1本の指を用いて上下左右のいずれかの方向へ向けてキートップを引っ掻く。各方向へのキートップの引っ掻き方を図3.1に示す。ホームポジションの位置からの手の移動量を減らすために、上下方向へ向けて引っ掻く際には右手人差し指を、左右方向へ向けて引っ掻く際には右手親指を用いる事とした。さらに、ユーザがキートップを引っ掻いた距離を入力量とする。キートップを引っ掻いた距離は、ユーザがキートップを引っ掻いた際に通過したキー間の溝の個数である。すなわち、ユーザが「M」キートップ表面から「B」キートップ表面まで引っ掻いた場合、キートップを引っ掻いた距離は2となる（図3.2）。

3.2 本手法の特徴

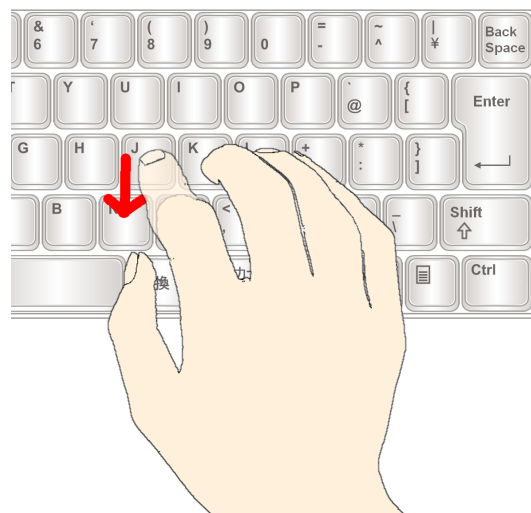
本手法は、ユーザが入力を行う際にホームポジションから手を動かすこと無く方向と量の入力を行うことを可能とする。また、実現に必要なハードウェアがマイク1つのみであるので、ハードウェア面におけるコストを低く抑えられるという利点がある。以下、本手法の特徴について詳述する。

方向の入力

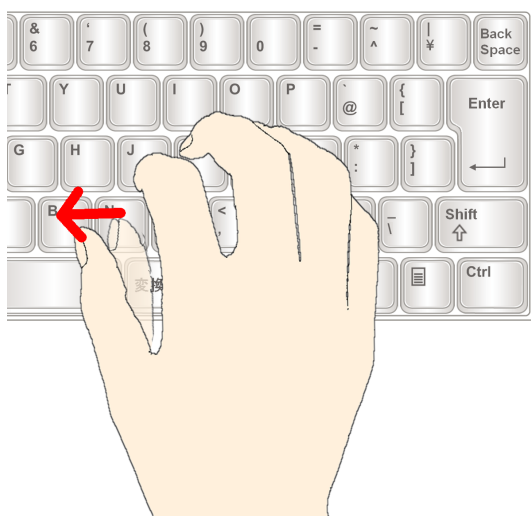
Keyboard Clawing は、ユーザが上下左右のうちのいずれかの方向の入力を行うことを可能とする。ユーザは、実際の指の動きに対応した方向の入力を行うことができる。そのため、手元を見ずに方向の入力を行うことができる。



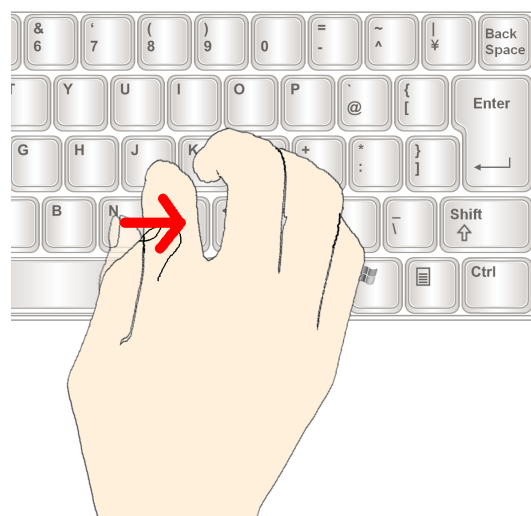
(a) 上



(b) 下



(c) 左



(d) 右

図 3.1: 各方向へのキートップの引っ掻き方

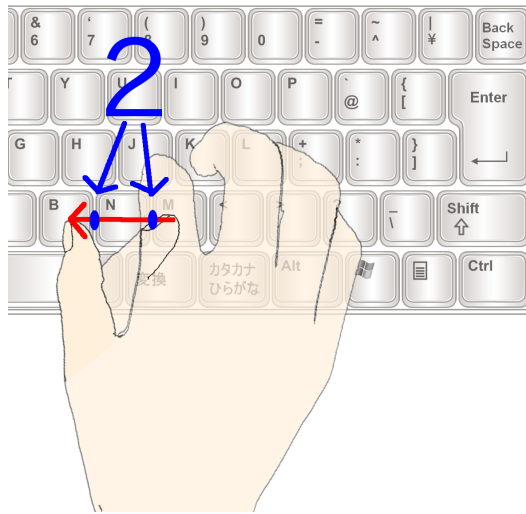


図 3.2: 具体的なキートップの引っ掻き方の例

量の入力

Keyboard Clawing は、ユーザが量の入力を行うことを可能とする。ユーザはキートップを引っ掻く際に通過するキー間の溝の数の概数を、触覚的フィードバックにより知ることができる。そのため、ユーザは手元を見ることなく、画面を注視したまま量の入力を行うことができる。

素早い入力

Keyboard Clawing は、ユーザがキーボードのホームポジションの位置から手を動かさことなく素早く入力を行うことを可能とする。そのため、ユーザは文字入力時などキーボードを用いたタスクを行なっている際に、素早く Keyboard Clawing による入力と文字入力を切り替えることができる。

ハードウェア面において低コストに実現

[DEWB09, BGV10] 等においては、ユーザの操作の識別のためにキーボードを改造する必要があるのに対し、Keyboard Clawing はハードウェア面における改良が、ユーザが普段使用しているキーボードにマイクを1つ装着する事のみである。そのため、ハードウェアの面において低コストに実現できるという利点がある。

3.3 認識方法のアイデア

ユーザが複数のキートップを引っ掻く際、ユーザの指がキー間の溝に引っかかり、操作音の波形にピークが生じる。そこで、操作音の波形のピークの数を知る事により、距離が認識できると考えた。また、ユーザがキーボードのホームポジションの位置に手を置いた状態からキートップを引っ掻く際、ユーザの指がキー間の溝を通過する際にキートップに触れる指とその部位を、表 3.1 に示す。表 3.1 に示した部位の違いによる操作音の違いを認識する事により、ユーザがキートップを引っ掻いた方向を認識できると考えた。

表 3.1: 各方向へ向けて引っ掻く際に触れる指とその部位（右手の場合）

方向	上	下	右	左
指	人差し指	人差し指	親指	親指
指の部位	爪先	腹	側面（腹）	側面（爪）

第4章 実装

本章では、Keyboard Clawing の実装について述べる。まずシステム構成について述べた後、ノイズ処理、距離の認識、方向の認識について述べる。

4.1 システム構成

本節では、今回開発したシステムにおける実行環境、ハードウェア、ソフトウェアを述べる。

実行環境

本システムを実行する際に使用した計算機の仕様を表 4.1 に示す。

ハードウェア

操作音を取得するためのマイクとして、Shadow®社製のピエゾピックアップ SH710 (図 4.1) を用いた。ピエゾピックアップを用いた理由は、空気音中のノイズを防ぐためである。また、キーボードとして Dell 社製の KB212-B を用いた。ピエゾピックアップは、キーボードの Enter キー付近に装着した。この様子を図 4.2 に示す。

ソフトウェア

システムのフローチャートを図 4.3 に示す。

入力の識別の流れについて述べる。システムはまず起動直後の 5 フレーム（フレーム幅：256）の音を取得し、得られた音を環境騒音とする。サンプリング周波数は 96kHz とした。システムは、キーボードの表面に装着したマイクから取得した音に対し、環境騒音を基にノイズ除去を行う。次に、ノイズ除去後の音から音圧が閾値 A を超えたサンプル（ピーク）が検

表 4.1: 実行する際に使用した計算機の仕様

マシン	Dell®Vostro 260MT Desktop
OS	Ubuntu®10.04 Lucid Linx ¹
CPU	Intel®Core i5 3.10GHz



図 4.1: 使用したピエゾピックアップ (SH710)



図 4.2: ピエゾピックアップを装着したキーボード

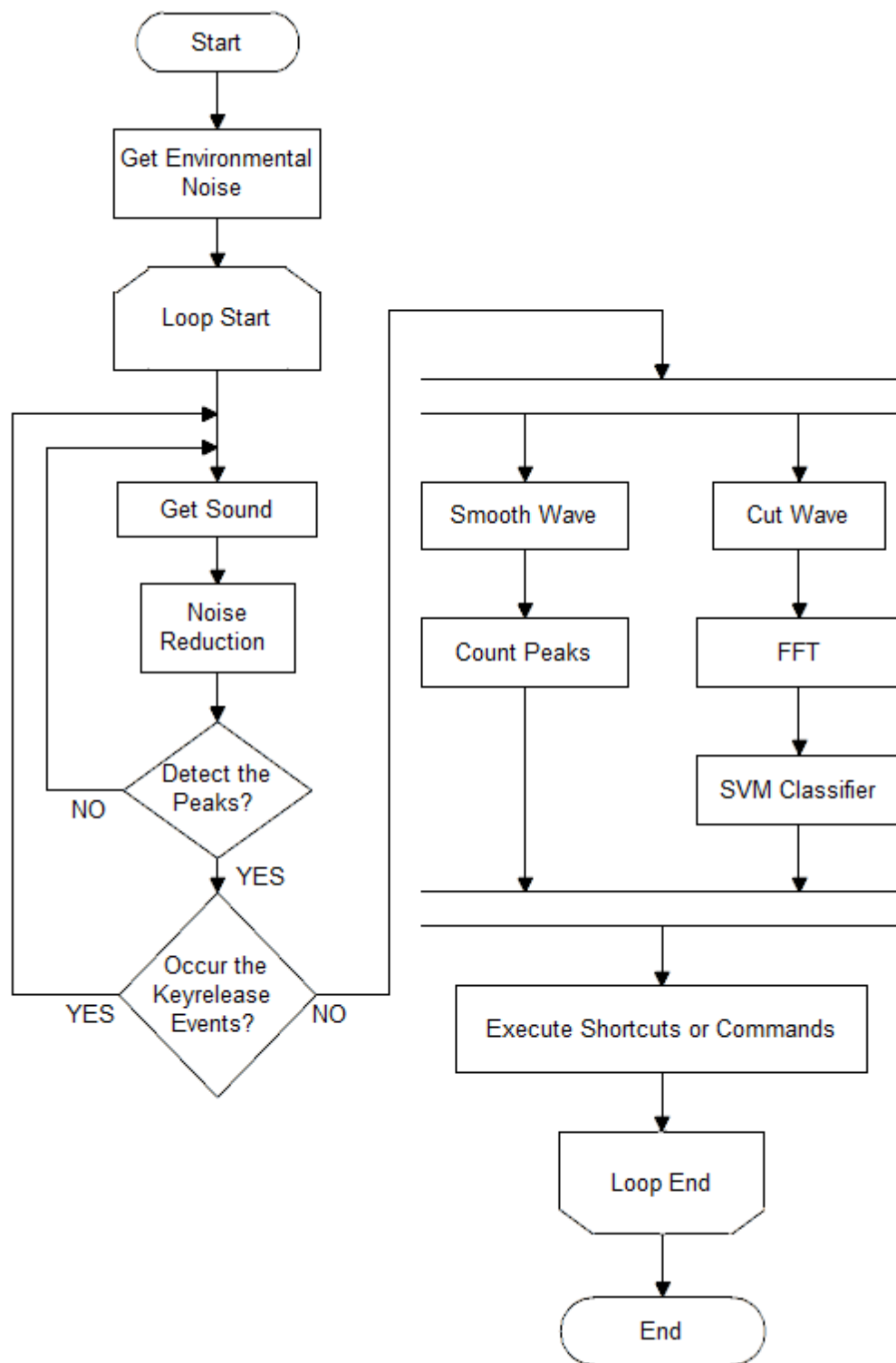


図 4.3: システムのフローチャート

出された場合、ユーザがキートップ上の引っ掻きジェスチャを行ったとする。ただし、キーリリースイベントが発生してから 0.5 秒以内に発生した音については、ユーザの打鍵の際に発生した音とみなし、無視することとした。なお、閾値 A の値は筆者が経験的に求めた値である。また、ユーザがキートップを引っ掻いた距離を認識するために、一定時間内に発生するピークの数进行数える。この際、波形のピークを強調するために予め波形に対して平滑化を行う。並行して、ユーザがキートップを引っ掻いた方向を認識するために、操作音の始めのピーク周辺の音を切り出す。得られた音に対して高速フーリエ変換 (FFT) を行い、得られた周波数スペクトルを特徴量とする。識別器として、サポートベクターマシン (SVM) を用いる。その後、得られた距離と方向を用いて、ユーザが予め引っ掻きジェスチャと対応させておいたショートカットまたはコマンドを実行する。

なお、本システムの開発のためのプログラミング言語として Python 2.6²を用いた。

4.2 操作音の取得

操作音を取得する際のサンプリング周波数を 96kHz、量子化ビット数を 16bit、チャンネル数を 1 チャンネル (モノラル) とした。また、量子化ビット数が 16bit であるので、システムは音圧を 0~65535 の数値のデータとして取得する。システムは、取得した音圧を -1 から 1 までにスケールする。マイク端子から音を取得し、処理するためのライブラリとして、PyAudio³を用いた。

4.3 ノイズ処理

本節では、ノイズ処理について述べる。

4.3.1 環境音のノイズ除去

本システムは、ピエゾピックアップを通して得られた音に対してまず Minimal Mean-Square Error Short-Time Spectral Amplitude (MMSE-STSA) 法 [EM84] を用いてノイズ処理を行う。MMSE-STSA 法は、本来の音声信号 (原音声) と推定した音声信号 (推定音声) の短時間振幅スペクトルの平均二乗誤差を最小にする手法である。時刻 t における原音声を $x(t)$ 、環境騒音を $d(t)$ とすると、ピエゾピックアップから得られる入力音声 $y(t)$ を式 4.1 により表す事が出来る。

$$y(t) = x(t) + d(t) \quad (4.1)$$

²<http://www.python.org/>

³<http://people.csail.mit.edu/hubert/pyaudio/>

ここで、原音声と入力音声の k 番目の周波数スペクトル X_k 、 Y_k をそれぞれ式 4.2、4.3 により定義する。

$$X_k := A_k \exp(j\alpha_k) \quad (4.2)$$

$$Y_k := R_k \exp(j\theta_k) \quad (4.3)$$

なお、 j は虚数単位を表す。その後、原音声の振幅スペクトル $\hat{A}_k$ を式 4.4 により推定する。

$$\hat{A}_k = \Gamma(1.5) \frac{\sqrt{v_k}}{\gamma_k} M(-0.5, 1, -v_k) R_k \quad (4.4)$$

Γ はガンマ関数、 M は合流超幾何関数であり、 v_k は式 4.5 により定義されたものである。

$$v_k := \frac{\xi_k}{1 + \xi_k} \gamma_k \quad (4.5)$$

式 4.5 中における ξ_k は事前 S/N 比、 γ_k は事後 S/N 比を表す。最後に、 $\hat{A}_k \exp(j\theta_k)$ に対し逆フーリエ変換を適用する事により、原音声の推定波形を得る。なお、ノイズ処理を行う際の窓サイズを 256 とした。また実装に際し、「Python によるオーディオ処理 - Kaiseki」⁴ を参考にした。MMSE-STSA 法によりノイズ処理を行う前の波形を図 4.4a に、ノイズ処理を行った後の波形を図 4.4b に示す。ノイズ処理の前後において波形中のノイズが軽減されている事が分かる。

4.3.2 打鍵音と操作音の区別

打鍵音と操作音を区別するために、システムはキーリリースイベントを監視する。キーリリースイベントが発生してから 0.5 秒以内に発生した音をユーザの打鍵音と認識し、操作音とは認識しない事とした。なお、0.5 秒という値は筆者が経験的に求めた値である。

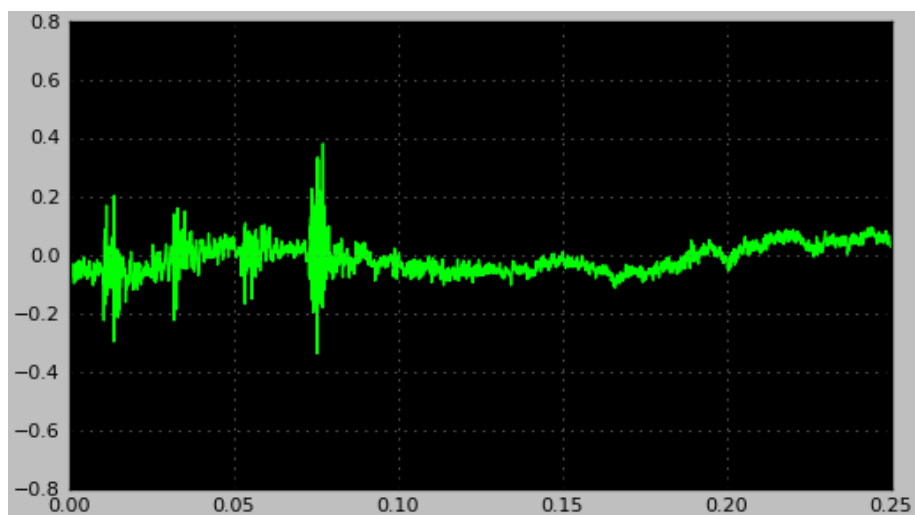
キーリリースイベントを監視するために、グローバルフックを行った。この実装に際し、Xlib ライブラリ⁵を用いた。

4.4 距離の識別

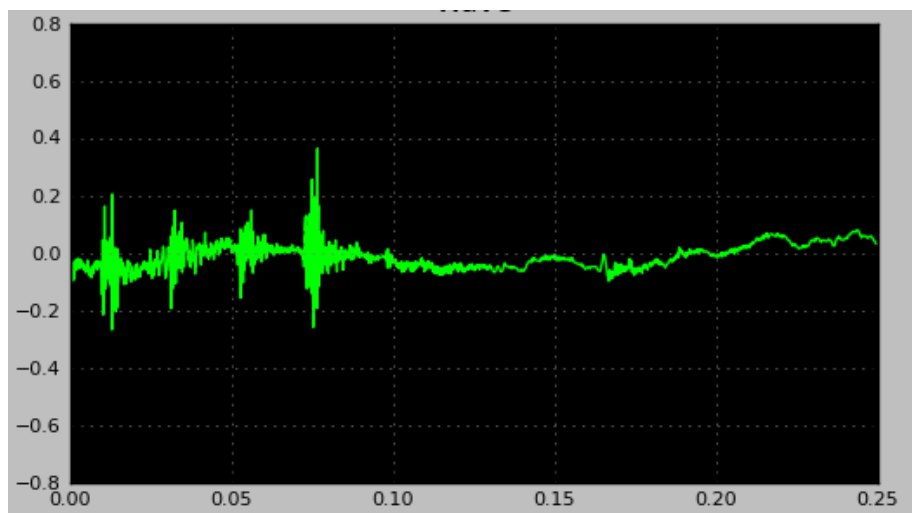
システムは、ユーザがキートップを引っ掻いた距離を識別するために、操作音の波形のピークの数を数える。まず、操作音の波形のピークをより強調するために、波形の各サンプルに

⁴http://kaiseki-web.lhd.nifs.ac.jp/wiki/index.php/Python_によるオーディオ処理

⁵<http://www.x.org/wiki/>



(a) ノイズ処理前の波形



(b) ノイズ処理後の波形

図 4.4: ノイズ処理前後の波形

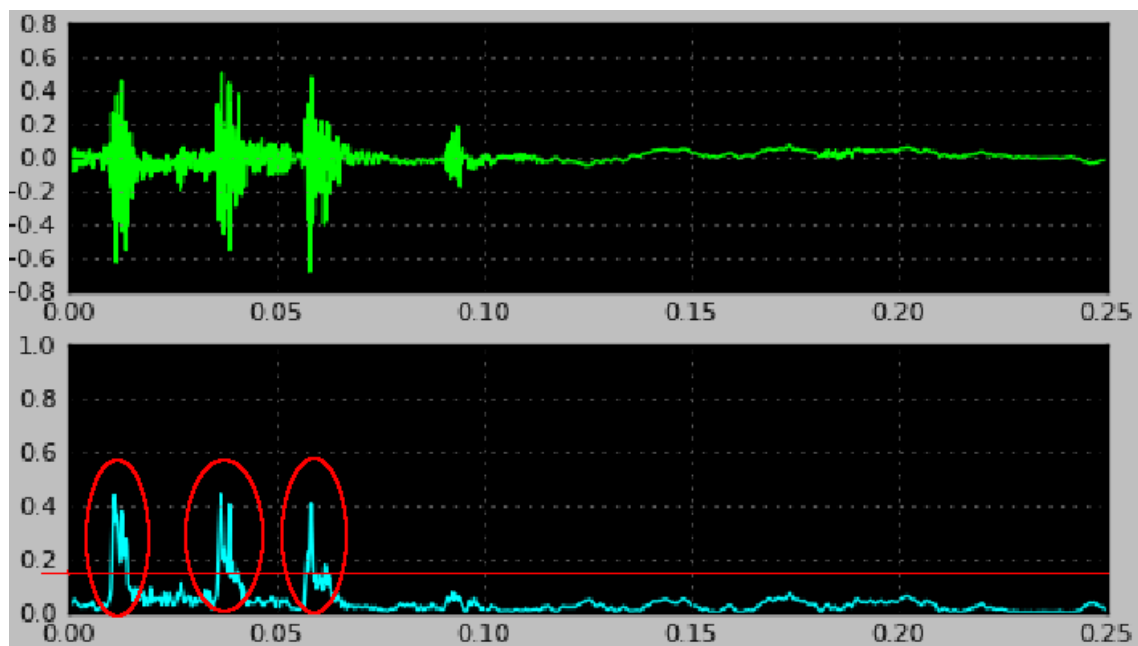


図 4.5: 上：平滑化前の波形 下：平滑化後の波形

対して前後 50 サンプルの音圧の絶対値との平均を取り、波形を平滑化する。平滑化前の波形を図 4.5 上に、平滑化後の波形を図 4.5 下に示す。次に、平滑化後の波形を走査し、操作音の始めのピークから 24000 サンプル（0.25 秒間相当）以内に現れた絶対値が閾値 B を超えたサンプル（以降ピーク B と呼ぶ）の数を、操作音の波形のピークの個数とする。ただし、ピーク B が現れてから 2000 サンプル以内に現れたピーク B については、同じキートップを引っ掻いている間に発生したとみなし、無視することとした。なお、閾値 B の値は筆者が経験的に求めた値である。図 4.5 の例においては、波形のピークの個数を 3 個とする。

4.5 方向の識別

方向を識別するための特徴量として操作音の周波数スペクトルを用いた。ユーザの指がキー間の溝を 1 つだけ通過する際に生じる操作音は十分短いので、時系列データは用いずに単一スペクトルのみを扱う事とした。システムは、まず操作音の始めのピークを中心として 2000 サンプルを抜き出す（図 4.7）。次に、システムは得られた波形に対し FFT を行う事により、周波数スペクトルを得る。なお、FFT 窓サイズを 2000 サンプルとし、窓関数としてハミング窓を用いた。各方向に引っ掻いた際に現れる周波数スペクトルを図 4.6 に示す。

方向の識別器として SVM を用いた。SVM のタイプを C-SVC、カーネル関数を線形カーネル、cost 値を 512.0、gamma 値を 0.000125 とした。また、周波数スペクトルは窓の中央において左右対称であるので、一度の操作につき FFT 窓サイズの半分の 1000 次元の特徴量が得

られる。システムは、この特徴量に対して予め-1 から 1 までにスケーリングする。なお、実装には LIBSVM[CL11] ライブラリを使用した。

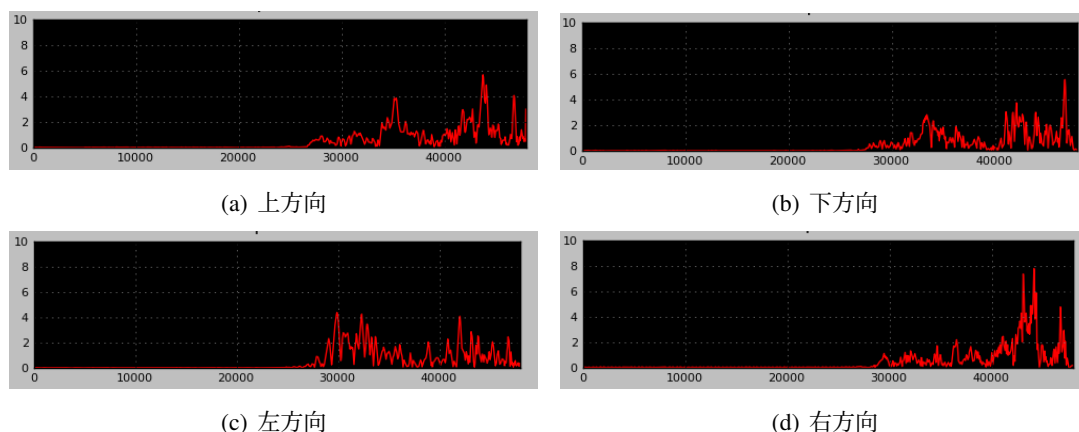


図 4.6: 各方向に対する操作音の周波数スペクトル

4.6 GUI ツール

音声解析を行う事を目的とし、GUI ツールを作成した。ツールの画面を図 4.8 に示す。このツールを用いる事により、以下の操作を行う事が出来る。

波形・周波数スペクトル・平滑波形の表示

波形、周波数スペクトル、平滑波形の表示を行う事が出来る。操作音が検出された際、始めのピークから 24000 サンプルの波形を表示する (図 4.8a)。また、操作音の始めのピークを中心とした 2000 サンプルの周波数スペクトルを表示する (図 4.8b)。さらに、図 4.8a にて表示した波形を平滑化した波形を表示する (図 4.8c)。

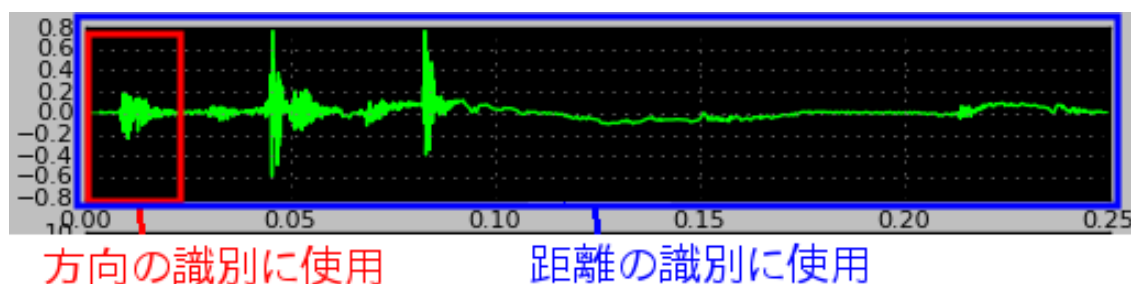


図 4.7: 距離の識別と方向の識別それぞれに用いる部分

特徴量データのラベル付け

操作音が検出された際、図 4.8b にて表示した周波数スペクトルの特徴量データに対してラベル付けを行い、特徴量データをメモリに保存する。ラベルは、図 4.8d にて示したラジオボタンのうちチェックが付いているものとなる。また、図 4.8i の「DEL」ボタンを押す事により、現在チェックが付いているラジオボタンのラベルの特量量を全てメモリから削除する事が出来る。

特徴量データの保存/読み込み

「SAVE」ボタン（図 4.8f）を押す事により、ラベル付き特徴量を LIBSVM 形式で保存する事が出来る。この際、保存するファイル名を図 4.8k のテキストボックスにて指定する。また、「LOAD」ボタン（図 4.8g）を押す事により、LIBSVM 形式で保存した特徴量のデータを読み込む事が出来る。この際、読み込むファイル名を図 4.8l のテキストボックスにて指定する。

操作音の識別

GUI ツールは起動時に、まず LIBSVM 形式にて保存された特徴量データを読み込む。その特徴量データをもとに、操作音が検出される度に操作音がどのラベルの音か識別される。識別結果のラベルは予め指定しておいた方向に対応付けられ、距離の識別結果とともに図 4.8e のボックス中に表示される。また、図 4.8m のテキストボックスに学習用の特徴量データが保存されたファイル、図 4.8q のテキストボックスにテスト用の特徴量データが保存されたファイルを指定し、「EXEC」ボタン（図 4.8h）を押す事により、識別率と予想結果のリストをコンソール上に表示する。この際、図 4.8s のテキストボックスに複数のラベルを指定すると、指定されたラベル間のみにおける識別を行う事が出来る。図 4.8 の例においては、「model」ファイルを学習用、「test」ファイルをテスト用として用い、学習用ファイルとテストファイルそれぞれからラベルが「0」「1」「2」「3」の特徴量データのみを全て抜き出し、それらの識別結果のリストをコンソール上に表示する。

入力音声に対するノイズ処理の ON/OFF の切り替え

入力音声に対して、ノイズ処理の ON/OFF を切り替える事が出来る。図 4.8o の「noise_reduction」チェックボックスの ON/OFF が、入力音声に対するノイズ処理を ON/OFF に対応する。

入力音声の録音

入力音声を録音する事が出来る。「REC」ボタン（図 4.8j）を押した時点から入力音声の録音を開始する。もう一度「REC」ボタンを押すと、それまで録音していた音声を WAV ファイルとして保存する。この際、保存するファイル名を図 4.8n のテキストボックスにて指定する。

WAV ファイルの入力

WAV ファイルを入力音声として用いる事が出来る。「PLAY」ボタン（図 4.8r）を押すと、指定した WAV ファイルを入力音声として扱い、操作音の検出とその特徴量に対するラベル付けを行う事が出来る。この際、読み込むファイル名を図 4.8t のテキストボックスにて指定する。この機能により、同じ操作音を用いて識別アルゴリズムの比較を行う事が出来る。

キーリリースイベント監視の ON/OFF

キーリリースイベントの監視の ON/OFF を切り替える事が出来る。「MONITOR」ボタン（図 4.8p）にフォーカスを合わせるとキーリリースイベントの監視を始め、フォーカスを外すと監視を行わない。監視中にキーリリースイベントが発生した際、発生してから 0.5 秒の間だけ図 4.8e のボックスにキーリリースイベントが発生した旨を表示するとともに、操作音の検出を行わない。

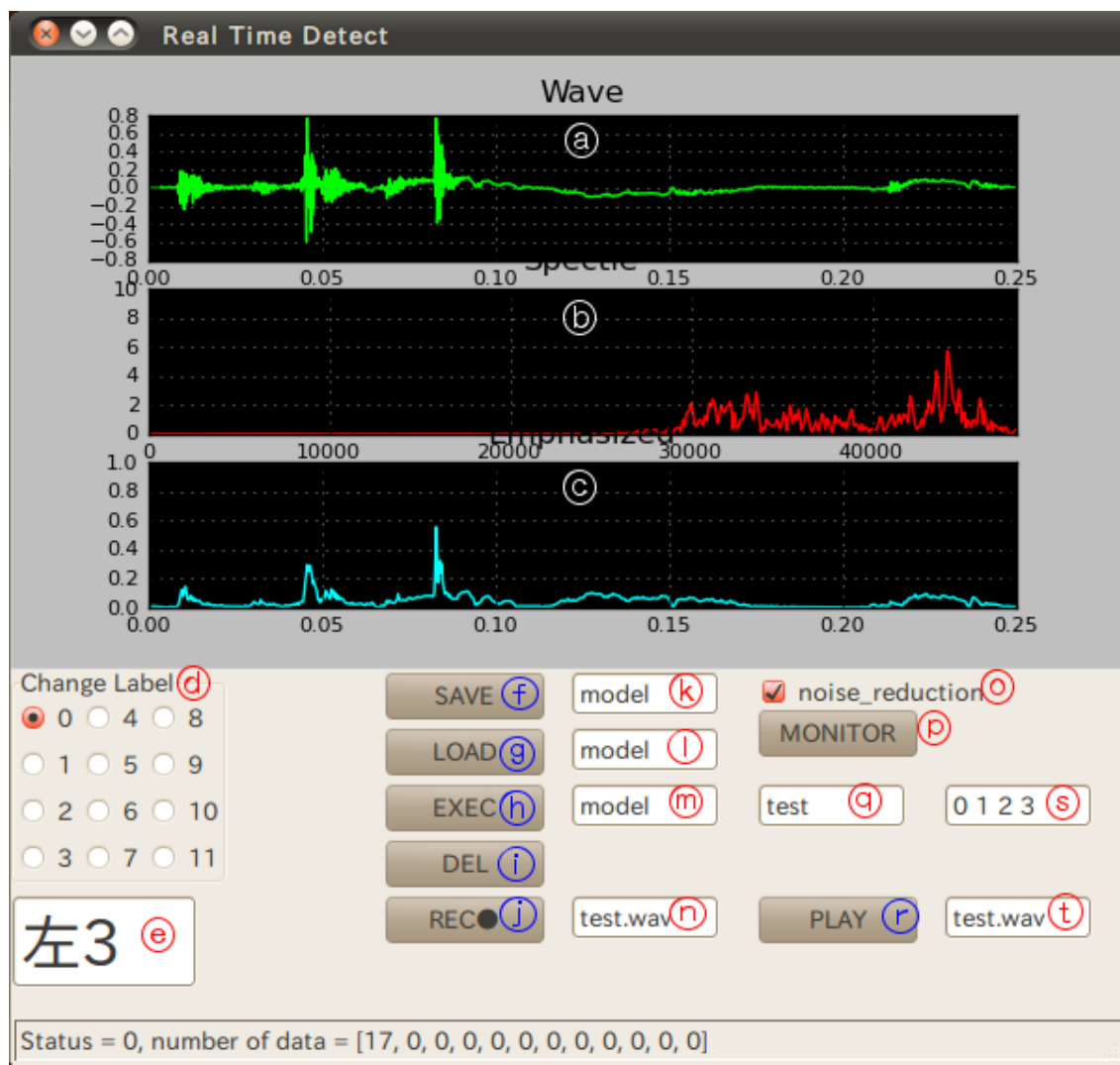


図 4.8: GUI ツールの画面

第5章 応用例

本章では、Keyboard Clawing の応用例を述べる。ユーザは、Keyboard Clawing を任意のショートカットに割り当てる事が出来る。例として、以下の機能が挙げられる。

- 音量調節機能
- スクロール機能
- フォントサイズ変更機能

以下、簡単のためにキートップの引っ掻きジェスチャを行うことを「(距離) - (方向) clawing」と表す。

5.1 音量調節機能

音楽再生アプリケーションを使用中に音量を調節する従来の操作方法として、GUI による操作とショートカットキーの押下が挙げられる。ユーザが GUI による操作を行う際、ユーザはポインティングデバイスに手を移動させる時間と、GUI 要素を操作する時間を要する (図 5.1)。また、ユーザがショートカットキーを押下する際、音量を大きく調整したい場合はショートカットキーを連打する必要がある。さらに、音量調整に対応するショートカットキーはキーボードの外側に付いているケースが多いため、ユーザはキーボードのホームポジションから手を大きく動かす必要があるケースが多い。音楽再生アプリケーションの例として、Windows Media®Player (以降 WMP と呼ぶ) が挙げられる。WMP では、音量の上昇と下降のショートカットキーとしてそれぞれ F9 キー、F8 キーが割り当てられている。また、Macintosh® (以降 Mac と呼ぶ) では、音量の上昇と下降のショートカットキーとしてそれぞれ F11 キー、F12 キーが割り当てられている。このような例を始めとして、音量調節のためのショートカットキーはキーボードの外側のキーである事が多く、またショートカットの対応を覚える事が困難である。このため、ユーザが WMP または Mac 等を使用中にショートカットキーを用いて音量調節を行う際、ユーザはキーボードのホームポジション上に手を置いた状態から手を大きく移動させる必要がある (図 5.2)。

そこで、ホームポジションに手を置いた状態から素早く音量調節を行う事を目的とし、表 5.1 に示すように音量の調節のショートカットに左右方向への引っ掻きジェスチャを対応させる。この時、ユーザはホームポジションの位置から手を動かすことなく、素早く連続量を入力することができる。操作の様子を図 5.3 に示す。図 5.3 では、1-right clawing により音量を 10% 上昇させ、2-left clawing に音量を 20% 下降させている。

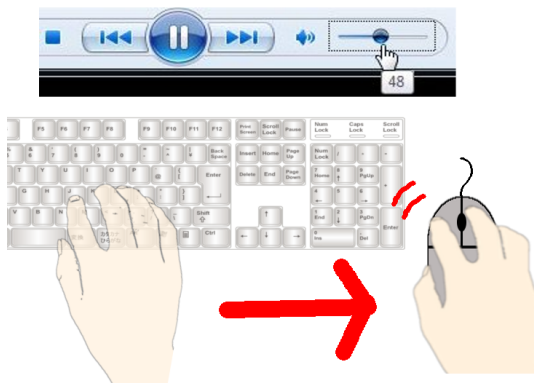


図 5.1: GUI の操作による音量調節

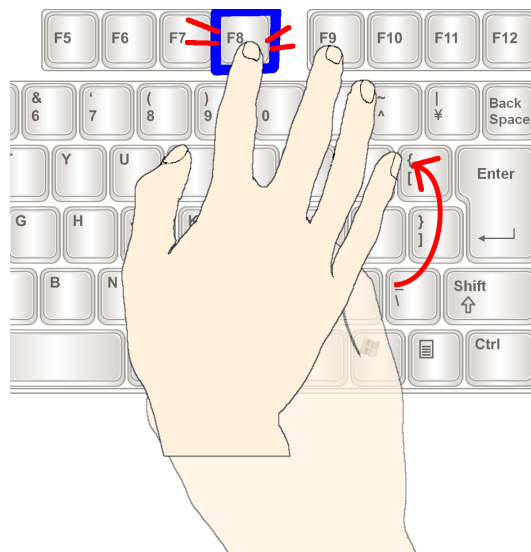
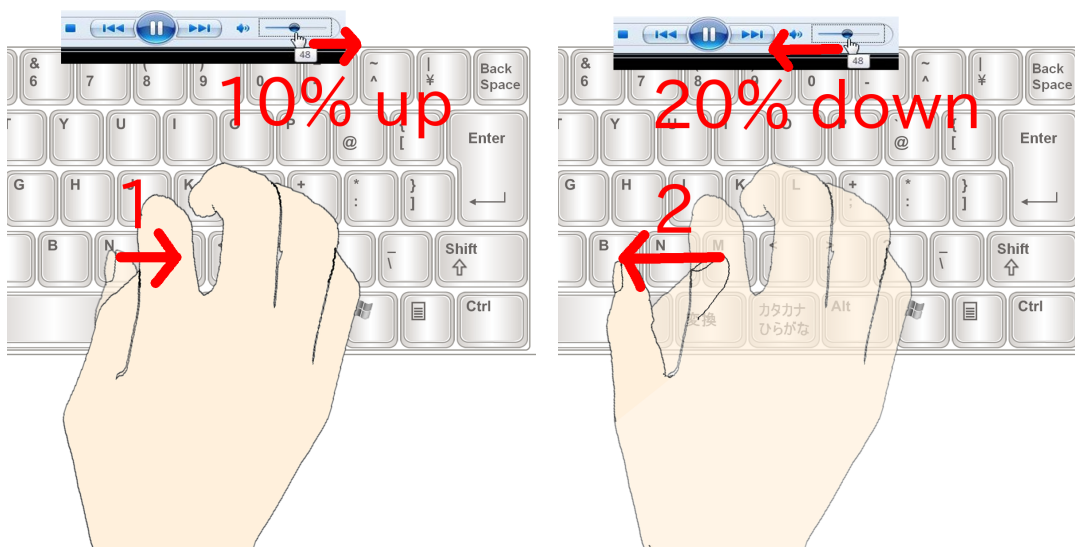


図 5.2: WMP におけるショートカットキー (F8) による音量調節

表 5.1: 各方向と距離の引っ掻きジェスチャに対応させる音量の調節量

方向\距離	1	2	n
左	-10%	-20%	$-n \times 10\%$
右	+10%	+20%	$+n \times 10\%$



(a) right

(b) left

図 5.3: Keyboard Clawing による音量調節

5.2 スクロール機能

スクロールを行うための従来の操作として、スライダのドラッグによる操作、マウスホイールの操作、そしてキーの押下が挙げられる。ユーザがスライダまたはマウスホイールによる操作を行う際、ユーザはポインティングデバイスに手を移動させる時間を要する（図 5.4）。また、ユーザがキーの押下によるスクロールを行う際、カーソルキー・PageDown キー・End キーといったスクロールに対応するキーがキーボードの外側についている事が多いため、ユーザはキーボードのホームポジション上に手を置いた状態から手を大きく移動させる必要がある（図 5.5）。

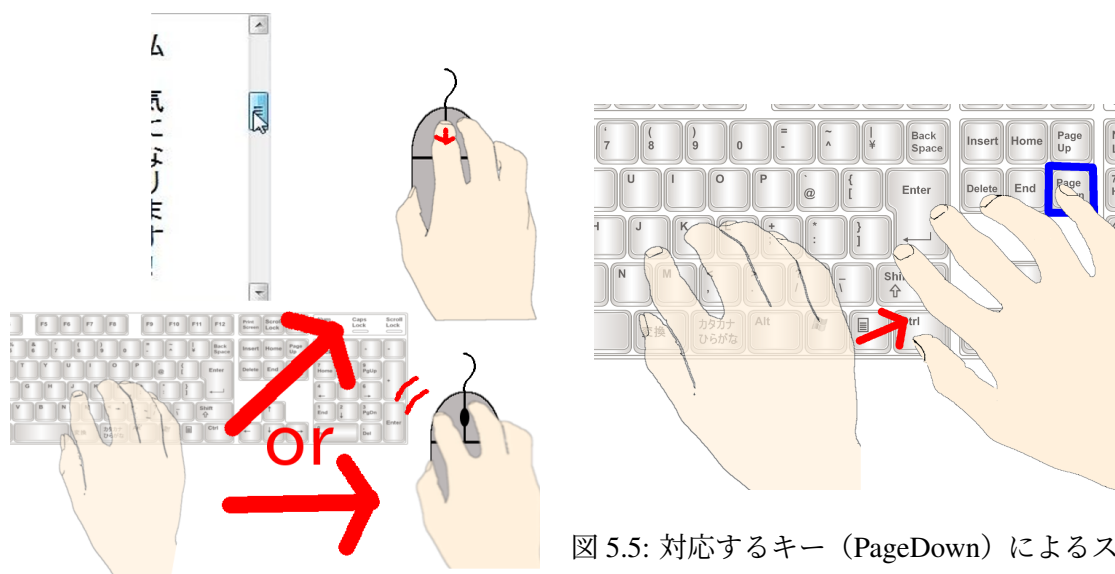


図 5.5: 対応するキー（PageDown）によるスクロール

図 5.4: スライダのドラッグまたはホイールの操作によるスクロール

そこで、ホームポジションに手を置いた状態から素早くスクロールを行う事を目的とし、表 5.2 に示すようにスクロールのショートカットに上下方向への引っ掻きジェスチャを対応させる。この時、ユーザはホームポジションの位置から手を動かすことなく、素早くスクロールを行うことができる。操作の様子を図 5.6 に示す。図 5.6 では、1-up clawing により 1 行上へスクロールし、2-down clawing により 1 ページ下へスクロールし、3-down clawing により末尾までスクロールしている。

表 5.2: 各方向と距離の引っ掻きジェスチャに対応させるスクロールの向きと量

方向\距離	1	2	3
上	上に 1 行	上に 1 ページ	先頭まで
下	下に 1 行	下に 1 ページ	末尾まで

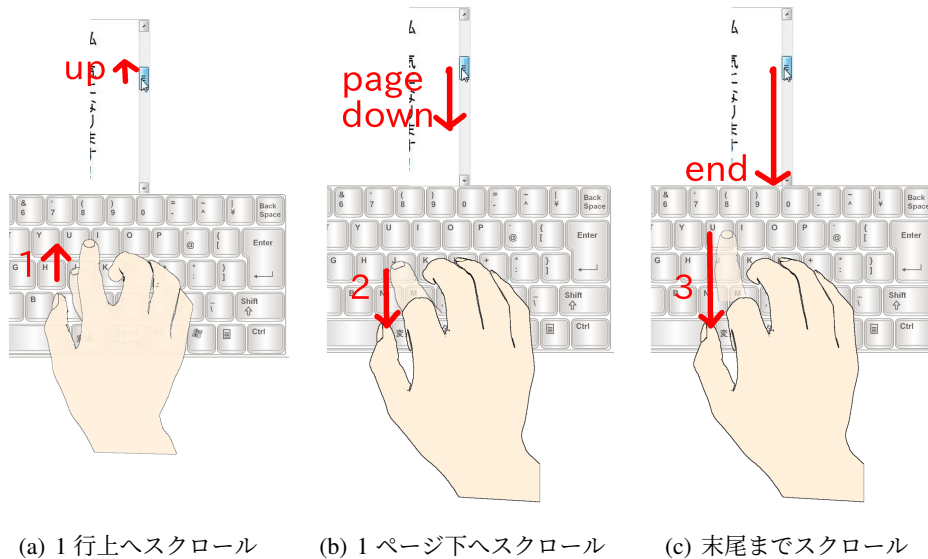


図 5.6: Keyboard Clawing によるスクロール

5.3 フォントサイズ変更機能

文書作成ソフトにおいてフォントサイズを変更するための従来の操作として、GUI による操作とショートカットキーの押下が挙げられる。ユーザが GUI による操作を行う際、ユーザはポインティングデバイスに手を移動させる時間と、GUI 要素を操作する時間を要する（図 5.7）。また、ユーザがショートカットキーを押下する際、フォントサイズを大きく変更したい場合はショートカットキーを連打する必要がある。文書作成ソフトの例として、Microsoft®Word®2010（以下 Word と呼ぶ）が挙げられる。Word では、フォントサイズの拡大のショートカットキーに「Shift + Ctrl + >」、縮小のショートカットキーに「Shift + Ctrl + <」が割り当てられている。このように、ショートカットキーを押下する際に、キーの組み合わせによっては同時に押すキーの数が多い、またはユーザが対応を覚える事が困難である（図 5.8）。また、フォントサイズを変更するためには文字を選択する必要がある事が多い。文字を選択するための従来の操作として、ドラッグによる操作及びショートカットキーの押下が挙げられる。ユーザがドラッグによる文字選択を行う際、ユーザはポインティングデバイスに手を移動させる時間と、文字を選択する時間を要する（図 5.9）。文字選択のショートカットキーの例として、「Shift + 左（右）カーソルキー」で左（右）1 文字選択、「Shift + Ctrl + 左（右）カーソルキー」で左（右）1 単語選択、「Shift + Home（End）」で先頭（末尾）まで選択、が挙げられる。ここにおいて挙げたキーはキーボードの外側に位置している事が多いため、ユーザはキーを押すためにホームポジションから手を大きく動かす必要がある（図 5.10）。

そこで、ホームポジションに手を置いた状態から素早くフォントサイズの変更と文字の選択を行う事を目的とし、表 5.3 に示すように文字選択のショートカットに左右方向への引っ掻きジェスチャを、フォントサイズの変更のショートカットに上下方向への引っ掻きジェスチャ

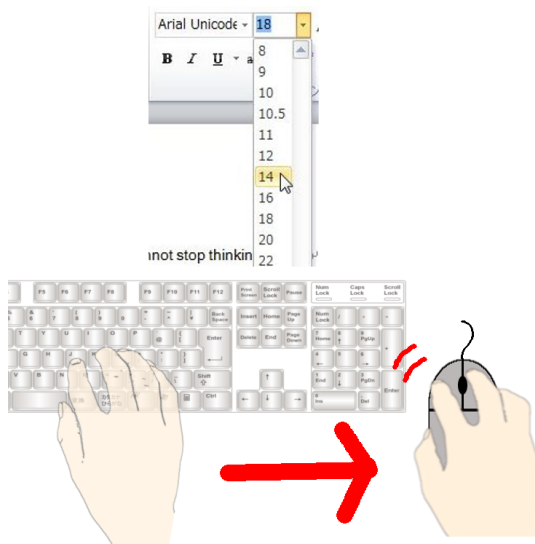


図 5.7: GUI の操作によるフォントサイズの変更

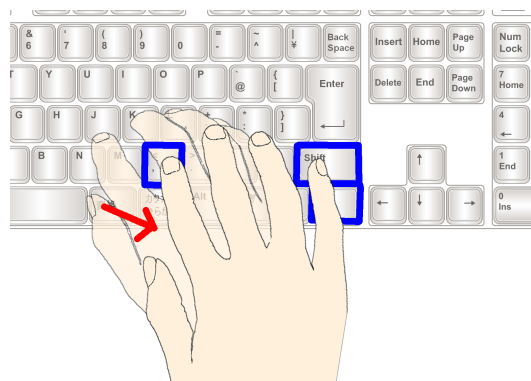


図 5.8: Word におけるショートカットキー (Shift + Ctrl + <) によるフォントサイズの変更

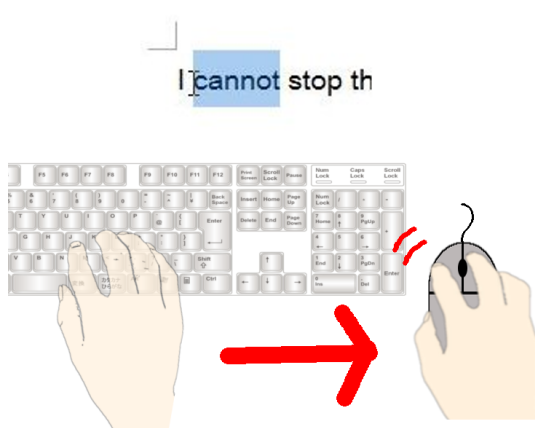


図 5.9: GUI の操作による文字の選択

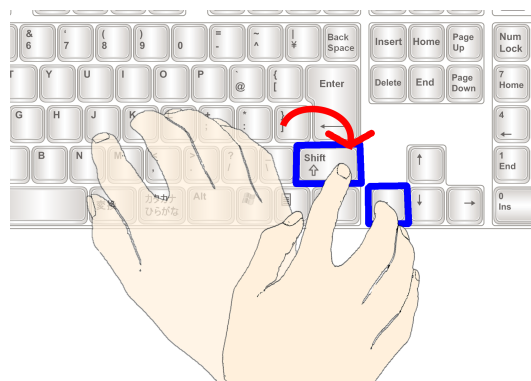


図 5.10: ショートカットキー (Shift + 左キー) による文字の選択

を対応させる。この時、ユーザはホームポジションの位置から手を動かすことなく、素早く文字を選択し、フォントサイズを変更することができる。文字選択の操作とフォントサイズの変更の操作の様子をそれぞれ図 5.11、図 5.12 に示す。

表 5.3: 各方向と距離の引っ掻きジェスチャに対応させる機能

方向\距離	1	2	3
上	フォントサイズ+5pt	フォントサイズ+10pt	フォントサイズ+15pt
下	フォントサイズ-5pt	フォントサイズ-10pt	フォントサイズ-15pt
左	左 1 文字選択	左 1 単語選択	先頭まで選択
右	右 1 文字選択	右 1 単語選択	末尾まで選択

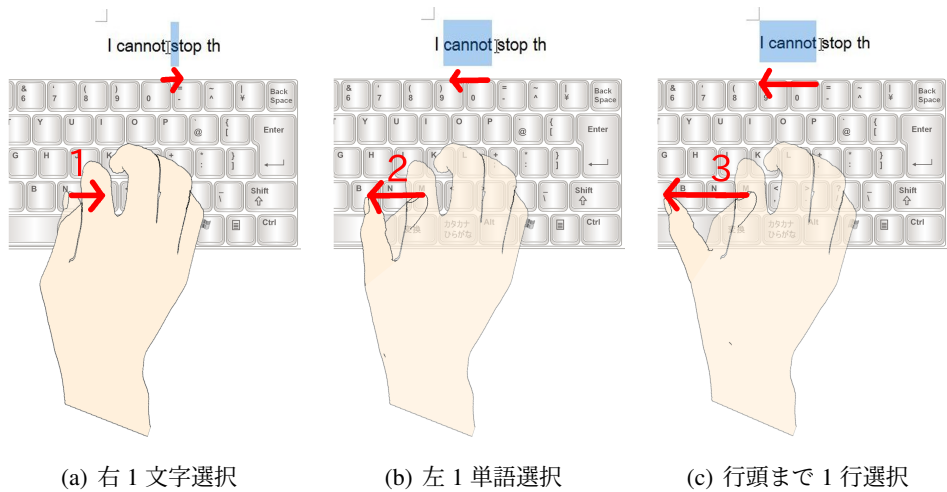
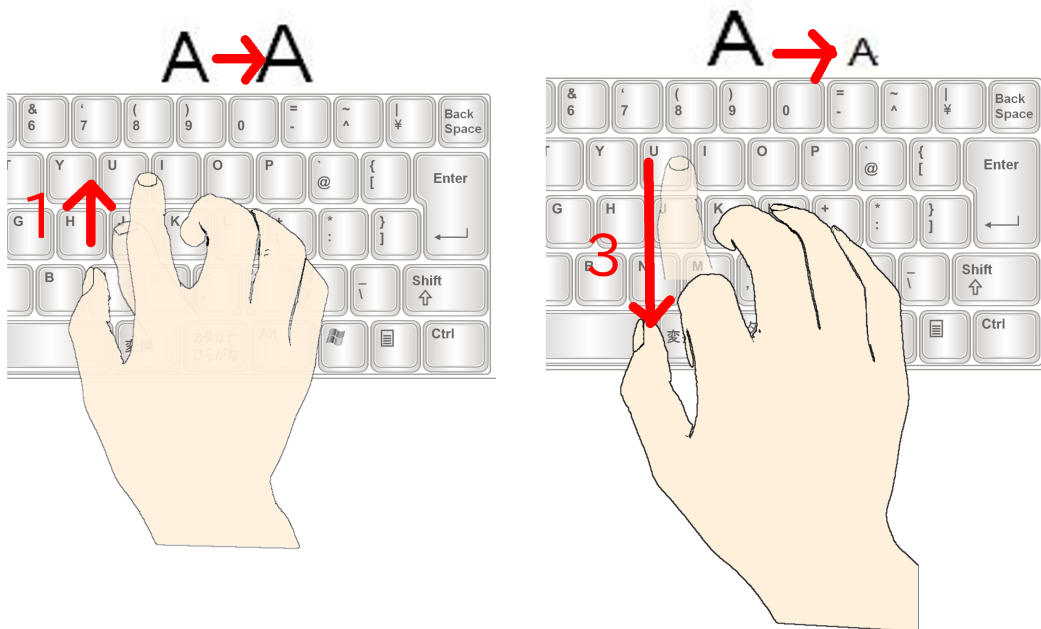


図 5.11: Keyboard Clawing による文字の選択



(a) 5pt 拡大

(b) 15pt 縮小

図 5.12: Keyboard Clawing によるフォントサイズの変更

第6章 評価実験

今回実装した Keyboard Clawing を評価するための被験者実験を行った。本章では、評価実験の詳細を述べる。

6.1 実験目的

本実験の目的を以下に示す。

- 周波数スペクトルを用いる事による、キートップを引っ掻いた方向の識別率の測定
- 本システムにおける、引っ掻いたキートップの距離の識別率の測定
- ユーザがキートップ上を引っ掻いた際に、本システムが操作音を認識しなかった割合の測定
- ユーザによる引っ掻きジェスチャの際に発生するキー押下率の測定

6.2 実験環境

本実験は大学の研究室内の静音環境にて行った。実験環境を図 6.1 に示す。被験者は 22 歳-24 歳の大学生・大学院生で、男性 11 名、女性 1 名の計 12 名であった。なお、被験者の内 1 名は操作ミスが非常に多かったため、その被験者のデータを外れ値として扱った。そのため、本章にて述べる実験結果は全て、その 1 名を除く 11 名のデータである。

6.3 実験内容

実験内容について述べる。実験者は始めに被験者に実験同意書を渡し、実験についての説明を行った。その後、本手法による入力を実験者が実演し、被験者は本手法に慣れるまで十分に練習を行った。なお、練習中において被験者は実験者に本手法及び本実験について自由に質問をする事が出来た。被験者が本手法に慣れたと申告した後に、後述するタスクを行った。タスク終了後、実験者は被験者に対してアンケートを採った。被験者がアンケートへの回答を完了すると、実験を終了した。

なお、本実験に用いた実験同意書とアンケート用紙、及びアンケートにて得られたコメントの一覧を本論文の末尾にそれぞれ付録 A、B として添付した。



図 6.1: 実験環境

タスク

タスクの詳細を述べる。

1. 被験者は、実験用コンピュータの前に座り、ホームポジションに手を置く。
2. 被験者は、ディスプレイ上に提示されたキーを押す。
3. ディスプレイに、方向・距離・キーが提示される。(図 6.2)
4. 被験者は、提示された方向へ向けて提示された距離だけキートップを引っ掻く。
5. 被験者はキートップを引っ掻いた後に、提示されたキーを押す。
6. 3～5 を繰り返す。

指定した方向は、上下左右の4通りのうちのいずれかであった。また、指定した距離は、上下方向への引っ掻きの際は1-4の4通り、左右方向への引っ掻きの際は1-10の10通りであった。ただし、本手法はユーザが素早く入力を行うことを可能にすることを目的の1つとするため、ユーザが本手法を使用している際に、ユーザが厳密に距離を数えることを想定していない。そのため本実験において提示された距離はあくまで目安であり、被験者は厳密に従う必要は無いこととした。

本実験では、合計 $(10 \text{ (通り)} + 4 \text{ (通り)}) \times 2 \text{ (方向)} \times 5 \text{ (試行)} \times 12 \text{ (人)} = 1680$ (回) のタスクを行った。なお、被験者の疲れによる集中力の低下を防ぐために、被験者はタスク間において自由に休憩を取ってよいものとした。

フィードバックとして、システムが操作音を検出した事を示す視覚フィードバックのみを提示した。被験者は、このフィードバックを見た後に、画面に提示されたキーを押し、次の



図 6.2: 本実験の際に被験者に提示される画面

タスクに進んだ。また、被験者が誤って指示通りの操作を行わなかった場合、そのタスク中に提示されたキーを押す前であれば、操作のやり直しを認めることとした。そのため、1つのタスク中に複数回の音を検出される場合がある。この場合、提示されたキーを押す直前に発生した音のみを、タスクにおいて被験者が行った操作の操作音とする事とした。

6.4 実験結果と考察

本節では、実験結果と考察について述べる。

6.4.1 距離の識別率

結果

距離の識別率を表 6.1 に示す。なお、被験者に提示された距離とシステムが検出した距離の差が許容誤差以内であれば、システムは被験者が引っ掻いた距離を正確に識別したものとみなした。上下方向への引っ掻きの際は許容誤差が 2 の場合にそれぞれ 98.2%、90.5% と高い識別率を示し、左右方向への引っ掻きの際は許容誤差が 3 の場合に 91.5%、89.6% と高い識別率を示した。

考察

許容誤差を 0 または 1 に設定した際における距離の識別率は低い一方、2 または 3 に設定した際には高い識別率を示した。

許容誤差が 0 の時の識別精度が悪い原因として、以下の事が挙げられる。ユーザがキートップを引っ掻く際に、まずユーザの指がキートップに触れる。ユーザが引っ掻く勢いの強弱に

表 6.1: 距離の識別率 (%)

許容誤差 方向	0	1	2	3	4
上	39.5	83.6	98.2	100.0	100.0
下	37.3	75.0	90.5	100.0	100.0
左	24.4	58.5	80.2	91.5	96.9
右	19.6	54.7	76.2	89.6	95.6

よっては、この際に発生する音をピークとして検出する場合としない場合とがある（図 6.3）。また、ユーザが最後に引っ掻こうとしたキートップに指が到達する前に、指がキートップから離れ、ユーザの意図した通りに引っ掻くことができず、ピークが生じない場合がある（図 6.4(b)）。これらの原因により、正確な距離の入力及び識別が困難であると思われる。

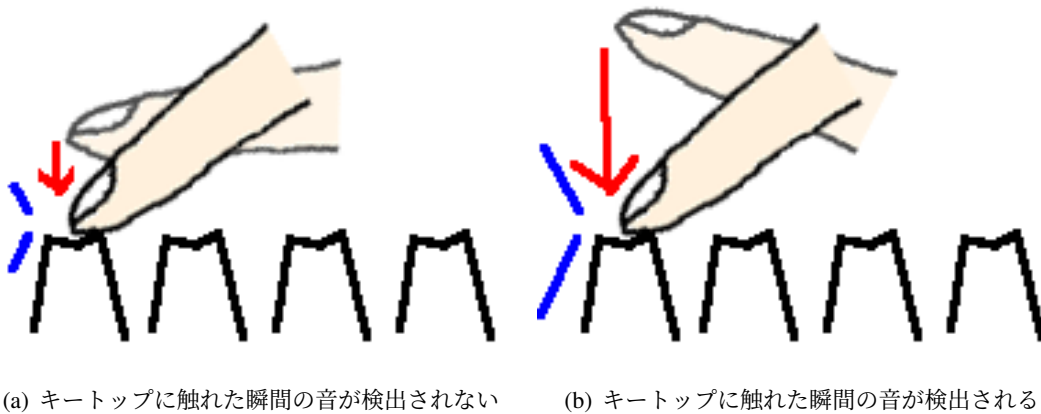
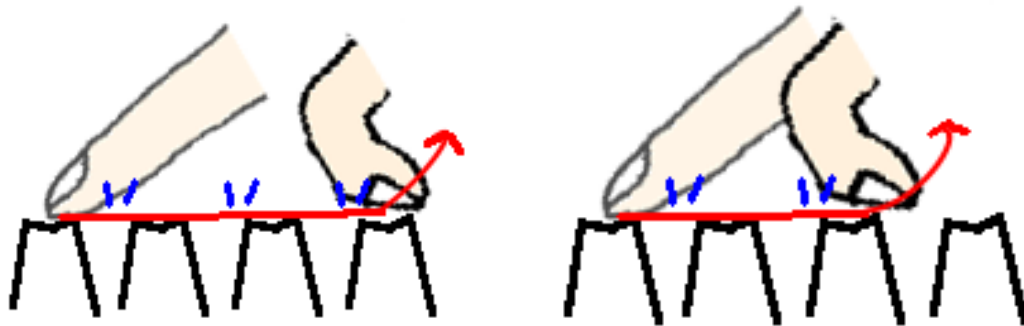


図 6.3: 指がキートップに触れる瞬間

6.4.2 方向の識別率

結果

方向の識別率について述べる。本実験において得られた各被験者の操作音から特徴量を抽出した。Per-User Test では、各被験者ごとに操作音の特徴量に対してそれぞれ 35 分割交差検定を行った結果の平均値を求めた。得られた混同行列を表 6.2 に示す。全ての方向の識別率の平均値を取ると、68.2%であった。また、Cross-User Test では、全ての被験者の操作音の特徴量に対して 35 分割交差検定を行った。得られた混同行列を表 6.3 に示す。全ての方向の識別率の平均値を取ると、56.4%であった。なお、交差検定の際に WEKA [HFH⁺09] を用いた。識



(a) 最後に引っ掻こうとしたキートップに指が到達する (b) 最後に引っ掻こうとしたキートップに指が到達する前に離れる

図 6.4: 下方向への引っ掻きジェスチャ（横から）

別器として LIBSVM を使い、SVM のタイプを C-SVC、カーネル関数を線形カーネル、cost 値を 512.0、gamma 値を 0.000125 とし、Normalize を True にした。

考察

Per-User Test の結果は 68.2% であった。この結果から、操作音の周波数スペクトルを特徴量として、ユーザがキートップを引っ掻いた方向を 7 割程度の精度で識別することが可能であることが分かった。Cross-User Test の結果は 56.4% であり、Per-User Test の結果に比べ低い精度であった。このことから、ユーザが引っ掻きジェスチャを行った際に生じる音の周波数スペクトルには個人差があると思われる。このため、実際の使用場面においては、ユーザは使用する前にキャリブレーションを行う必要があるであろう。

また、左右方向への引っ掻きジェスチャに比べ、上下方向への引っ掻きジェスチャの方が方向の識別精度が低い傾向にある事が分かった。これは、6.4.3 節にて述べるように、上下方向への引っ掻きジェスチャの際に生じる操作音の音圧は、左右方向への引っ掻きジェスチャの際に生じる操作音の音圧に比べて小さく、操作音の特徴量がうまく表れない事例が多かった事が原因であると考えられる。

6.4.3 操作音を検出されなかった割合

結果

被験者が引っ掻きジェスチャによる入力を行ったにも関わらず、システムが操作音を検出しなかった (FN) 回数を計測した。その結果、 FN となる事例は合計で 10.3% あった。この内訳を方向と距離別にまとめた結果を表 6.4 に示す。なお FN 率とは、被験者の引っ掻きによる操作のうち FN が発生した割合を示すものであり、式 6.1 にて求める。

表 6.2: Per-User Test における方向の識別率 (%)

入力値 \ 予測値	上	下	左	右	識別率
上	60.5	14.1	15.9	9.5	60.5
下	12.7	62.3	13.2	11.8	62.3
左	4.9	2.7	72.6	19.8	72.6
右	2.4	3.5	16.7	77.4	77.4
平均	-	-	-	-	68.2

表 6.3: Cross-User Test における方向の識別率 (%)

入力値 \ 予測値	上	下	左	右	識別率
上	41.8	22.7	23.2	12.3	41.8
下	24.5	46.4	12.3	16.8	46.4
左	10.0	7.5	56.9	25.6	56.9
右	4.4	4.5	25.3	65.8	65.8
平均	-	-	-	-	56.4

$$(FN \text{ 率}) = \frac{FN}{FN + (\text{タスクの総数})} \quad (6.1)$$

表 6.4: FN 率の内訳 (%)

方向 \ 距離	1	2	3	4	5	6	7	8	9	10	全体
上	9.8	12.7	8.3	0.0	-	-	-	-	-	-	7.9
下	36.8	30.4	15.3	25.7	-	-	-	-	-	-	27.9
左	20.3	11.3	14.1	3.5	6.8	1.8	0.0	0.0	0.0	0.0	6.3
右	19.1	14.1	8.3	9.8	1.8	1.8	0.0	0.0	0.0	0.0	6.0
全体	22.8	17.6	11.6	10.9	4.3	1.8	0.0	0.0	0.0	0.0	10.3

考察

結果を方向別に見ると、被験者が下方向への引っ掻きジェスチャを行う際に操作音が検出されない事例が 27.9%と多かった。これは、下方向に向けて引っ掻きジェスチャを行う際に音を出す事が他の方向への引っ掻きジェスチャに比べ困難である事が原因であると思われる。

この根拠として、実験終了後のアンケートにおいて、下方向への引っ掻きジェスチャを行う事が困難であると答えた被験者が12名中4名いた事が挙げられる。



図 6.5: *FN* 率を距離別に表したグラフ

結果を距離別に見ると、距離が少ない場合においてより多く *FN* が発生する事が分かった。*FN* 率を距離別に表したグラフを図 6.5 に示す。今回の実験においては上下方向への引っ掻きジェスチャを行う際に指定した距離が 1-4 の通り、左右方向への引っ掻きジェスチャを行う際に指定した距離が 1-10 の 10 通りであったことから、上下方向は左右方向に比べ短い距離を入力することが多かった。このことが、上下方向への引っ掻きジェスチャの際の *FN* 率が左右方向への引っ掻きジェスチャに比べ高い事の原因の 1 つになっていると考えられる。

6.4.4 引っ掻きジェスチャ時における打鍵率

結果

被験者が、引っ掻きにジェスチャを行なっている際に、誤ってキーを打鍵したタスク (*FK*) の回数を計測した。計測の際、タスク中の最後の操作音が検出される 0.5 秒以内にキーリリースイベントが発生していた場合、被験者は引っ掻きによる入力を行なっている際に誤って打鍵した事とみなした。その結果、*FK* となる事例の割合は 4.5% であった。この内訳をまとめた結果を表 6.5 に示す。なお *FK* 率とは、全タスクのうち *FK* が発生したタスクの割合を示すものである。

考察

結果を距離別に見ると、距離が大きい場合においてより多く *FK* が発生する事が分かった。*FK* 率を距離別に表したグラフを図 6.6 に示す。

また、結果を方向別に見ると、上下方向に比べ左右方向に引っ掻く際により多くの *FK* が発生する事が分かった。今回の実験においては上下方向への引っ掻きの際に指定した距離が 1-4 の通り、左右方向への引っ掻きの際に指定した距離が 1-10 の 10 通りであったことから、

表 6.5: *FK* 率の内訳 (%)

方向 \ 距離	1	2	3	4	5	6	7	8	9	10	全体
上	1.8	0.0	3.6	1.8	-	-	-	-	-	-	1.8
下	0.0	0.0	0.0	0.0	-	-	-	-	-	-	0.0
左	0.0	0.0	1.8	3.6	5.5	7.3	14.6	9.1	16.4	18.2	7.6
右	3.6	1.8	1.8	0.0	1.8	3.6	1.8	9.1	10.9	7.3	4.2
全体	1.4	0.5	1.8	1.4	3.6	5.5	8.2	9.1	13.6	12.7	4.5

左右方向は上下方向に比べ大きい距離を入力することが多かった。このことが、左右方向への引っ掻きの際の *FK* 率が上下方向への引っ掻き比べ高い事の原因だと考えられる。距離が 1-4 のケースのみを見ると、左右方向への引っ掻きの際の *FK* 率は上下方向への引っ掻きの際の *FK* 率とほとんど変わらない事が分かる。このことから、*FK* 率の違いは入力する方向に依存しているのではなく、距離のみに依存していると考えられる。

入力する距離の大きさに比例してユーザが誤って打鍵する事が多い理由として、ユーザがキートップを引っ掻く勢いが距離の大きさに比例して強くなる傾向にある事が挙げられる。距離の大きさに比例してユーザの指の移動量が大きくなり、短時間にて入力を行うので、ユーザの指の移動速度が上昇する。そのため、入力する距離が大きい場合は小さい場合に比べ、ユーザがキーボードの面に対して水平方向にかける力が強い。ユーザが完全に水平方向にのみ力をかける事は困難であるので、垂直方向にかかる力も必然的に強くなる傾向がある。これにより、入力する距離の大きさに比例して、ユーザが誤って打鍵してしまうケースも増加すると考えられる。

図 6.6: *FK* 率を距離別に表したグラフ

次に、*FK* が発生した際に誤って打鍵されたキーの位置関係を、方向別に示したものを図 6.7 に示す。図 6.7 から、入力する距離が大きい場合に、最後の方に引っ掻くキーを誤って打鍵する事が多い事が分かる。

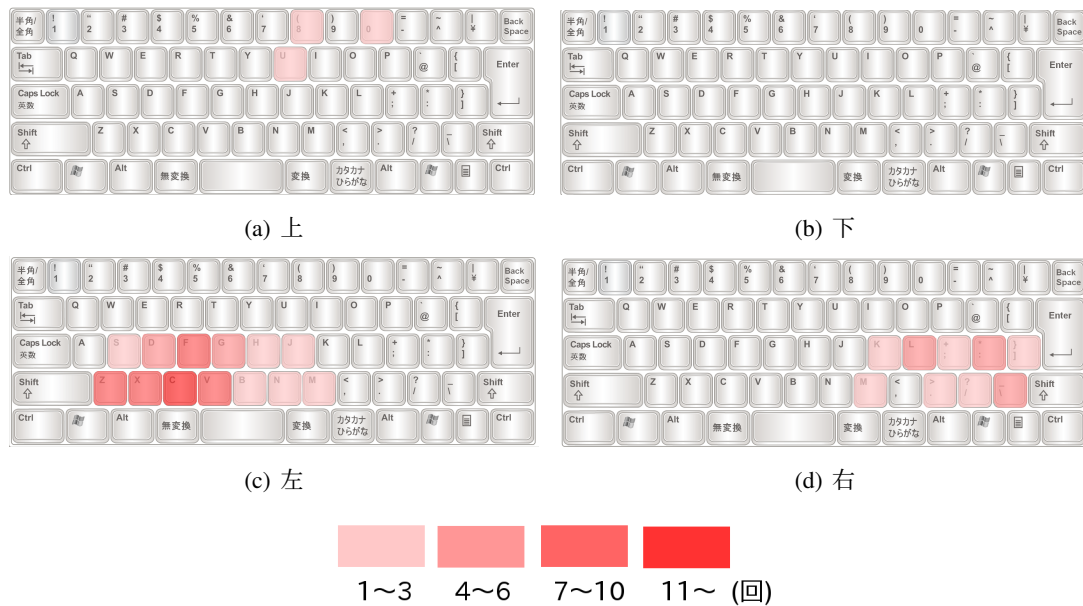


図 6.7: 被験者が引っ掻きジェスチャの際に誤って打鍵したキーとその回数（方向別）

6.4.5 アンケート結果より

本節では、実験後に被験者に対して行ったアンケートの結果より得られた知見について述べる。

使用感

アンケート結果より得られた、本手法の使用感を述べる。12 名中 4 名の被験者は、上下方向への引っ掻きジェスチャに比べ左右方向への引っ掻きジェスチャを行う方が容易であると答えた。上下方向への引っ掻きジェスチャを行う事が困難である理由として、キーが各段ごとにずれている点を挙げた被験者が 1 名いた。また、慣れれば手元を見る事なく意図通りの距離だけ引っ掻きジェスチャを行う事が出来ると答えた被験者が 2 名いた。ある被験者は、距離を入力する際は 3 段階程度に区切ると適切だろうと答えた。

応用例

被験者にこの手法を利用したい場面を尋ねた所、文書作成中に使用したいと答えた被験者が 3 名いた。具体的には、ページの切り替え、タブの切り替え、Undo/Redo 等の機能が挙げられた。また、ブラウザ上におけるタブの切り替え及び進む/戻る、及びアプリケーションの切り替えに利用したいと答えた被験者もいた。さらに、ピアノを弾くアプリケーションに応用したいという被験者が 1 名いた。これは、キートップ上を大きく引っ掻く動作がグリッサ

ンド奏法（ピアノ上において爪を用いて複数の鍵盤を一度に撫でる奏法）の動作に似ている事が理由であると思われる。

第7章 議論

本章では、評価実験の結果を踏まえて Keyboard Clawing についての議論を行う。

7.1 距離の識別

評価実験にて、距離の識別率は許容誤差が0または1に設定した際は低く、2以上に設定した際は高いという結果を示した。また、考察にて正確な距離の入力または識別が困難であることを述べた。以上の理由により、キートップを引っ掻いた距離を入力に用いる際は、識別した距離を次のようにいくつかの段階に区切り、それぞれの段階に入力量または機能を割り当てる事が良いと思われる。

- 上下
 1. キートップ 2 つ分以下（手を動かさず、小さく引っ掻く）
 2. キートップ 3 つ分以上（人差し指以外の指も動くくらい大きく引っ掻く）
- 左右
 1. キートップ 2 つ分以下（手を動かさず、小さく引っ掻く）
 2. キートップ 3～5 つ分（手を動かさず、大きく引っ掻く）
 3. キートップ 5 つ分以上（手を動かして、大きく引っ掻く）

7.2 方向の識別

評価実験にて、方向の識別率は Per-User Test において平均で 68.2%であるという結果が出た。方向の識別率が低い結果が出たのは、方向の識別に用いた音が、操作音の始めのピークのみであったためと思われる。なぜならば、6.4.1 節にて述べたように、ユーザが引っ掻きジェスチャを行う勢いの強弱によってはユーザの指がキートップに触れた時点において発生する音をピークとして検出する場合としない場合とがあるからである（図 6.3b）。

そこで、新たな方向認識アルゴリズムを提案する。方向を識別する際、ピーク検出のための窓において検出された全てのピークに対し、方向の認識を行う。その後、得られた方向の認識結果から、最も多く認識された方向を、この時の引っ掻きジェスチャを行った方向とする。図 7.1 の例においては、検出された 3 つのピークのうち、1 つが下方向への引っ掻きジェ

スチャ、2つが上方向への引っ掻きジェスチャと認識された。この時、引っ掻きジェスチャ全体の方向を上方向と認識する。

また別の方向認識アルゴリズムの案として、隠れマルコフモデル（HMM）を用いる事により操作音を時系列解析する方法も考えられる。音響信号を短時間にて観測した場合、近似的に定常過程と見なす事が出来る。HMMは、操作音を多くの確率過程のマルコフ連鎖と見なし、そのパターンを推定する事により操作音を識別する。

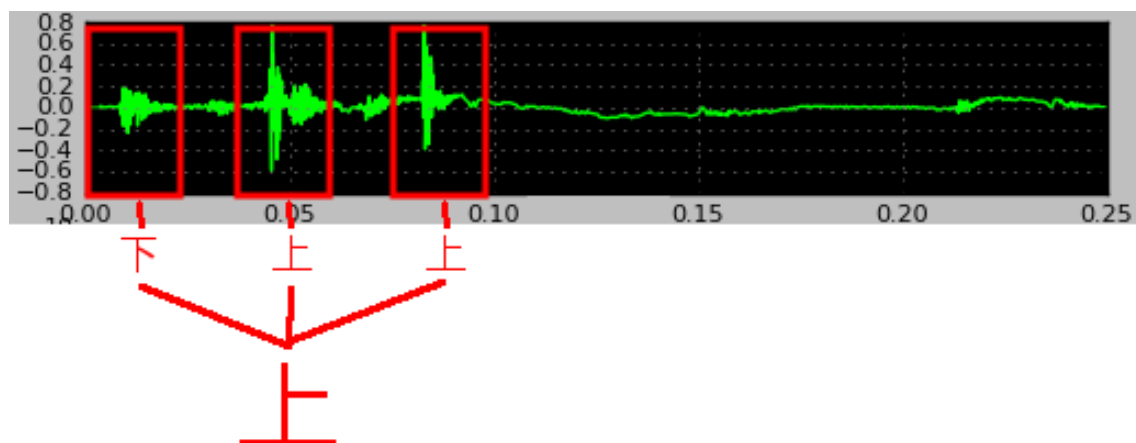


図 7.1: 新たに提案する方向認識アルゴリズム

7.3 ラップトップ型計算機に付属するキーボードへの適用

本手法をラップトップ型計算機上にて適用する事を考えた。そこで、図 7.2 にて示すように、画面サイズが 10.1 型のラップトップ型計算機（NEC 社製 BR340/V）上にピエゾピックアップを配置し、操作音を検出されるか否かを調査した。その結果、実装の章にて述べたデスクトップ型計算機用のキーボードに比べて操作音が小さく、システムが検出しない事例が多かった。これは、ラップトップ型計算機に付属しているキーボードのキーストロークが小さいため、キートップ上において引っ掻きジェスチャを行う際、キートップが弾かれにくい。そのため、キーボードの振動量が小さいため、振動がピックアップに伝わりにくいと思われる。

一方、ラップトップ型計算機に付属しているマイクを用いた方がユーザの操作音を検出しやすい。そのため、本手法をラップトップ型計算機に付属するキーボードへ適用する際は、ピエゾピックアップではなく、計算機に付属しているマイクを用いた方が良いだろう。この際、ピエゾピックアップを用いた場合に比べ、周囲の人の会話や吐息等の環境騒音の影響がより多く出るので、ノイズ除去がより重要になるとと思われる。

7.4 引っ掻きジェスチャに用いる指の本数の識別

本手法においては、引っ掻きジェスチャを行う際に用いる指は1本であった。これに加えて、引っ掻きジェスチャに用いる指の本数の識別も可能になれば、本手法を拡張出来る。この時、ユーザはより多くのショートカットまたはコマンドを引っ掻きジェスチャに割り当てる事ができるため、ユーザの操作の幅が広がると考えた。そこで、著者1名を被験者として引っ掻きジェスチャに用いる指の本数を識別する実験を行った。この実験の際に識別した指の本数及び用いた指を表7.1に示す。実験では、各本数につき20回ずつ、合計80回の1-up clawingを行った。この際、SVMに与えるパラメータとして、cost値を8192.0、gamma値を 2^{-15} とした。なお、cost値とgamma値は、LIBSVMに付属するgrid.py¹を用いて求めたものである。grid.pyは、グリッドサーチによりSVMに与える最適なcost値及びgamma値を求めるプログラムである。

実験の結果から得られた混同行列を表7.2に示す。以下、簡単のために1本、2本、3本、4本の指を用いて引っ掻きジェスチャを行う事を、それぞれ1-finger clawing、2-finger clawing、3-finger clawing、4-finger clawingと表す。混同行列より、1-finger clawingを行った際に4-finger clawingを行ったと予測された割合が0.0%、4-finger clawingを行った際に1-finger clawingを行ったと予測された割合も0.0%であるので、1-finger clawingと4-finger clawingの識別は可能である事が予想される。ただし、今回は1-up clawingのみを用いて実験を行ったため、本手法を拡張出来る事を示すためには、他の距離及び方向の識別との両立が可能である事を示す必要がある。また、全ての本数の識別率の平均値を取ると、57.5%と低かった。そのため、引っ掻きジェスチャに用いる指の本数を、操作音を用いて細かく識別する事は困難であると思われる。

表 7.1: 本数識別の実験に用いる指

指の本数 (本)	1	2	3	4
用いる指 (右手)	人差し指	人差し指+中指	人差し指+中指+薬指	親指以外全て

表 7.2: 本数の識別率 (1-4 本、%)

入力値 \ 予測値	1 本	2 本	3 本	4 本	識別率
1 本	65.0	10.0	25.0	0.0	65.0
2 本	25.0	40.0	30.0	5.0	40.0
3 本	0.0	5.0	90.0	0.0	90.0
4 本	0.0	10.0	55.0	35.0	35.0
平均	-	-	-	-	57.5

¹http://www.csie.ntu.edu.tw/~cjlin/libsvmtools/#grid_parameter_search_for_regression

7.5 机から受ける反射音の影響

評価実験を行う際、キーボードを固い机の上に直接置いて行なった（図 6.1）。そのため、被験者がキートップ上の引っ掻きジェスチャを行った際に生じた振動が机にて反射し、それにより生じるノイズの影響により、方向の識別精度が低下した可能性がある。この問題の対策として、机とキーボードの間にゴムシート等を挟む事により、反射音を軽減する事が挙げられる。

第8章 結論と今後の課題

本稿では、キートップ上の引っ掻きジェスチャによる入力手法を提案し、その評価を行った。本手法では、ユーザはキーボード上のホームポジションに手を置いた状態から1本の指を用いて上下左右のいずれかの方向へ向けてキートップを引っ掻く。この動作により、ユーザがキーボード上のホームポジションから手の位置を動かさずに、方向と量の入力を素早く行う事を可能とした。また評価実験を行い、本手法は大まかな量の入力に用いる事が可能であり、かつ方向を7割程度の精度で識別する事が可能である事を示した。今後の課題として、方向と距離の識別精度の向上と、実際の使用場面における従来手法（マウスやショートカットキー等）との入力速度の比較が挙げられる。

謝辞

本論文の執筆にあたり、志築文太郎先生をはじめ、高橋伸先生、田中二郎先生、三末和男先生には多くのご助言やご指導を頂きました。特に志築文太郎先生には、研究の方針、論文の執筆、そして私生活に至るまで、丁寧なご指導を頂きました。心より感謝致します。また、インタラクティブプログラミング研究室の皆様にも、大変お世話になりました。日頃から貴重な助言やご意見を頂きました。特に WAVE チームの皆様には、チームゼミでの意見のみならず研究生生活全体に渡っても多くのご意見を頂きました。ここに深く感謝致します。最後に、これまで自分を支えてくれた家族、お世話になった多くの友人達にも心から感謝致します。

参考文献

- [AHT02] Brian Amento, Will Hill, and Loren Terveen. The sound of one hand: a wrist-mounted bio-acoustic fingertip gesture interface. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '02, pp. 724–725, New York, NY, USA, April 2002. ACM.
- [BGV10] Florian Block, Hans Gellersen, and Nicolas Villar. Touch-display keyboards: transforming keyboards into interactive surfaces. In *Proceedings of the 28th international conference on Human factors in computing systems*, CHI '10, pp. 1145–1154. ACM, April 2010.
- [BWY06] Yigael Berger, Avishai Wool, and Arie Yeredor. Dictionary attacks using keyboard acoustic emanations. In *Proceedings of the 13th ACM conference on Computer and communications security*, CCS '06, pp. 245–254, New York, NY, USA, November 2006. ACM.
- [CL11] Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, Vol. 2, No. 3, pp. 27:1–27:27, May 2011.
- [DEWB09] Paul Henry Dietz, Benjamin Eidelson, Jonathan Westhues, and Steven Bathiche. A practical pressure sensitive computer keyboard. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, UIST '09, October 2009.
- [DPPS07] Travis Deyle, Szabolcs Palinko, Erika Shehan Poole, and Thad Starner. Hambone: A bio-acoustic gesture interface. In *Proceedings of the 2007 11th IEEE International Symposium on Wearable Computers*, ISWC '07, October 2007.
- [EM84] Yariv Ephraim and David Malah. Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator. *IEEE Transactions on Acoustics, Speech and Signal Processing*, December 1984.
- [FBFS⁺06] Wolfgang Fallot-Burghardt, Morten Fjeld, C. Speirs, S. Ziegenspeck, Helmut Krueger, and Thomas Läubli. Touch&Type: a novel pointing device for notebook computers. In *Proceedings of the 4th Nordic conference on Human-computer interaction: changing roles*, NordiCHI '06, October 2006.

- [FBKL⁺03] Wolfgang Fallot-Burghardt, Helmut Krueger, Thomas Läubli, C. Speirs, and S. Ziegenspeck. Touch&Type - a novel input method for portable computers. In *Proceedings of IFIP INTERACT03: Human-Computer Interaction 2003*, pp. 954–957, September 2003.
- [HFH⁺09] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The WEKA data mining software: an update. *Explorations Newsletter*, Vol. 11, No. 1, pp. 10–18, November 2009.
- [HH08] Chris Harrison and Scott E. Hudson. Scratch input: creating large, inexpensive, unpowered and mobile finger input surfaces. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, UIST '08, October 2008.
- [HSH11] Chris Harrison, Julia Schwarz, and Scott E. Hudson. Tapsense: enhancing finger interaction on touch surfaces. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*, UIST '11, October 2011.
- [HTM10] Chris Harrison, Desney Tan, and Dan Morris. Skinput: appropriating the body as an input surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, April 2010.
- [HXH12] Chris Harrison, Robert Xiao, and Scott Hudson. Acoustic barcodes: passive, durable and inexpensive notched identification tags. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, UIST '12, October 2012.
- [KM12] Seung Wook Kim and Stefan Marti. Ghost fingers: a hybrid approach to the interaction with remote displays. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '12, May 2012.
- [KSI10] Jun Kato, Daisuke Sakamoto, and Takeo Igarashi. Surfboard: keyboard with microphone as a low-cost interactive surface. In *Adjunct proceedings of the 23rd annual ACM symposium on User interface software and technology*, UIST '10, October 2010.
- [LJJ11] Pedro Lopes, Ricardo Jota, and Joaquim A. Jorge. Augmenting touch interaction through acoustic sensing. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '11, November 2011.
- [MSWHQ08] Roderick Murray-Smith, John Williamson, Stephen Hughes, and Torben Quaade. Stane: synthesized surfaces for tactile input. In *Proceedings of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, CHI '08, April 2008.

- [PLCH02] Joseph A. Paradiso, Che King Leo, Nisha Checka, and Kaijen Hsiao. Passive acoustic knock tracking for interactive windows. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '02, April 2002.
- [Rek03] Jun Rekimoto. ThumbSense: automatic input mode sensing for touchpad-based interactions. In *CHI '03 extended abstracts on Human factors in computing systems*, CHI EA '03, April 2003.
- [ZZT09] Li Zhuang, Feng Zhou, and Doug Tygar. Keyboard acoustic emanations revisited. *ACM Transactions on Information and System Security*, Vol. 13, No. 1, pp. 3:1–3:26, November 2009.
- [綾塚 01] 綾塚裕二, 暦本純一. 簡易なノック位置検出装置を用いた実世界アプリケーション. 第9回日本ソフトウェア科学界インタラクティブシステムとソフトウェアに関するワークショップ (WISS 2001) 予稿集, December 2001.
- [塚田 02] 塚田有人, 星野剛史. Pointing Keyboard: キー入力／ポインティングが可能な入力デバイス. 第10回日本ソフトウェア科学会インタラクティブシステムとソフトウェアに関するワークショップ (WISS 2002) 予稿集, pp. 61–66, December 2002.
- [堀 10] 堀竜慈, 志築文太郎, 田中二郎. タッチパネル面におけるスポイト操作の認識の実現. 情報処理学会第72回全国大会, March 2010.

付録A 評価実験の際に用いた実験同意書及びアンケート用紙

次ページに、評価実験の際に用いた実験同意書及びアンケート用紙示す。

キートップを引っ掻く際に生じる音のデータ収集への協力をお願い

文責：黒澤敏文

この度は実験にご協力頂き、ありがとうございます。本実験の目的は、キートップを引っ掻いた際に生じた音の識別精度を測定する事です。

実験中の様子はビデオカメラにて録画させていただきます。また、実験への参加は被験者の自由意志によるものであり、実験への参加を随時拒否・撤回することができます。

実験内容

実験内容を説明します。まず、ディスプレイ上に上下左右いずれかの方向と数字が表示されます。以下、表示された方向と数字を、それぞれ「指定された方向」「指定された数字」と呼びます。次に、キーボードのホームポジションに手を置いた状態から、指定された方向に向けて指定された数字の数だけキートップを引っ掻いて頂きます。上下方向への引っ掻きには右手人差し指を、左右方向への引っ掻きには右手親指を使用して下さい。キートップを引っ掻いた後、完了の合図として「A」キーを押して頂きます。以上のタスクを140回繰り返して頂きます。所要時間は5分程度となります。また、タスク間には自由に休憩を取って良いものとします。

データの取り扱いに関して

本実験において得られた入力データ及び動画は、実験の分析、論文、学内外の発表において使用する可能性があります。その際は個人が特定できないように処理します。

調査に関して、上記内容を理解・同意した上で実験に参加していただける場合は、下部の署名欄に署名をお願い致します。

平成 年 月 日

所属 _____ 署名 _____

説明者：所属 _____ 情報学群 情報科学類 _____ 署名 _____

アンケート

以下のアンケートにご協力下さい(いずれかの項目に○をつけてください)

性別

1. 男 2. 女

年齢

() 歳

利き手

1. 右 2. 左 3. 両方

普段使用する PC (複数回答可)

1. デスクトップ型 PC 2. ノートブック型 PC 3. タブレット型 PC
4. その他()

普段 PC の操作に使用するキーボードの型 (複数回答可)

1. アイソレーション型 2. その他物理キーボード 3. ソフトウェアキーボード
4. その他()

実際にこの手法を用いて入力したい場面や応用アプリケーション案がございましたらご記入下さい

最後に、本実験について感想やご意見等がございましたらご記入下さい

付録B 評価実験のアンケートにて得られたコメント

評価実験のアンケートにて得られたコメントを以下に示す。

実際にこの手法を用いて入力したい場面や応用アプリケーション案がございましたらご記入下さい。

- 文書作成時におけるカーソル移動などの操作。文書作成はキーボードを使用するため、キーボードから手をあまり動かさないで済むこの手法を用いたいと思う。
- スクロールやページ送りなど、大まかな入力を行う時に使いたい。
- マウスカーソルの操作。
- 使用するアプリケーションの切り替えをキーボードから手を離さずに行えたら便利。
- ページ送り。
- 文書作成アプリケーション等、主にキーボード入力を行うアプリケーション上において使用出来たら便利だと思う。Ctrl キーと組み合わせたショートカットキーの押下は、Ctrl キーを押すのが面倒なので。
- アイズフリーに入力が行えるので、マウスジェスチャのようなショートカット機能に使用出来そう。
- タブの切り替え、ページ送り。
- 改行したい時やストレスが溜まっている時。
- ブラウザの戻る/進む。ウィンドウの切り替え
- 文字入力中に用いたい。文字の範囲指定、デスクトップとの切り替え、Undo/Redo、ブラウザのタブ切り替え。オブジェクトが飛んでいくイメージ。
- ピアノを弾くアプリケーション、動物を育成するアプリケーション。

本実験について感想やご意見がございましたらご記入下さい。

- 右及び下方向については入力しやすい、認識の反応も早かったが、左及び上方向については入力しづらく、認識が遅かったと感じた
- 下方向の入力が認識されにくかった。入力する距離が大きいと音が響くので、(他人への迷惑を考えると) 静音環境下における使用には向かないと思った。入力する距離が小さいときは普段キーボードを使用している時と似たような音なのであまり気にならない。
- ピアノを弾くような楽しさがあった。
- キーをある程度押し込まないと音が出にくかったため、実際に使用した際にキーイベントが発生しそう。
- 引っ掻きジェスチャに用いる指を統一しないと途中で分からなくなってしまう。親指を用いた操作はよく行えたが、人差し指を用いた操作は難しかった。
- 指サックのようなものを付けても良いと思う。
- 初めのうちは距離を気にしながら入力していたが、後半は距離を数えるのが面倒になったのと、慣れてきておおよその程度動かせばいいか分かるようになったので、キーボードに目を向けていなかった。どの程度正確に認識されたのかが気になる。
- がんばってください。
- 距離 10 を入力するのが気持ちいい。入力が合っているのか否かは分からないが。
- 上下キーの並びが均等でないので、上下方向への引っ掻きジェスチャが行いにくい。
- 左右方向については指の腹がキーボードに触れるので、どれだけの距離を引っ掻いているのかがなんとなく分かった。上下方向についてはあまり分からなかった。引っ掻くキートップの個数で分割するのではなく、大/中/小の3段階程度に分割すると良いと思った。
- シフトキーを邪魔に感じた。