

単一加速度センサによる入力キーの判別

小野田 敦斗[†]梅澤 猛[‡] 大澤 範高[‡]千葉大学工学部総合工学科情報工学コース[†]千葉大学大学院工学研究院[‡]

1 はじめに

キーボードを使った文字入力、入力したいキーを指で押し込んでから離す打鍵動作を繰り返すことで行われる。Liu らは、手首に装着したスマートウォッチの内蔵センサによって入力された単語の推定を試みている [1]。キーボードをいくつかのエリアに分け、音響データと加速度データを組み合わせることでのどのエリアのキーが入力されたかとその順序を推定し、同じパターンを示す単語を入力候補としている。

本研究では、キーボード本体にセンサを取り付けることで打鍵によって生じる微細な振動を分析し、それに基づいて 1 打鍵単位での入力キー特定を試みる。入力キーを特定するためには、次の 2 段階の処理が必要となるが、本稿では特に 2. の振動データを使ったキー推定部分に着目し、1. の区間検出についてはキーボードの入力信号を使って特定を行う。

1. 打鍵を行っている区間の検出
2. 当該区間の振動データを基にした入力キーの推定

キーボードの所定の位置に 3 軸 MEMS 加速度センサを取り付け、入力時の微小振動を 3 次元加速度データとして記録する。得られたデータから 1 打鍵ずつの区間データを抽出して集め、畳み込みニューラルネットワーク (CNN) を用いて入力キーの判別を行う。

2 打鍵による振動データの収集

実験において加速度センサには Analog Devices 製、EVAL-ADCM の ADcmXL3021 3 軸振動センサを使用した。サンプリング周波数は、最大で 220,000 samples/s である。また、記録できる最小単位の加速度は、重力加

速度を $1g$ とすると、 $0.001908g$ である。キーボードには図 1 に示す ELECOM 製、TK-FCM084BK を用いた。このキーボードはキーピッチ 19mm のメンブレン方式であり、安定したガラス天板のテーブルの上に直接設置した。加速度センサは、“F5” キーの上部に養生テープを用いて固定した。軸方向は、キーボード下方向を x 軸、右方向を y 軸、垂直手前方向を z 軸とした。

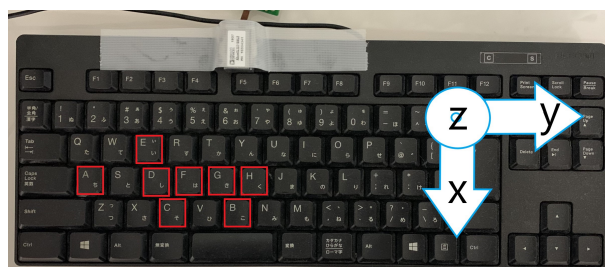


図 1 加速度センサの位置と軸の方向

データの収集方法については、加速度センサの記録時間を 40 秒に設定し、同一のキーを約 1 秒間隔で打鍵し続け、その時の x , y , z 軸それぞれに対する振動データと打鍵時刻を記録した。打鍵時刻の記録は、python の pynput を用いてキー入力を検知した時刻を記録し、打鍵時刻とした。サンプリング周波数は、用いる加速度センサの最大値の 220,000 samples/s で収集した。次に、収集した振動データから、それぞれ 1 打鍵分の区間を含む範囲を抽出し全打鍵分の振動データを作成した。このときの抽出範囲は、事前に振動データの波形と打鍵時刻の関係を調べて確認された両者の若干のタイムラグを考慮して、1 打鍵分の波形をカバーできる範囲として打鍵時刻とその前後を含む 0.521 秒間とした。このため、1 打鍵分の振動データは x , y , z の 3 軸分それぞれ 114,688 (samples) の値を含むことになる。

加速度センサから得られたデータの例を、図 2 に示す。実際には、重力により z 軸には本来 $1g$ の加速度が発生しているが、本データおよび実験ではすべての軸に対して、キーボード上に設置して打鍵をしていないときに観測される加速度を初期値とし、実際に観測された加

Keystroke Recognition of Multiple Keys Using Single Accelerometer

[†] Atsuto Onoda, Department of Information Engineering, Faculty of Engineering, Chiba University

[‡] Takeshi Umezawa, Noritaka Osawa, Graduate School of Engineering, Chiba University

速度から初期値を減算している。0.16 秒前後で観測されている大きな振幅変化は、キーを押し込む際に発生する振動によるものであり、0.34 秒前後で観測されている振幅変化は、手を放す際に発生する振動によるものである。本研究では、図 1 中の赤棒で示した“a”，“b”，“c”，“d”，“e”，“f”，“g”，“h”の8つのキーについて、データを収集した。

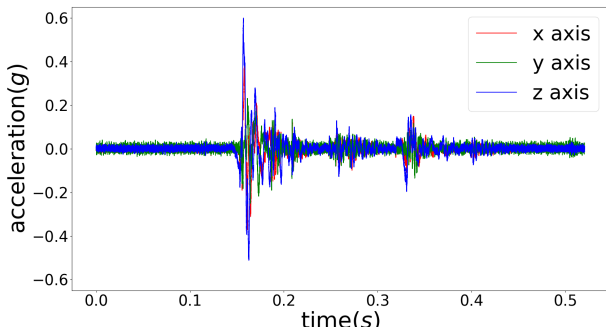


図 2 “a” キーを打鍵したときの各軸に対する加速度

3 CNN によるキー種別の判定

収集したデータを，CNN モデルで学習する。8つのキーについてそれぞれの学習データは，訓練データ 1,600，検証データ 180 とし，テストデータを 97 とした。CNN モデルの中間層は，畳み込み層とプーリング層を交互に重ねた 12 層で構成した。プーリング層については，最初の 11 層には MAX プーリングを，最後の 1 層には GlobalAverage プーリングを採用した。また過学習対策のため，4，8，12 層目のプーリング層の前に確率 $p = 0.4$ でドロップアウト層を加えた。最後に全結合層を加えている。活性化関数には ReLU を用いた。

表 1 CNN の訓練時のハイパーパラメータ

最適化アルゴリズム	ADAM
損失関数	多クラスクロスエントロピー
batch サイズ	64
epoch 数	1000
weight decay	0.0001

訓練時のハイパーパラメータを表 1 に示す。過学習を抑制するため，weight decay の正則化項の導入も行った。表 2 に，テストデータに対する混同行列を示す。検証用データに対する正解率は 0.951，F 値は 0.951，テストデータに対する正解率は 0.729，F 値は 0.727 であった。

表 2 テストデータに対する混同行列

		予測								再現率
		a	b	c	d	e	f	g	h	
真	a	96	0	0	0	1	0	0	0	0.99
	b	0	92	0	0	0	0	5	0	0.95
	c	0	7	69	19	0	1	1	0	0.71
	d	2	0	1	90	0	4	0	0	0.93
	e	2	2	0	5	37	5	45	1	0.38
	f	3	0	0	9	3	76	6	0	0.78
	g	0	0	0	1	1	30	64	1	0.66
	h	0	1	0	2	0	17	35	42	0.43
適合率		0.93	0.90	0.99	0.71	0.88	0.57	0.41	0.95	

誤って特定の別のキーだと 10 回以上予測したケースが 5 通りあった。“h” → “g”，“g” → “f”，“c” → “d” の 3 つの誤った回数は，それぞれ 35 回，30 回，19 回であるが，いずれも隣り合わせのキーである。また 17 回の誤った “h” → “f” ケースについても間に “g” キーを挟んではいるが，“f”，“g”，“h” が隣り合わせで並んでおり，“h” → “g”，“g” → “f” の誤りが多い。一方で，1 番誤りが多かったのは “e” → “g” で 45 回であるが，これは比較的離れている。

4 おわりに

本研究では，キーボード本体に加速度センサを取り付け，8つのキーを入力する際の打鍵による振動データを収集して，CNN モデルによるキー判別を行った。テストデータに対する正解率は 0.729 であり，実用的なキー入力の見点からは十分とは言えない結果となった。

判別精度向上のために，Transformer などの他の深層学習を用いたモデルの検討や，データ拡張の検討を行う必要がある。本研究では 8つのキーのみを対象としたが，他のキーも含めた時の精度についても調査することが今後の課題である。

参考文献

[1] Xiangyu Liu, Zhe Zhou, Wenrui Diao, Zhou Li, and Kehuan Zhang. When good becomes evil: Keystroke inference with smartwatch. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1273–1285, 2015.