

次世代ペン型入力インタフェースに関する研究

豊住 尚弥 (15811064)

ロペズ研究室

1. はじめに

著しい ICT の発達で新しいデバイスが増加し、システムやインタフェースが複雑化している。近年ではスマートフォンの台頭によりタッチパネルでの操作が日常的となった。しかしその操作に慣れないユーザが一定数おり、慣れているユーザでも扱いが煩雑になっている。そこで本研究では筆記動作に着目し、直観的かつ柔軟なデジタル操作が可能な次世代ペン型インタフェースの開発を目的とする。これまでにペン筆跡によるメール送信システムの提案、構築を行ってきた [1]。しかし機能が限定的であり、ユーザが意図しないコマンド判定がされるなどといった問題があった。そこで機能の改善・強化を行い、ユーザビリティの向上を図る。また、従来のシステムでは筆跡を取得するためにペン先取得モジュールが必要であり、携帯性に問題がある。加速度センサと角速度センサからペン先の位置を推定する手法が斎藤 航氏ら [2] によって提案されているが、精度としては高くない。そこでこれとは異なる手法として、ひずみゲージ用いてペン筆跡を推定する手法を提案する。

2. ペン筆跡によるメッセージ転送システム

2.1 システム構成

ペン型入力インタフェースとして、ペンを製する airpen を利用する (図 1)。airpen はペン本体と受信モジュールで構成され、受信モジュールと携帯端末を Bluetooth 接続し、ペン先の位置座標、紙との接触状態をリアルタイムで端末に送信する。

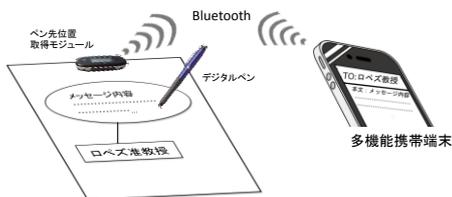


図 1: システム構成 ([1] より)

2.2 メッセージ・コマンド判定アルゴリズム

メッセージコマンド (MC)、宛先コマンド (AC)、送信コマンド (SC)、ツイートコマンド (TC) とメッセージの判定アルゴリズムについて説明する。筆跡データを取得してからコマンドまたはメッセージと判定するまでのフローを図 2 に示す。まず、閾値以上の時間/距離で描かれ、閉図形であるストロークを抽出する。間引き処理ではコマンド判定に必要最低限の座標を抽出する。角度計算では各座標の隣接点とのなす角 θ_i を計算する。コマンド候補判定 2 で θ_i の分布に着目する。 $0 \leq \theta_i \leq 60$ の個数 (N_s)、 $60 < \theta_i < 120$ の個数 (N_r)、 $120 \leq \theta_i < 180$ の個数 (N_o) を算出し、 N_s, N_r, N_o の合計を N_{all} と定義する。 $3 \leq N_s \leq 5$ かつ $N_o = N_{all} - N_r$ ならば AC、 $N_o = N_{all}$ ならば MC、それ以外をコマンドではなく Message と判定する。

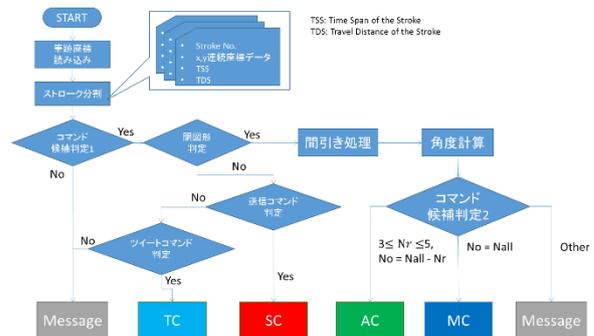


図 2: アルゴリズムフロー

2.3 評価実験

楕円 2 つ、長方形 6 つ、星型 2 つ、台形 3 つを用いて評価実験を行った。その結果を表 1 に示す。改良の結果、星型や台形の誤判定が減少したことで正解率が 92% 以上となり、精度の向上が示された。

表 1: 各図形のと判定結果と正解率

	c1	c2	r1	r2	r3	r4	r5	r6	s1	s2	t1	t2	t3
$N_s(0^\circ - 60^\circ)$	0	0	0	0	0	0	0	0	5	4	1	0	1
$N_r(60^\circ - 120^\circ)$	0	0	4	4	4	4	4	4	0	1	0	4	0
$N_o(120^\circ - 180^\circ)$	23	35	12	12	12	10	12	8	11	26	15	11	15
判定	改良前	MC	MC	AC									
	改良後	MC	MC	AC	AC	AC	AC	AC	M	M	M	M	AC
正解		MC	MC	AC	AC	AC	AC	AC	M	M	M	M	M
正解率	改良前	8/13(61.5%)											
	改良後	12/13(92.3%)											

3. ひずみゲージを搭載したペンデバイスの開発

3.1 システム構成

ひずみゲージ 2 基を直角に接着したリフィルとそれを覆う外装の写真を図 3 に示す。その値から筆跡を復元するまでのフローを図 4 に示す。まずリフィルの歪みをホイットストーンブリッジの出力電圧として取得する。その後 Arduino で A/D 変換を行い、デジタルデータを PC 転送する。最後に PC で電圧値を解析し、描画を行う。

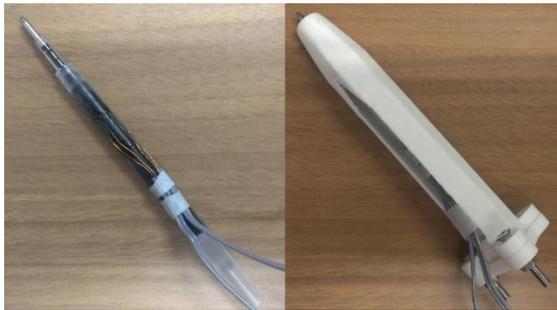


図 3: ひずみゲージを接着したリフィル(左)

3D プリンタで製作したペン外装(右)

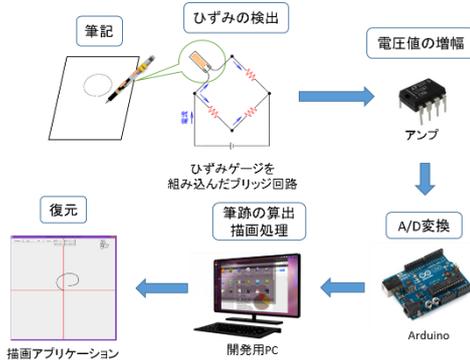


図 4: ひずみゲージによる筆跡復元フロー

3.2 筆跡位置取得フロー

Arduino からデータを受け取ってから描画するまでのフローを図 5 に示す。まず、最初に取得した電圧値(V_x, V_y)を基準値(V_{x_ref}, V_{y_ref})とする。この基準値と現在値の差分をとることで、上下と左右それぞれの進行方向を判断し、これを差分値($\Delta V_x, \Delta V_y$)とする。次に閾値 T 以下の差分値をノイズと判断しフィルタリングする。次に上下・左右の移動量を補正係数 α, β で補正し、移動値(dy, dx)を算出する。これはゲージ伸縮で感度が異なる場合があるためである。最後に現在のペン先位置座標と移動値を足すことで新たな位置座標(x, y)を求め、描画処理を行う。

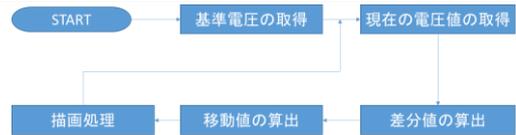


図 5: 筆跡推定アルゴリズムフロー

3.3 評価実験

円形、四角形、三角形の 3 種を各 8 つ、計 24 のデータを用いて精度実験を行った。図 6 に各図形の代表例、表 2 に各図形と全筆跡の平均を示す。黒線が実際の筆跡、青線が今回推定した筆跡である。図形の大きさを縦横 1 としたとき、円形で約 35%、四角形で 24%、三角形で 44%、全体で 34%の誤差が検出された。

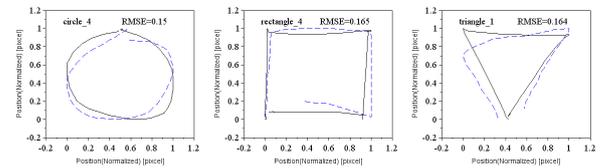


図 6: 各図形の代表例

表 2: 各図形と全筆跡の平均 RMSE

円形	0.351286
四角形	0.237286
三角形	0.436286
全筆跡	0.341619

4. おわりに

ペン筆跡によるメッセージ転送システムにおけるコマンド認識アルゴリズムの改良とひずみゲージによるペン先位置推定のシステムの構築を行った。コマンド認識で精度の向上がみられた一方、ペン先推定はまだ精度としては低い。今後の課題として、補正係数の自動キャリブレーションやひずみ検出回路の最適化、アルゴリズムの改善を行うことによって推定精度を高め、最終的にシステムの統合を目指す。

参考文献

- [1] 高橋淳二、中里画、豊住尚弥、ロペズギョーム, "ペン型入力インタフェースを用いた筆記作業のみによるメールの送信", 本機械学会, 2014 年
- [2] 齋藤航、中井一文、出江幸重、江崎修央, "センサフュージョンを用いた電子ペンの筆記軌跡復元", 情報通信学会, 2014 年