

小型タッチスクリーン上の効率的なタイルプログラミング環境の設計 Design of Tile Programming Environment for Reducing Operating Time on Small Touch Screen

磯脇 直人[†]

Naoto Isowaki

梅澤 猛[†]

Takeshi Umezawa

大澤 範高[†]

Noritaka Osawa

1. はじめに

スマートフォン用のアプリケーションをスマートフォン上で作成できれば、開発と使用を端末で完結することができる。スマートフォン上の主な操作手段である画面へのタッチ操作は、長いテキストの入力には向かないため、プログラムコードを入力するには工夫が必要である。既存のスマートフォン向けのプログラミング環境においては、プログラミング用にカスタマイズされたソフトウェアキーボードや予測変換機能を使って入力を補助する例があるが、結果的に入力が必要な文字数は汎用の開発環境と変わらない。そこで本研究では、少ないテキスト入力でアプリケーション作成が可能なタイルプログラミングに注目し、スマートフォンのような小型タッチスクリーン上で少ない操作時間でプログラミングできる環境について検討を行う。

2. タイルプログラミング環境

対象とするタイルプログラミング環境として、App Inventor を用いる。App Inventor は、ボタンなどの部品を編集スペースに配置してアプリケーション画面のレイアウトを作成し、タイルプログラミング形式で Java のプログラムを組む、Android 用アプリケーション開発環境である。

図 1 は操作画面の例を示す。本論文では、タイルの種類を選択する左側の矩形領域を”Blocks”，タイルを選択する中央の矩形領域を”List”，タイルを配置してプログラムを組む右側の矩形領域を”Viewer”と呼ぶ。App Inventor でプログラムを作成する時には、Viewer に配置するタイルごとに、Blocks の項目をタッチして List を表示させる操作、List 中の必要なタイルを選択する操作、Viewer にタイルを配置する操作の 3 つの基本操作が必要である。



図 1 App Inventor の操作画面

3. 操作時間の導出

3.1 全体の操作時間の導出

小型タッチスクリーンのように画面サイズが一定の制約下で操作する場合、タイルの大きさに起因する誤操作や、表示領域の狭さに起因するスクロール操作により追加の操作

[†] 千葉大学大学院融合科学研究科 Graduate School of Advanced Integration Science, Chiba University

時間(遅延時間)が発生し、プログラミング作業全体の操作時間 T が増加する。そこで、タイルの大きさを変化させた時のプログラミング作業をシミュレーションして T を算出し、 T が最小となるタイルの大きさを求める。

2 節で述べた 3 つの基本操作に要する時間をそれぞれ T_1 , T_2 , T_3 、誤操作による遅延時間を T_e 、スクロール操作による遅延時間を T_s とすると、 T は式(1)で求められる。

$$T = \sum_{n=1}^N \left(\sum_{k=1}^3 T_k + T_e + T_s \right) \quad (1)$$

N は配置するタイルの総数である。

3.2 基本の操作時間の導出

配置するタイルの種類・座標と画面の拡大倍率が決定している時、タイルを選択・配置するためにタッチする座標と指の最短移動距離が一意に定まるため、 T_1 , T_2 , T_3 は Fitts の法則[1]と村田らの研究[2]より式(2)(3)で求められる。

$$ID(z) = \log_2 \left(\frac{\sqrt{(x_f z - x_i)^2 + (y_f z - y_i)^2}}{z \sqrt{W_t H_t}} + 1 \right) \quad (2)$$

$$T_k(z) = a + b ID(z) \quad (3)$$

z は画面の拡大倍率であり、編集画面はそれを z 倍に拡大して表示される。App Inventor で用いられる標準のターゲットの中央座標 (x_f, y_f) 、幅・高さ (W_t, H_t) を基準値とした時、 (x_f, y_f) と (W_t, H_t) は z に比例する。 (x_i, y_i) は指の初期位置であり、最初の操作では $(0,0)$ 、2 回目以降の操作では前の操作が終了した時の指の座標とする。 a, b は操作者の習熟度に応じた定数であり、指を画面から離して移動させる時には (a_1, b_1) 、指を画面に付けて移動させる時には (a_2, b_2) とする。

3.3 誤操作による遅延時間

タイルの大きさの影響を調べるために、誤操作の中でターゲットの小ささが原因で発生する、ターゲット近辺の誤った箇所をタッチする誤操作のみを考慮し、式(4)で求める。

$$T_e(z) = \lim_{n \rightarrow \infty} \sum_{k=1}^n E^k(z) \overline{T_{mp}}(z) \quad (4)$$

E はエラー率で、駒崎らの研究[3]を基に、式(5)で求めた。

$$E(z) = \alpha (z^2 W_t H_t)^\beta \quad (5)$$

$\alpha = 471.01$, $\beta = -0.887$ である。 T_{mp} は本来の意図と異なるエラー状態を復元するための操作時間であり、誤操作が発生した時の操作と誤ってタッチした箇所決定する。例えば、List 上の空白を誤ってタッチした場合 List が閉じるため、Blocks の項目をタッチし再度 List を表示させ List の正しいタイルをタッチする操作が復元操作となり、 $T_{mp} = T_1 + T_2$ となる。 T_e は、エラー率と復元操作時間の期待値から遅延時間を求める。誤操作は 1 回の操作の中で複数回発生し得るため、誤操作が k 回発生した時の遅延時間の総和の極限を遅延時間とする。

3.4 スクロール操作による遅延時間

編集画面全てを一画面に表示できずタッチする座標が画面外にある時、画面のスクロール操作が必要となる。スクロール操作は、ターゲットの無い空白をタッチして指を必要なスクロール量分動かすことで行う。スクロール操作による遅延時間は、式(2)(3)を基に、式(6)(7)(8)で求める。

$$s_x(z) = \begin{cases} x_s - x_{fz} & \text{if } x_{fz} < x_s \\ x_{fz} - x_s - W_d & \text{if } x_{fz} > x_s + W_d \\ 0 & \text{if } x_s \leq x_{fz} \leq x_s + W_d \end{cases} \quad (6)$$

$$s_y(z) = \begin{cases} y_s - y_{fz} & \text{if } y_{fz} < y_s \\ y_{fz} - y_s - H_d & \text{if } y_{fz} > y_s + H_d \\ 0 & \text{if } y_s \leq y_{fz} \leq y_s + H_d \end{cases} \quad (7)$$

$$T_s(z) = \begin{cases} 0 & \text{if } s_x(z) = s_y(z) = 0 \\ a_2 + b_2 * \log_2 \left(\sqrt{\frac{s_x^2(z) + s_y^2(z)}{S}} + 1 \right) & \text{Otherwise} \end{cases} \quad (8)$$

スクリーンには、既にスクロールされている分を(x_s, y_s), スクリーンの画面の幅・高さ(W_d, H_d)をとすると、編集画面の(x_s, y_s)から($x_s + W_d, y_s + H_d$)までの矩形領域が表示される。この時タッチ座標(x_{fz}, y_{fz})がこの矩形領域外にある場合、画面を(s_x, s_y)だけスクロールさせる必要がある。 S はスクロール操作時にタッチする空白の幅であるが、空白の広さが一定ではなく、空白ならば自由な箇所をタッチしてよいことから、文献[4]より充分なタッチ精度を持つ幅 9.0mm とする。

4. 実験

Viewer に配置する各タイトルの種類・座標と、実験で求めた定数パラメータ a, b , 画面の拡大倍率 z から、操作をシミュレーションして T を式(1)から導出し、操作時間が最小となる拡大倍率と、その時のタイトルの大きさを求める。

式(3)(8)に必要なパラメータ a_1, b_1, a_2, b_2 は駒崎らの研究[3]にしたがい実験した結果、指を画面から離して動かす場合は $a_1=212.78, b_1=52.00$ であり(自由度調整済決定係数 0.895153), 指を画面に付けて動かす場合は $a_2=366.53, b_2=81.56$ であった(自由度調整済決定係数 0.997319)。

4.1 シミュレーションする作業

本実験では、ユーザが時間を指定してその時間にアラームを鳴らすプログラムのプログラミング作業のシミュレーションを行う。用いるタイトル数は 64 個。最小の操作回数は 192 回。 $z=1$ の時の操作に用いるタイトルの大きさの平均は 16.99mm^2 , 基本操作における指の移動距離の平均が、基本操作ごとにそれぞれ $37.81\text{mm}, 15.52\text{mm}, 30.04\text{mm}$ である。

4.2 シミュレーション環境

シミュレーションは、スマートフォンに近い 5 インチのタブレット端末(Nexus5)の画面サイズを基に、画面の幅・高さを $109.6*61.6\text{mm}$, 画面の解像度を $1920*1080\text{pixel}$ で行う。また、比較として 7 インチのタブレット端末(Nexus7)の画面サイズに合わせ、画面の幅・高さ $151.7*94.8\text{mm}$, 画面の解像度 $1280*800\text{pixel}$ でもシミュレーションを行う。

4.3 シミュレーション結果

式(1)(3)(4)(8)から全体の操作時間 T 及び T_e, T_s の総和を求めた。スマートフォン想定画面サイズでのシミュレーション結果が図 2 であり、 $z = 1.512$ の時 $T = 145572$ (msec) で最小となり、 $T_e = 48558$ (msec), $T_s = 2620$ (msec) であった。

タブレット想定画面サイズの場合、 $z = 1.120$ の時 $T = 114589$ (msec) で最小となり、 $T_e = 17451$ (msec), $T_s = 2744$ (msec) となった。表 1 に T が最小の時のタイトルの大きさを、表 2 に T が最小の時のエラー率を示す。

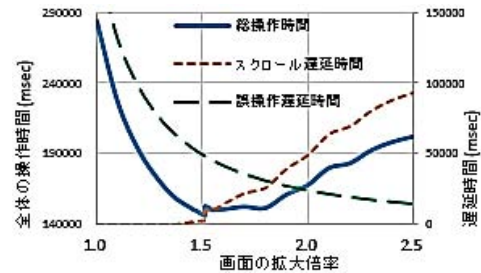


図 2 スマートフォン画面での操作時間と拡大倍率

表 1 T が最小の時のタイトルの大きさ (mm^2)

シミュレーション環境	最小	平均	最大	標準偏差
スマートフォン画面	13.17	38.84	182.77	31.63
タブレット画面	31.15	91.91	432.45	74.84

表 2 T が最小の時のエラー率

シミュレーション環境	最小	平均	最大	標準偏差
スマートフォン画面	4.64%	25.52%	47.87%	12.71
タブレット画面	2.16%	11.89%	22.30%	5.92

4.4 考察

スマートフォン画面での平均エラー率が 25.52% と高く、4 回に 1 回誤操作が発生するのでは、快適に操作できているとは言い難い。タブレットにおいてもエラー率 11.89% と高い値を取っていることから、 E が実際より大きいことが原因として考えられ、 E の導出方法の再検討が必要である。また、 T_e が実際より小さいため、 T_e が T に与える影響が小さくなっている可能性も考えられる。その場合、 T_e が小さくなるのは、 T_{mp} を過小評価していることが要因と考えられる。 T_{mp} は最小の操作時間から求めたが、実際には誤操作であることを認識して修正するまでの時間が生じると予想されるため、 T_{mp} の導出に用いる a, b を、最小の操作時間の時とは別に決定する必要があると考えられる。

5. 今後の展望

今後は、他のプログラムの場合におけるシミュレーションの結果の変化を調査するとともに、求めたタイトルの大きさで実際の操作時間が短縮されるかを実験するとともに、全体の操作時間が小さくなるタイトルの形状・位置関係を探り、全体の操作時間が小さいタイトルプログラミング環境の設計を提案していく。

参考文献

- [1] Fitts, P.M., "The Information Capacity of The Human Motor System in Controlling The Amplitude of Movement", Journal of Experimental Psychology, Vol. 47, No. 6 (1954), pp. 381-391.
- [2] 村田 厚生, マウスのパフォーマンスモデルについて, 電子情報通信学会論文誌, Vol.79, No.9, pp.1645-1648 (1996)
- [3] Masanobu K, and Masanori I, "A Study of Human Pointing Features on Touch-Screens - Age Effect, Difference by Gender, and Difference in Direction", Journal of Robotics and Mechatronics, Vol.19, No.6(2007)
- [4] Microsoft CORP., "Touch interactions for Windows (Windows)", <http://msdn.microsoft.com/en-us/library/windows/apps/hh465415.aspx> (参照日 2014 年 6 月 30 日)