

ジェスチャ操作で入力する絵文字ソースの携帯プログラミング

瀬戸 優之[†] 宮下 芳明[†]

本稿では携帯端末上でプログラムを編集・実行するシステムを提案する。Android アプリケーションとして実装した提案システムでは、絵文字によるソースコード記述が行われ、一画面あたりの情報量を増やしている。また、プログラム入力インタフェースでは、ジェスチャ認識による関数入力が行える。プログラムをメールとして送信するとインタラクティブなデコレーションメールとなる。

Programming on Mobile Devices with Pictogram Source Code and Gesture Operation

HIROYUKI SETO[†] and HOMEI MIYASHITA[†]

In this paper, we proposed a system to edit and execute programs on mobile devices. We developed a prototype system as an Android application. It shows the source code using pictograms to increase the amount of information on screen. Moreover, the user can input functions and parameters with gesture operation on touch screen. By sending the program to other mobile devices via e-mail, it becomes an interactive decorated mail.

1. はじめに

情報技術の発展に伴い、携帯電話の普及と機能の進化が著しい。スマートフォンにおいては、OS ごとにアプリケーションをダウンロードできるシステムを備え、数百～数十万というアプリケーションが登録されている。登録されている数こそ多いものの、広い分野に質の高いアプリケーションが十分に揃っているとは言いがたく、まだまだ拡大の可能性があると考えられる。

スマートフォンユーザの多くは、スマートフォンのことを携帯電話としてではなく、携帯するコンピュータとして認識しているという¹⁾。当初スマートフォンはビジネス向けに販売されてきたが、高機能やカスタマイズ性から現在は一般の利用者向けに売られている²⁾。外出先でも高速回線に常時接続しているというメリットが大きく、電源を入れなければネットワークに接続できないPCより有利である。しかし、ユーザがコンピュータと感じるほどに高度な環境が整っているにも関わらず、スマートフォンをプログラマが開発環境として利用することは少ない。

近年では、携帯キャリアの各社がデコメと省略される cHTML を用いた装飾可能なメールサービスを提供

している。これは、アニメーションに対応したメールの送受信を可能としたものであり、アニメーション絵文字をメール文章中に挿入することやメールの文字色の変更を行うことができる。しかしながら、デコメは基本的にテンプレートを用いることが前提とされており、ユーザ自身が一から作り上げるものはほとんど存在しない。それどころか、ボタン操作に反応するインタラクティブな機能の実装もされていない。そこで筆者らは、インタラクティブなプログラムを簡単に作成することができ、かつそのプログラムをメールによって送受信することができるシステムを提案する。

プログラム入力インタフェースでは、絵文字によるソース表示やジェスチャ認識による関数入力といった機能を備え、メールによって受信したプログラムの編集も可能である。またコンパイラによって、ユーザが作成したプログラムはもちろん、メールによって受信したプログラムを実行することが可能である。

2. 言語仕様

提案するシステムでは、絵文字によるソースコード記述が行われ、すべての関数が絵文字と数字で表示されるようになっている。そのため表示されるプログラムの文字数が減り、一画面あたりの情報量を増やすことが可能となっている。

現在は if 文、for 文、直線を描く、円を描く、四角

[†] 明治大学 理工学部 情報科学科

Department of Computer Science, Meiji University.

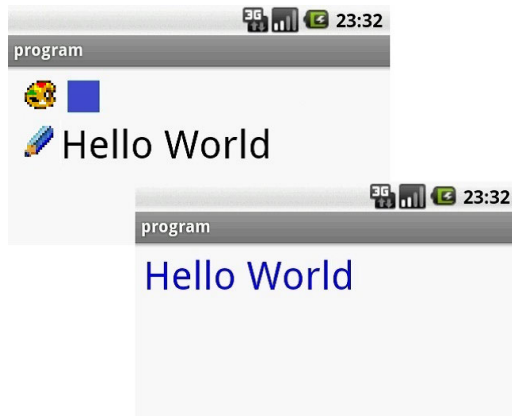


図 1 Hello World のソースと実行結果

を描く，文字を書く，画像を表示するといった関数を実装している．図 1 の上部は「Hello World」という文字列を出力するプログラムのソースコードである．最初の [パレットの絵文字] は文字を書くことを示し，次の [青い四角] が文字色の設定を行う関数となっている．つまりこの場合，文字色が青であることを示している．このように，提案システムではキャンパスや文字の色を設定する際，プログラム内で書かれた色が四角形で出力されるようになっている．これにより，直感的に色を判断することが可能になる．続いての [鉛筆の絵文字] は，先ほどと同様文字を書く関数であるため，その後にくる文字列は引数となる．図 1 の場合は「Hello World」が引数である．また，図 1 の下部はプログラムの実行結果であり，青色で“Hello World”という文字列が出力される．

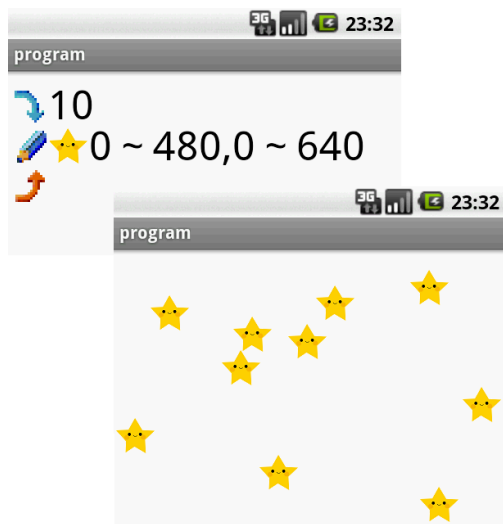


図 2 画像と for 文を用いたプログラムのソースコードと実行結果

図 2 は for 文によって画像を表示する例である．下向きの矢印は for 文の開始を表しており，10 という数

字が 10 回繰り返すことを示している．2 行目の最初の絵文字では画像ファイルの利用をすることもでき，その場合は直後に画像のサムネイルが表示されるようになっている．この関数の引数は 2 つあり，画像を描画する x 座標と y 座標になっている．2 つの数字間に「~」を入れるとその間のランダムな数字を生成するようになっている．よってこの画像は x 座標が 0 から 480 の間，y 座標が 0 から 640 の間に表示される．最後の上向きの矢印は for 文の終了を表している．図 2 下部はプログラムの実行結果画面であり，星の画像がランダムな位置に 10 個表示されている．

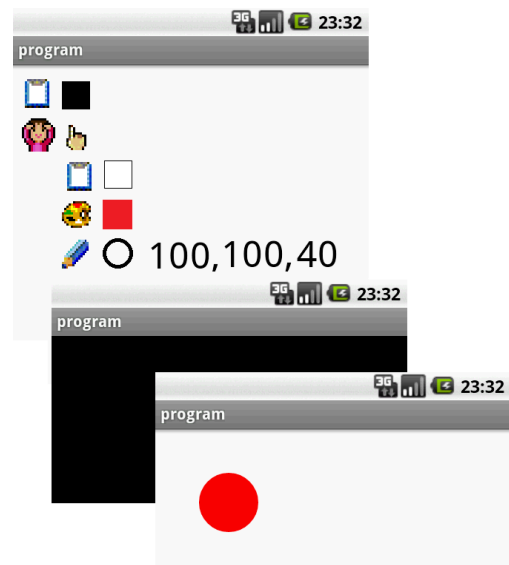


図 3 タッチイベント用いたプログラムのソースと実行結果

図 3 のソースコードでは，最初の行でキャンパスの色を黒に設定している．2 行目の人が丸印を作っている絵文字が if 文となっており，その次の手の絵文字はタッチ操作が行われているかを調べる関数である．まずキャンパスを白に設定している．次に 4 行目で描画色を赤に設定し，5 行目は円を描く関数を示している．引数は中心の x 座標，y 座標と半径になっており，この場合 100,100 の座標に半径 40 の円を描くことを示している．実行すると，最初は図 3 の (左) の画像のように黒い画面となるが，画面に指を触れると図 3 の (右) のような日の丸が表示される．

3. システム

本稿では，プログラムを入力するインタフェースとコンパイラを Android アプリケーションとして構築した．また，Android のインスタンス機能によってメールアプリケーションと連携しているため，PC で作成

したプログラムを受信して編集することや、作成したプログラムを添付ファイルとして送信することができる。これにより、いつでもどこでもプログラムを作成・実行することができる。

3.1 プログラム入力インタフェース

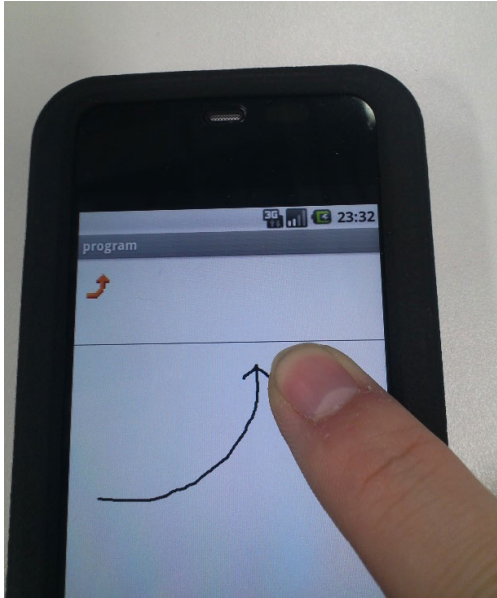


図 4 プログラム入力インタフェース

プログラム入力インタフェースは、関数を絵文字と数字によって表現するだけでなく、ジェスチャによって入力する機能を備えている。図 4 では for 文を記述した様子である。画面は上部分のソースコード表示部と下部分のジェスチャ入力部に分かれている。ジェスチャ入力機能は、画面のタッチ操作の軌道によって文字列が入力されるというものである。ジェスチャ入力部分でタッチ操作を行うとプログラム入力部に入力内容が反映される。これにより筆者らは、ユーザがエラーやタイプミスなどの些細な問題に躓いてしまうことを減少させ、より円滑にプログラムを作成することが可能となると考えており、短時間でより効率的にプログラムを作成できるようになることを期待している。なお、ユーザが入力したい文字列に対応しているジェスチャを忘れてしまった場合は、メニュー駆動型で文字列を呼び出すことができる。またユーザが初期設定のジェスチャが使いにくいと感じた場合は、自分の使いやすいようにカスタマイズ可能である。

3.2 コンパイラ

コンパイラは、携帯端末上でプログラムを実行することができる。プログラム入力インタフェースによって作成されたプログラムをメールで受信した場合も、PC で作成したプログラムを受信した場合も、そのプ

ログラムを実行することができる。プログラムのコンパイルを開始すると、ソースコードを解析し、プログラム入力画面とは別のアクティビティによりプログラムが実行される。

3.3 メールとの連携

さらに提案システムはメールアプリケーションと連携しており、システムによって作成したプログラムを他の携帯端末に添付ファイルとして送信、または受信することができる。そのため、PC で作成したプログラムを受信して編集、実行することも可能である。

3.4 実装

絵文字プログラム言語の実装は Java で行い、文字が入力されるとバックグラウンドでソースコードファイルに関数名と引数が書き込まれる。この関数名は Java に準じたものとなる。これを適宜読み出し、前もって設定した関数と絵文字の関連付けによって変換することで画面上にプログラムのソースコードが出力される。

また、入力インタフェース、コンパイラはともに Java で実装し、Android OS 上で動作する。プログラム入力インタフェースでのジェスチャ入力機能では、Android Gesture API を利用している。あらかじめ登録されたジェスチャに対して入力されたジェスチャの得点付けを行い、設定した値以上であれば同じジェスチャが入力されたとみなしているため、特定のジェスチャからは特定の関数を入力することが可能となっている。関連付けの方法として、現在はプログラムの絵文字を簡略化した図をジェスチャとして採用しており、覚えやすさを優先している。このジェスチャ認識を利用することで、ユーザは関数の入力をスムーズに行うことができる。プログラムコンパイルインタフェースでは、ワンパスでソースコードの解析を行う。ソースコードファイルを 1 行ずつ読み込んでいき、if 文によって文字列毎の処理を分けてゆく。

4. 関連研究

プログラミングに関する研究はこれまでも多く行われている。安村の Programming2.0³⁾ では使いやすいプログラミングについて言及しており、ヒューマンインタフェースの観点から考えるとユーザの記憶に頼るよりもメニュー駆動型の方がユーザ思考であると述べている。本稿第二著者らの HMMMML⁴⁾ および HMMMML2⁵⁾ では多少の入力ミスがあってもコンパイラが好意的に解釈することによって実行が可能になっている。また、命令自体も短くなっているので入力時間の短縮が見込める。原田の Viscuit⁶⁾ はビジュ

アルプログラムとなっており、ユーザの曖昧な記述をコンパイラが許容してくれるものとなっている。コンパイラが独自に判断するので予想外の動きも楽しむことができる。Google 社の App Inventor⁷⁾ では、ブロックを組み合わせることで Android OS 向けのアプリケーションを作成できる。これによりプログラムの知識が一切ないユーザでもアプリケーション開発が可能となっている。

携帯電話を対象としたユーザ支援についての研究として、西本らの MoCoPro⁸⁾ がある。これは、タッチパネルを搭載した端末上でのエンドユーザプログラミングを支援するシステムであり、ECA ルールに基づくビジュアルプログラミングとなっている。これにより、エンドユーザへの理解とプログラミングのしやすさを補助している。小牧らのタッチパネル搭載型携帯端末のための Web 検索クエリ入力支援インタフェース⁹⁾ では、web 上で見つけた単語をドラッグし、その後ジェスチャ操作を行うことで単語の検索の際の下位語を予想して提案してくれる。

また、日常生活にプログラミング技術を導入する試みとして、福地らの Push-pin¹⁰⁾ がある。これは、ホームオートメーションの設定を住人自身が物理ピンの差し替えによって変更することができ、手軽な操作でプログラミングを行うことができるようになっている。Greg らのエンドユーザによる椅子デザインシステム¹¹⁾ ではエンドユーザが椅子の簡単なスケッチ操作を行うだけで、物理シミュレーションや座り心地の確認などができる。ユーザが難しい設定をすることなく結果を即時知ることができる点については本研究と類似している。

このように現代社会において、情報技術の発展と共にプログラミングに対する研究も行われてきている。

5. まとめと今後の展望

本稿では、新しい言語として絵文字ソースのプログラムを開発した。また専用のプログラム入力インタフェースとコンパイラを Android アプリケーションとして実装することにより、携帯端末上でプログラムを作成・実行の可能な環境を提案した。これにより、作業効率の向上やスマートフォンの可能性の拡大を目指した。

本稿で提案したプログラム入力インタフェースでは、使用可能な関数に制限があったため、作成可能なプログラムが限定されてしまっている。今後はこの点を改善し、作成可能なプログラムの拡大を行っていきたい。また実際に提案システムを利用してもらうことで、プ

ログラム作成時のタイプミスの軽減、作業速度の向上といった作業効率の改善や、環境にとらわれない作業の実現が行えたかということについて実験を行いたい。この結果を踏まえて、ユーザにとってより使いいたくなるインタフェースへの改良を目指す。

参 考 文 献

- 1) 株式会社アスキー・メディアワークス:iPhone 利用実態,
<http://asciimw.jp/info/release/pdf/20090626.pdf>
- 2) 根岸正光: モバイル情報アクセス今昔: 情報の消費者主権のゆくえ, 情報知識学会誌, Vol. 18, No. 5, pp.512-515 (2008) .
- 3) 安村通晃: Programming2.0: ユーザ指向のプログラミング, 情報処理学会夏のシンポジウム 2006 (2006) .
- 4) 宮下芳明: プログラミングに対するモチベーションを向上させる新言語 HMMMML の開発, 第 51 回プログラミング・シンポジウム予稿集, pp.57-64 (2010) .
- 5) 中橋雅弘, 宮下芳明: HMMMML2: 超好意的に解釈するコンパイラ. 夏のプログラミングシンポジウム 2010 (2010) .
- 6) 原田康徳: Viscuit: 柔らかい書き換えによるアニメーション記述言語, 情報処理学会インタラクシオン 予稿集, pp.183-184 (2004) .
- 7) Google Inc.: App Inventor for Android,
<http://appinventor.googlelabs.com/about/>
- 8) 西本裕貴, 志築文太郎, 田中二郎: 携帯端末上でコンテキスト依存プログラムを記述するためのビジュアルプログラミング環境, WISS 第 16 回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論文集, pp.145-146 (2008) .
- 9) 小牧大治郎, 荒瀬由紀, 原隆浩, 服部元, 滝嶋康弘, 西尾章治郎: タッチパネル搭載型携帯端末のための Web 検索クエリ入力支援インタフェース, DEIM 2010 第 2 回データ工学と情報マネジメントに関するフォーラム (2010) .
- 10) 福地健太郎, 杉本麻樹, Charith Fernandoy, Shengdong Zhaoy, 稲見昌彦, 五十嵐健夫: Push-pin: ピン型タグを用いたホームオートメーションのためのプログラミングシステム, 情報処理学会インタラクシオン 予稿集, pp.1-8 (2010) .
- 11) Greg Saul, Manfred Lau, 三谷純, 五十嵐健夫: エンドユーザによる椅子デザインシステム, WISS 第 18 回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論文集, pp.17-22 (2010) .