

雛型を用いた携帯電話向け 3D ユーザインタフェース カスタマイズシステムの検討

中西 正洋[†] 尾上 孝雄[†]

[†] 大阪大学 大学院情報科学研究科
大阪府吹田市山田丘 1-5

E-mail: †nakanishi-masahiro@sharp.co.jp, ††onoye@ist.osaka-u.ac.jp

あらまし 携帯電話端末はパーソナル志向が強い機器であり、利用者の趣味・嗜好に応じたユーザインタフェース(UI)のカスタマイズ要望が強い。このような中、携帯電話端末において 3D UI を利用する動きが広がっている。しかしながら、3D UI は、2D UI と比較してコンテンツ作成のコストが高いため、限られた開発費の中、ユーザのカスタマイズ要望を満たす UI のバリエーションを揃えることができない。そこで、UI コンテンツをプログラマの観点での「プログラム要素」とデザイナーの観点での「デザイン要素」に独立して作成し、「プログラム要素」を雛型として、「デザイン要素」をはじめ込むことで、携帯電話上で UI を実現する手法を検討し、UI コンテンツの作成コストの削減を目指す。キーワード 携帯電話、ユーザインタフェース、カスタマイズ、雛型、ユーザエクスペリエンス、3D

Development of Mobile Phone 3D User Interface Customizing System Utilizing Templates

Masahiro NAKANISHI[†] and Takao ONOYE[†]

[†] Osaka University
1-5 Yamadaoka, Suita, Osaka

E-mail: †nakanishi-masahiro@sharp.co.jp, ††onoye@ist.osaka-u.ac.jp

Abstract Customizing the user interface by users' preference is strongly required, as mobile phones are primarily designed for personal use. In addition, mobile phones have been adopting to the 3D UI. However, 3D UI calls for far more development cost than that of 2D UI, and thus it is impossible to provide number of 3D UI contents sufficient enough to meet users' demand for customizing. So, we invented a method that would reduce the 3D UI content development cost, with which we let the engineer create the "program factor" as a "template", while designer separately create "design factor" that would be merged to the created template.

Key words mobile phone, user interface, customize, template, user experience, 3D

1. はじめに

携帯電話端末はパーソナル志向が強い機器であり、利用者の趣味・嗜好に応じたユーザインタフェース(UI)のカスタマイズ要望が強い。このような中、携帯電話端末において 3D UI を利用する動きが広がっている。しかしながら、3D UI は、2D UI と比較してコンテンツ作成のコストが高いため、限られた開発費の中、ユーザのカスタマイズ要望を満たす UI のバリエーションを揃えることができない。

また、携帯電話端末のような組込み機器向けに、コストをかけずに 3D UI コンテンツを作成するための手法に関して、先行研究はない[1]。

そこで、UI コンテンツをプログラマの観点での「プログラム要素」とデザイナーの観点での「デザイン要素」に独立して作成し、「プログラム要素」を雛型として、「デザイン要素」をはじめ込むことで、携帯電話上で UI を実現する手法を検討し、UI コンテンツの作成コストの削減を目指す。

図 1 に、目指す 3D UI カスタマイズシステムの全体図を示す。ユーザは、コンテンツ提供サイトにアクセスを行い、所望の 3D UI コンテンツをダウンロードする。その後、端末の設定画面で 3D UI コンテンツを設定し、利用することが可能となる。

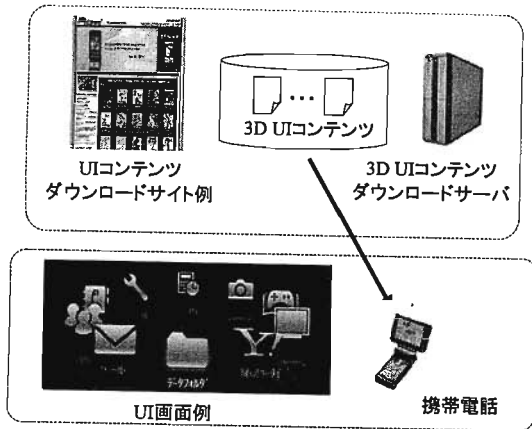


図1 3D UIカスタマイズシステム全体図
Fig.1 3D UI customize system overall view.

2. 関連仕様/プロダクト/研究

本章では、本研究と関連した仕様やプロダクトおよび研究について、いくつか紹介を行う。

2.1 OpenGL/ES

OpenGL/ES (OpenGL for Embedded Systems) [2] は、2003年にOpenGL 1.3のサブセットとして、標準化団体であるKhronos Groupで策定された。主に携帯電話などの組み込みシステムに多く使用される。その後、OpenGL 1.5、2.0のサブセットとして、仕様が更新されている。

2.2 X3D

X3D [3] とは、Web上で3次元グラフィックスを表現するための言語であり、VRML (Virtual Reality Modeling Language) の後継規格となる3次元映像技術である。X3Dでは、VRMLの短所が見直され、軽快に動作するように設計されている。X3Dは2002年7月に草案が公開され、2004年12月に正式に公開された。

2.3 Flash Lite

Flash Lite [4] とは、PC上でデファクトとなっているアドビシステムズ社のFlashをベースに、処理能力の低いモバイル機器向けに開発されたプロファイルである。アニメーションや、インタラクションにも対応しており、アクションスクリプトを埋め込むことで、スクリプト処理も可能である。2004年6月にバージョン1.1が発表されると、携帯電話端末のUIや簡易型アプリの実現手段として利用されはじめた。また、同社からFlash CS3というオーサリングツールが販売されている。また、Flashのアクションスクリプトから3D描画を行うモジュール (Papervision3D) [5] も開発されている。

2.4 ViVidUI

ViVidUI [6] とは、アクロディア社が提供する携帯電話向けミドルウェアであり、ユーザのニーズに合わせて携帯電話のUI一式をカスタマイズすることが可能となっている。携帯電話端末を買い換えたとき、使い慣れたUIを新しい携帯電話端末に引き継ぐ機能を実現するために使われている。また、3Dコン

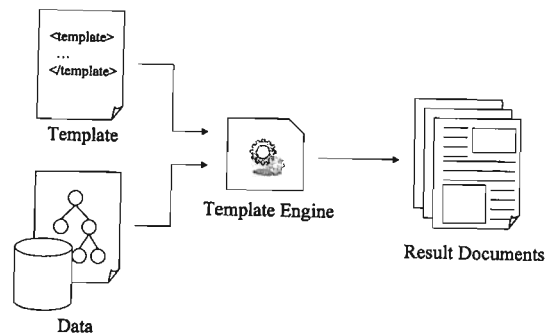


図2 テンプレートエンジンの処理フロー
Fig.2 Flowchart of template engine.

テンツの再生も可能である。

2.5 MVC モデル

MVCモデルとは、複雑/巨大化するGraphic User Interface (GUI) の開発効率を向上させるために、ソフトウェアの構造をModel (M)・View (V)・Controller (C) の3つの部分に分割して、各モジュールを設計・実装する事が行うモデルである [7]。

MVCモデルに基づいたWebアプリケーションプラットフォームとして有名なものに、オープンソースのApache Strutsが存在する [8] [9] [10]。また、UIにおいて、入力と出力は分離することが難しいため、VとCを一体化させた拡張MVCモデルが提案されている [11]。

2.6 関連研究

処理能力の低い携帯電話向けの工夫や、UIカスタマイズに関して、次にあげる関連研究が存在する。春名らは、デジタルAV機器向けに、フォーカス管理を必要とするGUIソフトウェア開発システムを発表している [12]。佐藤らは、携帯電話向けに、WAP (OMA) のEFI機能を利用したUIのカスタマイズに関して発表している [13]。情報提示するシステムの研究として、[14] [15] [16] が存在する。UIのデザインパターン利用に関して次の関連研究が存在する [17] [18] [19]。平田らは、携帯電話向けに、サーバでUIを生成し、端末にアプリケーションとして配信するシステムの研究報告を行っている [20]。また、我々は雛型を用いて2DのUIを生成する研究報告を行っている [21] [22]。なお、大澤らは、検索結果に対し、3D効果を用いた視覚化に関して発表を行っている [23]。

3. 雛型を用いた携帯電話向け3D UIカスタマイズシステム

本研究では次の3点を備えたUIカスタマイズシステムの実現を目指す。

- 3D UIコンテンツの作成コストを削減する
- 3D UIのユーザカスタマイズを実現する
- 携帯電話向けとして実用化する

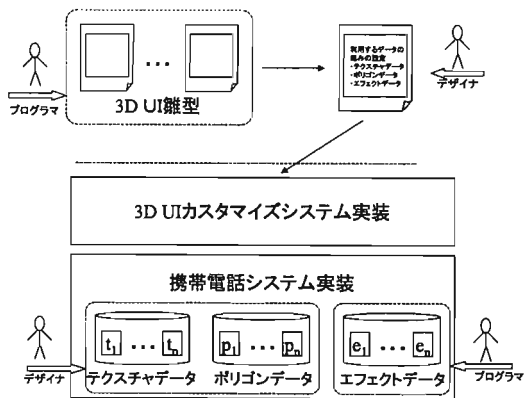


図3 機能ブロック図

Fig.3 Functional block diagram.

3.1 システム設計

3.1.1 雛型モデル

ソフトウェアの開発期間を短縮するために、雛型を利用して再利用性を高め、コンテンツを素早く生産するというデザインパターンを用いたアプローチが広く行われている。

このようなアプローチは、雛型モデルとも呼ばれ、図2に示したように、データと雛型 (template) を合成し、合成結果として文書やソースコードや Web ページなどを出力する。また、このような合成モジュールは、一般にテンプレートエンジンと呼ばれる。

具体例として、Web ページのデザインを制御するために、W3Cにおいて Cascading Style Sheets (CSS) [24] という規格の標準化が行われている。本例では CSS ファイルが雛型となり、HTML ファイルをデータとして扱い、デザイン記述を再利用する。他の例として、 \LaTeX のクラスファイルや、Microsoft社の WORD や EXCEL のテンプレート機能も雛型といえる。

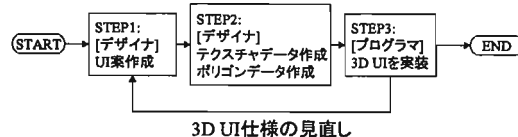
3.1.2 雛型モデルの 3D UI への適用

本研究では、この雛型アプローチを応用し、3D UI の構成要素を雛型とデータの記述に分けて、さらにデータを、エフェクトデータ/ポリゴンデータ/テクスチャデータの記述に分け、お互い独立して作成し、実行時に組み合わせて 3D UI を実現する仕組みを検討する。

ここで、3D UI 空間内でユーザーが選択できるポリゴンオブジェクトとして定義する。このオブジェクトとは、3D UI 空間内に存在する仮想的な立方体の領域である。これらのオブジェクトを選択/実行することで、オブジェクトに関連付けられたアプリケーションを起動することが可能である。なお、オブジェクトは、それ単体では可視状態ではない。

雛型は、スクリプト言語で記述される。我々は、これまで ECMA スクリプト [25] のサブセットである組み込み機器向け軽量/高速スクリプトエンジンを開発しており、これを利用する。雛型では、オブジェクトの配置と、イベントに対する振る舞い処理が記述されている。振る舞い処理として、(1) 選択状態の移動 (2) アプリケーションの起動 (3) エフェクトデータの

■従来の3D UI作成の流れ



■本研究で目指す3D UI作成の流れ

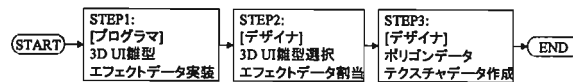


図4 UI作成フロー図

Fig.4 Flowchart of UI construction.

実行などの種別がある。また雛型は、後述のポリゴンデータにテクスチャデータに紐付ける。

エフェクトデータは、OpenGL/ES の関数 API で構成されたマクロ機能である。指定されたオブジェクトに対する動作が、OpenGL の命令セットとして記述される。

テクスチャデータは、画像データである。PNG や JPEG の画像フォーマットが利用可能である。

ポリゴンデータは座標データであり、1点 (x, y, z) で構成される点の集合である。また、構成される図形の座標と、テクスチャの座標の対応関係も本ポリゴンデータに含まれる。

システムの機能ブロック/各データとその作成者を図3に示す。本図では、プログラマは、3D UI 雛型とエフェクトデータを「プログラム要素」として作成する。一方、デザイナーはテクスチャデータとポリゴンデータを「デザイン要素」として作成する。また、3D UI 雛型は、デザイナーによって、端末に内蔵されたテクスチャデータとポリゴンデータとエフェクトデータを組み合わせの情報を付加され、3D UI コンテンツとなる。その3D UI コンテンツは、3D UI カスタマイズシステム実装に入力され、ユーザーに UI としての機能を提供する。

3.1.3 UI コンテンツの開発プロセス

3D UI を作成する場合、デザイナーとプログラマという別の職種の間が、連携して作成を行う。デザイナーは、PC のツールで絵を描くのが得意であり、プログラマはプログラミングが得意である。

端末の UI 開発の一般的な流れと、本研究での UI 開発の流れを図4に示す。

従来の 3D UI の開発では、まずデザイナーが UI 案の絵コンテとなる仕様作成を行い、ポリゴンデータを PC 上のツールで作成し、テクスチャ画像などをグラフィックパーツとして作成する。プログラマが、仕様を参考にしながら 3D UI としてストレスなく動作するように端末上に実装を行う。

UI の開発時には、デザイナーが用意された 3D UI 雛型を選択し、雛型の各オブジェクトにポリゴンデータとエフェクトデータとテクスチャデータを割り当て、3D UI コンテンツを作成する。

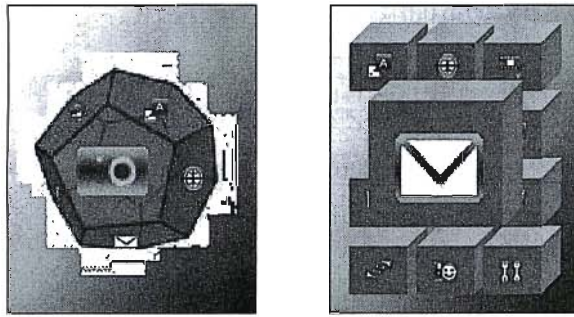


図5 画面例
Fig. 5 Screenshot.

雛型とデータは再利用可能となり、3D UI 作成は組み合わせるだけになる。そのため、3D UI の作成コストを大幅に下げることが可能となる。

3.1.4 データ量の削減

端末で利用するデータサイズを削減することで、端末に搭載するROM/RAMのサイズを削減でき、端末コストを低減できる。一般に3D UIのポリゴンデータとテクスチャデータのサイズは大きい。

本システムでは、差し替え可能な3D UIコンテンツを交換するだけで、UIのバリエーションを増やすことが可能となっている。

例えば、図5のように、3D UIコンテンツを変更するだけで、端末に内蔵されたテクスチャ画像やポリゴン画像を使いまわして、別の操作感を持ったUIをユーザに簡単に提供可能となる。

3.1.5 PCビューア

支援ツールの一つとして3D UIコンテンツをプレビューするためのアプリケーションを開発している。デザイナーは、携帯電話の開発の初期段階でも、PC上で動作確認が可能である。また、利用メモリを事前に知ることもできる。

4. おわりに

本研究報告では、雛型を用いた携帯電話向け3D UIカスタマイズシステムの紹介と、実現のための設計を検討した。

本研究では次の3点を備えたUIカスタマイズシステムの実現を目指す。

- 3D UIコンテンツの作成コストを削減する
- 3D UIのユーザカスタマイズを実現する
- 携帯電話向けとして実用化する

本システムが目指す特長は次の3点である。

現在、PC向けツールで作成した簡単なポリゴンデータのサイズが1MBを越えており、サイズと読み込みに必要な処理時間の両面で、課題となっている。今後は、まずこの課題の解決に取り組む。

同時に実用化のために、3D UIコンテンツ提供サイトの構築と、デザイナー用オーサリングツールの開発も並行して行っていく。

文 献

- [1] D. A. Bowman, E. Kruij, J. Joseph J. LaViola and I. Poupyrev: "3D ユーザインタフェース", 丸善 (2005).
- [2] Khronos Group: "OpenGL ES", <http://www.khronos.org/opengles/>.
- [3] Web3D Consortium: "Web3D Consortium - Royalty Free, Open Standards for Real-Time 3D Communication", <http://www.web3d.org/>.
- [4] Adobe Systems: "Adobe - Flash Lite", <http://www.adobe.com/jp/products/ashlite/>.
- [5] Papervision3D.org: "Papervision3D", <http://www.papervision3d.org/>.
- [6] アクロディア: "VIVID UI", <http://www.acrodea.co.jp/product/ui/index.html>.
- [7] F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad and M. Stal: "Pattern-Oriented Software Architecture: A System of Patterns", John Wiley & Sons Ltd, Chichester, England.
- [8] A. S. project: "Struts", <http://struts.apache.org/>.
- [9] 黒住: "Apache Struts ハンドブック 改訂版", ソフトバンククリエイティブ (2006).
- [10] 中村, 河村, 細川, 菅野, 首藤, 日立ソフトウェアエンジニアリングインターネットビジネス部: "入門 J2EE システム開発 - MVC モデルから Web サービス構築まで", 日経 BP (2002).
- [11] 増田, 笠原: "Smalltalk - 80 における拡張 mvc モデルとその応用", 情報処理学会論文誌, 31, 2, pp. 259-267 (1990).
- [12] 春名, 眞田, 喜納, 角谷, 瀧: "デジタル av 機器向け gui ソフトウェア開発システム", 電子情報通信学会論文誌, J84-D-I, 6, pp. 693-701 (2001).
- [13] 佐藤, 岡田, 中本: "カスタマイズ可能な携帯電話向けユーザインタフェースの実装と評価", 情報処理学会研究報告, UBI-13, 28, pp. 195-202 (2007).
- [14] 星合, 山本, 柴田: "パーソナライズ情報提案型コンテンツ配送コンポーネント: preference", 電子情報通信学会論文誌, J83-D-I, 2, pp. 306-312 (2000).
- [15] 増井, 宮本, 小澤: "携帯端末に適した情報表示・操作方式の検討", 情報処理学会研究報告, 91-4, pp. 25-30 (2000).
- [16] 荒瀬, 前川, 原, 上向, 西尾: "携帯電話を用いた web 閲覧のためのコンテンツ適応的提示システム", 情報処理学会論文誌, 47, 12, pp. 3149-3164 (2006).
- [17] 田中, 一杉: "Mixjuice 言語によるデザインパターンの改善", 情報処理学会論文誌, 44, SIG04, pp. 25-46 (2003).
- [18] 浅羽, 松永: "再利用可能な web ビジネス情報システムの設計について", 情報処理学会研究報告, 2001, 30, pp. 43-48 (2001).
- [19] 満田, 鯉坂, 沢田: "情報相分割に基づくユーザインタフェース設計", 電子情報通信学会論文誌, J85-D-I, 11, pp. 1047-1056 (2002).
- [20] 平田, 千葉, 仁田: "携帯電話向けブログクライアントの開発", 情報処理学会研究報告, HI-114, 4, pp. 25-30 (2005).
- [21] 中西, 坂倉, 財満, 畑山, 小野, 尾上: "組込み機器上で web サービスを扱うプラットフォームの構築", 情報処理学会研究報告, 2007, 34, pp. 83-88 (2007).
- [22] 中西, 榎東, 千葉, 片山, 小野, 尾上: "組込み機器向け文書配信システム", 情報処理学会研究報告, 2008, 34, pp. 75-80 (2008).
- [23] 大澤, 曾根: "Web 検索情報の 3 次元可視化", インタラクシオン 2008 論文集, 2008, 4 (2008).
- [24] The World Wide Web Consortium (W3C): "Cascading style sheets", <http://www.w3.org/Style/CSS/>.
- [25] European Computer Manufacturers Association (ECMA): "ECMAScript Language Specification", <http://www.ecma-international.org/publications/standards/Ecma-262.htm>.