

Android 端末を用いた直感的インターフェースを持つ楽器アプリケーションの開発

Development of musical instrument application with intuitive interface

宮島明彦

Miyajima Akihiko

法政大学情報科学部デジタルメディア学科

E-mail: 08k1135@stu.hosei.ac.jp

Abstract

This research aims at development of a musical instrument like a synthesizer without a keyboard that utilizes features of Android efficiently. The designed application outputs a sound suitable for an image of a color. If a bright color of a screen is touched, a high sound will come out. An operation screen is designed paying attention synesthesia. Synesthesia is a phenomenon in which a color can be seen, when a sound is heard. A comparative experiments of operate time until a subject understands operation were conducted. A comparative experiment of two screen designs showed usability of the interface. A screen of one was pure white and another colored a screen. An operate time of colored screen was an average of 10.7 seconds quicker than a white screen. The experiment showed that colored screen tends to memorize tone change.

1 まえがき

音声通話通信機能だけでなく、ネットワーク機能やその他多様な機能を搭載したスマートフォンの普及が進んでいる。スマートフォンの大きな特徴は直感的に誰でも簡単に扱えるインターフェースを持つことである。そのため、様々な年齢層や習熟度の低い人々でも扱いやすいPCのような役割を果たす端末になると注目されている。これらの端末で、センサーやカメラ、タッチディスプレイ、ネットワークなどの機能を用いたユニークなアプリケーションが開発可能である。Android 端末を用いてそれらスマートフォンの機能を使い、直感的な操作が可能な楽器アプリケーションを開発することで、ユーザーに楽器の楽しさを伝え、音楽を楽しむきっかけを作ることが出来る。本研究では誰にでも簡単に操作が出来る使いやすいインターフェースデザインを作成することを目的とする。

2 楽器アプリケーション

楽器は発音原理によって分類され、その演奏法は様々である。演奏によって音の3大要素である音程・音色・音量を制御する。そのため楽器アプリケーションは演奏方法を定義し、音程・音色・音量を演奏によって制御する機能が必要である。

アコースティックギターの場合は、手で弦を弾くという演奏動作によってボディを振動させ発音を行う。張られている6本の弦のどの太さの弦を弾くか、どこフレットを押さえて弾くかによって音程が制御される。また弦を弾く強さや速さなどで音色・音量の制御が行われる。

演奏法に注目し開発されたものでは電子楽器や電子端末での楽器アプリケーションが挙げられる。演奏法を実際の楽器に近づけ、発音に電子音を用いるものが多い。鍵盤楽器や打楽器などがある。先行研究 [1] ではタッチパネル式の小型ディスブ

レイを備えたPDA(Personal Digital Assistant)を用いたタッチ操作でのモバイル楽器が開発されている。しかしこれらは実際の楽器を完全に再現することが難しいなどの課題点がある。

そのため音程の制御や音色の制御に特徴をもつ新しい演奏法の携帯電子楽器が開発されている。タッチパッドで直感的操作の可能な、KAOSILATOR などである。この電子楽器の特長はいつでも、どこでも、誰でも楽器を演奏出来ることである。このメリットはスマートフォンを用いることでさらに向上出来ると考える。タッチ操作のシンセサイザーはスマートフォンを用いることで、特定のデバイスを持つ必要がなくなる。本研究では画面のデザインによって楽器アプリケーションの使いやすさを向上させることを目的とする。

3 楽器アプリケーションの構築

3.1 アプリケーションの概要

Android 端末の特徴の調査を行い、音程制御と音色制御の操作性を考慮し設計を行った。演奏方法は画面のタッチ操作を主とした。色の明るい箇所を指で触れると高い音が鳴り、暗い箇所は低い音が鳴るというように、視覚で捉えた色のイメージに合った音声が出力される画面インターフェースを持つシンセサイザーのような楽器アプリケーションである。

色の特徴と音の特徴を関係付けることで、操作性の向上を目的とした。色と音のパラメータの関係性を用いたインターフェースの先行研究 [2] は行われているが演奏楽器のアプリケーションに用いられていない。色は色相、彩度、明度などで分類されている。音の性質には音圧、音高、音色などがある。これらの関係性を調査し用いることで、ユーザーが直感的に操作方法を理解しやすいインターフェースを設計を目指した。

3.1.1 色と音のマッピング効果の利用

心理学の分野に「共感覚」知覚と呼ばれる現象がある。これは音を聴くと色が見える、または味から物の形が感じられるなどの、一つの物理的刺激が本来独立な異なる感覚を誘発することを言う [3]。最近では脳内での共感覚現象が医学的に確認されており、共感覚は万人に起こっているが限られた人にだけ表出しているという説もある。その中でも音に色が見える共感覚は「色聴」と呼ばれる。先行研究 [4] では色聴保持者の色と音の対応付けの規則性を明示的に表現し、色聴を持たない一般人においても音と色のマッピングが潜在的に保有される可能性を示している。この感覚に注目し聴覚と視覚の情報を関連付け、直感的に操作しやすい音程制御と音色制御を考えた。

先行研究 [4] において、音楽における調・音色・音高のパラメータを変化させた際に色聴保持者が感じる色の対応付け(マッピング)の規則性を、色相・明度・彩度のパラメータを用いて明示的に表現した。絶対音感を持つ色聴者においては、音の調性と色相には個人間では高い相関を持つことが明らかにされ、色聴者間にもばらつきを含む程度の共通性が見られた。また高調波成分を増やし音色を変化させた時、高調波成分が増えると彩度が高く、明度が低い色を選ぶことを示した。さらに音

高が高くなると色の明度が高い色を選ぶ傾向を示した。一般人においては音と色のマッピングを明示的に表すことが出来なかった。しかし色調保持者のマッピングを補助的に与えることで、音色・音高については同様の感覚を持つ可能性があることを明らかにした。これらの結果から色調保持者と一般人に同じ傾向が受容された音色・音高と色のマッピングに注目し直感的操作の可能な画面のデザインを行った。

3.1.2 アプリケーションの特徴

音程制御と音色制御に先行研究 [4] で明らかにされた次の傾向を用いる。音の倍音成分が増えるにつれて彩度が高く、明度低い色をイメージする。また音の高さが高くなるにつれて、色の明度が高い色をイメージする。しかしインターフェースとしての今回の用途としては配色がくすんだ色に偏り色の差が分かりにくいいため、音色変化に関しては倍音成分増加し歪音になるにつれて彩度が低く、明度低い色とした。

色相によって明度の最大値は異なる。色相と音の関係については先行研究では明確な関連性が見られなかったため、色相の違いは明度の違いであるとして配色を行う。明度の高い色相の色と明度の低い色相の色を画面に配色したとき、明るい色相の箇所を触れると高い音を、暗い箇所は低い音を出力する。また彩度の高く明度の高い色の箇所は倍音成分の少く歪の弱い音を、彩度の低く明度の低い色の箇所は倍音成分が多い歪の強い音を出力する。ユーザーが触れた画面の箇所を表示し、画面の変化でユーザーの操作の補助になる機能を実装した。

図 1 に最終的なアプリケーションの機能を示す。

3.2 楽器アプリケーション機能の調査と検証

楽器アプリケーションを構築するために機能を設計し検証を行った。

- 発音部分
 - Android に適した音声出力の調査
 - 音の連続再生の実装
- 音程の制御
 - スライドタッチ時の周波数の動的変化処理の実装
 - 一画面で行う周波数変化の検討
 - トーンベンド機能の実装
- 音色の制御
 - スライドタッチ時の音色の動的変化処理の実装

今回音量制御については Android 端末本体の出力音声のボリュームボタンを使用し特に実験は行わない。OS のバージョン

は Android1.5 を使用し、検証はエミュレータと実機端末 HT-03A で行った。OS バージョンを変えて行った実験についてはその都度記載する。

3.2.1 Android に適した音声出力の調査

Android 端末の音声出力の実装方法にはいくつかの方法がある [5][6]。SoundPool(API レベル 1)、AudioTrack(API レベル 3) の二つを実装し用途に合わせた音声の出力の調査を行った。

SoundPool は保存されている音声ファイルを再生できるクラスである。実験として 1 秒間の正弦波の音声ファイルの出力を SoundPool クラスで実装した。その結果、音の遅延も少なく指定した音声出力されることを確認した。しかしエミュレータでの動作時は、音声の連続再生時に無音区間が発生した。実機でも音の遅延などから即時的に連続的に変化する音声の出力にはあまり適していないという結論を得た。

AudioTrack は Android 端末内で音声を生成し出力するクラスである。実験として 1 秒間の正弦波の音声ファイルの出力を AudioTrack クラスで実装し連続再生を行った。その結果、エミュレータと実機ともに音の遅延も少なく指定した音声連続再生されることを確認した。このことから AudioTrack クラスを音声出力のメインとして即時的に連続的に変化する音声の実装に用いる。

3.3 音の連続再生の実装

音の連続再生機能を実装し検証を行った。ギターでは弦を弾き振動させるといった楽器の発音部分の実装である。OS のバージョンは Android1.5、Android4.0 を使用し、検証はエミュレータと実機端末 HT-03A、GALAXY NEXUS SC-04D で行った。

本研究の楽器アプリケーションの発音方法として、ユーザーが指で画面に触れ続けると電子音が端末から鳴り続け、指を離すと音が止まる発音方法を提案する。AudioTrack を使用し音声の生成と連続再生の実装を行った。純音である正弦波の音声波形の生成を実装した。1 秒間の音声データを端末内で生成し、音声の再生を繰り返す処理を実装した。

音声の再生時の遅延や音切れのない連続再生を実現した。Android4.0 のエミュレータで行った場合の方が Android1.5 より音切れが少なく OS 間で差があることが分かった。またエミュレータでは音切れや遅延を起こしていたが実機端末の GALAXY NEXUS SC-04D ではなめらかな連続再生を確認した。

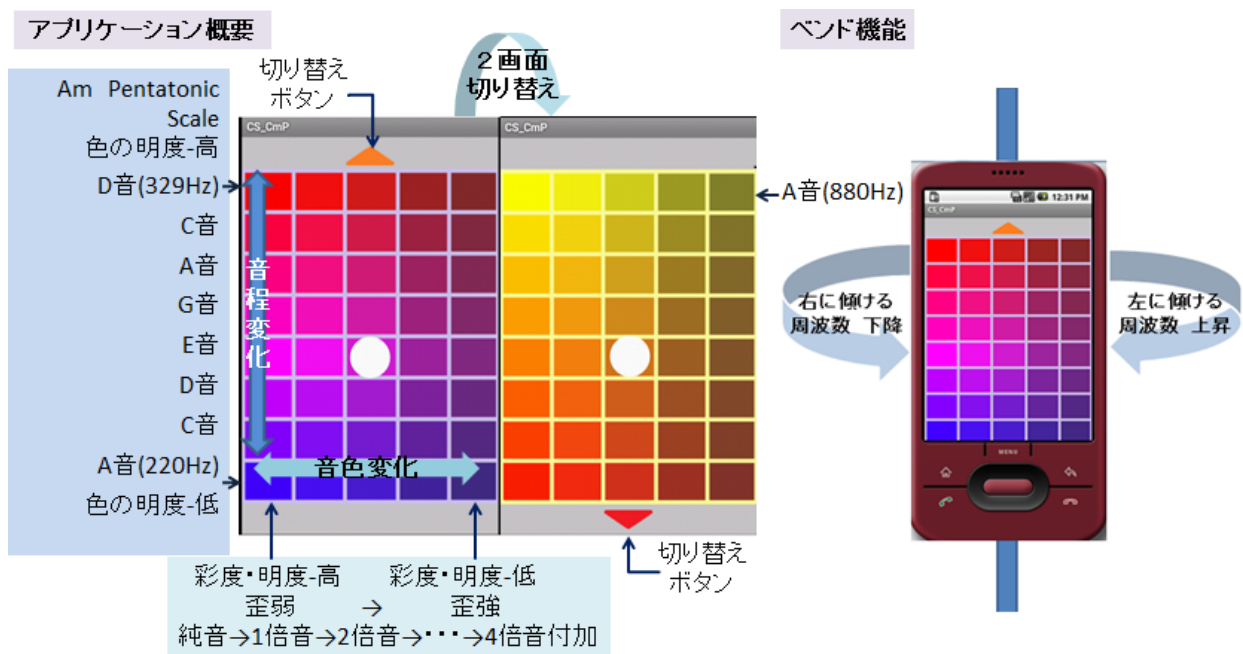


図 1. アプリケーションの機能

3.4 スライドタッチ時の周波数の動的変化処理の検証

ユーザーが触れた画面の Y 座標を取得し、出力される音声の周波数を動的に変化させる機能を実装し動作検証を行った。ギターでは弦を押さえる位置によって音程を変えるといった、楽器の音程制御の部分の実装と検証である。Y 軸の値を 3 分割し画面下の値を C 音 (260Hz)、画面真ん中の値を E 音 (330Hz)、画面上の値を G 音 (390Hz) の 3 音に割り当てた。音と音の間は矩形的な変化とした。

出力音声周波数の違和感のない動的変化を確認した。出力音は 3 音とも同時に音が鳴り続ける処理を行い、取得した座標を判別し割り当てられた座標の音以外は周波数 0Hz (無音) の音声を出力するといった処理を行うことで遅延のない周波数切り替えを実装した。

3.5 一画面で行う周波数変化の検討

携帯端末は小型の端末であるため、使用できる画面に制限がある。一画面で操作できる周波数変化の数が、どの程度なら演奏しやすいかを検証し適当な数を調査した。この実験はギターでは弦を何本にするか、フレット数をどのくらいにするかといった楽器の使いやすさを考慮した音程制御の部分にあたる。実験に使用した端末 HT03-A の画面サイズは約 3.2 インチで 320 × 480 ドットである。画面には何も表示しない。一画面で変化させる音は 3 音から 15 音まで 3 音ずつ増やし演奏のしやすさの検証を行った。

検証の結果から一画面に約 6 音から 9 音の間であれば、メロディーを奏でるときに狙った音が出しやすく、音程のパリエーションも妥当であることを確認した。この結果から、一画面で操作する音の周波数変化の数を一般的なスケールの音数である 8 音とした。

また画面の上部と下部にボタンを配置し、ユーザーがそのボタンを押すことで画面の配色が変わり、オクターブ違いの音階の演奏を可能にした。この機能によって演奏の幅を広げることが出来た。

3.6 トーンバンド機能の実装

演奏の表現の幅を広げるため、無段階で音程を変化させるトーンバンド機能を実装した。この実験はギターでは弦を上下に引っ張り無段階で音の周波数を変えるといた楽器の音程制御の部分にあたる。OS のバージョンは Android4.0 を使用した。検証は実機端末 GALAXY NEXUS SC-04D で行った。

トーンバンド機能には傾きセンサーを用いる。ギターの場合トーンバンド時に手首をひねるような動作を行う。ギターのように通常演奏中に簡単にトーンバンドを行うには、本体を傾ける回転動作が直感的にわかりやすく適していると考えた。

傾きセンサーでは方位角、傾斜角、回転角の 3 つの角度の値が取得できる。この中の回転角の値を取得し周波数に反映させる。本体を床と平行に近い状態にして持った時に手首を回転させやすい方向であったため回転角を使用する。左に傾けると出力音の周波数が無段階で半音上の音程まで高くなり、右に傾けると低くなるよう設計した。

3.7 スライドタッチ時の音色の動的変化処理

画面タッチ時の画面の X 座標を取得し、出力される音声の倍音成分を動的に変化させ、出力音の歪み方を制御する機能を実装を行い動作検証を行った。エレキギターでは弦の弾く位置を変える、弾く速さや強さを変える、エフェクターを使用するといった、楽器の音色制御の部分の実装と検証である。画面を 5 分割し左から右にスライドタッチすることによって徐々に音の歪みが強くなる機能を実装する。倍音成分は最大 4 倍音まで付加させる。音程変化は周波数の動的変化処理の検証と同様に実装した。

スライドタッチすることによって意図通りに徐々に音色が変化していくことを確認した。倍音成分の変化や歪の強さによって、若干音量が違って聞こえたため、出力音のラウドネスを調べ音色によって出力音の音量を調整した。

4 ユーザビリティ評価実験

4.1 実験目的

定量的評価実験を行い、操作の処理時間からアプリケーションの使いやすさを調査する。本実験はインターフェースの性能向上も兼ねている。ユーザーインターフェース開発プロセスでは、問題点の発見と改善を重視するため、定性的手法 [7] を多く用いる。代表的な定性的手法としては「ユーザーテスト」が挙げられる。

ユーザーテストとは、被験者がタスク (課題) を実行する過程を観察し、被験者の行動、発話からユーザーインターフェース上の問題点を発見する評価手法である。ユーザーテストでは、5 人の被験者でユーザビリティ問題の 85% を発見出来ることが明らかになっている [8]。ユーザビリティを向上させるには、何十人も一度にテストするよりも、5 人程度の小規模なユーザーテストを繰り返した方が効果がある。これによってインターフェースの問題発見も行う。先行研究 [9] を参考に実験法の計画を立てた。

4.2 実験法

法政大学内の教室を使用し、被験者は 10 人 (20 代男性 9 名、20 代女性 1 名) に依頼した。実験の流れは以下の通りである。

1. 事前インタビュー
2. タスクの実行観察
3. 事後インタビュー

実験は一人ずつ行い、その様子は音声レコーダーとビデオカメラで記録を行った。

4.2.1 事前インタビュー

被験者に楽器の訓練経験と経験年数、スマートフォンの使用経験と経験年数のアンケートを行った。

4.2.2 タスクの実行観察

被験者に課題を提示し、その実行過程を観察しながら適宜インタビューを行った。まず被験者をカラーチームと白黒チームに分ける。カラーチームには画面に音声にあった配色の有るものを使用してもらい、白黒チームには画面に格子線のみ表示される配色無しのものを使用してもらった。図 2 はそれぞれの操作画面である。

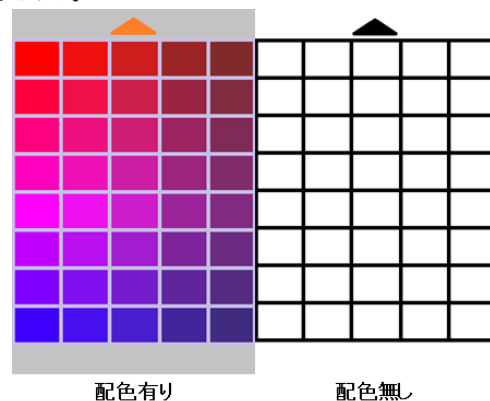


図 2. 各操作画面

簡単な音声を提示し、提示した音声と同じ演奏が出来るまでの時間を計測し比較した。タスクには単音を 4 つ、音色固定で音程変化するフレーズを 3 つ、音程固定で音色変化するフレーズを 2 つ、音色・音程変化するフレーズを 1 つ用意した。各タスクの平均処理時間を図 3 に示す。

処理時間を平均するとカラーチームの方がほとんどのタスクにおいて処理時間が短く、処理時間の平均はカラーチームの方が 10.7 秒短い。個人差はあったがカラーチームの方が音の配列の法則に気づく時間が早く、狙った音が出せるようになる時間は短かった。

タスク 3 とタスク 10 のみ白黒チームの方が処理時間が短くなっている。タスク 3 は単音の純音の最高音を鳴らす操作であ

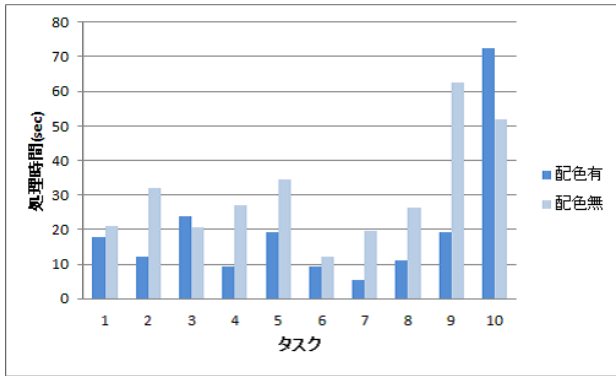


図 3. タスクの平均処理時間

る。このタスクでは画面の切り替え操作が必要とされ、その操作に気づく時間に被験者間で差が出たことが原因であったと考えられる。タスク 10 は音色・音程変化をするフレーズの演奏操作である。この操作は画面を指で斜めにスライドさせる動作を行う。被験者のコメントから格子線が斜め操作を連想しにくくした可能性があることが確認された。

この実験より配色が音の探索や記憶の手助けとなっていることを確認した。

4.2.3 事後インタビュー

タスク実行観察後にカラーチーム白黒チームともに画面に配色の有るものと無いもの両方を使用してもらい、主観的評価のために以下のアンケートを実施した。

1. 操作法は理解しやすいか？
2. 音の高さの変化を覚えやすいか？
3. 音色の変化を覚えやすいか？
4. 使っていて楽しいか？

被験者には上記の質問にどれくらい同意できるかを Likert 尺度の 5 段階評価 [10] で回答してもらった。賛成の度合いを指摘させ、その総合点やその平均で態度を判定するものである。簡易で行って段階的選択によって被験者の微妙な感情の変化を測定するのに簡単でやりやすい方法であるためこの方法を使用する。配色有りと配色無し各画面設計の主観評価結果を図 4 で示す。

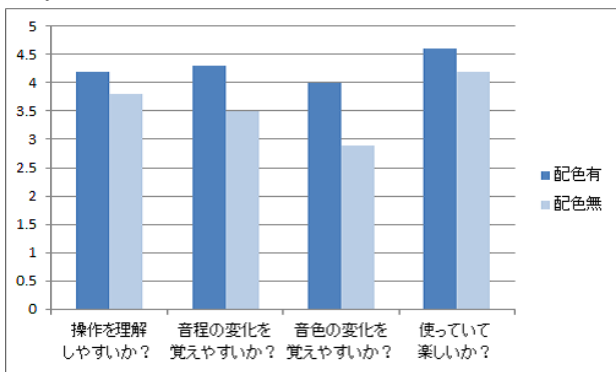


図 4. 操作の主観評価

どの項目も配色有りの画面インターフェースの方が高い評価を得た。特に配色有りの画面の方は「2. 音の高さの覚えやすさ」と「3. 音色の覚えやすさ」に関して配色無し画面と比較し高い評価を得ている。Wilcoxon 符号付順位和検定で有意水準 5% の評価をした結果、それぞれ $p=0.027$ 、 $p=0.018$ となり有意差が認められる。

被験者のコメントでは音色の変化は配色があった方が覚えやすい、濁った色の箇所が音のイメージに合っていたなど特に音色と色の変化に関するコメントが多かった。このことから彩度と明度が低い色のイメージが歪音の音のイメージと非常に合っていたことが音色の変化を分かり易くしたと言える。

5 あとがき

本研究では音高、音色など音の性質と、色の彩度、明度など特徴の規則性を調査し、その結果に基づき視覚で捉えた色のイメージに合った音声出力される画面インターフェースを持つシンセサイザーのような楽器アプリケーションを設計した。色と音の関連付けで音程制御と音色制御の設計を行った。音の倍音成分が増え、音が歪むにつれて色の彩度、明度が低くなる。音の高さが高くなるにつれて、色の明度が上がる。これにより使いやすさを向上させるとともに、視覚でも楽しめる楽器アプリケーションを実現した。

被験者に与えたタスクの処理時間を比較し、主観的な使いやすさも比較した。タスク 9 は音色変化を行う 2 回目のタスクである。音程は固定で歪音から純音に変化する演奏フレーズを提示する。このタスクにおいて配色有りの画面のほうが平均 43.2 秒短かったことから音と色のパラメータを対応付けることによって音の変化を感じやすく記憶に残しやすい事を示した。

また Android 端末のセンサーシステムを用い取得できるユーザーの動作の情報を調査し、傾きセンサーによるトーンバンド機能を付加することで演奏の幅を広げた。しかしトーンバンドの演奏時に端末を回転させる動作に加え、ほとんどの被験者が試しに端末を振っていた。このことから振るという動作は、スマートフォンの操作において簡単で思いつきやすい動作であることを確認した。端末を振る動作に何らかの機能を付けることで使いやすさを向上できる可能性がある。

出力音声を矩形的に変化させていたため格子線を用いて色を配色していた。しかし評価実験より格子線が斜め操作を連想しにくくした可能性があることから他の配色法も考えることでより操作しやすい画面になると考えられる。今回は和音、高調波成分以外の音色を変化させるパラメータ、複雑なスケールと色の関係については調査を行っていない。それらの規則を用いることで歪音以外の様々な音色に合った配色が出来るだろう。

参考文献

- [1] 寺田努, 塚本晶彦, 西尾章治郎: “2 つの PDA を用いた携帯型エレキベースの設計と実装”, 情報処理学会論文誌, Vol.44, No.2, pp. 266-275, 2003 年 2 月.
- [2] 藤澤隆史, 長田典子, 谷光彬, 片寄春弘: “和音の定量的評価モデルに基づいた楽曲ムードの色彩表現インターフェース”, 情報処理学会論文誌, Vol.50, No.3, pp. 1133-1138, 2009 年 3 月.
- [3] リチャード E. シトウィック, 共感覚者の驚くべき日常 - 形を味わう人, 色を聴く人, 草思社, 2002 年
- [4] 岩井大輔, 他共著: “音と色のノンパルマッピング: 色聴保持者のマッピング抽出とその応用”, 電子情報処理通信学会論文誌, Vol.J86-A, pp. 1219-1230, 2002 年 11 月.
- [5] 布留川英一, Android プログラミング 1.5 バイブル, ソシム, 2009/06
- [6] 「Android Developers」, <http://developer.android.com/>, 2011/08/07 アクセス
- [7] 「ユーザビリティの評価方法」, <http://www.usability.gr.jp/whatis/methods/>, 2011/10/26 アクセス
- [8] Nielsen J, Landauer TK, A mathematical model of the finding of usability problems. Proceedings of ACM INTERCHI'93 Conference (Amsterdam, The Netherlands, April 24-29, 1993), p.206-213
- [9] 加藤晴久, 米山暁夫, “携帯電話のカメラワーク推定を利用した操作インターフェースの評価-3D マップを用いた操作性の評価-”, ITE Technical Report Vol32.No.26.PP.13-16, 2008 年 6 月
- [10] 原岡一馬, 心理学研究の基礎, ナカニシヤ出版, 2002 年