

色相判定による通信システムの提案

先端工学基礎課程 学籍番号：1324020 寺田研究室 岡川翔子

1 背景

スマートフォンやタブレットなど携帯端末が普及している中、様々な通信方法が確立されている。通信方法の代表的なものとして Bluetooth, WIFI, QRコードによる情報通信などが挙げられる。Bluetoothは登録された媒体間でのみ通信を行うことができる。WIFIは通信可能区域を制限して通信を行うことができる。QRコードは一枚の画像コードを解析することで通信を行っている。しかし、QRコードには写真からのコードの読み取りも可能なため、画像がインターネット上で簡単に拡散できてしまう昨今では配布場所を制限することができないという欠点が存在する。

2 目的

色を高速で切り替える映像を端末のカメラから取得し、色相情報に割り振られているビットを解析することで情報通信を行うことを提案し、実装、評価することを目的とする。

3 関連研究

3.1 Vinteraction [1]

加速度センサと振動モータを利用した情報送信インタラクションの研究である。2台の端末を重ねて置き、片方が発する振動をもう一方の端末で取得してビットに直し複合することで通信を可能にする。

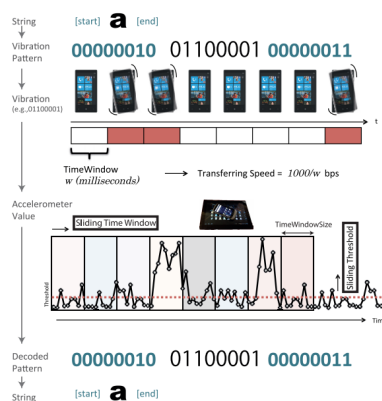


図1 振動モータと加速度センサによる情報通信手法 [1] 図1

4 提案システム

4.1 概要

本研究では端末に搭載されているカメラ機能を利用し、取得した映像から色相を解析することで行う通信システムを提案する。具体的には、送信したいものを色相を利用してエンコードし順に描写する動画を作成する。この動画を流しているところを端末のカメラ機能を利用して取得し、逐一検知した色相をデコードすることで通信を行う。色相情報を利用するメリットとして、表示する環境による影響が受けにくいと考えたことと、解析の精度によっては使用可能な色を増やすことが可能であり、色を増やすことによって乗せられる情報量を大きくできるだろうと考えたためである。

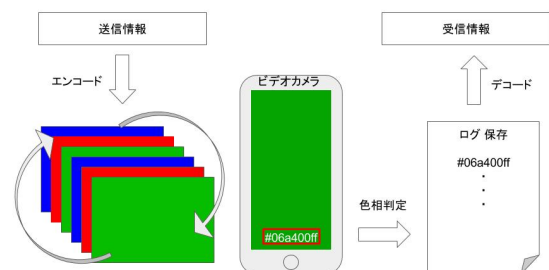


図2 色相判定による通信システムの概要

5 実装

5.1 実装環境

- 端末機種：Xperia Z3(SO-01G)
- androidバージョン：Android 6.0.1

5.2 ビデオカメラ

送信元の情報から作成された色情報の動画を取得するためのビデオカメラのアプリケーションを実装した。ビデオカメラの起動中に取得された色情報は画面に表示されるようになっている。実装には、AndroidのAPI Level 21から実装されたCamera2APIを利用した。

5.3 色相判定

カメラを起動している間、カメラが捉えている場所の中央付近のピクセルから色相情報を取得し、ログとしてテキストファイルに保存している。色相情報は最初に YUV という色空間として取得し、YUV の値を利用して RGB カラーモデルを計算して求めている。この時 RGB は 16 進法で #FF0000 というように表記されているが、ログとして保存する前に RGB=(255,0,0) という表記に変換してから保存している。

YUV とは輝度信号 Y と色差信号 U-V から表現される色空間のことである。また、RGB とは赤、緑、青の三原色から幅広い色の種類を再現することが可能なカラーモデルのことである。

5.4 未実装

示された色情報に情報を載せて実際に送受信を行うことができなかった。計測できた色情報の 255 を 1, 0 を 0 とビットを指定し、送信情報を 2 進数に変換することでデータの送受信が行えることの確認を行おうと考えていた。以上のことが確認できた後に、判別する色の種類を増やすことでビット数を増やしていきどれほど大きな情報がやり取りできるようになるのか計測する予定であった。

6 結果

カメラアプリは正常に動作させることができ、取得した色相のデータは実際の色相と 10~20 の誤差はあれど正確に計測できた。また、色相の判別は無事三色に判別することが確認できた。

7 課題

7.1 未実装部分の実装

5.4 で説明したシステムの未実装部分である通信システムの実装が最優先であると言える。

7.2 取得色相の細分化

現在、取得された色情報は赤、青、緑の三原色に分類されているが、より複雑な通信を可能にするため分類可能な色彩を増やす必要があると思われる。赤、青、緑のうち二つを重ねることで生まれる紫、黄、水色、さらに全て重ねなかった時の黒と、逆に全て重ねたときの白の計 8 色を増やしたいと思う。

7.3 色相判別の正確化

5.2 の細分化が行われるということは一色が抱えられる色の範囲が狭くなるため、現在の実装のままに対応できるかという問題が生じると考えられる。

7.4 対応機種拡大

現在カメラの実装に使用している Camera2 API は Android5.0 以上の機種しか対応することができない。Android4.4 以下の Android バージョンが搭載されている端末には、同様に公開されている Camera API を使用する必要がある。もしくは、C#を用いることで Android だけでなく iPhone にも対応可能にすることができると考えられる。

参考文献

- [1] 米澤拓郎, “Vinteraction:スマート端末のための振動を利用した情報通信インタラクション”, 情報処理学会論文誌, vol.54, pp.1498 - 1506 (2013).