

ColorSignal 暫定仕様と活動

目次

ColorSignal 暫定仕様と活動	1
ColorSignal 送信側仕様（暫定）	2
送信側実行環境	2
推奨付属デバイス	2
伝達時間（開始シグナルから終了シグナルまで）	2
有効距離	2
環境光	3
ColorSignal 受信側仕様（暫定）	3
受信側実行環境	3
読み取り精度	3
読み取り速度	4
同時読み取り可能なシグナル数	4
動画種別	4
データ出力	4
利用可能カメラ	5
活動内容 2018.2～2019.7	6
目標	6
主な活動内容	6
今後の活動フロー	7

ColorSignal 送信側仕様（暫定）

送信側実行環境

- Arduino 互換マイコン

推奨付属デバイス

- カラーセンサー
 - 用途：環境光のセンサー値に応じた色味と明るさの自動調整
 - 今回のサンプル実装：S11059-02DT（浜松ホトニクス）→500円（税込）

伝達時間（開始シグナルから終了シグナルまで）

伝達時間(実測値) ※色と色の間の黒インターバル 10msec

		1 色当たりの発光時間（ミリ秒）			
16 進文字列	色配列長※	70ms	100ms	150ms	200ms
89AB	20 文字	1,628ms	2,228ms	3,228ms	4,228ms
781B	21 文字	1,709ms	2,339ms	3,390ms	4,439ms
0GA5F9	24 文字	1,953ms	2,674ms	-	-
783412	27 文字	2,196ms	3,008ms	-	-

※16 進文字列の組合せによって変換された色の配列の長さは異なります

※Arduino nano 互換マイコンで測定

ポイント💡

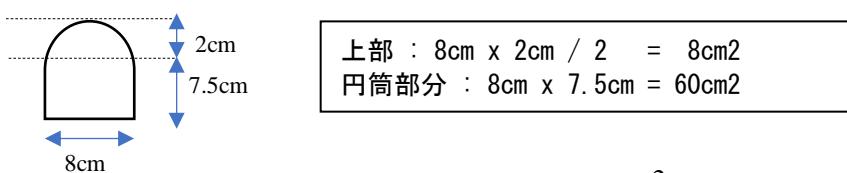
背景の明るさや受信側のマシンスペックにもよりますが、受信側仕様「読み取り速度」の表にあるように上記の時間+10 ミリ秒ほどでデコード完了となります。

有効距離

【限界値ではありません】 12m まで確認済み。※撮影場所（弊社ビルの廊下）の物理的限界

※デバイスの光っている面の面積で変わります。

(送信デバイスの光っている面の面積→正面を投影図で約 68cm²)



環境光

【限界値ではありません】※撮影場所（弊社ビルの廊下）での色彩照度計での測定値

- 照度 : 323～375lx
- 色度座標 x : 0.324～0.328
- 色度座標 y : 0.339～0.347

ColorSignal 受信側仕様（暫定）

受信側実行環境

- C++ライブラリが動作可能な OS (Windows/MacOS/Linux 等)
- 必須ライブラリ : OpenCV(>3.4.3)

読み取り精度

※動画ファイルのみ

	OK の個数	NG の個数	精度(%)	精度(%) 3 繰返しを 1 セットとした場合
全体	503	1	99.8	100 (=168/168)
6m / 70ms	36	0	100	100(=12/12)
6m / 100ms	81	0	100	100(=27/27)
6m / 150ms	27	0	100	100(=9/9)
6m / 200ms	36	0	100	100(=12/12)
9m / 70ms	36	0	100	100(=12/12)
9m / 100ms	62	1	98.4	100(=21/21)
9m / 150ms	27	0	100	100(=9/9)
9m / 200ms	36	0	100	100(=12/12)
12m / 70ms	36	0	100	100(=12/12)
12m / 100ms	63	0	100	100(=21/21)
12m / 150ms	27	0	100	100(=9/9)
12m / 200ms	36	0	100	100(=12/12)

読み取り速度

テスト用動画を 40 回繰り返し実行して平均した結果。

- 同時に 100 個のシグナル領域が出現して同時に終わる
- フレームサイズ 720p (1280 x 720)
- 背景の明るさによる処理負荷：**黒 < 実際の映像 < 白**

読み取りマシン

- MacBook Pro (13-inch, 2017, Two Thunderbolt 3 ports)**
- プロセッサ: 2.3 GHz Intel Core i5 (2 コア 4 スレッド)
- メモリ: 8 GB 2133 MHz LPDDR3

計測環境	背景: 黒		背景: 撮影実験動画		背景: 白	
送信文字列	各フレーム 平均処理時 間:ms	デコード時間 (from 終了シ グナル):ms	フレーム平均処 理時間:ms	デコード ms	フレーム平均 処理時間:ms	デコード ms
78IB(色数 20)	3.59	2.60	7.06	2.50	11.27	2.70
00GA5F9(色数 24)	3.65	2.90	7.01	2.65	11.23	2.75

同時読み取り可能なシグナル数

1 ~ 1 0 0 ※100 以上も可能ですが読み取り速度が低減する可能性があります

動画種別

動画ファイルでもストリームで読み取り可

データ出力

任意の形式に出力が可能

- ファイルに出力
- 特定の IP アドレスに対して TCP 出力
- REST API を呼びだす、など

説明) 一つのシグナルに対してフレーム開始、フレーム終了、デコードのタイミングで callback 関数が呼ばれ引数にはシグナルの領域情報（中心点、矩形の大きさ、発光領域のピクセルマップ等）、読み取り文字列が渡されるので、それらの情報を元に callback 関数が任意の出力をすることができる。

利用可能カメラ

- 家庭用ネットワークカメラ
- 業務用ネットワークカメラ
- 監視カメラ
- ミドルレンジ以上の Web カメラ

型番	種類	結果	備考
Logicool C920r	Web カメラ	可	メインの実験用
Logicool C922	Web カメラ	可	C920r の上位機種
elecom UCAM-C750FBBK	Web カメラ	可	カメラの設定でガンマ補正と逆光補正を無効にする必要あり
M.Way HD720P USB ウェブカメラ	Web カメラ	不可	AWB※が正常に動作せず、画面中央部が青っぽく映る 720p CMOS センサー, 1280x720 30fps / 640x480 30fps, 固定フォーカス(8cm~∞), 4 層ガラスレンズ, AWB
Logitubo 925A	Web カメラ	可	
elecom NCC-EWNP100WH	ネットワークカメラ	可	
Panasonic WV-SP105A	監視カメラ	可	

※AWB：オートホワイトバランス機能

活動内容 2018.2～2019.7

目標

中距離向け送信側・受信側共に静止しているシーンでのデータ通信を可能なものを開発し、処理速度や読み取り精度等の限界値を把握することを目標に活動を行いました。

主な活動内容

● 受信側

- 受信プログラムの開発
 - ✧ シグナル領域検知処理
 - ✧ シグナル領域の状態管理
 - ✧ シグナル領域のフレーム毎の色の判定処理の工夫
 - ✧ ノイズ除去処理の工夫
 - ✧ 処理高速化の工夫
 - ✧ ソースコードの Swift 言語から C++ 化 (マルチデバイス対応)
 - ✧ 読み取りパフォーマンス測定
 - ✧ 反射による重複領域の判定補助処理

● 送信側

- 送信デバイスのサンプル実装
 - ✧ カバーの工夫
 - ✧ LED と目的色の発色の工夫
 - ✧ カラーセンサーの装着
- 送信プログラムの開発
 - ✧ 目的色の発色の工夫
 - ✧ 環境光のセンサー値に応じた色味と明るさの自動調整
 - ✧ 省メモリ化

● 受信側・送信側共通

- 撮影動画を用いた読み取り実験
- 複数カメラを用いた読み取り実験

今後の活動フロー

