

# 情報家電ハブとしての 光ナビゲーションシステムの提案

前田 勇希<sup>1</sup> 山本 幸司<sup>2</sup> 香川 景一郎<sup>2</sup> 政木 康生<sup>3</sup> 太田 淳<sup>2</sup> 布下 正宏<sup>2</sup>

<sup>1</sup> 独立行政法人科学技術振興機構 研究成果活用プラザ大阪

<sup>2</sup> 奈良先端科学技術大学院大学

<sup>3</sup> 船井電機株式会社

近年、IT が白物家電など身の回りのさまざまなモノへと浸透しつつある。しかし、機能の高度化に伴って操作が難解になりつつある。そこで、身の回りの IT 機器同士を結合するための情報家電ハブとして、光ナビゲーションシステムを提案する。携帯端末に内蔵するイメージセンサから通常の画像だけでなく機械可読な情報も取得する機能を設け、画面上に機器オブジェクトを表示しドラッグアンドドロップ操作などの実現をもくろむ。

現在、高速で情報を読み取る光ナビチップの開発を進めているが、それと並行して、システム開発の要件を明確にするため既存の Web カメラを流用した光ナビシミュレータの開発を進めている。

## OptNavi System as Home network hub

Yuki Maeda<sup>i</sup> Koji Yamamoto<sup>ii</sup> Keiichiro Kagawa<sup>ii</sup>

Yasuo Masaki<sup>iii</sup> Jun Ohta<sup>ii</sup> Masahiro Nunoshita<sup>ii</sup>

<sup>i</sup> Innovation Plaza Osaka, Japan Science and Technology Agency

<sup>ii</sup> Nara Institute of Science and Technology

<sup>iii</sup> Funai Electric Co.LTD

In late years, some home electric appliances are changing themselves as internet appliance. But their operation becomes quite complicated. We propose “OptNavi” as home network hub for their simple and intuitive operations. We make a dedicated image sensor to get a machine readable ID information as well as normal image. OptNavi makes possible for a user to operate appliances on the terminal display with drag-and-drop action. We are developing an OptNavi chip as well as OptNavi simulator using a legacy web-camera. In this paper, we describe an OptNavi system and an OptNavi Simulator.

### 1. はじめに

高度成長期以来、家電メーカーなどの努力により家庭内にはさまざまな電化製品が増え続けている。それらは生活の質を向上させるために大きな役割を果たしてきた。しかし、リビングにはリモコンの山があふれかえる

こととなり、その操作を覚えることが消費者にとって大きな負担となりつつある。

たとえば、テレビには当初は電源スイッチとチャンネルと画質調整程度しかついていなかった。しかし、現在販売されているテレビのリモコンには機能の高度化の代償とし

て無数のボタンがついてくることとなった。同時に、テレビに接続する DVD レコーダーにもリモコンが付属し、テレビのリモコンに類似したボタン、チャンネル選択やボリューム調整などが付属してくることとなった。

我々は、家庭内の家電を集中管理する枠組みとして「光ナビゲーション」<sup>1</sup>を提案している。

これは、携帯端末を介して家電同士を相互結合し、ドラッグアンドドロップによる操作の実現を目指すものである。具体的には、携帯端末に内蔵するイメージセンサから、通常の画像だけでなく家電側に内蔵した LED からの機械可読な ID 情報を取得する機能を設ける。ID 取得・位置検出後は、RF 無線<sup>2,3</sup>、IrDA<sup>4</sup>などを用いて通信を確立する。これによって、携帯端末の画面上に家電情報を投影し相互のラッグアンドドロップ操作やプロパティ設定を実現することを目指す。

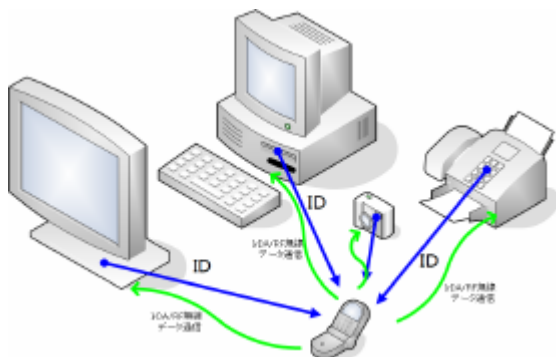


図 1 光ナビイメージ図

現在、図 1 に示すように、この ID 検出機能を内蔵したイメージセンサである光ナビチップの開発を進めている。これと並行して画像処理上の問題点の洗い出しのため、既存の Web カメラを用いて ID 検出を実現する光ナビゲーションシミュレータの開発に取り組んでいる。

## 2. 光ナビチップの概要

開発中の光ナビゲーションチップは、通信モード及び撮像モードを備えたイメージセンサである。撮像モードでは通常のイメージ

センサと同様に振る舞い、QVGA で 60fps のモノクロ画像を出力する。通信モードでは、指定された ID 画素に対して、2.125kHz で画素値の出力を行う。

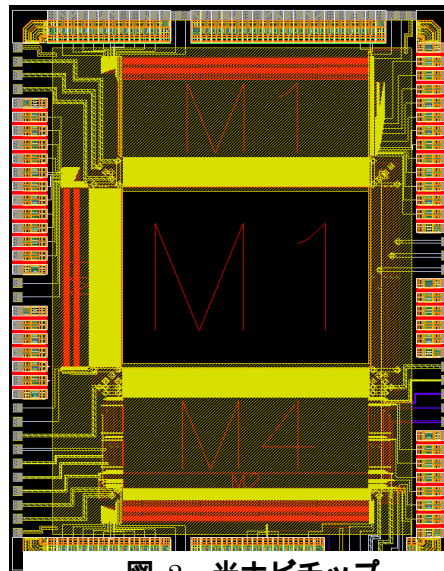


図 2 光ナビチップ

撮像モードと通信モードは、1 フレーム画像を取得する都度、交代する。

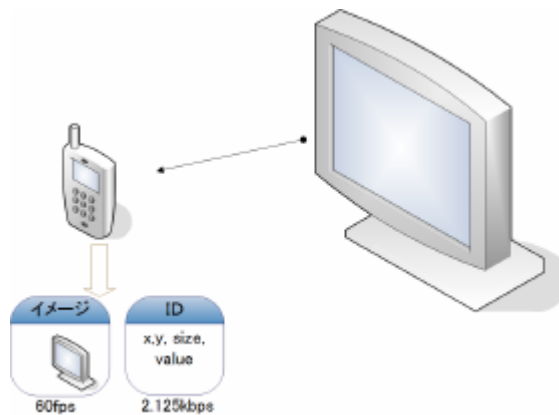


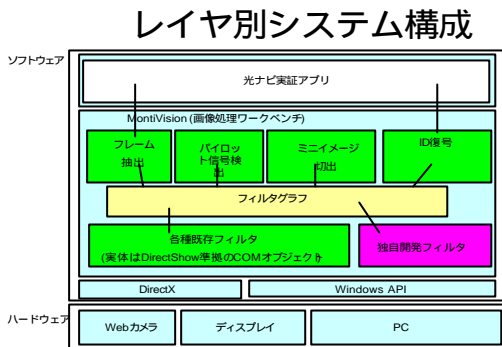
図 3 光ナビデバイス・イメージ図

## 3. 光ナビゲーションシミュレータ

### 4.1 構成

光ナビシミュレータのハードウェアは、Windows PC と Direct X 準拠の Web カメラ、ディスプレイを利用している。Windows のマルチメディア操作 API である Direct X、特に動画の操作を行う Direct Show を用いている。この上で、ドイツの Engineering Firm Cymontkowski 社<sup>5</sup>製の動画処理フレームワーク montiVision を稼働している。この上で

動作する画像処理フィルタを組み合わせたフィルタグラフとして実装している。



#### 4.2 光ナビシミュレータにおけるパイロット信号とID

光ナビシミュレータでは、ID・パイロット信号とも通常の 30fps のイメージセンサで検出可能なものを用いている。そのため、2.125K の光ナビチップと異なり、4bit のエンベロープを用い、最初の 2bit 分でパイロット信号、次の 1bit で ID 値、最後の 1bit を 1 の個数が奇数となるようにパリティビットとする。

パイロット信号としては、'00', '00', '11', '11' と交互になるよう値を設定する、ID 値、パリティビットを含めた 4bit 分の加算平均を取ることにより、'001100110011...' の点滅パターンを得ることが出来る。

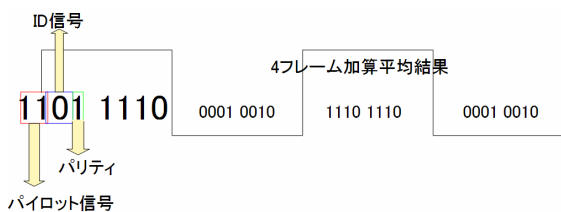


図 4 4 ビットエンベロープ

#### 4.3 光ナビシミュレータ処理内容

光ナビの処理の全体の流れは図5のとおりである。Web カメラや AVI 動画ファイルから読み込み、これをモノクロ化したものを低速系に渡して ID の重心を求める。この重心座標を元に ID デコーダで ID 値の取得を行

う。

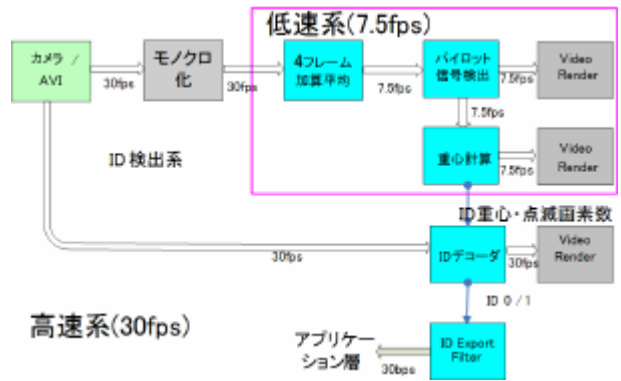
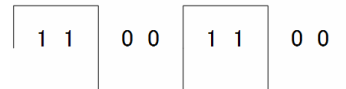


図 5 光ナビフィルタグラフ

#### パイロット信号検出

入力として与えられる画像の各フレームの各画素について、図 6 のような積和演算を行い、1/4 周期ずつずれたテンプレートパターンとの内積を計算し、一定周期で点滅するパイロット信号の検出を行う。

各画素値の時系列サンプル



以下の値との内積をもとめ、その最大値を出力

- 1, 1,-1,-1, 1, 1,-1,-1
- 1, 1, 1,-1,-1, 1, 1,-1
- 1,-1, 1, 1,-1,-1, 1, 1
- 1, -1,-1, 1, 1,-1,-1, 1

図 6 パイロット信号検出法

#### 重心計算

ID の重心計算は二つのプロセスにわかれている。

まず、複数の ID を識別するため領域分割を行う。これは、画面内に複数の LED が点滅しているとき、ID 同士は物理的にはいくらかの間隔を持っていること・一つの ID においては連続した画素が点滅していることを前提条件として、その領域の塗りわけを行う。

続いて、それぞれの領域に対して、重心座標の検出と ID 画素数のカウントを行う。

#### ID 値読み取り

ID 画素数の count を元に、円形に画素が並んでいると仮定すると半径は以下のとおり

である。

$$r = \sqrt{\frac{3 \cdot \text{count}}{4p}}$$

ここで、円形について画素値の計算を行うのはプログラムが煩雑となるため、円に内接する正方形を求め、その幅・高さを以下の式で求める。

$$h = \sqrt{\frac{3 \cdot \text{count}}{2p}}$$

この正方形中の各画素について平均値を求める。



図 7 ID 読み取り領域形状

これを時間軸方向に 8 個求め、8 個の値から時間平均値を求める。各時刻の正方形平均値が時間平均を超えているとき 1、時間平均値以下のとき 0 として ID の値を読み取る。

#### 4. 結論

既存の Web カメラを用いて、光ナビゲーションを実現するシミュレータの構築を行った。30fps と低速で ID から取得可能な情報はわずか 7.5bps ではあるが、今後の光ナビチップを用いたシステムの実装にむけて、パイロット信号検出などの手法の有効性を確認することができた。

#### 5. 今後の課題

##### 5.1 アルゴリズムの改善

ID 領域分割の処理が、前段のパイロット検出の処理結果によって、大きく異なる。単純な形状の領域であれば問題なく処理でき

るが、Web カメラに移動する人影の服など複雑な形状の領域が出現した場合、処理時間が著しく増大するため、フレーム落ちを起こしてしまうことがある。今後、領域分割手法の再考とタイムアウト時の領域分割中断手法について検討したい。

##### 5.2 光ナビチップへの適応

パイロット信号検出では、四重のループを用いており、Pentium4 で 30msec 程度の処理時間が必要であった。60fps で画像を取得できる光ナビチップへの適応を行うためには、処理時間を全体で 16.7msec 以内に抑える必要がある。

##### 5.3 携帯端末への実装

現在のところ、PC 上でメモリを潤沢に用いて処理を行っているが、今後携帯端末への適応に際しては、省メモリなアルゴリズムの開発を進めていく必要がある。

#### 謝辞

本研究は、研究成果活用プラザ大阪・育成研究「携帯情報機器搭載用光ナビゲーションシステムの開発」の助成により行っているものである。

<sup>1</sup> [http://www.osaka.jst-plaza.jp/kadai/15kadai\\_2.html](http://www.osaka.jst-plaza.jp/kadai/15kadai_2.html)

<sup>2</sup> <http://www.wi-fi.org/OpenSection/index.asp>

<sup>3</sup> <http://www.bluetooth.com/>

<sup>4</sup> <http://www.irda.org>

<sup>5</sup> <http://www.montivision.com/>