

OpenCV を使った画像処理コンポーネントの作成例

田窪 朋仁(大阪大学)

概要:

USBカメラで取り込んだ画像をOpenCVで提供されているライブラリで処理し出力するためのコンポーネント群を開発しました。コンポーネント間のデータのやりとりは画像のみに限定し、複数コンポーネントの処理を重ねることで様々な画像処理結果を得ることができます。画像処理における各種パラメータはRTCLinkのConfigurationViewから調整可能にすることで高機能画像処理ライブラリの機能を容易に試用することができます。

特徴:

- ◆ 広く利用されている画像処理ライブラリOpenCVをRTMコンポーネントにすることで作成例を学ぶ。
- ◆ 複数の実用性のある画像処理を簡単に連結できる。
- ◆ 主要な処理パラメータをRTCLinkから調整できるようにすることでライブラリの試用ができる。
- ◆ VMWareイメージにより仮想環境を配布。すぐに試せる。

インターフェース・使用OS環境:

- ・ 入力ポート1, 出力ポート1
- ・ 入出力共にカラー画像データのみ
- ・ VMWareでFedraCore6(Linux)を使用

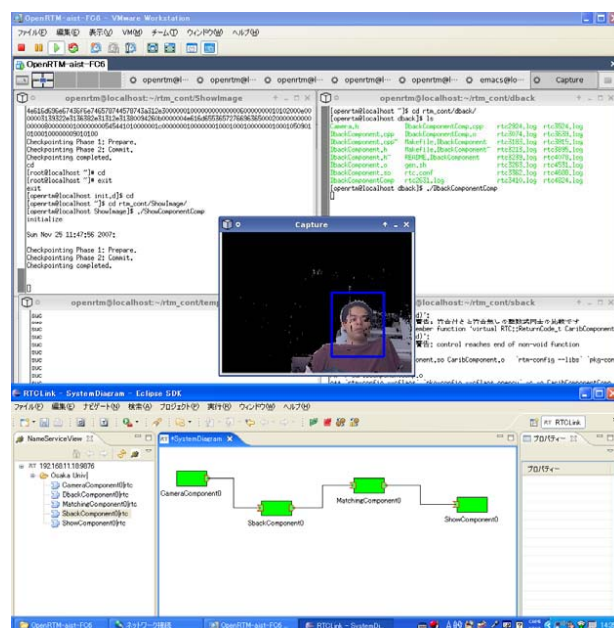


図1 背景差分とテンプレートマッチングモジュールによる人物追従の例

連絡先:

大阪大学大学院基礎工学研究科 田窪 朋仁
 takubo<at>arai-lab.sys.es.osaka-u.ac.jp
 URL: <http://www-arailab.sys.es.osaka-u.ac.jp/~takubo/howto.html>



無料高機能画像処理ライブラリOpenCVの各機能をコンポーネント化することで、プログラムの書き換え無しに様々な処理を試し、結果を表示することができます。

今回作成した処理コンポーネントは下記の13種類です。

- カメラキャリブレーション: カメラの内部パラメータ計算
- 背景差分: モジュールを起動した時を基準とした差分
- フレーム差分: フレーム間差分により動体を表示
- テンプレートマッチング: テンプレートに近い画像を探索
- オプティカルフロー: 区間画像の動きをベクトルで示す
- ハフ変換: 画像内の直線らしき場所を検出
- 閾値処理: カラー画像をある輝度値で2値化する。
- 回転: 画像の回転だけ。あまり使い道はない。
- 膨張と拡大: モフォロジー処理を行える。
- 平滑化: 雑音の除去。このあとに他の処理をするといいいことがあるかもしれません。エッジは甘くなる。
- エッジ画像: カラー画像を入れるとモノクロのエッジ画像になります。
- グレイ画像: カラー画像をグレイ画像に変換します。
- 輪郭抽出: 閾値処理後の画像を入れると輪郭を抽出してくれます。

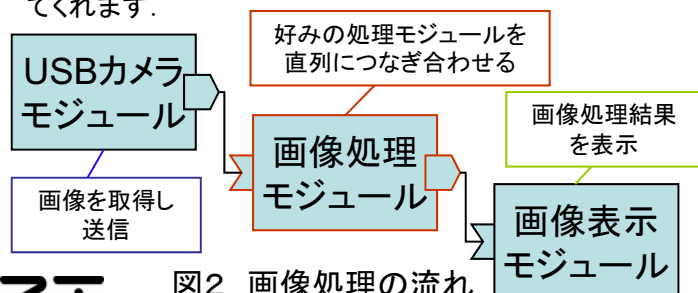


図2 画像処理の流れ

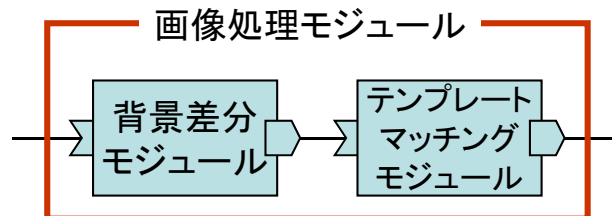


図3 人物追跡機能の構成例

1. 背景差分モジュールの起動時に環境の画像を背景として登録し、人がカメラ前に来たときに背景画像との差分で人のシルエットが抽出されるようにする。
2. 人物シルエットのテンプレートを用意しテンプレートマッチングにより人物の頭部位置を特定する。

モジュールの入力と出力は画像サイズを固定したカラー映像のみとなっているので、1つのモジュールで処理した結果を他のモジュールに数珠繋ぎにして処理を重ねていくことができます。

用意したコンポーネントを使った人物追跡の例を図3に示します。使用するコンポーネントは「背景差分モジュール」、「テンプレートマッチングモジュール」の2つです。まず、カメラから取得された画像を背景差分の処理により人物領域を抽出します。次に、人物のシルエットをテンプレート画像として上半身の位置をテンプレートマッチングにより検出し、画像上に人物の上半身を囲む表示をしています。

詳細に関しては、ホームページをご覧ください。

