情報計測学基礎

2017 年度　春学期　月曜日　2 限

工学研究科　機械システム工学専攻　T11709M　田口 皓一

担当教官：清水 優 教授

課題内容：画像中で動いている物体を判断し，その物体に対して矩形領域を囲う.

0. Ubuntu の環境構築

- 下記コマンドをターミナル上で入力すること.

① sudo apt-get install vim
//vim をインストールする.

② sudo apt-get install chromium-browser
//Google Chromium をインストールする.

③ sudo apt-get install git
//Git をインストールする.

④ sudo apt-get update
//パッケージリストの更新

⑤ sudo apt-get upgrade
//インストールされてるパッケージの更新

⑥ git clone https://github.com/t11709m-chukyo/My1stRepository
//github からフォルダをダウンロードする.

⑦ ren My1stRepository catkin_ws
// My1stRepository というフォルダの名前を catkin_ws に変更する.

⑧ vim .bashrc
//bashrc を開く.

・ 最後の行に　source ~/catkin_ws/devel/setup.bash　と書き込む.

1. catkin_ws 内のプログラムの実行

- 3 つのターミナルを使用し，以下のコマンドを実行する.

・ ターミナル 1

① roscore
//ros を起動する.

・ ターミナル 2

① rosrun usb_cam usb_cam_node
//ros のノードに usb_camera を接続すること.
※usb カメラを usb ポートに接続すること.

・ ターミナル 3

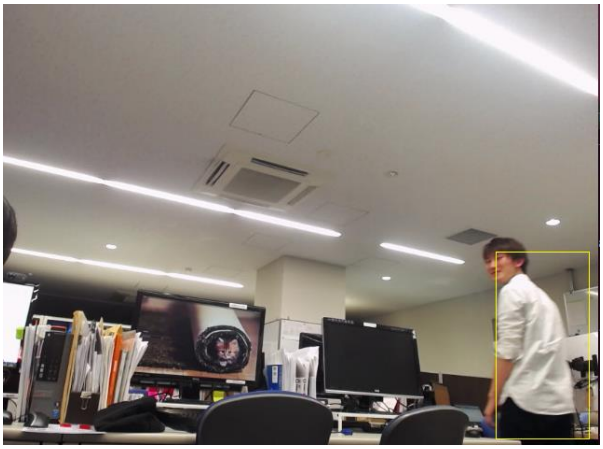
① rosrun my_image_converter image_converter
//ros で image_converter を実行する.

2. image_converter のソースコードの説明

- 画像中で動いている領域を推定し，矩形領域で囲う.
ex. 図では，動き出した人を推定している.

移動前

移動後



・ プログラムの処理の流れ

① usb カメラから取得した画像をグレースケール化する.

- 関数名：cvtColor(src, dst, code)
  - ✓ src：RGB 画像
  - ✓ dst：グレースケール画像
  - ✓ code：処理内容
    ※CV_BGR2GRAY でグレースケール化

② オプティカルフローを算出する.

- 関数名：opticalFlow->calc(src, dst, x, y)
  - ✓ src：現フレームの 1 つ前の画像
  - ✓ dst：現フレームの画像

- ✓ x：x 方向のフロー
- ✓ y：y 方向のフロー

➢ 関数名：cartToPolar(x, y, magnitude, angle, True)

- ✓ x：x 方向のフロー
- ✓ y：y 方向のフロー
- ✓ magnitude：オプティカルフローの強度
- ✓ angle：オプティカルフローの角度

③ 膨張・縮小処理をおこなう.

➢ 関数名：morphologyEx(src, dst, code, karnel, point, iter)

- ✓ src：入力画像
- ✓ dst：出力画像
- ✓ code：処理内容
- ✓ karnel：フィルタのサイズ
- ✓ point：フィルタの重心の指定
- ✓ iter：適用回数

④ 連続した領域をラベリングする.
※labeling.h は/opt/ros/kinetic/include/opencv-3.2.0-dev/opencv2 に置くこと.

➢ 関数名：labeling.Exec(src, dst, w, h, true, size)

- ✓ src：入力画像
- ✓ dst：出力画像
- ✓ w：画像の幅
- ✓ h：画像の高さ
- ✓ true：領域のソート
- ✓ size：小領域の消去（size 以下）

⑤ 矩形領域の表示

➢ 関数名：rectangle(src, point1, point2, scalar, line, num)

- ✓ src：入力画像
- ✓ point1：矩形の左上の座標
- ✓ point2：矩形の右上の座標
- ✓ scalar：矩形の線の色

- ✓ line：線の太さ
- ✓ num：連結する近傍数

3. 参考資料

- 【OpenCV】色変換(cvCvtColor)の組合せ
  - http://imagingsolution.blog107.fc2.com/blog-entry-242.html
- Opencv で密なオプティカルフローを計算する.
  - http://whoopsidaisies.hatenablog.com/entry/2013/12/15/020420
- OpenCV による画像の膨張と縮小
  - https://iwaki2009.blogspot.jp/2013/01/opencv_29.html
- ラベリングクラス
  - http://imura-lab.org/products/labeling/
- C++版 OpenCV で四角形を描画
  - http://opencv.blog.jp/cpp/draw_rectangle
- Github t11702m-chukyo
  - https://github.com/t11702m-chukyo