find_object_2d

Last update: 23/05/2019
Student name: Kate
Official find-object-2d package
http://wiki.ros.org/find_object_2d
http://introlab.github.io/find-object/
Follw the tutorial from husarion docs
https://husarion.com/tutorials/ros-tutorials/4-visual-object-recognition/

Table of Contents

Install find object 2d

```
sudo apt-get install ros-kinetic-find-object-2d
```

Create new package under the workspace

```
catkin_create_pkg replace_here_with_the_name_you_prefer std_msgs rospy roscpp
```

```
cd ~/catkin_ws
```

```
catkin build --continue-on-failure
```

```
. ~/catkin_ws/devel/setup.bash
```

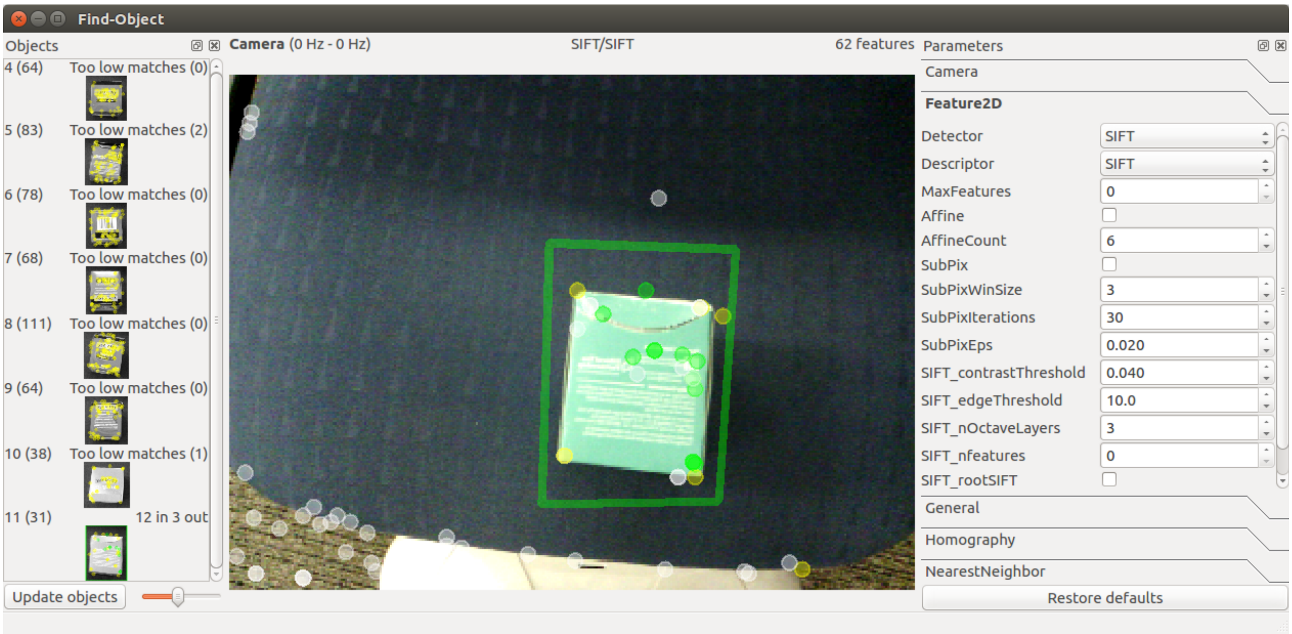
Create folder to store the data

```
mkdir data
```

Run find object 2d

```
rosrun find_object_2d find_object_2d image:=/pepper_robot/camera/front/image_raw
```

Turn on the parameters pane
View → Parameters
Under the Feature 2D, maksure both Detector and Descriptor are set to BRISK, or SFIT, or ORB
Do not use SURF, because once move the object outside of the view, then not be able to detect.
The pannle should look like this:



Find-Object

Objects

4 (64) Too low matches (0)
5 (83) Too low matches (2)
6 (78) Too low matches (0)
7 (68) Too low matches (0)
8 (111) Too low matches (0)
9 (64) Too low matches (0)
10 (38) Too low matches (1)
11 (31) 12 in 3 out

Camera (0 Hz - 0 Hz) SIFT/SIFT 62 features

Parameters

Camera

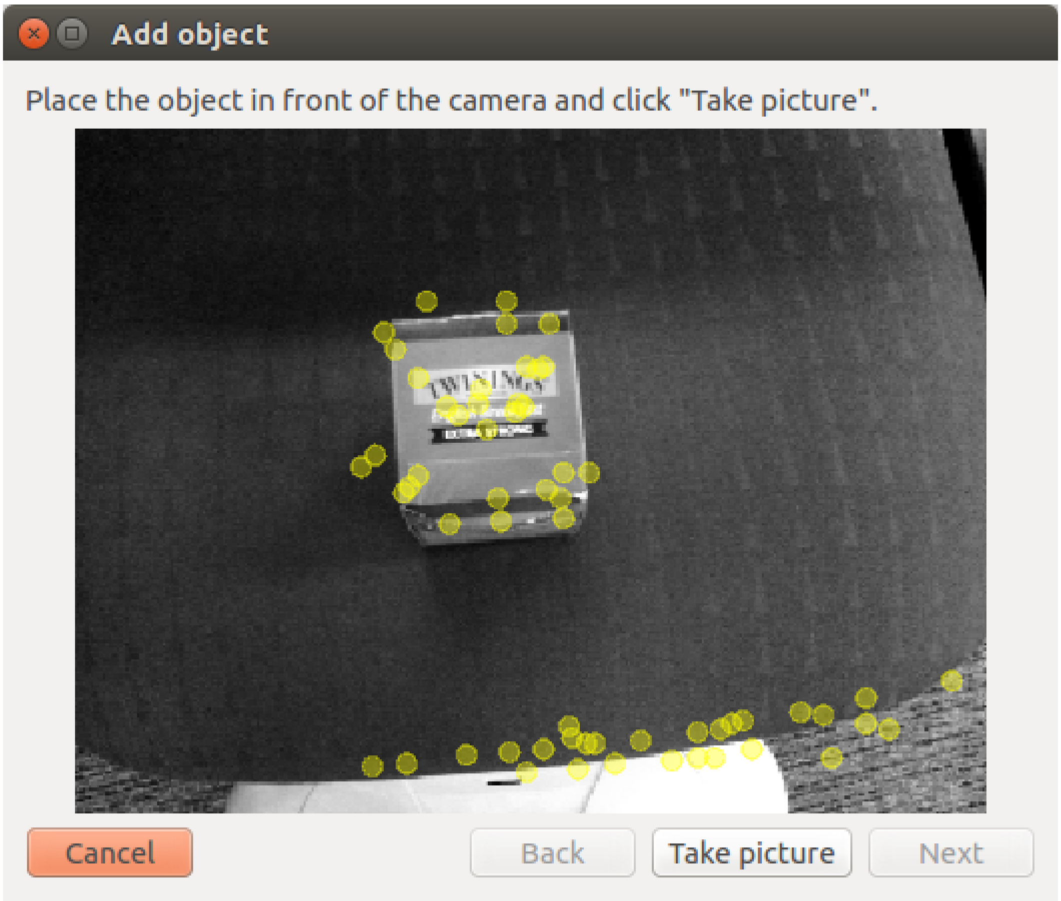
Feature2D

Detector: SIFT
Descriptor: SIFT
MaxFeatures: 0
Affine: □
AffineCount: 6
SubPix: □
SubPixWinSize: 3
SubPixIterations: 30
SubPixEps: 0.020
SIFT_contrastThreshold: 0.040
SIFT_edgeThreshold: 10.0
SIFT_nOctaveLayers: 3
SIFT_nfeatures: 0
SIFT_rootSIFT: □

General
Homography
NearestNeighbor

Add object for detection

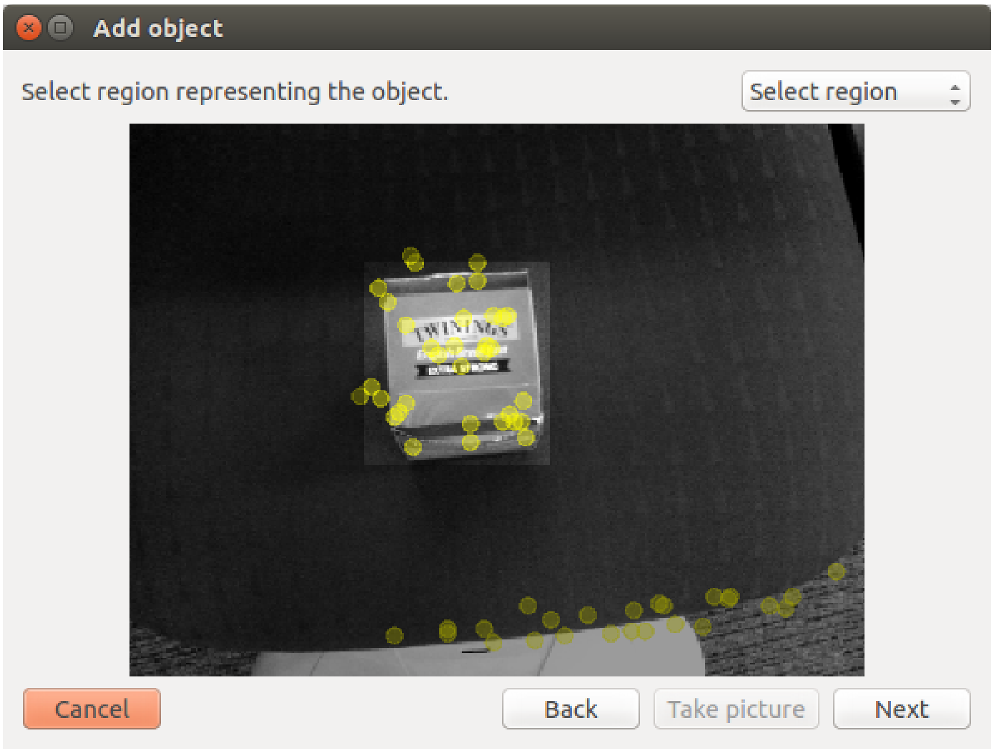
from Edit→Add objects from scence



Add object

Place the object in front of the camera and click "Take picture".

Take picture → Select region (select only side, try to not include other side view, still experimenting which is best)

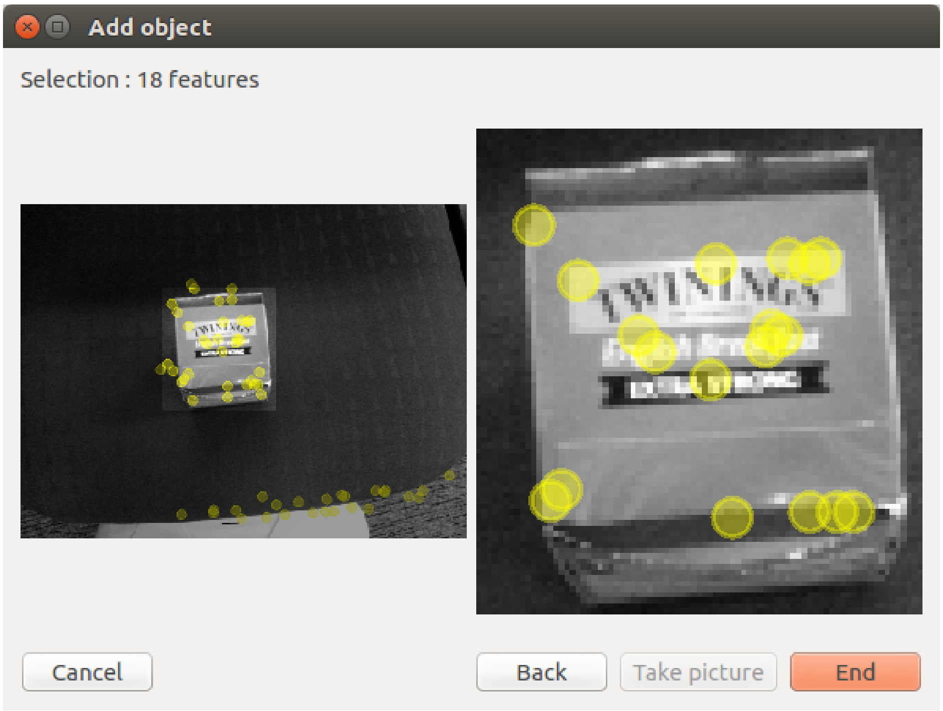


Add object

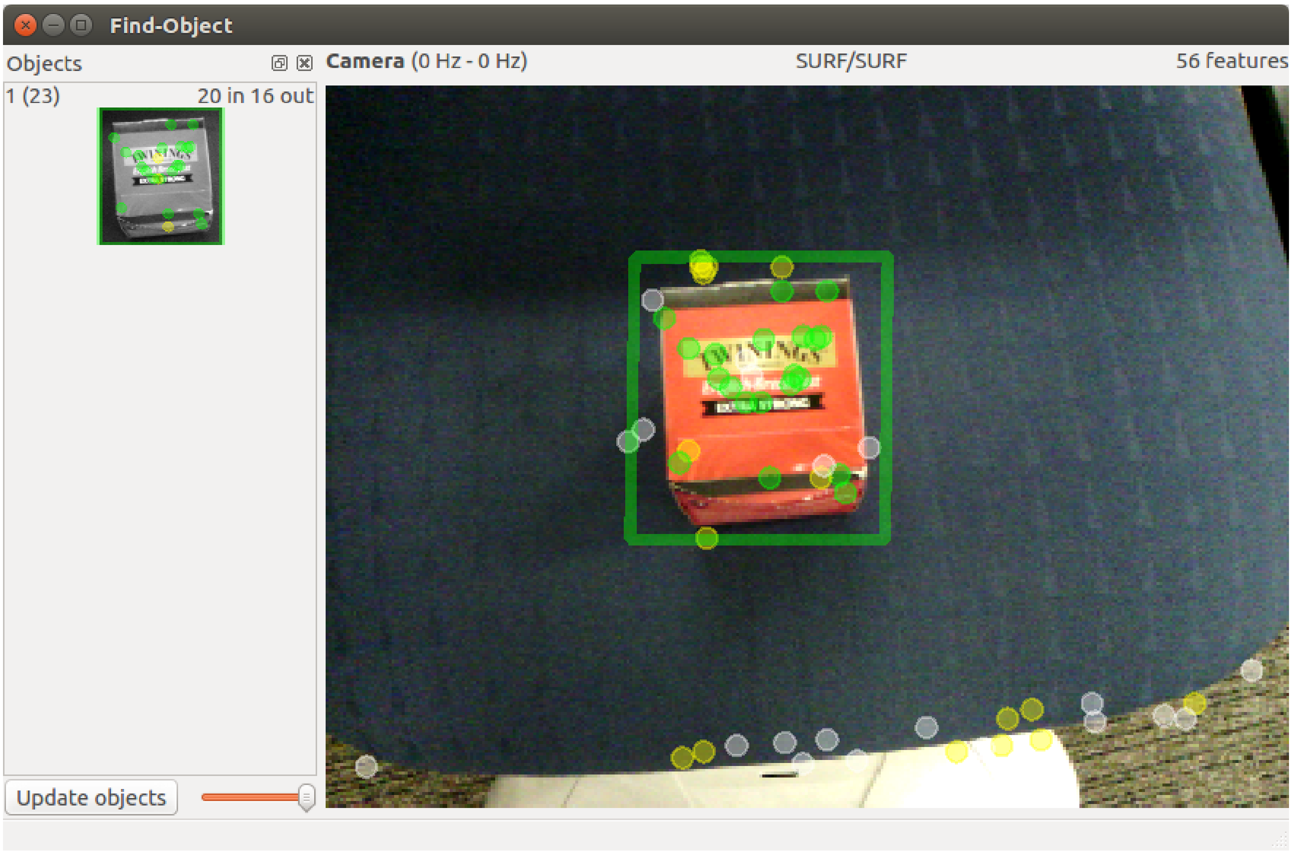
Select region representing the object.

Select region

Next (more freatures, easire to detect)



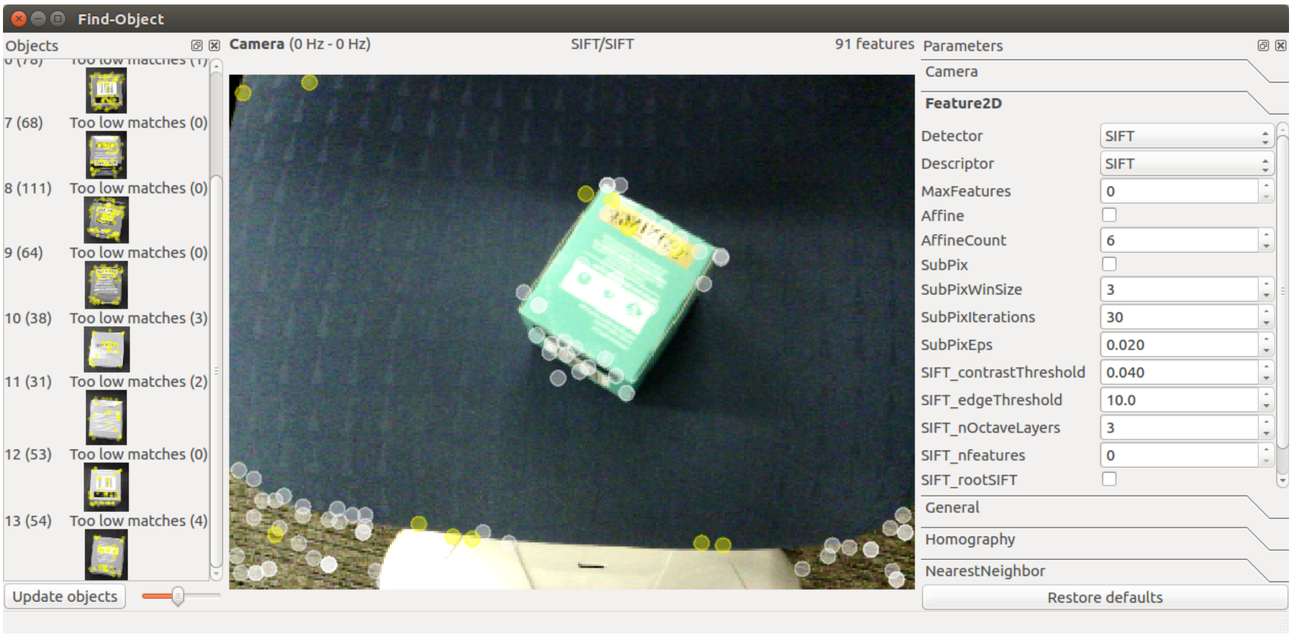
End



Save the objects into the desired folder

Downside of find object 2d package

The package is depending on the features of the object heavily instead of both feature and color. Therefore, when the color of the object are too light, or the contrast of the object are too less, and the room light condition which gives large amount of the reflection, then detected features can reduce dramatically. Following is the experiment with teabox that has light color, less contrast, and with reflection of the room light.
The object already added, so it should be able to detect the object no matter move to what direction. However it is not able to detect.


Find-Object

Objects

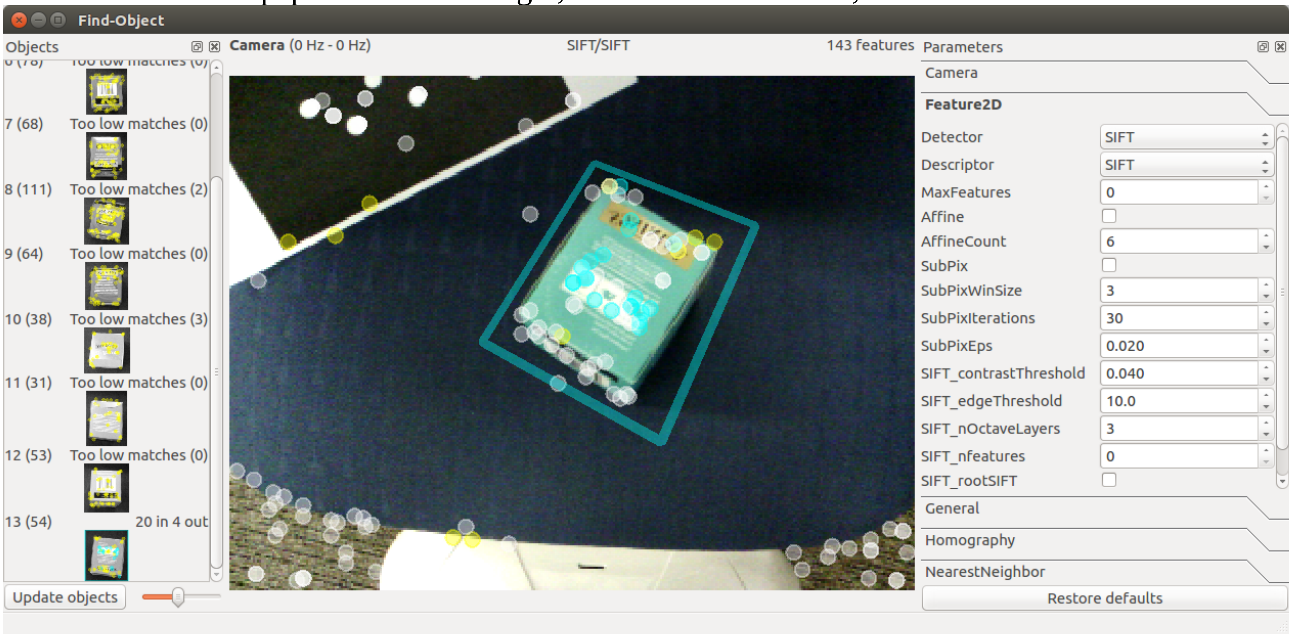
6 (78) Too low matches (1)
7 (68) Too low matches (0)
8 (111) Too low matches (0)
9 (64) Too low matches (0)
10 (38) Too low matches (3)
11 (31) Too low matches (2)
12 (53) Too low matches (0)
13 (54) Too low matches (4)

Update objects

Camera (0 Hz - 0 Hz) SIFT/SIFT 91 features

Parameters
Camera
Feature2D
Detector SIFT
Descriptor SIFT
MaxFeatures 0
Affine □
AffineCount 6
SubPix □
SubPixWinSize 3
SubPixIterations 30
SubPixEps 0.020
SIFT_contrastThreshold 0.040
SIFT_edgeThreshold 10.0
SIFT_nOctaveLayers 3
SIFT_nfeatures 0
SIFT_rootSIFT □
General
Homography
NearestNeighbor
Restore defaults

Once use a sheet of paper to cover the light, reduced the reflection, it then be able to detect.


Find-Object

Objects

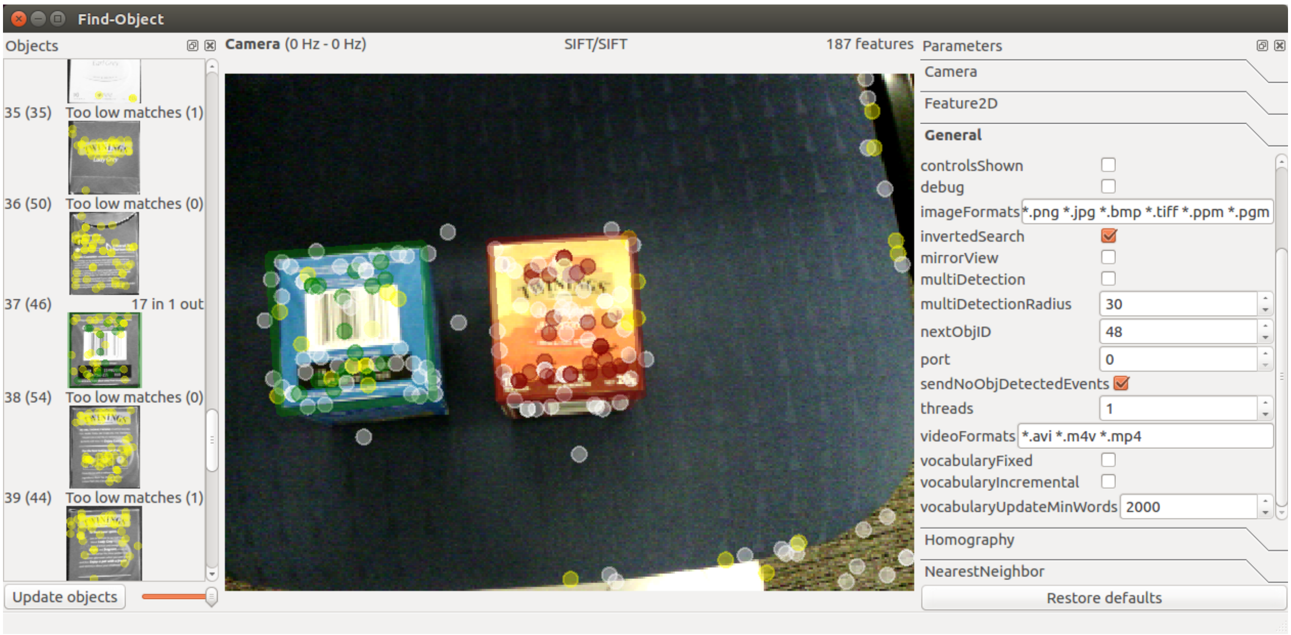
6 (78) Too low matches (0)
7 (68) Too low matches (0)
8 (111) Too low matches (2)
9 (64) Too low matches (0)
10 (38) Too low matches (3)
11 (31) Too low matches (0)
12 (53) Too low matches (0)
13 (54) 20 in 4 out

Update objects

Camera (0 Hz - 0 Hz) SIFT/SIFT 143 features

Parameters
Camera
Feature2D
Detector SIFT
Descriptor SIFT
MaxFeatures 0
Affine □
AffineCount 6
SubPix □
SubPixWinSize 3
SubPixIterations 30
SubPixEps 0.020
SIFT_contrastThreshold 0.040
SIFT_edgeThreshold 10.0
SIFT_nOctaveLayers 3
SIFT_nfeatures 0
SIFT_rootSIFT □
General
Homography
NearestNeighbor
Restore defaults

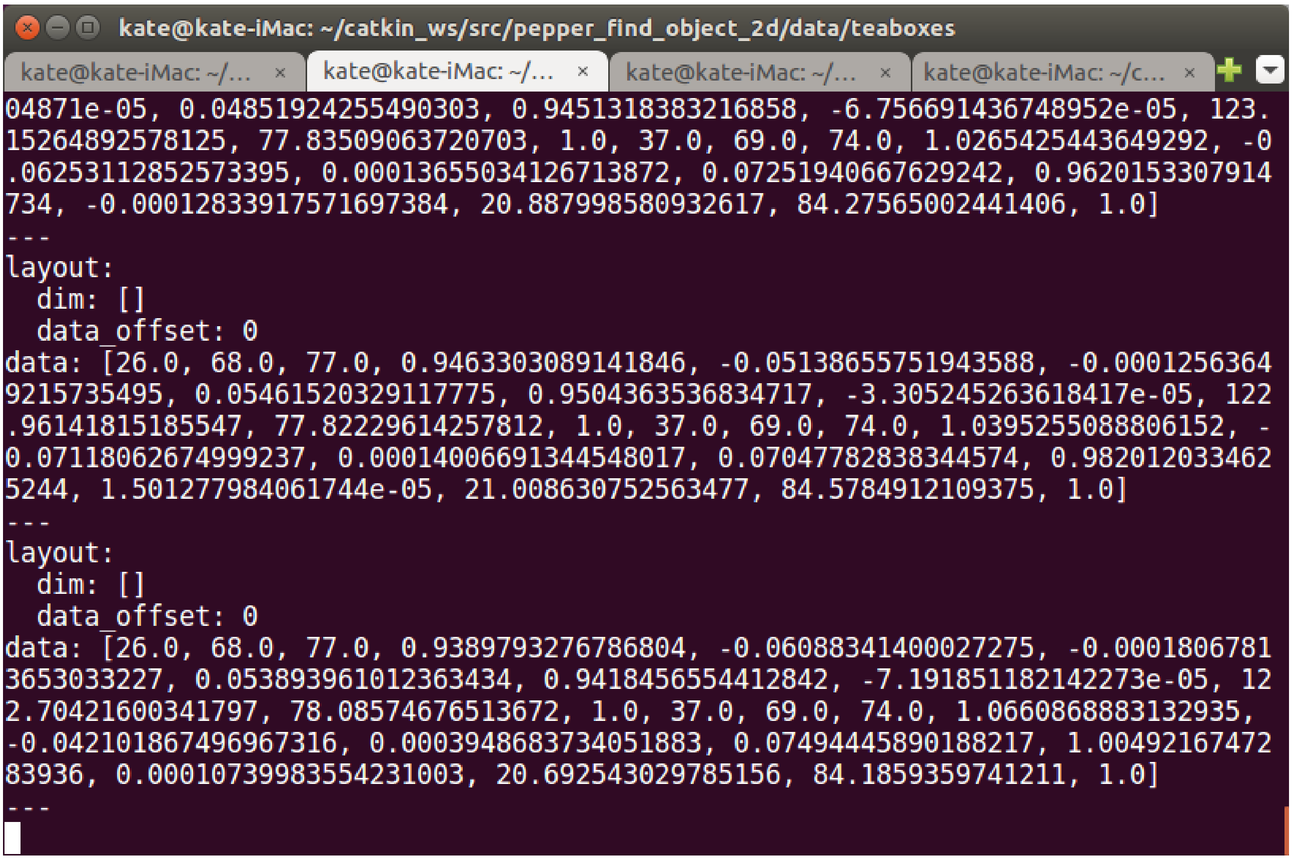
Output of multiple detected objects



controlsShown □
debug □
imageFormats *.png *.jpg *.bmp *.tiff *.ppm *.pgm
invertedSearch ☑
mirrorView □
multiDetection □
multiDetectionRadius 30
nextObjID 48
port 0
sendNoObjDetectedEvents ☑
threads 1
videoFormats *.avi *.m4v *.mp4
vocabularyFixed □
vocabularyIncremental □
vocabularyUpdateMinWords 2000

See the published data

rostopic echo /objects



```
04871e-05, 0.04851924255490303, 0.9451318383216858, -6.756691436748952e-05, 123.
15264892578125, 77.83509063720703, 1.0, 37.0, 69.0, 74.0, 1.0265425443649292, -0
.06253112852573395, 0.00013655034126713872, 0.07251940667629242, 0.9620153307914
734, -0.00012833917571697384, 20.887998580932617, 84.27565002441406, 1.0]
---
layout:
  dim: []
  data_offset: 0
data: [26.0, 68.0, 77.0, 0.9463303089141846, -0.05138655751943588, -0.0001256364
9215735495, 0.05461520329117775, 0.9504363536834717, -3.305245263618417e-05, 122
.96141815185547, 77.82229614257812, 1.0, 37.0, 69.0, 74.0, 1.0395255088806152, -
0.07118062674999237, 0.00014006691344548017, 0.07047782838344574, 0.982012033462
5244, 1.501277984061744e-05, 21.008630752563477, 84.5784912109375, 1.0]
---
layout:
  dim: []
  data_offset: 0
data: [26.0, 68.0, 77.0, 0.9389793276786804, -0.06088341400027275, -0.0001806781
3653033227, 0.053893961012363434, 0.9418456554412842, -7.191851182142273e-05, 12
2.70421600341797, 78.08574676513672, 1.0, 37.0, 69.0, 74.0, 1.0660868883132935,
-0.042101867496967316, 0.0003948683734051883, 0.07494445890188217, 1.00492167472
83936, 0.00010739983554231003, 20.692543029785156, 84.1859359741211, 1.0]
---
```

According to http://wiki.ros.org/find_object_2d
The data array is consist of [object id, object width, object height, h11, h12, h13, h21, h22, h23, h31, h32, h33, object2 id, …] where hxx is a 3x3 homography matrix

Create own node to work with the data from /objects

https://husarion.com/tutorials/ros-tutorials/4-visual-object-recognition/