

OpenCV and CVBridge



Prerequisites and Installation:

OpenCV is automatically comes with ROS. So, there's no need to install anything separately.

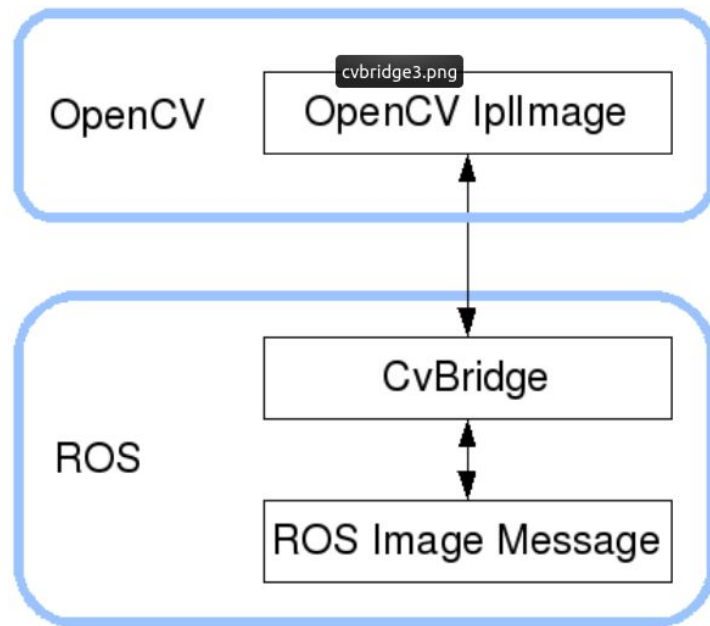
Prerequisite - prev theory lecture

OpenCV:

- OpenCV - Open Source Computer Vision Library
- OpenCV was started at Intel in 1999 by Gary Bradsky and the first release came out in 2000.
- In 2005, OpenCV was used on Stanley, the vehicle who won 2005 *DARPA Grand Challenge*.
- Primarily written in C++.
- Has bindings in Python, Java, MATLAB, and Javascript
- Source Code: <https://github.com/opencv/opencv>

CVBridge

- Image stored by ROS: sensor_msgs/Image
- Image stored by OpenCV: cv::Mat / numpy.array
- Bridge between ROS image transportation and OpenCV



Converting ROS image message to CV image:

```
from cv_bridge import CvBridge
```

```
bridge = CvBridge()
```

```
cv_image = bridge.imgmsg_to_cv2(image_message, desired_encoding='passthrough')
```

-> Passthrough encoding means the encoding of input and output image is same. But can obviously change the encoding. All the available options are given here:

http://wiki.ros.org/cv_bridge/Tutorials/ConvertingBetweenROSImagesAndOpenCVImagesPython

Converting CV image to ROS message:

```
from cv_bridge import CvBridge
```

```
bridge = CvBridge()
```

```
image_message = bridge.cv2_to_imgmsg(cv_image, encoding="passthrough")
```

Demo: cv.py in the zip file

Using Computer Vision in the real world:

1. Visual and Depth Sensors

- a. Kinect
- b. Intel Realsense
- c. ZED
- d. MyntEye S, etc.

2. Visual Sensor

- a. Simple Webcam
- b. PiCam, etc.

Using Computer Vision in the real world:

- Camera Driver
 - Its ROS wrapper
-
- Kinect : Openni or libfreenect
 - Realsense: librealsense and realsense-ros
 - ZED: ZED SDK and ZED ROS wrapper
 - Webcams: usb_cam
 - PiCam: PiCamera lib in python

OpenCV:

Best resource to learn about OpenCV-python: <https://opencv-python-tutroals.readthedocs.io>

Most of the applications have been discussed in the last lecture.

Jupyter-notebook : It's in the drive folder