How to calibrate monocular camera

★ Before Starting

- Make sure that you have a large checkerboard with known dimensions.
  - Calibration uses the interior vertex points of the checkerboard, so an "9x7" board uses the interior vertex parameter "8x6"
- And a monocular camera publishing images over ROS

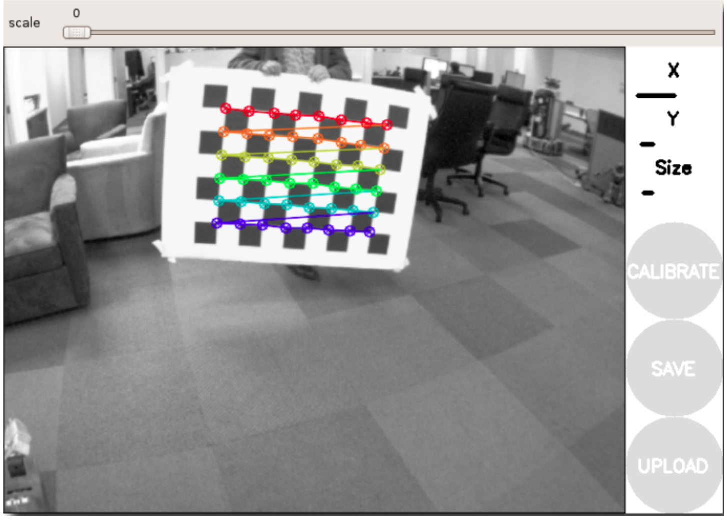
★ Compiling

```
$ rosdep install camera_calibration
```

★ Running the Calibration node

```
$ rosrun camera_calibration cameracalibrator.py --size 8x6
--square 0.108 image:=/camera/image_raw camera:=/camera
```

This will open up the calibration window which will highlight the checkerboard:



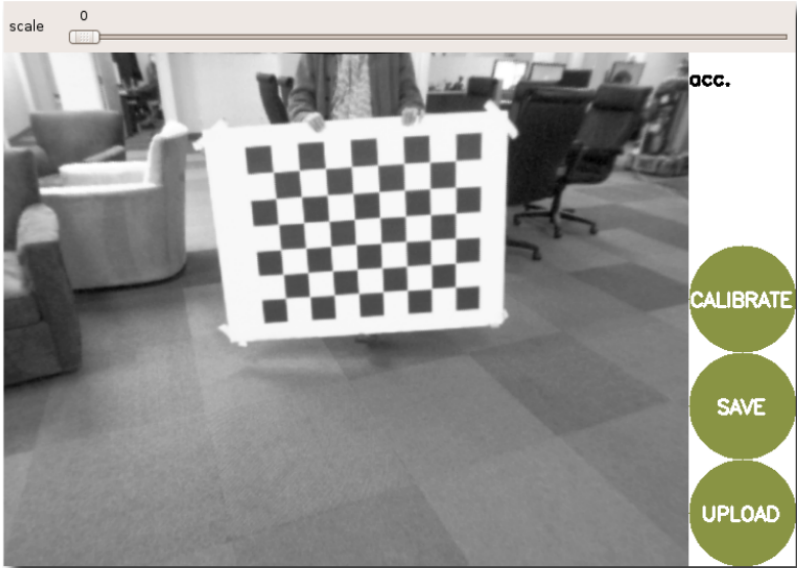
★ Moving the Check board

- In order to get a good calibration you will need to move the checkerboard around in the camera frame such that:
  - checkerboard on the camera's left, right, top and bottom of field of view
  - checkerboard filling the whole field of view
  - checkerboard tilted to the left, right, top and bottom (Skew)
- At each step, hold the checkerboard still until the image is highlighted in the calibration window.
- As you move the checkerboard around you will see three bars on the calibration sidebar increase in length.
- When the CALIBRATE button lights, you have enough data for calibration and can click CALIBRATE to see the results.

⇛ Calibration can take about a minute. The windows might be greyed out but just wait, it is working.

★ Calibration Results

⇛ After the calibration is complete you will see the calibration results in the terminal and the calibrated image in the calibration window:



⇛ If you are satisfied with the calibration, click COMMIT to send the calibration parameters to the camera for permanent storage.

⇛ The GUI exits and you should see "writing calibration data to ..." in the console.

★ Creating a YAML file

⇛ The Camera Calibration Parser helps you to create a yml file, which you can load with nearly all ros camera driver using the camera_info_url parameter.