Calibration Methodology

I. Introduction

This document describes how to calibrate the stereo camera rig in the simulated phonomicrosurgery setup. The stereo camera rig consists of two *Basler aCA640-100gc* machine vision cameras equipped with 24 mm lenses. Calibration consists of :

1. Properly positioning the laryngoscope and cameras so of the phonomicrosurgery instruments during an exercise can be properly captured.
2. Acquiring a set of images of a checkerboard calibration pattern.
3. Estimating the stereo camera rig calibration parameters using the Caltech Camera Calibration Toolbox for Matlab.

All of these steps are described in this document. After estimating the calibration parameters, a file named *Calib_Results_stereo.mat* will be generated. This file is used by different Matlab routines to estimate 3D instrument position from 2D tracking data.

II. Laryngoscope and Microscope Positioning

Prior to adjusting the cameras, you should check the positioning of the laryngoscope. It should be oriented ~ 20 degrees below level. A protractor can be used to verify this. You should be able to orient the microscope so that a subject can view directly down the scope. The depth of the scope will be adjusted in future steps. If the laryngoscope and microscope are setup properly, changes in laryngoscope depth should not change the microscope's field of view. The depth of the laryngoscope is controlled by adjusting the position of the rail identified in **Figure 1**.

Figure 1 : Laryngoscope Rail for Depth Adjustment

III. StereoScreenGrab Application

StereoScreenGrab is an application used for viewing and capturing images from the stereo camera rig. Open the program at the following location :

C:\Simulated_Surgery_Software\Calibration\StereoScreenGrab.exe

First, a console window will open. The program will connect to the cameras and two additional windows will open. These windows contain live views from the left and right camera. Screen captures of the live views can be taken by pressing the "s" key. One of the live view windows must be selected for the keystroke to register with the application. An image for the left and right camera will be saved when this is done. A folder is created in the same directory as the application in which captured images are saved. This folder has the form [Year]_[Month]_[Day]_[Index]. The [index] part is used to ensure that the folder is unique to application instance. When the application is started, the folder name is displayed in the console window. Additionally, the name of captured images are displayed in the command window when they are captured. To close the application, press the "q" key or close the console window. You will use the application to capture images in a later step, for now it will be used to view the surgical setup.

IV. Camera Positioning Setup

a. Camera Positioning

You want to position/orient cameras such that instruments fall within the FOV of both cameras. This can be done by positioning the cameras with respect to the tip of the laryngoscope. Adjust the depth of the scope along its rail so it appears in the FOV of both cameras. Adjust the positioning and orientation of both cameras so the tip of scope is completely viewable. It should hug the "inside" (right side of left camera view and left side of right camera view) of both views. **Figure 2** shows an example of this.

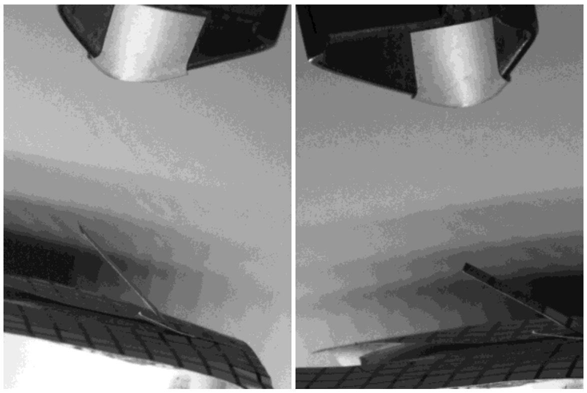


Figure 2 : Laryngoscope Positioning for Camera Alignment

To verify that the instrument will stay in view, insert an instrument into the scope and move it along the width of the scope. The instrument should stay within the horizontal limits of both views. If it does not, adjust the orientation of each camera to optimize the field of view with respect to the instrument and scope. After properly setting the cameras, adjust the scope depth along its rail so the tip of the scope is barely out of view. Sweep the instrument again and verify that it stays in both cameras' field of view. **Figure 3** shows the instrument at the beginning and end of a sweep in one view. At the horizontal extremes of the scope the instrument is still viewable. Note also in this figure, the depth of the scope has been adjusted so that it is barely out of view.

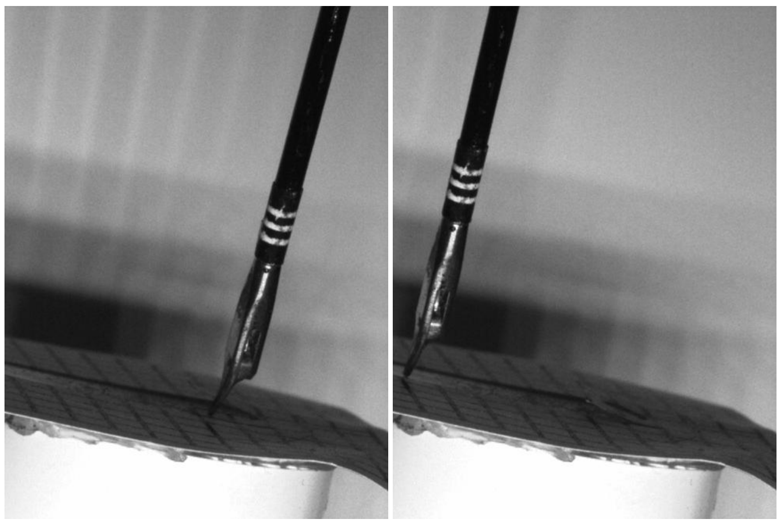


Figure 3 : Field of View Check by Sweeping Instrument

The cameras are now properly positioned. If you have loosened any positioning mechanisms, make sure to re-tighten them after finding the correct positioning/orientation.

b. Camera Focus

The focus of each lens is controlled by rotating its outer ring. A small black screw holds the lens focus ring in place. Hold an instrument within the scope and adjust each camera's focus. Stop adjustment when the instrument is in focus. As it goes out of focus, the instrument's edges will blur. Good focus is indicated by sharp edges.

V. Calibration Rig Setup

a. Calibration Rig

Prior to running any calibration routines, a calibration dataset must be acquired. This dataset consists of a set of images of a calibration pattern at different positions and orientations. A planar checkerboard pattern is used for calibration in this project. **Figure 4** shows the mechanical rig used to position the pattern within the surgical setup.

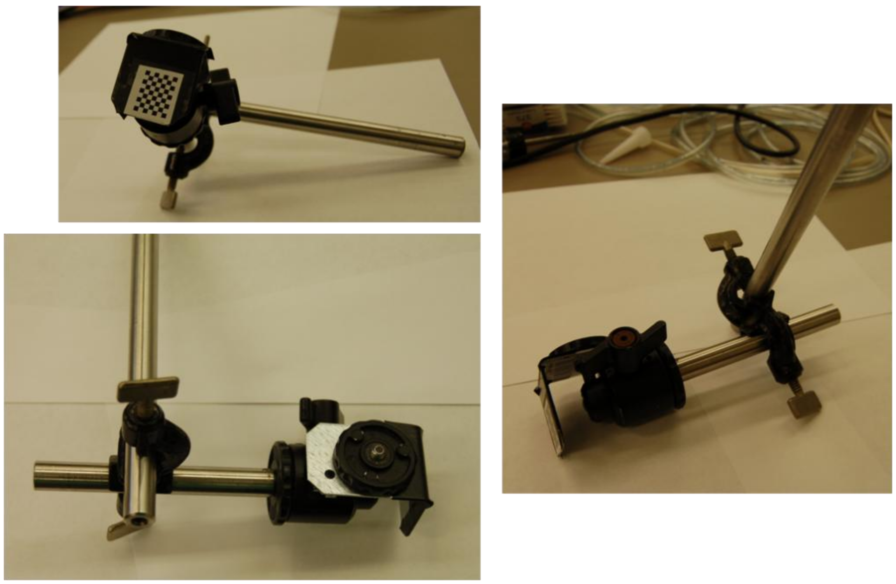


Figure 4 : Calibration Rig

The rig consists of the pattern connected to a tripod head and two adjustable rods. The tripod head is used to adjust the orientation of the checkerboard pattern. The two rods allow the depth and the horizontal position of the tripod head to be adjusted.

b. Calibration Rig Positioning

The calibration rig needs to be placed in the volume captured by the cameras during a surgical exercise. First, space must be made to properly position the rig. First, the background plane must be removed. In the setup, the background plane is held by a claw. This claw is connected to a clamp connected to a 3/4" diameter vertical post to the right of the laryngeal dissection station. Disconnect the claw from the clamp . **Figure 5** shows the background plane, claw and clamp.

Next, remove the piece holding the exercise task and adjust the depth of the laryngoscope so it is further from the viewing volume. Connect the calibration rig to clamp on the right vertical post as seen in **Figure 6**.

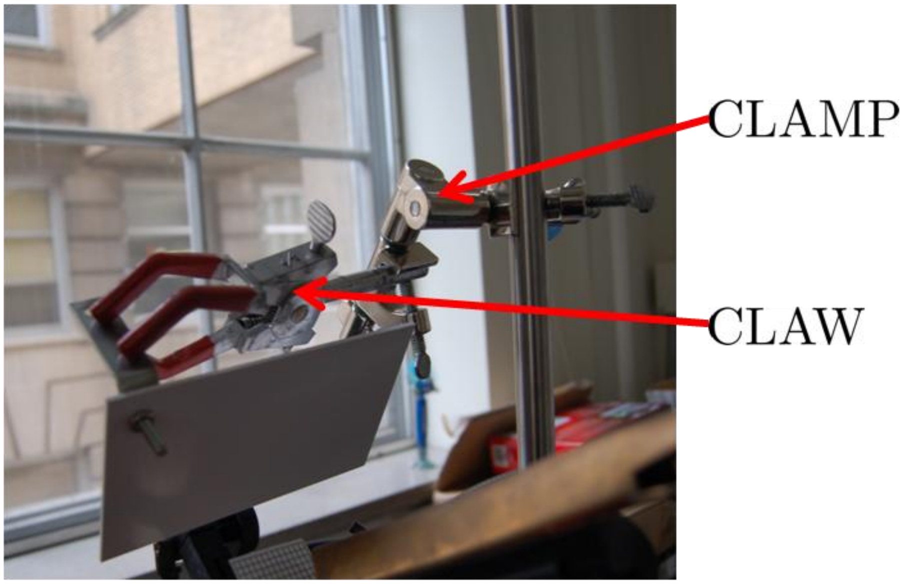


Figure 5 : Background Plane and Clamp

Figure 6 : Calibration Rig Connected to Clamp

Position the calibration rig so that the checkerboard pattern is in the field of view of both cameras. Turn on the LED array lighting. You should be able to see the calibration pattern in focus in both camera views. **Figure 7** shows this positioning.

Figure 7 : Calibration Rig Positioning for Image Capture

VI. Image Capture Procedure

You are now ready to capture the calibration dataset. Position the checkerboard pattern so that it is oriented vertically in the left camera's view. Position the checkerboard so that it is near the right edge of the left camera's view. Slightly adjust the positioning so that it is viewable in both camera's views. **Figure 8** shows an example of this.

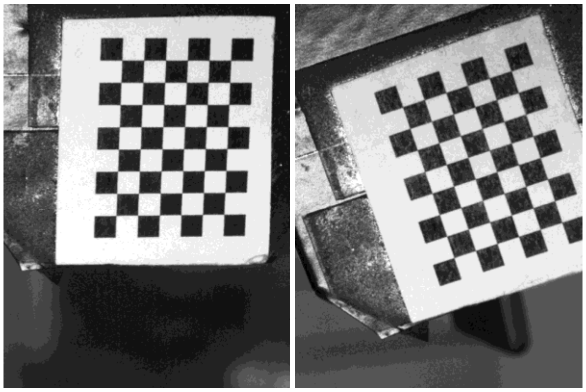


Figure 8 : Initial Pattern Positioning Left and Right Camera

In all calibration images, the inner checkerboard enclosed by a red rectangle in **Figure 9** needs to be visible in both views. To capture an image, select one of the camera view windows and press the “s” key. The name of the two images saved will appear in the console window. Take 10-12 images of the pattern. Adjust the orientation of the pattern between images. Now reposition the checkerboard so it is oriented vertically in the right camera’s view. Position it so it near the left edge of the view and visible in the left camera’s view. Capture another 10-12 images of the pattern at different orientations. Remember the inner checkerboard in **Figure 9** needs to be visible in both views. You have now collected the calibration dataset. You can reconfigure the surgical setup for simulated exercises now.

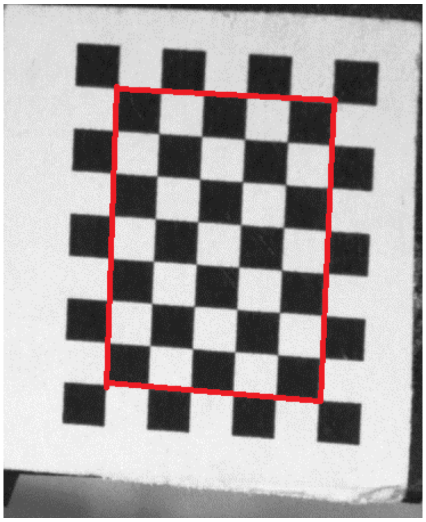


Figure 9 : Interior Checkerboard

VII. Camera Calibration Toolbox Procedure

After collecting the calibration dataset, a set of Matlab routines are used to estimate calibration parameters of the stereo camera rig. These parameters are needed to estimate 3D movement information from 2D tracking data. The routines used are from the Caltech Camera Calibration Toolbox for Matlab. More information on using the toolbox can be found at : http://www.vision.caltech.edu/bouguetj/calib_doc/. While this document explains how to use the toolbox, it is highly recommended that you view the section on calibration parameters, the first calibration example, and the stereo calibration example on this webpage.

a. Matlab Setup

Open the latest version of Matlab. First, you need to add the calibration toolbox routines to the Matlab system path. Set the current Matlab folder to the following path :

C:\Simulated_Surgery_Software\Calibration\CamCalToolbox\toolbox_calib

To add the toolbox to the Matlab path, enter the following command at the command line :

```
>> addpath(cd)
```

Now, change the current Matlab folder to the directory containing the calibration dataset. There should be a set of *.tif* images in this directory corresponding to the left {left_1.tif,left_2.tif,...} and right {right_1.tif,right_2.tif,...} camera. To start the calibration toolbox, enter the following command at the command line :

>> calib_gui

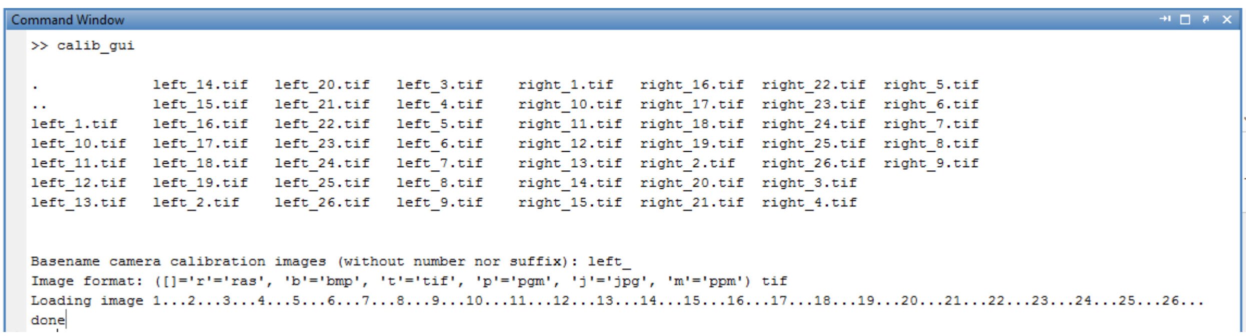
A window will open prompting you to select the toolbox's operation mode. Select the **Standard** mode button. The **Camera calibration tool** window will now open.

b. Individual Camera Calibration

First, you will calibrate the left and right camera individually. For each camera you will save a *.m* and *.mat* calibration file. In this subsection, the steps to perform the individual calibration for the left camera will be described. The *left_[number].tif* dataset will be used. To perform the individual calibration of the right camera, the steps in this subsection need to be repeated with the *right_[number].tif* dataset.

1) Read the Images

Click on the **Image names** button in the **Camera calibration tool** window. Enter the basename of the calibration images (**left_**) and the image format (**tif**).



```
Command Window
>> calib_gui

.            left_14.tif  left_20.tif  left_3.tif   right_1.tif   right_16.tif  right_22.tif  right_5.tif
..           left_15.tif  left_21.tif  left_4.tif   right_10.tif  right_17.tif  right_23.tif  right_6.tif
left_1.tif   left_16.tif  left_22.tif  left_5.tif   right_11.tif  right_18.tif  right_24.tif  right_7.tif
left_10.tif  left_17.tif  left_23.tif  left_6.tif   right_12.tif  right_19.tif  right_25.tif  right_8.tif
left_11.tif  left_18.tif  left_24.tif  left_7.tif   right_13.tif  right_2.tif   right_26.tif  right_9.tif
left_12.tif  left_19.tif  left_25.tif  left_8.tif   right_14.tif  right_20.tif  right_3.tif
left_13.tif  left_2.tif   left_26.tif  left_9.tif   right_15.tif  right_21.tif  right_4.tif


Basename camera calibration images (without number nor suffix): left_
Image format: ([]='r'='ras', 'b'='bmp', 't'='tif', 'p'='pgm', 'j'='jpg', 'm'='ppm') tif
Loading image 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...19...20...21...22...23...24...25...26...
done
```

Figure 10 : Read Images Command Line

All the images are then loaded in memory. The complete set of images is also shown in thumbnail format. You can close the thumbnail window if you wish.

2) Extract Grid Corners

Click on the **Extract grid corners** button in the **Camera calibration tool** window. Press "enter" (with an empty argument) to select all images (otherwise, you would enter a list of image indices like [2 5 8 10 12] to extract corners of a subset of images). Then, select the default window size of the corner finder: **wintx=winty=7** by pressing "enter" with empty arguments to the **wintx** and **winty** question. The corner extraction engine includes an automatic mechanism for counting the number of squares in the grid. This tool is specially convenient when working with a large number of images since the user does not have to manually enter the number of squares in both x and y directions of the pattern. Press "enter" with an empty argument to have the toolbox run with the automatic square counting mechanism. The only time you would not need this, is in the case of extreme lens distortion. Because high quality machine vision lenses are used in the surgical setup, you do not need to worry about this.

```
Extraction of the grid corners on the images
Number(s) of image(s) to process ([] = all images) =
Window size for corner finder (wintx and winty):
wintx ([] = 7) =
winty ([] = 7) =
Window size = 15x15
Do you want to use the automatic square counting mechanism (0=[]=default)
  or do you always want to enter the number of squares manually (1,other)?
```

Figure 11 : Corner Extraction Setup Command Line

A figure window with the first calibration image will open. The title of this figure will be "Click on the four extreme corners of the rectangular pattern (first corner = origin)". Click on the four extreme corners on the rectangular checkerboard pattern (see ordering rule below first). The clicking locations are shown in the **Figure 11**. (**WARNING:** try to click accurately on the four corners, at most 7 pixels away from the corners. Otherwise some of the corners might be missed by the detector). It is helpful to maximize this window prior to selecting corners.

Ordering rule for clicking: The first clicked point is selected to be associated to the origin point of the reference frame attached to the grid. The other three points of the rectangular grid can be clicked in any order. This first-click rule is especially important because you are going to calibrate a stereo camera rig. For each image, the same grid pattern reference frame needs to be consistently selected for the different camera images (i.e. grid points need to correspond across the different camera views). Therefore, whatever point on the checkerboard is selected as the origin (first click) in the first calibration image must be selected in all proceeding images for the left and right camera images.

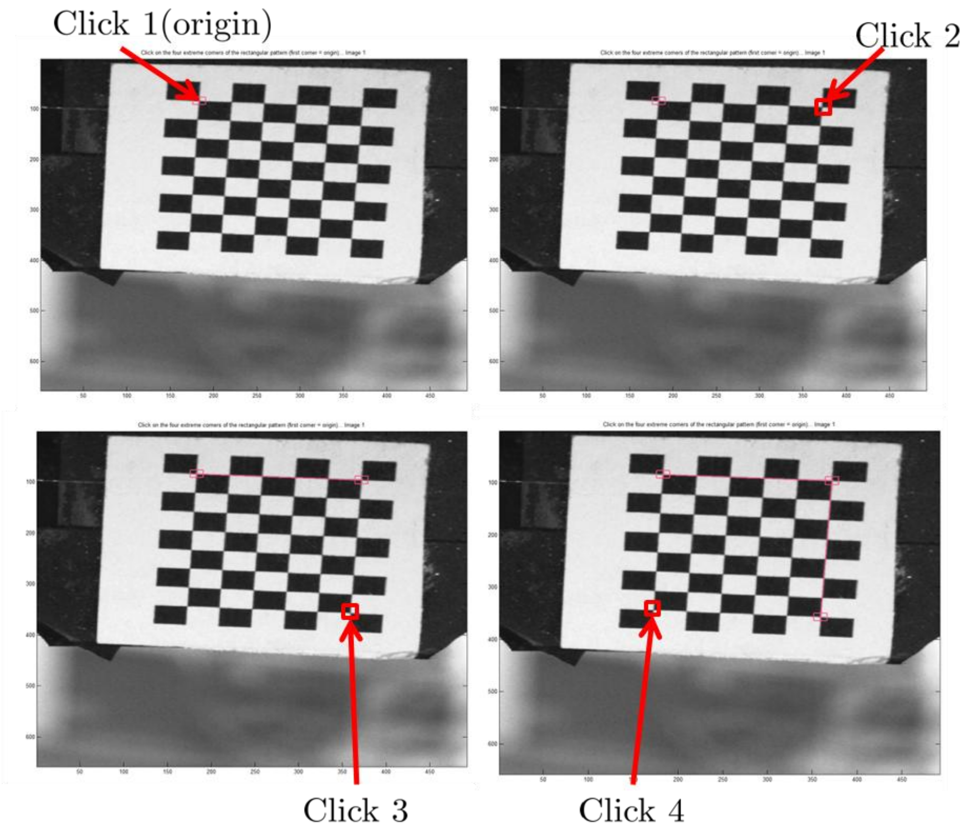


Figure 11 : Corner Clicking Order

After clicking the four corners, the boundary of the calibration grid will be shown in the window. At the command line you will be prompted to enter the sizes **dX** and **dY** in X and Y of each square in the grid. Enter the appropriate checkerboard side length in mm. As of 11/27/2011 a checkerboard with 2.1167 mm side-lengths was being used. The selection window will now be updated with the predicted corner locations as a set of red crosses. The predicted corners should be close to the actual checkerboard corners. The program will give you an option to guess a radial distortion coefficient if they are not. You do not need to do this, simply press "enter" with an empty argument when prompted for this.

```
Processing image 1...
Using (wintx,winty)=(7,7) - Window size = 15x15      (Note: To reset the window size, run script clearwin)
Click on the four extreme corners of the rectangular complete pattern (the first clicked corner is the origin)...
Size dX of each square along the X direction ([]=100mm) = 2.1167
Size dY of each square along the Y direction ([]=100mm) = 2.1167
If the guessed grid corners (red crosses on the image) are not close to the actual corners,
it is necessary to enter an initial guess for the radial distortion factor kc (useful for subpixel detection)
Need of an initial guess for distortion? ([]=no, other=yes)
Corner extraction...
```

Figure 12 : Corner Extraction Command Line

A figure window will open with the automatically extracted corner locations. The program is now ready to extract corners in the second calibration image. Repeat the corner extraction procedure for the remaining images. You do not need to set the **dX** and **dY** values after the first image. Remember to consistently select the same checkerboard corners and select the same origin point first.

3) Main Calibration Setup

After corner extraction, click on the **Calibration** button of the **Camera calibration tool** to run the main camera calibration procedure. Calibration is done in two steps: first initialization, and then nonlinear optimization. The initialization step computes a closed-form solution for the calibration parameters. The non-linear optimization step minimizes the total reprojection error (in the least squares sense) over all the calibration parameters. The optimization is done by iterative gradient descent with an explicit (closed-form) computation of the Jacobian.

```
Aspect ratio optimized (est_aspect_ratio = 1) -> both components of fc are estimated (DEFAULT).
Principal point optimized (center_optim=1) - (DEFAULT). To reject principal point, set center_optim=0
Skew not optimized (est_alpha=0) - (DEFAULT)
Distortion not fully estimated (defined by the variable est_dist):
     Sixth order distortion not estimated (est_dist(5)=0) - (DEFAULT) .
Initialization of the principal point at the center of the image.
Initialization of the intrinsic parameters using the vanishing points of planar patterns.

Initialization of the intrinsic parameters - Number of images: 26


Calibration parameters after initialization:

Focal Length:          fc = [ 5065.67865   5065.67865 ]
Principal point:       cc = [ 245.50000   328.50000 ]
Skew:             alpha_c = [ 0.00000 ]   => angle of pixel = 90.00000 degrees
Distortion:            kc = [ 0.00000   0.00000   0.00000   0.00000   0.00000 ]

Main calibration optimization procedure - Number of images: 26
Gradient descent iterations: 1...2...3...4...5...6...7...8...9...10...11...12...13...14...15...16...17...18...19...20...21...22...23...24...25...
Estimation of uncertainties...done


Calibration results after optimization (with uncertainties):

Focal Length:          fc = [ 5063.91959   5071.89822 ] ± [ 104.56112   79.38294 ]
Principal point:       cc = [ 256.46164   480.98843 ] ± [ 85.99802   142.32387 ]
Skew:             alpha_c = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc = [ -0.44321   -21.44653   -0.00302   -0.00834  0.00000 ] ± [ 0.40012   34.28720   0.01806   0.00818  0.00000 ]
Pixel error:          err = [ 0.11797   0.11138 ]

Note: The numerical errors are approximately three times the standard deviations (for reference).
```

Figure 13 : Calibration Command Line

Enter the following commands into the command line

>> est_dist(:) = 0

>> center_optim = 0

>> clear cc

Now, rerun the parameter estimation by pressing the **Calibration** button. The toolbox will estimate the calibration parameters for a simpler camera model.

4) Save Calibration Data

Select the **Save** button to save the individual camera calibration data. A file named *Calib_Results.m* and *Calib_Results.mat* will be saved in the current folder. Rename the files as *Calib_Results_left.m* and

Calib_Results_left.mat (replace "left" with "right" if you are repeating the steps for the set of right camera images).

c. Stereo Camera Calibration

After performing individual calibration for the left and right camera there should be four calibration files in the current folder : *Calib_Results_left.m*, *Calib_Results_left.mat*, *Calib_Results_right.m*, *Calib_Results_right.mat*. These four files will be used for estimation stereo calibration parameters. If the **Camera calibration tool** window is still open close it. Now run the following command at the command line to open the **stereo camera calibration tool** :

>> stereo_gui

Select the **Load left and right calibration files** button. You will be prompted at the command line for the name of the left and right calibration data files. Press enter with an empty argument to use the default file names. The initial stereo calibration data will be displayed.

```
>> stereo_gui

Calib_Results_left.mat   Calib_Results_right.mat  calib_data.mat

Loading of the individual left and right camera calibration files
Name of the left camera calibration file ([]=Calib_Results_left.mat):
Name of the right camera calibration file ([]=Calib_Results_right.mat):
Loading the left camera calibration result file Calib_Results_left.mat...
Loading the right camera calibration result file Calib_Results_right.mat...


Stereo calibration parameters after loading the individual calibration files:


Intrinsic parameters of left camera:

Focal Length:          fc_left = [ 5141.22440   5129.95964 ] ± [ 82.45640   86.03392 ]
Principal point:       cc_left = [ 245.50000   328.50000 ] ± [ 0.00000   0.00000 ]
Skew:             alpha_c_left = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc_left = [ 0.00000   0.00000   0.00000   0.00000  0.00000 ] ± [ 0.00000   0.00000   0.00000   0.00000  0.00000 ]


Intrinsic parameters of right camera:

Focal Length:          fc_right = [ 5182.00491   5192.20549 ] ± [ 86.59482   89.59404 ]
Principal point:       cc_right = [ 245.50000   328.50000 ] ± [ 0.00000   0.00000 ]
Skew:             alpha_c_right = [ 0.00000 ] ± [ 0.00000  ]   => angle of pixel axes = 90.00000 ± 0.00000 degrees
Distortion:            kc_right = [ 0.00000   0.00000   0.00000   0.00000  0.00000 ] ± [ 0.00000   0.00000   0.00000   0.00000  0.00000 ]


Extrinsic parameters (position of right camera wrt left camera):

Rotation vector:             om = [ -0.01848   0.41334  -0.36251 ]
Translation vector:           T = [ -107.99005   16.49140   18.65167 ]
```

Figure 14 : Stereo Calibration Command Line

Select the **Run stereo calibration** button. The final calibration parameters will be estimated and displayed in the command window. To save the parameters select the **Save stereo calib results** button.

A file named *Calib_Results_stereo.mat* will be saved in the current folder. This file will be used by Matlab routines to estimate 3D information from 2D tracking data.

VIII. Conclusion

After calibrating the simulated surgery setup, you are ready to capture videos of surgical exercises. The stereo camera calibration file (*Calib_Results_stereo.mat*) will be used by Matlab routines to estimate 3D instrument position from 2D tracking data. You may rename this file to be more descriptive.