Single Camera Calibration App

Camera Calibrator Overview

You can use the camera calibrator to estimate camera intrinsics, extrinsics, and lens distortion parameters. You can use these camera parameters for various computer vision applications. These applications include removing the effects of lens distortion from an image, measuring planar objects, or reconstructing 3-D scenes from multiple cameras.

Single Camera Calibration Workflow

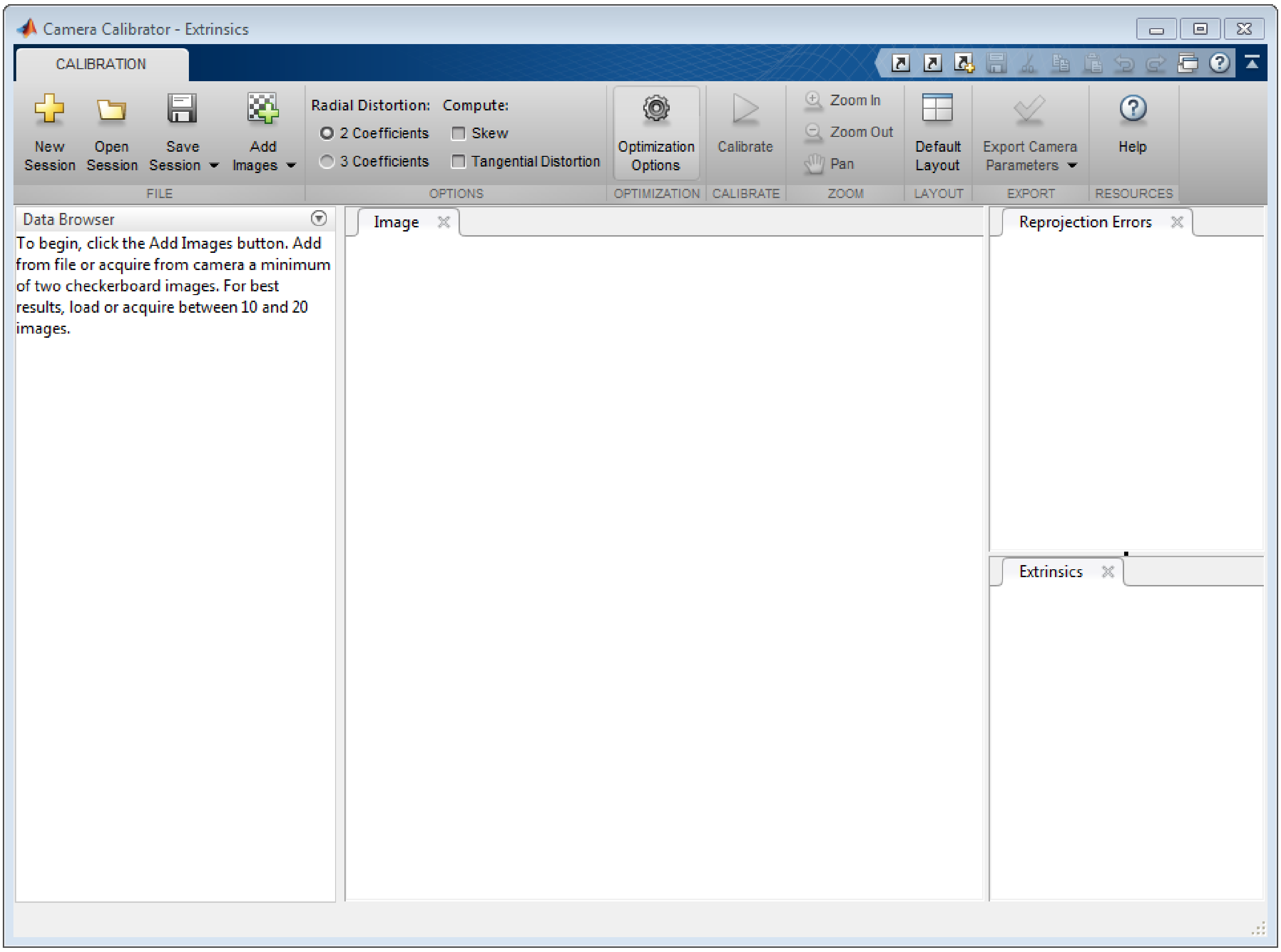
Follow this workflow to calibrate your camera using the app:

1. Prepare images, camera, and calibration pattern.
2. Load images.
3. Calibrate the camera.
4. Evaluate calibration accuracy.
5. Adjust parameters to improve accuracy (if necessary).
6. Export the parameters object.

In some cases, the default values work well, and you do not need to make any improvements before exporting parameters. If you do need to make improvements, you can use the camera calibration functions in MATLAB®. For a list of functions, see Single Camera Calibration.

Open the Camera Calibrator

- MATLAB Toolstrip: Open the Apps tab, under **Image Processing and Computer Vision**, click the app icon.
- MATLAB command prompt: Enter `cameraCalibrator`



Prepare the Pattern, Camera, and Images

For best results, use between 10 and 20 images of the calibration pattern. The calibrator requires at least three images. Use uncompressed images or lossless compression formats such as PNG. The calibration pattern and the camera setup must satisfy a set of requirements to work with the calibrator. For greater calibration accuracy, follow these instructions for preparing the pattern, setting up the camera, and capturing the images.

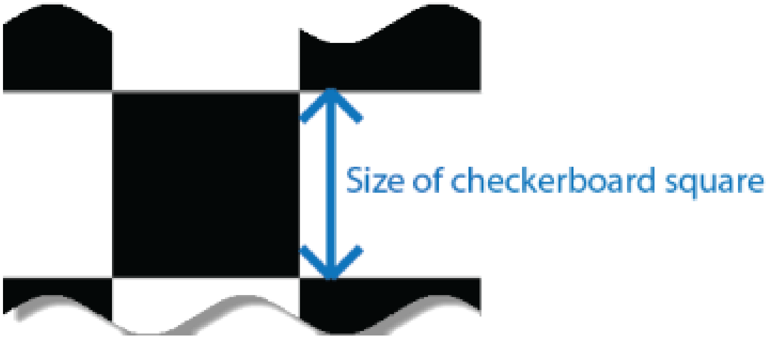
Prepare the Checkerboard Pattern

The Camera Calibrator app uses a checkerboard pattern. A checkerboard pattern is a convenient calibration target. If you want to use a different pattern to extract key points, you can use the camera calibration MATLAB functions directly. See Single Camera Calibration for the list of functions.

You can print (from MATLAB) and use the checkerboard pattern provided. The checkerboard pattern you use must not be square. One side must contain an even number of squares and the other side must contain an odd number of squares. Therefore, the pattern contains two black corners along one side and two white corners on the opposite side. This criteria enables the app to determine the orientation of the pattern. The calibrator assigns the longer side to be the *x*-direction.

To prepare the checkerboard pattern:

1. Attach the checkerboard printout to a flat surface. Imperfections on the surface can affect the accuracy of the calibration.
2. Measure one side of the checkerboard square. You need this measurement for calibration. The size of the squares can vary depending on printer settings.



3. To improve the detection speed, set up the pattern with as little background clutter as possible.

Camera Setup

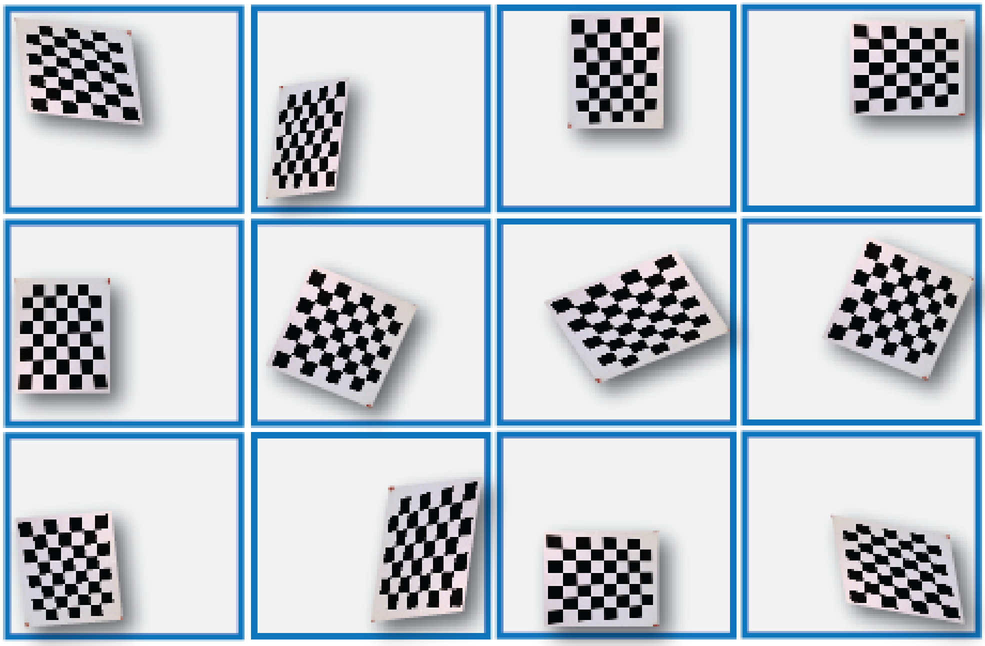
To properly calibrate your camera, follow these rules:

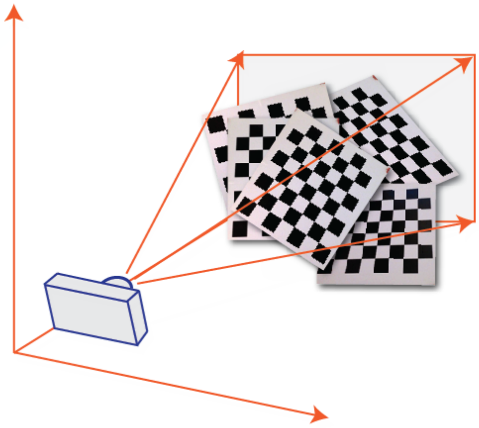
- Keep the pattern in focus, but do not use autofocus.
- Do not change zoom settings between images. Otherwise the focal length changes.

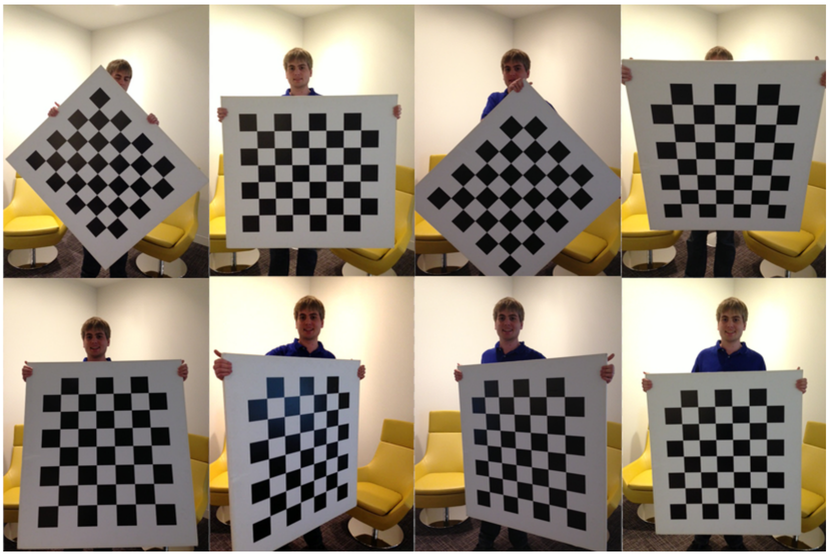
Capture Images

For best results, use at least 10 to 20 images of the calibration pattern. The calibrator requires at least three images. Use uncompressed images or images in lossless compression formats such as PNG. For greater calibration accuracy:

- Capture the images of the pattern at a distance roughly equal to the distance from your camera to the objects of interest. For example, if you plan to measure objects from 2 meters, keep your pattern approximately 2 meters from the camera.
- Place the checkerboard at an angle less than 45 degrees relative to the camera plane.
- Do not modify the images. For example, do not crop them.
- Do not use autofocus or change the zoom between images.
- Capture the images of a checkerboard pattern at different orientations relative to the camera.
- Capture enough different images of the pattern so that you have covered as much of the image frame as possible. Lens distortion increases radially from the center of the image and sometimes is not uniform across the image frame. To capture this lens distortion, the pattern must appear close to the edges.







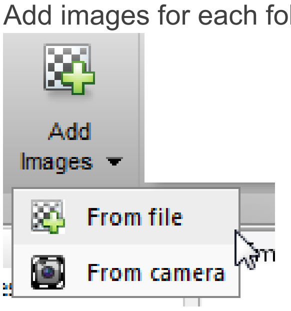
The Calibrator works with a range of checkerboard square sizes. As a general rule, your checkerboard should fill at least 20% of the captured image. For example, the preceding images were taken with a checkerboard square size of 108 mm.

Add Images

To begin calibration, you must add images. You can add saved images from a folder or add images directly from a camera. The calibrator analyze the images to ensure they meet the calibrator requirements and then detects the points.

Add Images from File

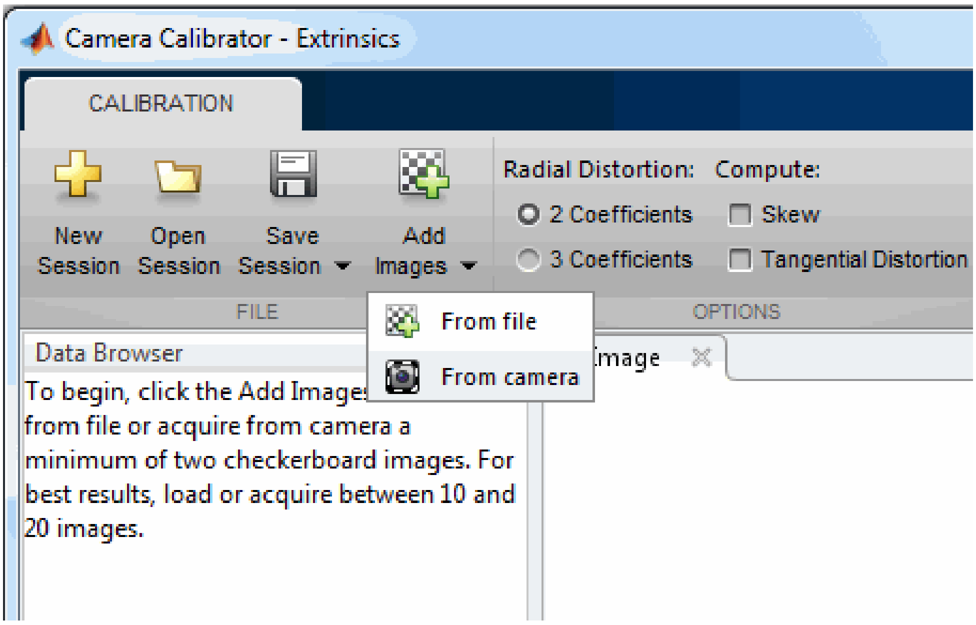
Click the Add images button, and select `From file`. You can add images from multiple folders by clicking Add images for each folder.



Acquire Live Images

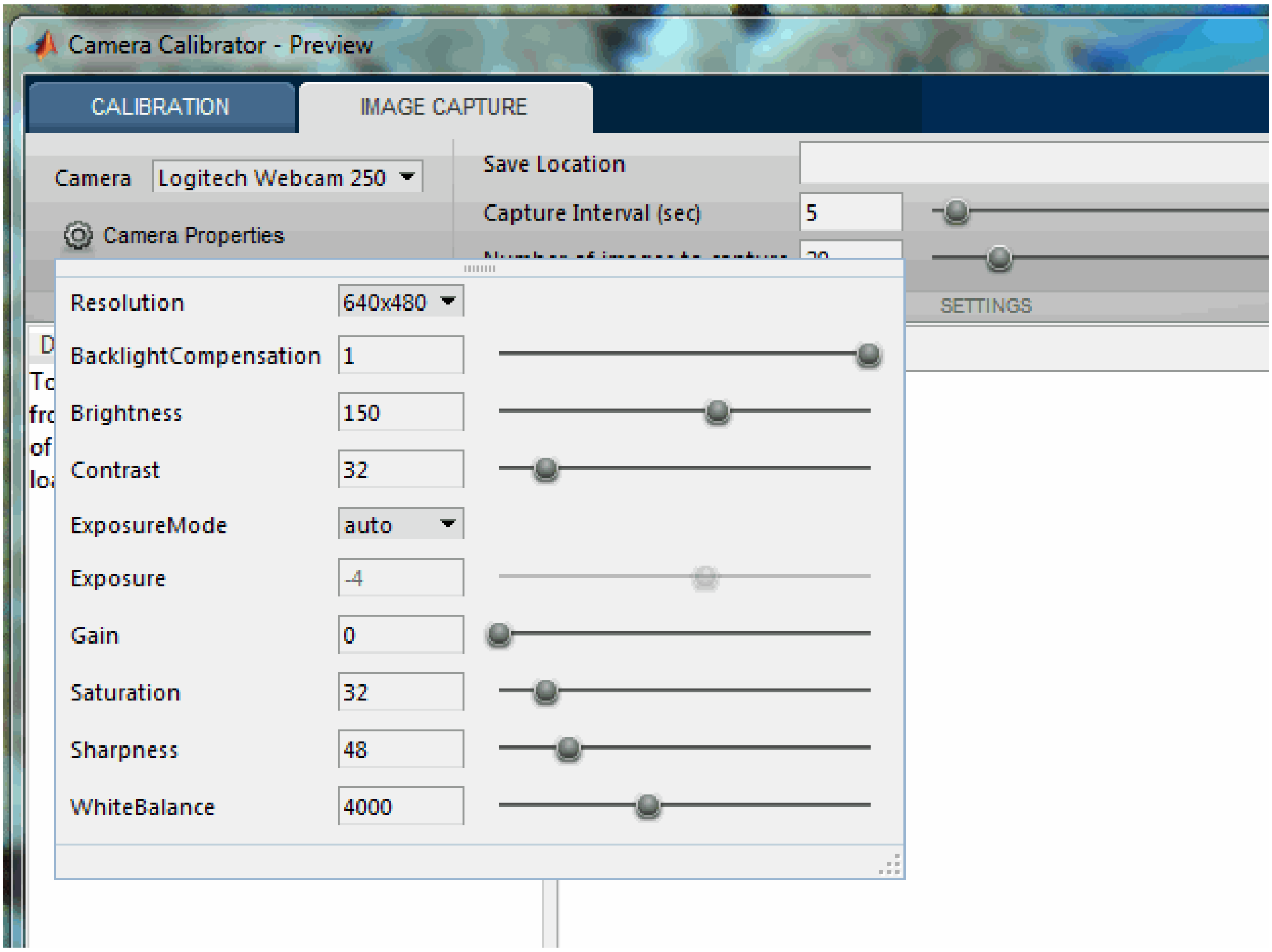
To begin calibration, you must add images. You can acquire images live from a Webcam using the MATLAB Webcam support. You must have the MATLAB Support Package for USB Webcams installed to use this feature. See Install the Webcam Support Package for information on installing the support package.

1. To add live images, click the **Add Images** arrow and select **From camera**.



2. The **Image Capture** tab opens. If you have only one Webcam connected to your system, it is selected by default and a live preview window opens. If you have multiple cameras connected and want to use a different one, select the camera in the **Camera** list.

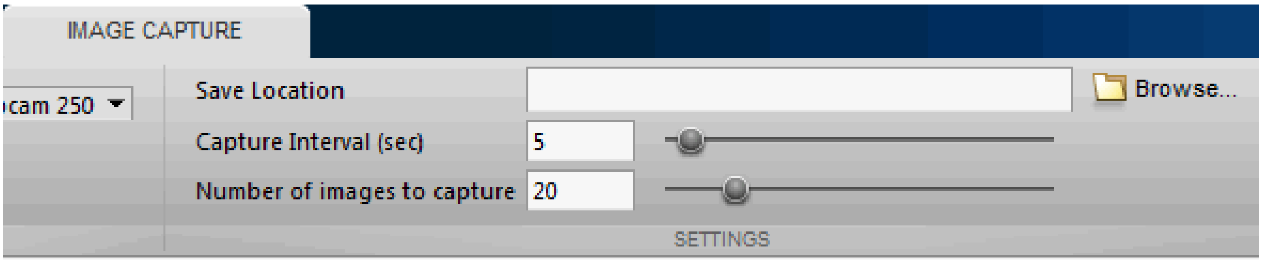
3. You can set properties for the camera to control the image. Click the **Camera Properties** field to open a list of your camera's properties. This list varies, depending on your device.



Use the sliders or drop-downs to change any available property settings. The **Preview** window updates dynamically when you change a setting. When you are done setting properties, click outside of the box to dismiss the properties list.

4. Enter a location for the acquired image files in the **Save Location** field by typing the path to the folder or using the **Browse** button. You must have permission to write to the folder you select.

5. Set your capture parameters.

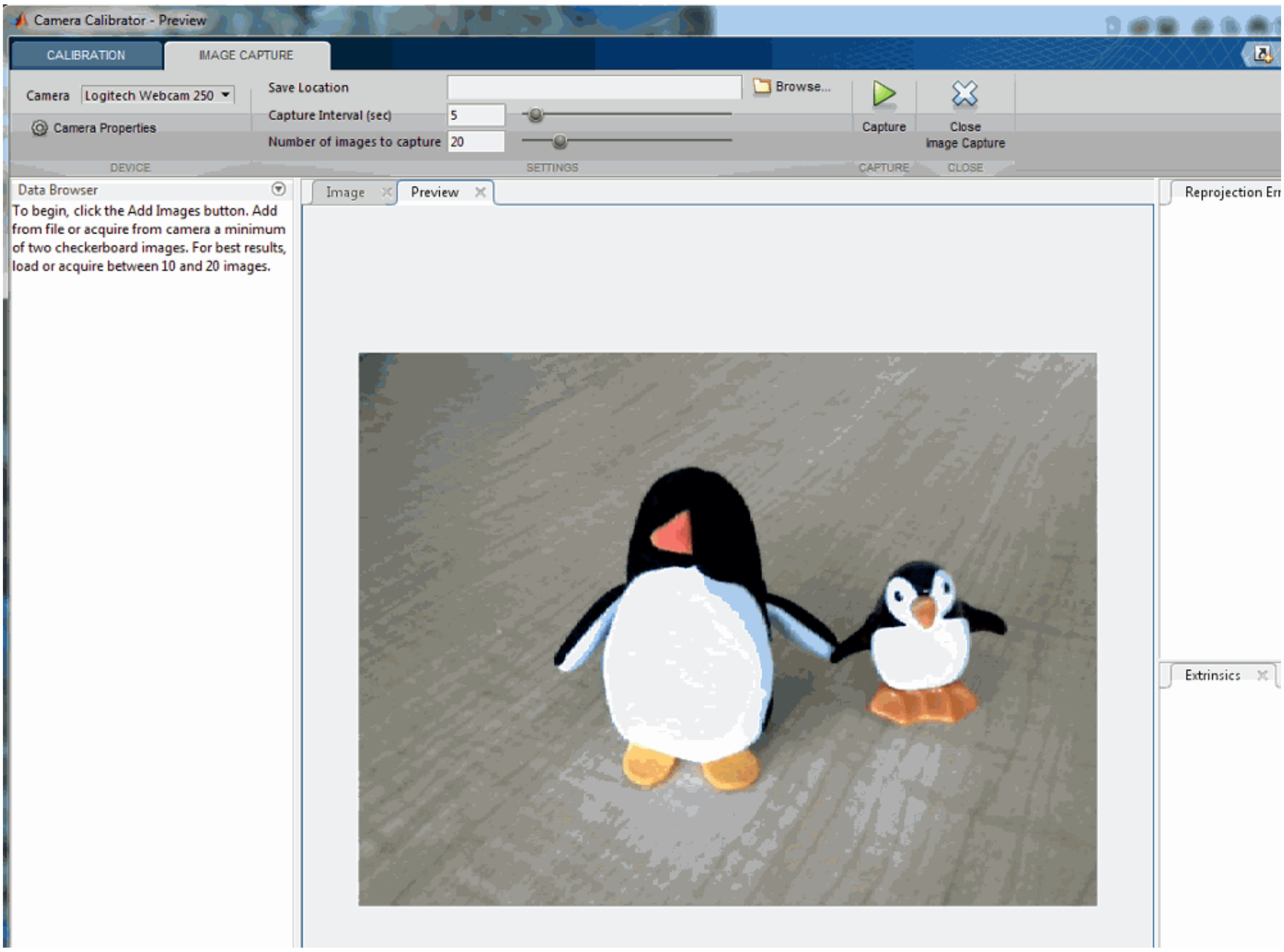


In the **Capture Interval** field, use the text box or slider to set the number of seconds between image captures. The default is 5 seconds, the minimum is 1 second, and the maximum is 60 seconds.

In the **Number of images to capture** field, use the text box or slider to set the number of image captures. The default is 20 images, the minimum is 2 images, and the maximum is 100 images.

In the default configuration, a total of 20 images are captured, one every 5 seconds.

6. It is helpful to dock the **Preview** window in the center of the tool. Move it from the right panel into the middle panel by dragging the banner. It then docks in the middle as shown here.

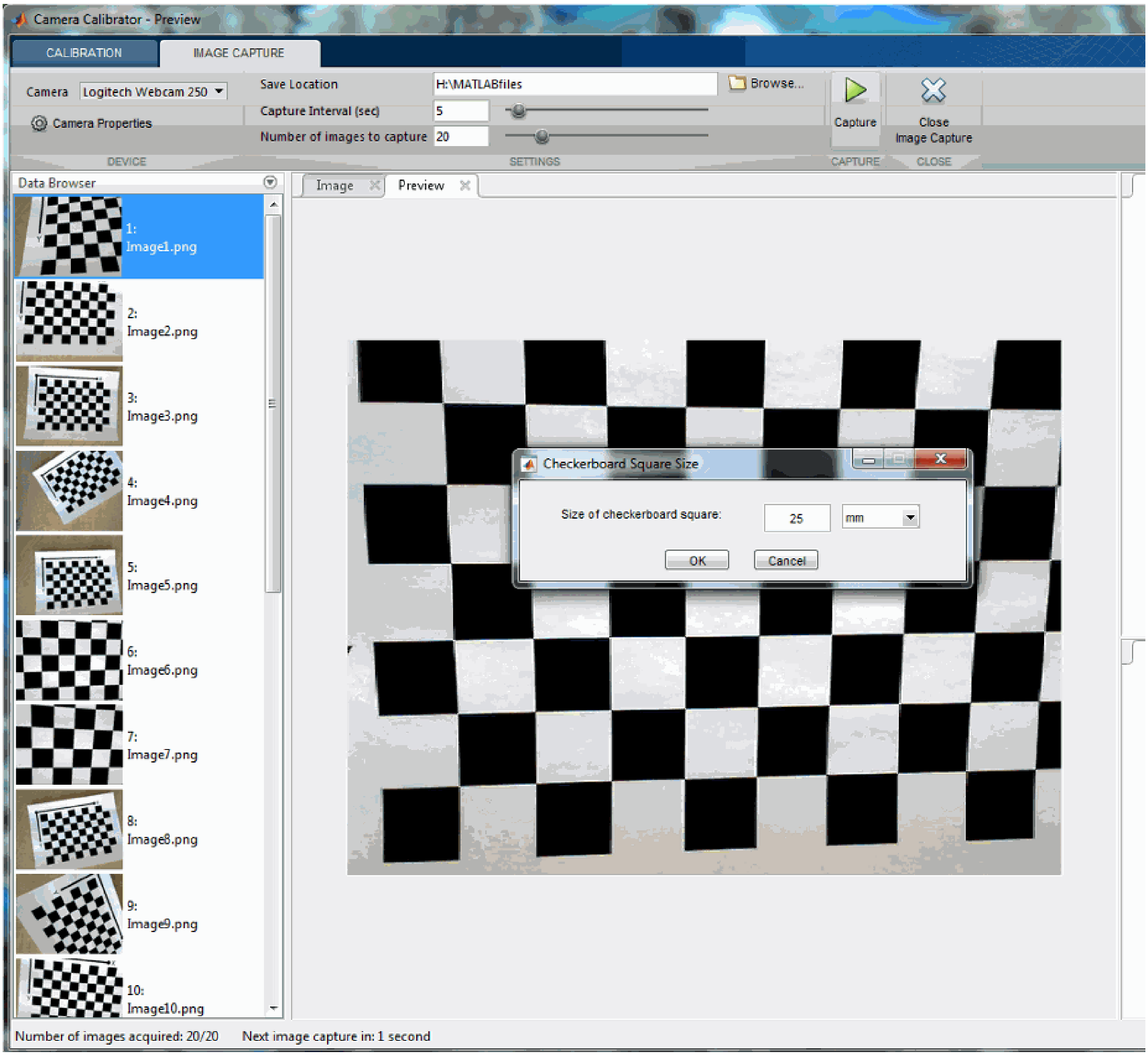


7. The **Preview** window shows the live images streamed as RGB data. After you adjust any device properties and capture settings, use the **Preview** window as a guide to line up the camera to acquire the checkerboard pattern image you wish to capture.

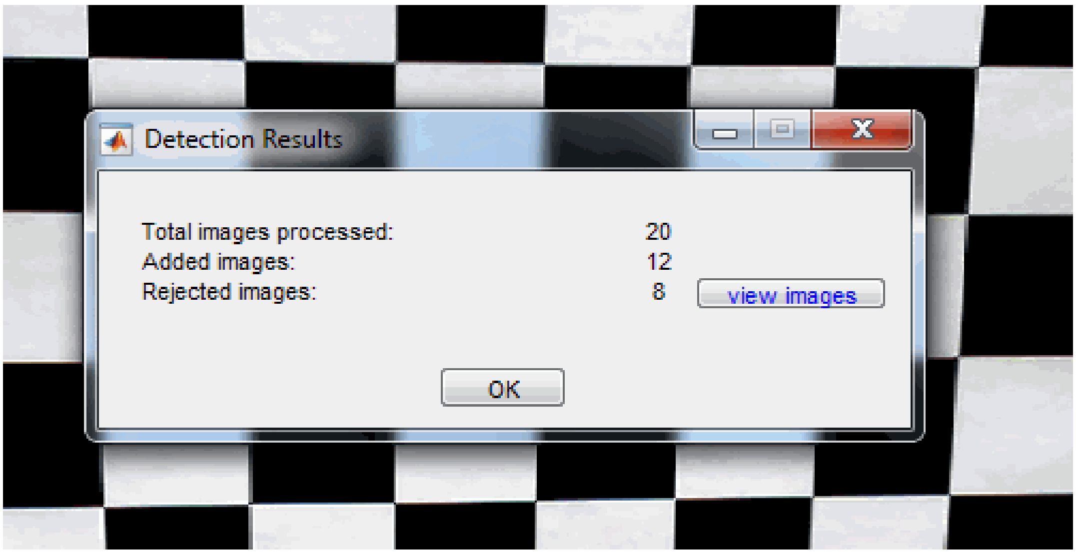
8. Click the **Capture** button. The number of images you set are captured and the thumbnails of the snapshots appear in the **Data Browser** panel. They are automatically named incrementally and are captured as `.png` files.

You can optionally stop the image capture before the designated number of images are captured by clicking **Stop Capture**.

When you are capturing images of a checkerboard, after the designated number of images are captured, a message displays with the size of the checkerboard square. Click **OK**.



The Detection Results are then calculated and displayed. For example:

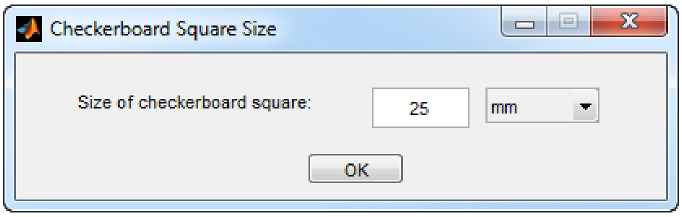


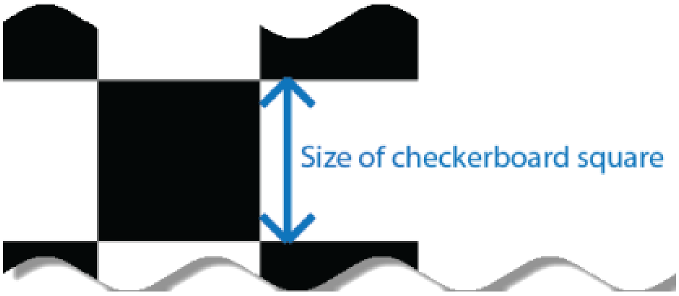
9. Click **OK** in the **Detection Results** dialog box.

10. When you have finished acquiring live images, you can click **Close Image Capture** to close the **Image Capture** tab and return to the **Calibration** tab.

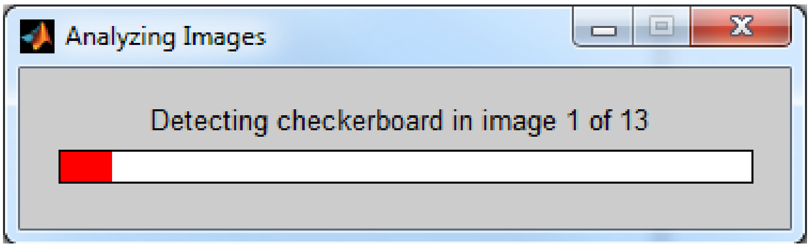
Analyze Images

After you select the images, in the Checkerboard Square Size dialog box, enter the length of one side of a square from the checkerboard pattern.

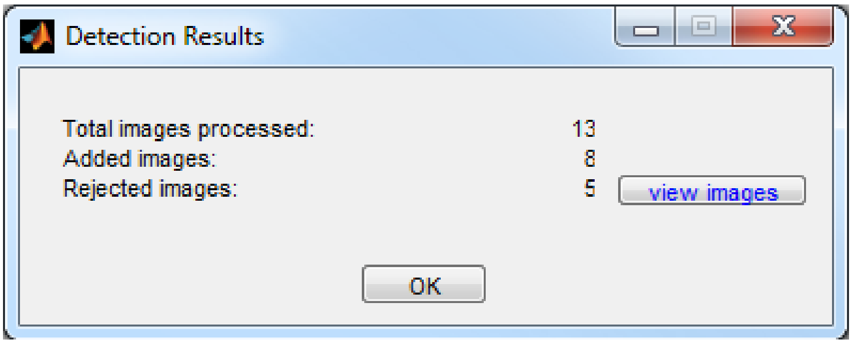




The calibrator attempts to detect a checkerboard in each of the added images. An Analyzing Images progress bar window appears, indicating detection progress.



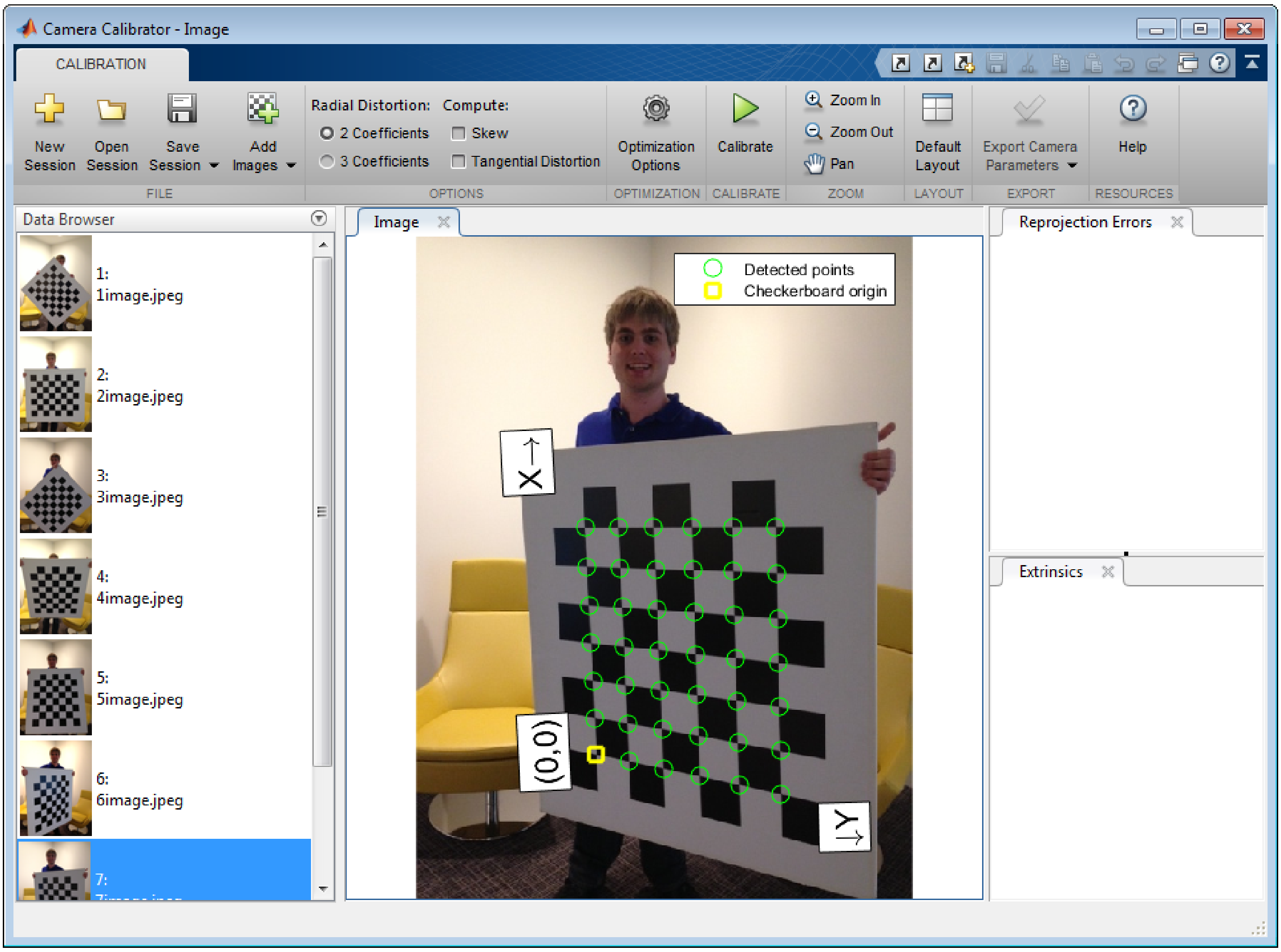
If any of the images are rejected, the Detection Results window appears, which contains diagnostic information. The results indicate how many total images were processed, and how many were accepted, rejected, or skipped The calibrator skips duplicate images.



To view the rejected images, click **view images**. The calibrator rejects duplicate images. It also rejects images where the entire checkerboard could not be detected. Possible reasons for no detection are a blurry image or an extreme angle of the pattern. Detection takes longer with larger images and with patterns that contain a large number of squares.

View Images and Detected Points

The Data Browser pane displays a list of images with IDs. These images contain a detected pattern. To view an image, select it from the **Data Browser** pane.



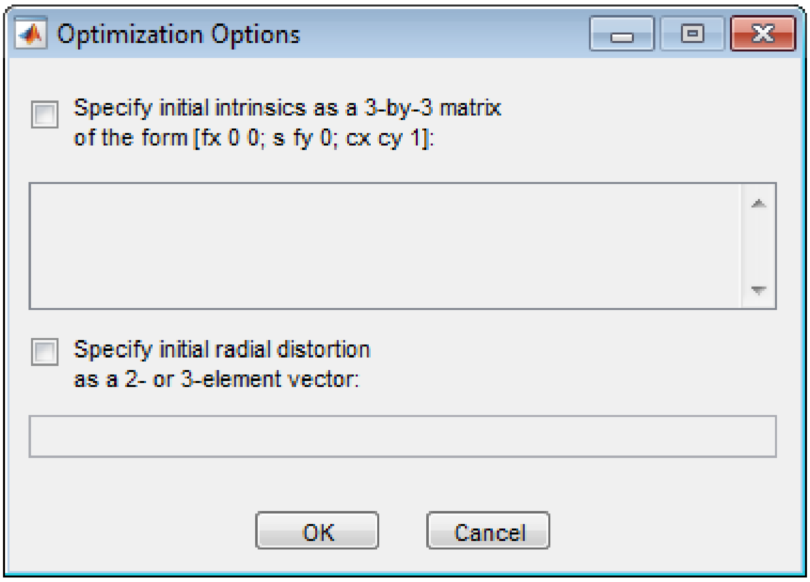
The **Image** pane displays the checkerboard image with green circles to indicate detected points. You can verify the corners were detected correctly using the zoom controls on the **View** tab. The yellow square indicates the (0,0) origin. The X and Y arrows indicate the checkerboard axes orientation.

Calibrate

Once you are satisfied with the accepted images, click Calibrate. The default calibration settings assume the minimum set of camera parameters. Start by running the calibration with the default settings. After evaluating the results, you can try to improve calibration accuracy by adjusting the settings and adding or removing images, and then calibrate again.

Set Initial Guesses for Camera Intrinsic and Radial Distortion

When there is severe lens distortion, the app can fail to compute the initial values for the camera intrinsics. If you have manufacturer's specifications for your camera and you know the pixel size, focal length, and/or lens characteristics, you can set the initial guesses for camera intrinsics and/or radial distortion manually. To set the initial guesses click the **Optimization Options** button.



- Select the top checkbox and then enter a 3-by-3 matrix to specify initial intrinsics. If you do not specify an initial guess, the function computes the initial intrinsic matrix using linear least squares.
- Select the bottom checkbox and then enter a 2- or 3-element vector to specify the initial radial distortion. If you do not provide a value, the function uses `0` as the initial value for all the coefficients.

Calibration Algorithm

The calibration algorithm assumes a pinhole camera model:

$$w\begin{bmatrix} x & y & 1 \end{bmatrix} = \begin{bmatrix} X & Y & Z & 1 \end{bmatrix}\begin{bmatrix} R \\ t \end{bmatrix}K$$

.

- (X,Y,Z): world coordinates of a point
- (x,y): image coordinates of the corresponding image point in pixels
- w: arbitrary homogeneous coordinates scale factor
- K: camera intrinsic matrix, defined as:

$$\begin{bmatrix} f_x & 0 & 0 \\ s & f_y & 0 \\ c_x & c_y & 1 \end{bmatrix}$$

The coordinates [c_x c_y] represent the optical center (the principal point), in pixels. When the x and y axis are exactly perpendicular, the skew parameter, s, equals `0`.

$f_x = F^*s_x$
$f_y = F^*s_y$
F, is the focal length in world units, typically expressed in millimeters.
[s_x, s_y] are the number of pixels per world unit in the x and y direction respectively.
f_x and f_y are expressed in pixels.

- R: matrix representing the 3-D rotation of the camera
- t: translation of the camera relative to the world coordinate system

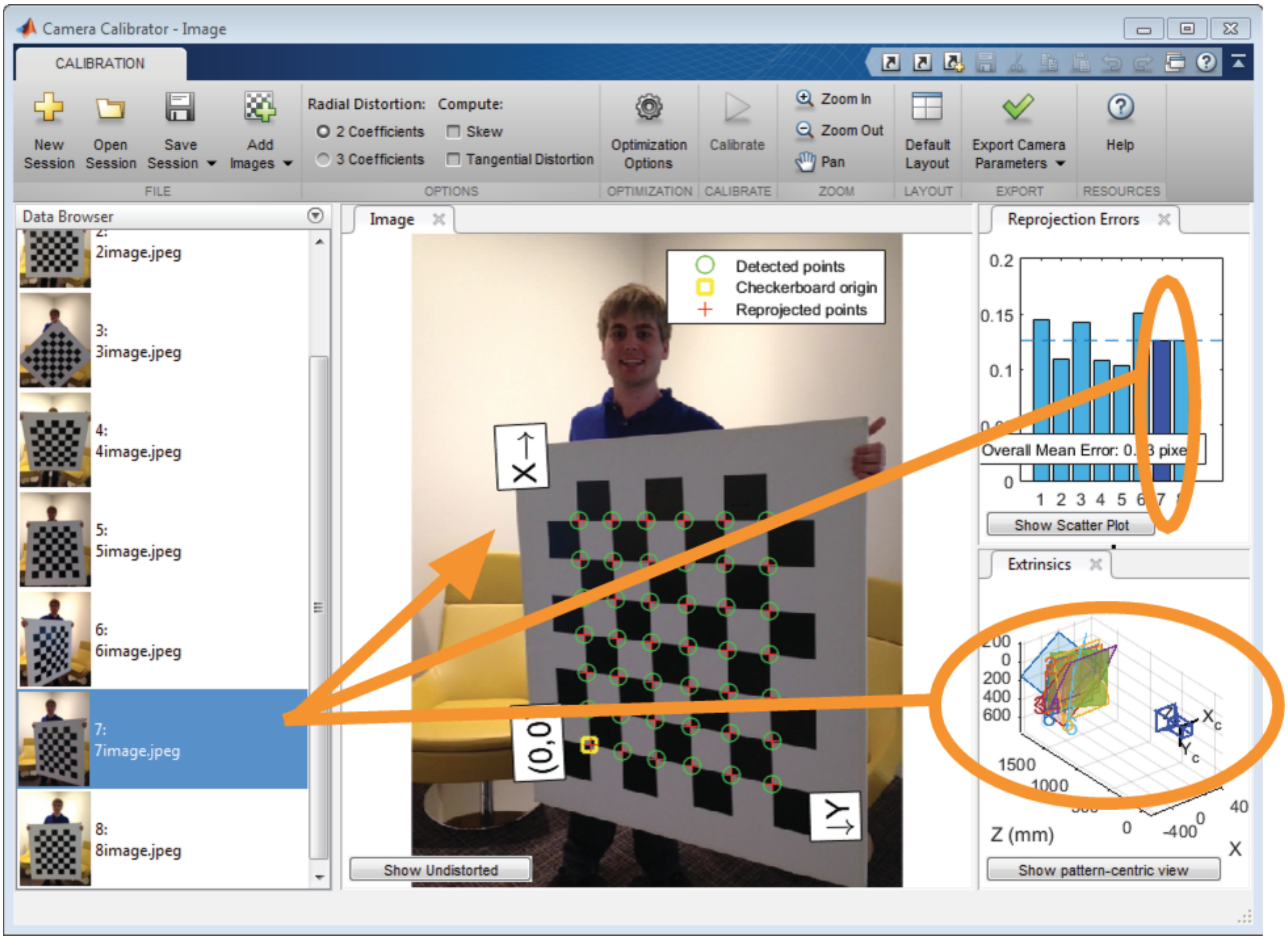
The camera calibration algorithm estimates the values of the intrinsic parameters, the extrinsic parameters, and the distortion coefficients. Camera calibration involves these steps:

1. Solve for the intrinsics and extrinsics in closed form, assuming that lens distortion is zero. [1]

2. Estimate all parameters simultaneously, including the distortion coefficients, using nonlinear least-squares minimization (Levenberg–Marquardt algorithm). Use the closed-form solution from the preceding step as the initial estimate of the intrinsics and extrinsics. Set the initial estimate of the distortion coefficients to zero. [1][2]

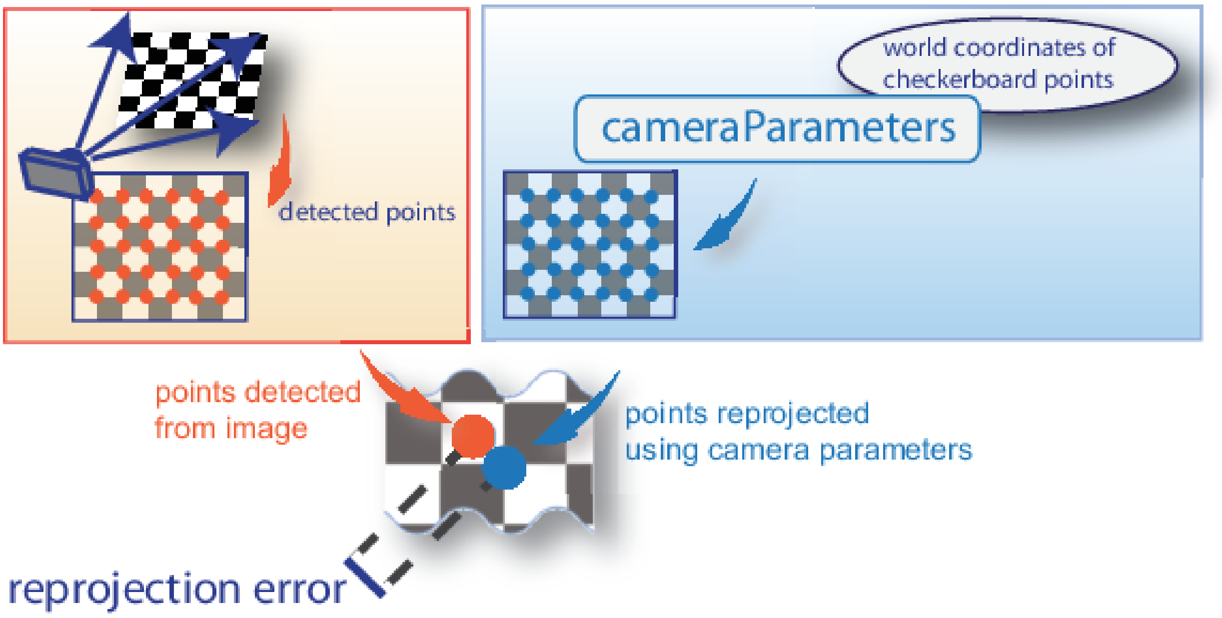
Evaluate Calibration Results

You can evaluate calibration accuracy by examining the reprojection errors and the camera extrinsics, and by viewing the undistorted image. For best calibration results, use all three methods of evaluation.



Examine Reprojection Errors

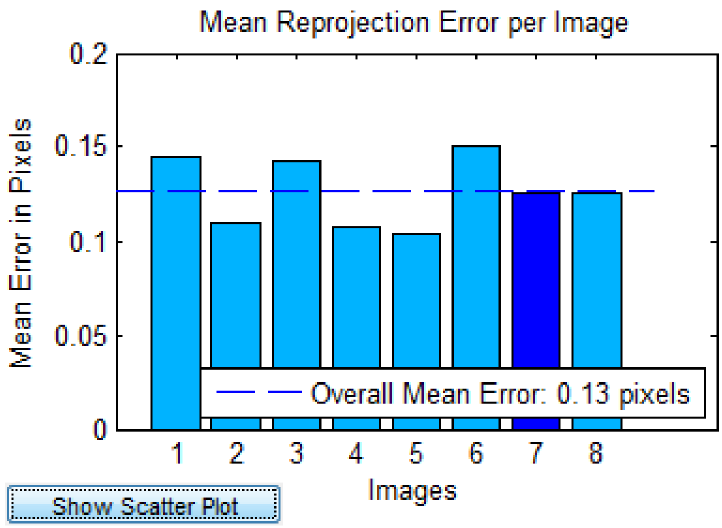
The *reprojection errors* are the distances in pixels between the detected and the reprojected points. The Camera Calibrator app calculates reprojection errors by projecting the checkerboard points from world coordinates, defined by the checkerboard, into image coordinates. The app then compares the reprojected points to the corresponding detected points. As a general rule, reprojection errors of less than one pixel are acceptable.



The Camera Calibrator app displays, in pixels, the reprojection errors as a bar graph and as a scatter plot. You can toggle between them using the button on the display. You can identify the images that adversely contribute to the calibration from either one of the graphs. You can then select and remove those images from the list in the **Data Browser** pane.

Bar Graph

The bar graph displays the mean reprojection error per image, along with the overall mean error. The bar labels correspond to the image IDs. The highlighted bar corresponds to the selected image.

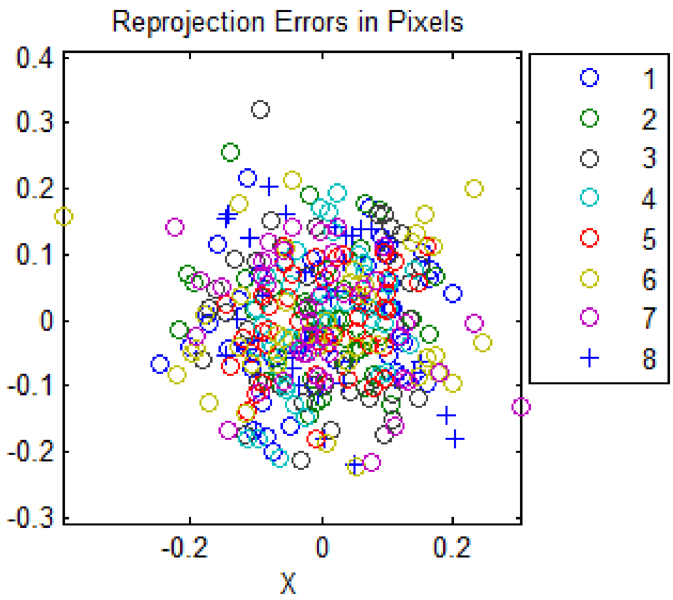


Select an image in one of these ways:

- Clicking the corresponding bar in the graph.
- Select the image from the list in the **Data Browser** pane.

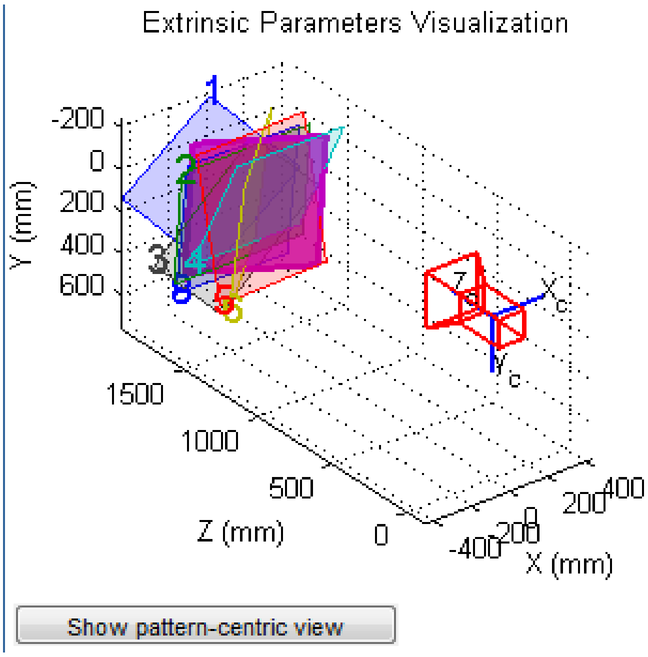
Scatter Plot

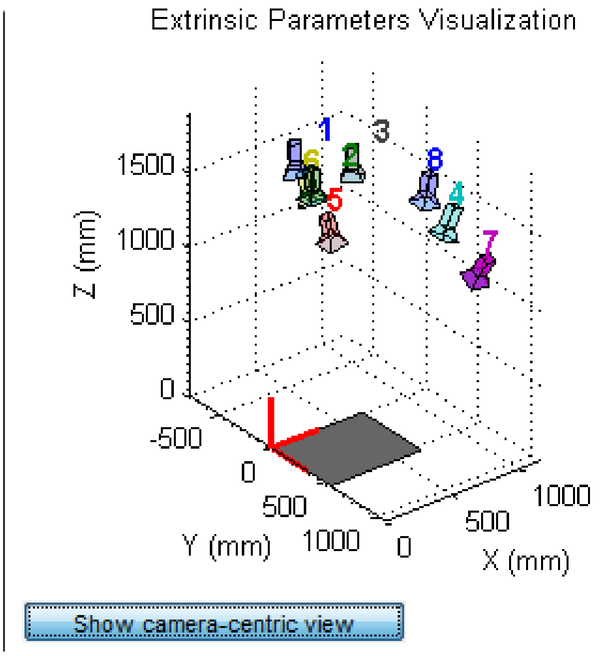
The scatter plot displays the reprojection errors for each point. The plus markers correspond to the points in the selected image. An accurate calibration typically results in a compact cloud of points. Outliers indicate potential issues with the corresponding images. To improve accuracy, consider removing those images.



Examine Extrinsic Parameter Visualization

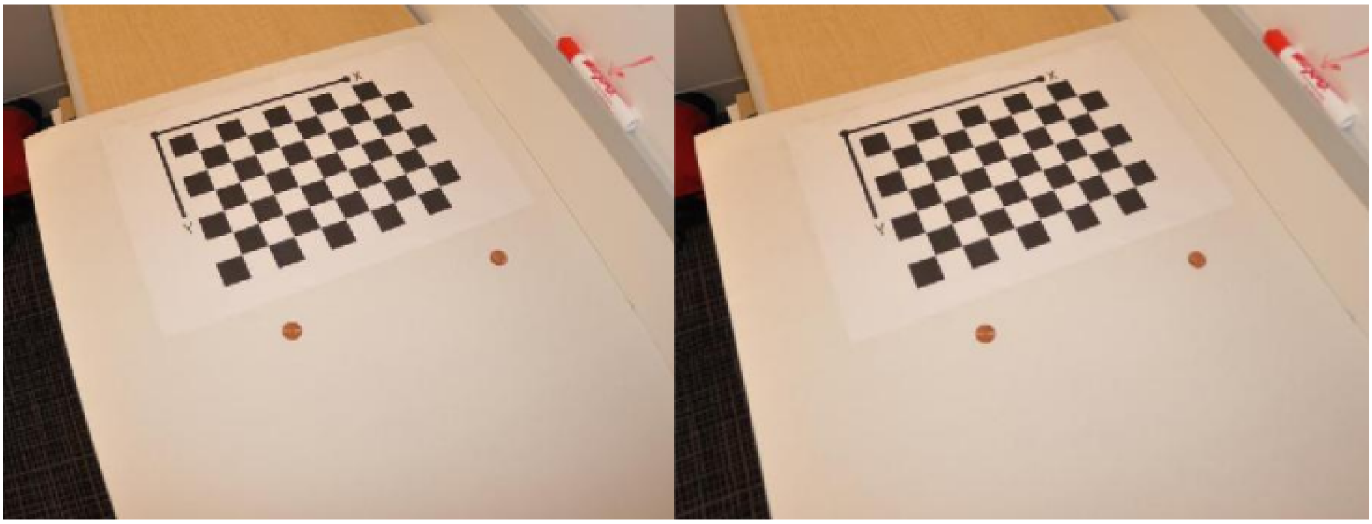
The 3-D extrinsic parameters plot provides a camera-centric view of the patterns and a pattern-centric view of the camera. The camera-centric view is helpful if the camera was stationary when the images were captured. The pattern-centric view is helpful if the pattern was stationary. Click the button on the display to toggle between the two visuals. Click and drag a graph to rotate it. Click a checkerboard or a camera to select it. The highlighted data in the visualizations correspond to the selected image in the list. Examine the relative positions of the pattern and the camera to see if they match what you expect. For example, a pattern that appears behind the camera indicates a calibration error.





View Undistorted Image

To view the effects of removing lens distortion, in the **Image** pane, click the **Show Undistorted**. If the calibration was accurate, the distorted lines in the image become straight.



It is important to check the undistorted images even if the reprojection errors are low. If the pattern covers only a small percentage of the image, the distortion estimation might be incorrect, even though the calibration resulted in few reprojection errors.

Improve Calibration

To improve the calibration, you can remove high-error images, add more images, or modify the calibrator settings.

Add or Remove Images

Consider adding more images if:

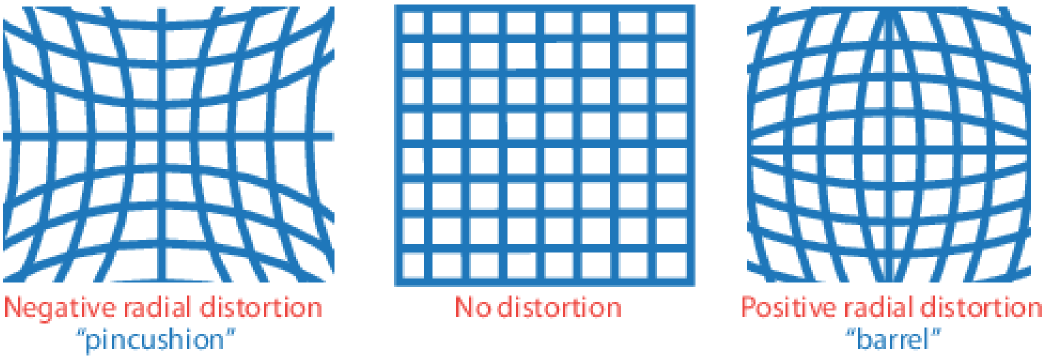
- You have less than 10 images.
- The patterns do not cover enough of the image frame.
- The patterns do not have enough variation in orientation with respect to the camera.

Consider removing images if the images:

- Have a high mean reprojection error
- Are blurry
- Contain a checkerboard at an angle greater than 45 degrees relative to the camera plane
- Contain incorrectly detected checkerboard points

Change the Number of Radial Distortion Coefficients

You can specify 2 or 3 radial distortion coefficients by selecting the corresponding radio button from the **Options** section. *Radial distortion* occurs when light rays bend more near the edges of a lens than they do at its optical center. The smaller the lens, the greater the distortion.



The radial distortion coefficients model this type of distortion. The distorted points are denoted as ($x_{distorted}$, $y_{distorted}$):

$$x_{distorted} = x(1 + k_1*r^2 + k_2*r^4 + k_3*r^6)$$

$$y_{distorted} = y(1 + k_1*r^2 + k_2*r^4 + k_3*r^6)$$

- x, y — Undistorted pixel locations. x and y are in normalized image coordinates. Normalized image coordinates are calculated from pixel coordinates by translating to the optical center and dividing by the focal length in pixels. Thus, x and y are dimensionless.
- k_1, k_2, and k_3 — Radial distortion coefficients of the lens.
- r^2: $x^2 + y^2$

Typically, two coefficients are sufficient for calibration. For severe distortion, such as in wide-angle lenses, you can select 3 coefficients to include k_3.

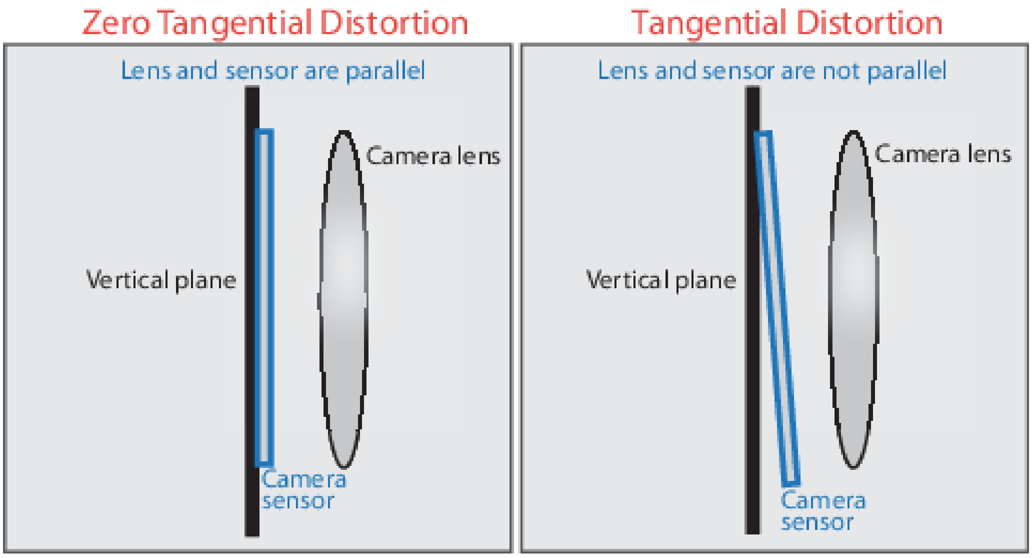
The undistorted pixel locations are in normalized image coordinates, with the origin at the optical center. The coordinates are expressed in world units.

Compute Skew

When you select the **Compute Skew** check box, the calibrator estimates the image axes skew. Some camera sensors contain imperfections that cause the x- and y-axis of the image to not be perpendicular. You can model this defect using a skew parameter. If you do not select the check box, the image axes are assumed to be perpendicular, which is the case for most modern cameras.

Compute Tangential Distortion

Tangential distortion occurs when the lens and the image plane are not parallel. The tangential distortion coefficients model this type of distortion.



The distorted points are denoted as ($x_{distorted}$, $y_{distorted}$):

$$x_{distorted} = x + [2 * p_1 * x * y + p_2 * (r^2 + 2 * x^2)]$$

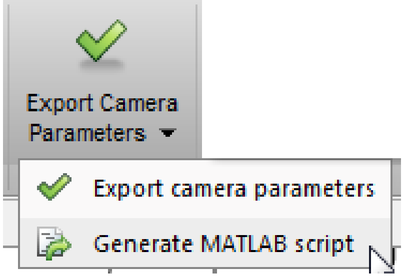
$$y_{distorted} = y + [p_1 * (r^2 + 2 *y^2) + 2 * p_2 * x * y]$$

- x, y — Undistorted pixel locations. x and y are in normalized image coordinates. Normalized image coordinates are calculated from pixel coordinates by translating to the optical center and dividing by the focal length in pixels. Thus, x and y are dimensionless.
- p_1 and p_2 — Tangential distortion coefficients of the lens.
- $r^2 = x^2 + y^2$

When you select the **Compute Tangential Distortion** check box, the calibrator estimates the tangential distortion coefficients. Otherwise, the calibrator sets the tangential distortion coefficients to zero.

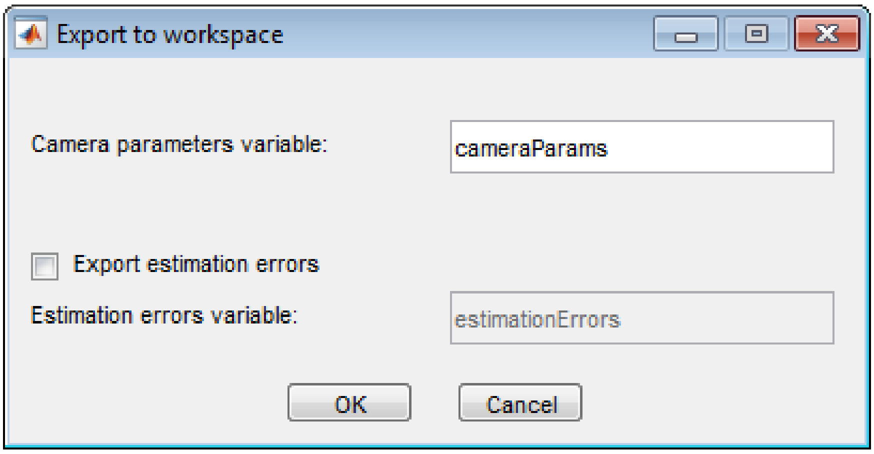
Export Camera Parameters

When you are satisfied with calibration accuracy, click **Export Camera Parameters**. You can save and export the camera parameters to an object or generate the camera parameters as a MATLAB script.



Export Camera Parameters

Click **Export Camera Parameters** to create a `cameraParameters` object in your workspace. The object contains the intrinsic and extrinsic parameters of the camera, and the distortion coefficients. You can use this object for various computer vision tasks, such as image undistortion, measuring planar objects, and 3-D reconstruction. See Measuring Planar Objects with a Calibrated Camera. You can optionally export the `cameraCalibrationErrors` object, which contains the standard errors of estimated camera parameters.



Generate MATLAB Script

Click **Generate MATLAB script** to save your camera parameters to a MATLAB script, enabling you to reproduce the steps from your calibration session.

References

[1] Zhang, Z. "A Flexible New Technique for Camera Calibration". *IEEE Transactions on Pattern Analysis and Machine Intelligence*.Vol. 22, No. 11, 2000, pp. 1330–1334.

[2] Heikkila, J, and O. Silven. "A Four-step Camera Calibration Procedure with Implicit Image Correction." *IEEE International Conference on Computer Vision and Pattern Recognition*. 1997.

See Also

Camera Calibrator | `cameraParameters` | `detectCheckerboardPoints` | `estimateCameraParameters` | `generateCheckerboardPoints` | `showExtrinsics` | `showReprojectionErrors` | Stereo Camera Calibrator | `stereoParameters` | `undistortImage`

Related Examples

- Evaluating the Accuracy of Single Camera Calibration
- Measuring Planar Objects with a Calibrated Camera
- Stereo Calibration and Scene Reconstruction
- Depth Estimation From Stereo Video
- 3-D Point Cloud Registration and Stitching
- Uncalibrated Stereo Image Rectification
- Checkerboard pattern

More About

- Stereo Calibration App

External Websites

- Camera Calibration with MATLAB