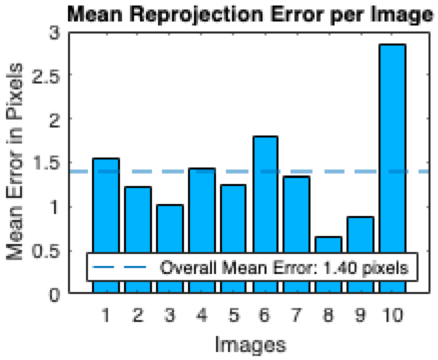
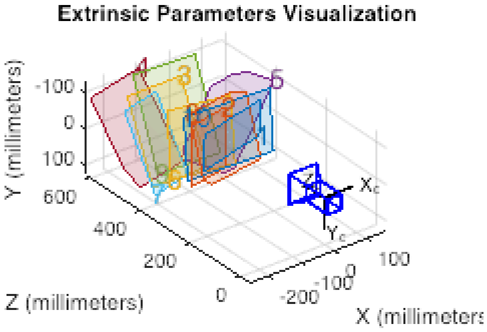
```
% Auto-generated by cameraCalibrator app on 23-Sep-2022
%-------------------------------------------------------



% Define images to process
imageFileNames = {'/Users/rakhi/Desktop/10.jpeg',...
    '/Users/rakhi/Desktop/9.jpeg',...
    '/Users/rakhi/Desktop/8.jpeg',...
    '/Users/rakhi/Desktop/7.jpeg',...
    '/Users/rakhi/Desktop/6.jpeg',...
    '/Users/rakhi/Desktop/5.jpeg',...
    '/Users/rakhi/Desktop/4.jpeg',...
    '/Users/rakhi/Desktop/3.jpeg',...
    '/Users/rakhi/Desktop/2.jpeg',...
    '/Users/rakhi/Desktop/1.jpeg',...
    };
% Detect calibration pattern in images
detector = vision.calibration.monocular.CheckerboardDetector();
[imagePoints, imagesUsed] = detectPatternPoints(detector, imageFileNames);
```

```
Warning: The checkerboard must be asymmetric: one side should be even, and the other should be odd.
Otherwise, the orientation of the board may be detected incorrectly.
```

```
imageFileNames = imageFileNames(imagesUsed);

% Read the first image to obtain image size
originalImage = imread(imageFileNames{1});
[mrows, ncols, ~] = size(originalImage);

% Generate world coordinates for the planar pattern keypoints
squareSize = 21;  % in units of 'millimeters'
worldPoints = generateWorldPoints(detector, 'SquareSize', squareSize);

% Calibrate the camera
[cameraParams, imagesUsed, estimationErrors] = estimateCameraParameters(imagePoints, wo
    'EstimateSkew', false, 'EstimateTangentialDistortion', false, ...
    'NumRadialDistortionCoefficients', 2, 'WorldUnits', 'millimeters', ...
    'InitialIntrinsicMatrix', [], 'InitialRadialDistortion', [], ...
    'ImageSize', [mrows, ncols]);

% View reprojection errors
h1=figure; showReprojectionErrors(cameraParams);
```



```
% Visualize pattern locations
h2=figure; showExtrinsics(cameraParams, 'CameraCentric');
```



```
% Display parameter estimation errors
displayErrors(estimationErrors, cameraParams);
```

```
            Standard Errors of Estimated Camera Parameters
            ----------------------------------------------

Intrinsics
----------
Focal length (pixels):   [ 1160.0469 +/- 11.1237    1158.5178 +/- 10.9550 ]
Principal point (pixels):[  547.6532 +/- 2.6145      358.7546 +/- 3.3394  ]
Radial distortion:       [    0.1792 +/- 0.0126       -0.3098 +/- 0.0468  ]

Extrinsics
----------
Rotation vectors:
                         [   -0.3525 +/- 0.0060       -0.1008 +/- 0.0084        3.0983 +/- 0.0010  ]
                         [   -0.1623 +/- 0.0070       -0.1534 +/- 0.0072        1.5691 +/- 0.0011  ]
                         [   -0.3932 +/- 0.0037        0.3318 +/- 0.0043        1.5475 +/- 0.0010  ]
                         [   -0.4915 +/- 0.0036        0.3237 +/- 0.0043        1.3969 +/- 0.0011  ]
                         [    0.0316 +/- 0.0037        0.1662 +/- 0.0049        2.1358 +/- 0.0006  ]
                         [   -0.2403 +/- 0.0035        0.1591 +/- 0.0029       -1.5996 +/- 0.0007  ]
                         [   -0.2123 +/- 0.0054       -0.2888 +/- 0.0055       -1.7329 +/- 0.0012  ]
                         [    2.3404 +/- 0.0027        2.0026 +/- 0.0024       -0.1374 +/- 0.0048  ]
                         [   -0.4842 +/- 0.0038       -0.1223 +/- 0.0031       -1.6000 +/- 0.0009  ]
                         [   -0.0997 +/- 0.0028        0.0473 +/- 0.0021        0.0052 +/- 0.0004  ]

Translation vectors (millimeters):
                         [   79.2641 +/- 0.7674       21.2392 +/- 1.0105      362.2074 +/- 3.2300  ]
                         [  -16.7311 +/- 0.8272      -87.6919 +/- 1.0368      371.5075 +/- 3.4795  ]
[   -5.9197 +/- 1.2663      -85.2636 +/- 1.6043      561.4882 +/- 4.8617  ]
[  -92.1246 +/- 1.4329     -111.5902 +/- 1.7641      617.0880 +/- 5.1640  ]
[  161.2940 +/- 0.9454      -80.4612 +/- 1.1688      412.1800 +/- 3.9543  ]
[ -148.6140 +/- 0.9031       87.6772 +/- 1.1632      413.2220 +/- 3.8292  ]
[ -179.6641 +/- 1.0427      130.4118 +/- 1.3010      436.2398 +/- 4.3920  ]
[ -113.3690 +/- 0.8488      -94.0202 +/- 1.0755      387.2548 +/- 3.6806  ]
[ -121.2970 +/- 1.1189      121.8683 +/- 1.4845      502.7400 +/- 4.8967  ]
[ -128.8892 +/- 0.8390      -89.6181 +/- 1.0825      389.3595 +/- 3.7033  ]
```

```
% For example, you can use the calibration data to remove effects of lens distortion.
undistortedImage = undistortImage(originalImage, cameraParams);

% See additional examples of how to use the calibration data.  At the prompt type:
% showdemo('MeasuringPlanarObjectsExample')
% showdemo('StructureFromMotionExample')
```