

Camera Model in Matlab before and after 2022b

Matlab used to have the camera model in a transposed way (before Matlab 2022b).

$$\lambda * [x_{cam} \ y_{cam} \ 1] = [x_w \ y_w \ z_w \ 1] * \begin{bmatrix} R^T \\ T^T \end{bmatrix} * K^T$$

Matlab (after version 2022b) finally adopted the standard model, which the rest of the vision community used:

$$\lambda * \begin{bmatrix} x_{cam} \\ y_{cam} \\ 1 \end{bmatrix} = K * [R \ T] * \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

Let's do a camera calibration with version 2020b and 2022b. Both sessions of calibration are similar, but the cameraParams differ totally. For instance, in cameraParams2020b, we get:

```
cameraParams2020b =  
  
cameraParameters with properties:  
  
Camera Intrinsic  
    Intrinsic: [1x1 cameraIntrinsic]  
  
Camera Extrinsic  
    RotationMatrices: [3x3x15 double]  
    TranslationVectors: [15x3 double]  
  
Accuracy of Estimation  
    MeanReprojectionError: 0.3344  
    ReprojectionErrors: [182x2x15 double]  
    ReprojectedPoints: [182x2x15 double]  
  
Calibration Settings  
    NumPatterns: 15  
    WorldPoints: [182x2 double]  
    WorldUnits: 'millimeters'  
    EstimateSkew: 0  
    NumRadialDistortionCoefficients: 2  
    EstimateTangentialDistortion: 0
```

Whereas in cameraParams2022b we get:

```
cameraParams2022b =  
  
cameraParameters with properties:  
  
Camera Intrinsic  
    Intrinsic: [1x1 cameraIntrinsic]  
  
Camera Extrinsic  
    PatternExtrinsic: [15x1 rigidform3d]  
  
Accuracy of Estimation  
    MeanReprojectionError: 0.3510  
    ReprojectionErrors: [182x2x15 double]  
    ReprojectedPoints: [182x2x15 double]  
  
Calibration Settings  
    NumPatterns: 15  
    DetectedKeypoints: [182x15 logical]  
    WorldPoints: [182x2 double]  
    WorldUnits: 'millimeters'  
    EstimateSkew: 0  
    NumRadialDistortionCoefficients: 2  
    EstimateTangentialDistortion: 0
```

We see that in the all version the extrinsic parameters are RotationMatrices and TranslationVectors, whereas in the new version, the extrinsic is an array of rigidform3d.

If we see the values of the cameraParams, we realize that the intrinsic matrix in 2022b has the standard form:

```
>> cameraParams2020b.IntrinsicMatrix
ans =

1.0e+03 *

    2.4441         0         0
         0    2.4442         0
    0.7466    0.5774    0.0010

>> cameraParams2022b.K
ans =

1.0e+03 *

    2.4449         0    0.7464
         0    2.4450    0.5781
         0         0    0.0010

>> cameraParams2020b.RadialDistortion
ans =

-0.1462    0.2314

>> cameraParams2022b.RadialDistortion
ans =

-0.1457    0.2185
```

The extrinsic rotations (that were previously transposed), have now the standard form:

```
>> cameraParams2020b.RotationMatrices(:, :, 1)
ans =

-0.9994    0.0155    0.0306
-0.0333   -0.6478   -0.7611
 0.0081   -0.7617    0.6479

>> cameraParams2022b.PatternExtrinsics(1).R
ans =

-0.9994   -0.0334    0.0081
 0.0155   -0.6475   -0.7619
 0.0307   -0.7613    0.6477
```

The translations have a similar form:

```
>> cameraParams2020b.TranslationVectors(1, :)
ans =

69.3705    59.5601   388.7291

>> cameraParams2022b.PatternExtrinsics(1).Translation
ans =

69.4025    59.4428   388.8578
```

Finally, there is a matrix A in PatternExtrinsics that corresponds to $[R, T; 0 \ 0 \ 0 \ 1]$:

```
>> cameraParams2022b.PatternExtrinsics(1).A
ans =

-0.9994   -0.0334    0.0081   69.4025
 0.0155   -0.6475   -0.7619   59.4428
 0.0307   -0.7613    0.6477   388.8578
         0         0         0         1.0000

>> [cameraParams2022b.PatternExtrinsics(1).R, cameraParams2022b.PatternExtrinsics(1).Translation']
ans =

-0.9994   -0.0334    0.0081   69.4025
 0.0155   -0.6475   -0.7619   59.4428
 0.0307   -0.7613    0.6477   388.8578
```