## <https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga7a6c4e032c97f03ba747966e6ad862b1>

## 

Returns the new camera intrinsic matrix based on the free scaling parameter.

Parameters

|  |  |
| --- | --- |
| cameraMatrix | Input camera intrinsic matrix. |
| distCoeffs | Input vector of distortion coefficients (*k*1,*k*2,*p*1,*p*2[,*k*3[,*k*4,*k*5,*k*6[,*s*1,*s*2,*s*3,*s*4[,*τx*,*τy*]]]]) of 4, 5, 8, 12 or 14 elements. If the vector is NULL/empty, the zero distortion coefficients are assumed. |
| imageSize | Original image size. |
| alpha | Free scaling parameter between 0 (when all the pixels in the undistorted image are valid) and 1 (when all the source image pixels are retained in the undistorted image). See stereoRectify for details. |
| newImgSize | Image size after rectification. By default, it is set to imageSize . |
| validPixROI | Optional output rectangle that outlines all-good-pixels region in the undistorted image. See roi1, roi2 description in stereoRectify . |
| centerPrincipalPoint | Optional flag that indicates whether in the new camera intrinsic matrix the principal point should be at the image center or not. By default, the principal point is chosen to best fit a subset of the source image (determined by alpha) to the corrected image. |

Returns

new\_camera\_matrix Output new camera intrinsic matrix.

The function computes and returns the optimal new camera intrinsic matrix based on the free scaling parameter. By varying this parameter, you may retrieve only sensible pixels alpha=0 , keep all the original image pixels if there is valuable information in the corners alpha=1 , or get something in between. When alpha>0 , the undistorted result is likely to have some black pixels corresponding to "virtual" pixels outside of the captured distorted image. The original camera intrinsic matrix, distortion coefficients, the computed new camera intrinsic matrix, and newImageSize should be passed to initUndistortRectifyMap to produce the maps for remap .

## <https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga246253dcc6de2e0376c599e7d692303a>

## 

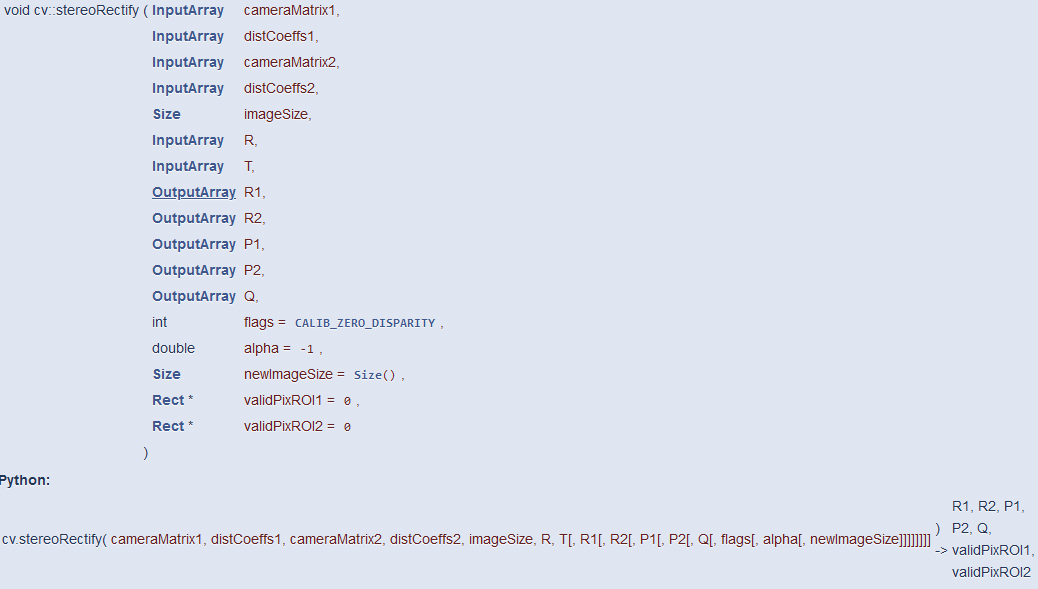
Calibrates a stereo camera set up. This function finds the intrinsic parameters for each of the two cameras and the extrinsic parameters between the two cameras.

Parameters

|  |  |
| --- | --- |
| objectPoints | Vector of vectors of the calibration pattern points. The same structure as in [calibrateCamera](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga3207604e4b1a1758aa66acb6ed5aa65d). For each pattern view, both cameras need to see the same object points. Therefore, objectPoints.size(), imagePoints1.size(), and imagePoints2.size() need to be equal as well as objectPoints[i].size(), imagePoints1[i].size(), and imagePoints2[i].size() need to be equal for each i. |
| imagePoints1 | Vector of vectors of the projections of the calibration pattern points, observed by the first camera. The same structure as in [calibrateCamera](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga3207604e4b1a1758aa66acb6ed5aa65d). |
| imagePoints2 | Vector of vectors of the projections of the calibration pattern points, observed by the second camera. The same structure as in [calibrateCamera](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga3207604e4b1a1758aa66acb6ed5aa65d). |
| cameraMatrix1 | Input/output camera intrinsic matrix for the first camera, the same as in [calibrateCamera](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga3207604e4b1a1758aa66acb6ed5aa65d). Furthermore, for the stereo case, additional flags may be used, see below. |
| distCoeffs1 | Input/output vector of distortion coefficients, the same as in [calibrateCamera](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga3207604e4b1a1758aa66acb6ed5aa65d). |
| cameraMatrix2 | Input/output second camera intrinsic matrix for the second camera. See description for cameraMatrix1. |
| distCoeffs2 | Input/output lens distortion coefficients for the second camera. See description for distCoeffs1. |
| imageSize | Size of the image used only to initialize the camera intrinsic matrices. |
| R | Output rotation matrix. Together with the translation vector T, this matrix brings points given in the first camera's coordinate system to points in the second camera's coordinate system. In more technical terms, the tuple of R and T performs a change of basis from the first camera's coordinate system to the second camera's coordinate system. Due to its duality, this tuple is equivalent to the position of the first camera with respect to the second camera coordinate system. |
| T | Output translation vector, see description above. |
| E | Output essential matrix. |
| F | Output fundamental matrix. |
| perViewErrors | Output vector of the RMS re-projection error estimated for each pattern view. |

## <https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga617b1685d4059c6040827800e72ad2b6>

## stereoRectify()



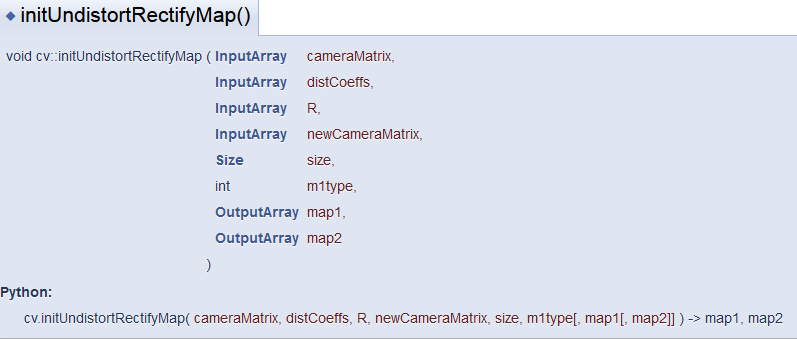
Computes rectification transforms for each head of a calibrated stereo camera.

Parameters

|  |  |
| --- | --- |
| cameraMatrix1 | First camera intrinsic matrix. |
| distCoeffs1 | First camera distortion parameters. |
| cameraMatrix2 | Second camera intrinsic matrix. |
| distCoeffs2 | Second camera distortion parameters. |
| imageSize | Size of the image used for stereo calibration. |
| R | Rotation matrix from the coordinate system of the first camera to the second camera, see [stereoCalibrate](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga91018d80e2a93ade37539f01e6f07de5). |
| T | Translation vector from the coordinate system of the first camera to the second camera, see [stereoCalibrate](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga91018d80e2a93ade37539f01e6f07de5). |
| R1 | Output 3x3 rectification transform (rotation matrix) for the first camera. This matrix brings points given in the unrectified first camera's coordinate system to points in the rectified first camera's coordinate system. In more technical terms, it performs a change of basis from the unrectified first camera's coordinate system to the rectified first camera's coordinate system. |
| R2 | Output 3x3 rectification transform (rotation matrix) for the second camera. This matrix brings points given in the unrectified second camera's coordinate system to points in the rectified second camera's coordinate system. In more technical terms, it performs a change of basis from the unrectified second camera's coordinate system to the rectified second camera's coordinate system. |
| P1 | Output 3x4 projection matrix in the new (rectified) coordinate systems for the first camera, i.e. it projects points given in the rectified first camera coordinate system into the rectified first camera's image. |
| P2 | Output 3x4 projection matrix in the new (rectified) coordinate systems for the second camera, i.e. it projects points given in the rectified first camera coordinate system into the rectified second camera's image. |
| Q | Output 4×4 disparity-to-depth mapping matrix (see [reprojectImageTo3D](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga1bc1152bd57d63bc524204f21fde6e02)). |
| flags | Operation flags that may be zero or [CALIB\_ZERO\_DISPARITY](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#gga7b31a379c097fb87997d28266762f12fae49985db1f4453c5dda1d701a3d8dd76) . If the flag is set, the function makes the principal points of each camera have the same pixel coordinates in the rectified views. And if the flag is not set, the function may still shift the images in the horizontal or vertical direction (depending on the orientation of epipolar lines) to maximize the useful image area. |
| alpha | Free scaling parameter. If it is -1 or absent, the function performs the default scaling. Otherwise, the parameter should be between 0 and 1. alpha=0 means that the rectified images are zoomed and shifted so that only valid pixels are visible (no black areas after rectification). alpha=1 means that the rectified image is decimated and shifted so that all the pixels from the original images from the cameras are retained in the rectified images (no source image pixels are lost). Any intermediate value yields an intermediate result between those two extreme cases. |
| newImageSize | New image resolution after rectification. The same size should be passed to initUndistortRectifyMap (see the stereo\_calib.cpp sample in OpenCV samples directory). When (0,0) is passed (default), it is set to the original imageSize . Setting it to a larger value can help you preserve details in the original image, especially when there is a big radial distortion. |
| validPixROI1 | Optional output rectangles inside the rectified images where all the pixels are valid. If alpha=0 , the ROIs cover the whole images. Otherwise, they are likely to be smaller (see the picture below). |
| validPixROI2 | Optional output rectangles inside the rectified images where all the pixels are valid. If alpha=0 , the ROIs cover the whole images. Otherwise, they are likely to be smaller (see the picture below). |

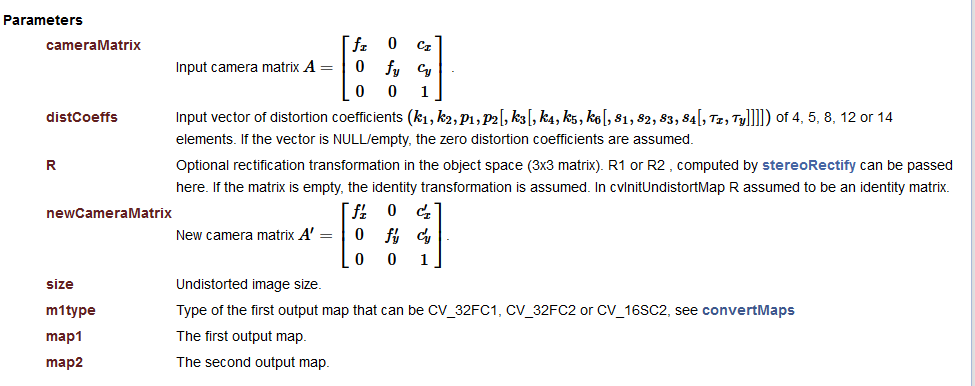
The function computes the rotation matrices for each camera that (virtually) make both camera image planes the same plane. Consequently, this makes all the epipolar lines parallel and thus simplifies the dense stereo correspondence problem. The function takes the matrices computed by stereoCalibrate as input. As output, it provides two rotation matrices and also two projection matrices in the new coordinates.

<https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html#ga7dfb72c9cf9780a347fbe3d1c47e5d5a>

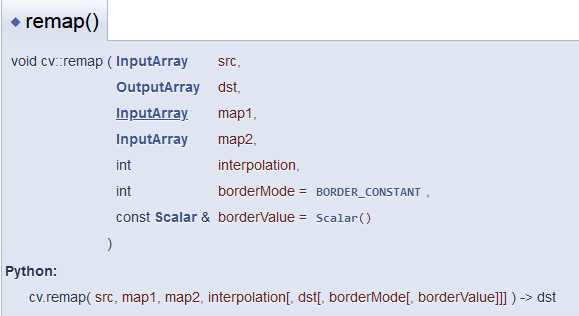


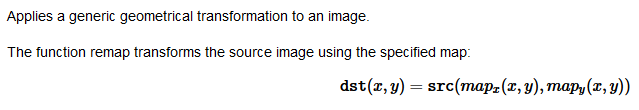
Computes the undistortion and rectification transformation map.

The function computes the joint undistortion and rectification transformation and represents the result in the form of maps for remap. The undistorted image looks like original, as if it is captured with a camera using the camera matrix =newCameraMatrix and zero distortion. In case of a monocular camera, newCameraMatrix is usually equal to cameraMatrix, or it can be computed by [getOptimalNewCameraMatrix](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga7a6c4e032c97f03ba747966e6ad862b1) for a better control over scaling. In case of a stereo camera, newCameraMatrix is normally set to P1 or P2 computed by [stereoRectify](https://docs.opencv.org/3.4/d9/d0c/group__calib3d.html#ga617b1685d4059c6040827800e72ad2b6) .



<https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html#gab75ef31ce5cdfb5c44b6da5f3b908ea4>





Parameters

|  |  |
| --- | --- |
| src | Source image. |
| dst | Destination image. It has the same size as map1 and the same type as src . |
| map1 | The first map of either (x,y) points or just x values having the type CV\_16SC2 , CV\_32FC1, or CV\_32FC2. See convertMaps for details on converting a floating point representation to fixed-point for speed. |
| map2 | The second map of y values having the type CV\_16UC1, CV\_32FC1, or none (empty map if map1 is (x,y) points), respectively. |
| interpolation | Interpolation method (see [InterpolationFlags](https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html#ga5bb5a1fea74ea38e1a5445ca803ff121)). The methods [INTER\_AREA](https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html#gga5bb5a1fea74ea38e1a5445ca803ff121acf959dca2480cc694ca016b81b442ceb) and [INTER\_LINEAR\_EXACT](https://docs.opencv.org/3.4/da/d54/group__imgproc__transform.html#gga5bb5a1fea74ea38e1a5445ca803ff121ac00f4a8155563cdc23437fc0959da935) are not supported by this function. |
| borderMode | Pixel extrapolation method (see [BorderTypes](https://docs.opencv.org/3.4/d2/de8/group__core__array.html#ga209f2f4869e304c82d07739337eae7c5)). When borderMode=[BORDER\_TRANSPARENT](https://docs.opencv.org/3.4/d2/de8/group__core__array.html#gga209f2f4869e304c82d07739337eae7c5a886a5eb6b466854d63f9e742d5c8eefe), it means that the pixels in the destination image that corresponds to the "outliers" in the source image are not modified by the function. |
| borderValue | Value used in case of a constant border. By default, it is 0. |