

The final argument, `mask`, is used only with the robust methods, and it is an output. If an array is provided, `cv::findHomography()` will fill that array indicating which points were actually used in the best computation of H .

The return value will be a 3×3 matrix. Because there are only eight free parameters in the homography matrix, we chose a normalization where $H_{33} = 1$ (which is usually possible except for the quite rare singular case $H_{33} = 0$). Scaling the homography could be applied to the ninth homography parameter, but usually prefer to instead scale by multiplying the entire homography matrix by a scale factor, as described earlier in this chapter.

Camera Calibration

We finally arrive at camera calibration for camera intrinsics and distortion parameters. In this section, we'll explain how to compute these values using `cv::calibrateCamera()` and also how to use these models to correct distortions in the images that the calibrated camera would have otherwise produced. First we will say a little more about just how many views of a chessboard are necessary in order to solve for the intrinsics and distortion. Then we'll offer a high-level overview of how OpenCV actually solves this system before moving on to the code that makes it all easy to do.

How many chess corners for how many parameters?

To begin, it will prove instructive to review our unknowns; that is, how many parameters are we attempting to solve for through calibration? In the OpenCV case, we have four parameters associated with the *camera intrinsic matrix* (f_x, f_y, c_x, c_y) and five (or more) *distortion* parameters—the latter consisting of three (or more) radial parameters ($k_1, k_2, k_3, k_4, k_5, k_6$) and the two tangential (p_1, p_2).³² The intrinsic parameters control the linear projective transform that relates a physical object to the produced image. As a result, they are entangled with the *extrinsic* parameters, which tell us where that object is actually located.

The distortion parameters are tied to the two-dimensional geometry of how a pattern of points gets distorted in the final image. In principle, then, it would seem that just three corner points in a known pattern, yielding six pieces of information, might be all that is needed to solve for our five distortion parameters. Thus a single view of our calibration chessboard could be enough.

³² It is commonplace to refer to the total set of the camera intrinsic matrix parameters and the distortion parameters as simply *the intrinsic parameters* or *the intrinsics*. In some cases, the matrix parameters will also be referred to as the *linear intrinsic parameters* (because they collectively define a linear transformation), while the distortion parameters are referred to as the *nonlinear intrinsic parameters*.

However, because of the coupling between the intrinsic parameters and the extrinsic parameters, it turns out that one will not be enough. To understand this, first note that the extrinsic parameters include three rotation parameters (ψ, ϕ, θ) and three translation parameters (T_x, T_y, T_z) for a total of six per view of the chessboard. Together, the four parameters of the camera intrinsic matrix and six extrinsic parameters make 10 altogether that we must solve for, in the case of a single view, and 6 more for each additional view.

Let's say we have N corners and K images of the chessboard (in different positions). How many views and corners must we see so that there will be enough constraints to solve for all these parameters?

- K images of the chessboard provide $2 \cdot N \cdot K$ constraints (the factor of 2 arises because each point on the image has both an x - and a y -coordinate).
- Ignoring the distortion parameters for the moment, we have 4 intrinsic parameters and $6 \cdot K$ extrinsic parameters (since we need to find the 6 parameters of the chessboard location in each of the K views).
- Solving then requires that we have: $2 \cdot N \cdot K \geq 6 \cdot K + 4$ (or, equivalently, $(N - 3) \cdot K \geq 2$).

So it would seem that if $N = 5$, then we need only $K = 1$ image, but watch out! For us, K (the number of images) must be more than 1. The reason for requiring $K > 1$ is that we are using chessboards for calibration to fit a homography matrix for each of the K views. As discussed previously, a homography can yield at most eight parameters from four (x, y) pairs. This is because only four points are needed to express everything that a planar perspective view can do: it can stretch a square in four different directions at once, turning it into any quadrilateral (see the perspective images in [Chapter 11](#)). So, no matter how many corners we detect on a plane, we only get four corners' worth of information. Per chessboard view, then, the equation can give us only four corners of information or $(4 - 3) \cdot K > 1$, which means $K > 1$. This implies that two views of a 3×3 chessboard (counting only internal corners) are the minimum that could solve our calibration problem. Consideration for noise and numerical stability is typically what requires the collection of more images of a larger chessboard. In practice, for high-quality results, you'll need at least 10 images of a 7×8 or larger chessboard (and that's only if you move the chessboard enough between images to obtain a "rich" set of views).

This disparity between the theoretically minimal 2 images and the practically required 10 or more views is a result of the very high degree of sensitivity that the intrinsic parameters have on even very small noise.