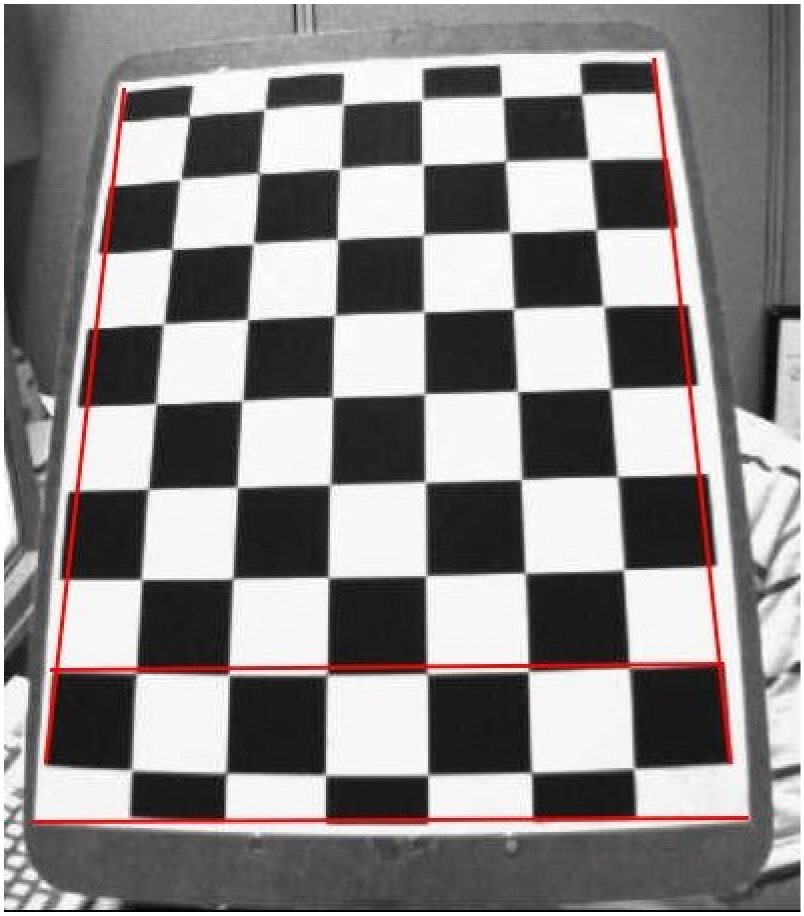
Calibration

Теория

Некоторые камеры вносят значительные искажения в изображения. Двумя основными видами искажений являются радиальное искажение и тангенциальное искажение. Примеры радиального искажения:

Радиальное искажение приводит к тому, что прямые линии кажутся изогнутыми. Радиальное искажение становится больше по мере удаления точек от центра изображения. Пример положительного радиального искажения на фото шахматной доски:



Радиальное искажение может быть представлено следующим образом:

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

Аналогичным образом, тангенциальное искажение возникает из-за того, что объектив для получения изображения не выровнен идеально параллельно плоскости изображения. Таким образом, некоторые области на изображении могут выглядеть ближе, чем ожидалось. Величина тангенциального искажения может быть представлена следующим образом:

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

Короче говоря, нам нужно найти пять параметров - коэффициенты искажения:

$$Distortion\ coefficients = (k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3)$$

В дополнение к этому, нам нужна некоторая другая информация, например, внутренние и внешние параметры камеры. Внутренние параметры специфичны для конкретной камеры. Они включают в себя такую информацию, как фокусное расстояние (fx, fy) и оптические центры (cx,cy). Фокусное расстояние и оптические центры могут быть использованы для создания матрицы камеры, которая может быть использована для устранения искажений, вызванных объективами конкретной камеры. **Матрица камеры уникальна для конкретной камеры, поэтому после расчета ее можно повторно использовать на других изображениях, сделанных той же камерой (поэтому нам и нужно сперва провести калибровку).** Это выражается в виде матрицы 3x3:

$$camera\ matrix = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Внешние параметры соответствуют векторам поворота и перемещения, которые преобразуют координаты трехмерной точки в систему координат.

$$z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A[R \quad T] \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix}$$

z_c - масштабный коэффициент

(u, v) - координаты точки на изображении

A - camera matrix

R, T - вектора поворота и перемещения

(x_w, y_w, z_w) - координаты в пространстве

Чтобы найти эти параметры, мы должны предоставить несколько образцов изображений четко определенного рисунка (например, шахматной доски). Мы находим некоторые конкретные точки, относительное расположение которых нам уже известно (например, углы квадратов на шахматной доске). Мы знаем координаты этих точек в реальном пространстве, и мы знаем координаты на изображении, поэтому мы можем рассчитать коэффициенты искажения. Для получения лучших результатов нам нужно не менее 10 тестовых шаблонов.

Реализация

Будем калибровать камеру используя несколько изображений шахматной доски под разными углами. Важными входными данными, необходимыми для калибровки камеры, являются набор трехмерных точек реального мира и соответствующие двумерные координаты этих точек на изображении. Точки на 2D-изображении в порядке, которые мы можем легко найти по изображению. (Эти точки изображения - места, где два черных квадрата соприкасаются друг с другом на шахматных досках)

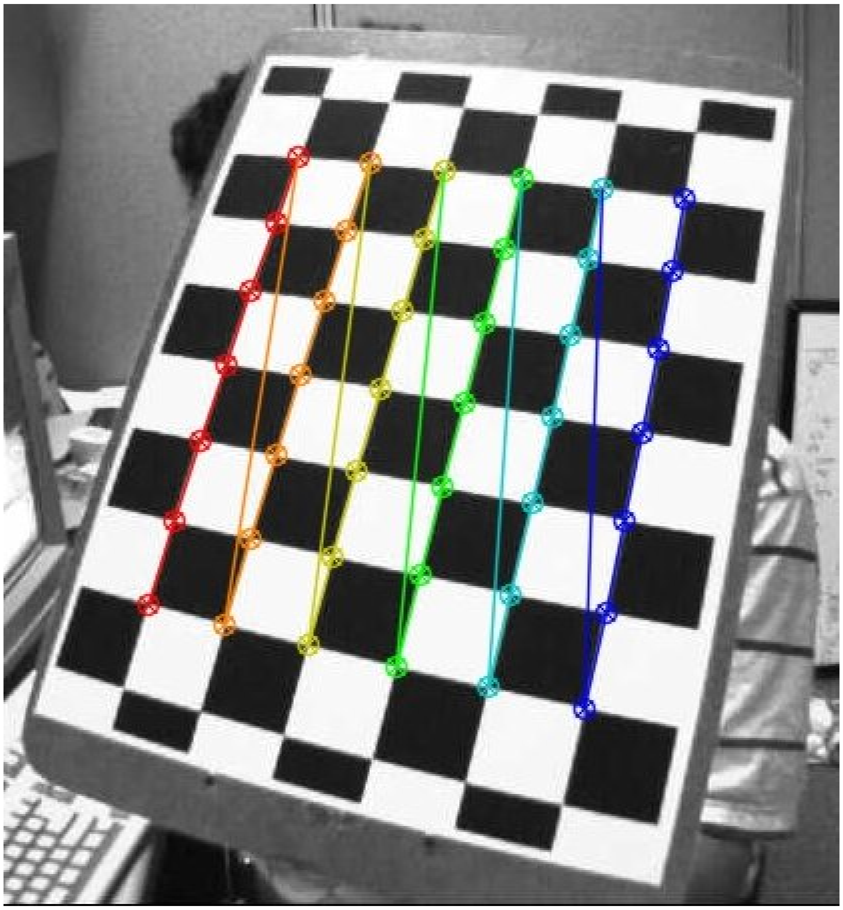
А как насчет трехмерных точек из реального пространства? Эти изображения сделаны со статической камеры, а шахматные доски размещены в разных местах и ориентациях. Итак, нам нужно знать значения (X,Y, Z). Но для простоты мы можем сказать, что шахматная доска оставалась неподвижной в плоскости XY (таким образом, Z всегда = 0), и камера была перемещена соответствующим образом. Это соображение помогает нам найти только значения X,Y. Теперь для значений X, Y мы можем просто передать точки как (0,0), (1,0), (2,0) и т.д. координаты точек точек. В этом случае результаты, которые мы получим, будут соответствовать размеру квадрата шахматной доски. Но если мы знаем размер квадрата (скажем, 30 мм), мы можем передать значения следующим образом (0,0), (30,0), (60,0) и т.д. Таким образом, мы получаем результаты в миллиметрах. (В данном случае мы не знаем размер квадрата, так как мы не делали этих изображений, поэтому мы передаем данные в терминах размера квадрата).

3D-точки называются точками объекта, а точки 2D-изображения - точками изображения.

Установка
Итак, чтобы найти узор на шахматной доске, мы можем использовать функцию cv.findChessboardCorners(). Нам также нужно указать, какой шаблон мы ищем, например, сетку 8x8, сетку 5x5 и т.д. Функция возвращает угловые точки и retval,

которое будет иметь значение True, если шаблон получен. Эти углы будут расположены в определенном порядке (слева направо, сверху вниз).

Одно изображение с нарисованным на нем узором:



Калибровка

Теперь, когда у нас есть наши объектные точки и точки изображения, мы готовы приступить к калибровке. Мы можем использовать функцию cv.calibrateCamera(), которая возвращает матрицу камеры, коэффициенты искажения, векторы поворота и перемещения и т.д.

Избавление от искажения

Теперь мы можем взять изображение и исправить искажение. OpenCV имеет два метода для этого. Однако сначала мы можем уточнить матрицу камеры на основе параметра свободного масштабирования, используя cv.getOptimalNewCameraMatrix(). Если параметр масштабирования alpha=0, он возвращает неискаженное изображение с минимальным количеством нежелательных пикселей. Таким образом, это может даже удалить некоторые пиксели в углах изображения. Если альфа=1, все пиксели сохраняются с некоторыми дополнительными черными изображениями.