P&O EAGLE:
Image formation and camera models

(background information for the image processing module)

Tinne Tuytelaars

September 2016

Contents

1 Introduction

Before you start performing measurements based on image input, it is important to first get some insight in the image formation process. This will help you understand the relation between image coordinates (typically expressed in rows and columns), camera coordinates and world coordinates. In this document, we will first explain the basics of the image formation process based on the pinhole camera model (section 2). Next, we will describe further distortions you can observe in an image, caused by deviations from the simple pinhole camera model (section 3). We will discuss the internal and external camera parameters (section 4) and explain how to estimate them (section 5). Finally we will discuss the special case of looking at a planar surface - which is the case in the context of this project, where the camera looks down towars the ground floor (section 6).

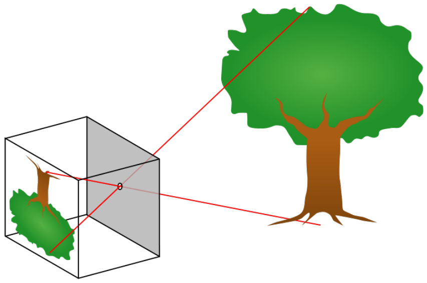

Figure 1: In a pinhole camera an image of the scene is formed by the rays of light that are reflected by the objects in the scene and fall through the center of the lens onto the image plane. The image plane is physically located behind the center of projection and appears upside-down.

2 The pinhole camera model

An image is formed by capturing the light falling onto a light-sensitive surface or sensor over a certain period of time (the *exposure time*, determined by the *shutter speed* of the camera). The amount of light (number of photons) and color of the light (wavelengths) reaching a certain *pixel* determine its (R,G,B) values for the red, green and blue color bands respectively. Light always travels in straight lines. Light originating from one or more light sources gets reflected by one or more object surfaces in the scene. How much light gets reflected in which direction depends on the material properties (color and reflectance properties) as well as the geometry (surface normal relative to the incoming light).

The simplest camera model is the *pinhole camera model*, as illustrated in Figure 1. It models the image formation process of the *camera obscura*. In this simple predecessor of the camera, an image of the outside world is formed on a plane wall, the image plane, that is enclosed in a completely opaque and reflectionless room or box. Light is only piercing in through a minute hole opposed to the image plane. Note how the recorded image appears upside-down.

The pinhole is also referred to as the *center of projection*. The distance between the center of projection and the image plane is referred to as the *focal length*. Using a coordinate axes frame with its origin in the center of projection (see Figure 2), a simple relation between the 3D world coordinates of a point P_o and its projection P_i can be derived. The light coming from (or reflected by) a point $P_o(X_o, Y_o, Z_o)$ on some object in the scene, in the direction of the hole $O(0, 0, 0)$, reaches the image plane in a point $P_i(X_i, Y_i, -f)$. This image formation geometry is referred to as *perspective projection*. As long as the diameter of the hole is sufficiently large compared to the wavelength of the light, the influence of diffraction will be negligible, and the light may be assumed

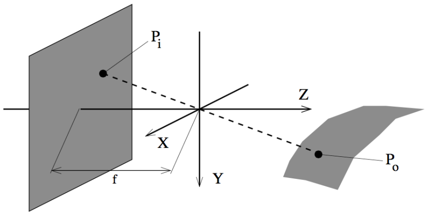

Figure 2: Using a coordinate axes frame with its origin in the center of projection (pinhole), a simple relation between the 3D world coordinates P_o and its projection P_i can be derived.

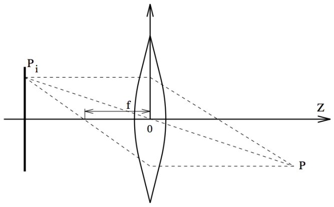

Figure 3: Modern cameras use lenses to focus all the light from within a small cone to fall onto the same image point.

to follow a straight path. From similar triangles we find

$$\frac{X_i}{X_o} = \frac{Y_i}{Y_o} = -\frac{f}{Z_o} = -m$$

m is called the *linear magnification*. This confirms some common intuitions about images: objects further away from the camera, will appear smaller in the image (the linear magnification m is inversely proportional to the distance from the projection center, Z_o).

Modern cameras actually still mimic this camera obscura setup, but with some technical refinements. The problem with the camera obscura is that the pinhole needs to be really small - otherwise, the image will get blurred, as light from different directions will reach the same point on the image sensor. But with a small pinhole, only a limited amount of light reaches the image plane, resulting in dark images, unless very long exposure times are used. Therefore, in modern cameras the small pinhole is replaced by a larger *diafragma*, letting more light fall onto the image plane, in combination with a *lens*, refocusing the light so as to ensure that the image will remain sharp. This is illustrated in Figure 3. The figure shows a section through the *optical axis*, i.e. the Z-axis, that passes through the center of the lens. The lens surfaces are assumed spherical on both sides. All light coming from a point P and falling on the lens will now contribute in the brightness formed at P_i. The geometry of the projection remains the same, as can be seen from the ray through the optical center of the lens, which is still unbroken. The price we pay is that only points at a certain distance (or in practice: over a certain range of distances) will be sharply imaged. The constant f is again referred to as the *focal length.* Note that it no longer corresponds to the distance between image plane and center of projection: we need to change the location of the image plane, depending on which depths we want to focus on. If the image plane is placed at a distance f from the optical lens center, points at infinity (or at large enough distances) are in focus.

3 Lens distortion

In the above model three important assumptions were made that are not perfectly matched by real lenses or lens systems:

1. all the rays emerging from a point in space are focused onto the same image point,
2. these image points all fall in a single image plane, and
3. magnification, for a given depth, is constant throughout the image plane.

These assumptions lead to a simple image formation model, where the image of a point is found by connecting the point and the center of the lens by a straight line and finding the single point where that line intersects the image plane, assumed to be orthogonal to the optical axis of the lens. The deviations from this ideal are called *aberrations.* Two general families of aberrations are distinguished: geometrical and chromatic.

Geometrical aberrations are deviations from the ideal model that become visible as image distortions or degradations like blurring. These remain small for rays close to the optical axis. Incoming rays at the outer portions of the lens do not fulfill the assumptions so well and deviations from the simplified model become visible.

Chromatic aberrations correspond to the visibly different behaviour of different wavelengths, due to the dependence of the refractive index on the wavelength. These effects mostly become visible in cheap camera sensors. In this project, the effect of chromatic aberrations can be neglected, as it plays only a minor role. So we will only consider geometric aberrations.

Radial distortion is the most important form of geometrical aberration, not only because it often occurs and its effects are often the strongest, but also because one can undo its effects if one succeeds in modeling it. The projection model resulting from the ideal case is linear in the sense that the scene point, the corresponding image point and the center of the lens (also called *center of projection*) are collinear, and that straight lines in the scene do generate straight lines in the image. This 'perspective' projection therefore only models the linear effects in the image formation process. Images taken by real cameras, on the other hand, also experience non-linear deformations or distortions which make the simple linear pinhole model inaccurate.

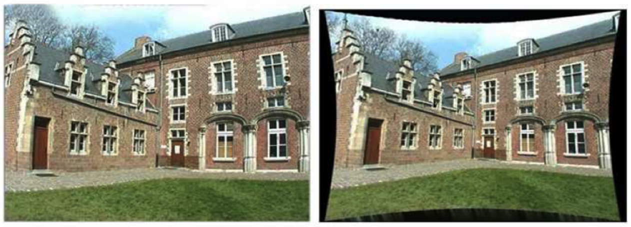

Figure 4: Left: An image exhibiting radial distortion. The vertical wall at the left of the building appears bent in the image and the gutter on the frontal wall on the right appears curved too. Right: The same image after correction for the radial distortion. Straight lines in the scene now appear as straight lines in the image as well.

Figure 4 shows an example of a radially distorted image. Radial distortion is caused by a systematic variation of the optical magnification when radially moving away from a certain point, called the *center of distortion.* The larger the distance between an image point and the center of distortion, the larger the effect of the distortion. Thus, the effect of the distortion is mostly visible near the image boundaries. This can clearly be seen in Figure 4. Straight lines near the edges of the image are no longer straight but are bent. For practical use, the center of radial distortion can often be assumed to coincide with the principal point, which usually also coincides with the center of the image. But it should be noted that these are only approximations. Dependent on the accuracy requirements, a more precise determination may be necessary.

Radial distortion is a non-linear effect and is typically modeled using a Taylor expansion. Typically, only the even order terms play a role in this expansion, i.e. the effect is symmetric around the center. This results in the following relation between the original image coordinates (x, y) and the corrected images coordinates $(x_{corrected}, y_{corrected})$:

$$x_{corrected} = x(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \ldots)$$
$$y_{corrected} = y(1 + \kappa_1 r^2 + \kappa_2 r^4 + \kappa_3 r^6 + \ldots)$$

with r the distance of the point (x, y) from the center of distortion. The lower order terms of this expansion are the most important ones and typically one does not compute more than three parameters $(\kappa_1, \kappa_2, \kappa_3)$.

Apart from radial distortion, also *tangential distortion* is sometimes corrected for. Tangential distortion is caused by a misalignment of the lens with respect to the camera plane, i.e. if they're not perfectly parallel to one another. For tangential distortion, we have:

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

We will later see how to estimate the parameters involved $(\kappa_1, \kappa_2, \kappa_3, p_1, p_2)$ for a given camera, to compensate for the effect of lens distortions (see section 5).

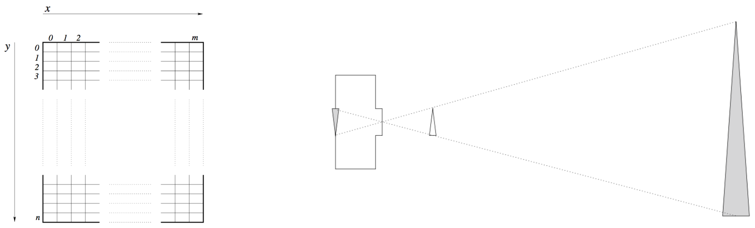

Figure 5: Left: Image coordinates typically have a coordinate axes frame with the y-axis pointing downwards and the origin in the top left corner of the image. Right: perspective projection results in an upside-down image on the image plane (grey). To avoid this inversion, one often works with a virtual image plane in front of the center of projection instead (white).

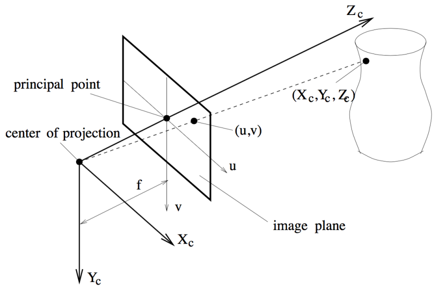

Figure 6: The pinhole camera model and camera coordinate axes frame. A point with coordinates (X_c, Y_c, Z_c) is projected at a point (u, v).

4 Intrinsic and extrinsic camera parameters

With the introduction of lenses, and after compensation for some of the most severe distortions they introduce (if needed), we can keep using the simple pinhole camera model, at the expense of limited focus for a given focal length. This model is shown again in the right part of Figure 5. In the figure, two projections are shown. The one shown in grey corresponds to the projection as it will physically exist in the camera. The one shown in white is positioned at the same distance from the center of projection, but *before* the camera. In this virtual image plane, the image is upright. This virtual placement of the image plane eliminates sign reversals of the (virtual) image coordinates with respect to the original point coordinates. This placement is also justified in the sense that if one looks at a photograph or if an image is presented on a screen, one also shows the images upright as they would be obtained when the image plane were at the virtual position. Henceforth, the image plane will always be assumed at this position.

Consider Figure 6. The camera coordinate system (X_c, Y_c, Z_c) has been chosen in a specific way:

- its origin lies at the center of projection, i.e. in the center of the lens of the camera,
- the Z_c axis coincides with the optical axis of the lens or objective, and
- the plane through the center of projection and perpendicular to the optical axis is the X_cY_c-plane, with the X_c-axis parallel to the main direction, the rows say, of the image. The orientation of the Y_c is fixed by then, as it is orthogonal to both X_c and Z_c.

This coordinate frame will be referred to as the *camera coordinate frame*, hence the subscript c. Note that the u and v coordinate axes in the image plane are parallel to the X_c and Y_c axes. They intersect at the *principal point*, i.e. the point where the optical axis intersects the camera plane. The image of a point P in the scene with coordinates (X_c, Y_c, Z_c) is the intersection of the line through P and the center of projection with the image plane, i.e. the plane with equation $Z = f$. The (u, v)-coordinates of this image point in the image plane are directly found from similar triangles:

$$u = f\frac{X_c}{Z_c}$$

$$v = f\frac{Y_c}{Z_c}$$

The perspective projection model we have derived still isn't a very practical tool. It is incomplete in at least two ways:

1. the coordinates of points will often be known in terms of some world coordinate frame, not the camera coordinate frame; we need to transform coordinates first to the latter, and
2. in the current model, the coordinates of points in the image plane are derived from camera frame coordinates; we need to express these coordinates in terms of the row and column coordinates (pixels) that we get out of the camera (see Figure 5).

These are the aspects we deal with next. The transition from the geometrical (u, v)- coordinates to the actual pixel coordinates, which will be referred to as (x, y)-coordinates, is modeled by an affine transformation of the form:

$$\begin{aligned} x &= k_x u + sv + x_0 \\ y &= \phantom{k_x u + {}} k_y v + y_0 \end{aligned}$$

where

- (x_0, y_0) are the pixel coordinates of the principal point.

- k_x is the number of pixels per unit length (as measured on the imaging sensor) in the horizontal direction (and thus gives 1/width of a pixel).
- k_y is the number of pixels by unit length in the vertical direction (and thus implicitly describes the 1/height of a pixel).
- s indicates how strong the shape of the pixels deviates from being rectangular. This parameter is usually referred to as the *skewness* of the pixels. Observe that $s = 0$ corresponds to rectangular pixels. This is the normal case.

Of course, in reality the pixel coordinates are discrete and small patches of the image plane will be lumped together into single pixels. These pixels have integer coordinates (column and row numbers). This discretisation is often not taken into account while performing calculations. The numbers k_x and k_y depend on the focal length and the zooming distance of the lens or objective. The ratio $\frac{k_y}{k_x}$ is called the *aspect ratio* of the pixels. Deviations from 1 indicate that the pixels are not square.

The numbers k_x, k_y, s, x_0 and y_0 are referred to as the *internal camera parameters*. When they are known, one says that the camera is *internally calibrated*. In practice this means that one can convert the observed (x, y)-image coordinates to metric (u, v) coordinates, and vice versa.

Going from 3D coordinates expressed in the camera coordinate axes frame to coordinates expressed in a world coordinate axes frame, involves further transformations. We won't give the full derivation here. We just point out that this additional transformation is characterized by the position of the camera (or actually, the origin of the camera-centered coordinate axes frame) $\mathbf{C}$ and its rotation $\mathbf{R}$ with respect to the world coordinate axes frame (which you can think of as aligned with the grid of red lines on the ground floor). These are referred to as the *external camera parameters*. When they are known, one says that the camera is *externally calibrated*. In practice this means that one can convert the scene coordinates into camera-centered coordinates, and vice versa. When both the internal and external camera parameters are known, we say the camera is *fully calibrated*.

In the context of the EAGLE project, you may need to apply further transformations, e.g. from the camera coordinate axes frame to a coordinate axes frame centered on the drone (for the control), or to a coordinate axes frame centered on the coil of the inductive charger.

5 Camera calibration

Camera calibration is the process of determining the internal and external camera parameters of a given camera. It allows to relate measurements in image coordinates to the actual 3D scene. Camera calibration tools often also include estimation of the parameters of radial and tangential distortion.

The process of camera calibration involves taking a couple of images of an object with known geometry, such as a planar checker board. We then bring

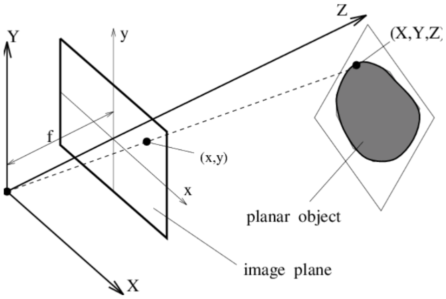

Figure 7: Pinhole camera model, f is the focal length, X,Y,Z are coordinates in the world coordinate frame and x, y coordinates in the image coordinate frame.

points in the image in correspondence with points on the physical object. Each of these correspondences provides some constraints, based on which we can ultimately determine all camera parameters. More details on this procedure can be found in the corresponding OpenCV tutorial:

http://opencv24-python-tutorials.readthedocs.io/en/latest/py_tutorials/py_calib3d/py_calibration/py_calibration.html#calibration

6 Looking at a planar surface

We focus here on the special case of a planar surface, viewed by a pinhole camera model, as depicted in Figure 7. In this section we discuss two different types of deformations to be expected under projection for such planar surfaces:

1. The surfaces are viewed from a perpendicular direction and can only rotate about an axis parallel to this direction; any 3D translation is allowed.

2. The surfaces can be viewed from any direction and can be placed at arbitrary distances from the camera.

In the EAGLE project, the actual setting may often be approximated by the first model: when the drone is hovering, and assuming the camera is mounted perfectly, we can assume the image plane is parallel to the ground floor and we can use this model. In practice, however, the camera mount may not be aligned perfectly. Also, the drone will tilt somewhat while it's flying. This will cause some deviations from the first model. For accurate measurements (e.g. to decode the QR codes), the more complex model of the second case may be needed.

Case 1 We suppose a point on the surface is described by specifying the three-dimensional coordinates (X, Y, Z). As we know, in this case we only allow the surface to rotate around an axis parallel to the Z axis, say over an angle θ.

Furthermore any three-dimensional translation can be applied. If we denote the coordinates after such transformation as (X', Y', Z'), we find a relationship of the form

$$\begin{aligned} X_0 &= cos\theta X - sin\theta Y && + t_1, \\ Y_0 &= sin\theta X + cos\theta Y && + t_2, \\ Z_0 &= && Z + t_3. \end{aligned}$$

Now, suppose that the surface point (X, Y, Z) projects to image point (x, y) and the point (X', Y', Z') to (x', y'). We are interested in the transformation bringing the coordinates (x, y) into correspondence with (x', y'), i.e. in the group of transformations describing the deformations between two camera views. This relationship is particularly easy to derive in this case. Since the two planes, i.e. before and after the transformation, are both parallel to the image plane, points on each of the planes will have a constant Z value. This brings us in the situation where the projection equations simplify to a scaling. We have that

$$\begin{aligned} x &= kX, & x' &= k'X', \\ y &= kY, & y' &= k'Y', \end{aligned}$$

with $k = f/Z$ and $k' = f'/Z'$ two constant scale factors. Thus, using the relationships between X' and X we e.g. find that

$$\frac{x'}{k'} = cos\theta\frac{x}{k} - sin\theta\frac{y}{k} = t_1$$

Also using the relationship between Y' and Y we obtain

$$\begin{pmatrix} \text{x'} \\ \text{y'} \end{pmatrix} = \frac{k'}{k} \begin{pmatrix} \cos\theta & \text{-}\sin\theta \\ \sin\theta & \cos\,\theta \end{pmatrix} \begin{pmatrix} \text{x} \\ \text{y} \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}$$

We conclude that the image projection of the surface undergoes a *similarity transformation*. In particular, the surface is transformed via combined scaling, rotation, and translation in the image plane. As a further simplification we could consider the case where the object doesn't come closer to or move away from the camera, i.e. $t_3 = 0$. Then $Z' = Z$, hence $k' = k$ if also the camera focal length remains unchanged and the transformation becomes a *Euclidean motion*, i.e. a combination of rotation and translation (in the image plane). The square grid painted on the ground floor will appear as a square grid in the image, with the size of the grid cells determined by the height of the drone (Z).

Case 2 In this case the plane can undergo arbitrary three-dimensional rotations and translations:

$$\begin{pmatrix} \text{X'} \\ \text{Y'} \\ \text{Z'} \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \begin{pmatrix} \text{X} \\ \text{Y} \\ \text{Z} \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix}$$

We again use the perspective projection model:

$$x = f\frac{X}{Z} \qquad x' = f'\frac{X'}{Z'}$$

$$y = f\frac{Y}{Z} \qquad y' = f'\frac{Y'}{Z'}$$

Expanding the expression for x' yields:

$$\begin{aligned} x' &= f'\frac{X'}{Z'} && (1)\\ &= f'\frac{r_{11}X + r_{12}Y + r_{13}Z + t_1}{r_{31}X + r_{32}Y + r_{33}Z + t_3} && (2)\\ &= f'\frac{r_{11}f\frac{X}{Z} + r_{12}f\frac{Y}{Z} + r_{13}f\frac{Z}{Z} + t_1\frac{f}{Z}}{r_{31}f\frac{X}{Z} + r_{32}f\frac{Y}{Z} + r_{33}f\frac{Z}{Z} + t_3\frac{f}{Z}} && (3) \end{aligned}$$

Thus,

$$\begin{aligned} x' &= f'\frac{r_{11}x + r_{12}y + r_{13}f + t_1\frac{f}{Z}}{r_{31}x + r_{32}y + r_{33}f + t_3\frac{f}{Z}} && (4)\\ y' &= f'\frac{r_{21}x + r_{22}y + r_{23}f + t_2\frac{f}{Z}}{r_{31}x + r_{32}y + r_{33}f + t_3\frac{f}{Z}} && (5) \end{aligned}$$

In order to eliminate Z we use the planarity of the surface (X, Y, Z) which implies that real numbers a, b, c and d can be found such that

$$aX + bY + cZ + d = a\frac{xZ}{f} + b\frac{yZ}{f} + c\frac{zZ}{f} + d = 0$$

Rewriting this we get:

$$Z = \frac{-df}{ax + by + cf}$$

This finally gives us:

$$\begin{aligned} x' &= f'\frac{(r_{11} - \frac{t_1}{d}a)x + (r_{12} - \frac{t_1}{d}b)y + (r_{13} - \frac{t_1}{d}c)f}{(r_{31} - \frac{t_3}{d}a)x + (r_{32} - \frac{t_3}{d}b)y + (r_{33} - \frac{t_3}{d}c)f} && (6)\\ y' &= f'\frac{(r_{21} - \frac{t_2}{d}a)x + (r_{22} - \frac{t_2}{d}b)y + (r_{23} - \frac{t_2}{d}c)f}{(r_{31} - \frac{t_3}{d}a)x + (r_{32} - \frac{t_3}{d}b)y + (r_{33} - \frac{t_3}{d}c)f} && (7) \end{aligned}$$

In other words, we find a two-dimensional *projective transformation* or *projectivity*, i.e. a transformation of the form

$$x' = \frac{p_{11}x + p_{12}y + p_{13}}{p_{31}x + p_{32}y + p_{33}}$$

$$y' = \frac{p_{21}x + p_{22}y + p_{23}}{p_{31}x + p_{32}y + p_{33}}$$

Under a projective transformation, a square in the 3D world will no longer be projected as a square on the image plane. Instead, it will be deformed into a quadrangle. This also affects the 2D grid of red lines. The effect of the perspective deformation in the image is that parallel lines of the grid are no longer parallel in the image. Instead, they converge to a single point, the *vanishing point* for that orientation. While the effect may be small for the red lines (read: can safely be ignored), it will also affect the QR codes.