Camera Coordinate Systems

Draw our 3D camera coordinate system:

It's defined by these choices:

- **Origin**: (Where?)
- **X-axis**: (Which direction?)
- **Y-axis**: (Which direction?)
- **Z-axis**: (Which direction?)

Other Camera Coordinate Systems

What properties should a camera coordinate system have?

OpenCV	**TIF**	**OpenGL**

Image Plane Coordinates

Using OpenCV coordinate conventions, draw a picture of a point at (X, Y, Z) being projected into the image plane.

Write the 3D coordinates of the point where the ray through the pinhole and (X, Y, Z) intersects the image plane.

Three coordinates is overkill to describe a point on a 2D image plane. Draw a sensible 2D coordinate system on the image plane, and write the equations relating the 2D coordinates to the 3D coordinates.

The Virtual Image Plane

Redraw the picture above, but this time add a second "imaginary" image plane in front of the pinhole.

Why do this?

More Image Plane Coordinates

If camera coordinates are in units of meters, what units are the image plane coordinates in?

We define two types of coordinates on the image plane:

1. **Metric coordinates** (x', y'): measured in meters (or inches, etc)

   - Origin:
   - x'-axis direction:
   - y'-axis direction:

2. **Pixel coordinates** (u, v): discrete array indices

   - Origin:
   - u-axis: (corresponds to _____ in camera coords)
   - v-axis: (corresponds to _____ in camera coords)
   - u indexes:
   - v indexes:

(again, this is the OpenCV convention)

An Example: TIF Coords

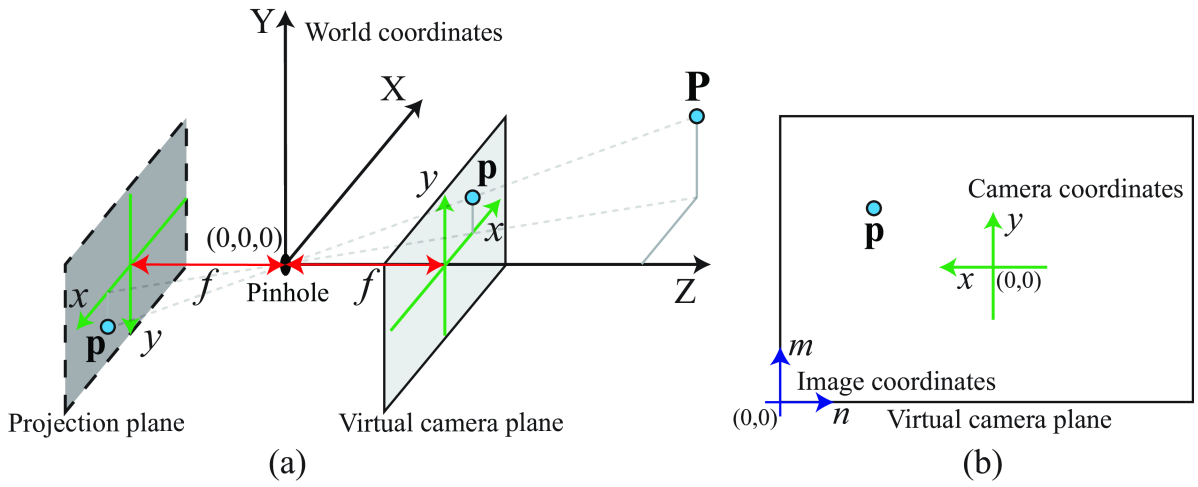

Figure 1: (a) Geometry of the pinhole camera. A 3D point **P** projects into the location **p** in the projection plane, located at a distance f of the pinhole. The virtual camera plane is a radially symmetric projection of the camera plane. (b) Relation between the camera, (x, y), and the image coordinate system (n, m).[2]

Activity:

Deriving the Projection Equations

1: Draw a picture like the one above, but using our (OpenCV's) coordinate systems.

[2]Figure from *Foundations of Computer Vision*. MIT Press. (CC-BY-NC-ND)

2: Using similar triangles, derive the equations for x' and y' in terms of X, Y, Z, and f.

3: Write down equations to convert from metric coordinates (x', y') to pixel coordinates (u, v). (What parameters do you need?)

Summary: Perspective Projection Equations

OpenCV Camera Coordinate System

Coordinate frame: Right-handed, origin at pinhole (optical center)

- X-axis: points right
- Y-axis: points down
- Z-axis: points forward (camera looks along $+Z$)

Projection to Metric Coordinates

Given a 3D point (X, Y, Z) in camera coordinates:

$$x' = f\frac{X}{Z}$$
$$y' = f\frac{Y}{Z}$$

Where:

- (x', y') are metric coordinates on the image plane (meters or mm)
- f is the focal length (meters or mm)
- Origin of (x', y') is where the optical axis intersects the image plane

Conversion to Pixel Coordinates

$$u = f_x\frac{X}{Z} + c_x$$
$$v = f_y\frac{Y}{Z} + c_y$$

Where:

- (u, v) are pixel coordinates (discrete array indices)
- $f_x = \frac{f}{s_x}$ is focal length in pixels (horizontal)
- $f_y = \frac{f}{s_y}$ is focal length in pixels (vertical)
- s_x, s_y are pixel dimensions in meters

- (c_x, c_y) is the principal point in pixels (optical axis intersection)
- Origin $(0, 0)$ is at the top-left corner of the image
- u increases to the right (columns), v increases downward (rows)

Note: Modern cameras typically have square pixels, so $s_x = s_y$ and $f_x = f_y$. People are often quite bad about distinguishing between f (in meters) and f_x, f_y (in pixels).