Pin Hole Cameras & Warp Functions

- Pinhole Camera
- Homogenous Coordinates
- Planar Warp Functions

Motivation

Humans view the world in a 2 dimensional way. Projecting 3D onto a 2D plane throws away a lot of information. Why would this be a good design? What's really interesting is that any 3D sensor that we're used to in the commercial world, there's an energy cost to that. It costs energy to project light into a 3D space. The probblem is that most things work as a function of distance. So if we need a lot of detail, we need a lot of distance and so a lot of energy. The sun, which is a source of a lot of energy, has motivated a lot of the geometry behind projecting light into 2D. Geometric methods can be seen as a tradeoff between hardware requirements and perception extent.

Pinhole Camera

Camera Obscura = Pinhole Camera in Latin. A pinhole in a wall in a dark room will project an image of the outside on the wall, only it's flipped (inverted). The eye also works similarly; the brain inverts it back.

Pinhole Camera model: In computer vision, we look things from a macro perspective. So it;s okay to make a ray assumption for photons (light). The frequency of light usually doesn't matter for this class. Light is a ray. We care about the light rays passing through the pinhole. In CV, we prefer to use the idea of a "virtual image", in *front* of the pinhole, rather than behind it, like it actually is in reality. Why? - We don't have to flip things and it's more convenient.

Object in world:

$$w = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

On image plane:

$$w = \begin{bmatrix} x \\ y \end{bmatrix}$$

Think about pixels as just mesaurements of the angle and the ray.

Normalized Camera: We assume that the focal length is 1, and so the only thing we really care about are the ratios. Pixels are defined in this way. However changing the focal length changes the image. For a normalized camera, focal length = 1.

The focal length has a big role to play in the field of view. Smaller focal length, larger field of view and vice versa. To determine y earlier, it was simply $\frac{v}{w}$ Now, it's $y = f\frac{v}{w}$.

Why do we have different focal lengths for different cameras? We discertize pixels. Often the resolution of pixels in the x-direction id diffrent than that in the y-direction. This brings in the aspect ratio. Thus the focal length encompasses both the distance of the image plane from the optical center, as well as the aspect ratio.

Offest paramters - When the principal ray does not strike the image plane at $(0, 0)$.

Skew - When the image plane is not perfectly perpendicular to the ray from the optical center. This affects the image. Thus we introduce a skew parameter. It is a heuristic that people have found does a good job acounting for these descripancies as it's applied to either one of the axes, usually the x-axis.

Radial distortion - There's a distortion in the way the rays are projected depending on the distance from the center.

Camera and World Coordinates

The world coordinate frame is where the 3D object actually is. A camera has its own notion of an origin and its axes. There are no "absolute" coordinates or an "absolute" coordinate frame, but relative coordinates make sense. To relate these two, we look at a transformation (a rotation and a translation). $w' = Rw + t$ where w' = point in camera frame, w = point in world frame. We have 9 parameters in the rotation matrix and 3 in the translation (this does not mean that there are these many degrees of freedom though).

R has the following constraints: Orthonormality, and $det(R) = +1$. If I did the SVD on R and I got the singular values, they should always be +1. What's the issue with these constraints?

Construct matrices R_1 and R_2 as in slides. Think of R_1 and R_2 as points in space. We literally draw a "line" and choose the point in the middle. Will this point lie in $SO(3)$? No. Thus, the set of 3D rotation matrices is not a convex set. This is a problem in computer vision since we usually add increments to things to find solutions. Doing this with R can lead to us going out of the set of rotation matrices. That's why you can't just apply many classical machine learning techniques to computer vision problems.

Complete pinhole camera model

$$\textit{The intrinsics}: \ K \ = \begin{bmatrix} f_x & \gamma & \delta_x \\ 0 & f_y & \delta_y \\ 0 & 0 & 1 \end{bmatrix}$$

Degrees of freedom:

Extrinsics refer to things changing from frame to frame, intrinsics remain the same.

Is the Pinhole camera model function a linear function? No. You cannot express a division like $\frac{v}{w}$ as a linear form $y = Aw' + b$. Also because parallel lines do not remain parallel if we apply the camera model. However there are certain situations where we can approximate this model as linear.

Perspective Transform

Point in the image = w_n

Point on the object in real wolrld = w_n

Learning extrinsic paramters: What are the best R and T to minimize the "rear projection error"? Learning intrinsic parameters: We often need to solve for K also at the same time. Use checkerboard patterns for camera calibration.

Homogenous Coordinates

We like these because we can get around the issue of having to "divide" by the depth. The camera model in homogenous coordinates: We adda zero-column at the end of the intrinsics matrix. We can now express the pinhole camera function as a linear transform. Homogenous coordinates allow you to change your objecctive so that it's easier to solve.

Planar Warp Functions

They are basically the best funtion to work with in geometry, we love playing with planes. Depth is constant. We can also break things up into planes piece-wise. There is also always a one-to-one mapping between points in two views, unlike other objects wen points are occluded. Important types of planar warps:

1. Euclidean
2. Similarity
3. Affine

4. Homography

Euclidean warp:

1. Assume the plane is fronto-parallel.
2. The distance between the camera and the plane is fixed = D.

However we can still rotate and translate in plane. We can get rid of one of the columns in the extrinsics because of the planar function ($w = 0$) and we can put D into the intrinsic matrix. 2D rotation matrix and a 2D translation matrix. How many DoF? There's 3, one for rotation and two for translation.