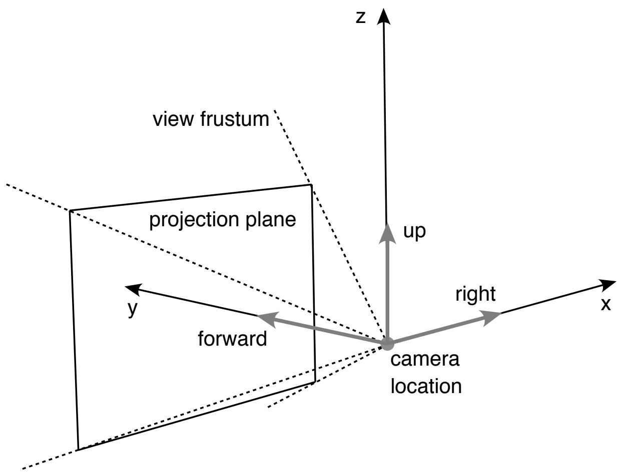
**Further details on deriving cube maps from equirectangular images.**

An equirectangular projection represents everything visible from a particular point in space. As such, any other 3D to 2D projection can be created, including a standard perspective projection. Since a perspective projection only captures a relatively small field of view, compared to the 360 by 180 degrees of the equirectangular projection, there are an infinity of possible perspective projections that might be calculated. Each of these possible perspective projections can be characterised by the pitch, roll and yaw rotation of the camera view frustum, as well as the horizontal and vertical field of view.

A single perspective projection cannot capture everything represented within an equirectangular projection, however multiple perspective projections can. An industry standard is to create 6 perspective views with the camera view frustum corresponding to the 6 faces of a cube centered at the camera position. Each perspective projection has a 90 degree field of view both horizontally and vertically.

The algorithm employed here to create a perspective projection starts by considering a virtual camera located at the origin with a view "forward" direction pointing down the positive y axis and with the z axis being the "up" vector. In the conventions used here a right hand coordinate system is used so the camera "right" vector is along the positive x axis.



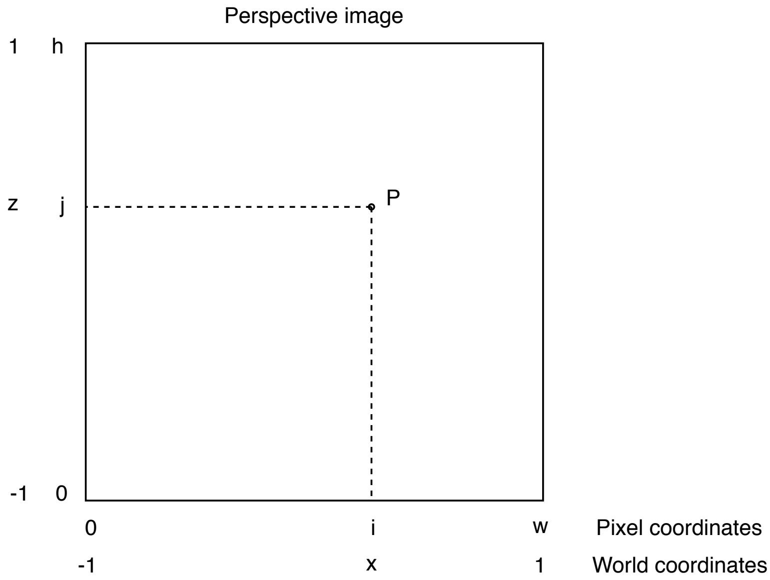

To create a particular perspective view one rotates this initial camera direction and orientation about any axis, or combination of axes. A roll in the chosen coordinate system is a rotation about the y axis (forward), panning is a rotation about the z axis (up) and tilting is a rotation about the x axis (right). To create the 6 faces of the cube map the initial camera view direction is rotated as shown, the horizontal and vertical field of view set to 90 degrees.

| Cube face | Rotation |
|---|---|
| front | - |
| left | Pan by 90 degrees |
| right | Pan by -90 degrees |
| back | Pan by 180 degrees |
| top | Tilt by -90 degrees |
| bottom | Tilt by 90 degrees |

The process of creating the perspective projection image from an equirectangular is performed in the reverse direction, that is, for every pixel (or subpixel for anti-alising) in the perspective image plane, what is the best RGB estimate in the equirectangular image. The high level process is as follows:

1. Initialise the camera. Set field of view to 90 degrees, located at the origin and looking down the y axis.

2. For every pixel (*i*,*j*) in the camera projection plane derive the corresponding 3D vector *P* in world coordinates.




$$P = \left(\left(\frac{2i}{w} - 1\right), 1, \left(\frac{2j}{h} - 1\right)\right)$$

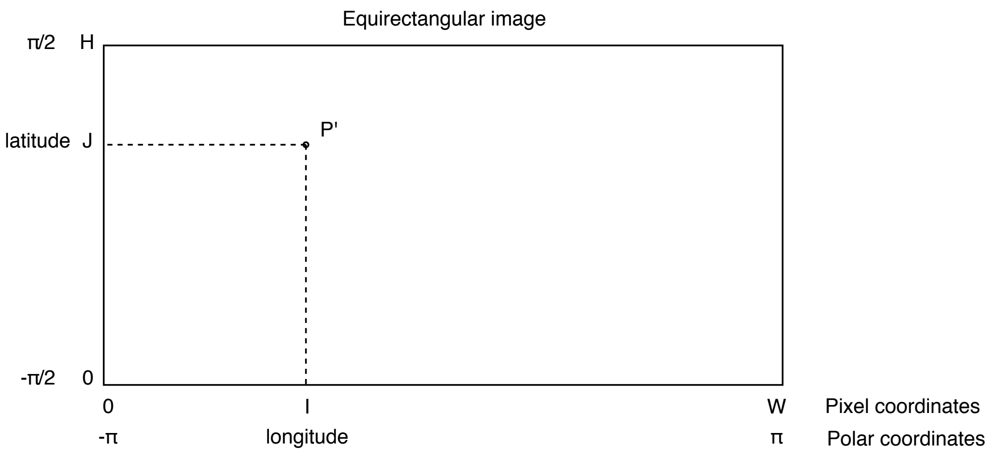
3. Rotate this vector *P* about the axes corresponding to roll, tilt and yaw to orientate the perspective camera as desired, call this vector *P*'

4. Calculate the longitude and latitude of this new vector.

$$longitude = atan2(P'_y, P'_x)$$

$$latitude = atan2\left(P'_z, \sqrt{P'^2_x + P'^2_y}\right)$$

5. Determine the image index (*I*,*J*) in the equirectangular image given the longitude and latitude of point *P'*. This gives the RGB value to assign to pixel (*i*,*j*) in the perspective image.




$$I = \frac{W(longitude + \pi)}{2\pi}$$

$$J = \frac{H(latitude + \pi/2)}{\pi}$$