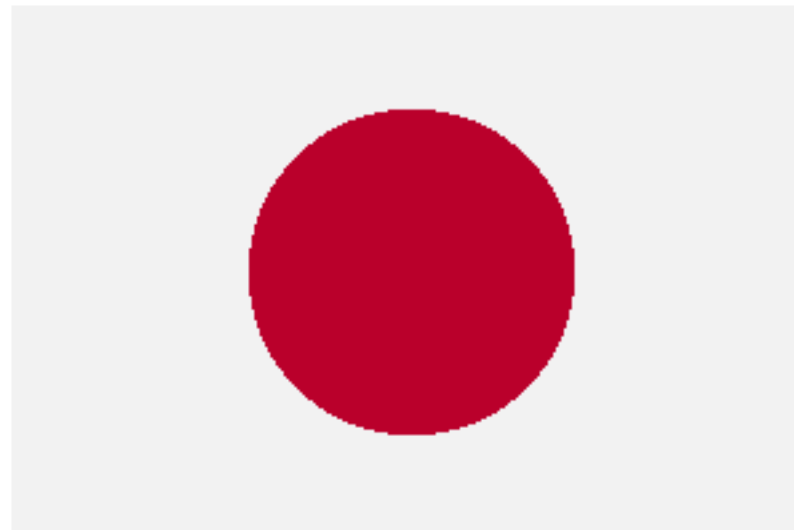


# Computer Graphics

Lecture 6

**General Perspective Cameras**
**Ray-Sphere Intersection**

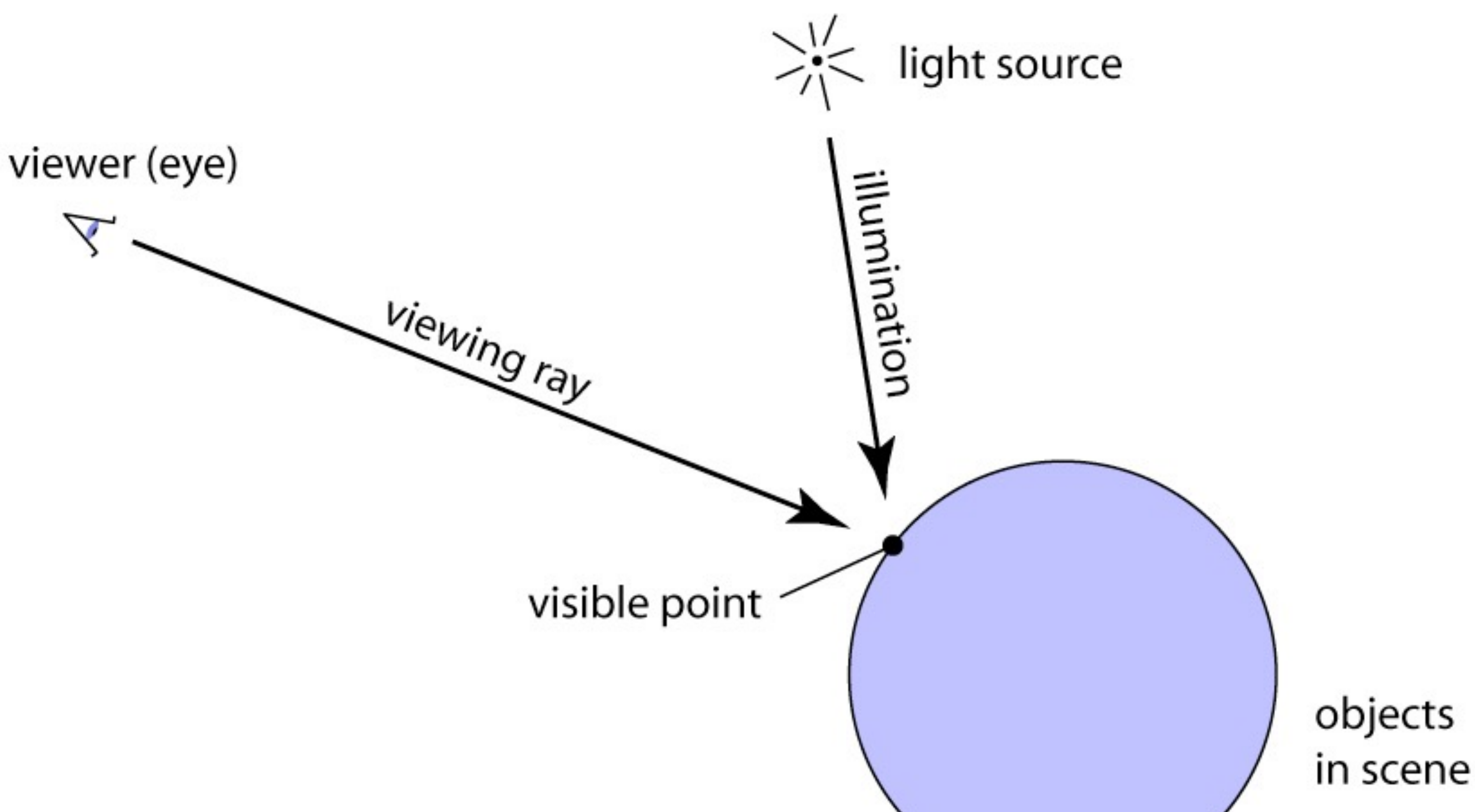
# Announcements

# Announcements

- Fill out the Github username form I sent out this morning

# Ray Tracing: Pseudocode

```
for each pixel:
    generate a viewing ray for the pixel
    find the closest object it intersects
    determine the color of the object
```

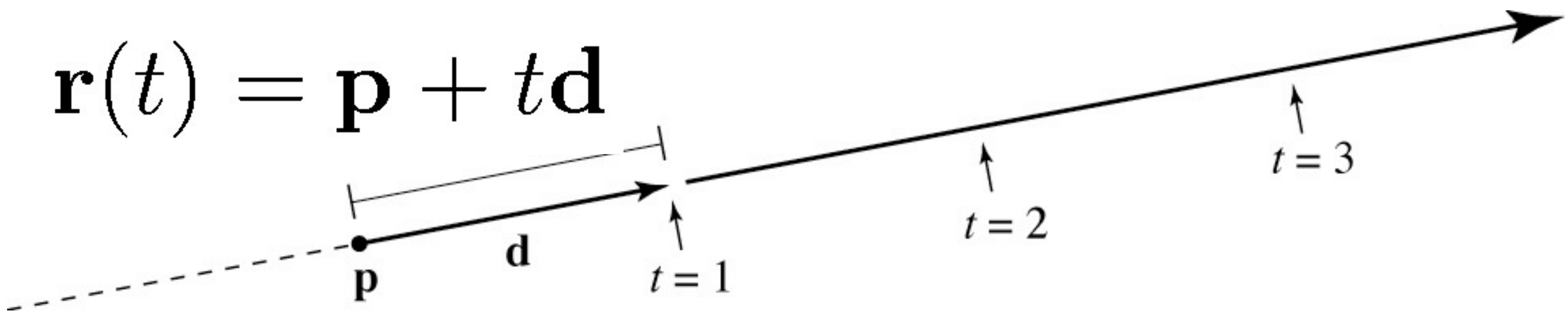


# A ray is half a line.

We'll describe rays using:

- An *origin* (**p**) where the ray begins
- A *direction* (**d**) in which the ray goes

$$\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$$

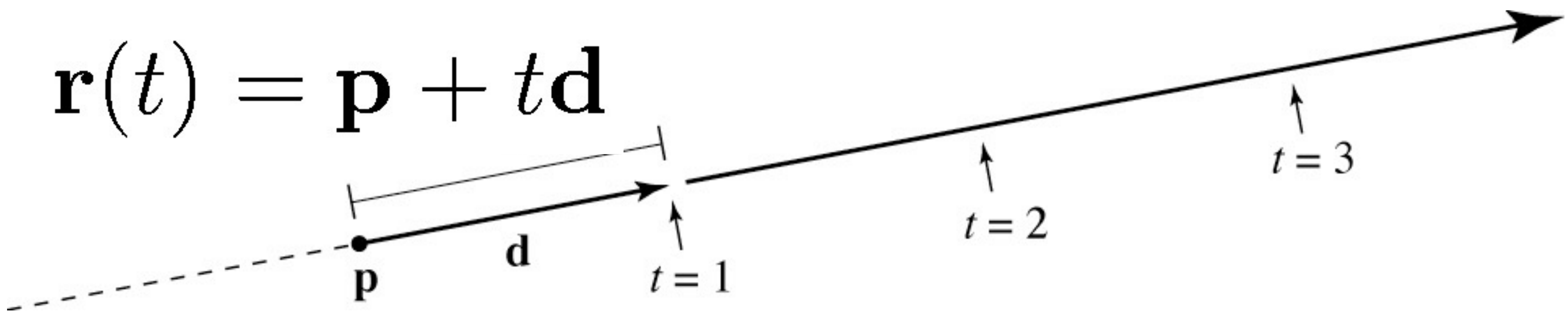


# A ray is half a line.

We'll describe rays using:

- An *origin* (**p**) where the ray begins
- A *direction* (**d**) in which the ray goes

$$\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$$



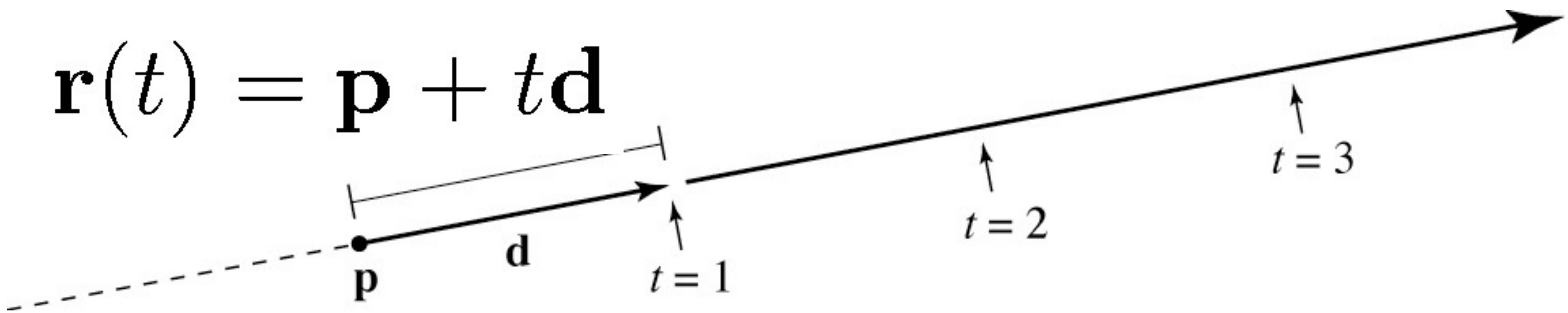

- This is a *parametric equation*: it **generates** points on the line

# A ray is half a line.

We'll describe rays using:

- An *origin* (**p**) where the ray begins
- A *direction* (**d**) in which the ray goes

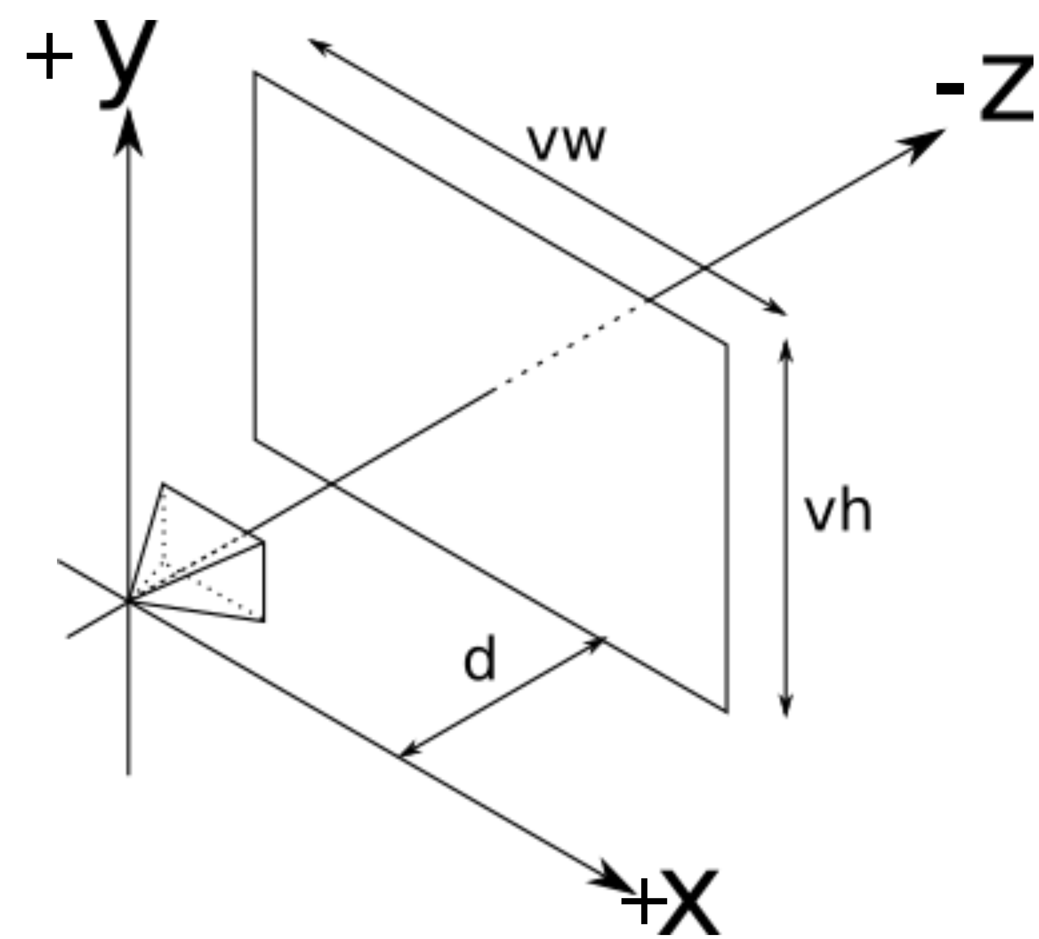
$$\mathbf{r}(t) = \mathbf{p} + t\mathbf{d}$$




- This is a *parametric equation*: it **generates** points on the line
- The set of points with *t > 0* gives all points on the ray

# A "canonical" camera

- Eye is at the origin (0, 0, 0)
- Looking down the **negative** z axis
- Viewport is aligned with the xy plane
- vh = vw = 1
- d = 1



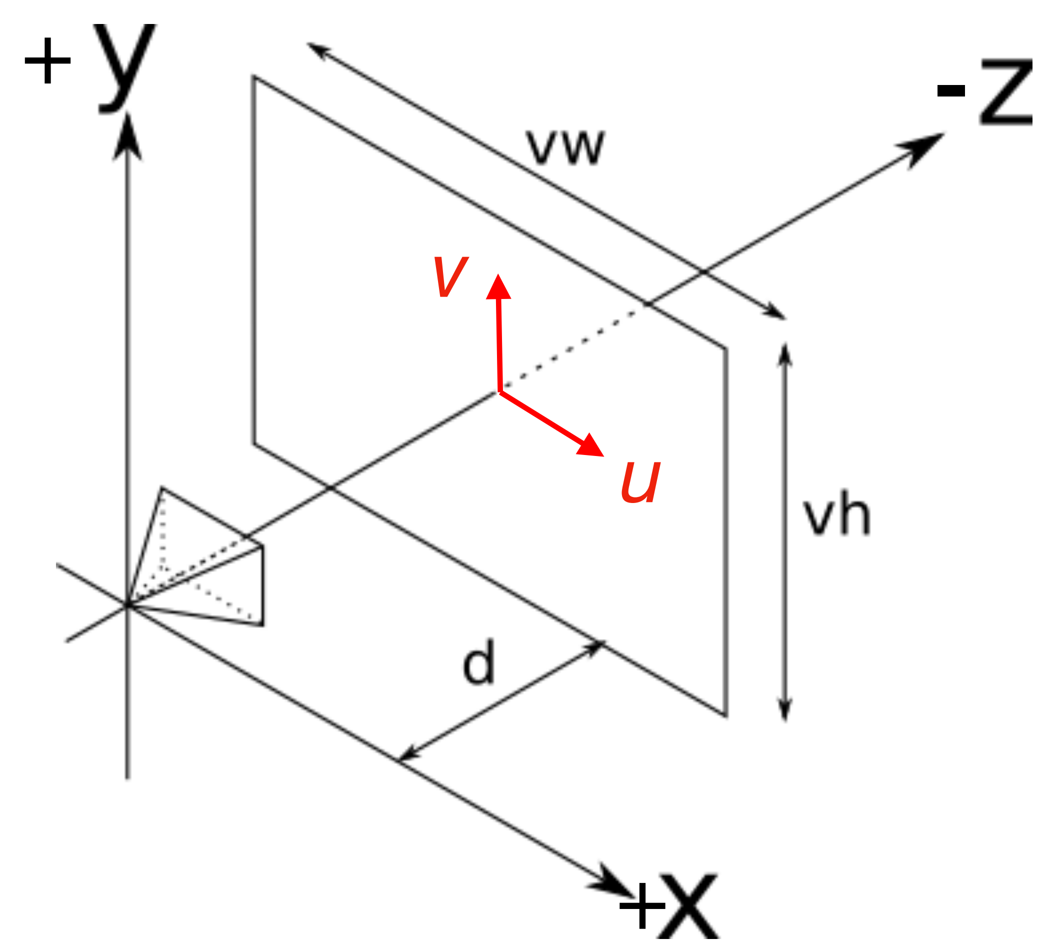
# Warmup: Viewing Rays

$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): (0, 0, 0)
Direction (**d**): (*u*, *v*, -1)

- Eye is at the origin (0, 0, 0)
- Looking down the **negative** z axis
- Viewport is aligned with the xy plane
- vh = vw = 1
- d = 1



# More General Cameras

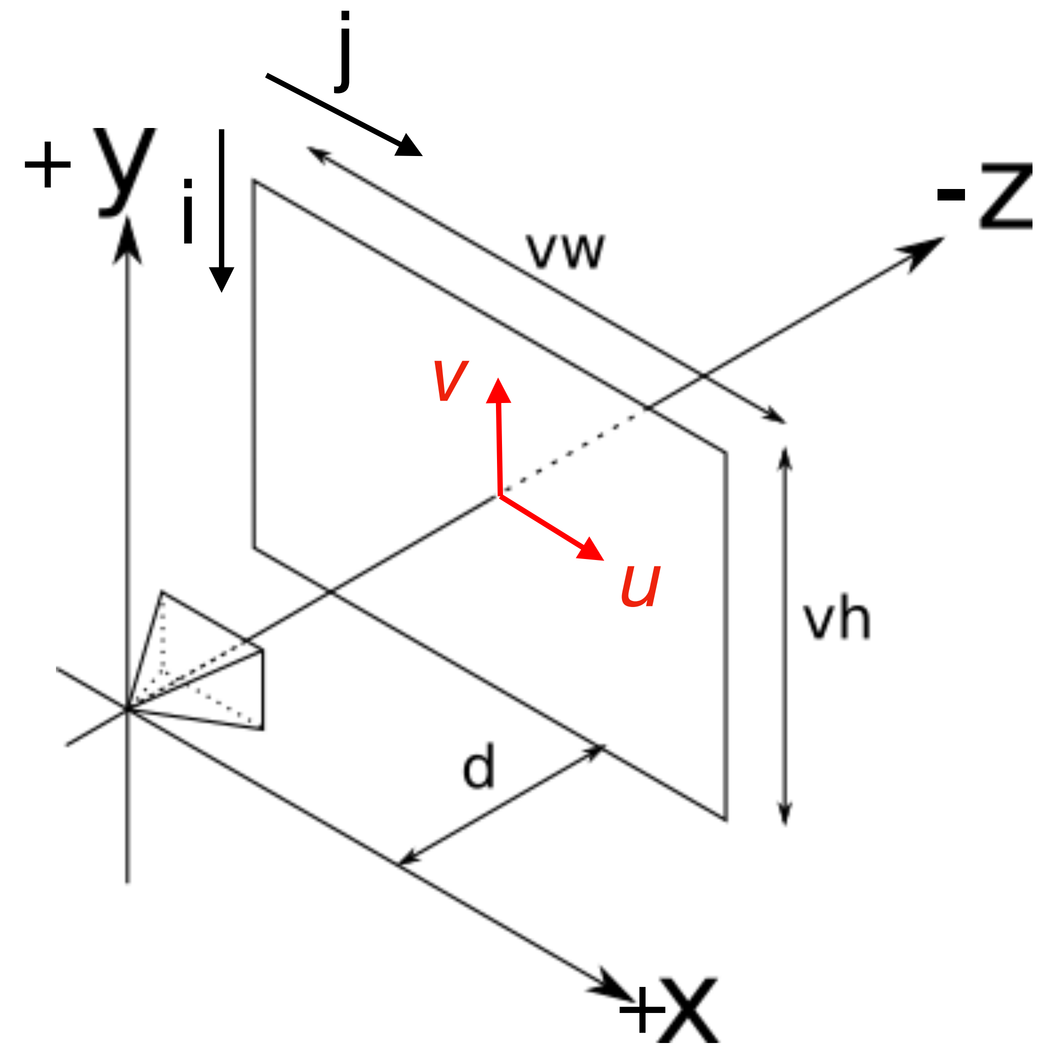
$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): (0, 0, 0)
Direction (**d**): (*u*, *v*, -1)

**Let's break some assumptions!**

- **d = 1**
- vh = vw = 1
- Eye is at the origin (0, 0, 0)
- Looking down the **negative** z axis



# More General Cameras

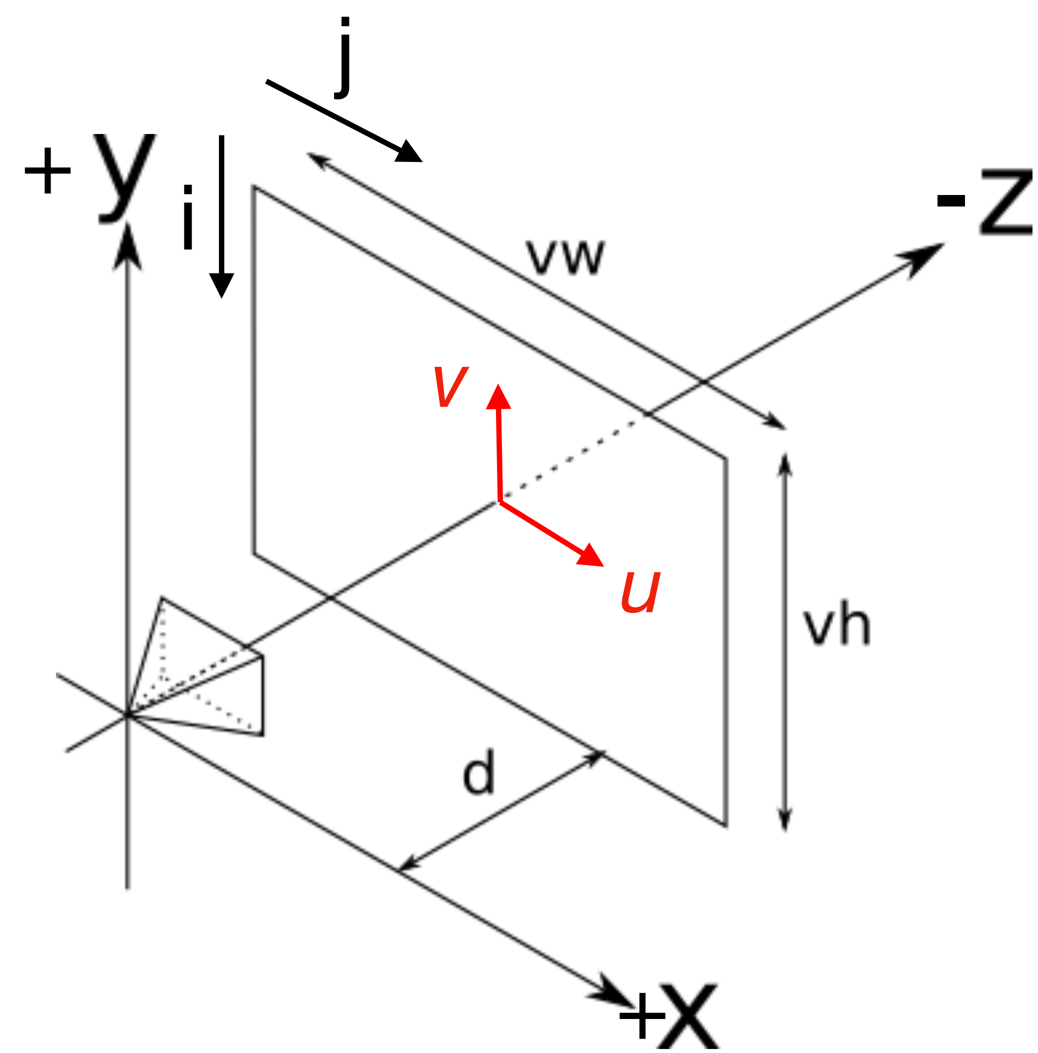
$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): (0, 0, 0)
Direction (**d**): ($u$, $v$, -d)

**Let's break some assumptions!**

- **d = 1**
- vh = vw = 1
- Eye is at the origin (0, 0, 0)
- Looking down the **negative** z axis



# More General Cameras

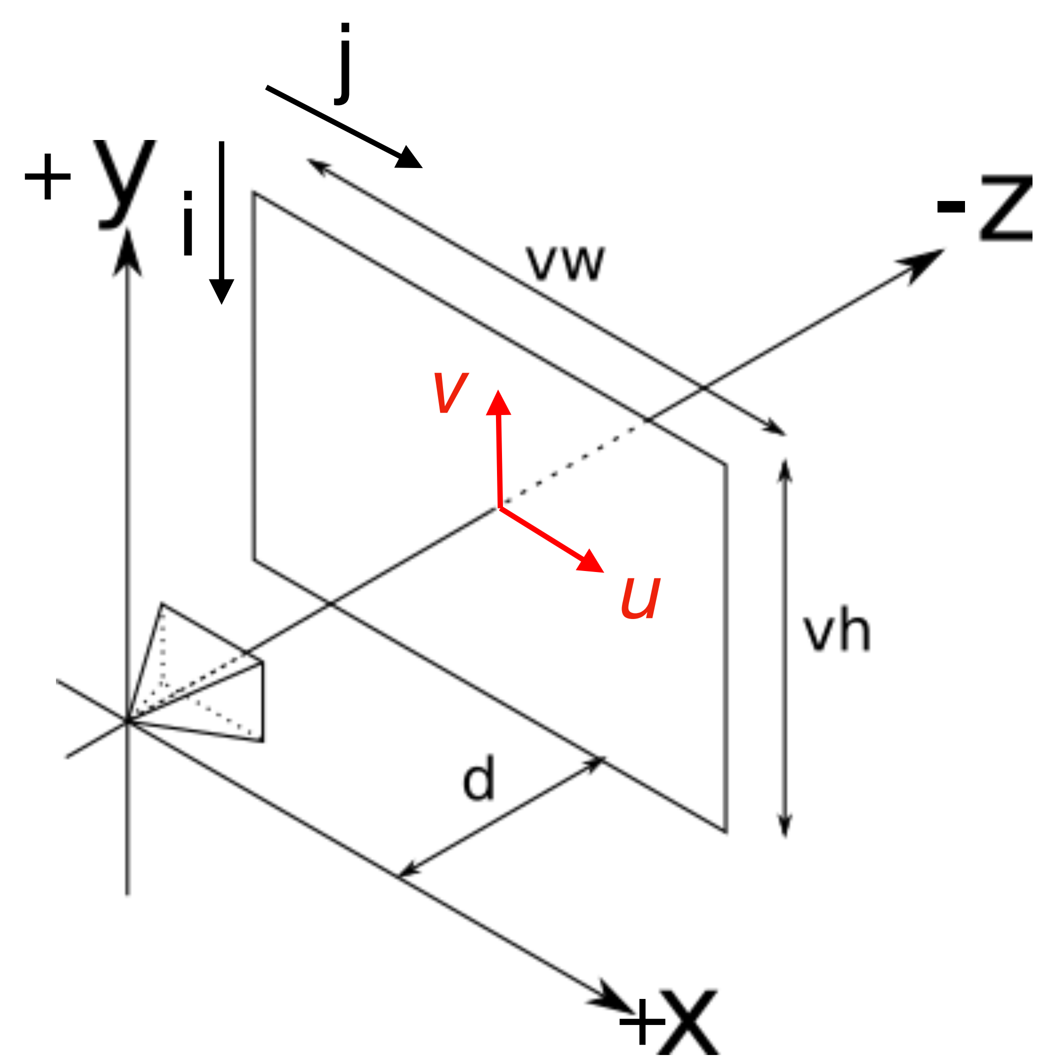
$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): (0, 0, 0)
Direction (**d**): (*u*, *v*, -1)

**Let's break some assumptions!**

- d = 1
- **vh = vw = 1**
- Eye is at the origin (0, 0, 0)
- Looking down the **negative** z axis



# More General Cameras

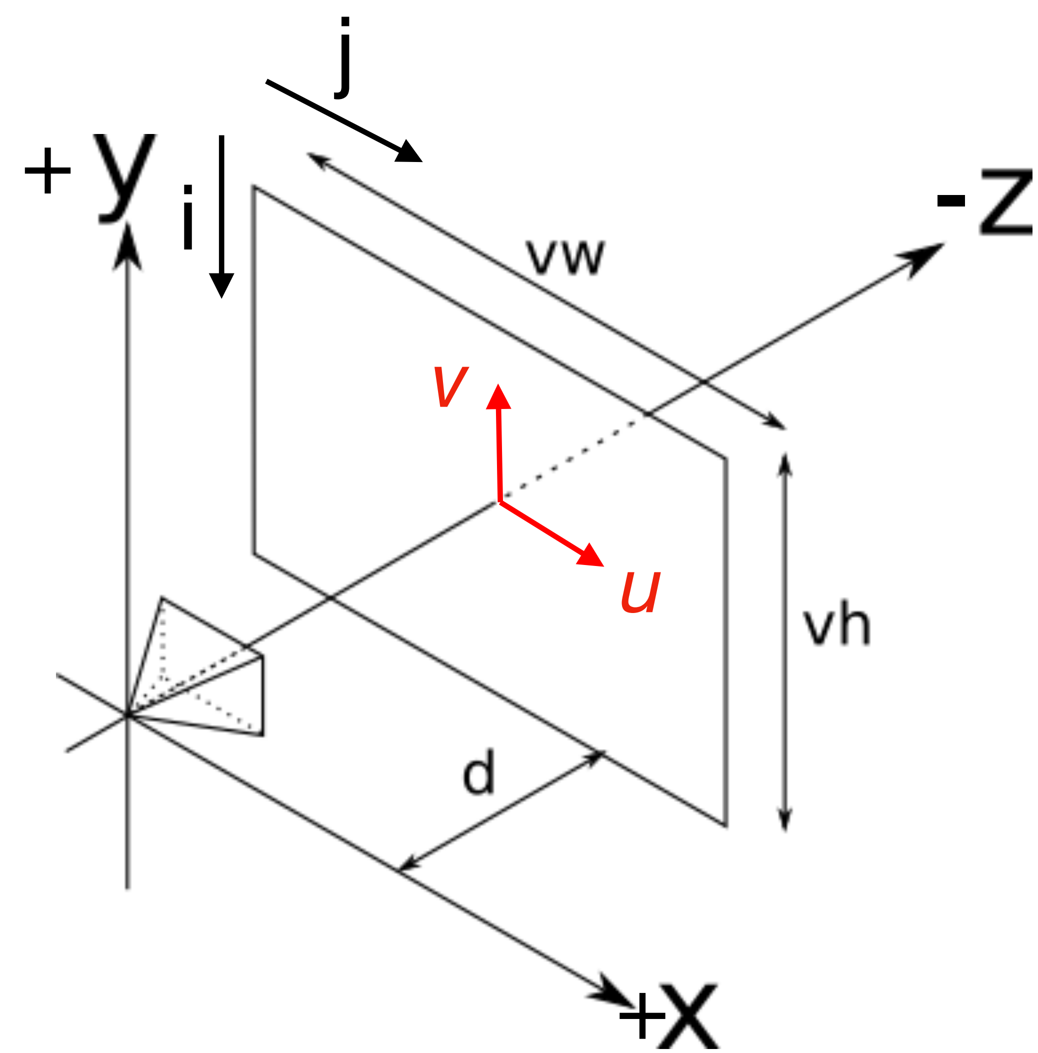
$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2} \quad * \text{ vw}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right) \quad * \text{ vh}$$

Origin (**p**): (0, 0, 0)
Direction (**d**): (*u*, *v*, -1)

**Let's break some assumptions!**

- d = 1
- **vh = vw = 1**
- Eye is at the origin (0, 0, 0)
- Looking down the **negative** z axis

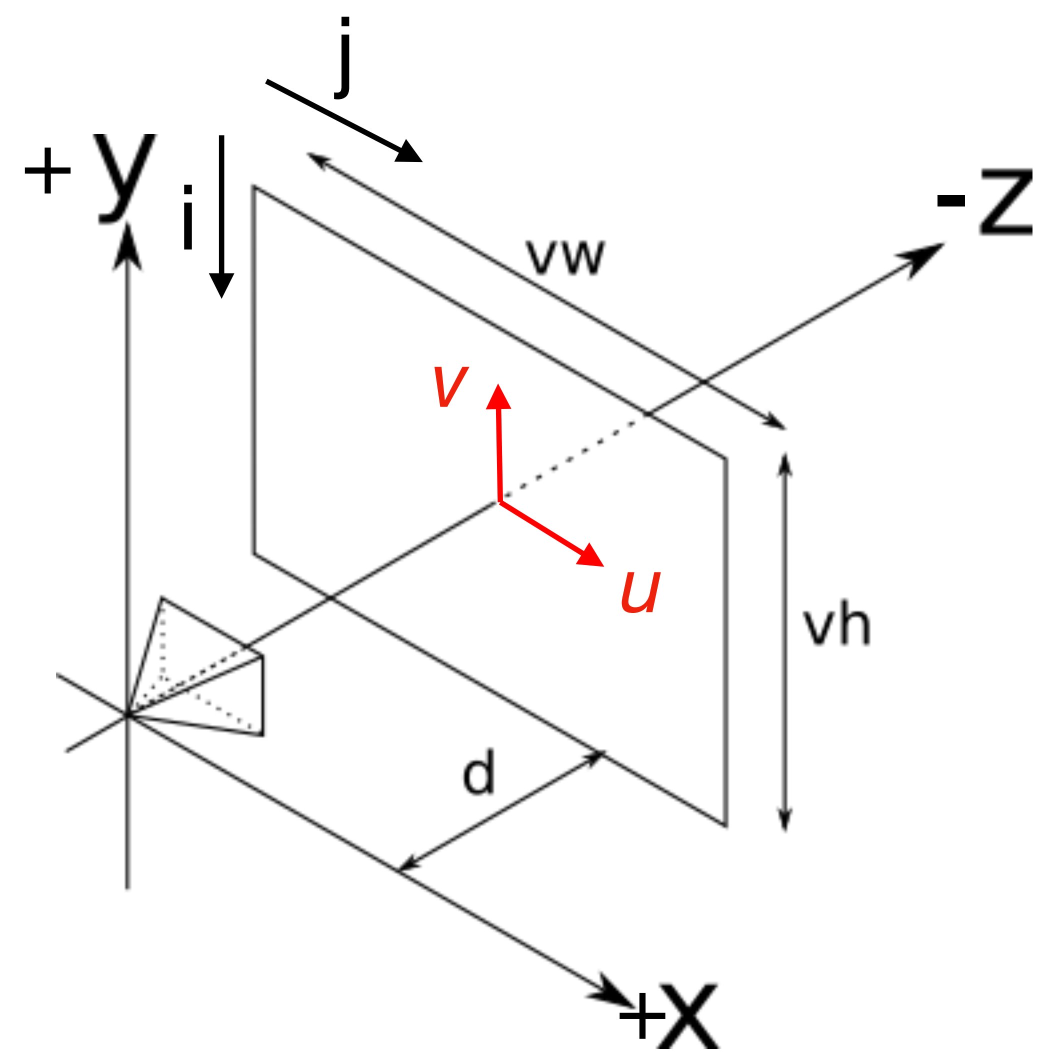


# More General Cameras

$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): (0, 0, 0)
Direction (**d**): (*u*, *v*, -1)




**Let's break some assumptions!**

- d = 1
- vh = vw = 1
- **Eye is at the origin (0, 0, 0)**
- Looking down the **negative** z axis

# More General Cameras

$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

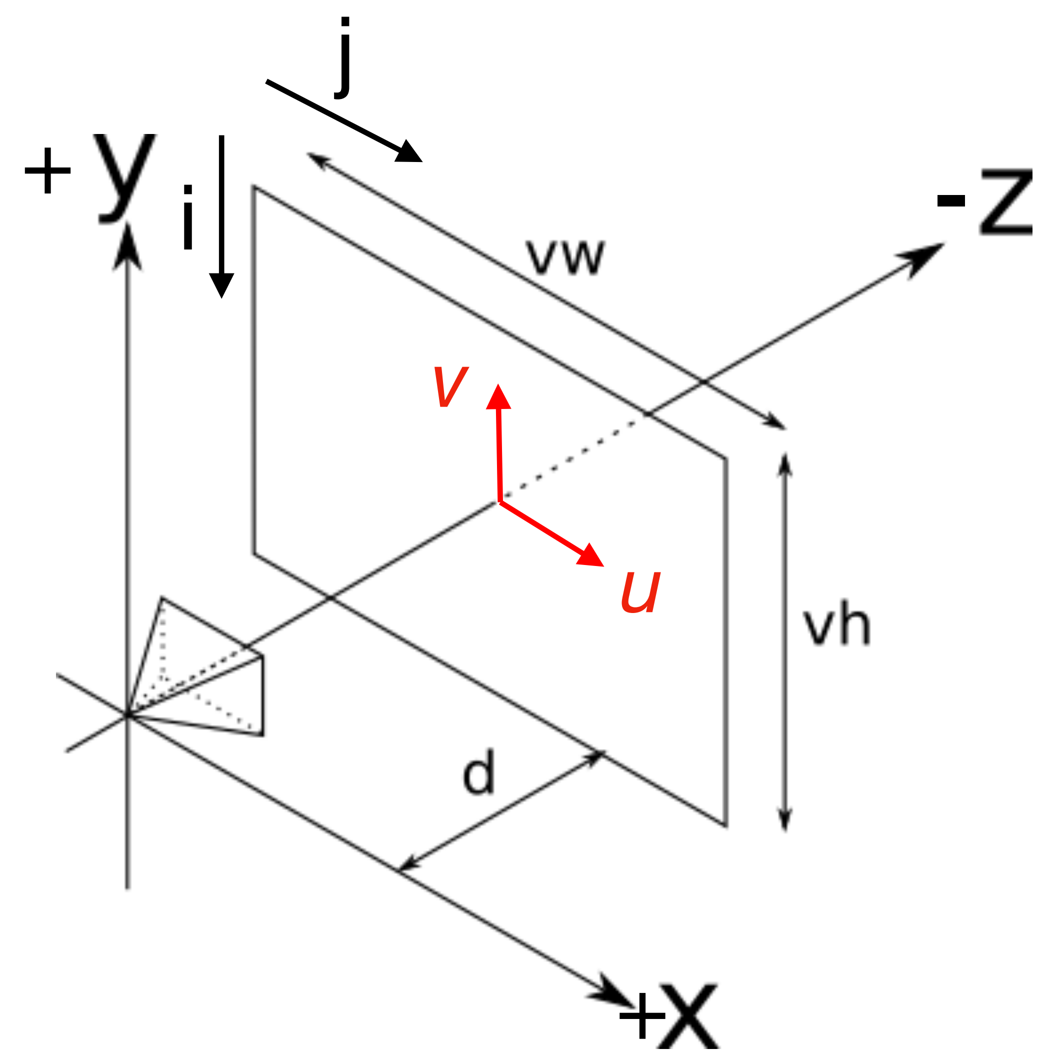
$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): $(e_x, e_y, e_z)$

Direction (**d**): ($u$, $v$, -1)

**Let's break some assumptions!**

- d = 1
- vh = vw = 1
- **Eye is at the origin (0, 0, 0)**
- Looking down the **negative** z axis

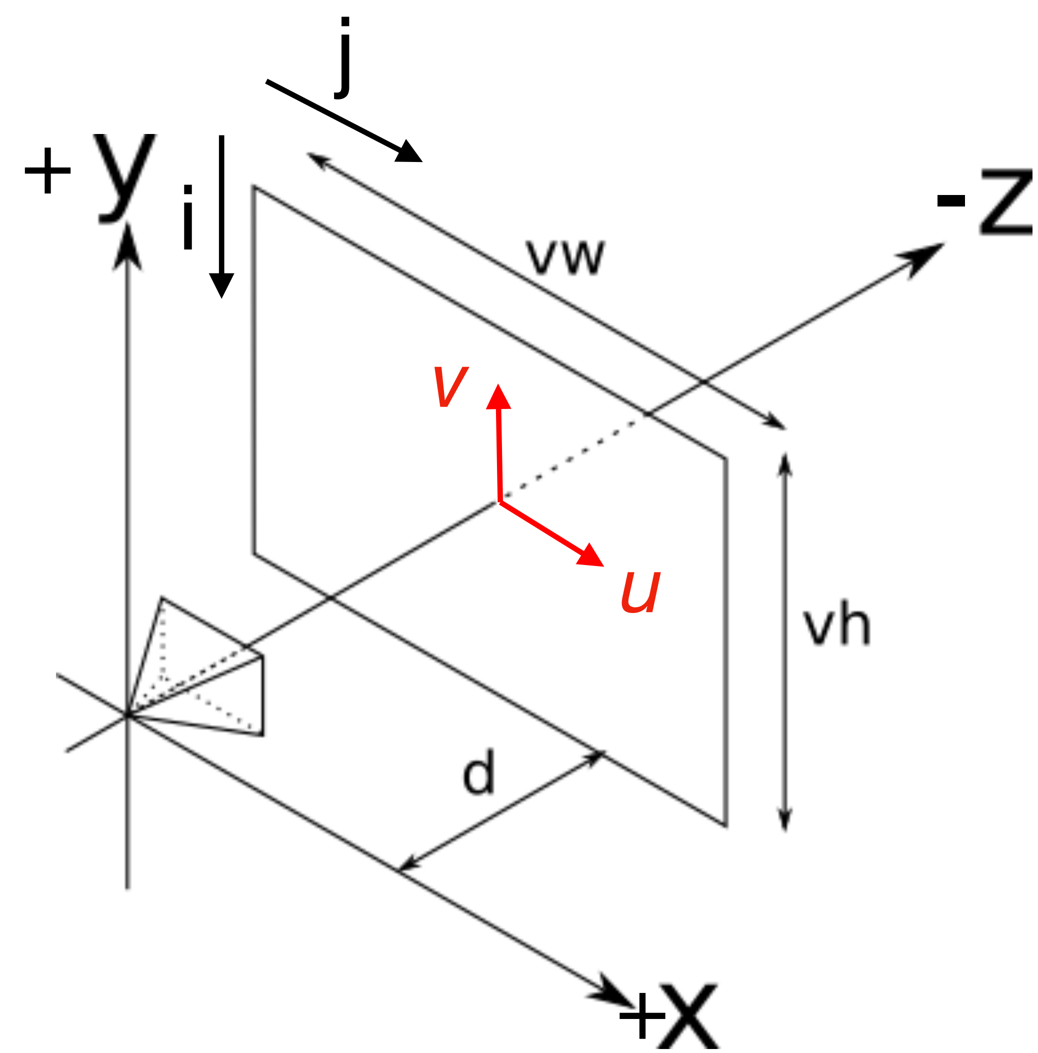


# More General Cameras

$$u = \frac{j - \frac{1}{2}}{W} - \frac{1}{2}$$

$$v = -\left(\frac{i - \frac{1}{2}}{H} - \frac{1}{2}\right)$$

Origin (**p**): (0, 0, 0)
Direction (**d**): (*u*, *v*, -1)




**Let's break some assumptions!**

- d = 1
- vh = vw = 1
- Eye is at the origin (0, 0, 0)
- **Looking down the negative z axis**

# Math Reminder

- Change of basis - see Section 2.4.5: Orthonormal Bases and Coordinate Frames

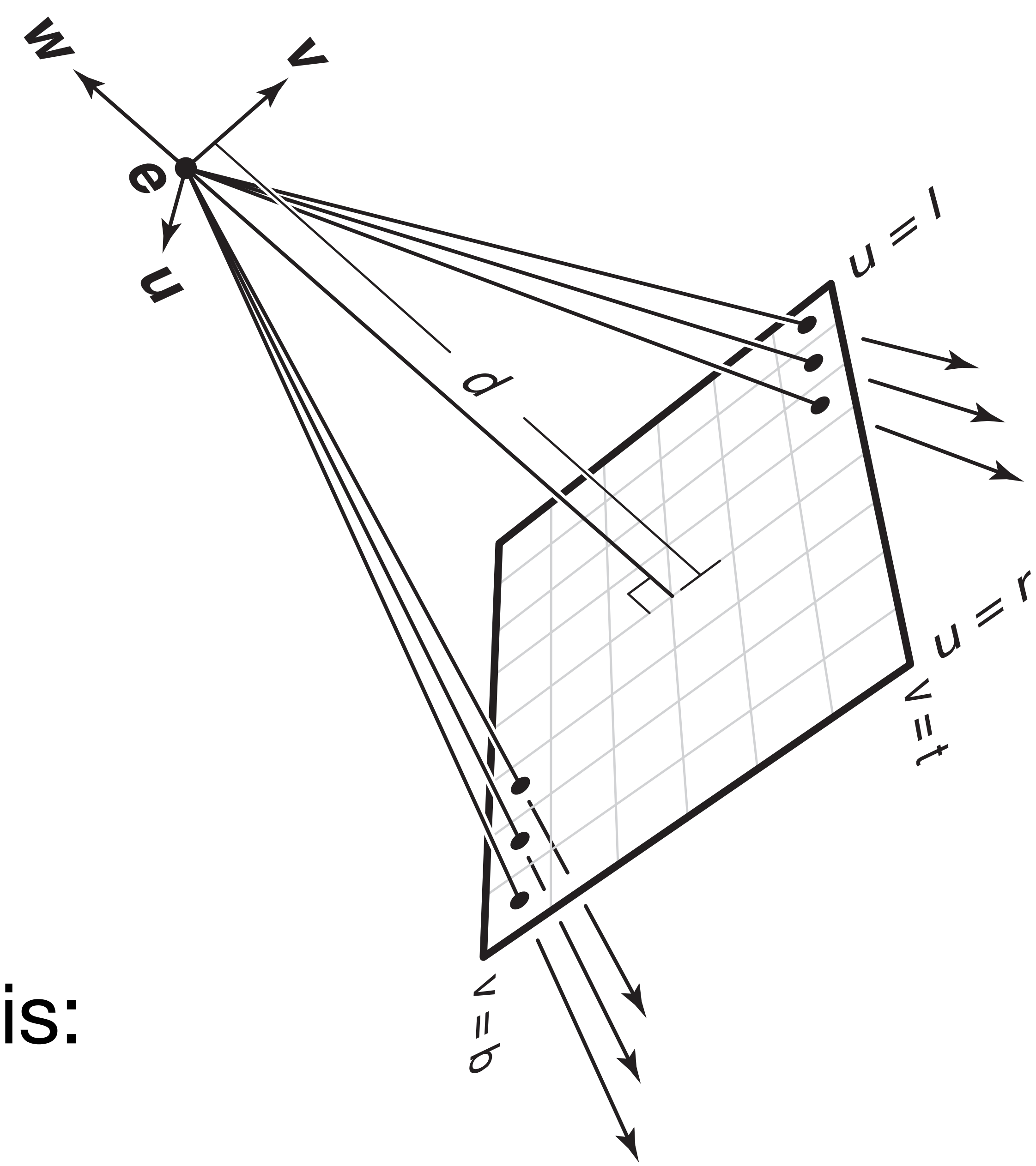
# If I want to put the camera somewhere else?

The camera's pose is defined by a **coordinate system:**

- **u** points right from the eye
- **v** points up from the eye
- **w** points back from the eye

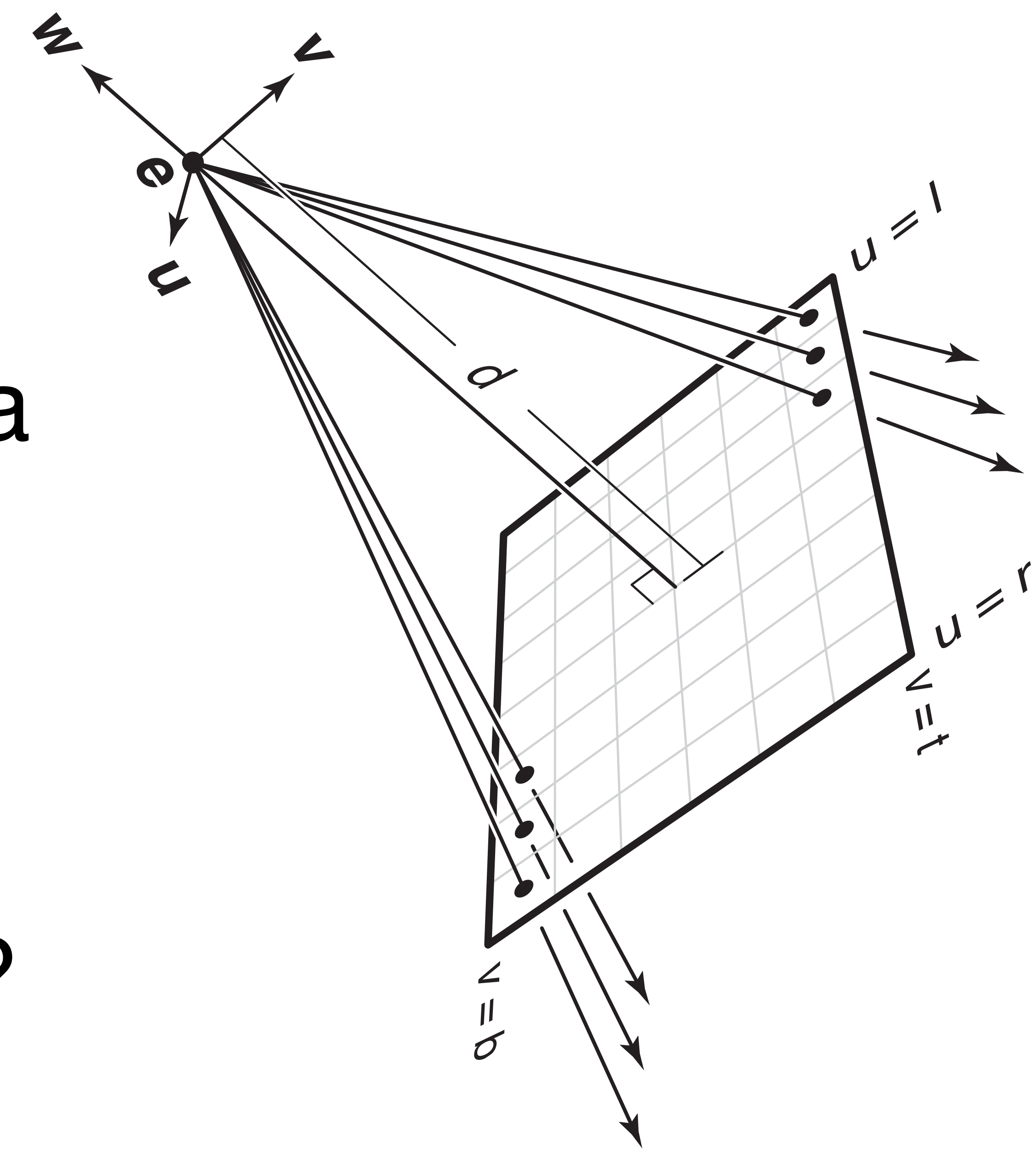
Given this, we can generate a viewing ray as follows:

1. Turn (i,j) into *u, v* as before
2. Viewing ray in (x, y, z) world is:
   origin = eye
   direction = $u * \mathbf{u} + v * \mathbf{v} + -d * \mathbf{w}$



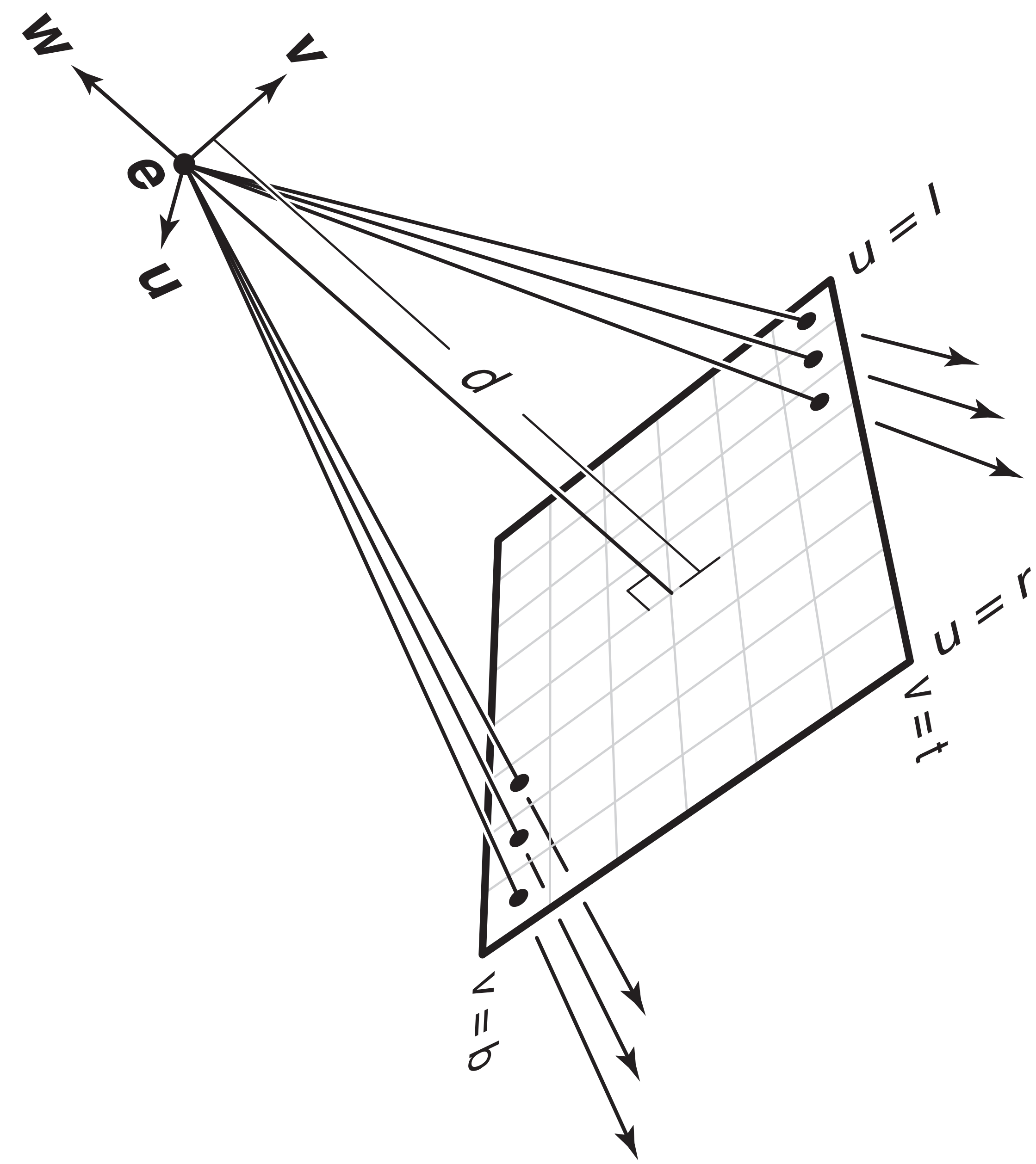
# Creating A Camera Basis

- **e, u, v, w** : simple math, but not very intuitive
- Can we position a camera based on:
  - eye
  - view direction or point?



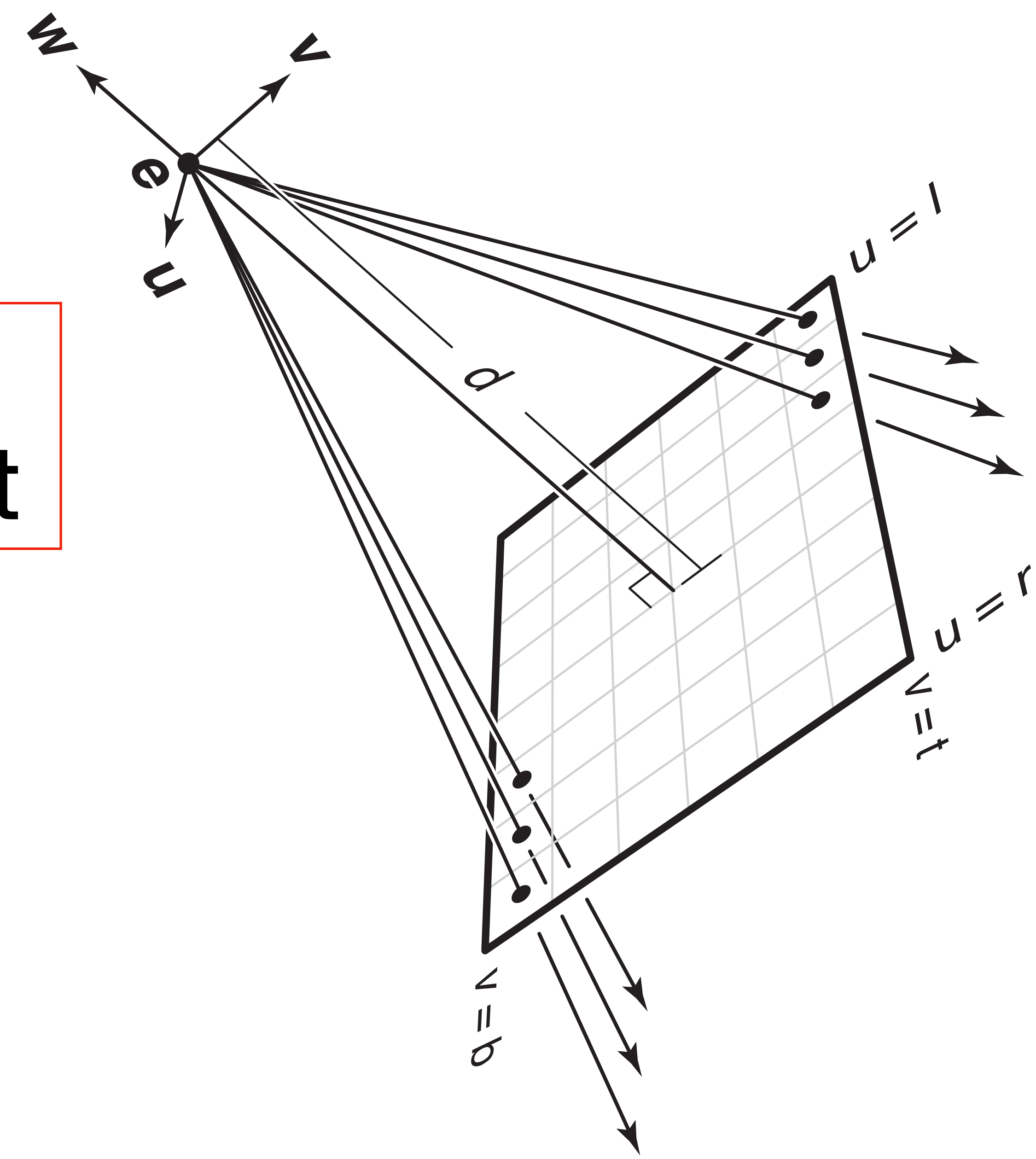
# Creating A Camera Basis

- **eye** - position of eye
- **view** direction - direction camera is looking
- "**up**" vector - points "up" in the scene, but not necessarily in image space.



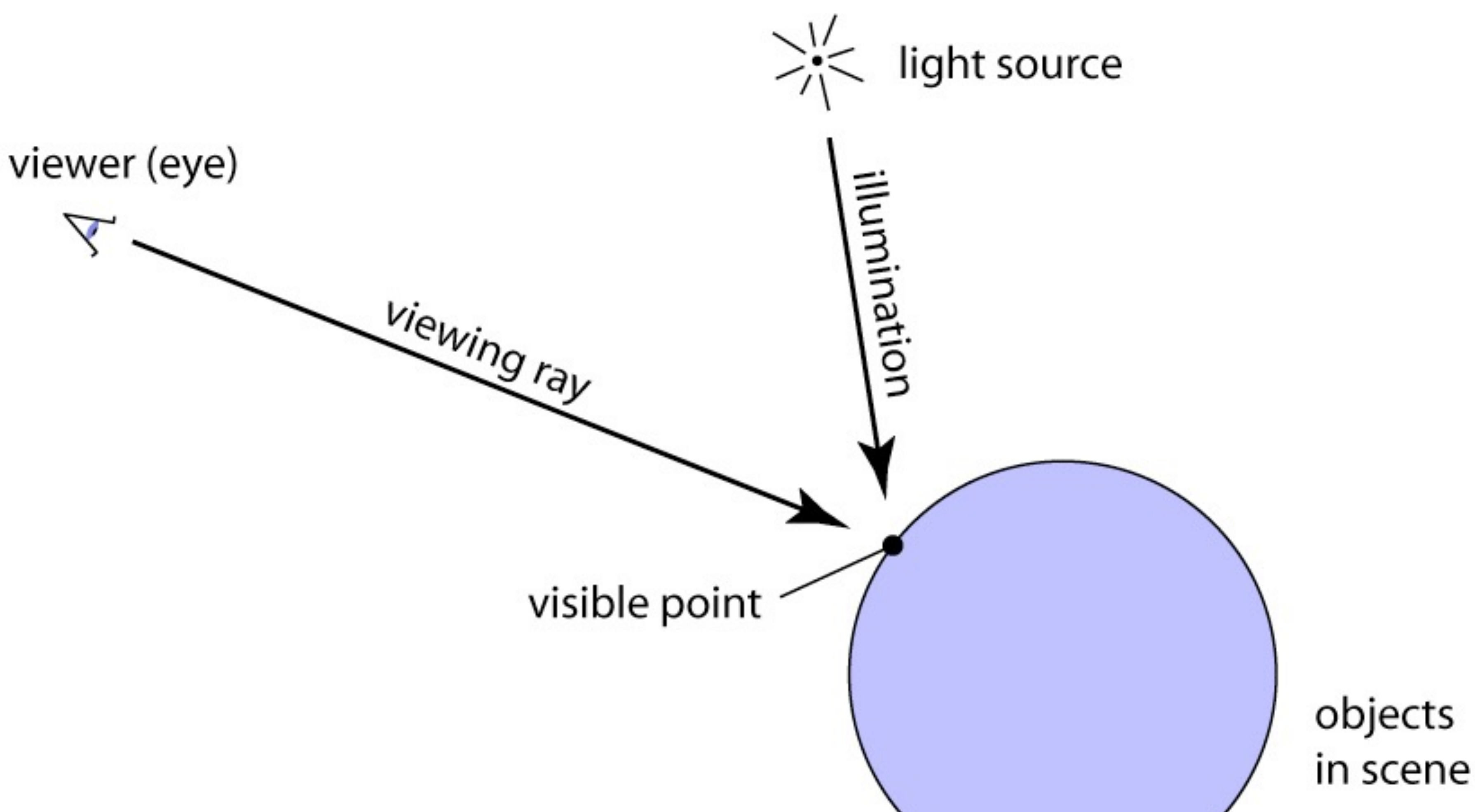
# Exercise 2

- **eye** - position of eye
- **at** - position of a point the camera is looking straight at
- "**up**" vector - points "up" in the scene, but not necessarily in image space.



# Ray Tracing: Pseudocode

```
for each pixel:
    generate a viewing ray for the pixel
    find the closest object it intersects
    determine the color of the object
```



# Implicit vs Parametric

- Implicit equations: a property true at all points
  - e.g., $ax + by + c = 0$, for a line
- Parametric equations: use a free parameter variable to *generate* all points:
  - e.g., $r(t) = \mathbf{p} + t\mathbf{d}$, for a line
- Intersecting parametric with implicit is usually cleanest.

# Ray-Sphere intersection

- For now, assume unit sphere centered at the origin. See 4.4.1 for general derivation.

# Ray-Sphere intersection

- For now, assume unit sphere centered at the origin. See 4.4.1 for general derivation.

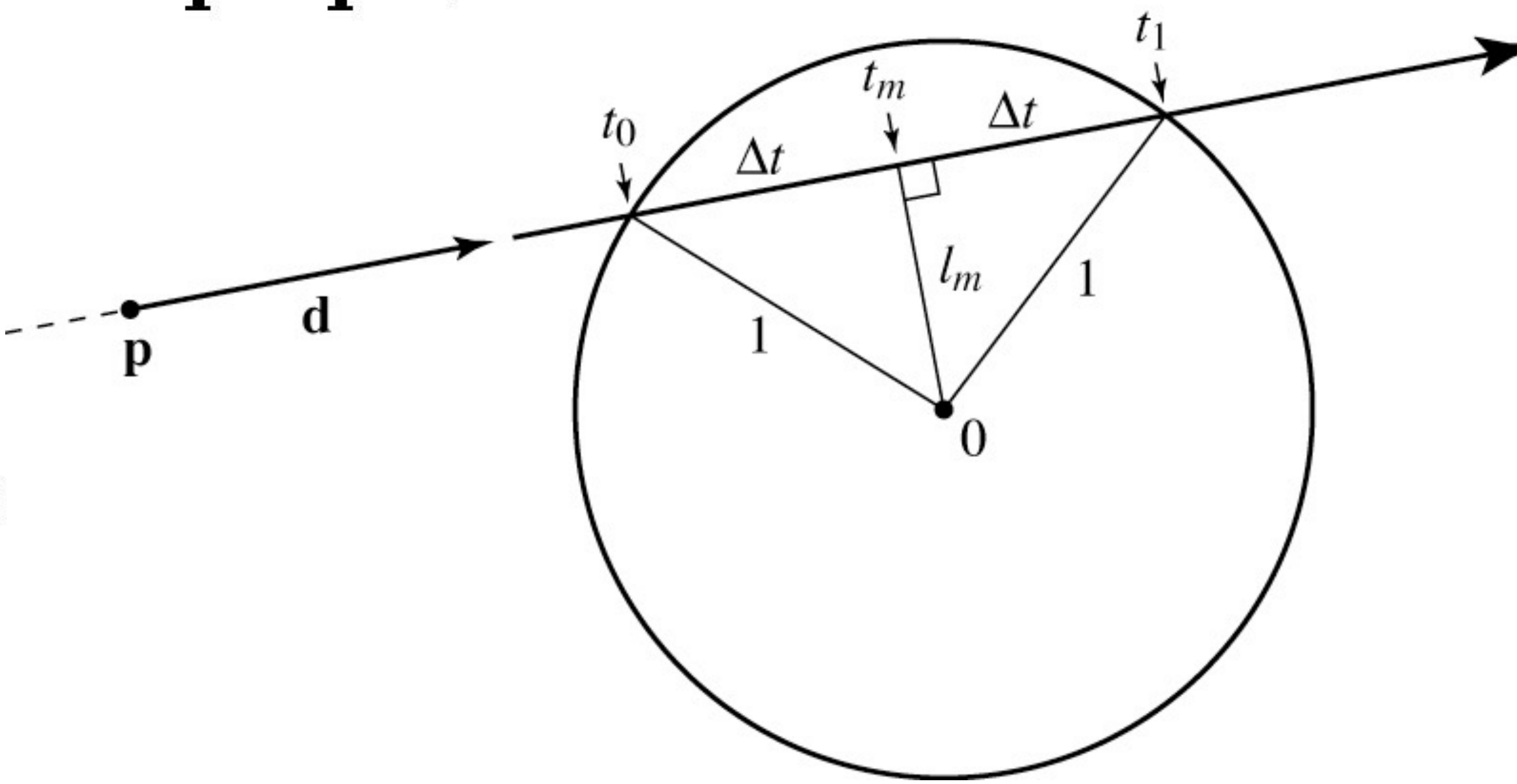
$$t = \frac{-\mathbf{d} \cdot \mathbf{p} \pm \sqrt{(\mathbf{d} \cdot \mathbf{p})^2 - (\mathbf{d} \cdot \mathbf{d})(\mathbf{p} \cdot \mathbf{p} - 1)}}{\mathbf{d} \cdot \mathbf{d}}$$

If **d** is unit-length:

$$t = -\mathbf{d} \cdot \mathbf{p} \pm \sqrt{(\mathbf{d} \cdot \mathbf{p})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$

# Ray-Sphere intersection: Geometric Intuition

$$t = -\mathbf{d} \cdot \mathbf{p} \pm \sqrt{(\mathbf{d} \cdot \mathbf{p})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$




$$t_m = -\mathbf{p} \cdot \mathbf{d}$$

$$l_m^2 = \mathbf{p} \cdot \mathbf{p} - (\mathbf{p} \cdot \mathbf{d})^2$$

$$\Delta t = \sqrt{1 - l_m^2}$$

$$= \sqrt{(\mathbf{p} \cdot \mathbf{d})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$

$$t_{0,1} = t_m \pm \Delta t = -\mathbf{p} \cdot \mathbf{d} \pm \sqrt{(\mathbf{p} \cdot \mathbf{d})^2 - \mathbf{p} \cdot \mathbf{p} + 1}$$