CSE4203: Computer Graphics
Chapter – 4 (part - C)
Ray Tracing

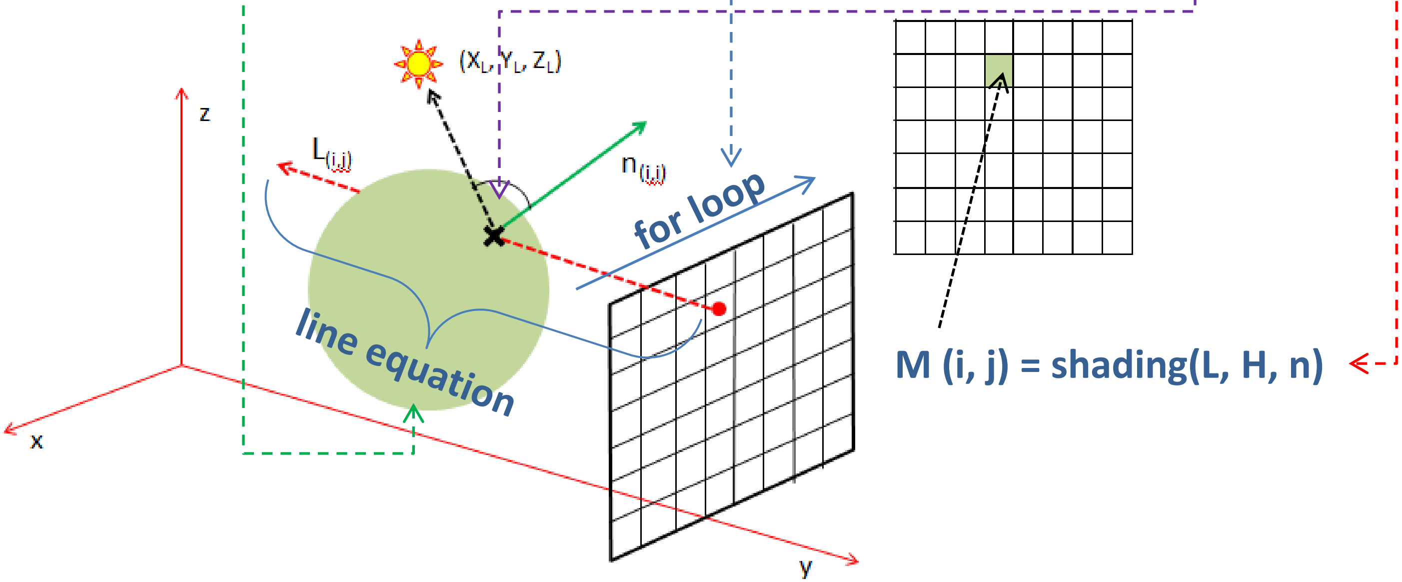
Mohammad Imrul Jubair

Outline

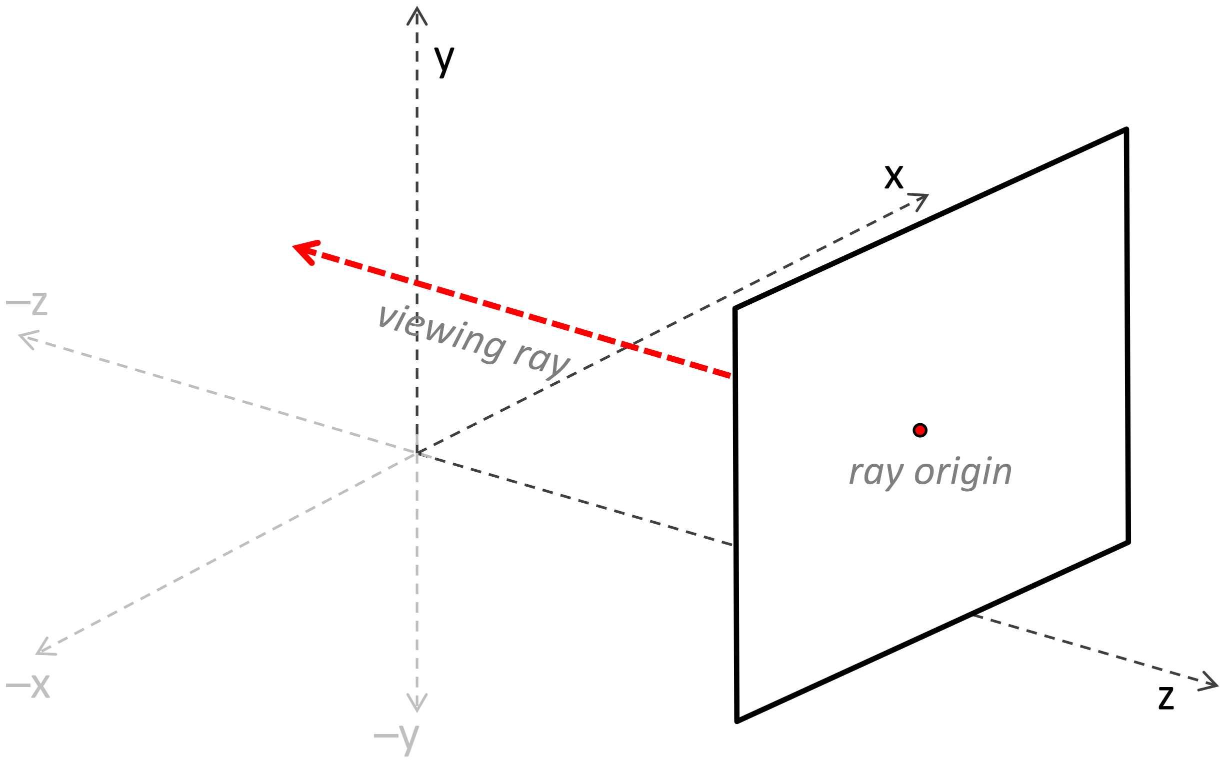
– Ray-tracing
– Camera Frame
– Image Plane and Raster Image
– Computing Viewing Rays
– Ray-sphere Intersection
– Shading

Ray-Tracing Algorithm

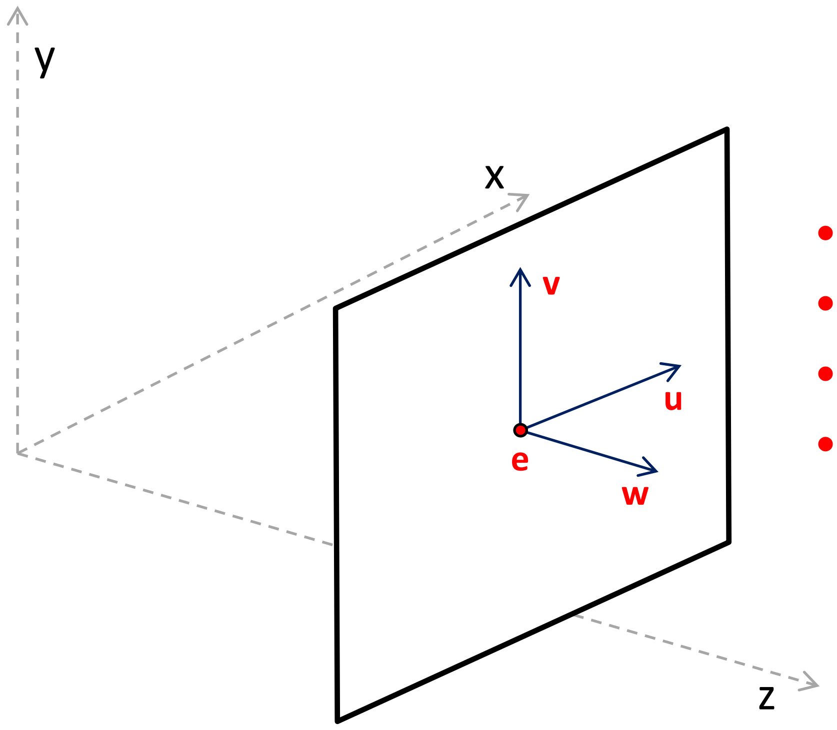
- **for each pixel do:**
  - compute viewing ray
  - find first object hit by ray and its surface normal **n**
  - set pixel color computed from hit point, light, and **n**



Camera Frame (1/11)



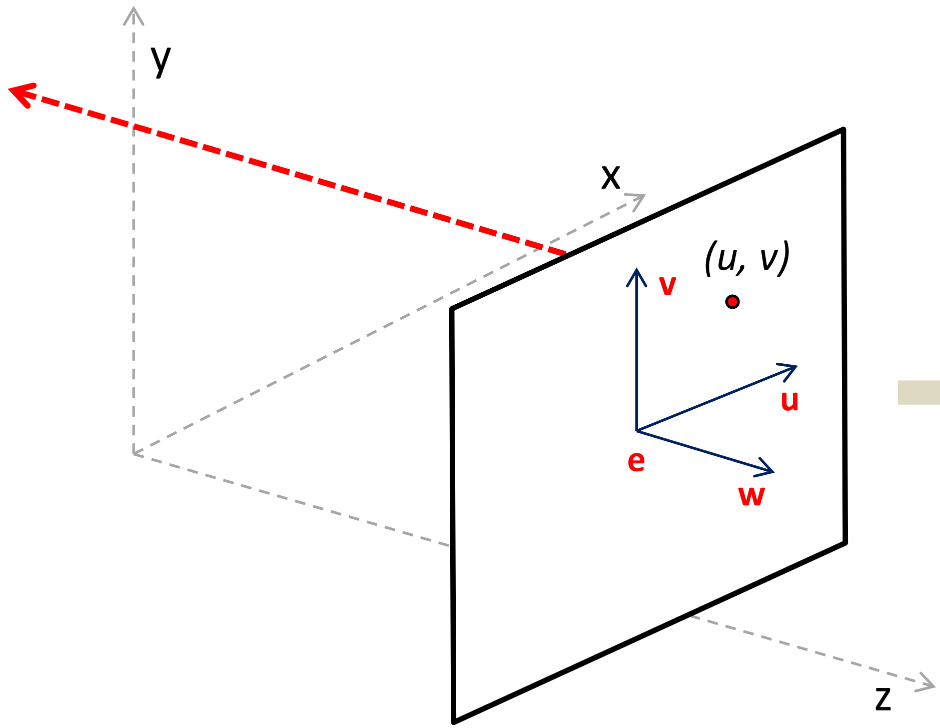
Camera Frame (2/11)

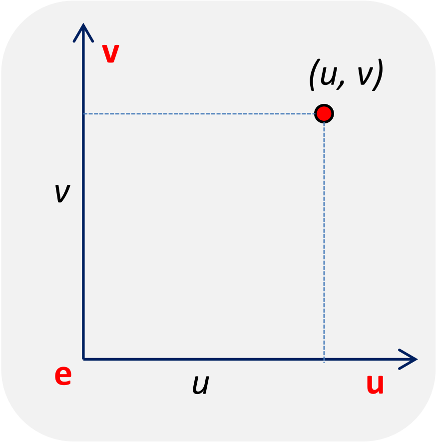


- **e**: *viewpoint*
- **u**: *right*
- **v**: *up*
- **w**: *back*

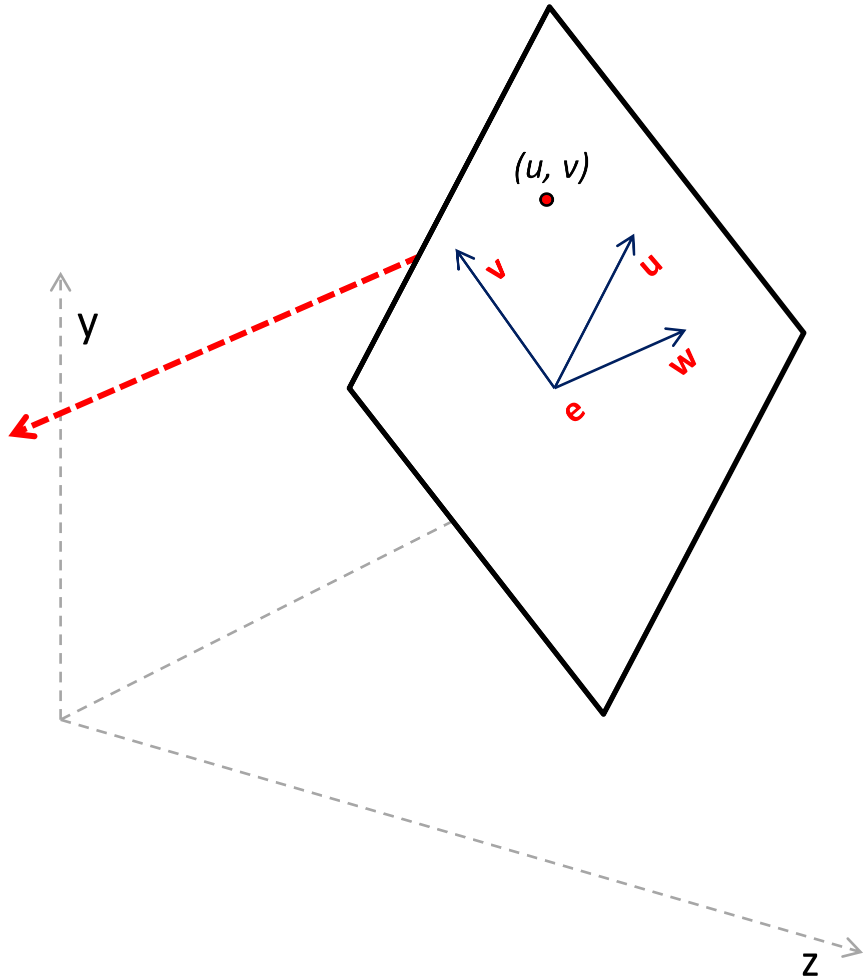
Camera Frame (3/11)

- ray origin = **e** + u **u** + v **v**
  - ray direction = −**w**

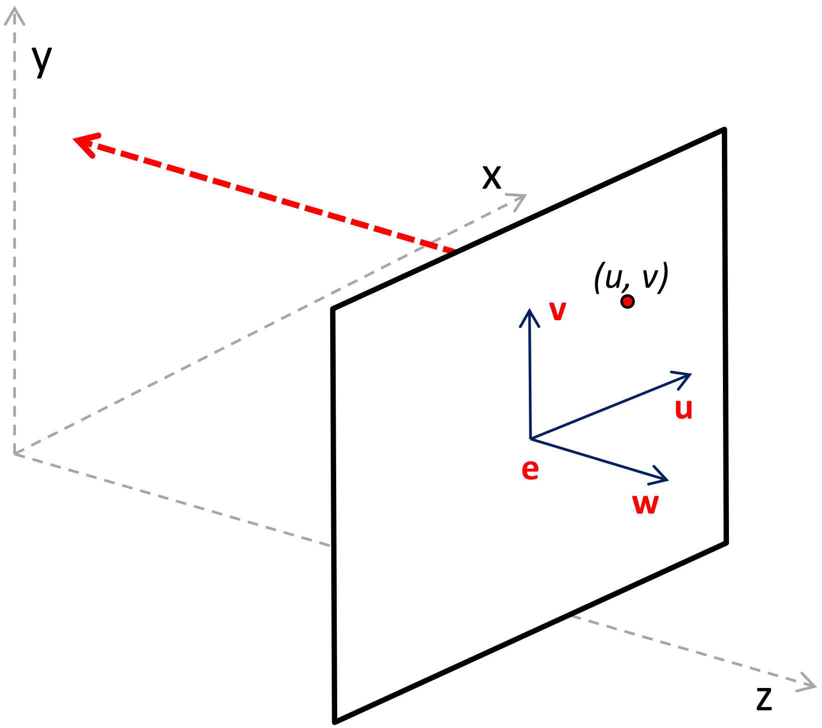




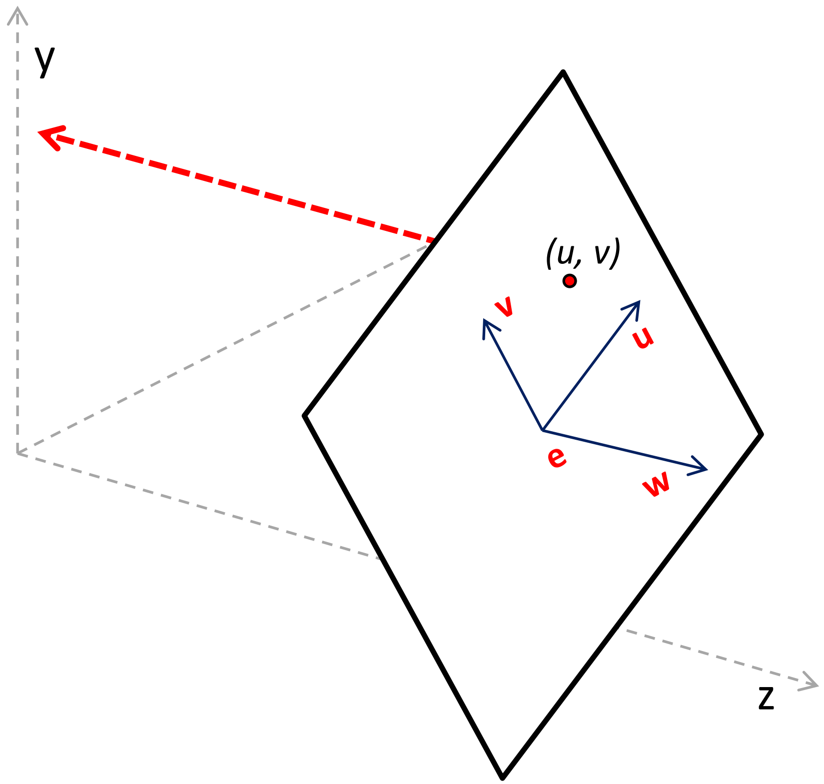
Camera Frame (4/11)



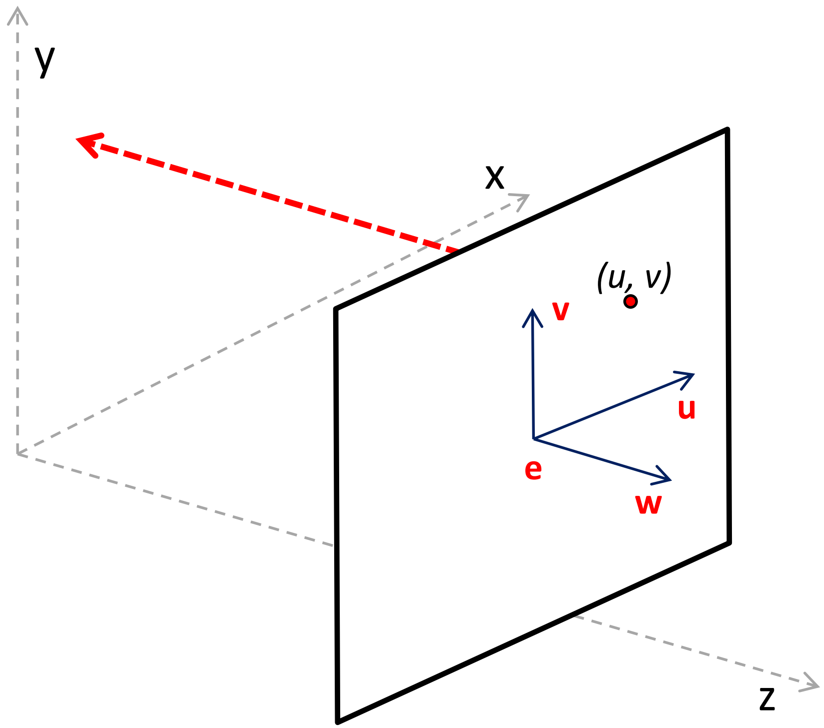
Camera Frame (5/11)



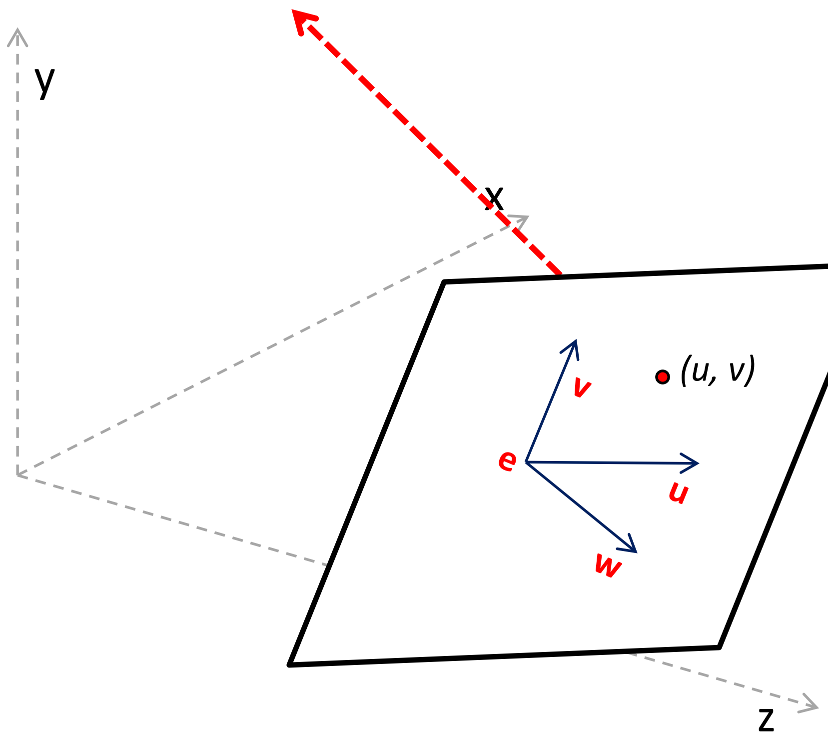
Camera Frame (6/11)



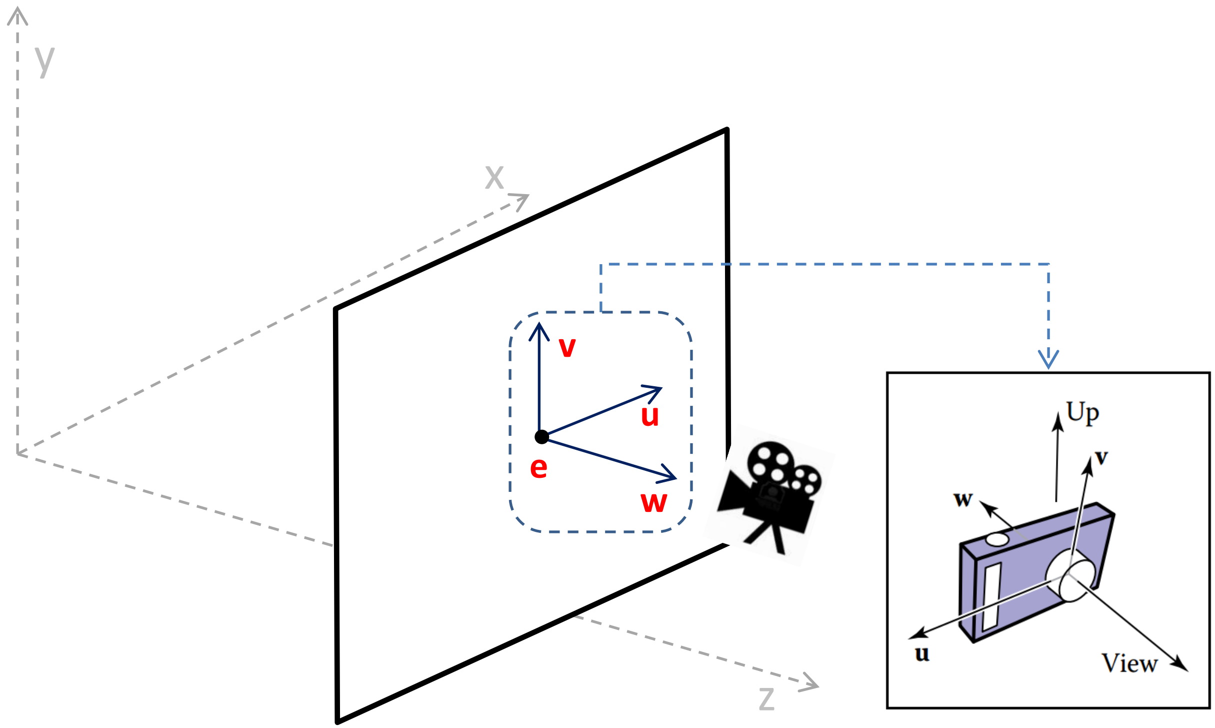
Camera Frame (7/11)



Camera Frame (8/11)

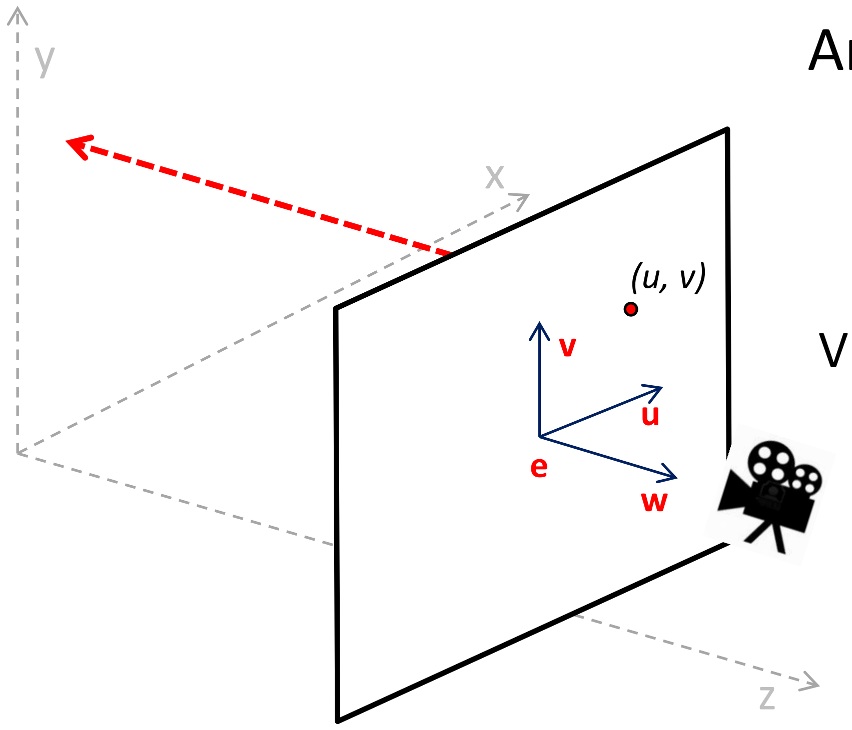


Camera Frame (9/11)



Camera Frame (10/11)

Camera frame: (Camera coordinate)



e: viewpoint

And 3 basis vectors:

- **u**: right
- **v**: up
- **w**: back

View direction **(−w)**

Camera Frame (11/11)

Orthographic:

- ray direction = $-\mathbf{w}$
- ray origin = $\mathbf{e} + u\,\mathbf{u} + v\,\mathbf{v}$

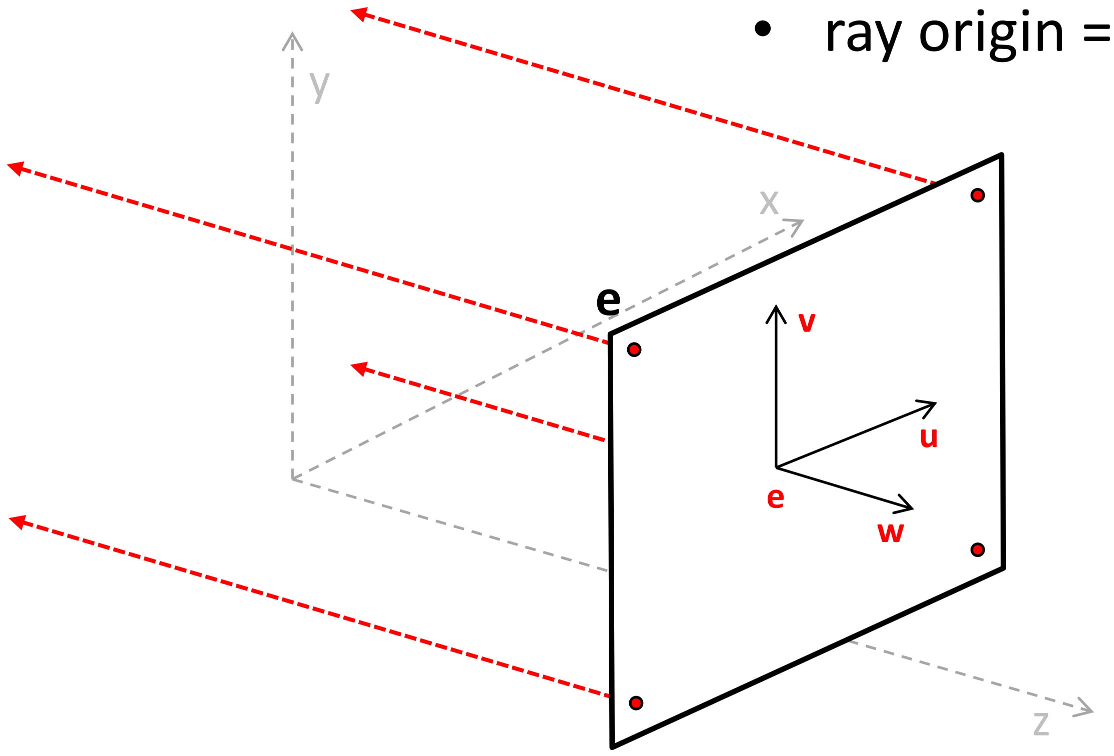


Image Plane (1/4)

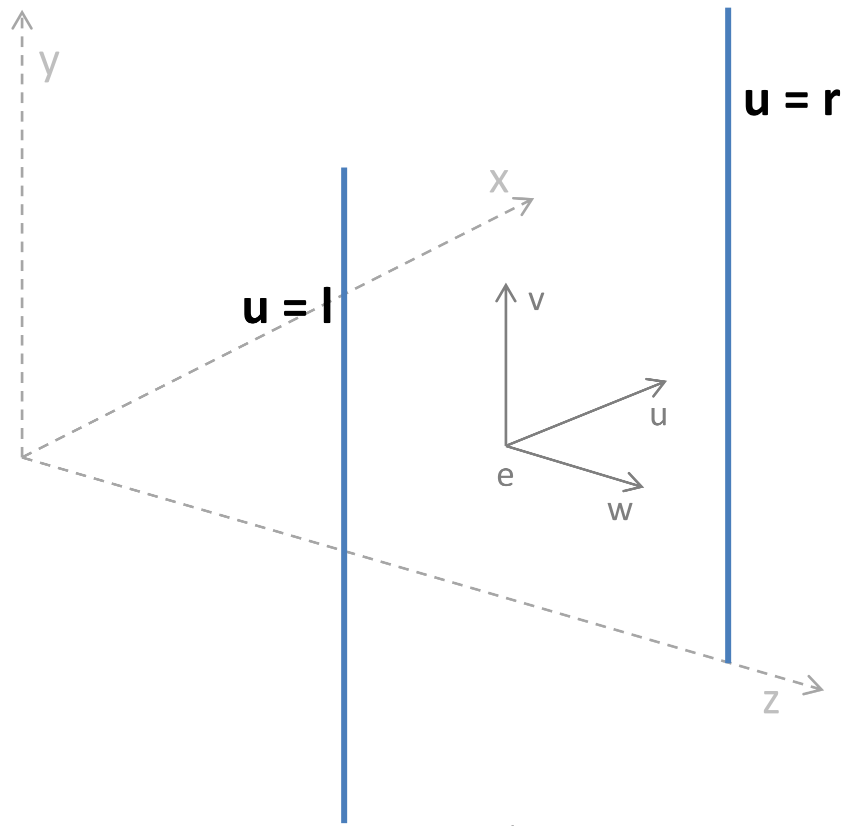


Image Plane (2/4)

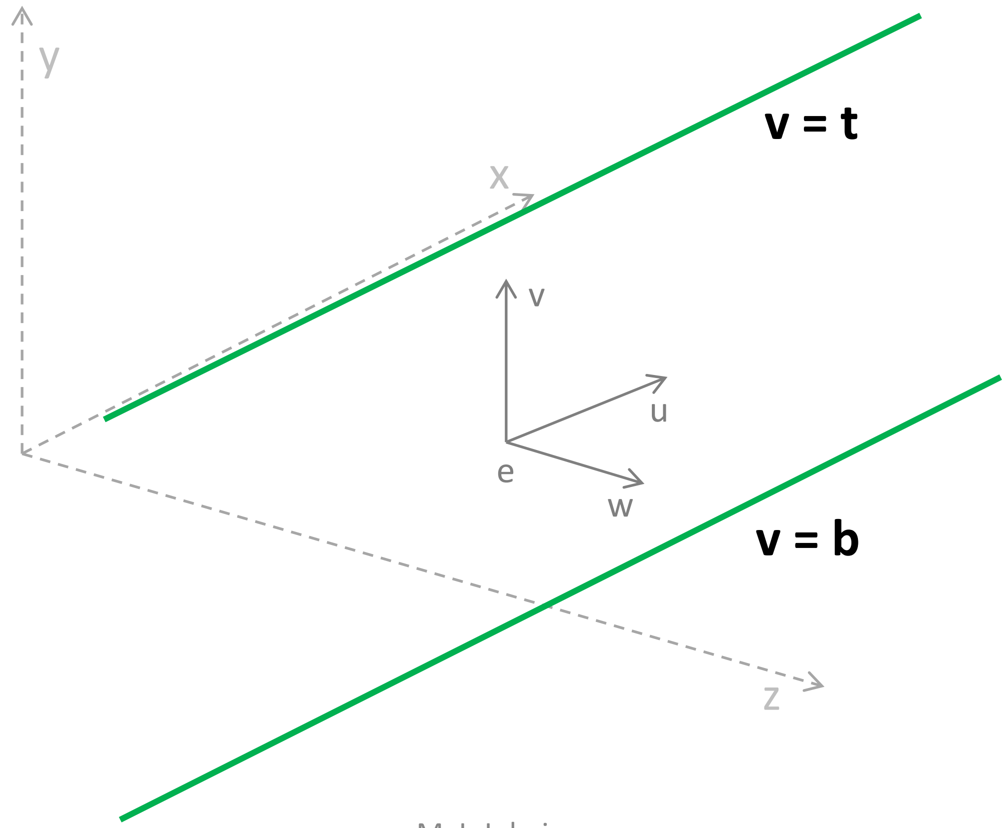


Image Plane (3/4)

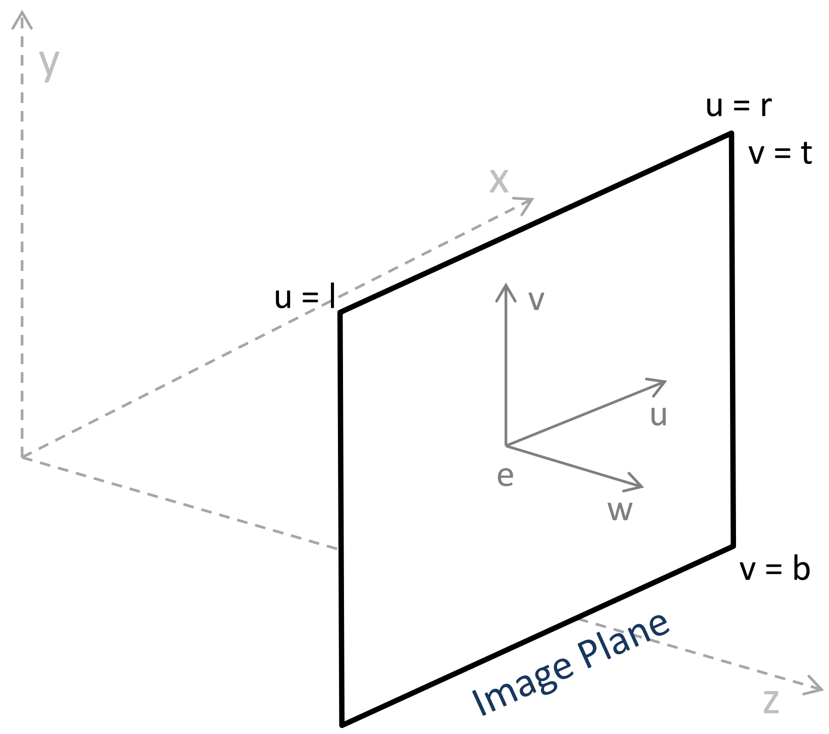
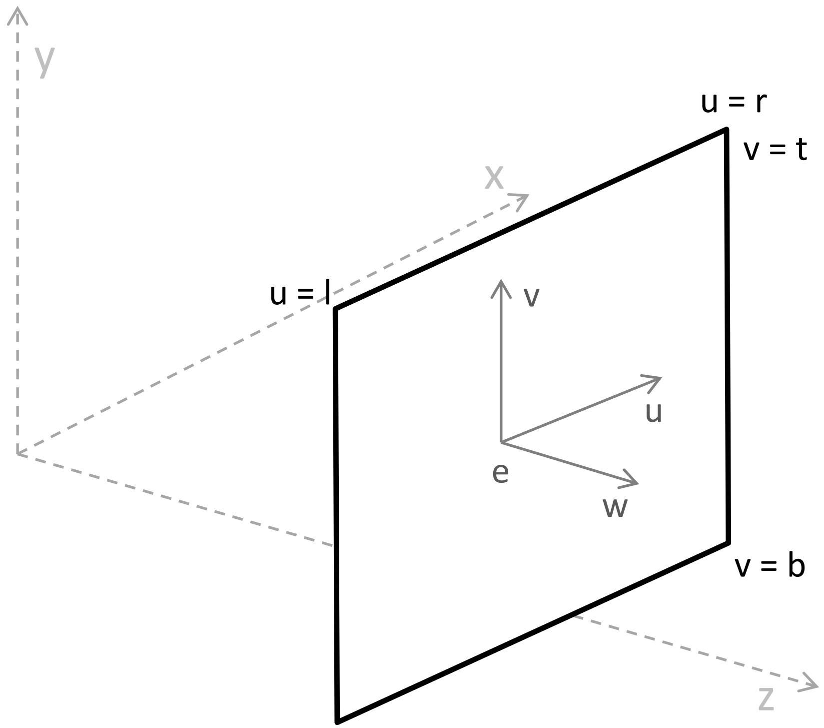
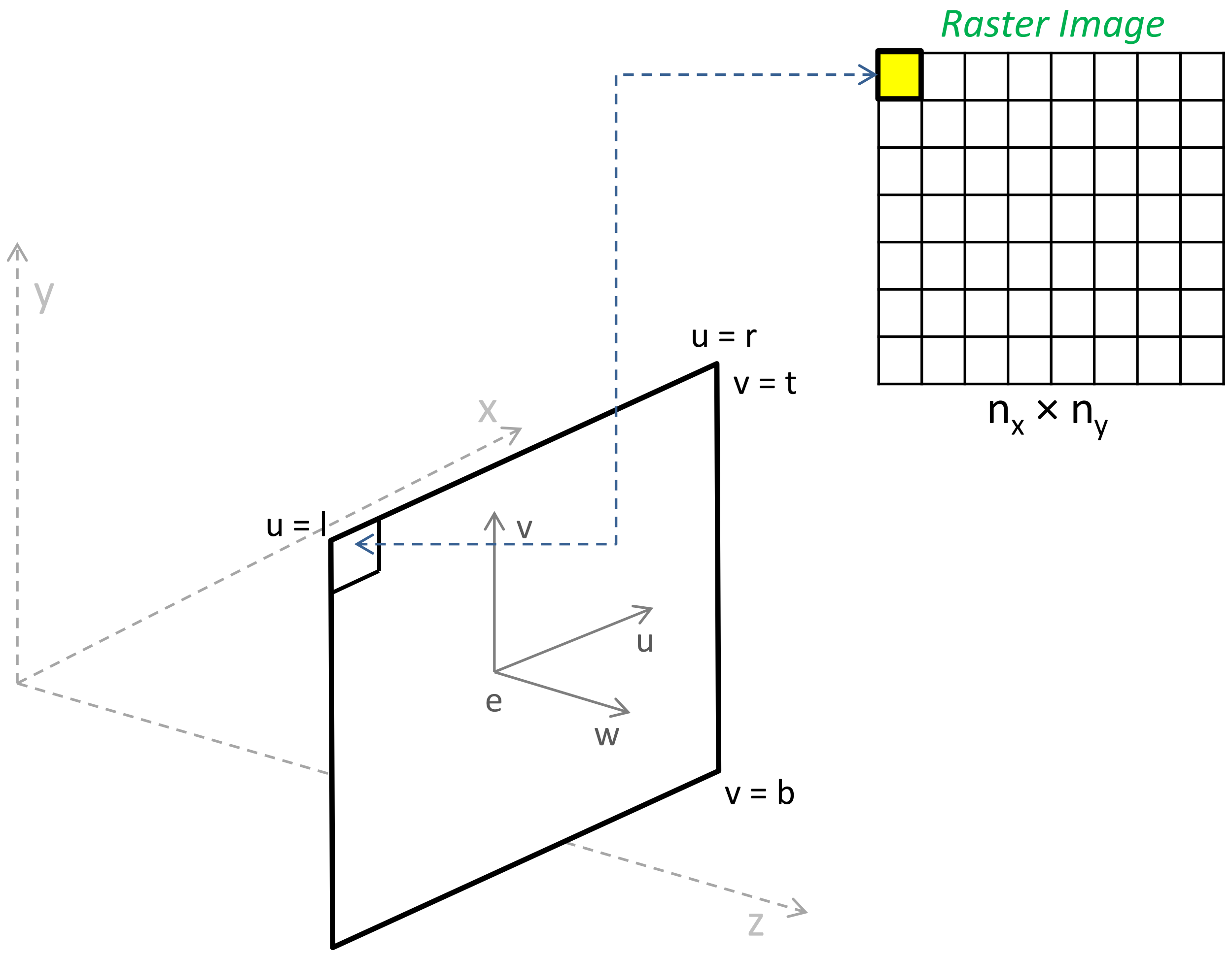


Image Plane (4/4)

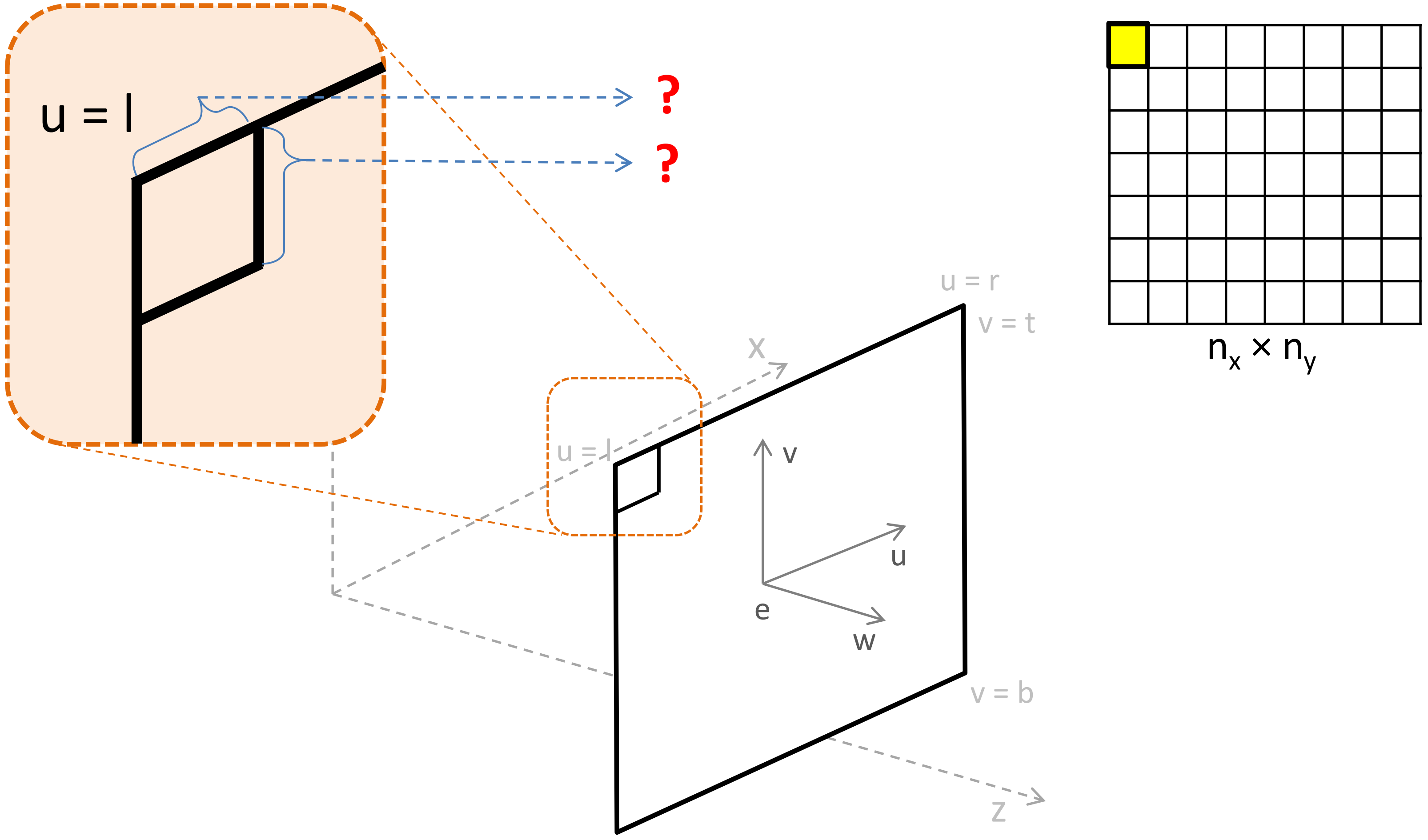
Q: determine the area of the image plane in terms of *l, r, t and b.*



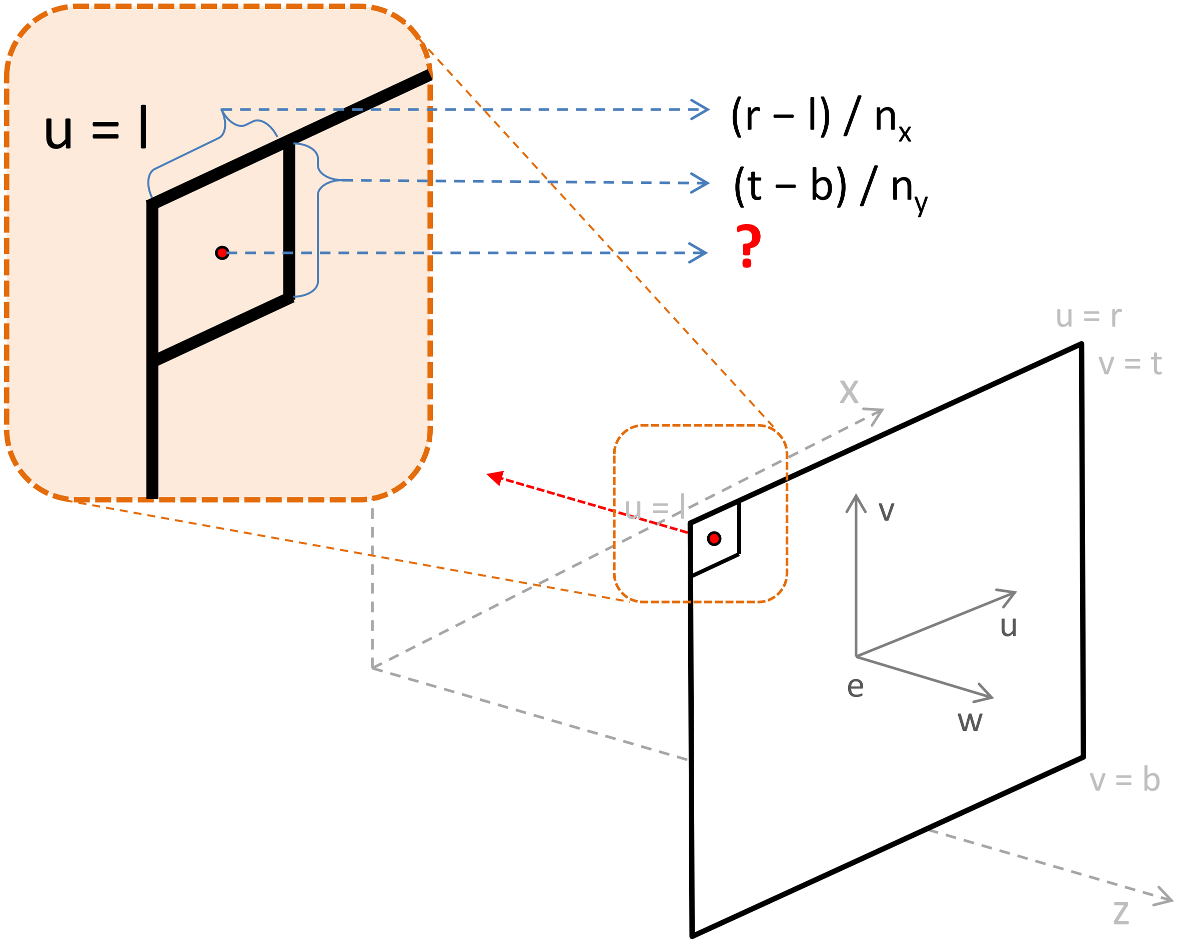
Raster Image ↔ *Image Plane* (1/8)

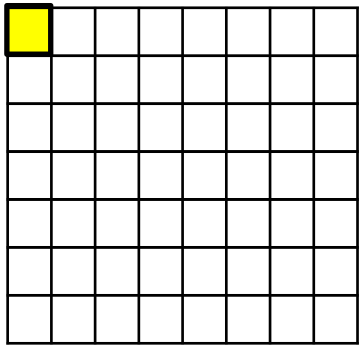


Raster Image ↔ Image Plane (2/8)

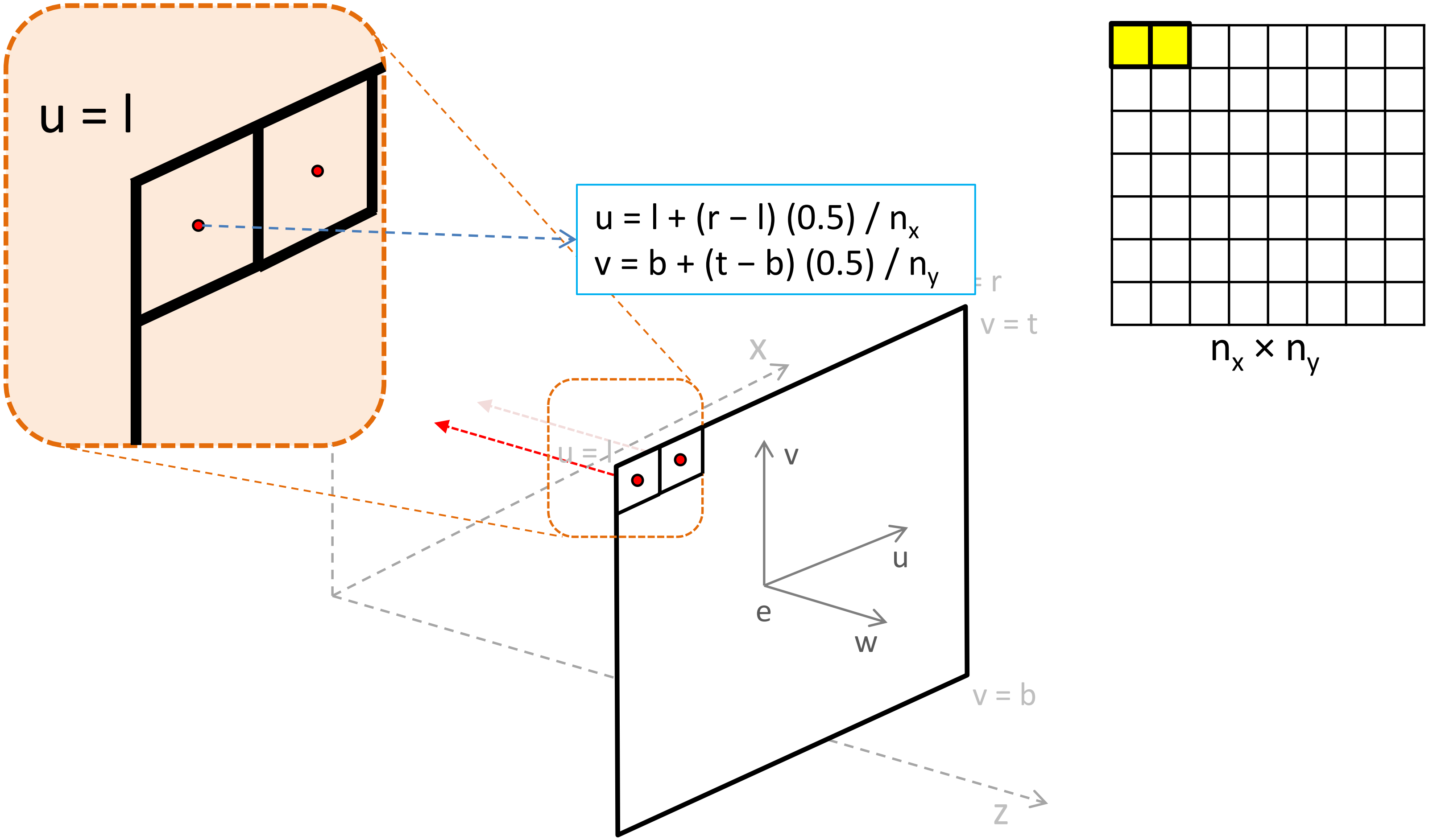


Raster Image ↔ Image Plane (3/8)



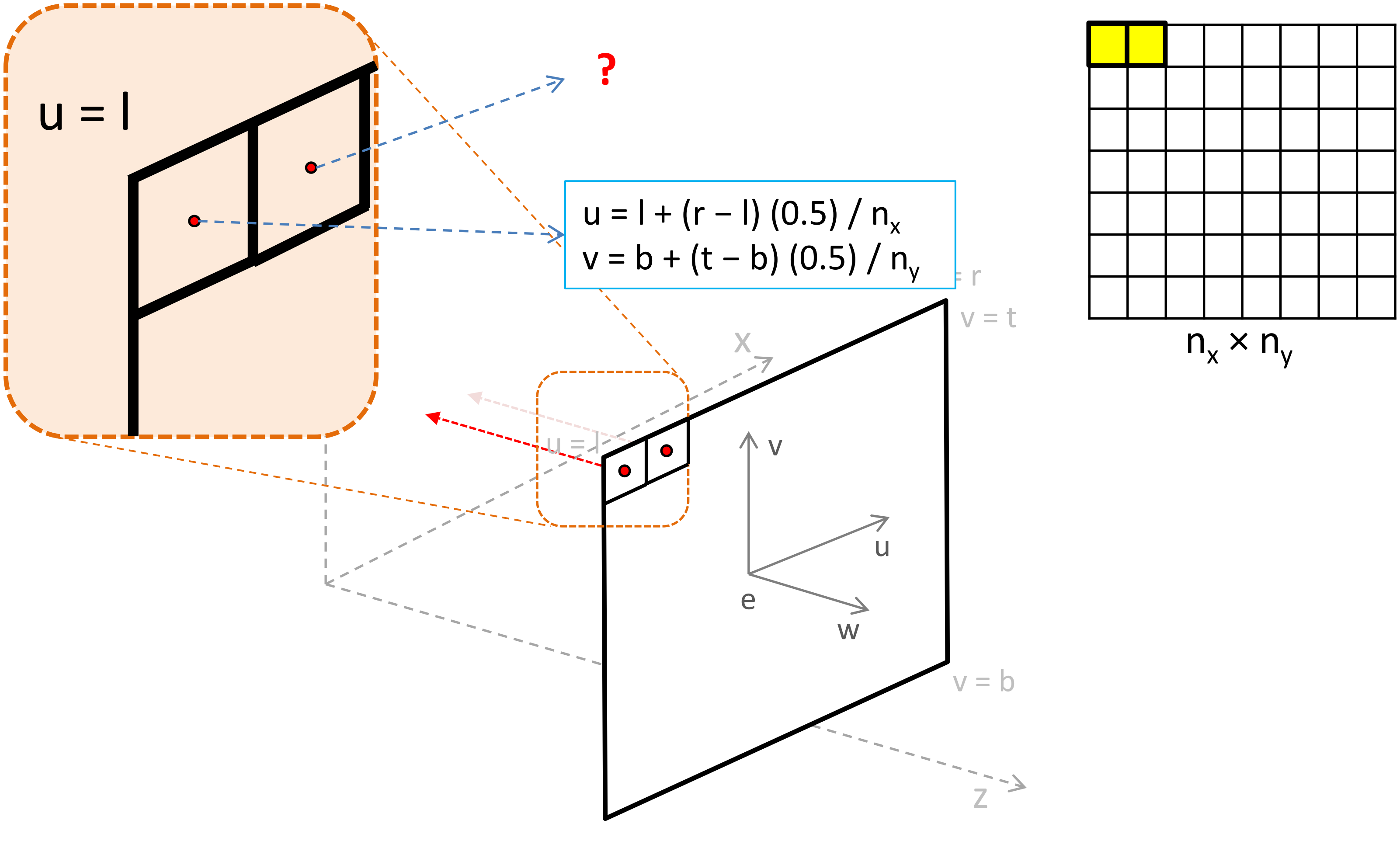


Raster Image ⟷ Image Plane (4/8)

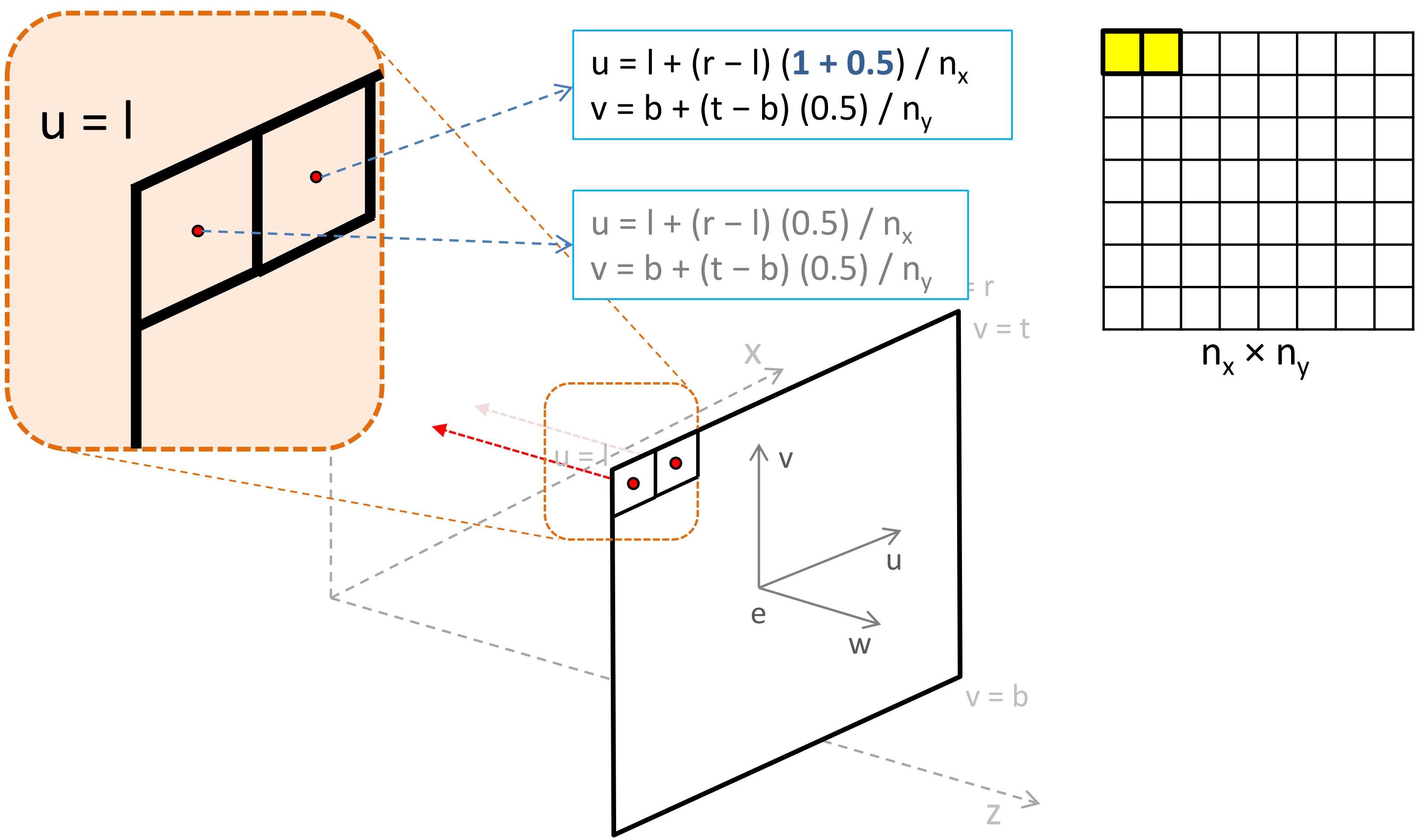


$$u = l + (r - l)\,(0.5) / n_x$$
$$v = b + (t - b)\,(0.5) / n_y$$

Raster Image ↔ Image Plane (5/8)

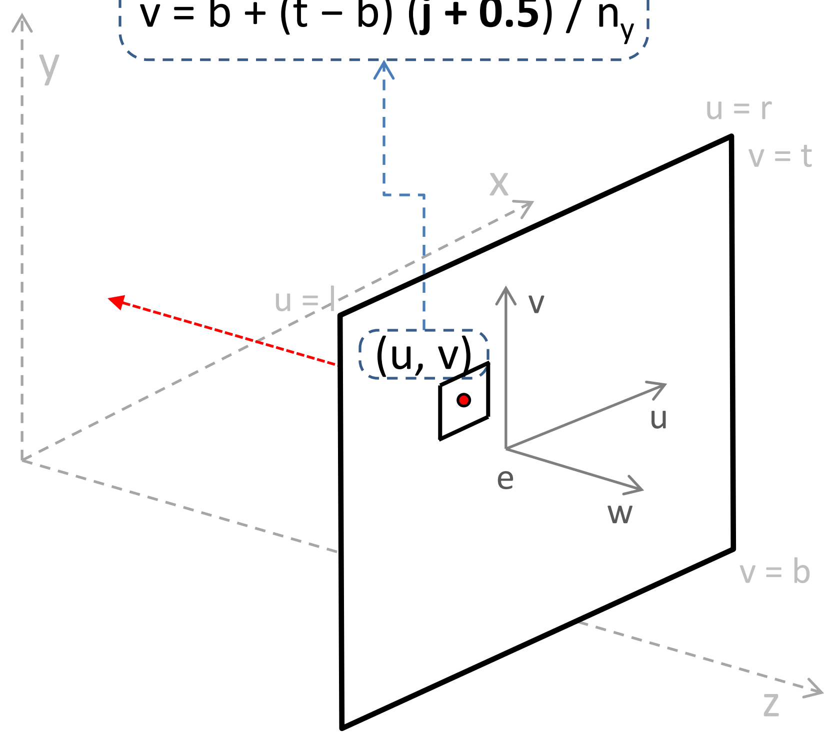


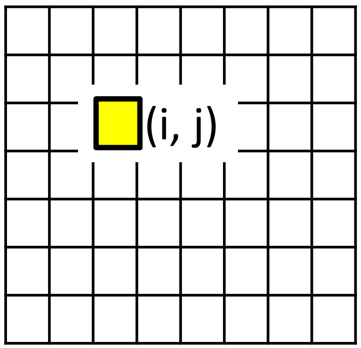
Raster Image ↔ Image Plane (6/8)



Raster Image ↔ Image Plane (7/8)

$$u = l + (r - l)\,(\mathbf{i + 0.5})\,/\,n_x$$
$$v = b + (t - b)\,(\mathbf{j + 0.5})\,/\,n_y$$

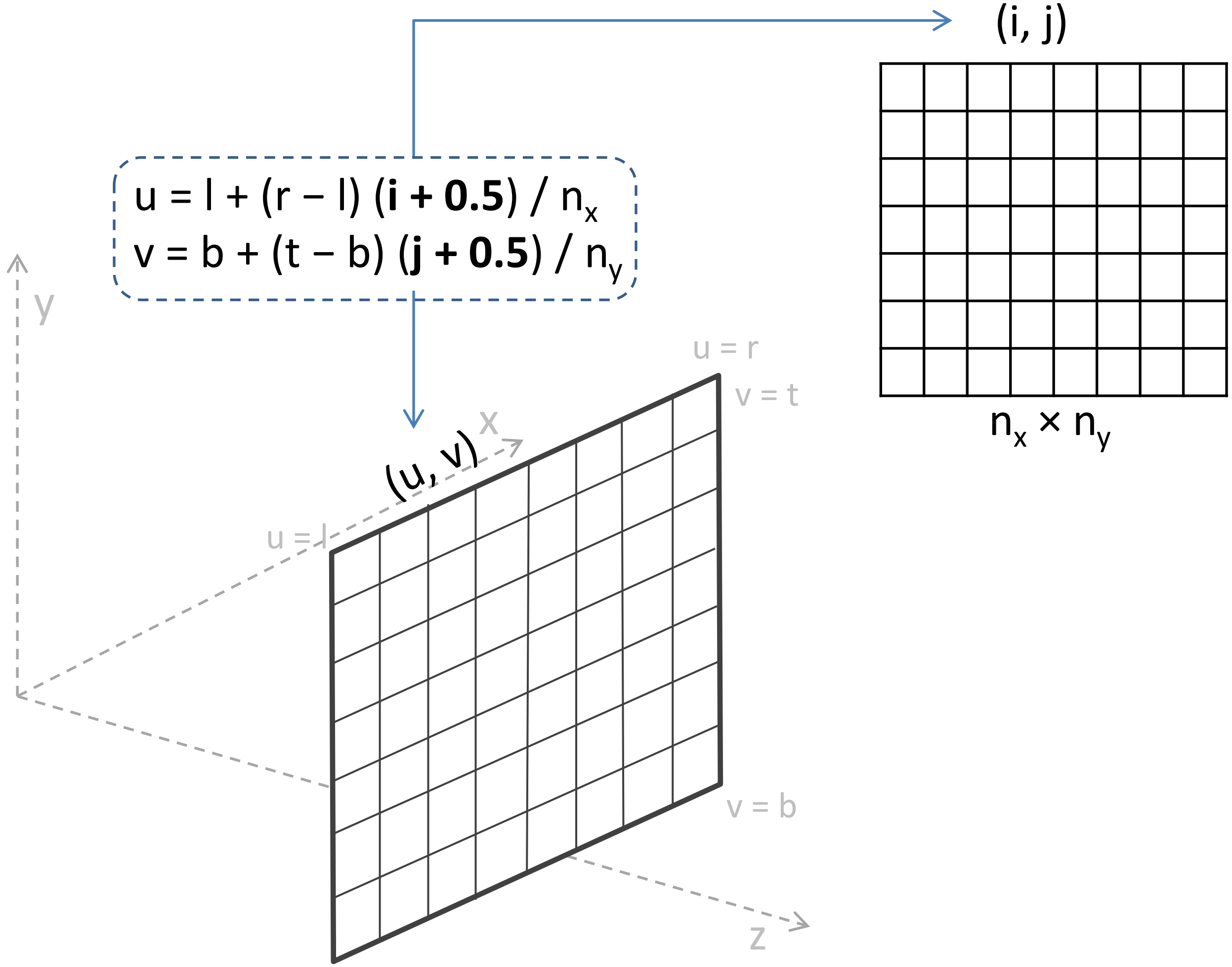




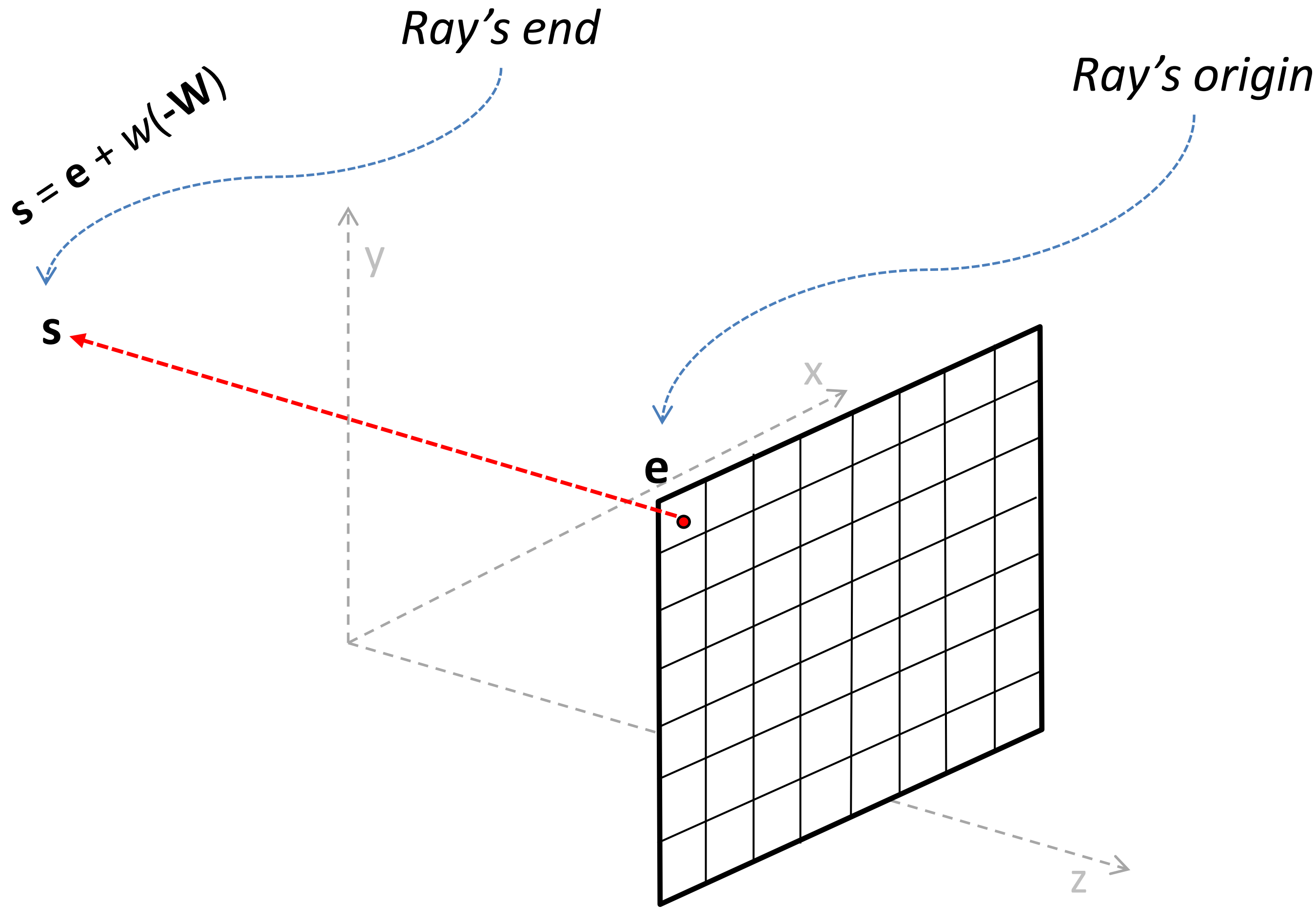
$n_x \times n_y$

Raster Image ↔ Image Plane (8/8)

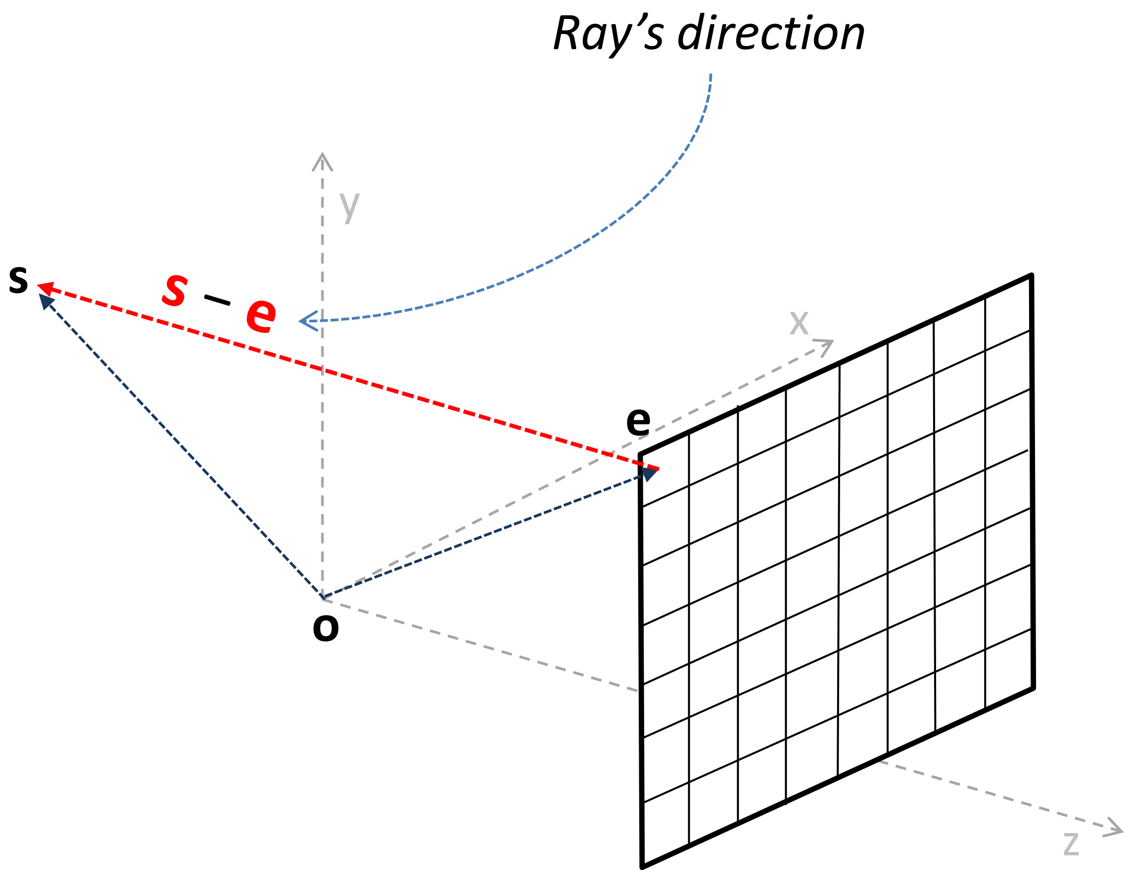
$$u = l + (r - l)\,(\mathbf{i + 0.5}) / n_x$$
$$v = b + (t - b)\,(\mathbf{j + 0.5}) / n_y$$



Computing Viewing Rays (1/4)

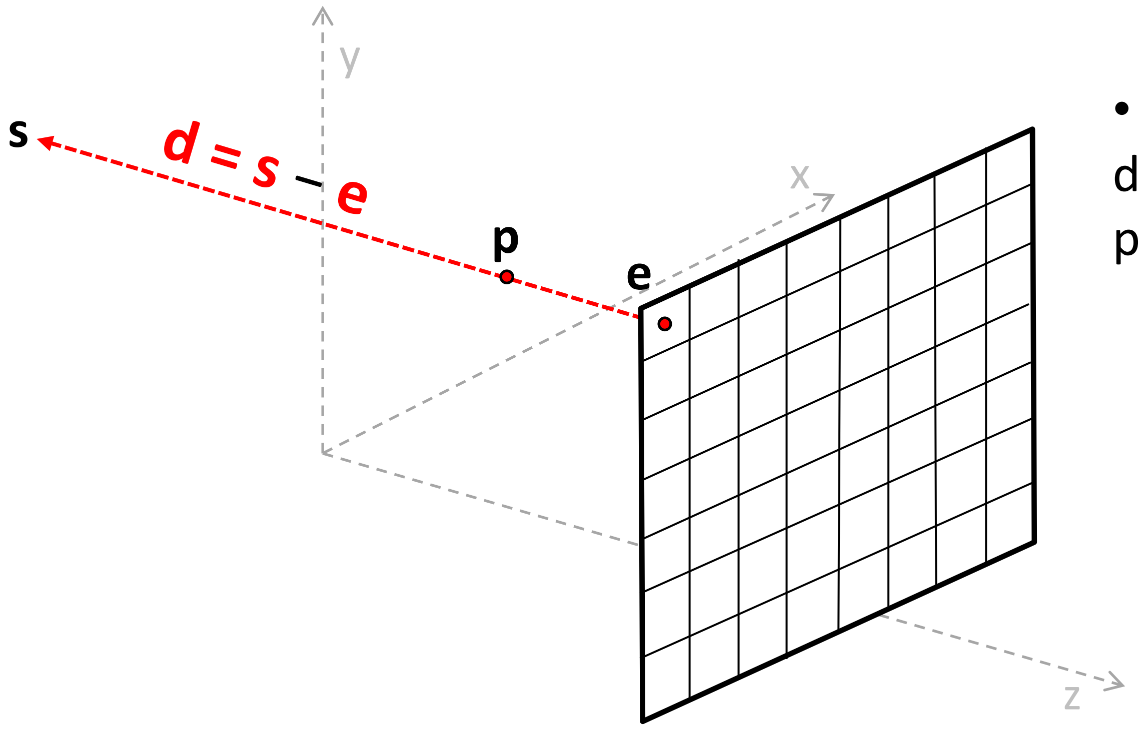


Computing Viewing Rays (2/4)



Computing Viewing Rays (3/4)

$$\mathbf{p} = \mathbf{e} + t\,(\mathbf{s} - \mathbf{e})$$
$$= \mathbf{e} + t\,\mathbf{d}$$

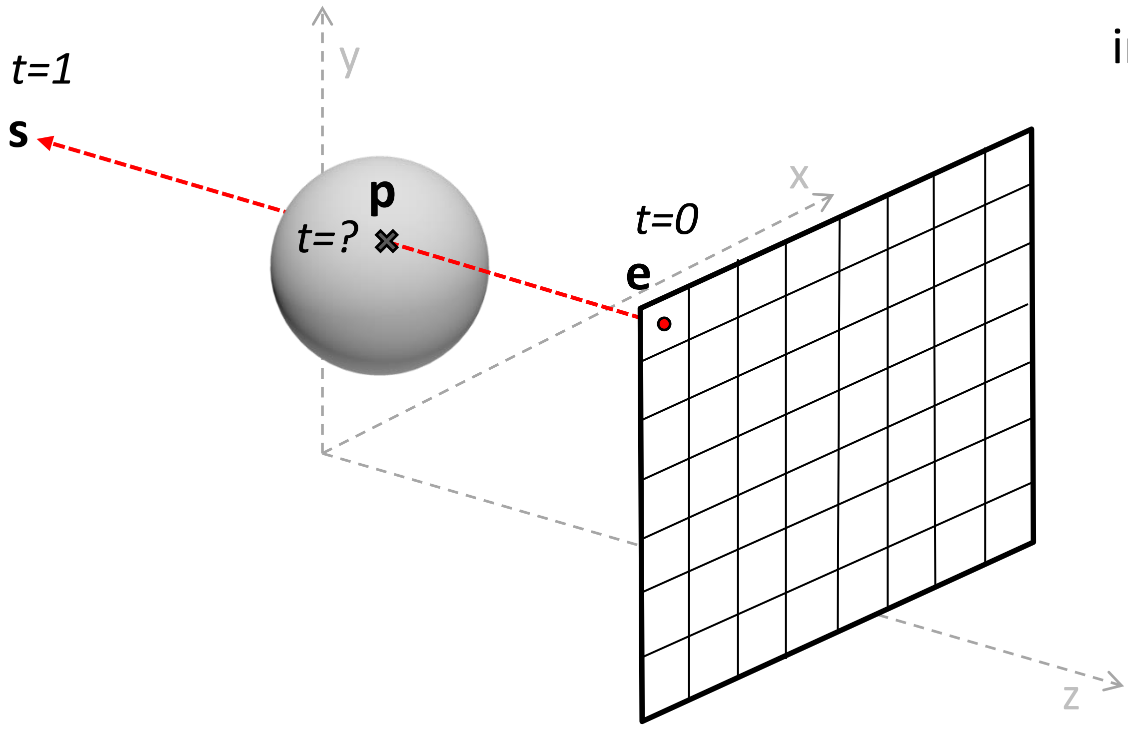


- Advancing from **e** along the vector **(s – e)**
- With a fractional distance ***t*** to find the point **p**

Computing Viewing Rays (4/4)

$$\mathbf{p} = \mathbf{e} + t\,(\mathbf{s} - \mathbf{e})$$
$$= \mathbf{e} + t\,\mathbf{d}$$

- We can use ***t*** to determine the intersection point **p**

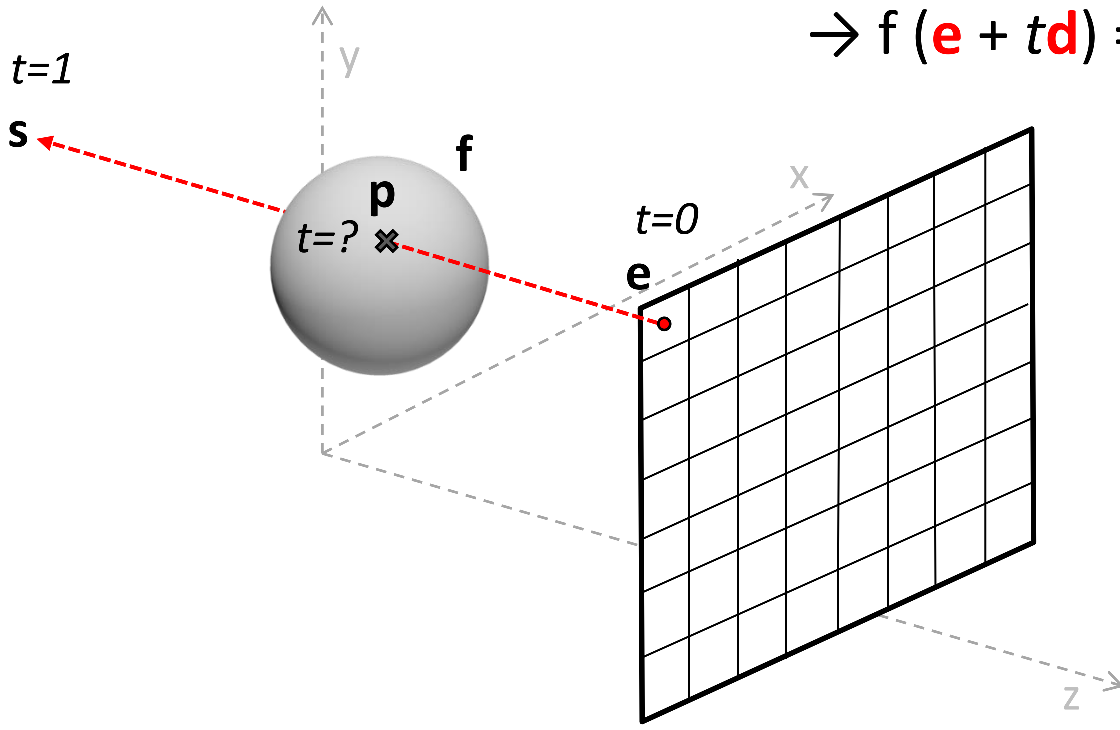


Ray - Sphere Intersection (1/8)

We have, $\mathbf{p} = \mathbf{e} + t\,(\mathbf{s} - \mathbf{e}) = \mathbf{e} + t\,\mathbf{d}$

$\rightarrow f(\mathbf{p}) = 0$

$\rightarrow f(\mathbf{e} + t\mathbf{d}) = 0$

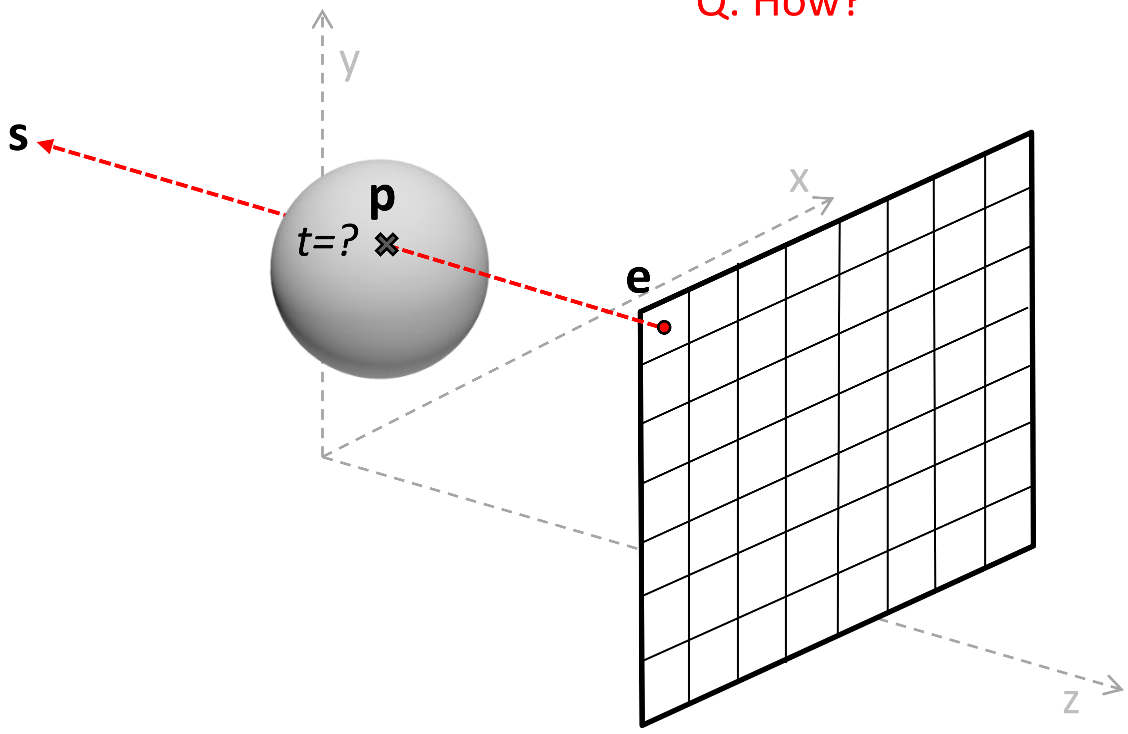


Ray - Sphere Intersection (2/8)

→ $(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 - R^2 = 0$

→ $(\mathbf{p} - \mathbf{c}) \cdot (\mathbf{p} - \mathbf{c}) - R^2 = 0$

Q: How?



Ray - Sphere Intersection (3/8)

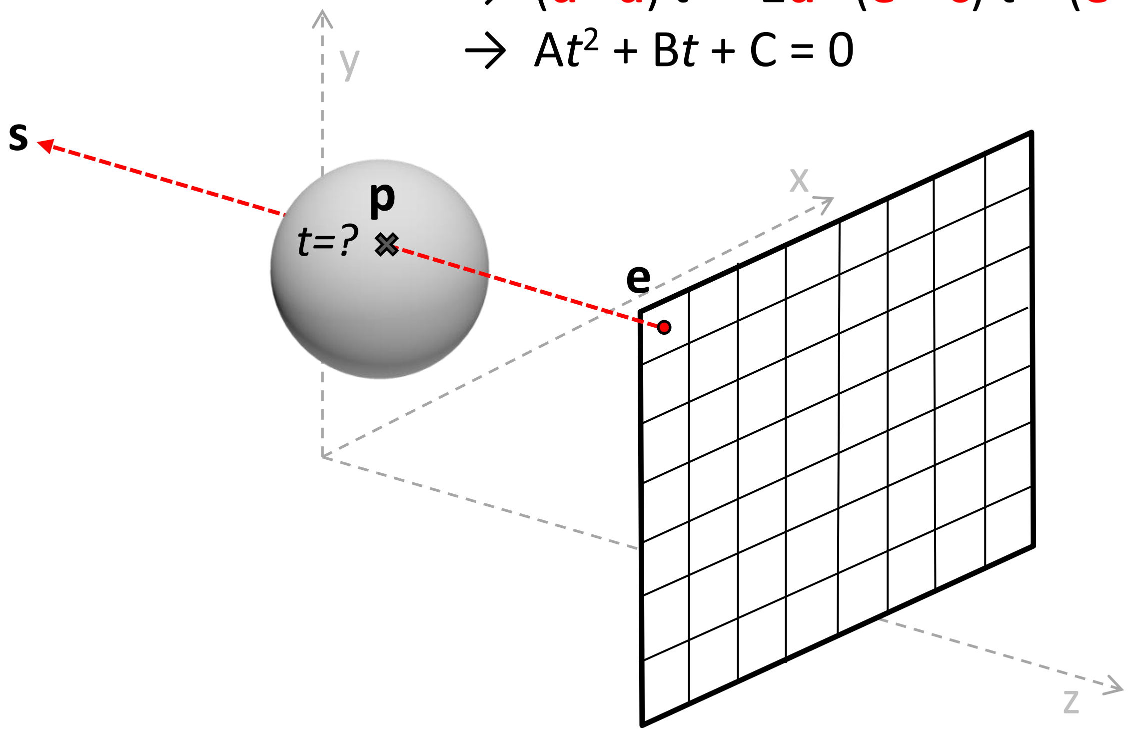
→ $(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 - R^2 = 0$

→ $(\mathbf{p} - \mathbf{c}) . (\mathbf{p} - \mathbf{c}) - R^2 = 0$

→ $(\mathbf{e} + t\mathbf{d} - \mathbf{c}) \cdot (\mathbf{e} + t\mathbf{d} - \mathbf{c}) - R^2 = 0$

→ $(\mathbf{d} \cdot \mathbf{d})\, t^2 + 2\mathbf{d} \cdot (\mathbf{e} - \mathbf{c})\, t + (\mathbf{e} - \mathbf{c}) \cdot (\mathbf{e} - \mathbf{c}) - R^2 = 0$

→ $At^2 + Bt + C = 0$



Ray - Sphere Intersection (4/8)

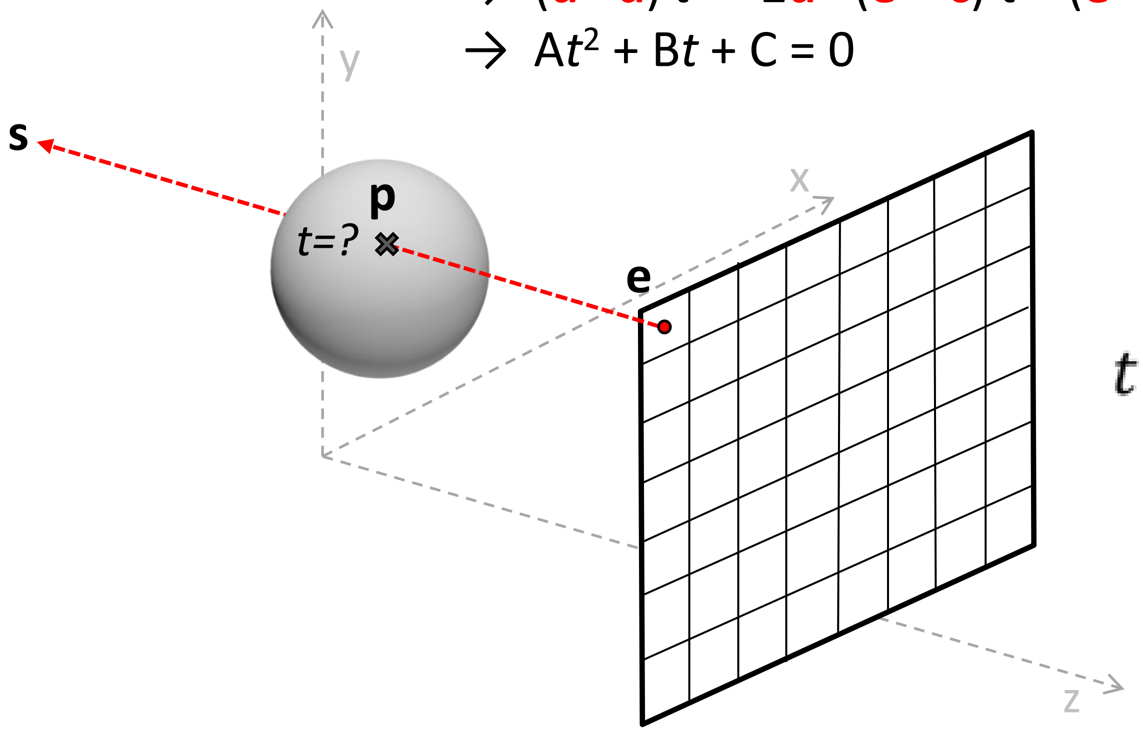
→ $(x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 - R^2 = 0$

→ $(\mathbf{p} - \mathbf{c}) . (\mathbf{p} - \mathbf{c}) - R^2 = 0$

→ $(\mathbf{e} + t\mathbf{d} - \mathbf{c}) \cdot (\mathbf{e} + t\mathbf{d} - \mathbf{c}) - R^2 = 0$

→ $(\mathbf{d} \cdot \mathbf{d})\, t^2 + 2\mathbf{d} \cdot (\mathbf{e} - \mathbf{c})\, t + (\mathbf{e} - \mathbf{c}) \cdot (\mathbf{e} - \mathbf{c}) - R^2 = 0$

→ $At^2 + Bt + C = 0$



$$t = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

Ray - Sphere Intersection (5/8)

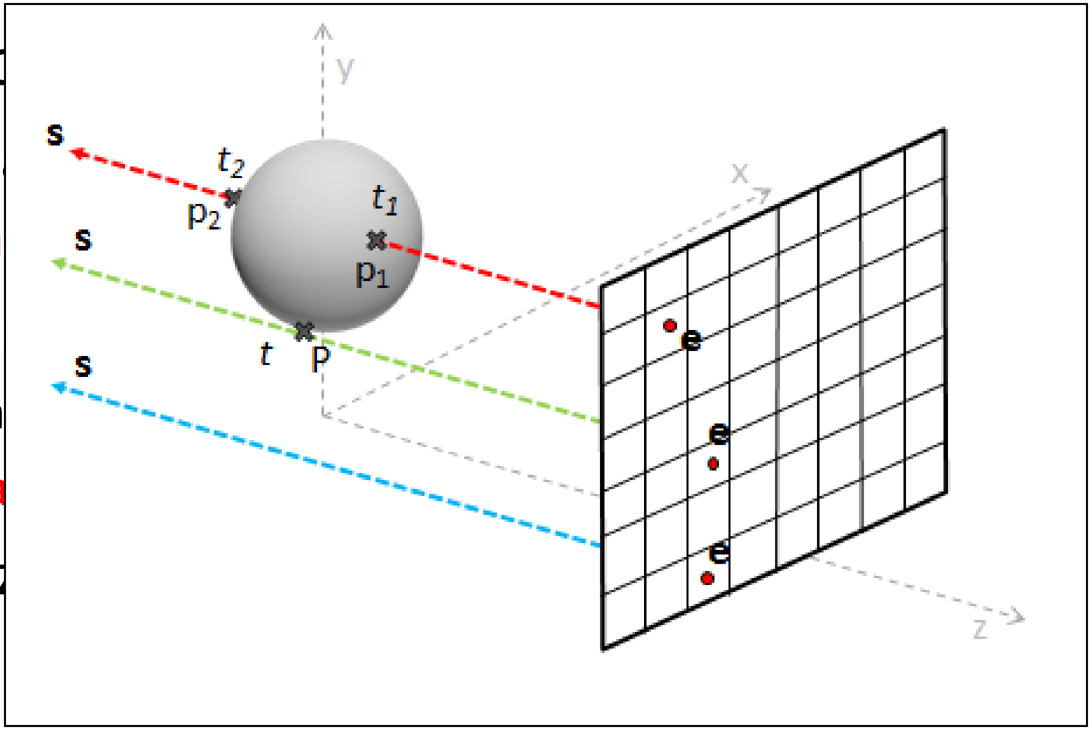
$$t = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

- ***$B^2 - 4AC$***, is called the discriminant and if it is –
  - **negative:** its square root is imaginary and the line and sphere **do not intersect**.
  - **positive**: there are two solutions –
    - one solution where the ray **enters** the sphere.
    - one where it **leaves**.
  - **zero:** the ray grazes the sphere, touching it at exactly **one point**.

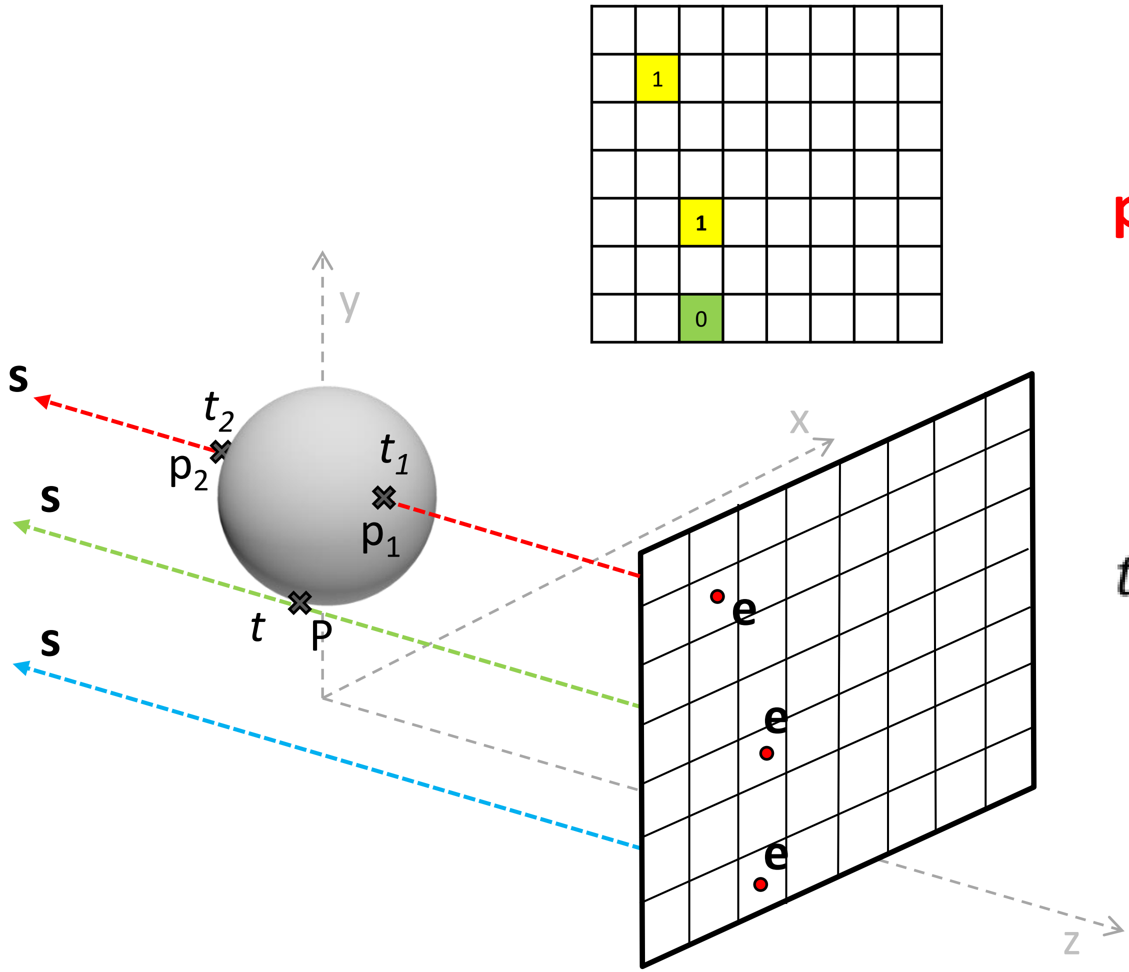
Ray - Sphere Intersection (5/8)

$$t = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

- $B^2 - 4AC$, is calle
  - **negative:** its squ sphere **do not in**
  - **positive**: there a
    - one solution wh
    - one where it **lea**
  - **zero:** the ray gra **point**.



Ray - Sphere Intersection (6/8)



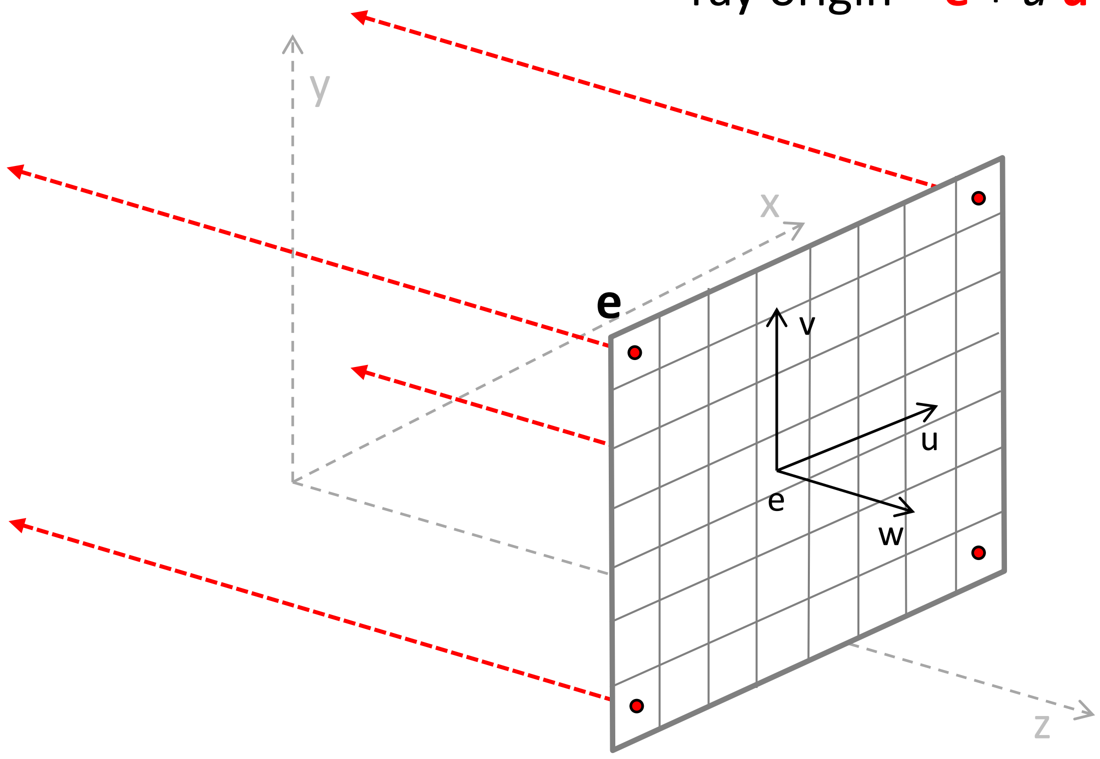
$$\mathbf{p} = \mathbf{e} + t\,(\mathbf{s} - \mathbf{e})$$
$$= \mathbf{e} + t\,\mathbf{d}$$

$$t = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$$

Ray - Sphere Intersection (7/8)

Orthographic:

- ray direction = −**w**
- ray origin = **e** + *u* **u** + *v* **v**

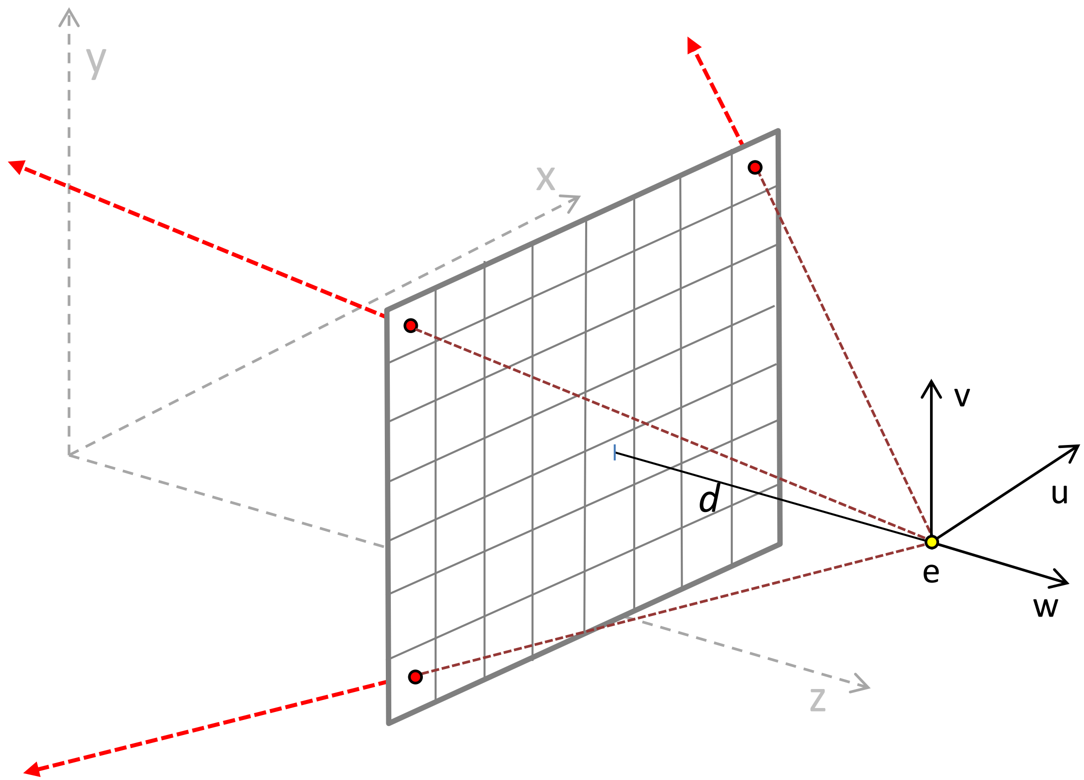


Ray - Sphere Intersection (8/8)

Perspective:

- ray direction = $-d\,\mathbf{w} + u\,\mathbf{u} + v\,\mathbf{v}$
- ray origin = $\mathbf{e}$

Q: why?

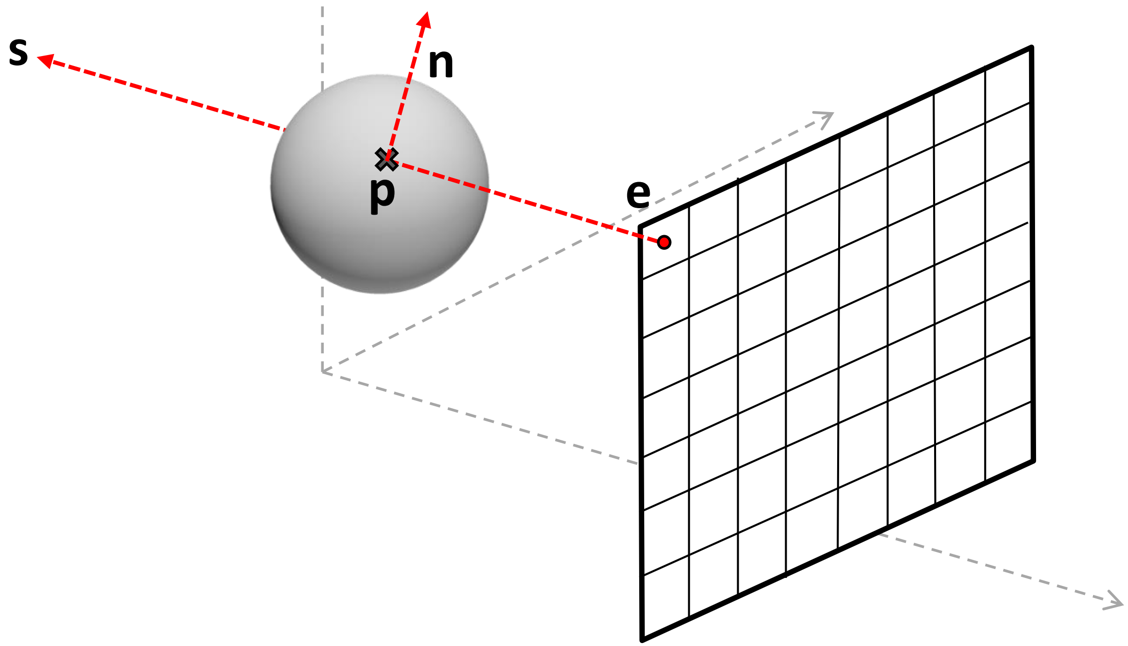


Shading (1/3)

Normal vector at point **p**:

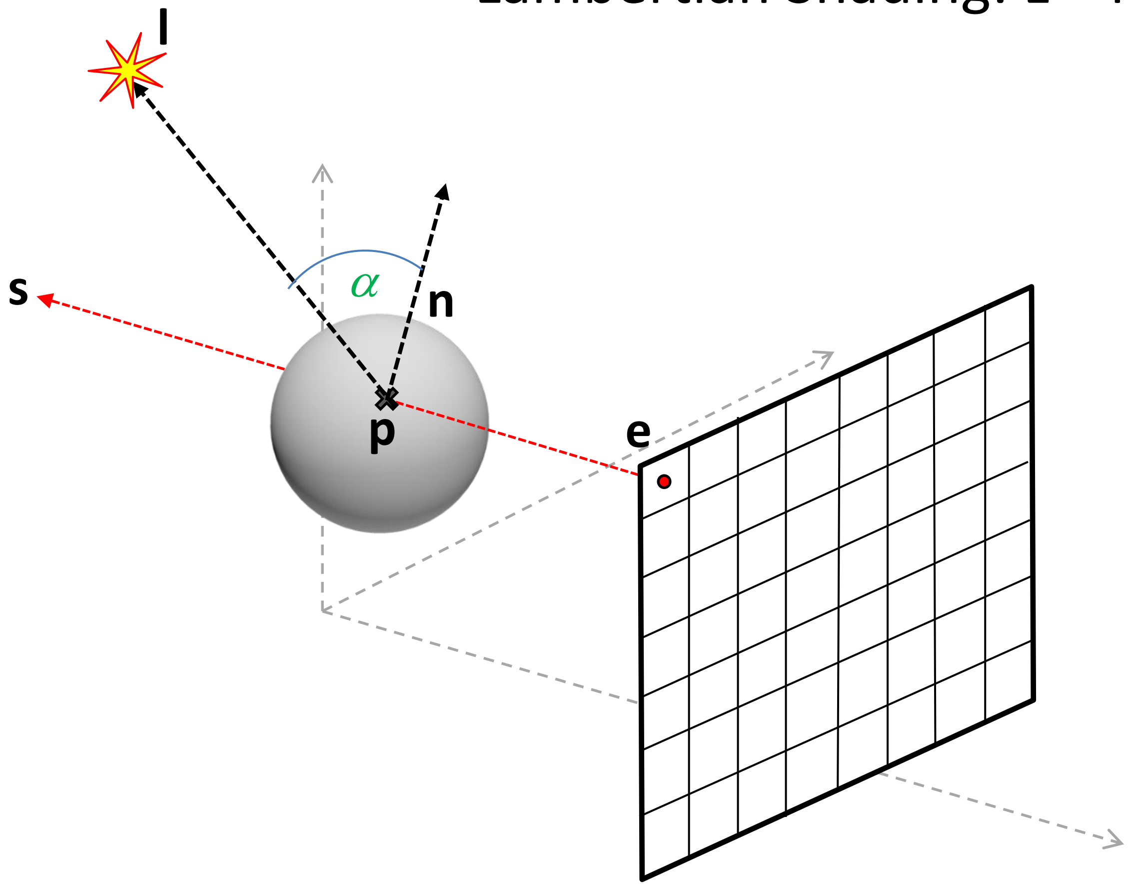
- Gradient, **n** = 2 (**p** − **c**).
- unit normal is (**p** − **c**)/R.

[See section 2.5.4]



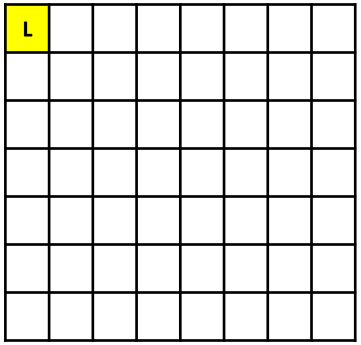
Shading (2/3)

Lambertian Shading: $L = k_d P \, max(0, \mathbf{n} \cdot \mathbf{l})$



where,

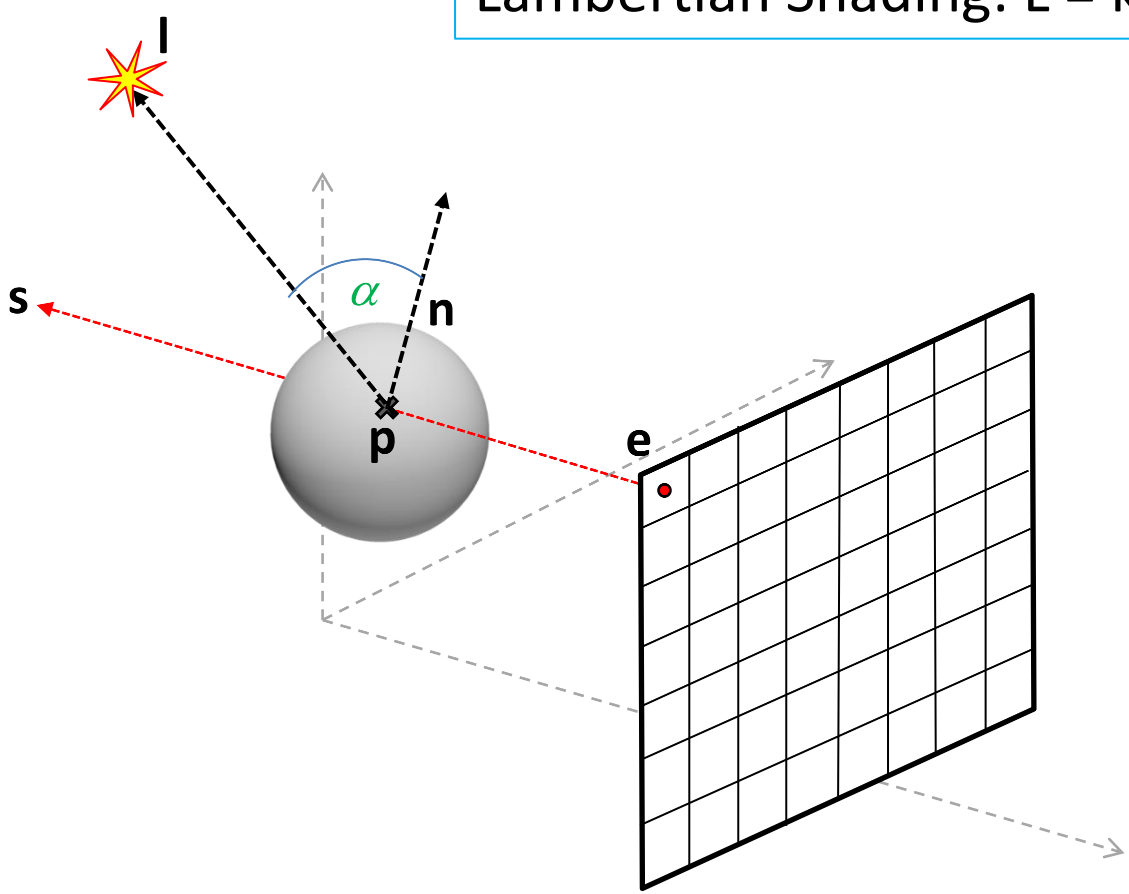
- L = pixel color
- k_d = surface color
- P = intensity of the light source.



Shading (3/3)

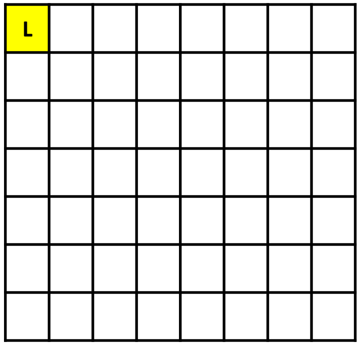
Q: Are we considering angle in this formula? If yes – how?

Lambertian Shading: $L = k_d \, P \, max\,(0, \mathbf{n} \cdot \mathbf{l})$



where,

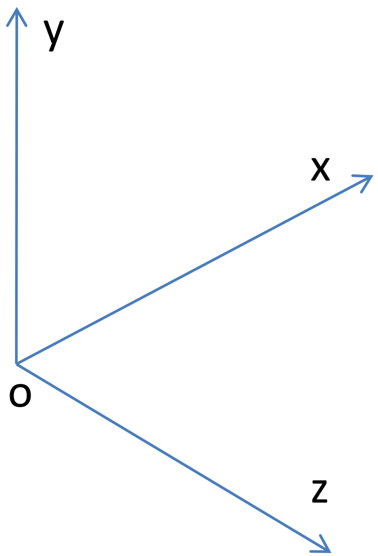
- L = pixel color
- k_d = surface color
- P = intensity of the light source.



Additional Reading

- 4.6: A Ray-Tracing Program

Practice Problem (1/3)



Camera frame *(orthographic)***:**

- **e** = [4, 4, 6]; **u** = [1, 0, 0]; **v** = [0, 1, 0]; **w** = [0, 0, 1]
  - Plot the camera frame on the given axis.

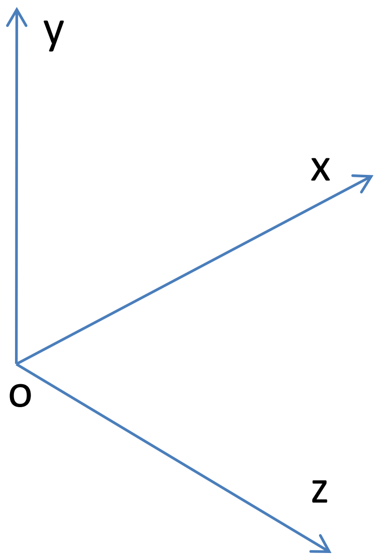
Viewing Ray:

- ray_1.origin = **e** + 2**u** + 2**v**; ray_1.end = [6, 6, 0]
- ray_2.origin = **e** – 1**u** + 1**v**; ray_2.end = [4, 4, 0]
  - Plot the origins for ray_1 and ray_2.

Sphere: $f(x, y, z) = x^2 + y^2 + z^2 - (4)^2 = 0$

1. What are the intersecting points for ray_1 and ray_2?
2. Plot the intersecting points.

Practice Problem (2/3)



Camera frame *(orthographic)***:**

- **e** = [4, 4, 8]; **u** = [1, 0, 0]; **v** = [0, 1, 0]; **w** = [0, 1, 0]

Image Plane:

- left: *u* = - 5; right: *u* = 5; top: *v* = 4; bottom: *v* = -4

1. Plot the image plane on the given axis.
2. For a *10 x 10* image matrix M, what is the position on the image plane for the ray origin at *M (4,3)?*
3. Will it intersect $f(x, y, z)= x^2 + y^2 + z^2 – 5^2 = 0$?

Practice Problem (3/3)

Consider the following parameters for an orthographic ray-tracing:

- *Camera frame:*

  $E = [-2, 7, 17]^T$, $U = [1, 0, 0]^T$, $V = [0, 1, 0]^T$, $W = [0, 0, 1]^T$
- *Image plane:*

  $l = -15$, $r = 15$, $t = 10$, $b = -10$
- *Raster image resolution:* 13×11
- *Sphere:* $(x+3)^2 + (y-5)^2 + (z-3)^2 = 64$

Determine the ray-sphere intersection point(s) for a ray (with *length = 25*) at the *center* of the raster image. Drawing figures is NOT mandatory.

Practice Problem (3/3)

<u>Solution steps:</u>

- Find *u* and *v*

  $u = l + (r - l) (i + 0.5) / nx$

  $v = b + (t - b) (j + 0.5) / ny$

- Determine the ray origin, $\mathbf{e} = \mathbf{E} + u\mathbf{U} + v\mathbf{V}$
- Find ray end point, $\mathbf{s} = \mathbf{e} + w(\mathbf{-W})$
- Determine, $\mathbf{d} = \mathbf{s} - \mathbf{e}$
- Determine, $D = B^2 - 4AC$

  $A = \mathbf{d} \cdot \mathbf{d}$

  $B = 2\mathbf{d} \cdot (\mathbf{e} - \mathbf{c})$

  $C = (\mathbf{e} - \mathbf{c}) \cdot (\mathbf{e} - \mathbf{c}) - R^2$

- Determine the intersection parameter, t

  $t_1 = (-B + \sqrt{D}) / (2A)$

  $t_2 = (-B - \sqrt{D}) / (2A)$

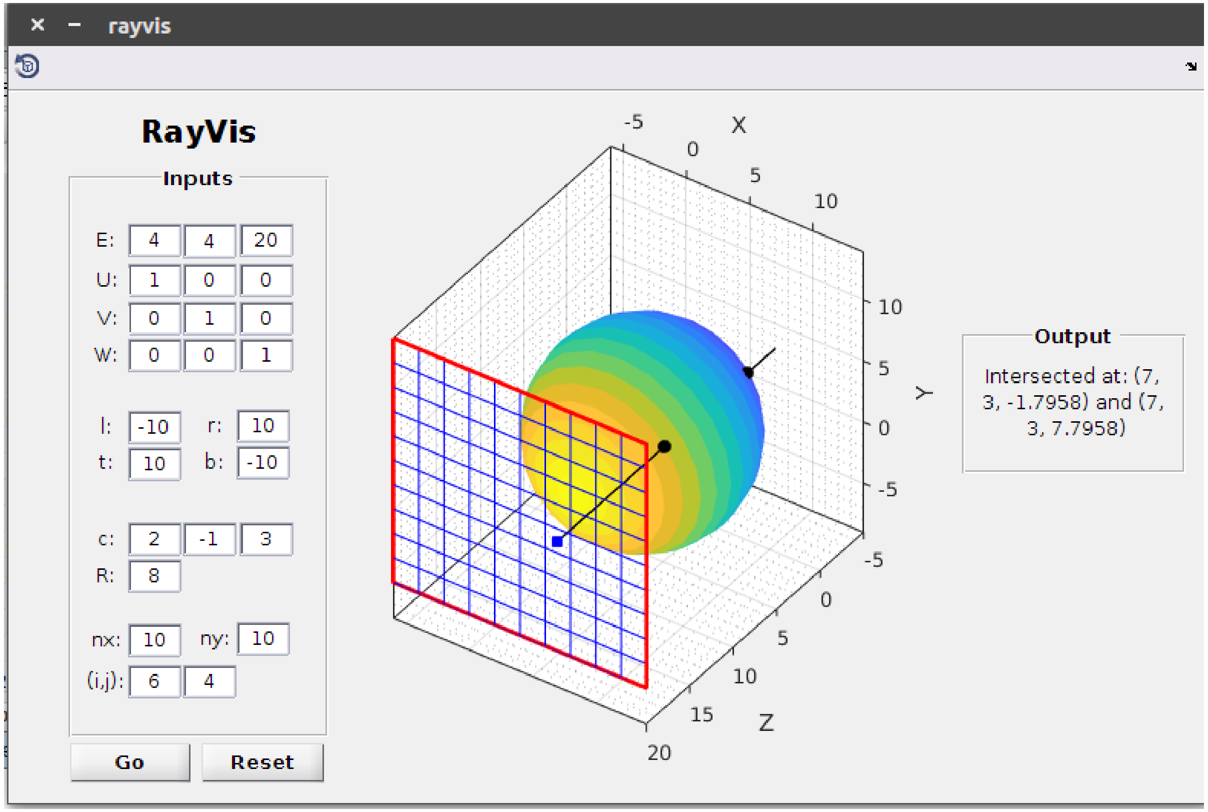
- Determine the intersection point,

  $\mathbf{P_1} = \mathbf{e} + t_1(\mathbf{s} - \mathbf{e})$

  $\mathbf{P_2} = \mathbf{e} + t_2(\mathbf{s} - \mathbf{e})$

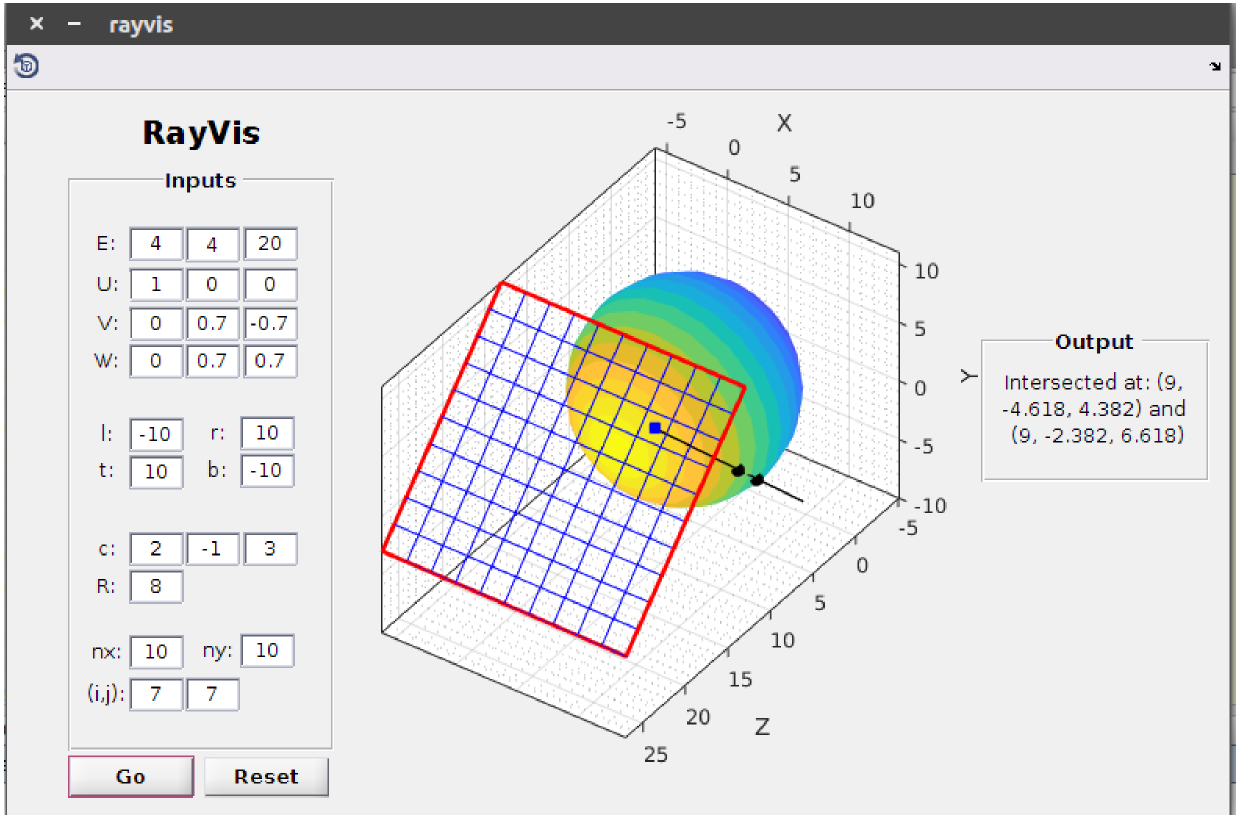
RayVis (1/2)

- https://github.com/imruljubair/RayVis



RayVis (2/2)

- https://github.com/imruljubair/RayVis



Exercise

- Textbook exercise
  - no: 1