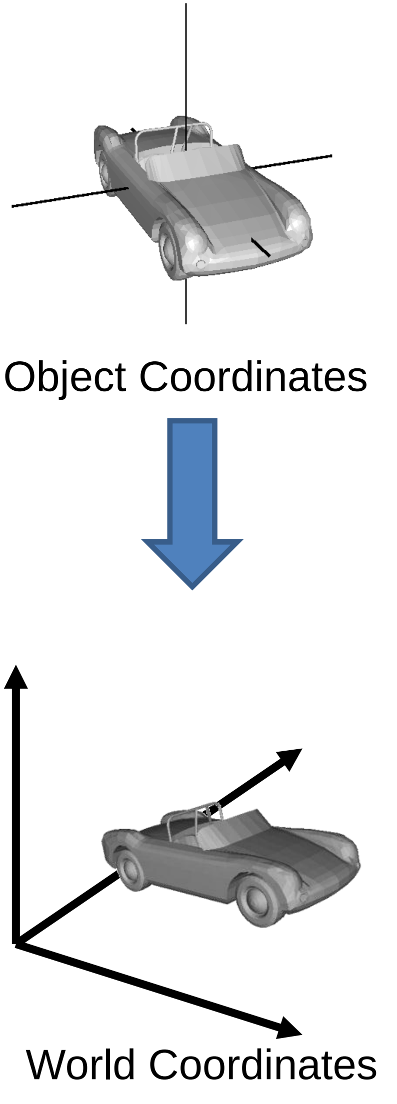
Physically Based Rendering (600.657)

Cameras

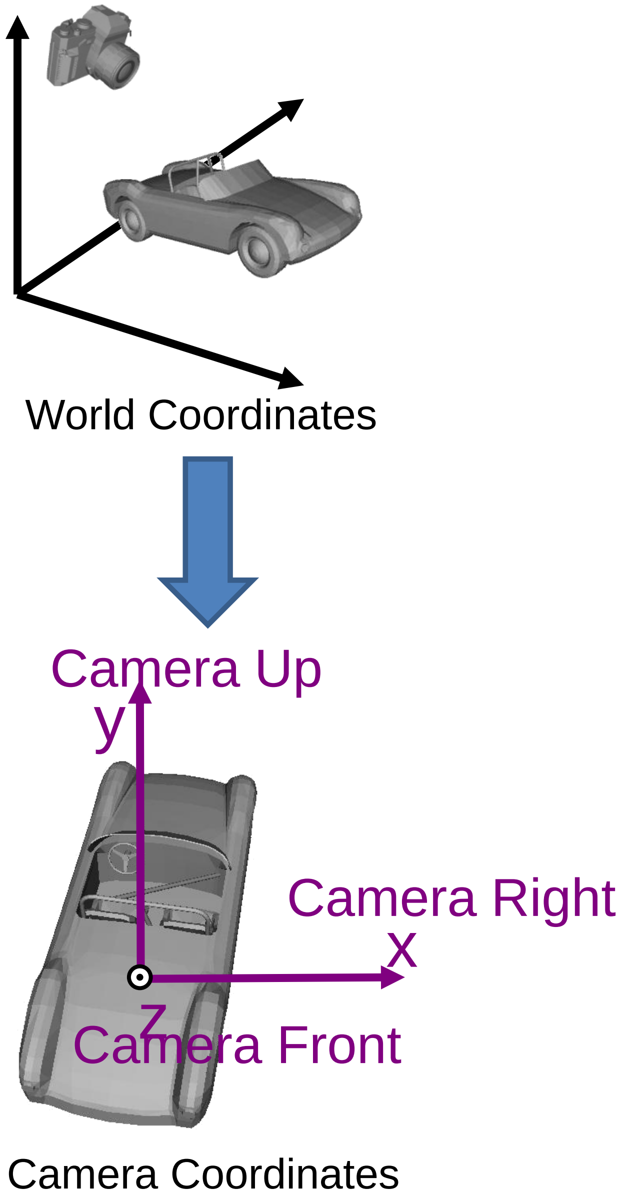
Coordinate Spaces

- **Object**
- **World**
- Camera
- Screen
- NDC
- Raster



Coordinate Spaces

- Object
- **World**
- **Camera**
- Screen
- NDC
- Raster



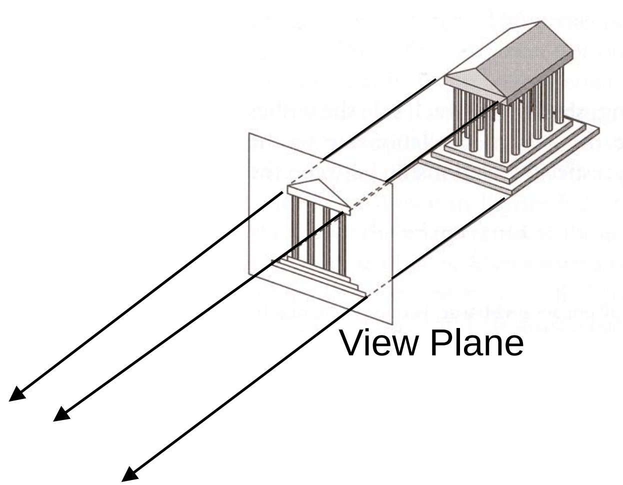
Camera-to-Screen Transformation

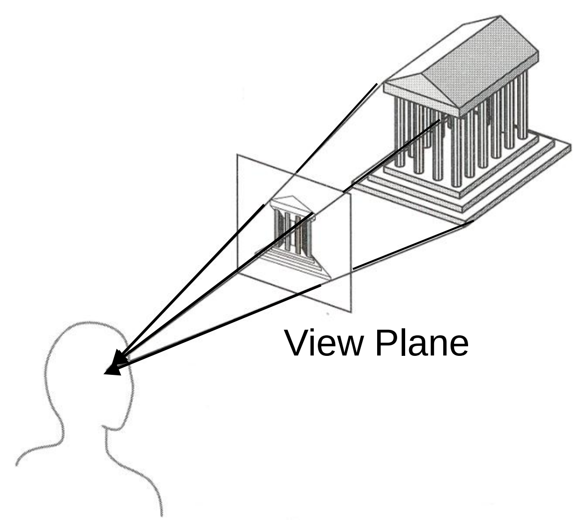
- Linear Models (Projective)
  - Orthographic
  - Perspective
- Non-Linear Models
  - Depth of Field

Linear Models

<u>Projective</u>:

The mapping from camera space to screen space is defined by a projective 4x4 matrix.





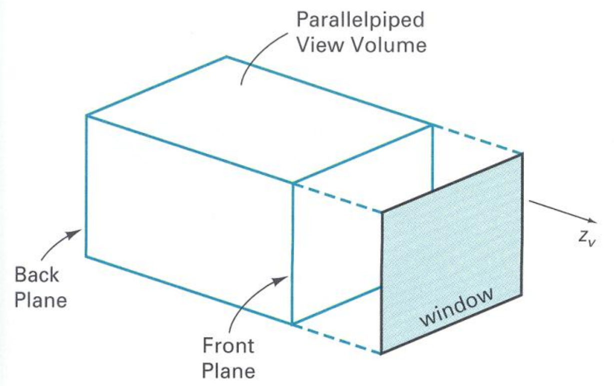
[Images courtesy of Angel]

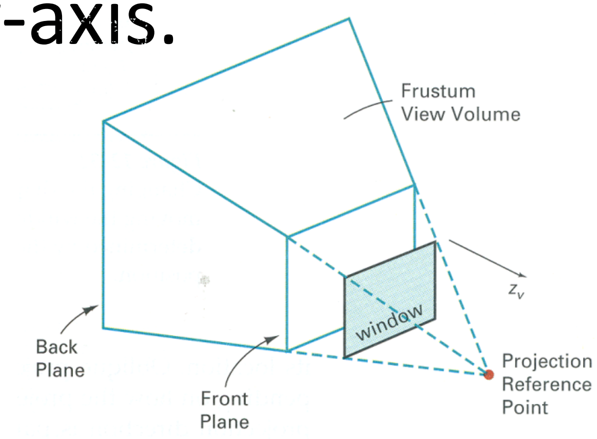
Linear Models

<u>Projective</u>:

The mapping from camera space to screen space is defined by a projective 4x4 matrix.

Given near/ far *z* values, transform the volume into a box with points on the corresponding planes at 0 and 1 along the *z*-axis.



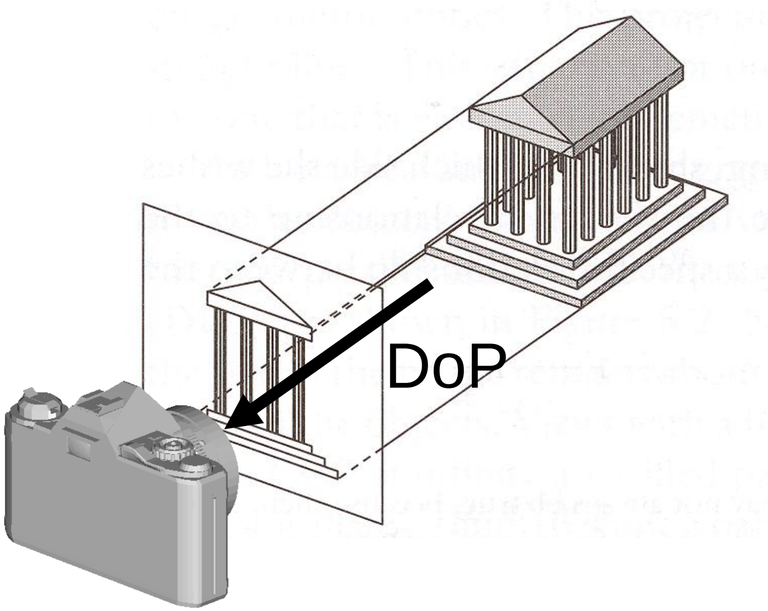


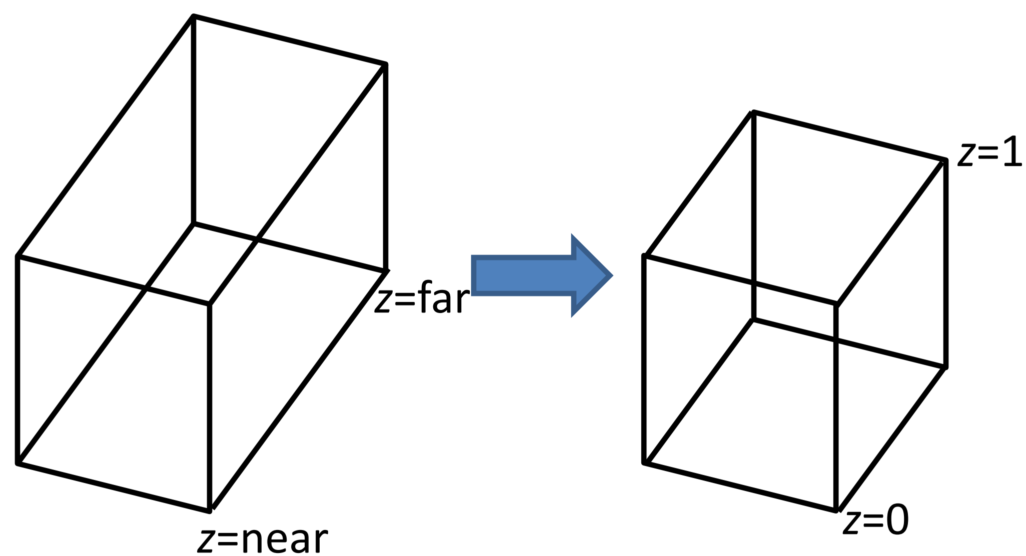
[Images courtesy of H&B]

Linear Models

<u>Orthographic</u>:

The mapping from 3D to 2D is performed by projecting along an axis (typically the *z*-axis).



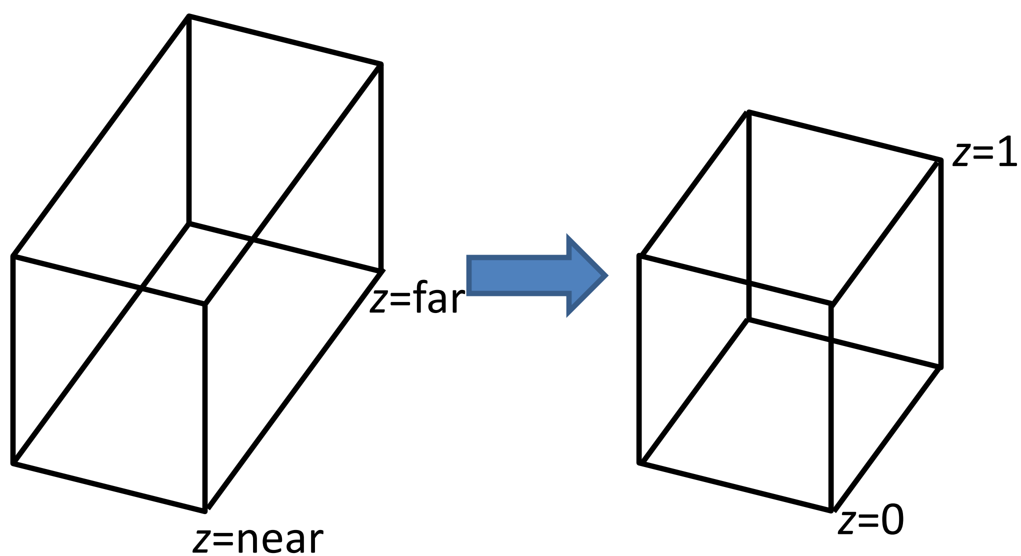


Linear Models

Orthographic:

The mapping from 3D to 2D is performed by projecting along an axis (typically the *z*-axis).

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ \dfrac{z - near}{far - near} \\ 1 \end{pmatrix}$$



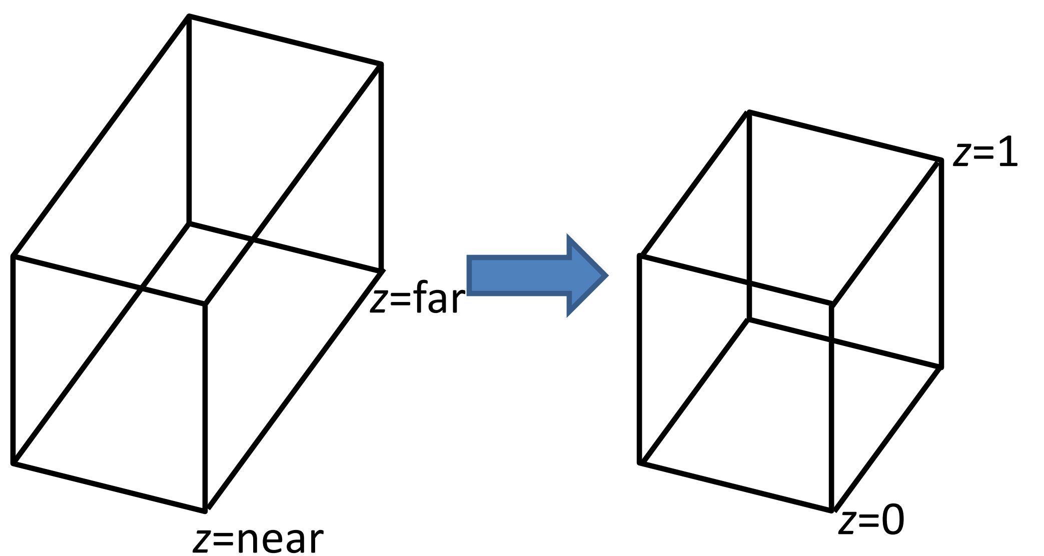
Linear Models

Orthographic:

The mapping from 3D to 2D is performed by projecting along an axis (typically the *z*-axis).

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ \dfrac{z - near}{far - near} \\ 1 \end{pmatrix}$$

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{1}{far - near} & \dfrac{-near}{far - near} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



Linear Models

Orthographic:

To generate a ray through pixel (*i*,*j*):

- Map to the front plane in camera-space:

$$(i, j) \quad \to \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right)$$

Linear Models

Orthographic:

To generate a ray through pixel (*i*,*j*):

- Map to the front plane in camera-space:

$$(i, j) \quad \rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right)$$

$$\rightarrow \quad (x, y, z) = \left(xStart + s\left(xStop - xStart \right), yStop + t\left(yStart - yStop \right), 0 \right)$$

Linear Models

Orthographic:

To generate a ray through pixel (*i,j*):

- Map to the front plane in camera-space:

$$
\begin{aligned}
(i,j) \quad &\rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right) \\
&\rightarrow \quad (x,y,z) = \left(xStart + s\left(xStop - xStart \right), yStop + t\left(yStart - yStop \right), 0 \right) \\
&\rightarrow \quad (c_x, c_y, c_z) = (x, y, z + near)
\end{aligned}
$$

Linear Models

Orthographic:

To generate a ray through pixel (*i,j*):

- Map to the front plane in camera-space:

$$(i,j) \rightarrow (s,t)=\left(\frac{i}{width},\frac{j}{height}\right)$$

$$\rightarrow (x,y,z)=\left(xStart+s\left(xStop-xStart\right), yStop+t\left(yStart-yStop\right),0\right)$$

$$\rightarrow (c_x,c_y,c_z)=(x,y,z+near)$$

- Set the ray start position and direction:

$$ray.start=(c_x,c_y,c_z)$$

$$ray.dir=(0,0,1)$$

Linear Models

Orthographic:

To generate a ray through pixel (*i,j*):

- Map to the front plane in camera-space:

$$
\begin{aligned}
(i, j) \quad &\rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right) \\
&\rightarrow \quad (x, y, z) = \left(xStart + s\left(xStop - xStart\right), yStop + t\left(yStart - yStop\right), 0 \right) \\
&\rightarrow \quad (c_x, c_y, c_z) = (x, y, z + near)
\end{aligned}
$$

- Set the ray start position and direction:

$$ray.start = (c_x, c_y, c_z)$$

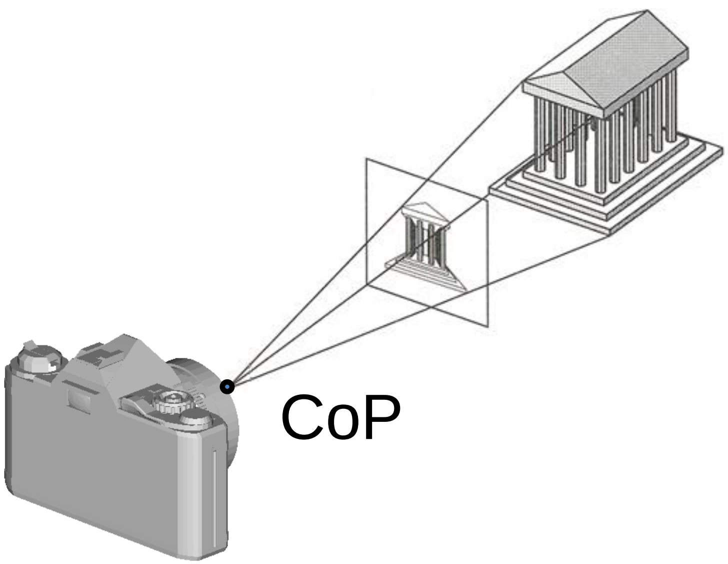
$$ray.dir = (0,0,1)$$

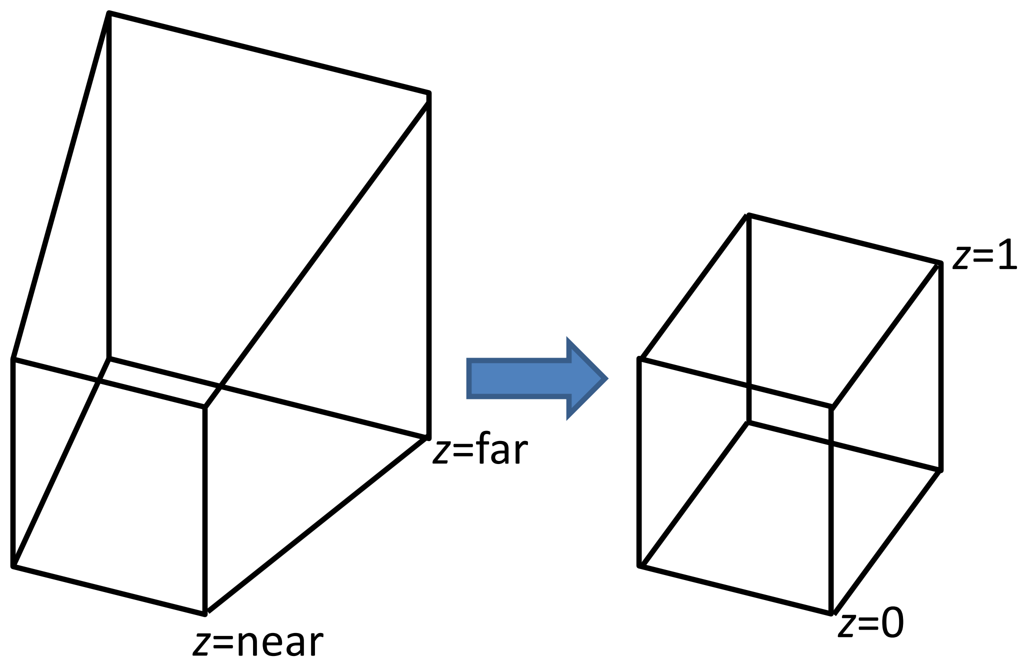
- Transform the ray to world space.

Linear Models

<u>Perspective</u>:

The mapping from 3D to 2D is performed by dividing through by the *z* position.



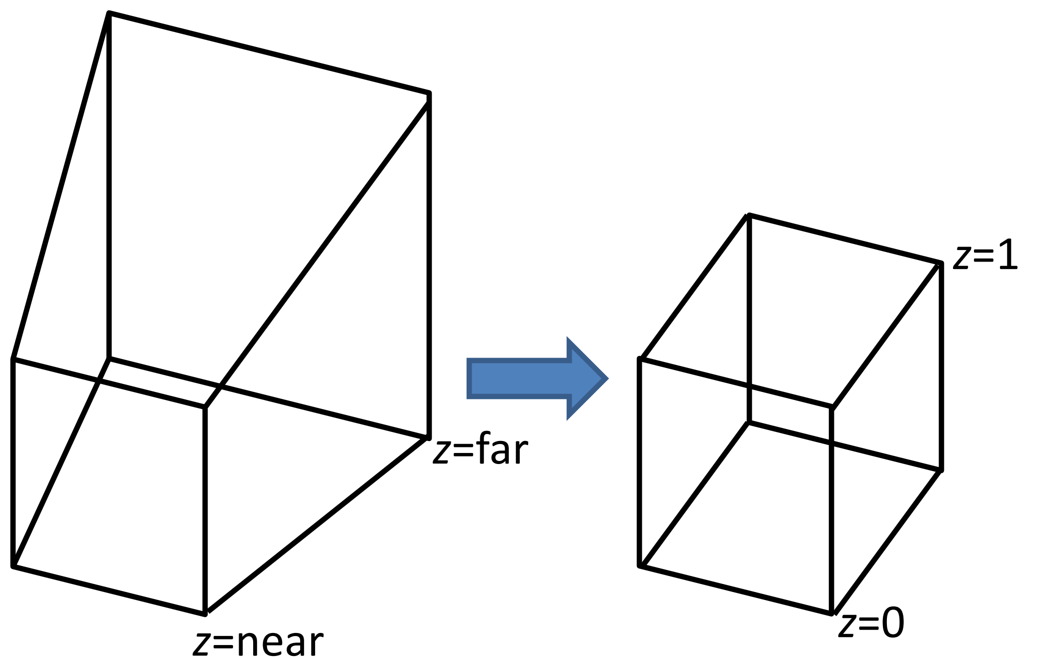


Linear Models

Perspective:

The mapping from 3D to 2D is performed by dividing through by the *z* position.

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} x/z \\ y/z \\ \dfrac{far(z-near)}{z(far-near)} \\ 1 \end{pmatrix}$$

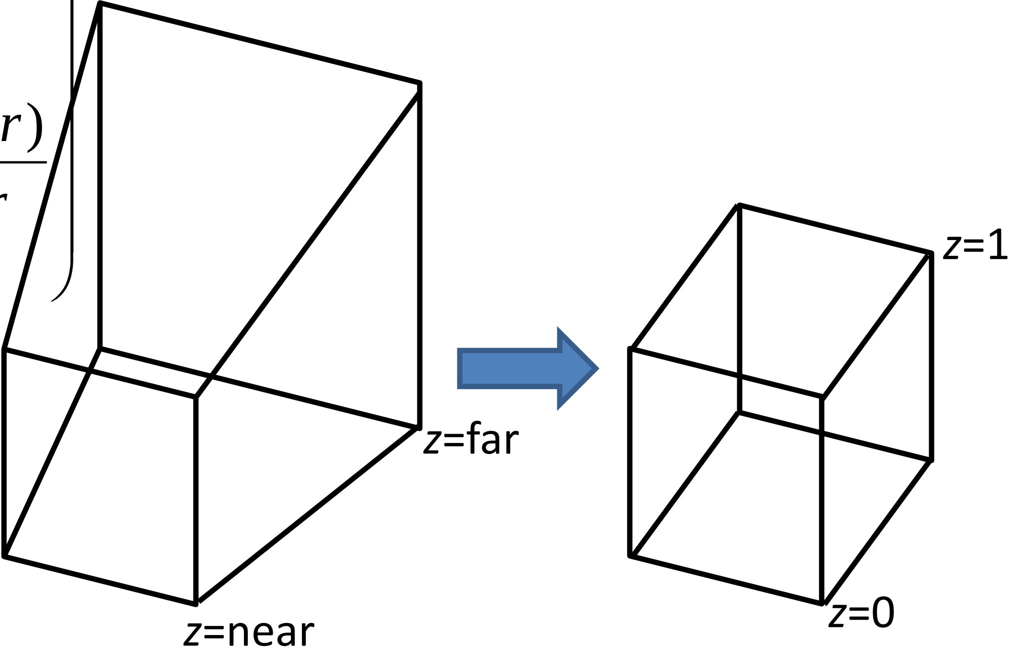


Linear Models

Perspective:

The mapping from 3D to 2D is performed by dividing through by the *z* position.

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} x/z \\ y/z \\ \dfrac{far(z-near)}{z(far-near)} \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ \dfrac{far(z-near)}{far-near} \\ z \end{pmatrix}$$



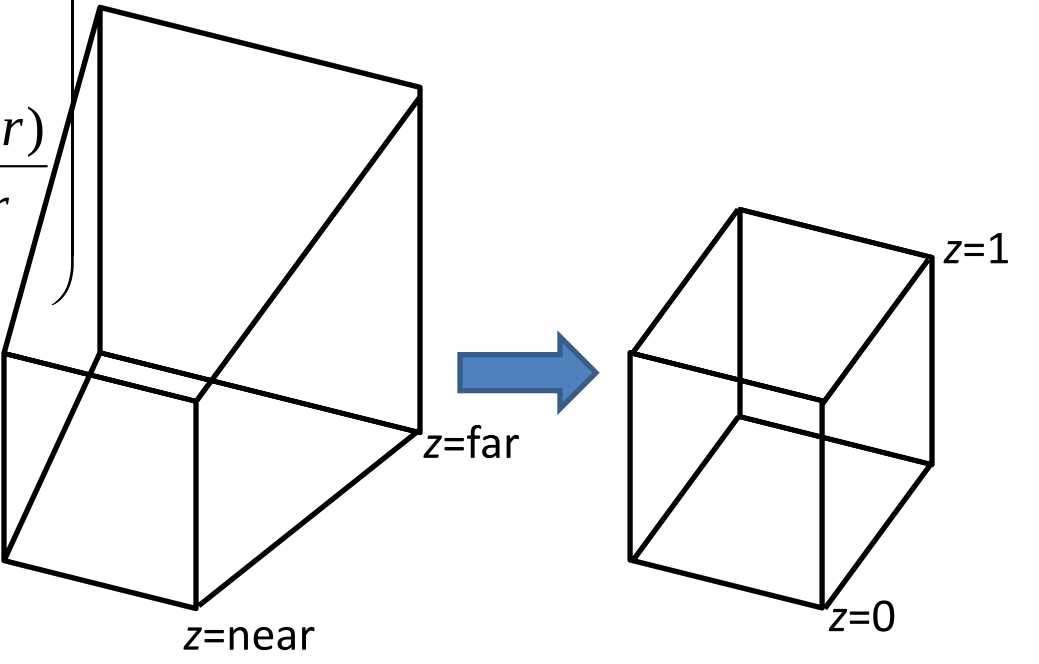
Linear Models

Perspective:

The mapping from 3D to 2D is performed by dividing through by the *z* position.

$$\begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} x/z \\ y/z \\ \dfrac{far(z-near)}{z(far-near)} \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ y \\ \dfrac{far(z-near)}{far-near} \\ z \end{pmatrix}$$

$$P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{far}{far-near} & \dfrac{-far\cdot near}{far-near} \\ 0 & 0 & 1 & 0 \end{pmatrix}$$



Linear Models

Perspective:

To generate a ray through pixel (*i*,*j*):

- Map to the front plane in camera-space:

$$(i, j) \quad \rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right)$$

Linear Models

Perspective:

To generate a ray through pixel (*i,j*):

- Map to the front plane in camera-space:

$$
\begin{aligned}
(i,j) \quad &\rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right) \\
&\rightarrow \quad (x,y,z) = \big(xStart + s(xStop - xStart),\ yStop + t(yStart - yStop), 0 \big)
\end{aligned}
$$

Linear Models

<u>Perspective</u>:

To generate a ray through pixel (*i,j*):

- Map to the front plane in camera-space:

$$
\begin{aligned}
(i, j) \quad &\rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right) \\
&\rightarrow \quad (x, y, z) = \big(xStart + s(xStop - xStart), yStop + t(yStart - yStop), 0\big) \\
&\rightarrow \quad (c_x, c_y, c_z) = \big(x \cdot near, y \cdot near, near, 1\big)
\end{aligned}
$$

Linear Models

Perspective:

To generate a ray through pixel (*i*,*j*):

- Map to the front plane in camera-space:

$$(i, j) \quad \rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right)$$

$$\rightarrow \quad (x, y, z) = \big(xStart + s(xStop - xStart), yStop + t(yStart - yStop), 0\big)$$

$$\rightarrow \quad (c_x, c_y, c_z) = (x \cdot near, y \cdot near, near, 1)$$

- Set the ray start position and direction:

$$ray.start = (0,0,0)$$

$$ray.dir = (c_x, c_y, c_z)$$

Linear Models

Perspective:

To generate a ray through pixel (*i,j*):

- Map to the front plane in camera-space:

$$
\begin{aligned}
(i, j) \quad &\rightarrow \quad (s,t) = \left(\frac{i}{width}, \frac{j}{height} \right) \\
&\rightarrow \quad (x, y, z) = \big(xStart + s(xStop - xStart), yStop + t(yStart - yStop), 0\big) \\
&\rightarrow \quad (c_x, c_y, c_z) = (x \cdot near, y \cdot near, near, 1)
\end{aligned}
$$

- Set the ray start position and direction:

$$ray.start = (0,0,0)$$

$$ray.dir = (c_x, c_y, c_z)$$

- Transform the ray to world space.

Non-Linear Models

Depth of Field:

In practice, cameras are not pinhole.
They have a finite aperture and a lens that defines a focal plane.

Close Focused

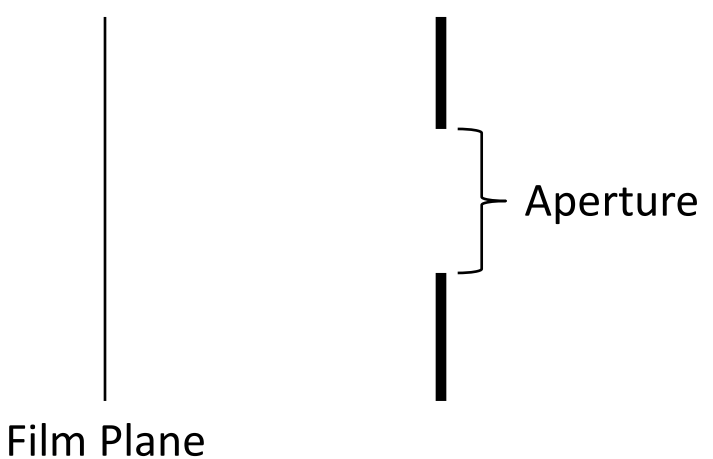
Distance Focused

[P. Haeberli]

Non-Linear Models

Depth of Field:

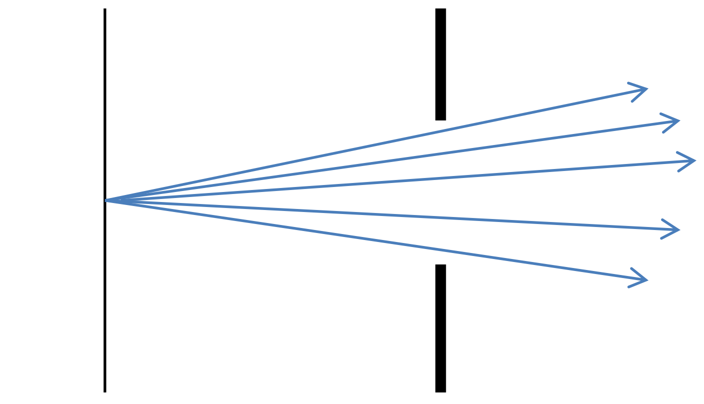
- Finite aperture



Non-Linear Models

<u>Depth of Field</u>:

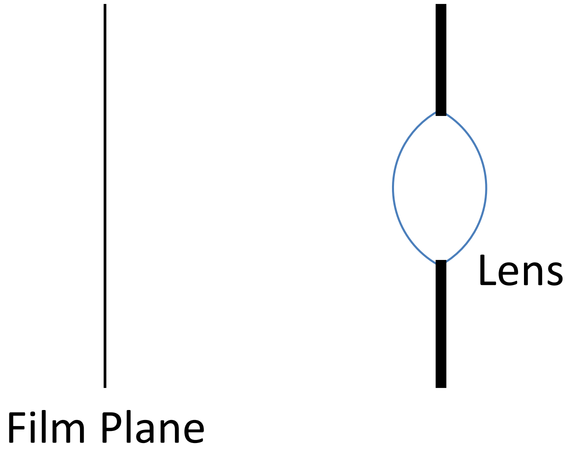
- Finite aperture
  - Many rays falling on a single point in the film plane results in blurring.



Non-Linear Models

Depth of Field:

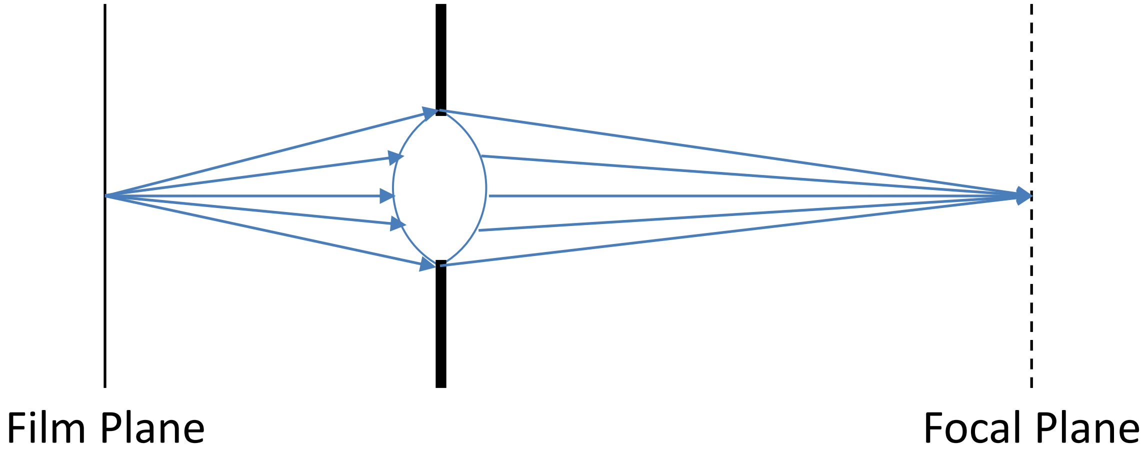
- Finite aperture
- Lens



Non-Linear Models

Depth of Field:

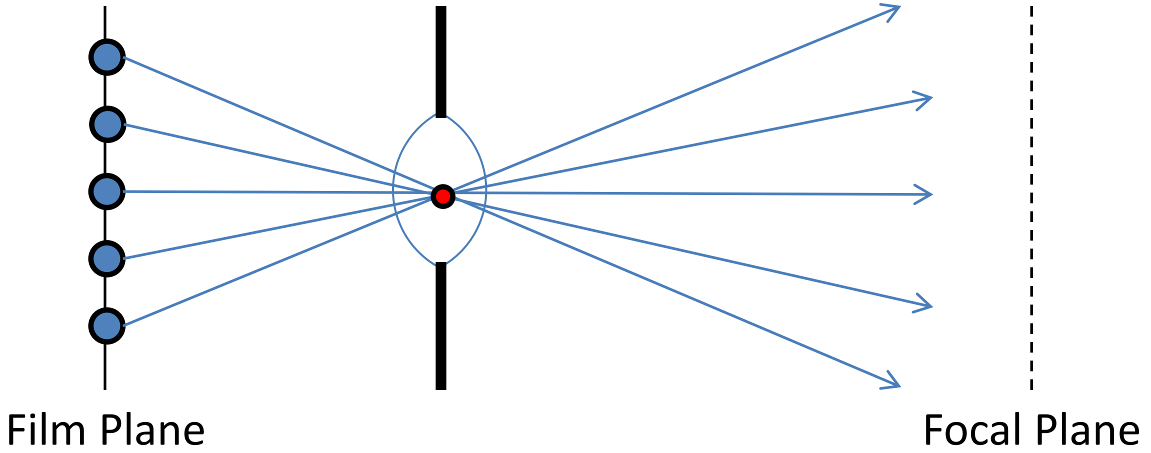
- Finite aperture
- Lens
  - All rays emanating from a point on the film will intersect on a common (focal) plane.



Non-Linear Models

Simulating Depth of Field:

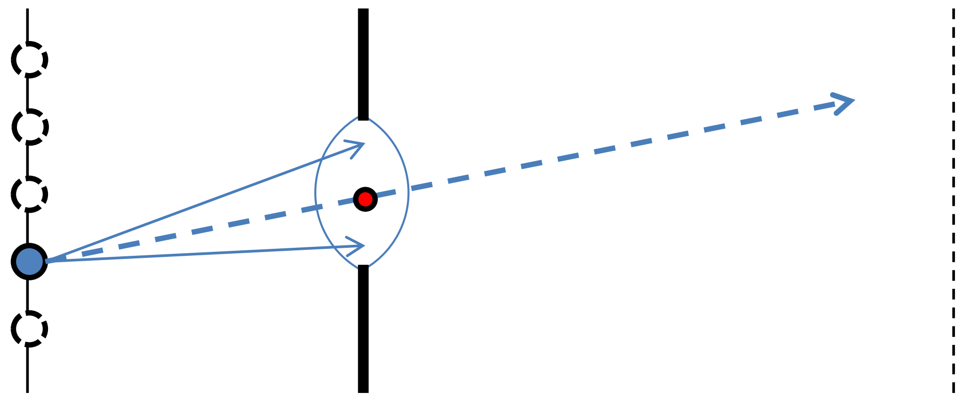
– For each point on the film plane construct a primary ray through the center of the aperture.



Non-Linear Models

Simulating Depth of Field:

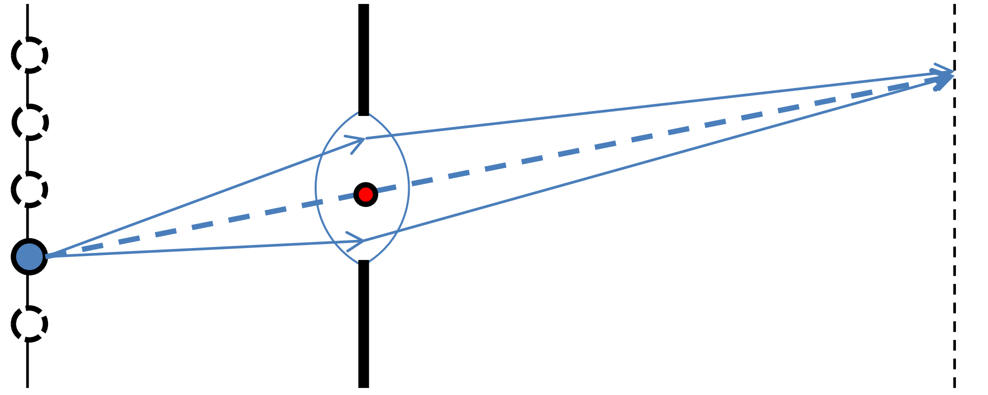
- For each point on the film plane construct a primary ray through the center of the aperture.
- Generate random rays through the lens



Non-Linear Models

<u>Simulating Depth of Field</u>:

- For each point on the film plane construct a primary ray through the center of the aperture.
- Generate random rays through the lens
- These rays meet at the film-plane



Non-Linear Models

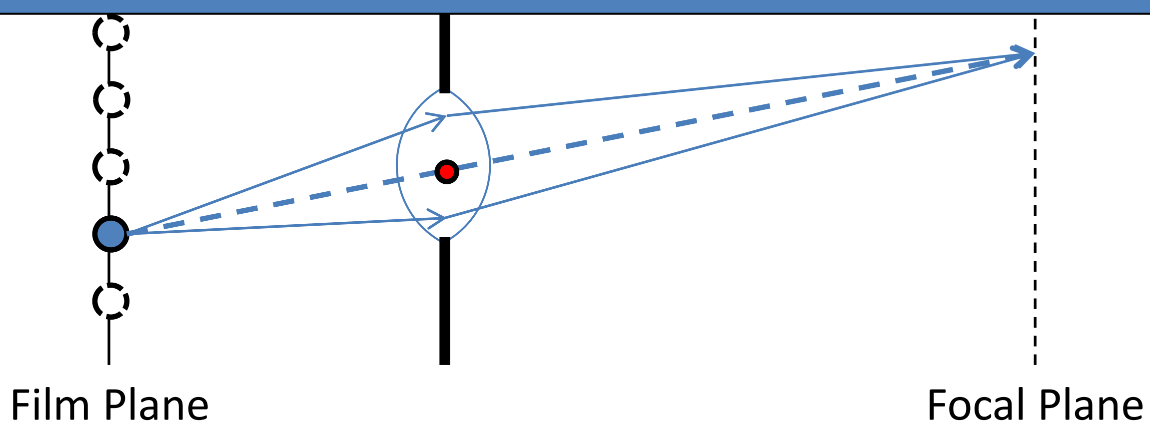
Simulating Depth of Field:

– For each point on the film plane construct a primary ray through the center of the aperture.

– Generate random rays through the lens

–

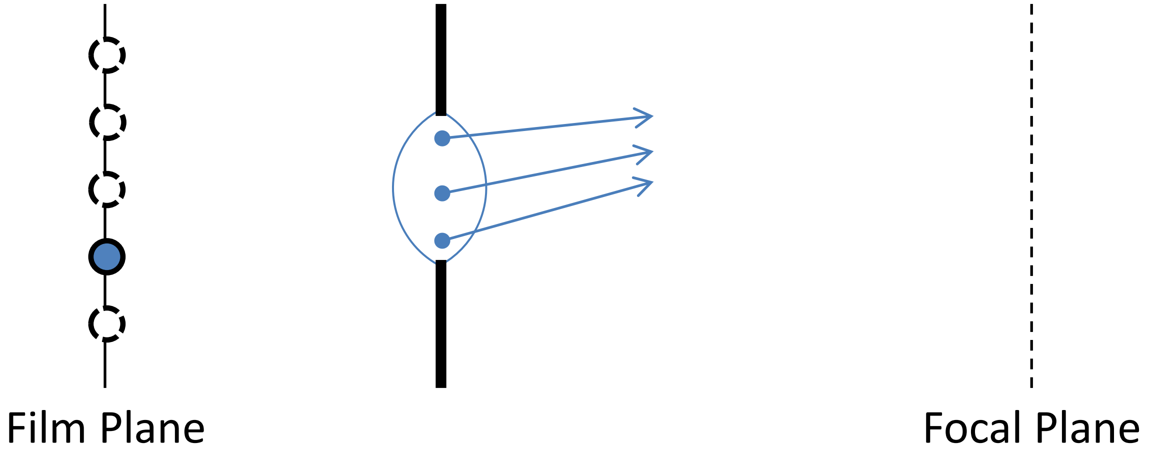
Since the primary ray travels in a straight line, we can get the point of intersection by intersecting the ray with the plane



Non-Linear Models

<u>Simulating Depth of Field</u>:

- For each point on the film plane construct a primary ray through the center of the aperture.
- Generate random rays through the lens
- These rays meet at the film-plane
- Average over the new rays starting on the lens.

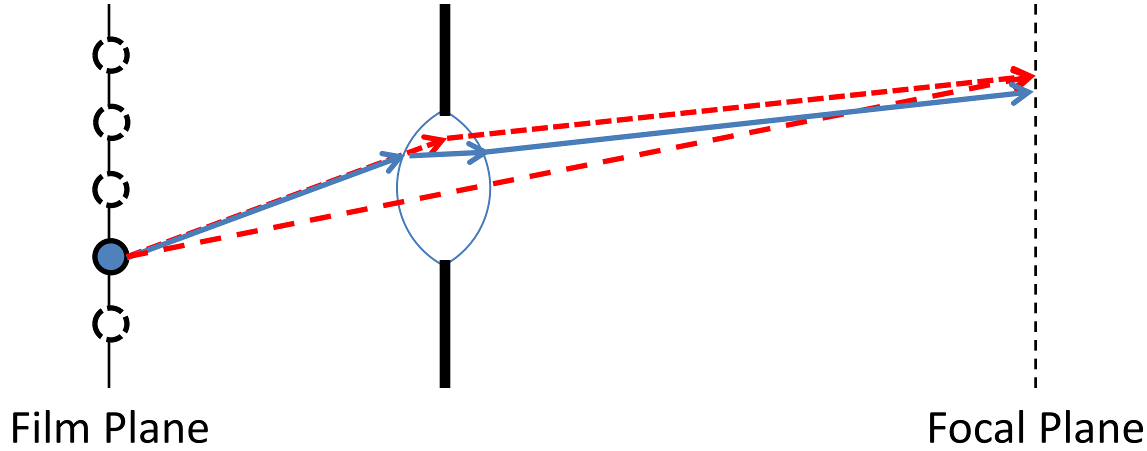


Non-Linear Models

<u>Simulating Depth of Field</u>:

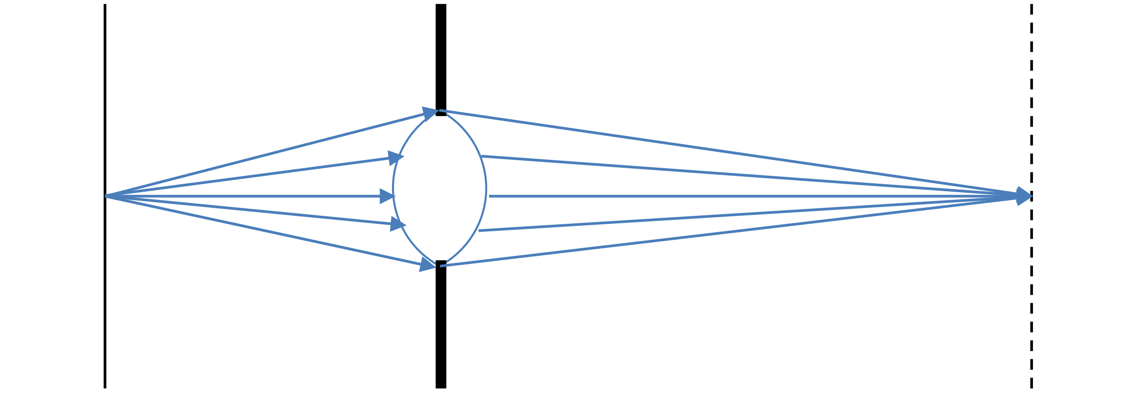
Note that this is an approximation based on the assumption that the lens is thin.

In practice, the ray will refract twice when passing through the lens.



Depth of Field

Why/When is it true that rays emanating from a common point on the film plane intersect on a common point on the focal plane?



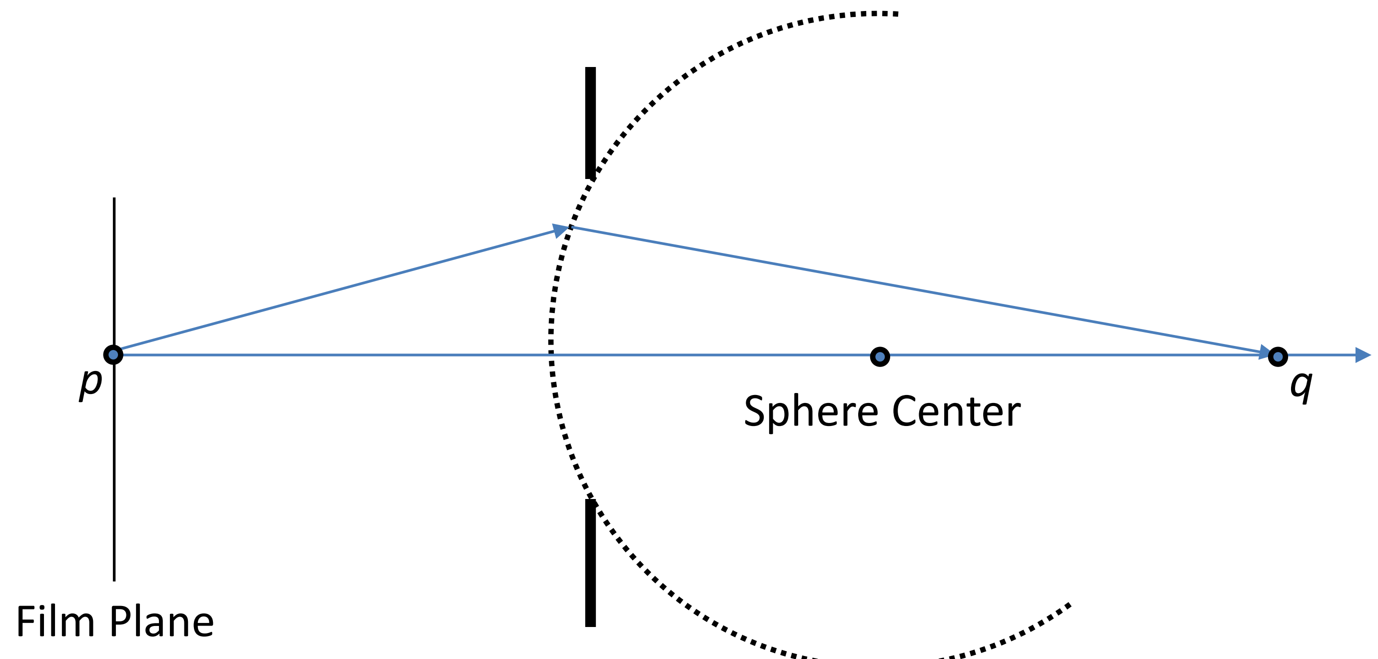
Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

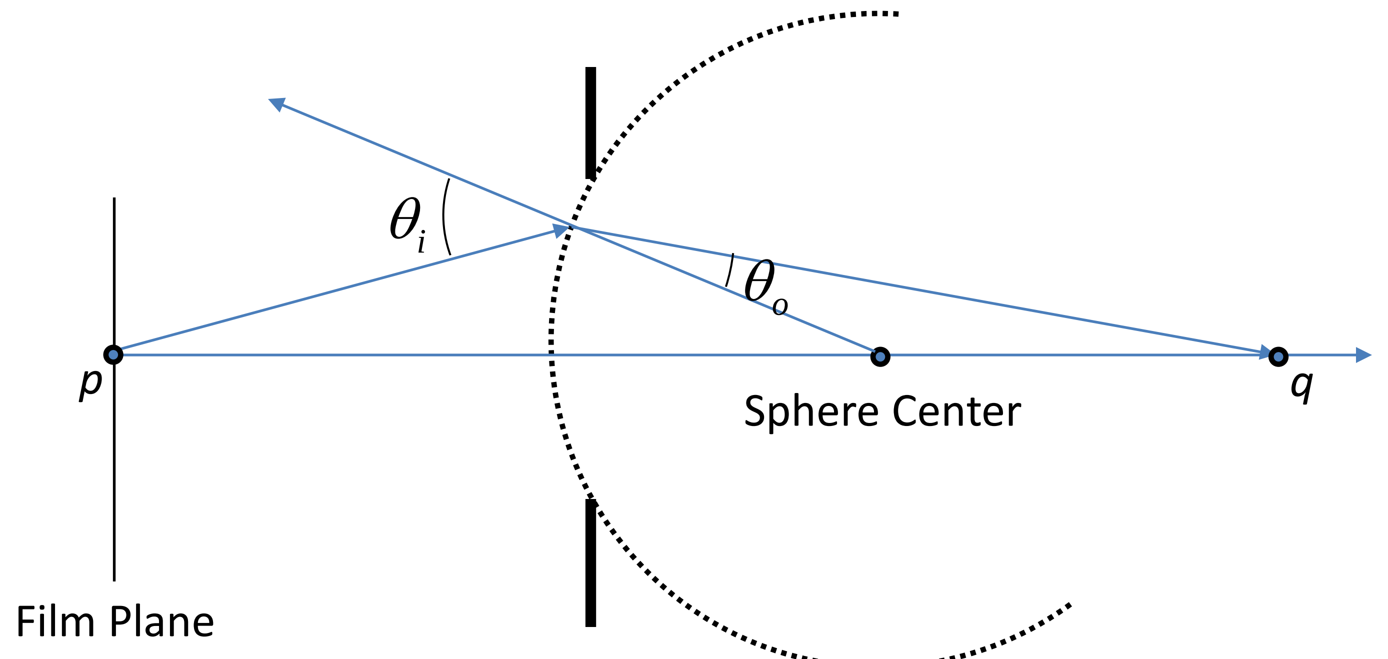
Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i \sin\theta_i = \eta_o \sin\theta_o$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

Consider light entering a spherical lens.

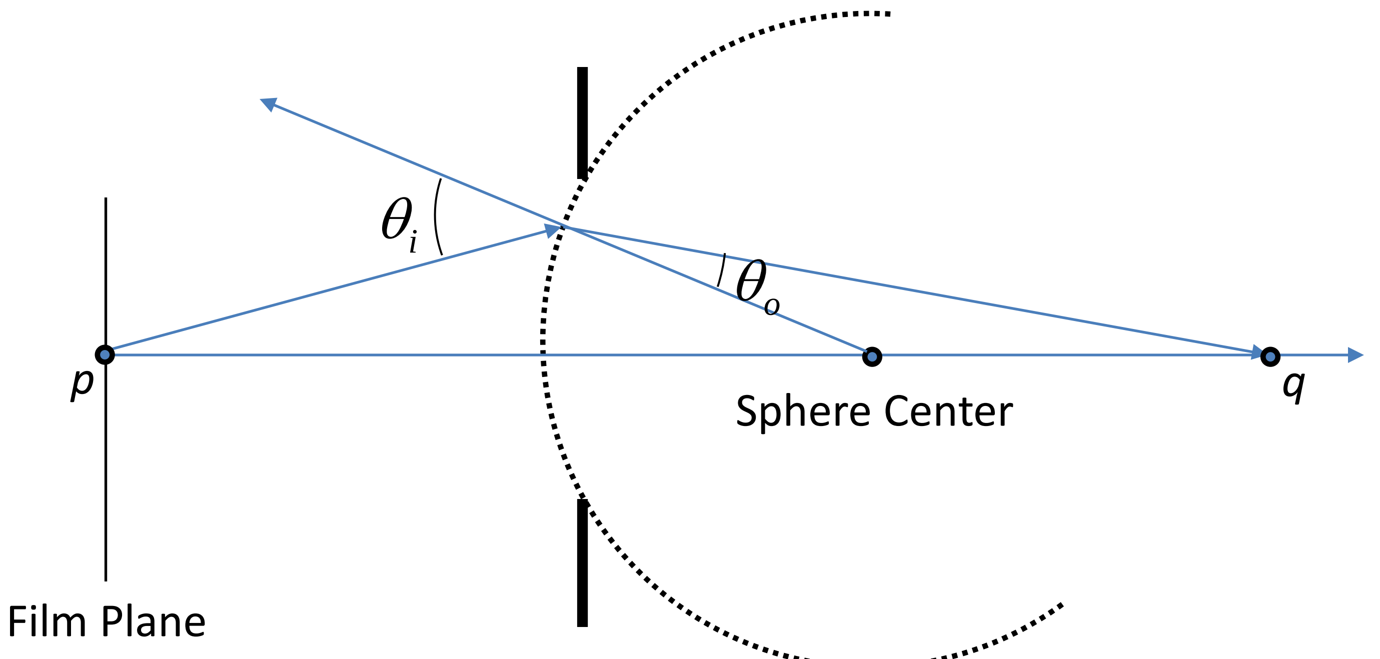
Assume:

1. **Sphere radius is large relative to aperture**
2. **Distance to film is large relative to aperture**

Snell's Law:

$$\eta_i \sin\theta_i = \eta_o \sin\theta_o$$

$$\eta_i\theta_i \approx \eta_o\theta_o$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

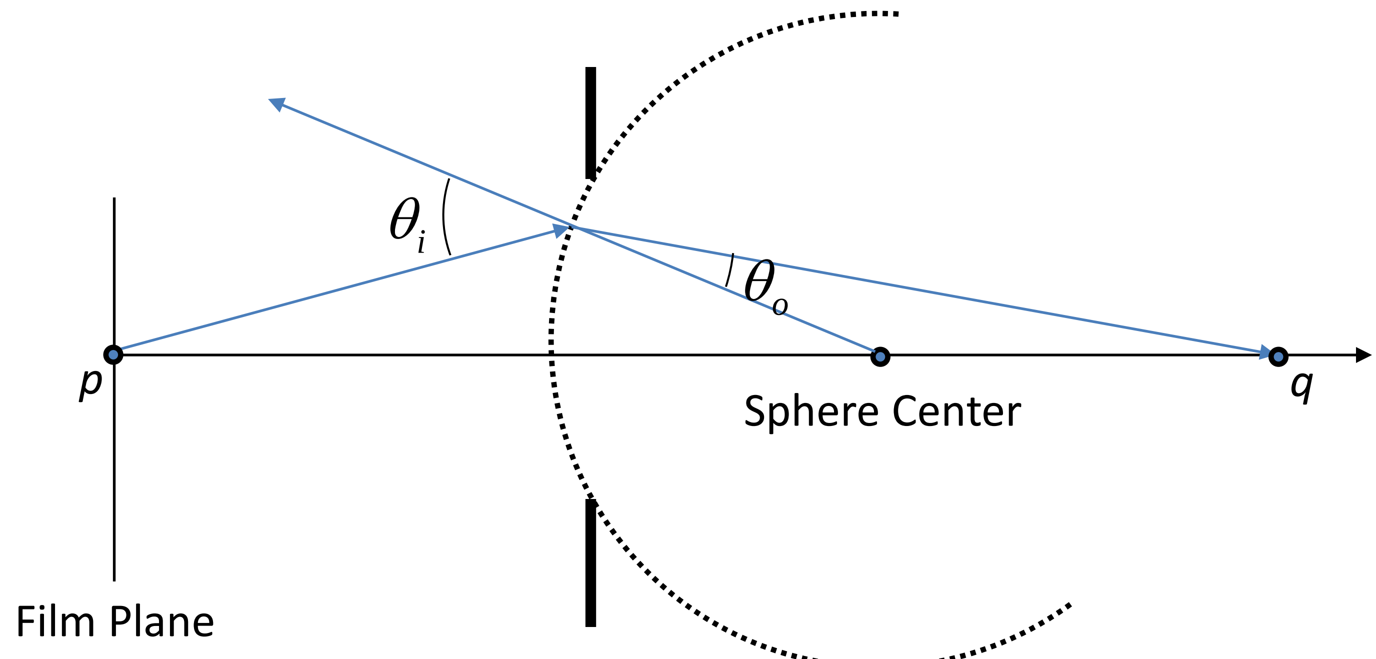
Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i \theta_i \approx \eta_o \theta_o$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

Consider light entering a spherical lens.

Assume:

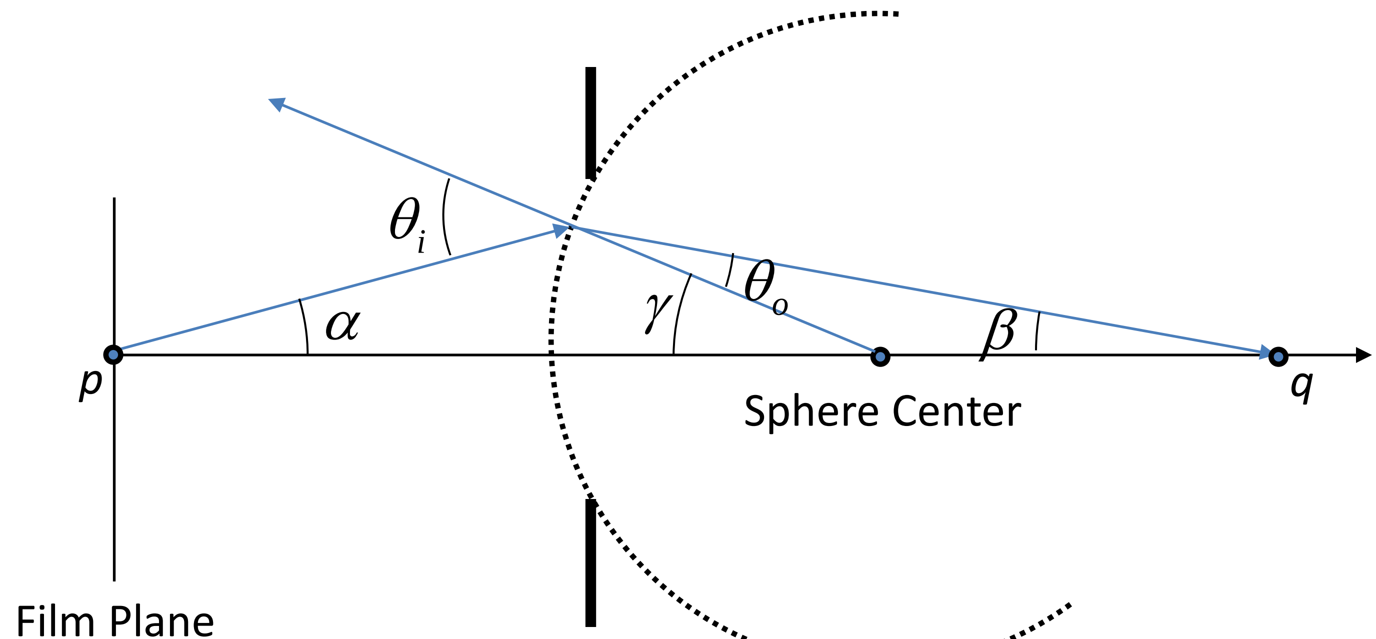
1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i \theta_i \approx \eta_o \theta_o$$

$$\theta_i = \alpha + \gamma$$

$$\theta_o = \gamma - \beta$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

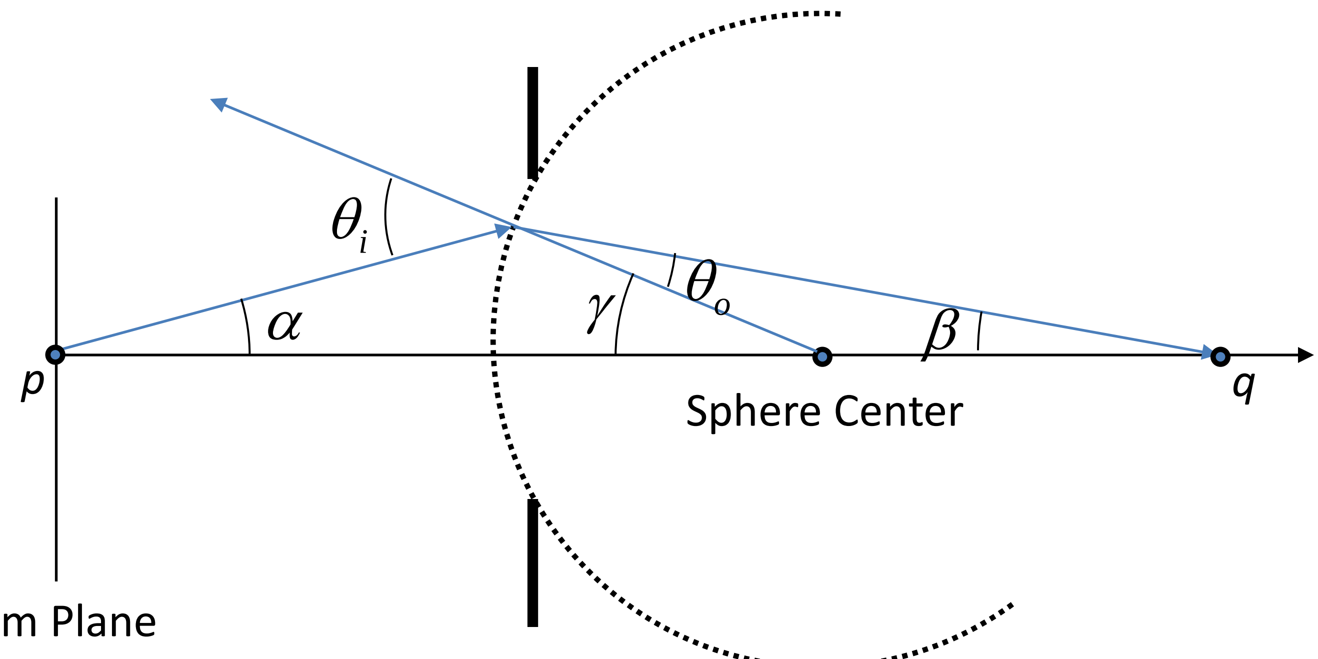
Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i(\alpha+\gamma)\approx\eta_o(\gamma-\beta)$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

Consider light entering a spherical lens.

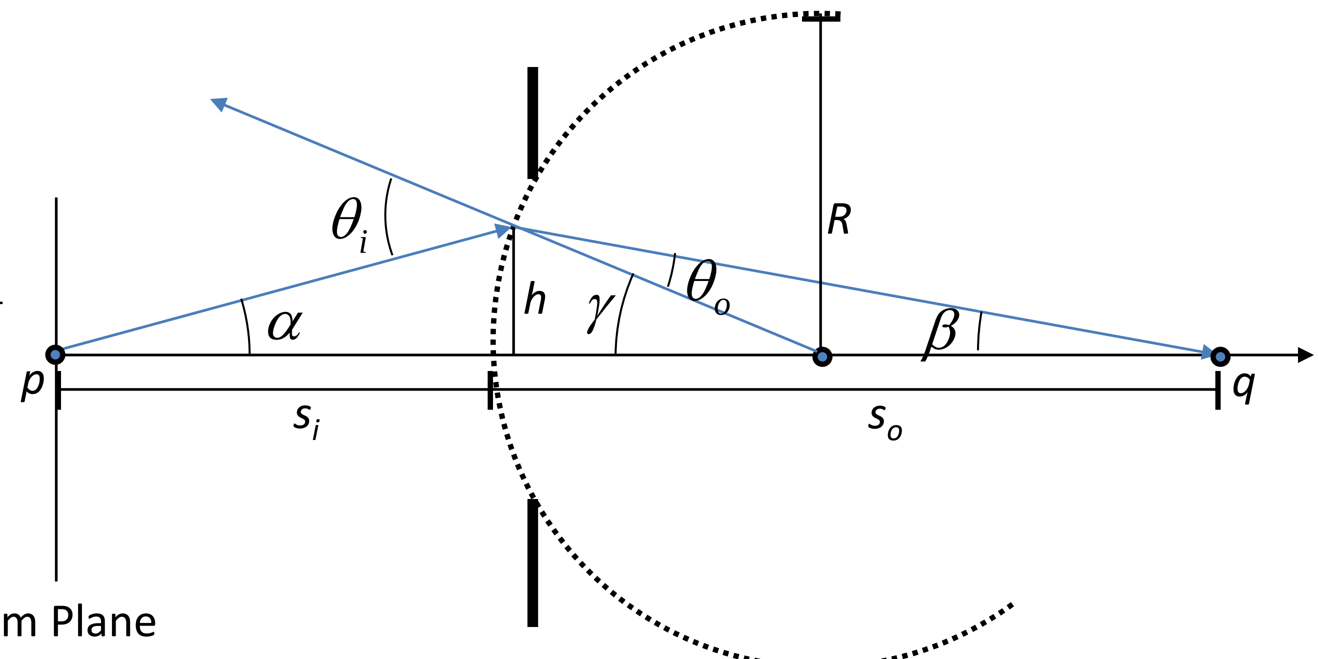
Assume:

1. **Sphere radius is large relative to aperture**
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i(\alpha + \gamma) \approx \eta_o(\gamma - \beta)$$

$$\alpha \approx \frac{h}{s_i}, \quad \beta \approx \frac{h}{s_o}, \quad \gamma \approx \frac{h}{R}$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

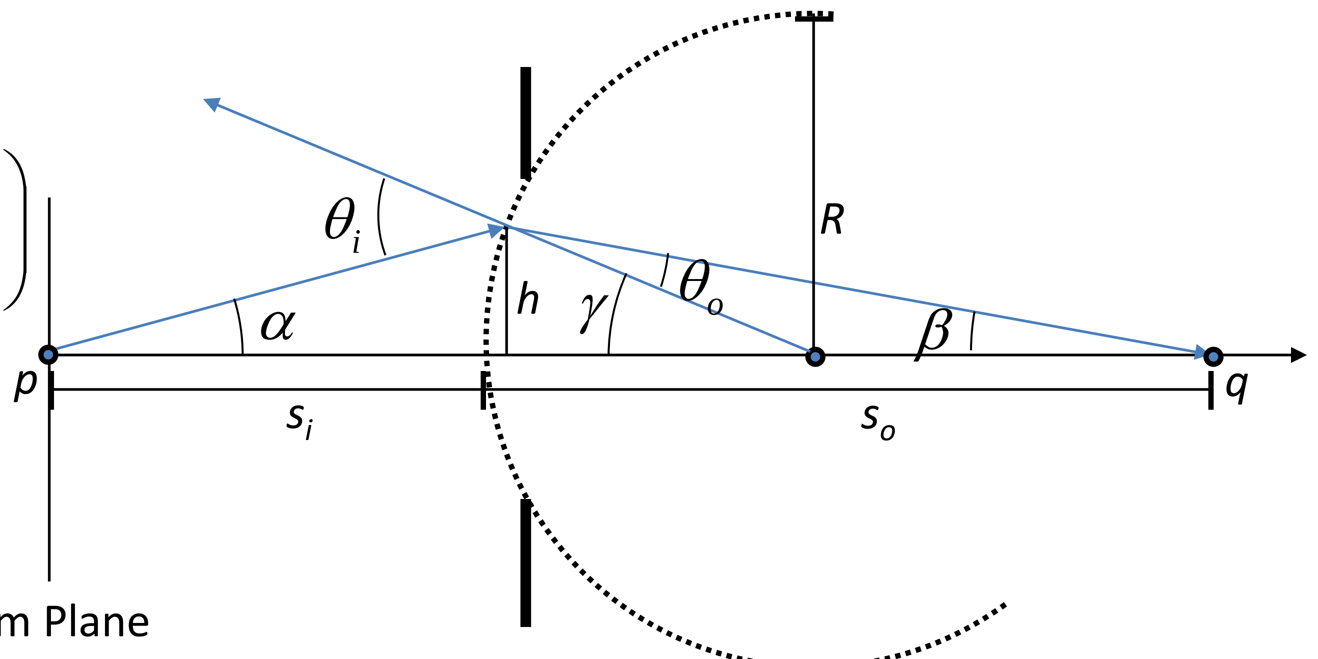
Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i\left(\frac{h}{s_i}+\frac{h}{R}\right)\approx\eta_o\left(\frac{h}{R}-\frac{h}{s_o}\right)$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

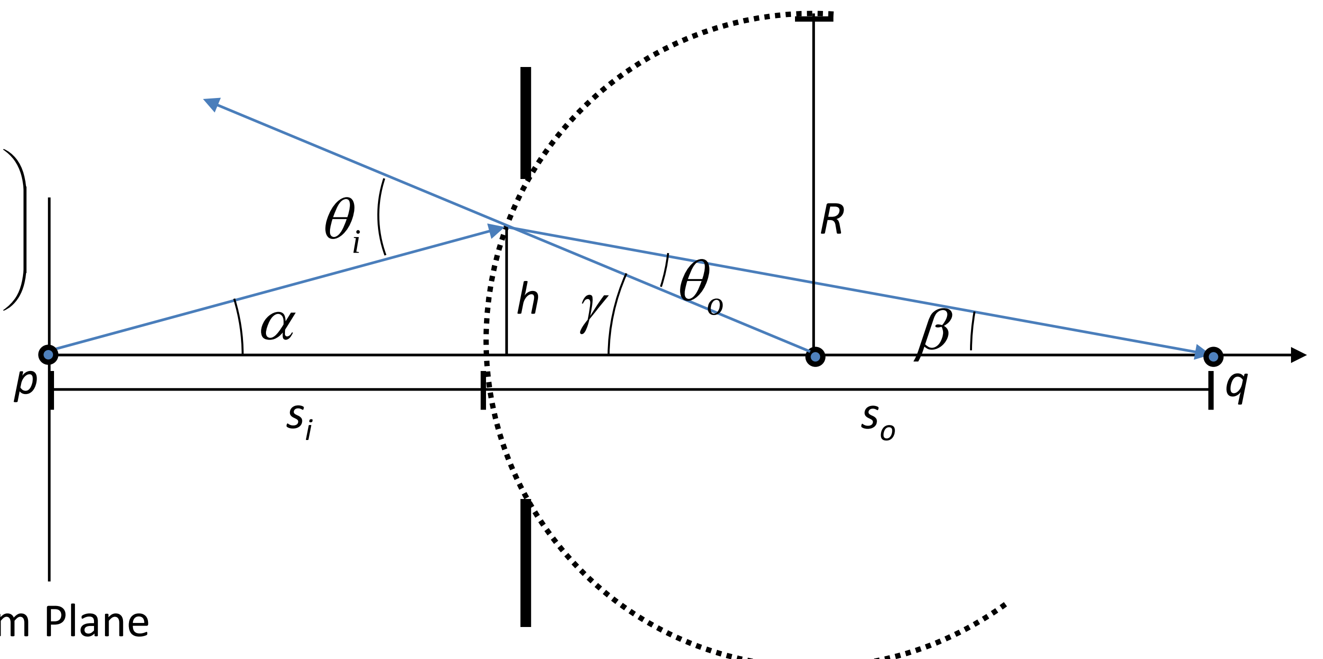
Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i\left(\frac{1}{s_i}+\frac{1}{R}\right)\approx\eta_o\left(\frac{1}{R}-\frac{1}{s_o}\right)$$



Paraxial Approximation

[Using slides from Pat Hanarhan and Marc Levoy]

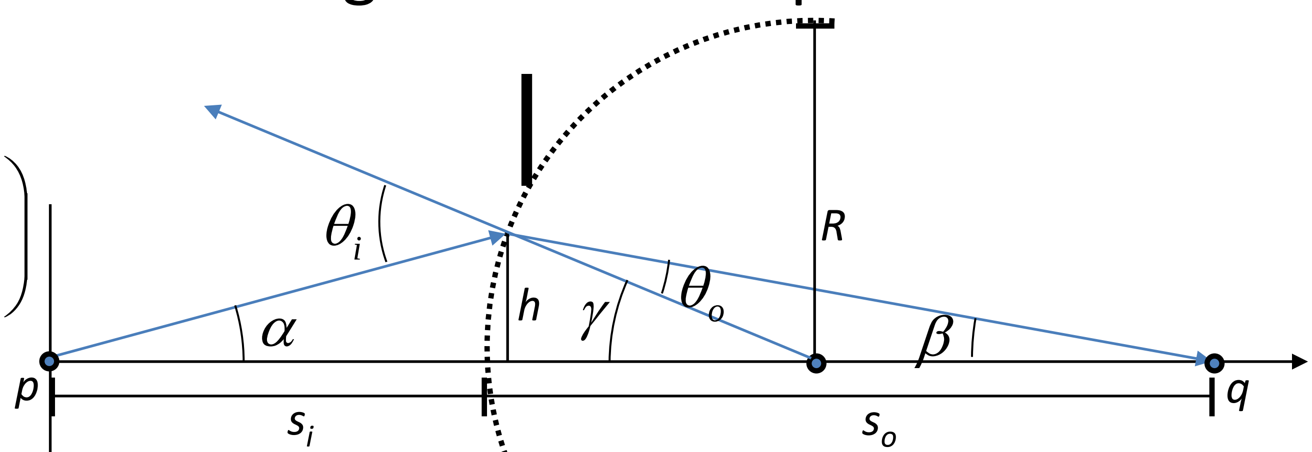
Consider light entering a spherical lens.

Assume:

1. Sphere radius is large relative to aperture
2. Distance to film is large relative to aperture

Snell's Law:

$$\eta_i\left(\frac{1}{s_i}+\frac{1}{R}\right)\approx\eta_o\left(\frac{1}{R}-\frac{1}{s_o}\right)$$



The distance s_o only depends on s_i and not on h.
So all rays leaving p pass through q, regardless of angle.