DOING PHYSICS WITH PYTHON

COMPUTATIONAL OPTICS

RAY (GEOMETRIC) OPTICS

MATRIX METHODS IN PARAXIAL OPTICS

CONVERGING THIN LENS

Ian Cooper

Please email me any corrections, comments, suggestions or additions: **matlabvisualphysics@gmail.com**

DOWNLOAD DIRECTORIES FOR PYTHON CODE

[Google drive]

[GitHub]

S004A.py

S004.py

The Python Code **S004.py** applies a set of ABCD matrices in the paraxial approximation to calculate the ray path from a point in the input (object) plane, through a converging thin lens to a point in the output (observation plane). The transformation sequence of matrices applies are:

Translation 0 → 1

Initial height $y_0 = y[0]$

Elevation (slope angle in radians) $a_0 = a[0]$

Distance: object point to lens $L01 \equiv s_0$

Refractive index of air $n_0 = 1.00$

Height $y[1]$

Elevation $a[1]$

Refraction 1 → 2

Position of lens $x_{lens} = 0$

Refractive index of lens $n_1 = 1.00$

Radius convex spherical surface of lens $R_0 > 0$

Height $y[2] = y[1]$

Elevation $a[2]$

Refraction 2 → 3

Radius concave spherical surface of lens $R_1 < 0$

Height $y[3] = y[2] = y[1]$

Elevation $a[3]$

Translation 3 → 4

Distance lens to observation point $L34 \equiv s_1$

Height $y4 = y[4]$

Elevation $a[4] = a[3]$

POSITIVE CONVERGING THIN LENS

A positive converging lens with convex spherical surfaces (lens is thicker at the centre than at the edges) is shaped so that all light rays that enter it parallel to its optical axis intersect at a single point called the **focus**. The focus is located on the optical axis and on the opposite side of the lens. A converging lens can form **real** or **virtual**, **inverted** or **upright**, and **magnified** or **diminished** images.

The **thin lens equation** relates the focal length f of a lens to the object distance s_0 and image distance s_1.

(1) $$\frac{1}{f} = \frac{1}{s_0} + \frac{1}{s_1}$$

The magnification m of a thin lens describes the ratio of the image size h_1 to the object size h_0. A magnification greater than 1 indicates the image is larger than the object, while a magnification less than 1 means the image is smaller. The sign of the magnification indicates whether the image is inverted (negative) or upright (positive).

(2) $$m = \frac{h_1}{h_0} = -\frac{s_1}{s_0}$$

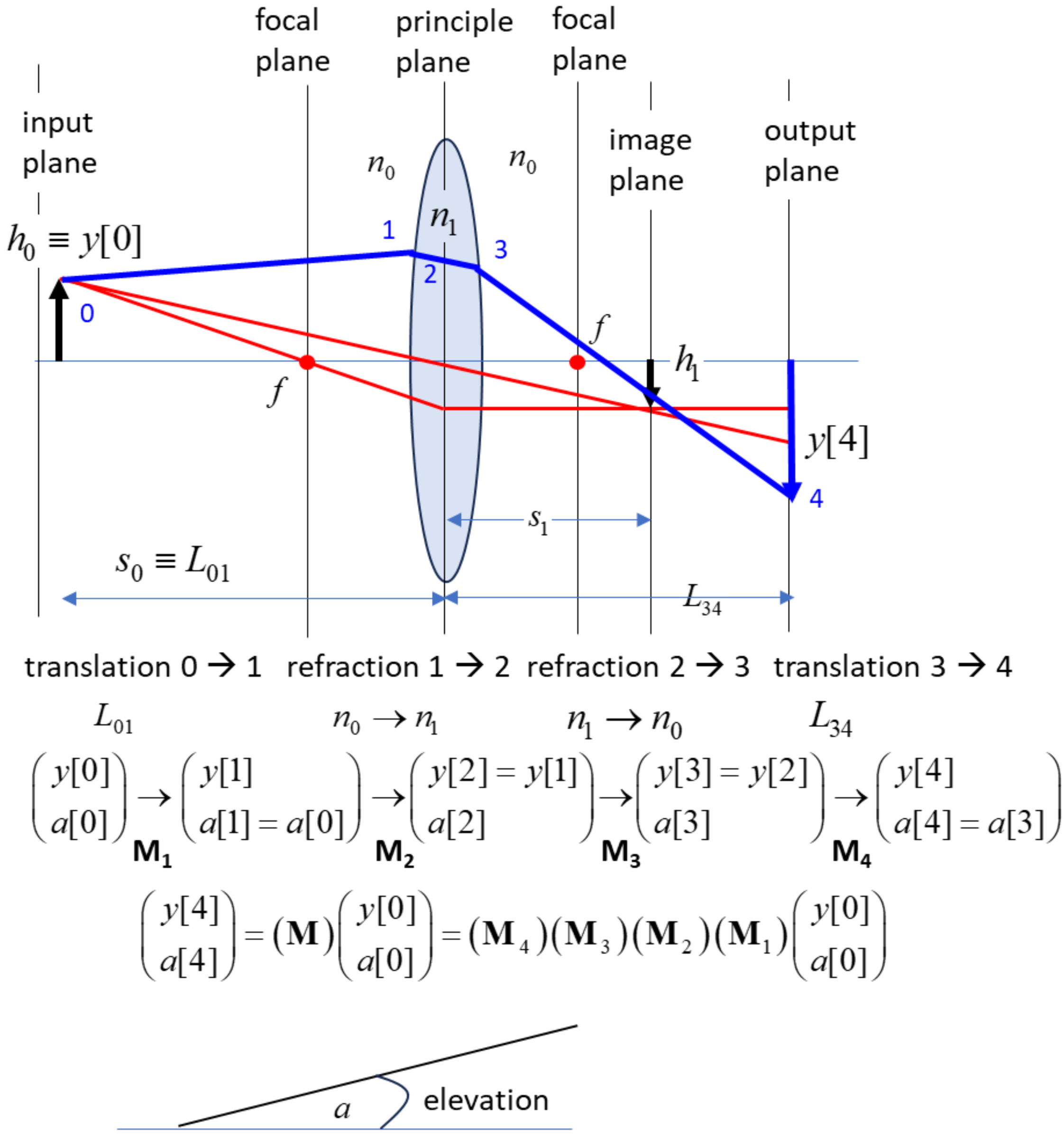
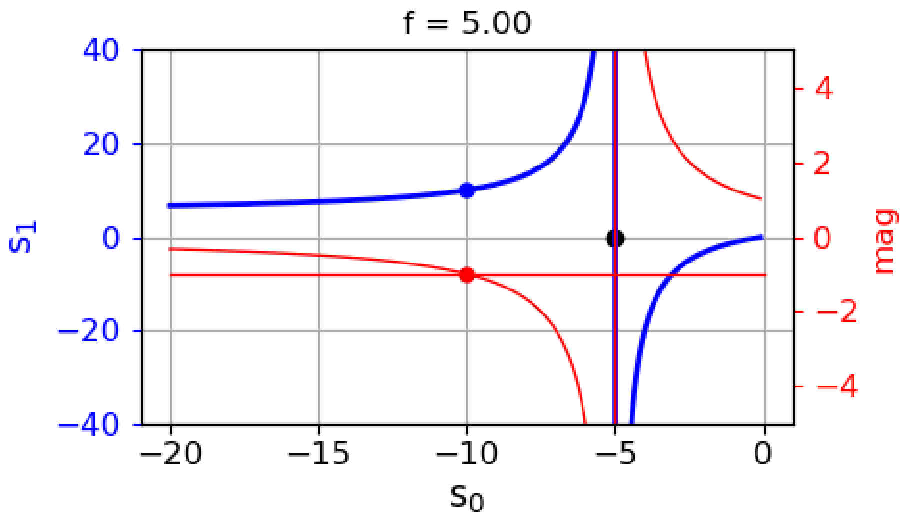


Fig. 1. Geometry of thin lens optical system and the transformation matrices.

The Python Code **S004.py** can be used to plot the ray path through a thin lens optical system as indicated in figure 1. The model parameters are displayed in the Console Window.

Figure 2 shows plots of the image distance s_1 and magnification mag as a function of the object distance s_0. Understanding this diagram completely explains the significance of equations 1 and 2.



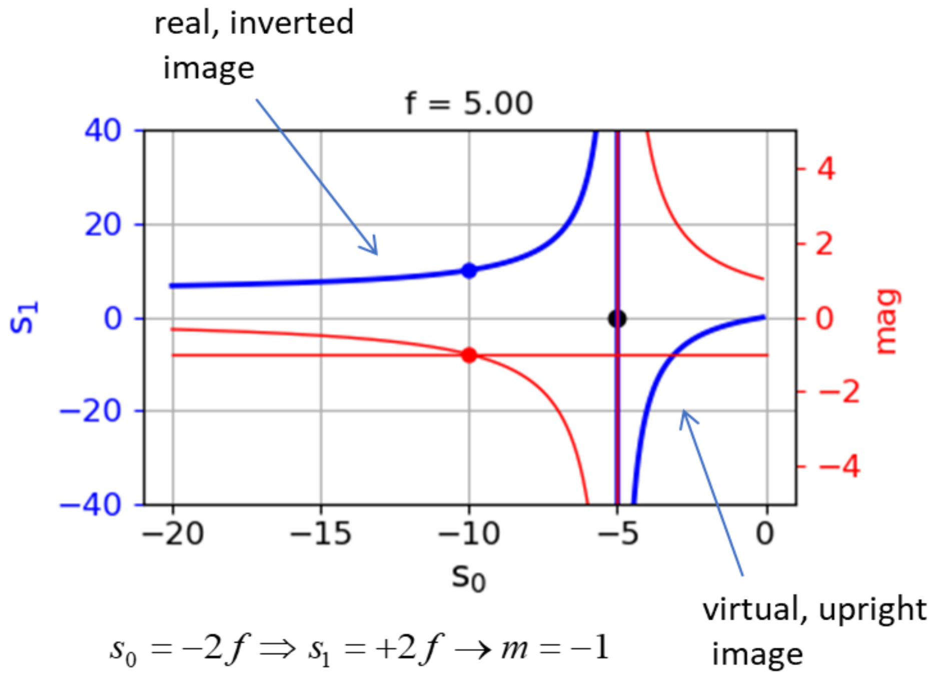


Fig. 2. Image distance s_1 and magnification mag as functions of the object distance s_0. **S004A.py**

Visit the link for details of the ABCD matrix method

[MATRIX METHODS FOR RAY PROPAGATION]

The system matrix **M** for the ray path 0 → 4 gives

$$\begin{pmatrix} y_4 \\ \alpha_4 \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} y_0 \\ \alpha_0 \end{pmatrix}$$

$$y_4 = A\,y_0 + B\,\alpha_0$$

$$\alpha_4 = C\,y_0 + D\,\alpha_0$$

Only when B = 0 do all the rays from an input (object) point converge to a single image point since the output point y_4 is independent of the angle of the ray from 0 → 1. In running the Code **S004.py** , only when the image plane is at the same distance from the lens as the object plane ($s_1 = L34$) does B = 0 and the rays from the input point converge to the single image point. When B = 0, then the magnification of the image is given by the matrix element A.

The results of running the Code for different input parameters are shown in the following figures.

To obtain multiple input rays, the Code is first run and then segments of the Code are run separately to superimpose additions plots.

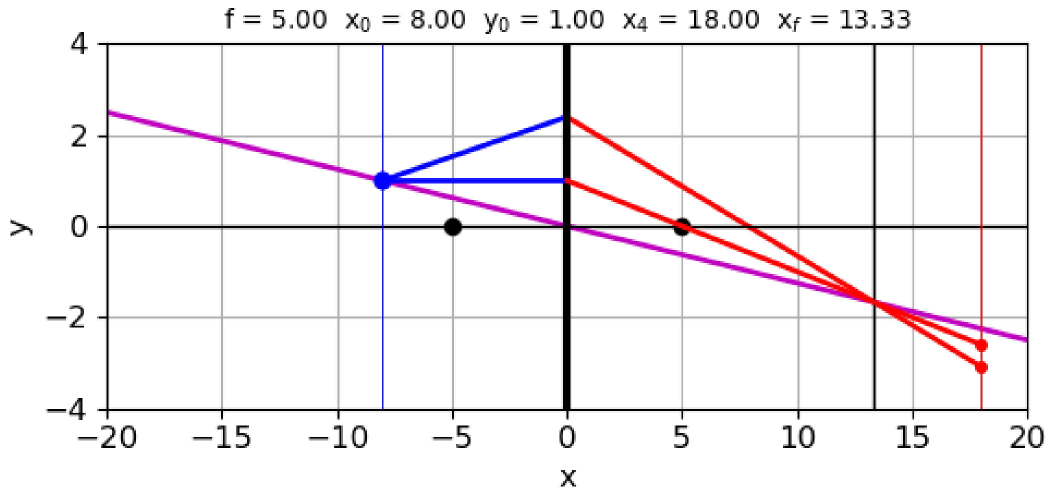


Fig. 3. The input ray 0 → 1, input point, and input plane are shown in **blue** $(x_0 = 8)$. The focal points, the plane of the lens and the image plane are shown in **black** $(x_{lens} = 0 \;\; s_1 = x_f = 13.33)$. The output rays, the output points and output plane are shown in **red** $(x_4 = 18)$. The **magenta** line is the ray from the input point through the centre of the lens. The matrix element B is not zero ($B \neq 0$), therefore, the rays from the input point do not intersect the output plane at a single point. If the output plane is placed at the image plane where $L34 = x_f = 13.33$ then all the rays from the input point converge to a single point. For an object located in the region between f and 2*f*, then the image is real, inverted and with a magnification greater than 1, $m > 1$ (see figure 2).

A sample of the Console Window summary is shown below for one of the inputs of figure 3.

S004.py

INPUTS

refractive index of air, n0 = 1.00

refractive index of lens n1 = 1.50

focal length, f = 5.00

radius convex front spherical surface of lens, R0 = 4.00

Object plane, L01 = 8.00

Object height, y0 = 1.00

Incident ray elevation, a01 = 0.00 deg

Translation distance 3 --> 4, L34 = 18.00

OUTPUTS

radius concave rear spherical surface of lens, R1 = -6.67

refracted ray elevation, a34 = -11.46

image plane,   x4 = 13.33

image height, y4 = -1.67

image plane, x4 = 13.33

lateral magnification, m = -2.60

M --> A = -2.60 B = -2.80 C = -0.20 B = -0.60

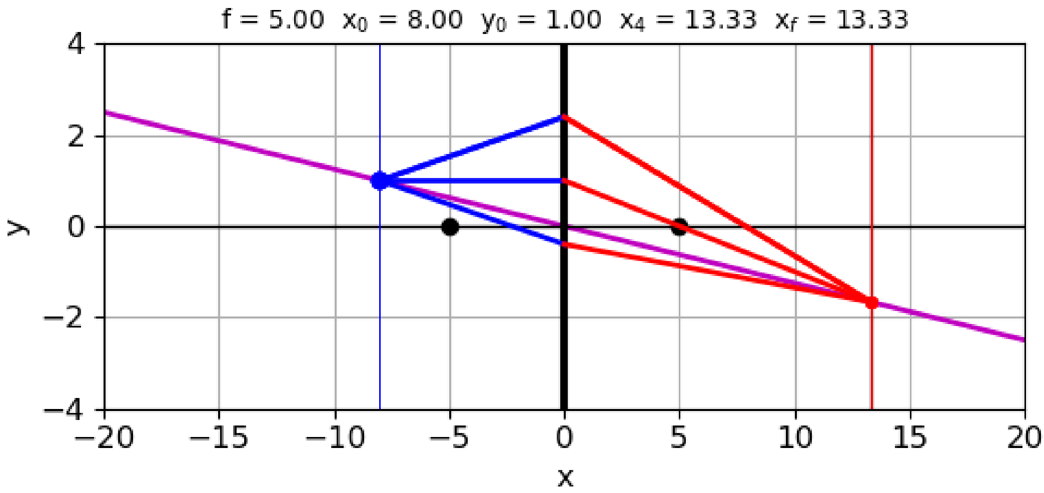


Fig. 4. The image plane and the object plane coincide and B = 0, therefore all rays from the input point converge to a single point in the image plane. The matrix element A = -1.67, which is the magnification. **S004.py**

When the input point is at a distance L01 > 2f and the output point is located in the image plane, then the image is real, inverted and diminished in height (figure 5).

INPUTS
Object plane, L01 = 13.00; Object height, y0 = 1.00
Incident ray elevation, a01 = 10.00 deg
Translation distance 3 --> 4, L34 = 8.12
OUTPUTS
refracted ray elevation, a34 = -27.46
image plane, x4 = 8.12
image height, y4 = -0.62
lateral magnification, m = -0.63
M --> **A = -0.62 B = 0.00** C = -0.20 B = -1.60

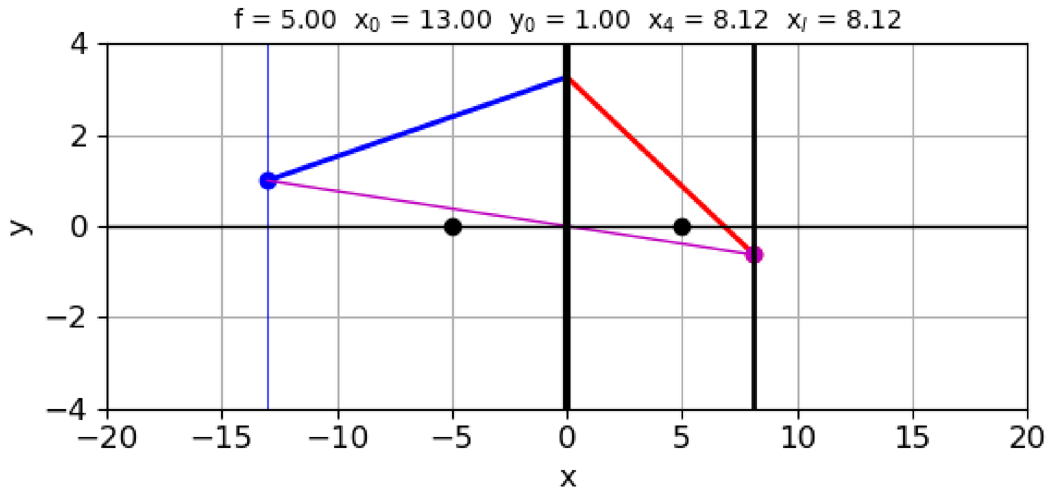


Fig. 5. When the input point is located at a distance greater than $2f$ from the lens, and the output point is in the image plane, then the image is real, inverted and diminished in height (mag = -0.63). **S004.py**

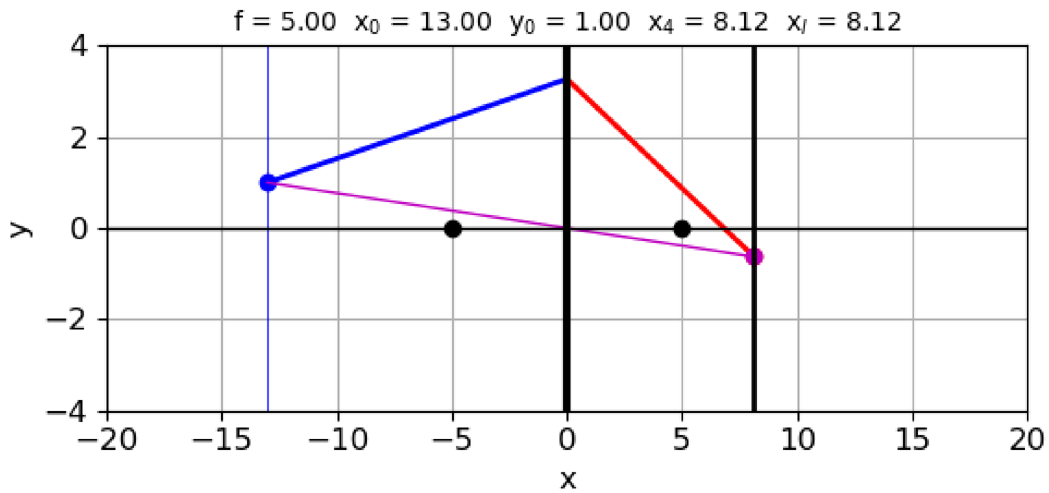


Fig. 6. When the input point is located at the distance $2f$ from the lens, and the output point is in the image plane, then the image is real, inverted and has the same height (mag = -1.00). **S004.py**

When the object point is within the focal distance

$(s_0 = L_{01} < f)$ then the image will be **virtual**, **upright** and

enlarged (figure 7).

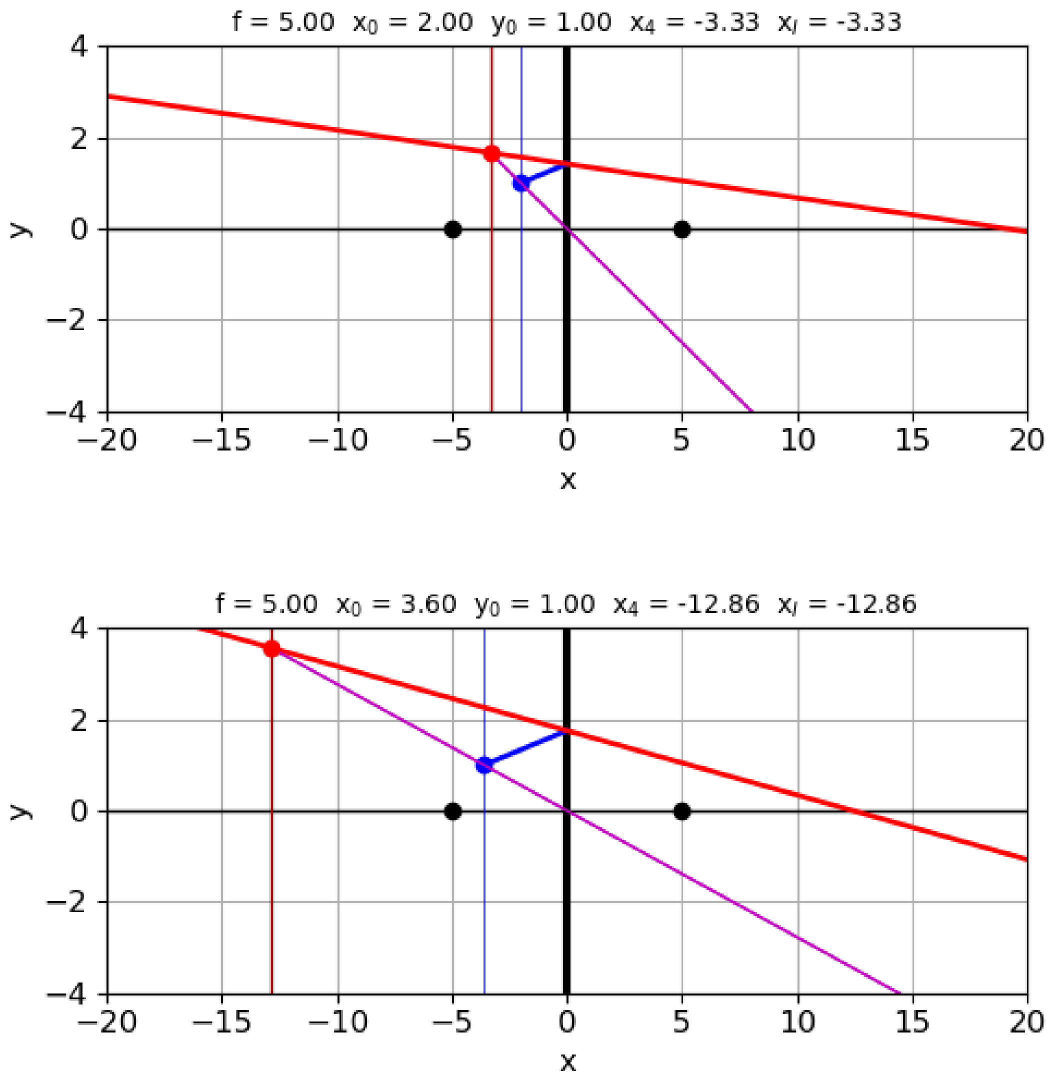


Fig. 7. The object point is within the focal distance

$(s_0 = L_{01} < f)$. The image will be virtual, upright and enlarged.

The lateral magnification of the image increases rapidly as the

object point moves closer to the focal point at x = -5.

Top figure m = +1.67, bottom figure m = +3.57. **S004.py**