Simulation of Real Photographic Phenomena in Computer Graphics

Benjamin Steinert
May 4, 2009

Overview

Contents

Some Real Photographs

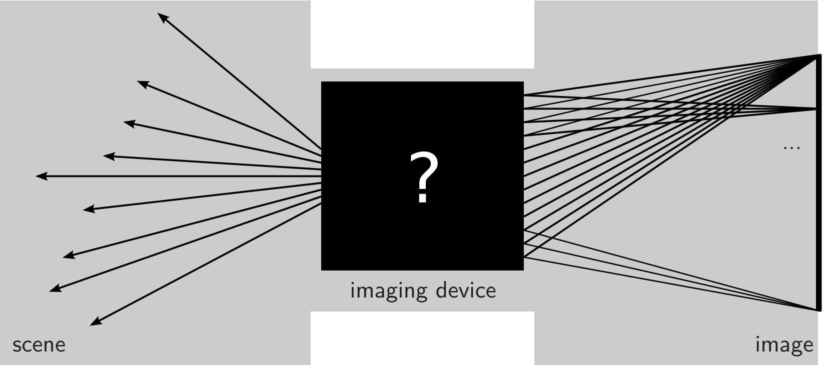
Some Real Photographs

Some Real Photographs

interdisciplinary Toolbox

- Optical Physics
  - Properties of Light and Units
  - Light Propagation and Interaction
- Optical Engineering
  - Lens Design
  - Glass Science
  - Aberration Theory
- Computer Graphics
  - data representation
  - physically based rendering
  - sampling
- Photography

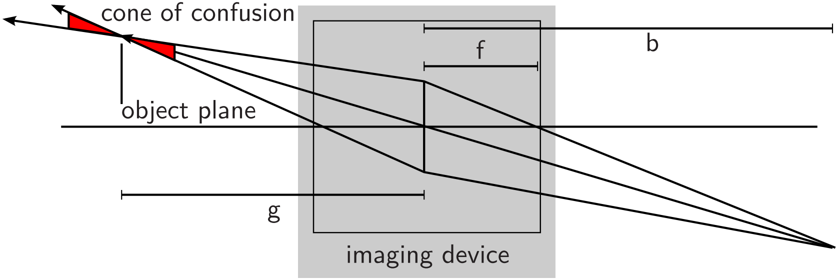
Rendering fragmented into different concerns



- scene: acceleration structures, materials, sampling
- image: sampling, (post) processing

Contents

Thin and Thick Lens Model

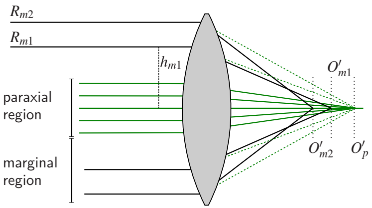


- (linear) Paraxial Optics - Gaussian Imaging based on

$$\frac{1}{f} = \frac{1}{b} + \frac{1}{g}$$

- Thick lens has two planes to account for
- Transformation can be represented by a matrix
- many lenes can be coupled by a product of matrices

What we Left Out



- Paraxial Optics fundamental law

$$\sin\theta \approx \theta$$

- Aberrations are introduced by higher order contribution:

$$\sin\theta = \sum_{n=0}^{\infty}(-1)^n \frac{\theta^{2n+1}}{(2n+1)!} = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} \pm \cdots$$

- Ray transfer matrices cannot handle inner reflections
- Wavelength was not considered at all
- Aperture size and shape needs further test

Contents

Previous Work

- Kolb et al introduced usage of a fully modelled lens to a distribution ray tracing setting
  - Use of analytic primitves suggested
  - Sequence of surfaces known in advance
  - Sampling ideas

Previous Work

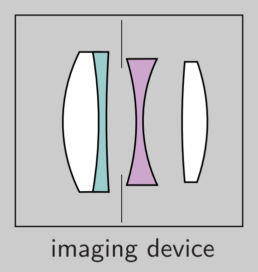
- Kolb et al introduced usage of a fully modelled lens to a distribution ray tracing setting
  - Use of analytic primitves suggested
  - Sequence of surfaces known in advance
  - Sampling ideas
- Material simulation of glass
  - Snell's Law
  - Fresnel Equations
  - Bougher-Lambert or Beers Law
  - Sellmaier Equation (dispersion)

Previous Work

- Kolb et al introduced usage of a fully modelled lens to a distribution ray tracing setting
  - Use of analytic primitves suggested
  - Sequence of surfaces known in advance
  - Sampling ideas
- Material simulation of glass
  - Snell's Law
  - Fresnel Equations
  - Bougher-Lambert or Beers Law
  - Sellmaier Equation (dispersion)
- Geometrical theory of diffraction introduced to graphics by Aveneau and Mériaux for scene account

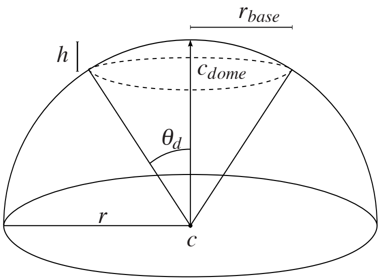
New Setting

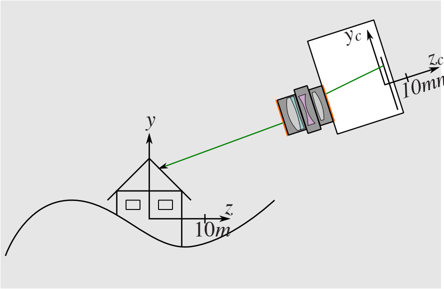
radius	thickness	material	diam.
42.970	9.8	LAK9	19.2
-115.33	2.1	LLF7	19.2
306.840	4.16	air	19.2
	4.0	air	15.0
-59.060	1.870	SF7	17.3
40.930	10.640	air	17.3
183.920	7.050	LAK9	16.5
-48.910	79.831	air	16.5



- Integration of real lens data in a Monte Carlo renderer
- Continuous spectral evaluation
- Full consideration of Snell's Law
- Inner reflections are demanded
- Explicit modelling of the aperture

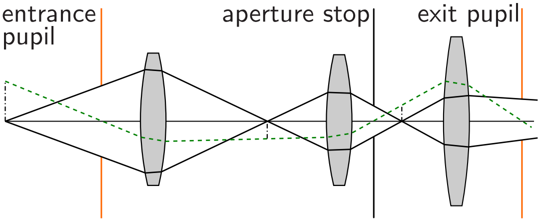
Representation and Handling





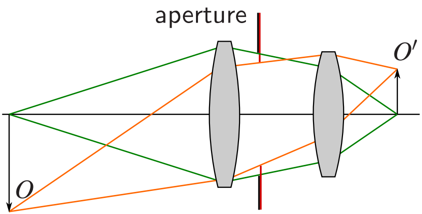
- Spherical caps and planes as primitives
- ray passage through the lens in an isolated program region
  - Allows change of measure, coordinate system or even precision
  - Optimisation on glass materials
  - Trigger surfaces connect scene and lens tracer

Entrance and Exit Pupil



- Image of the aperture on both sides forms pupils
- Entrance pupil defines acceptable irradiance
- Exit pupil identifies area of incoming irradiance contribution

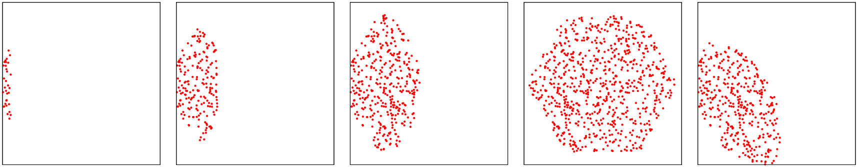
Optical Vignetting

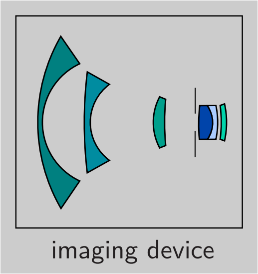


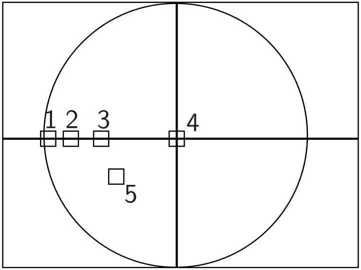
- Cone of illumination can get cropped by lens dimensions
- Effective aperture is reduced
- Main challenges:
  - Entrance pupil area varies with pixel position.
  - Path construction by simply choosing a location on the back lens can be awfully inefficient

Muller Fisheye example

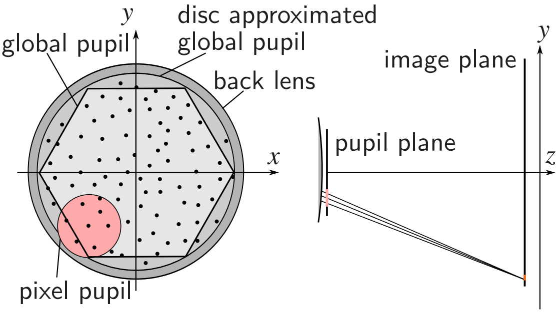
Image of the aperture varies significantly, seen from different pixel positions





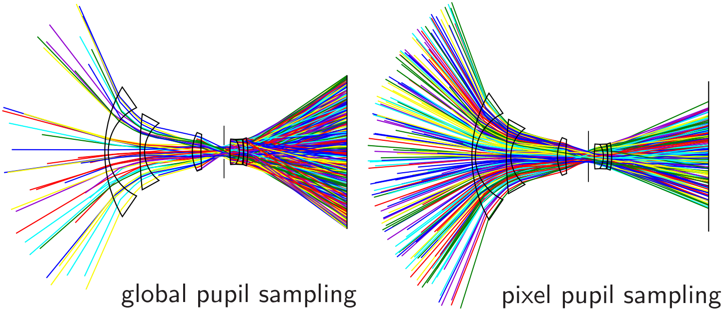


Per Pixel Pupil Computation



- For random points on the global exit pupil, evaluate paths from all pixels through the lens
- Record samples where the path reached the entrance side
- Estimate a per-pixel pupil from all positives

Path Sampling

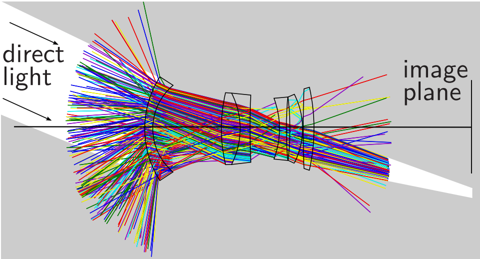


- Example shows an increase from 11% to almost 80% of ray passage probability.
- Pupil shape and precision limits number of zero contribution paths
- Sampling approximated pixel pupil reduces variance significantly

Optimisations

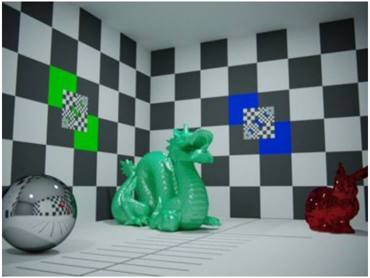
- Can be done in a precomputation step
- If lens is rotationally symmetric, only compute pupils for one quadrant and mirror results.
- Easy to parallelize
- Optimal pixel pupil makes aperture check obsolete and leads to 100% transmission
- Ideal for GPU computation

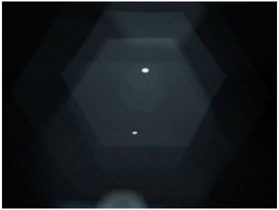
Progressive Rendering

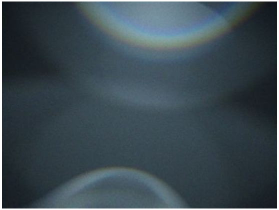


- Path pobability for an ES*L path sampled from the sensor side considering inner reflections is disastrous
- But deterministic connect on the lens from the light source is easy
- Separate image formation in stages:
  - Do pass backward without lens reflection using pixel pupil sampling
  - Use forward path tracing to collect direct light.

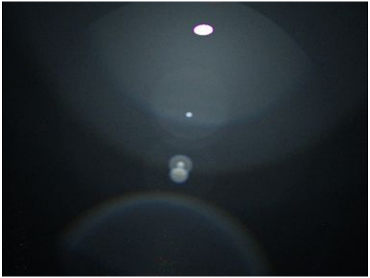
Lens Flare Results





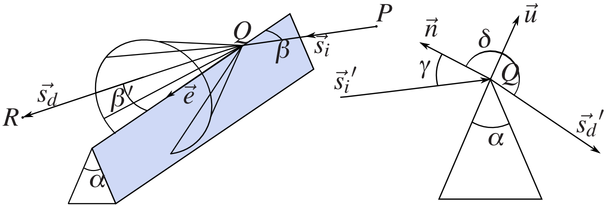


Coating Effects



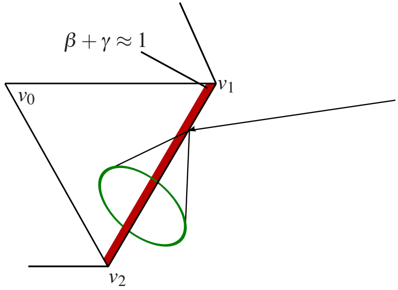
- Adjust Fresnel reflection results by reflection rate of a coating at normal incidence.
- Unfortunately no “real” data available

Geometrical Theory of Diffraction



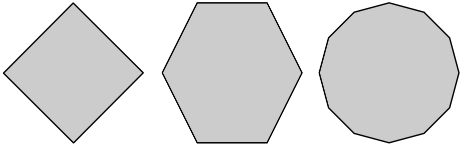
- First nice thing, contribution is summable without any change of the rest.
- Fermat's Law modified
- Diffracted rays at an edge lie on the sufrace of a cone, representing all valid diffraction directions
- Diffraction coefficent derived by Aveneau and Mériaux used. Is only valid for a point "on" the edge. But probability to hit an edge in 3D space is zero...

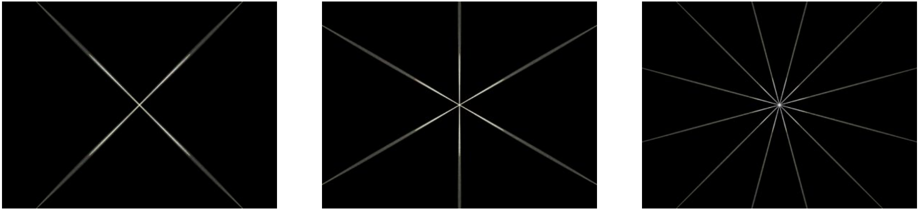
Keller Cone Sampling



- Check barycentric coordinates of hitpoint in the plane, whether we are close to the edge.
- When “hitting” the aperture edge, the diffraction direction needs to be sampled.
  - Choose sample on the base circle of the cone.
  - Take vector from Q through sample as diffraction direction.

Diffraction Results





- Diffraction forms streaks perpendicular to the diaphragm blade edges
- Same efficiency problem as lens flare simulation has
  $\longrightarrow$ Computation in a Light Tracing pass again

All together

Contents

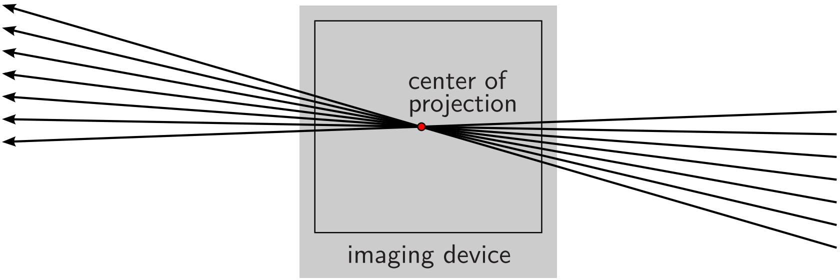
Conclusion

- Lens Simulation extended to MC Light Transport Simulation
- Various effects without extra cost
- Lens Flares by progressive approach
- Diffraction by material implementation
- Telescope or microscope simulation directly applies

Future Work

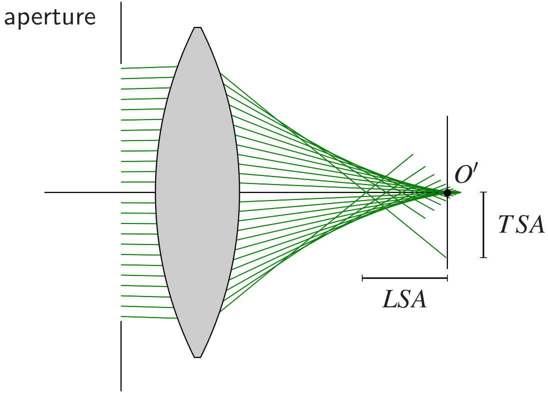
- Pupil Sampling for Light Tracing
- Diffraction validity
- Barrel inclusion
- Sensor simulation
- Interference and polarisation

Pinhole Camera

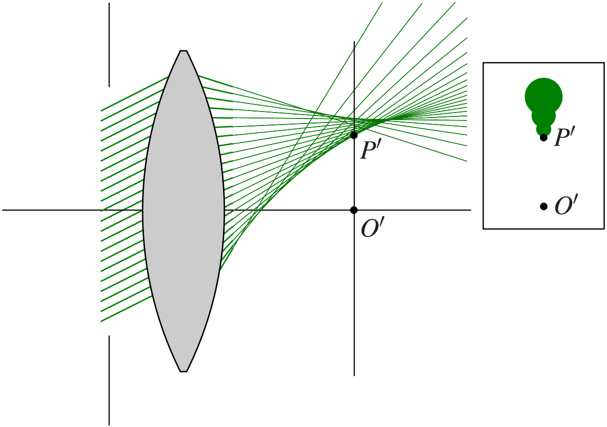


- all rays through center of projection
- no physical motivation
- aberrration free, all-focussed infinite depth of field

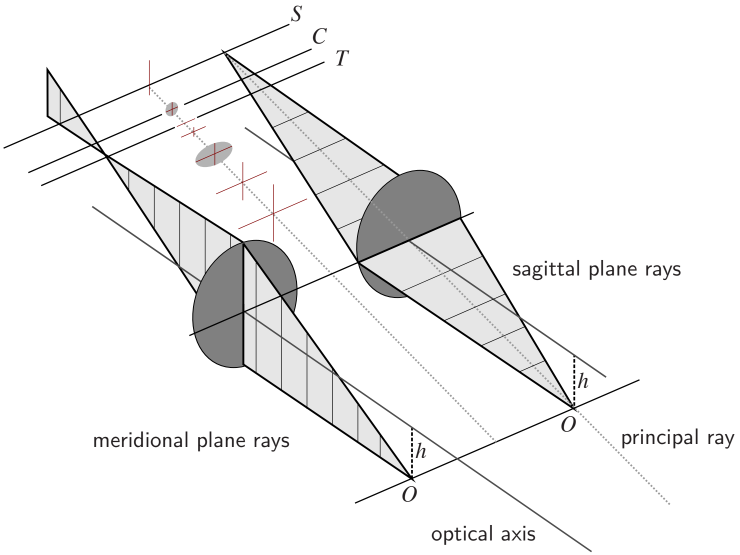
Spherical Aberration



Coma

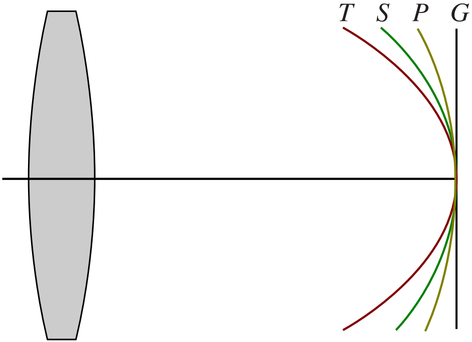


Astgmatism

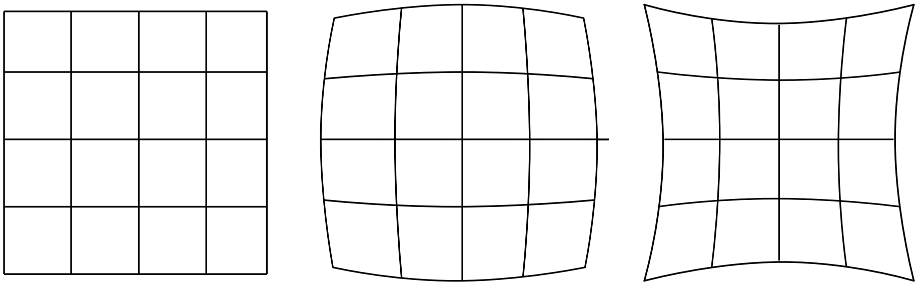


S
C
T
sagittal plane rays
h
O
principal ray
meridional plane rays
h
O
optical axis

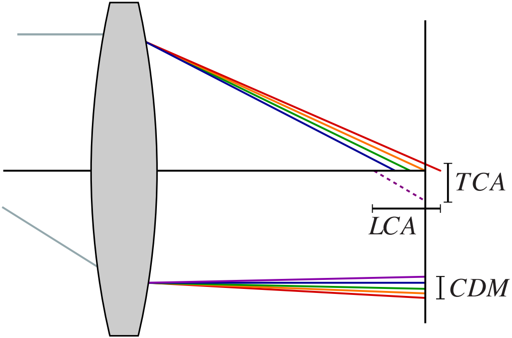
Field Curvature



Distortion

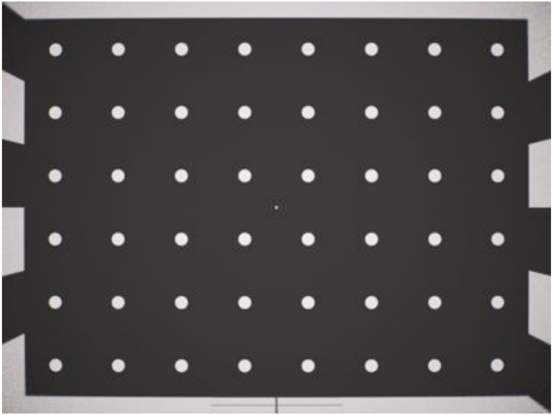


Chromatic Aberrations

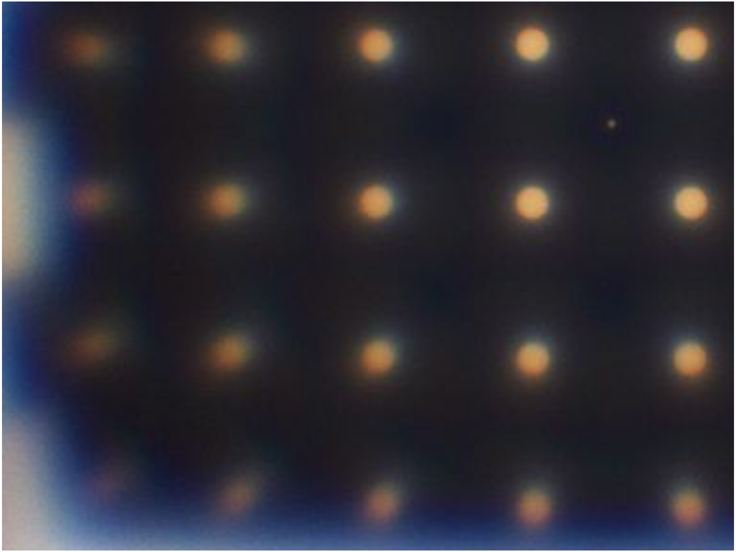


Aberrations

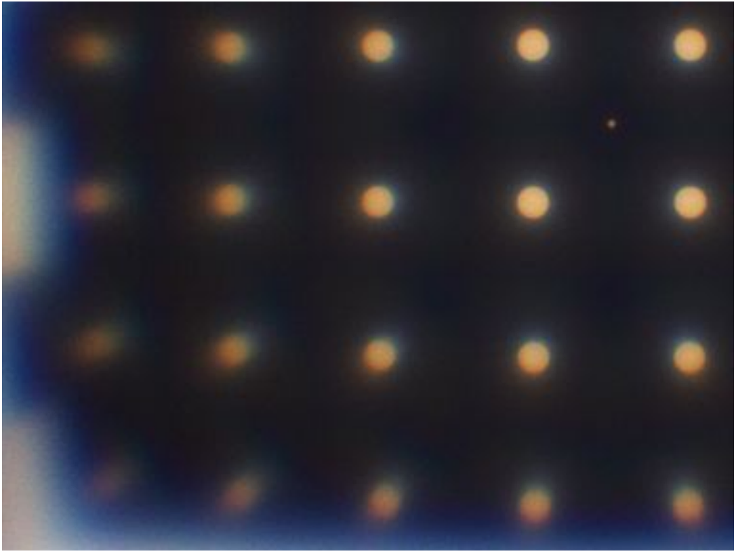
- Snell's Law evaluation for every surface, is the only thing we need...
- But todays lenses are highly corrected for all defects



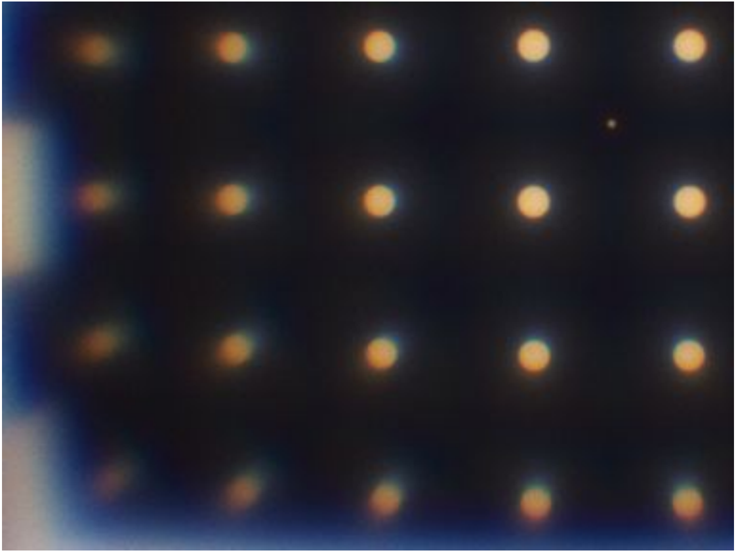
Single lens shot - image plane moving towards lens



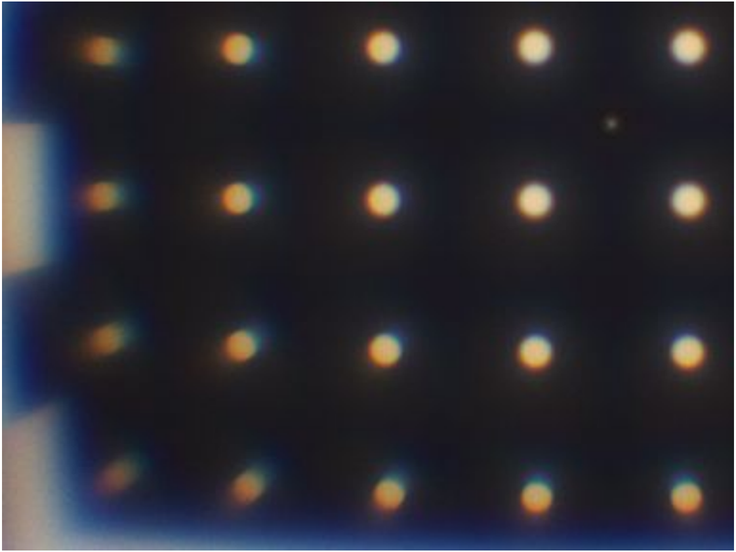
Single lens shot - image plane moving towards lens



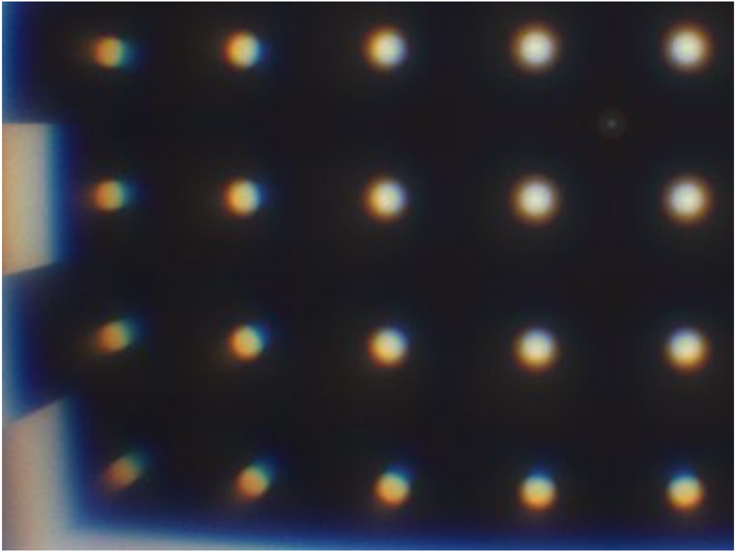
Single lens shot - image plane moving towards lens



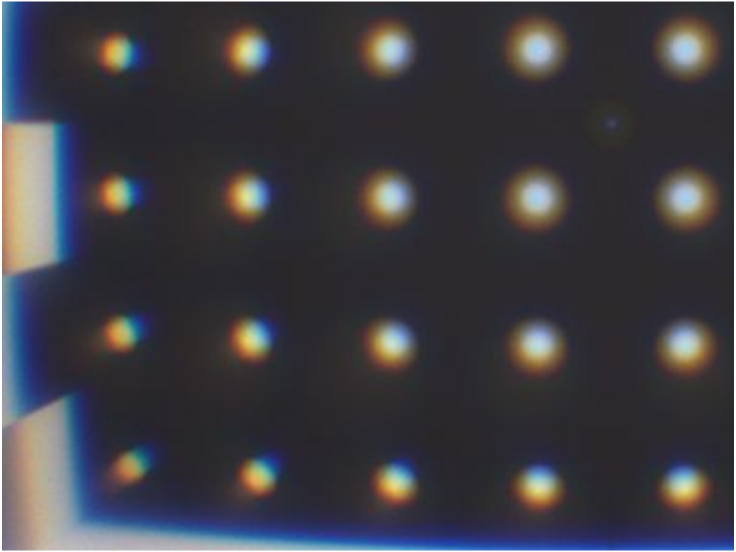
Single lens shot - image plane moving towards lens



Single lens shot - image plane moving towards lens

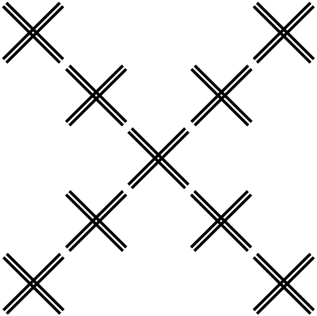


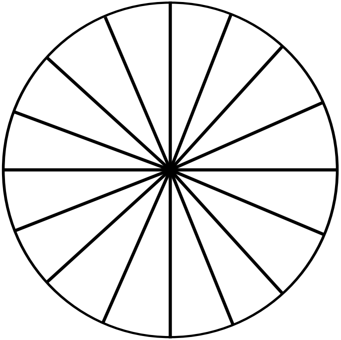
Single lens shot - image plane moving towards lens



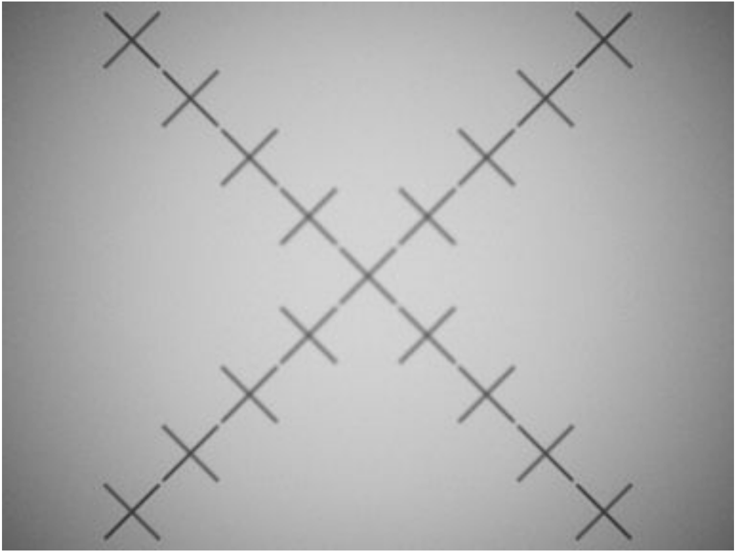
Not yet seen

Some aberrations hide each other and are only visible in certain situations.
Astgmatism is very tricky. . .

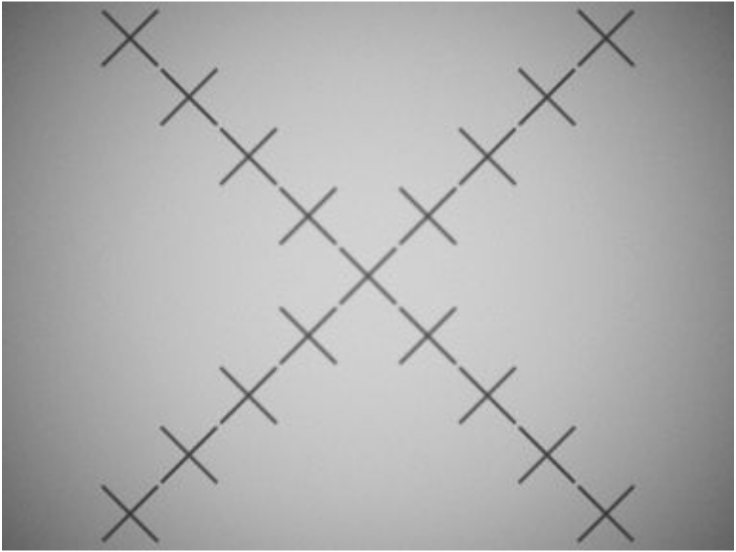




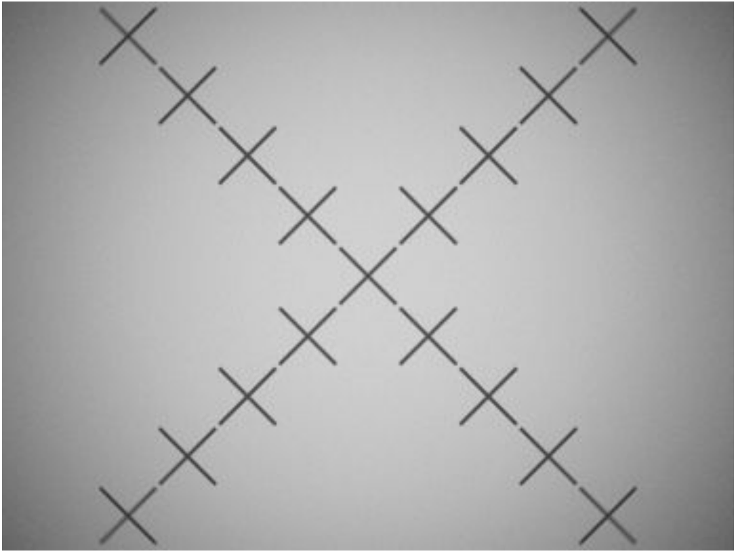
Petzval Portrait lens - image plane moving away from lens



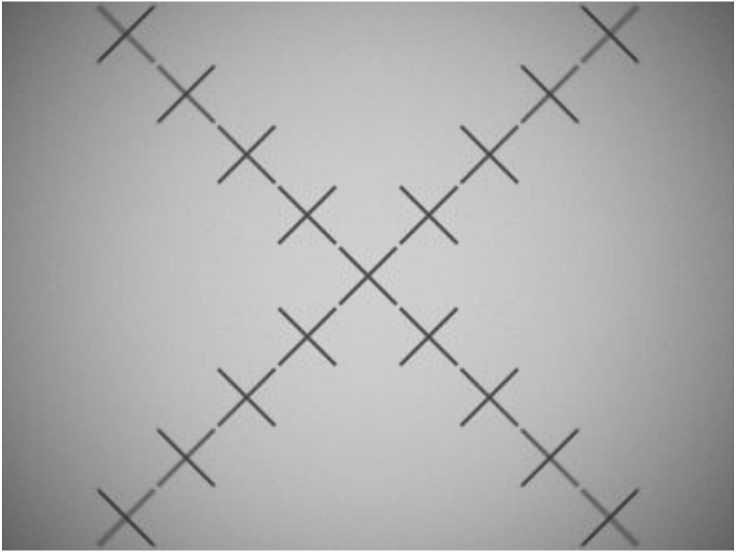
Petzval Portrait lens - image plane moving away from lens



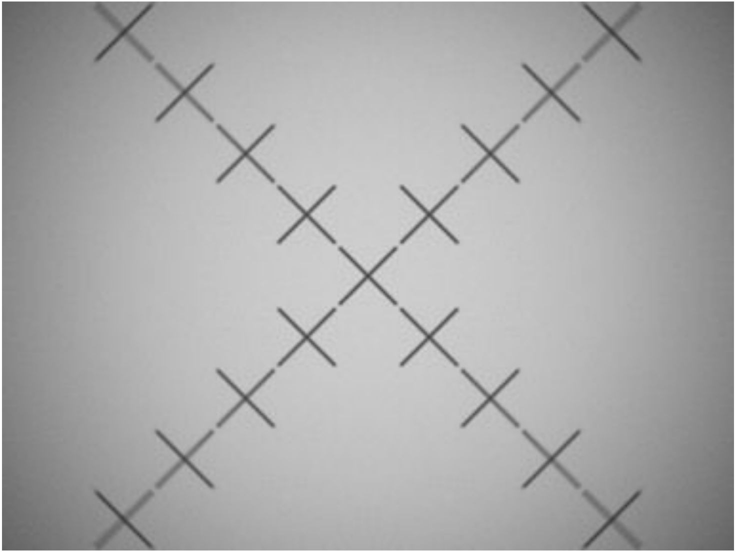
Petzval Portrait lens - image plane moving away from lens



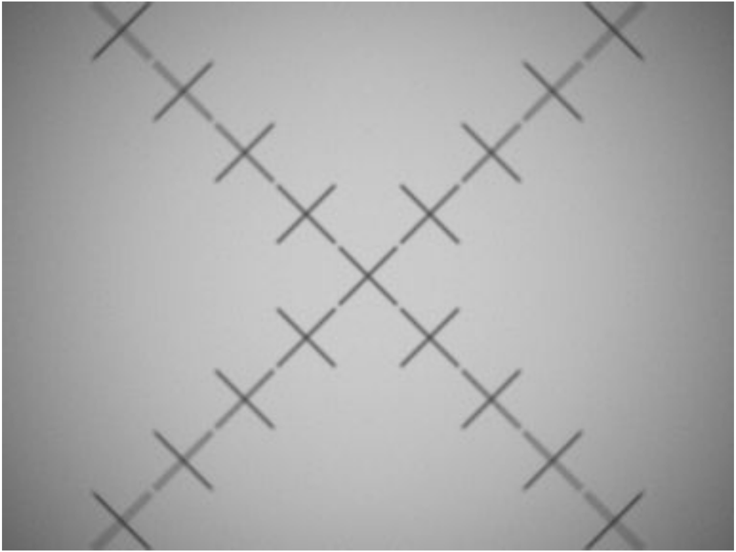
Petzval Portrait lens - image plane moving away from lens



Petzval Portrait lens - image plane moving away from lens



Petzval Portrait lens - image plane moving away from lens



Coating Data