Biologically Inspired Ray Tracing

Johannes Bohlig

Master's Thesis

Heidelberg, Germany, 24 January 2024

Introduction and motivation

- Ray tracing, the method of choice for realistic image generation
  - Increasing attention recently
  - Various innovations
- Realistic image rendering raises questions
  - How does the human eye see ?
  - Can this be visualized with ray tracing ?
  - What optical effects can be represented on this basis ?

[1] Rendering with ray tracing

[1] https://www.techspot.com/article/2485--pathpath--tracingtracing--vsvs--rayray--tracing/tracing/

Goals of the thesis

- Ray tracing model representing human vision

- Simulate visual effects based on human lens system
  - Visual effects due to properties of lens system
  - Effects arising through brain processing excluded

- Focus on lens system and optical effects
  - No focus on realistic rendering (shadows, lighting, etc.)

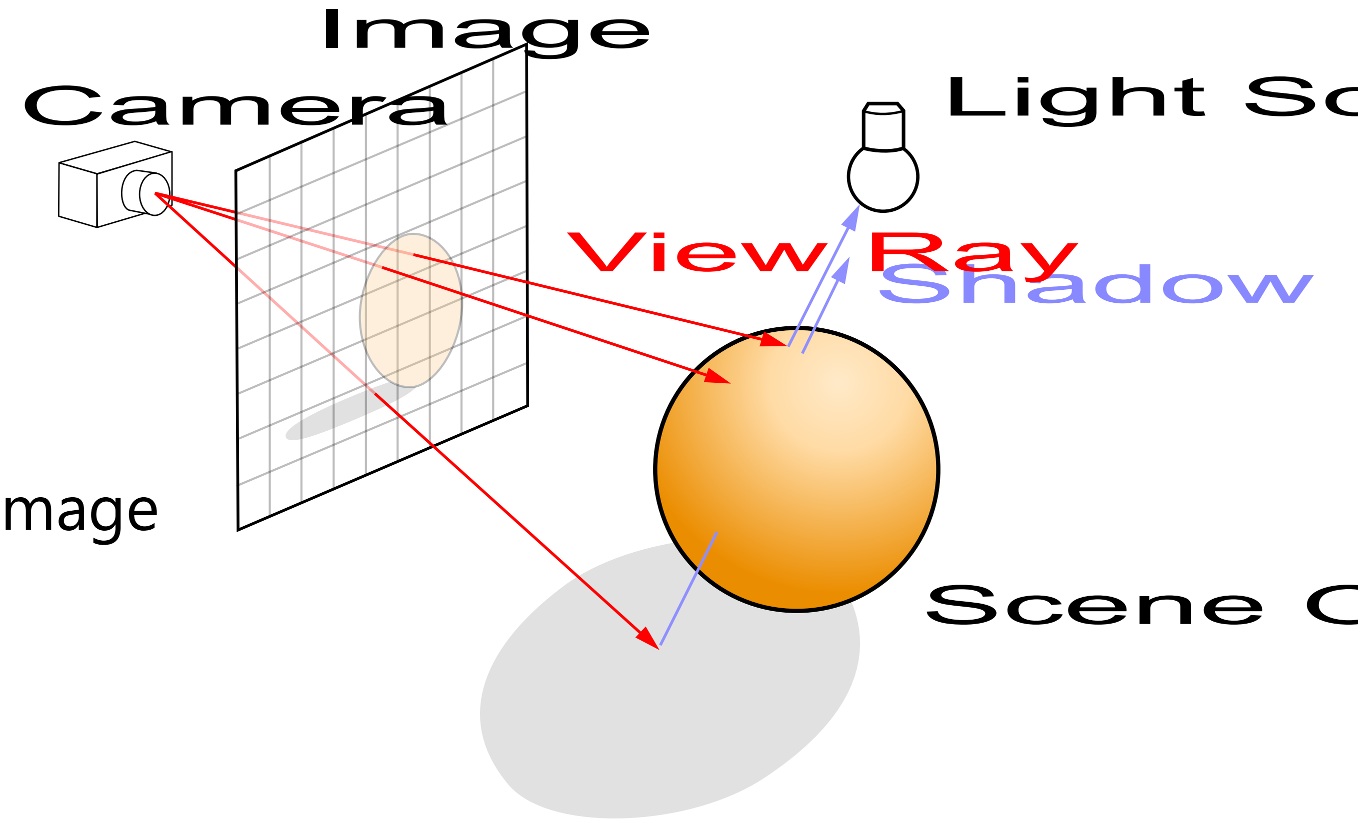
Fundamentals

Ray tracing

- Goal: Simulate light rays through the scene
  - Follow light paths to light source

- Shoot rays from camera through every pixel of image
  - Inverse Method
  - Color value of ray gives pixel color

- Check for ray-object intersections
  - If intersect:
    - Shoot secondary ray (e.g. Shadow Ray,etc.)
    - End raytracing for pixel



[1] Raytracing Scheme

Eye model

- Human eye is a complex organ
  - Demand of simplification

- Light passing several layers, causing refraction multiple times
  - E.g., cornea, lens, vitreous body, eye chamber
  - Layers refract differently $\Rightarrow$ Combined refractive power of 61.7 dpt

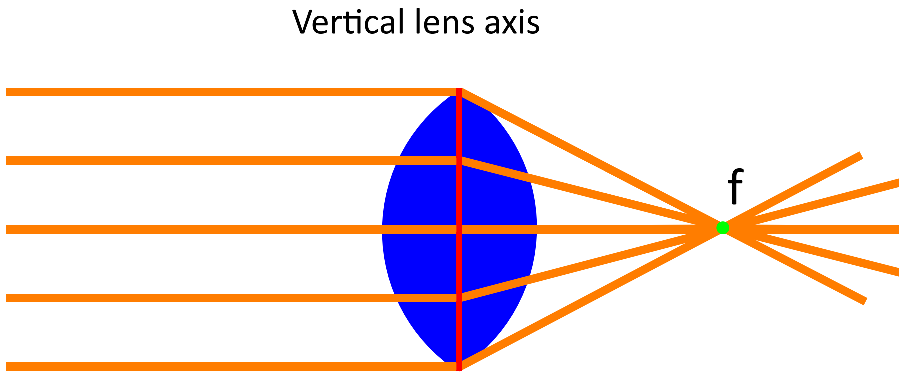
[1]

- Lens mainly responsible for refraction
  - Can adjust thickness and shape $\Rightarrow$ adjust refraction
  - Performs accommodation
    - Focus object on retina $\Rightarrow$ adapt different object distances

[1] https://www.researchgate.net/figure/The-accommodation-dependent-schematic-eye-model-as-used-for-all-simulations-The-numbered_fig1_26242870

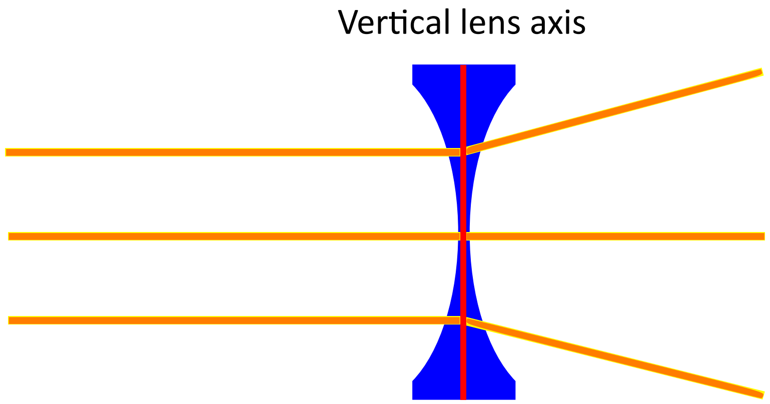
Lens

- Convex lenses have outwardly curved surface(s)
  - Parallel light converges behind the lens
  - Special case: rays meet behind the lens in one point $\Rightarrow$ focal point



Convex Lens, parallel rays meet in focal point (f)

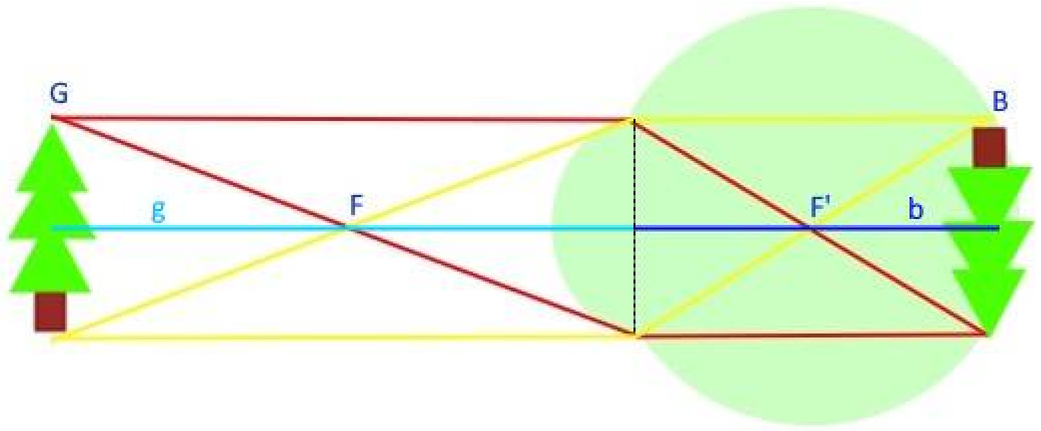
- Concave lenses have inwardly curved surfaces(s)
  - Parallel light diverges behind the lens
  - Never forms focal point



Concave lens, parallel rays diverge

Human eye lens system

- Human eye lens corresponds to rotational ellipsoid
  - Forms a biconvex, non symmetrical, converging lens
- Lens system consists of:
  - Focal points F and F'
  - Object plane G, where object is located
  - Image plane B, where image of G is focused
  - Object width g = distance from G to center of lens
  - Image width b = distance from B to center of lens
- All properties of the lens system relate by



Lens equation in eye

Method

Method

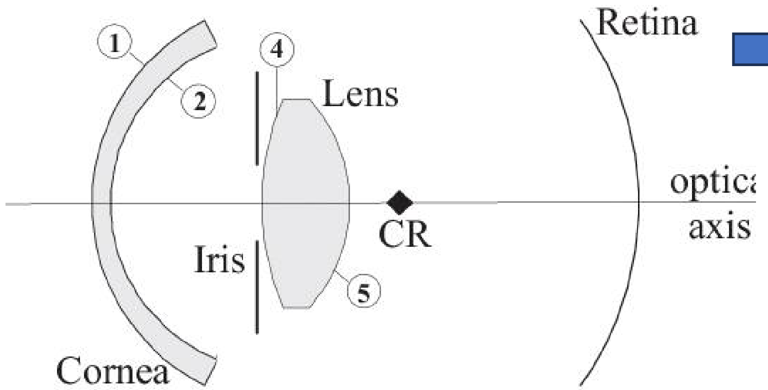
- Goal:
  - Simulate visual effects based on developed model

- Explorative research approach
  - Many possible effects to realize

- Method and validation
  - Model and effects based on literature
  - Validation by comparison with real world or theoretical validations

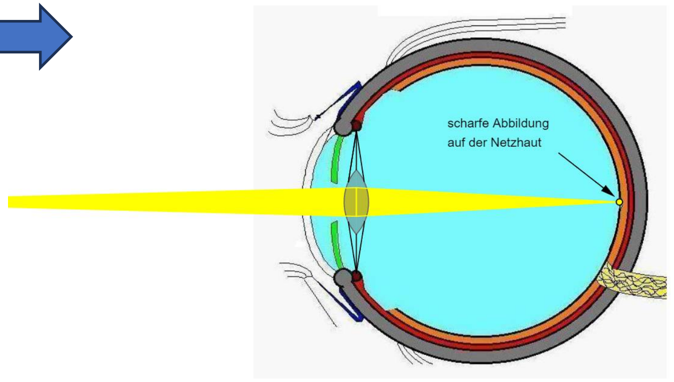
Iterative development of the eye model

Simplified model of human eye



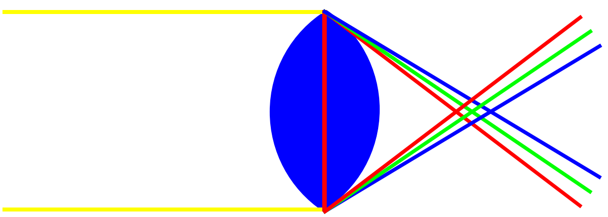
[1]

Path of light through the eye



[2]

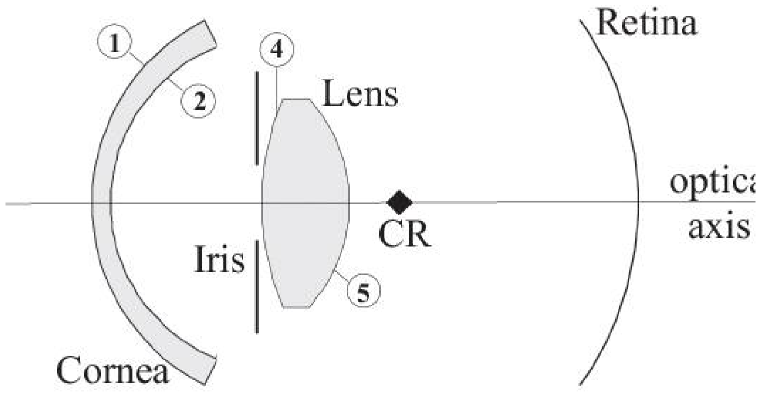
Goal: Simulate visual effect



[1] https://www.researchgate.net/figure/The-accommodation-dependent-schematic-eye-model-as-used-for-all-simulations-The-numbered_fig1_26242870
[2] https://www.geogebra.org/m/kzAM7w93

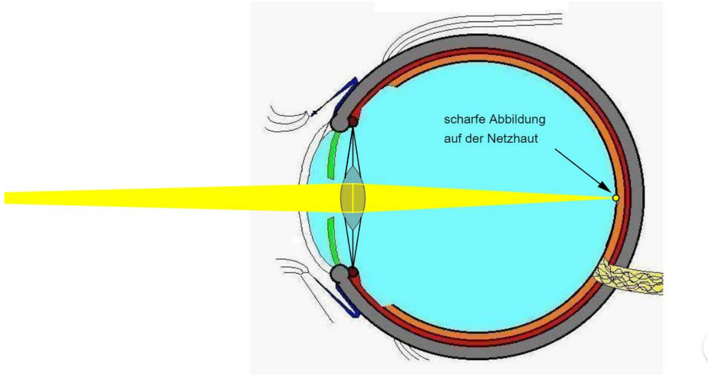
Iterative development of the eye model

Simplified model of human eye



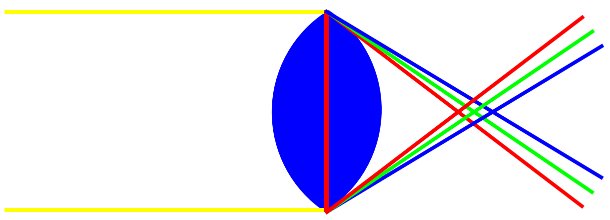
[1]

Path of light through the eye



[2]

Goal: Simulate visual effect

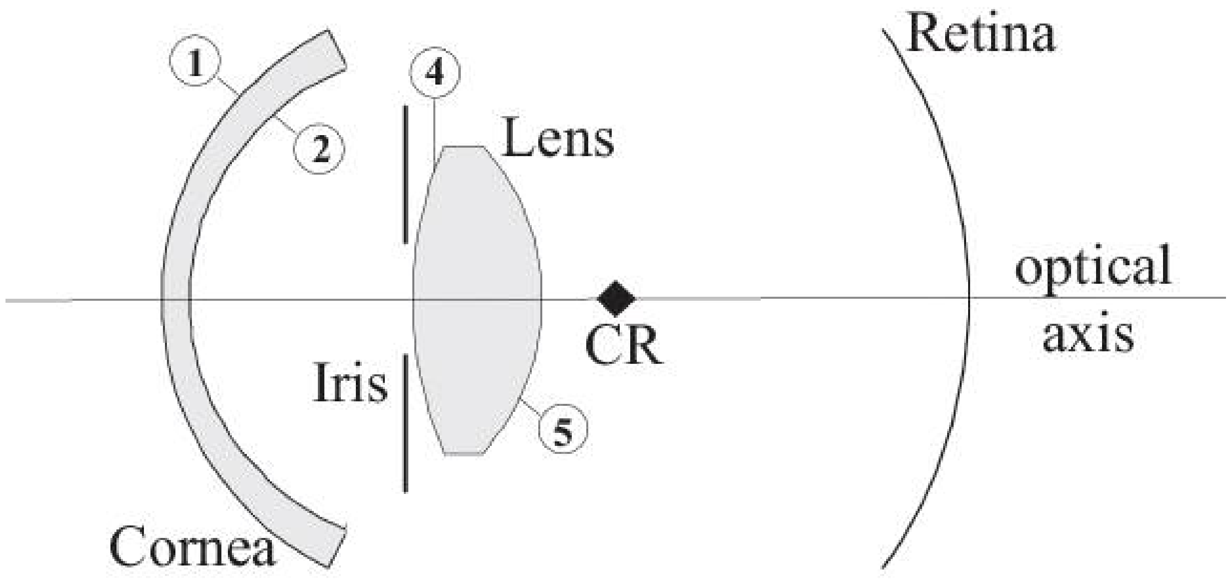


VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

[1] https://www.researchgate.net/figure/The-accommodation-dependent-schematic-eye-model-as-used-for-all-simulations-The-numbered_fig1_26242870
[2] https://www.geogebra.org/m/kzAM7w93

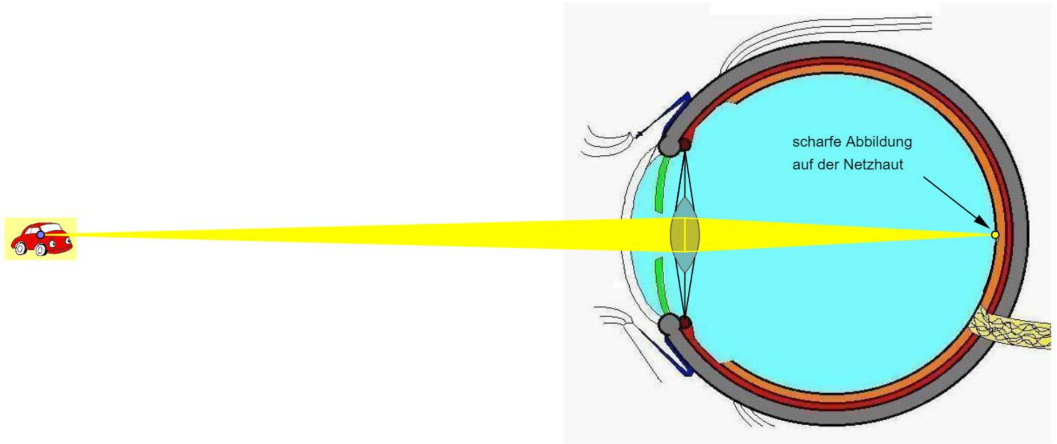
Method realization

- Schematic eye model used as template
- Parts to be modeled
  - Lens
  - Retina
- Functions to realize
  - Light traversal
  - Accommodation



[1]

Reduced model of eye



Cone of rays focus with focused object [2]

[1] https://www.researchgate.net/figure/The-accommodation-dependent-schematic-eye-model-as-used-for-all-simulations-The-numbered_fig1_26242870
[2] https://www.geogebra.org/m/kzAM7w93

Lens model

- Lens model criteria
  - Forming focal point and image plane
  - Adjusting refractive power by thickness
  - Feasible calculation of intersection point with ray

  \+ Low computational costs calculating intersection point

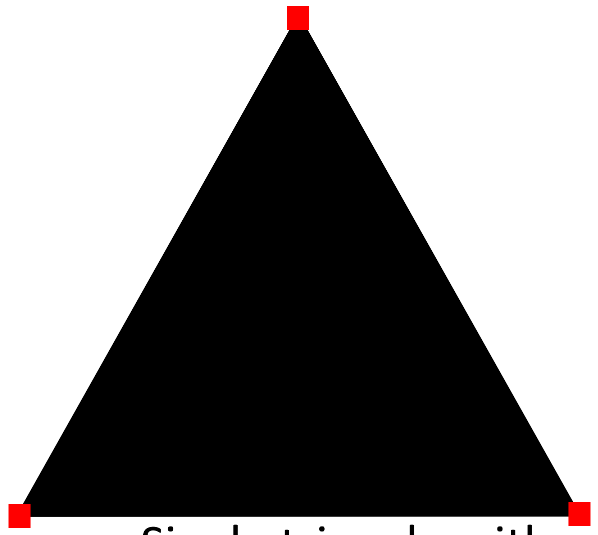
- Multiple ways to describe objects and surfaces mathematically
  - Due to implementational complexity and availability in the software

  -> triangle mesh, implicit
    - Comparison needed

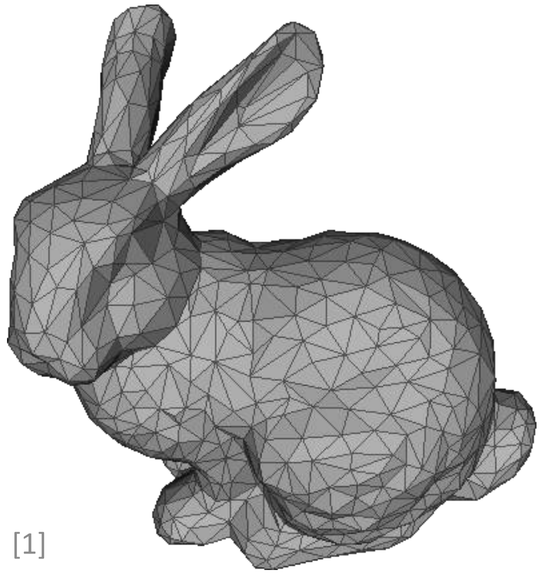
VISUAL COMPUTING GROUP
HEIDELBERG UNIVERSITY

Triangle Mesh Lens

- Triangle Mesh consists of connected triangled areas
  - Vertices and their connectivity triangled areas
- Eye lens consists of smooth curved surfaces
  - Hard to represent smoothness with triangles
- Refraction needs only smooth surface normal(Snell's law)
  - -> achievable with smooth shading



Single triangle with vertices (red)



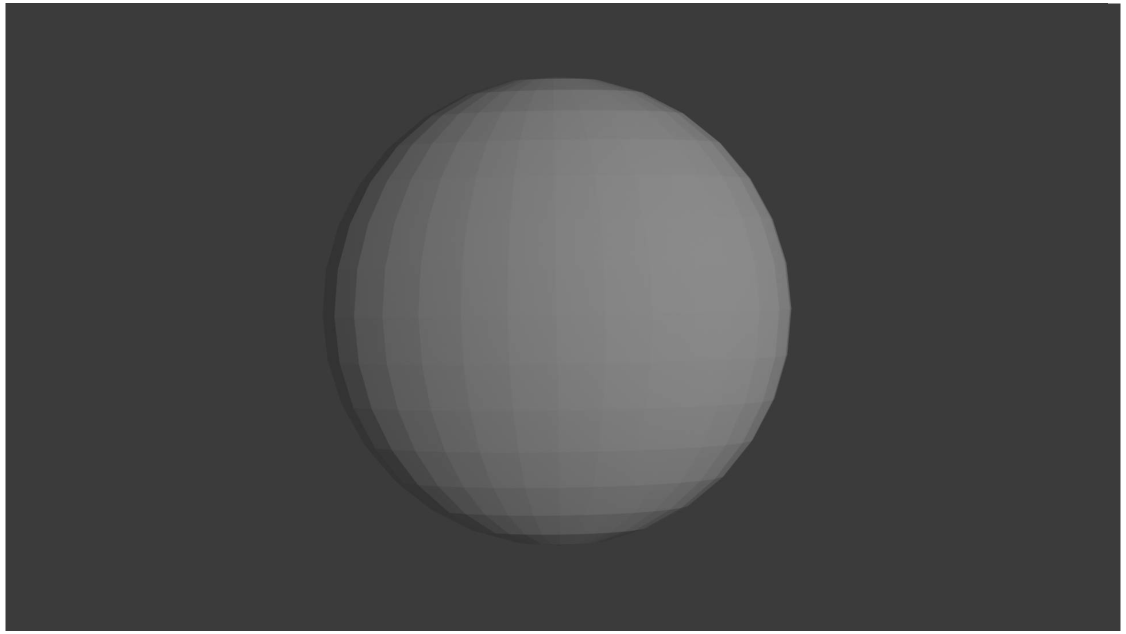
[1]

Triangle Mesh consisting of many triangles

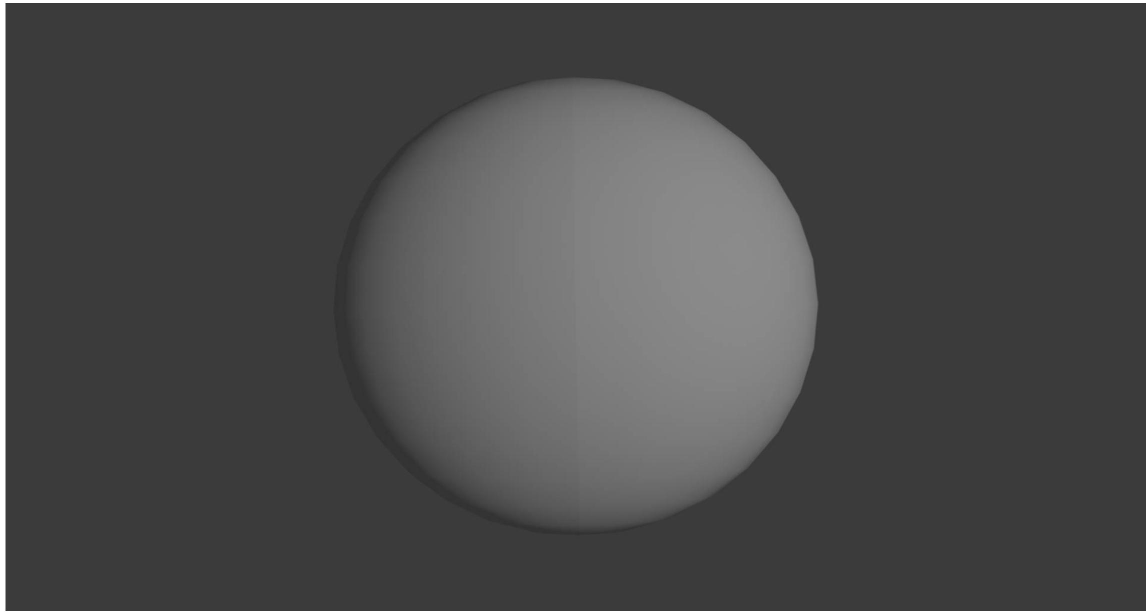
VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

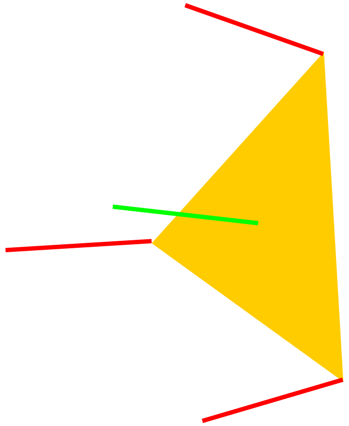
[1] https://www.lix.polytechnique.fr/~maks/Verona_MPAM/TD/TD2/

Triangle mesh lens



- Smooth surface normals as with smooth shading
  - Interpolating surface normal
  - Barycentric interpolation inside triangle
    - Interpolate normal at point between vertex normal
  - Problem: Introduces error in accuracy

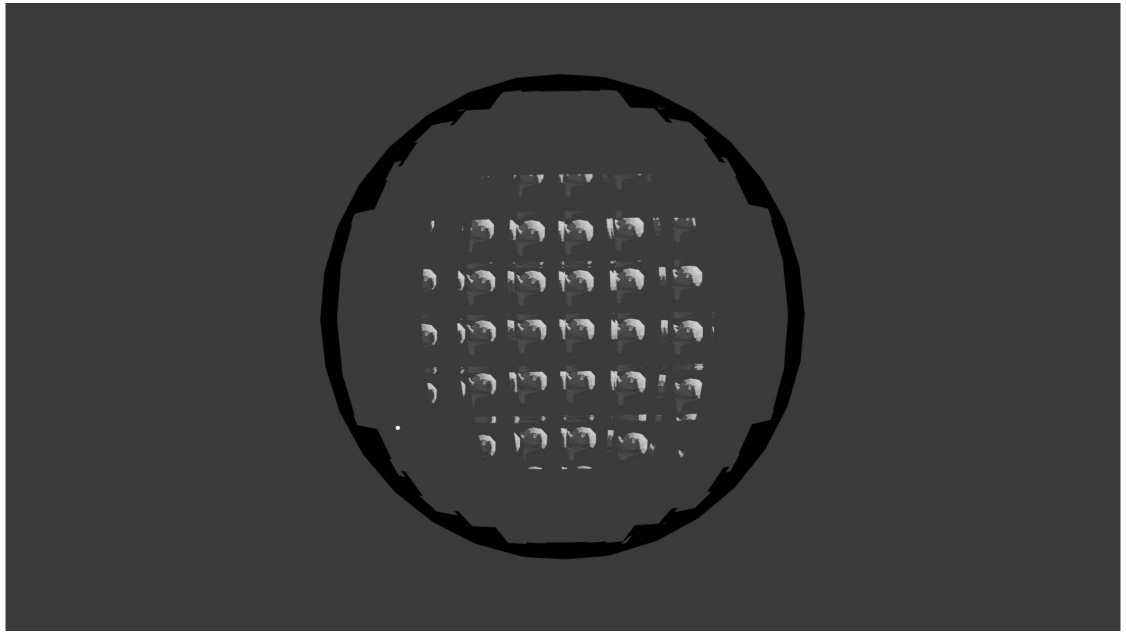




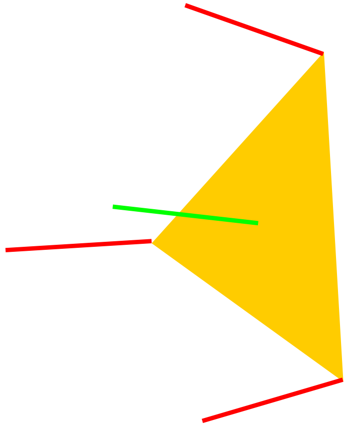
Barycentric normal interpolation

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Triangle mesh lens



- Smooth surface normals as with smooth shading
  - Interpolating surface normal
  - Barycentric interpolation inside triangle
    - Interpolate normal at point between vertex normal
  - Problem: Introduces error in accuracy

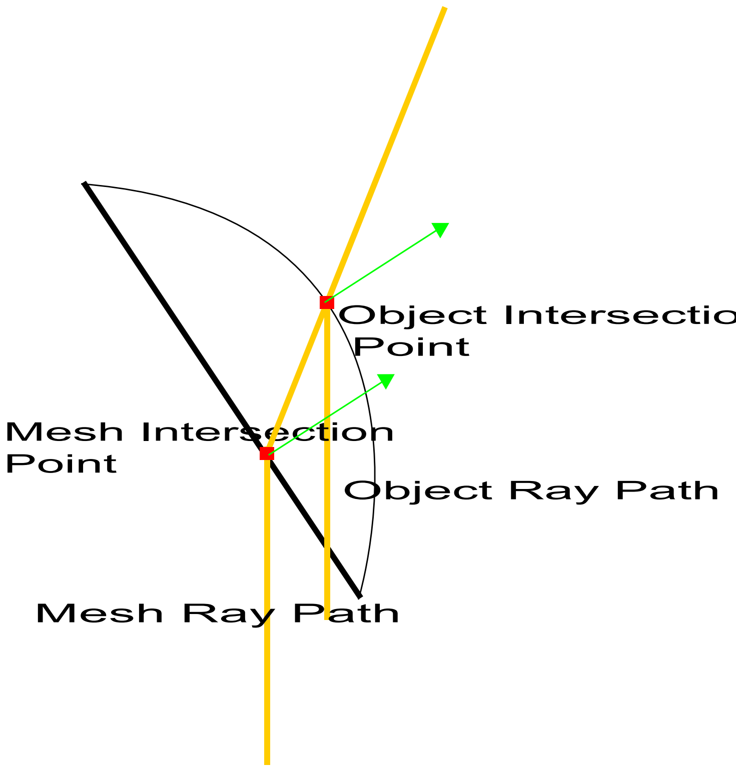


Barycentric normal interpolation

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Triangle mesh problems

- Changes in size of mesh can lead to changes in shape
  - Loss of focal point
- Difference between interpolated and object intersection point
  - Causes ray path deviating from object path
- Display error at edges due to interpolation problem
- High computational cost
  - Ray triangle intersection expensive
  - Many triangles required for high precision

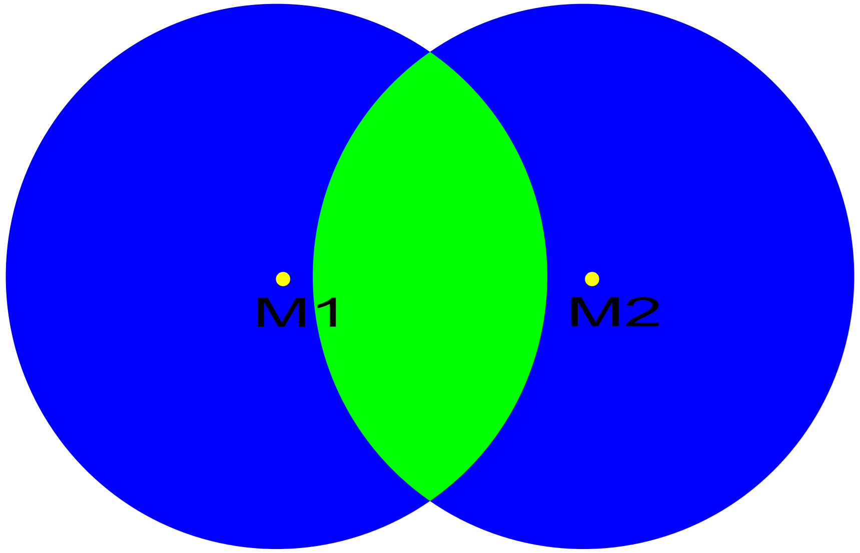


Difference between object and mesh path

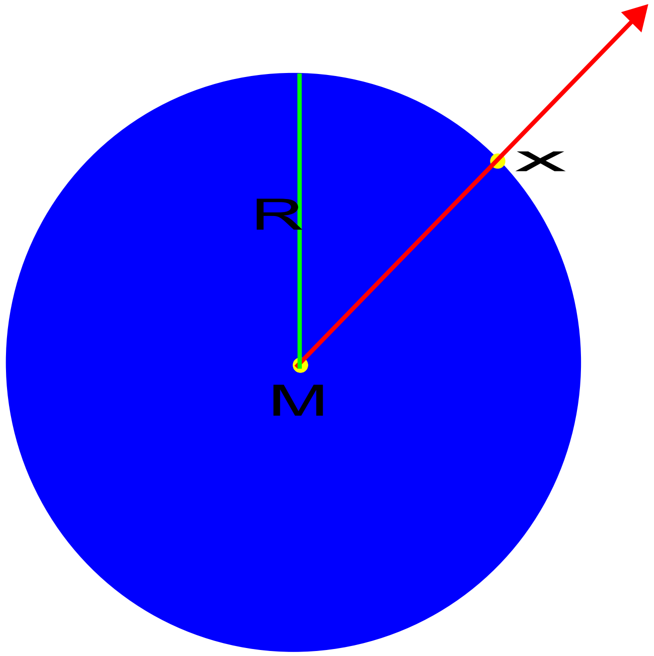
VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Sphere surface lens

- Lens can be described with implicit spherical surfaces
  - Overlap two spheres
  - Spheres defined by centers , and radii ,
  - Lens formed if | − | < +
  - Calculate normal with sphere center and surface intersection
- Normal calculation: sphere center minus intersection point
  - Sphere origin chosen by position of relative to plane
    - If on the left, is chosen and vice versa



Lens formed by intersecting spheres



Single Sphere, defined by center (M) and radius(R). Normal at intersection point (x)

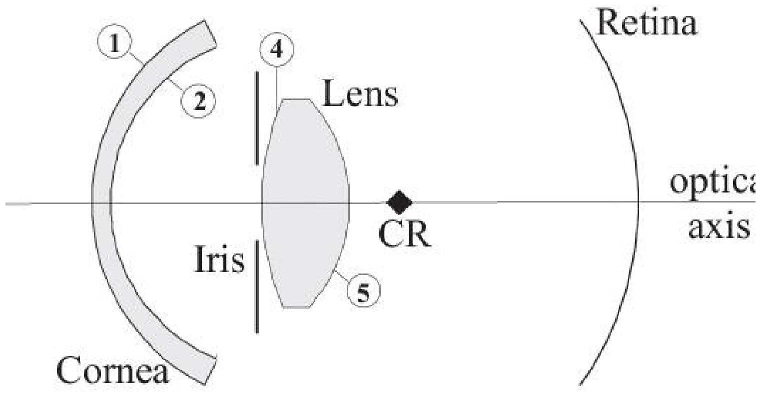
Choise of lens construction

Triangle Mesh Lens	Sphere Surface Lens
+ Precise modelling + Adaptation to different shapes	+ Computational costs + Precision
- Precision problems - Computational costs	- Increased design effort
⇒ Used for initial design	⇒ Used in final model

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

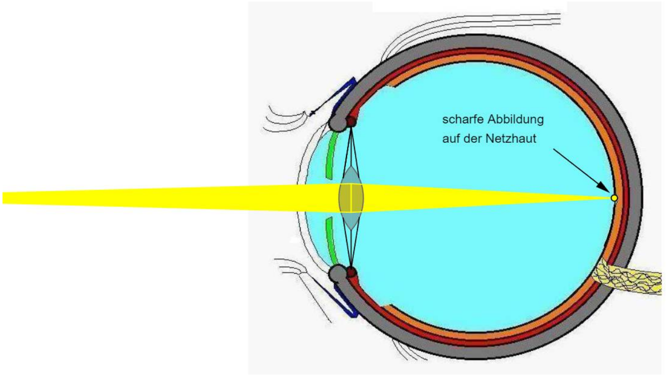
Iterative development of the eye model

Simplified model of human eye



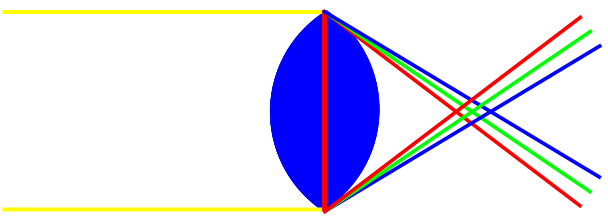
[1]

Path of light through the eye



[2]

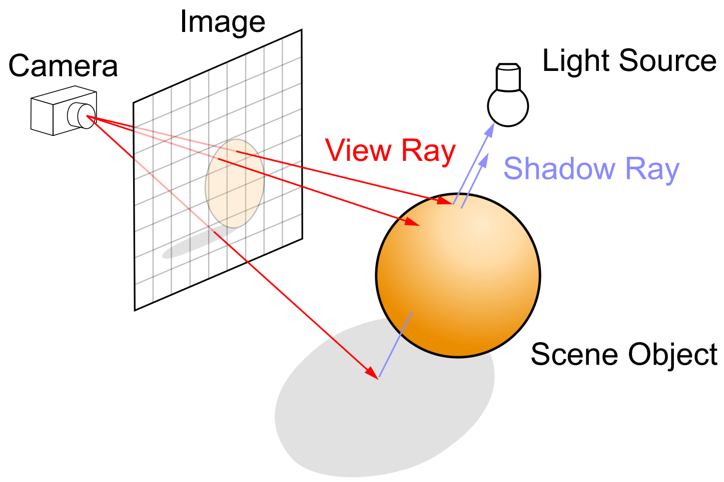
Goal: Simulate visual effect

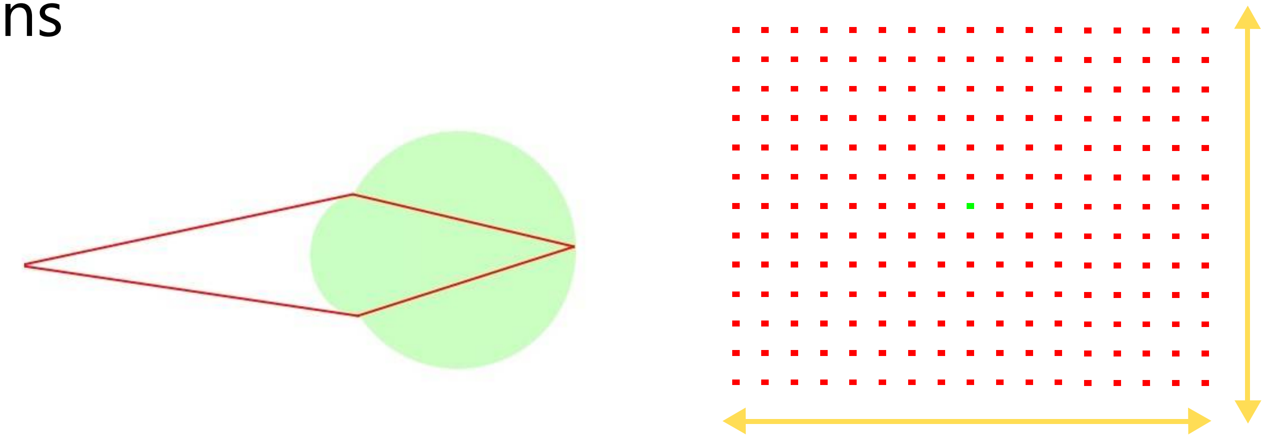


[1] https://www.researchgate.net/figure/The-accommodation-dependent-schematic-eye-model-as-used-for-all-simulations-The-numbered_fig1_26242870
[2] https://www.geogebra.org/m/kzAM7w93

Ray tracing methods

Ray tracing	Biologically Inspired Ray Tracing
One view ray per pixel	Multiple view rays per pixel
One ray origin per image (camera)	One ray origin per pixel
Ray directions determined by pixel raster (image)	Sampling to determine ray direction through lens



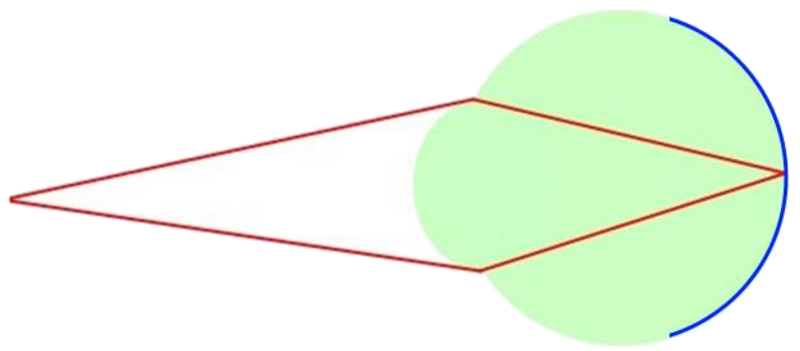


VISUAL COMPUTING GROUP
HEIDELBERG UNIVERSITY

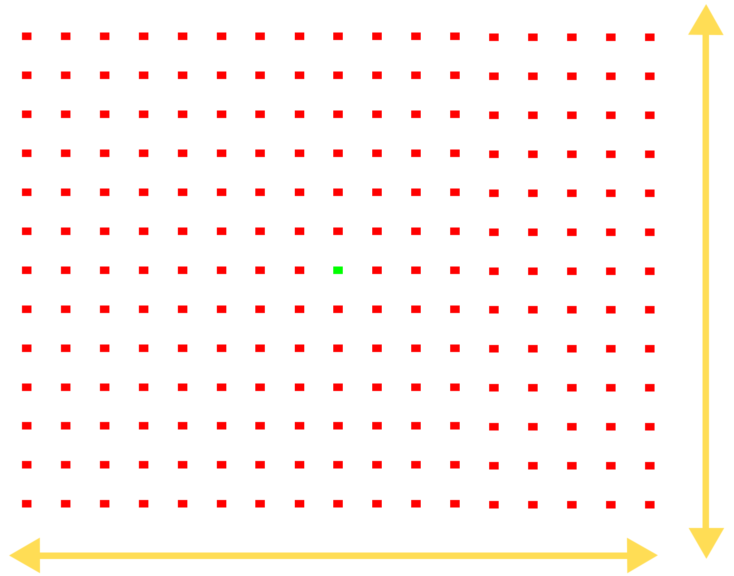
Simulation of the retina

- Retina part of the eye receiving light
  - Screen serves as retina
  - Located at image plane

- To simulate retina incident light must be simulated
  - Light rays originate on retina, travel through scene to object
  - Pixels serve as light receptors
    - Every Pixel emits multiple light rays
  - Follow conical lightpath from object to retina

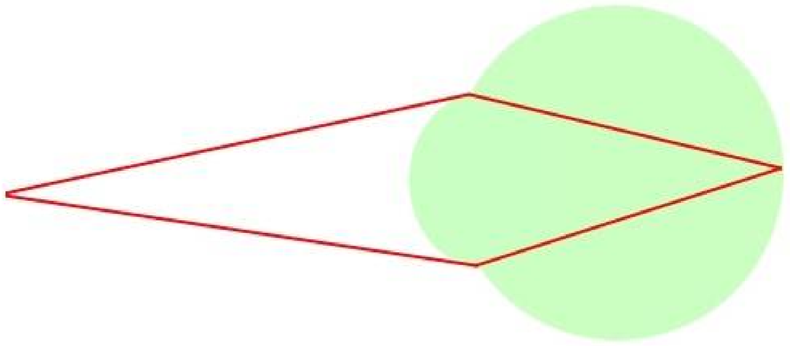


Eye ball with retina located at



Raster of pixels, every serving as light receptor

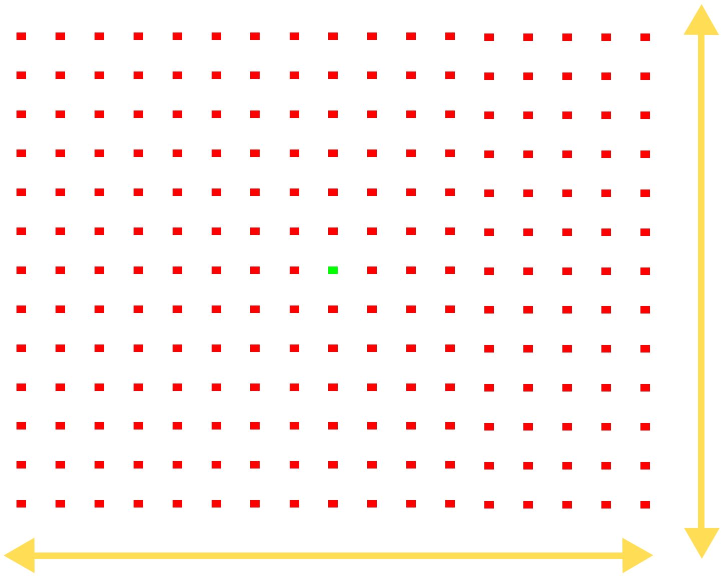
Simulation of the retina



Cone of possible rays

- Light ray needs origin and direction values

- To use Pixels as ray origin, positions must be determined
  - 2D screen coordinates projected into 3D space
    - Camera position as reference in middle of screen
    - For every pixel, left to right and bottom to top position



2D screen coordinates into 3D coordinates.
Green pixel in middle camera position

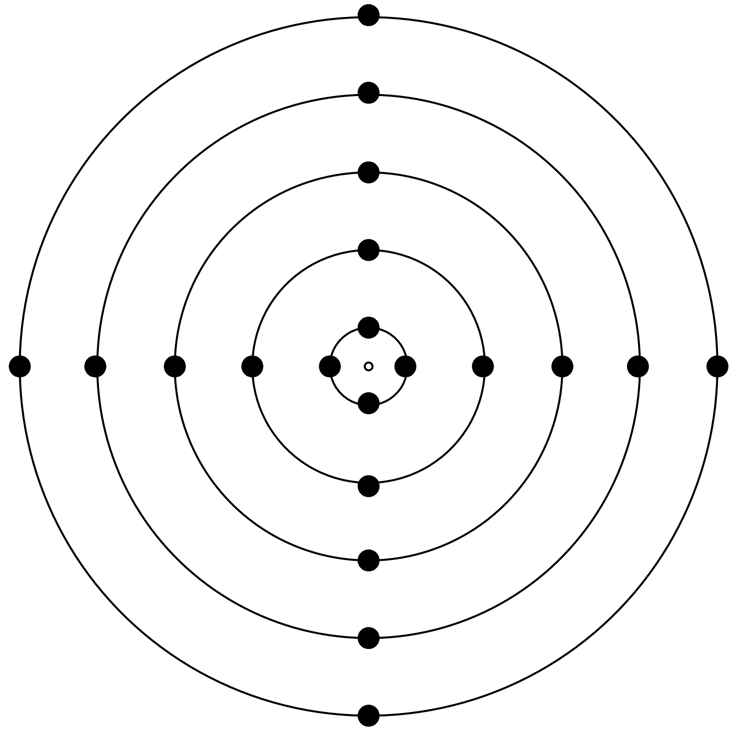
- Light rays used to determine pixel color value
  - Ray values integrated in color value
  - More rays $\Rightarrow$ more accurate result

$$col = \frac{1}{n} \sum_{0}^{n-1} f(r, g, b)$$

Lens sampling

- To determine ray direction reference points are needed
  - Sampling points are used approximating lens shape

- Approximating lens shape with concentric circles
  - Sampling points evenly distributed across circles
    - Even distribution leads to better accuracy
  + Decoupling sampling points from lens



Concentric circles with sample points

Light regulation

- Eye is a sensitive organ, possible damage by high light intensity

- Protection provided by the iris
  - Controls light intensity amount
    - Minimum required te see
    - Maximum to prevent damage
  - Iris adapts to environmental light intensity

[1] https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT3ulia2xxUmvDmIlJ85h8lYpaEufyPwiat1g&usqp=CAU

Simulation of the iris

- Ring shaped body in front of lens
  - Open and closed by muscles
    - Closed if light intensity reaches maximum
    - Opened if light intensity reaches minimum

- Simulation with 3D ⇒ disc shaped object
  - Placed in front of virtual lens
  - Defined by origin and radius
    - Origin located in middle of lens
    - Size adjusted by radius
  - Rays hitting Iris is entering scene
    - Others directly resulting as black

[1]

Real iris

[1] https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT3ulia2xxUmvDmIlJ85h8lYpaEufyPwiat1g&usqp=CAU

Light intensity regulation

- Determine light intensity to decide if iris size should adapt
  - Light level (brightness) on retina, i.e. screen, needs to be calculated
  - Brightness $\Rightarrow$ average of all pixel values

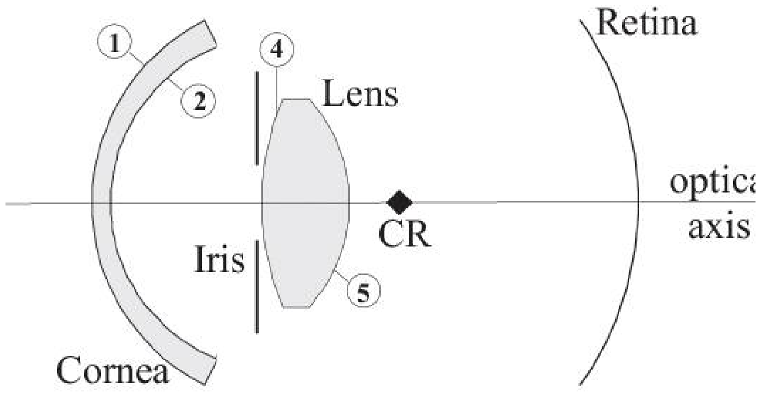
$$brightness = \frac{1}{n} \sum_{0}^{n-1} \frac{p_r + p_g + p_b}{3}$$

- Regulation on brightness value

- Adjustment to target bightness performed iteratively
  - First iteration, look for derivation
  - If needed adapt size in small steps
  - Measure derivation, if needed perform adaption again

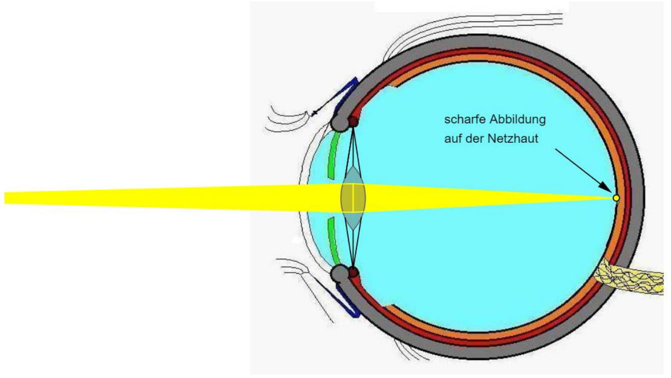
Iterative development of the eye model

Simplified model of human eye



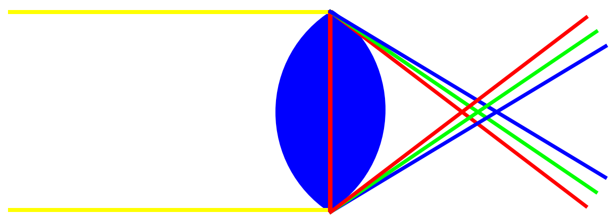
[1]

Path of light through the eye



[2]

Goal: Simulate visual effect



VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

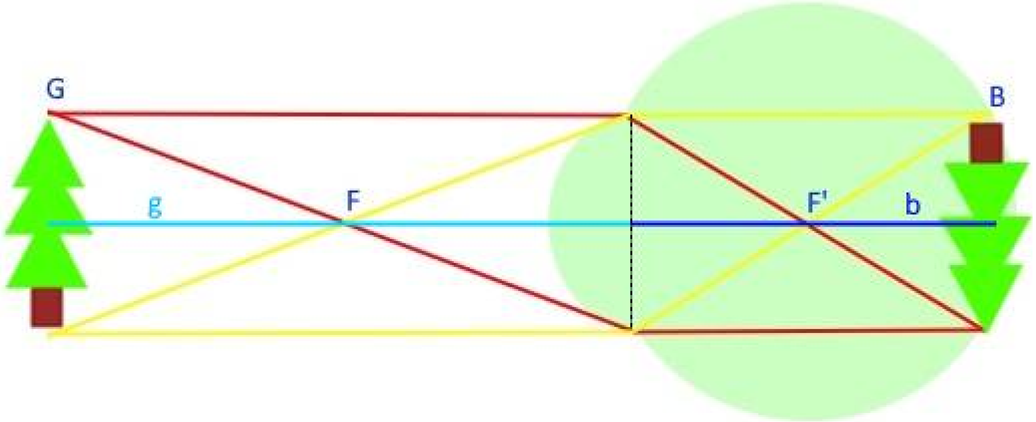
[1] https://www.researchgate.net/figure/The-accommodation-dependent-schematic-eye-model-as-used-for-all-simulations-The-numbered_fig1_26242870
[2] https://www.geogebra.org/m/kzAM7w93

Accommodation

- Accommodation important function for correct refraction
  - Adaption to given image plane

Lens equation

- Adaptation of lens
  - Size necessary due to fixed eye size
- Send ray to measure distance between lens and object
  - Use measured value as of lens equation
- Parameter lens equation fixed given value



Lens equation in eye

Model the eye function

- Thickness of lens has to be calculated
  - Lens makers equation is used

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

Lens equation

- Set lens equation equal to lens makers equation

$$\frac{1}{f} = \frac{n - n_0}{n_0}\left(\frac{1}{R_1} - \frac{1}{R_2} + \frac{(n - n_0)\,d}{n\,R_1\,R_2}\right)$$

Lens makers equation

- Rearrange to

- Reshape lens with given

$$d = \frac{n\left(bg\,(n-1)\,(R_1 - R_2) + b\,R_1\,R_2 + g\,R_1\,R_2\right)}{bg\,(n-1)^2}$$

Lens and lens makers equation rearranged to thickness (d)

[1]

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Chromatic aberration

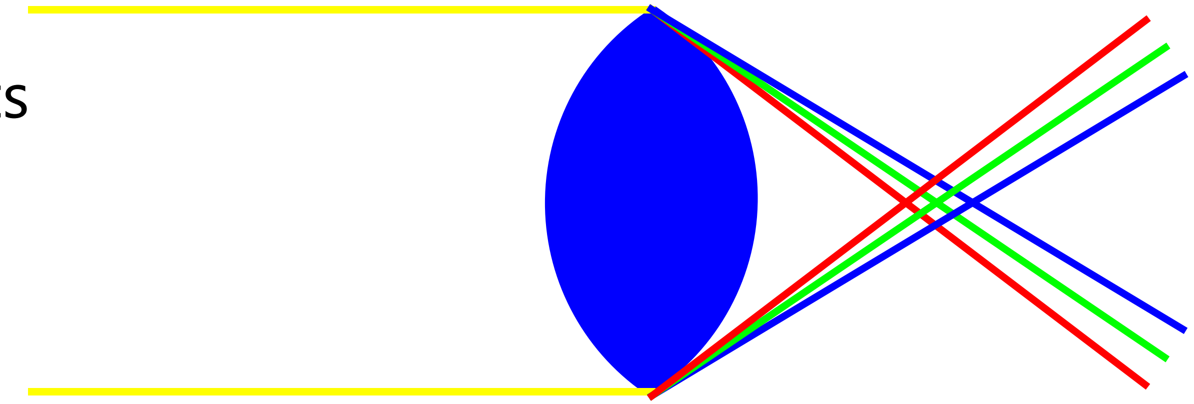
- Different wavelengths refracted differently

- Implementation by usage of different refraction indices
  - Introduce three independent render passes (red, green, blue)
  - Every pass $\Rightarrow$ different refraction index
  - Finally pixel values of three images added

- Resulting image shows color shift at edges of objects
  - Stronger dispersion $\Rightarrow$ clearer shift

[1]

Real world example of chromatic aberration



Chromatic aberration at lens

[1] Https://www.fotolehrgang.de/images/im2/ca.jpg

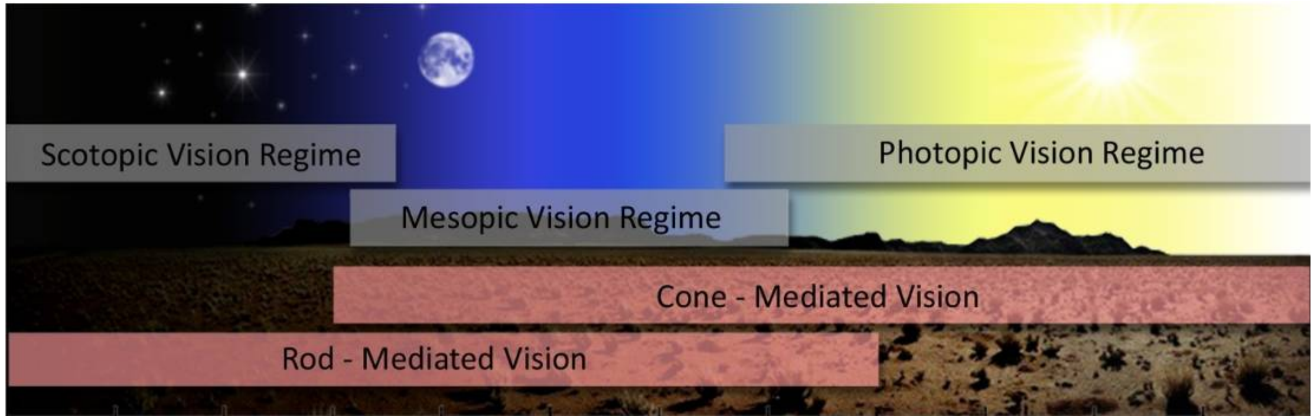
Soiled lens

- Particles on lens common disease
- Causing dark shadows in eye sight
- Simulation by placing particle(s) in front of lens
  - Construction via triangle mesh
- Corresponding to this water drops placed in similar way
  - Simulate tears and water droplets

Photopic, scotopic and mesopic Vision

- Central eye function is adaption to different light intensities

- Different perception levels
  - Photopic $\Rightarrow$ daylight
  - Mesopic $\Rightarrow$ twilight
  - Scotopic $\Rightarrow$ night



Different vision types in day to night scenario [1]

- Simulation requires several steps and components
  - Ambient brightness and lighting of scene
  - Information on receptor behavior at different brightness levels
  - Application of information to color values of pixels

[1] https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQWGF86qRc06Z1b5uO6Exp_G24ZDCJB_i0yBA&usqp=CAU

Ambient light

- Realizing ambient brightness ⇒ ambient lighting introduced
  - Represents general brightness
  - Using Phong lighting model
  - Ambient term

  = material constant

  = incident light intensity

$$I = k_a I_a + k_d \left(I_d \cdot \left(\hat{L} \cdot \hat{N} \right) \right) + k_s \left(I_s \cdot \left(\hat{R} \cdot \hat{V} \right)^n \right)$$

$$I = k_a I_a$$

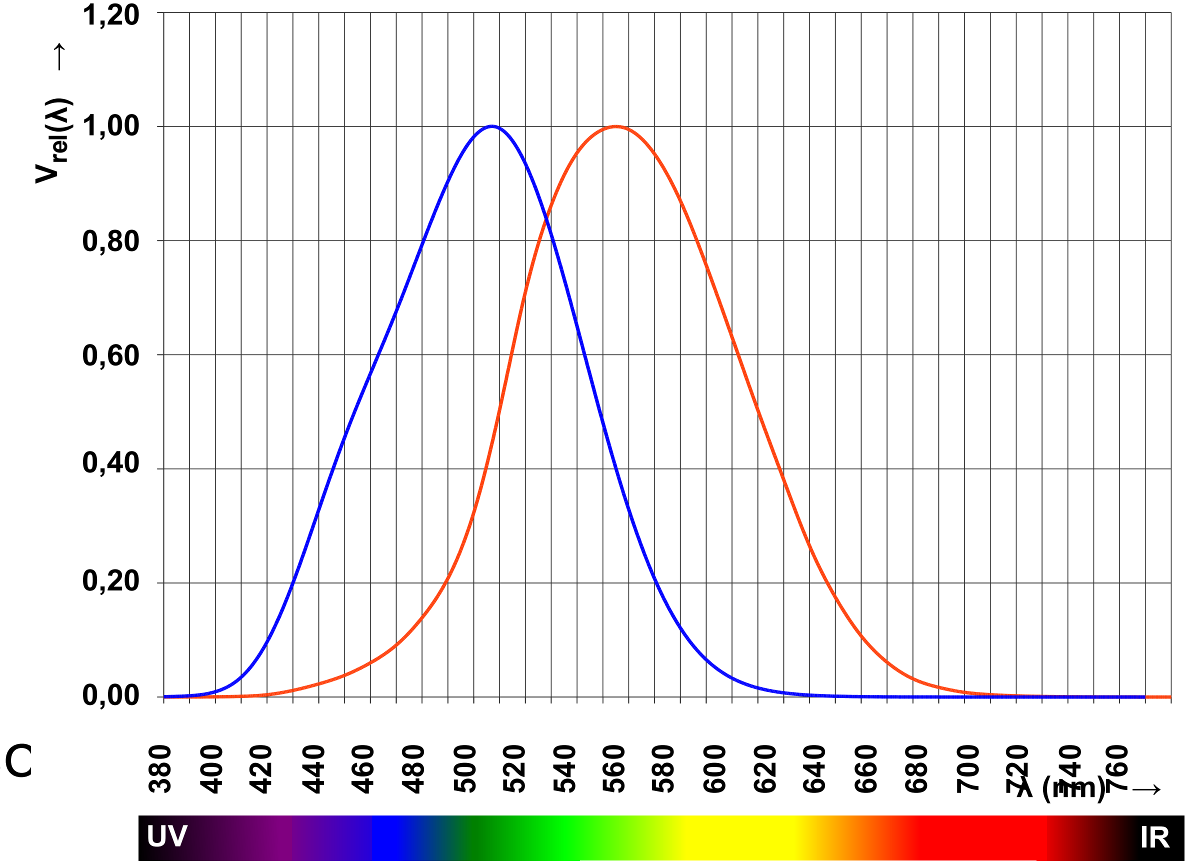
$$k_d \left(I_d \cdot \left(\hat{L} \cdot \hat{N} \right) \right)$$

$$k_s \left(I_s \cdot \left(\hat{R} \cdot \hat{V} \right)^n \right)$$

- Enabling simulation of differences in lighting by implementing other terms

Receptor behavior

- Simulation of rods and cones on retina
- Different thresholds levels
  - Above maximum ⇒ photopic vision
  - Below minimum ⇒ scotopic vision
  - Brightness levels between thresholds lead to Mesc



[1]

- In order to simulate vision types V and V‘-Lambda curves exists
  - V-Lambda ⇒ photopic, V‘-Lambda ⇒ scotopic
  - Mesopic vision interpolated between V and V‘ curve

V-Lambda Curve (blue)
V‘-Lambda Curve (red)

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

[1] https://commons.wikimedia.org/wiki/File:V-lambda-phot-scot.svg

Application on pixels

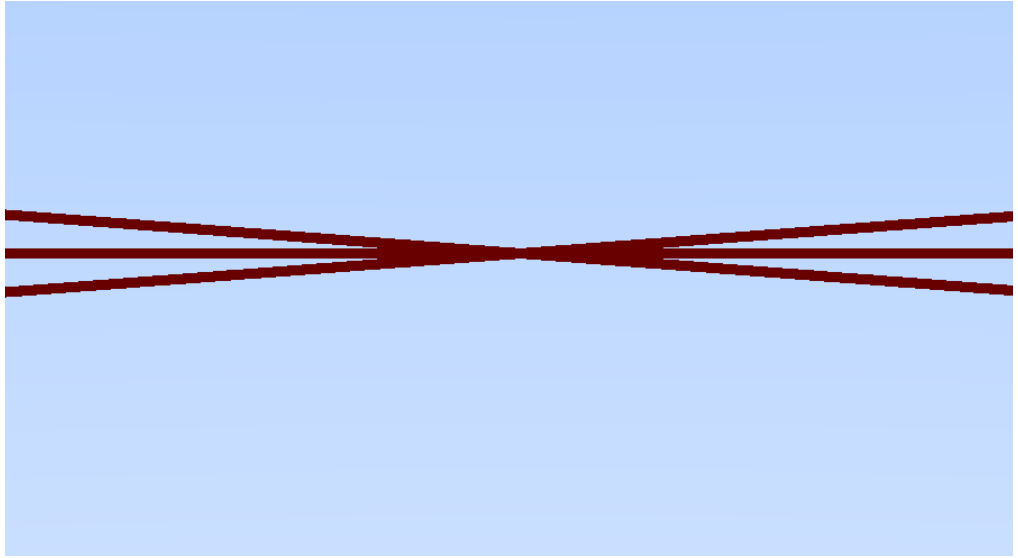
- Brightness to be determined
  - Ambient lighting value of objects
  - Lighting values assigned to corresponding receptor curve

- Curve is applied to pixels values as weights
  - Continuous curve, discrete values to be chosen
  - red $\Rightarrow$ 600 nm; green $\Rightarrow$ 540 nm; blue $\Rightarrow$ 440 nm

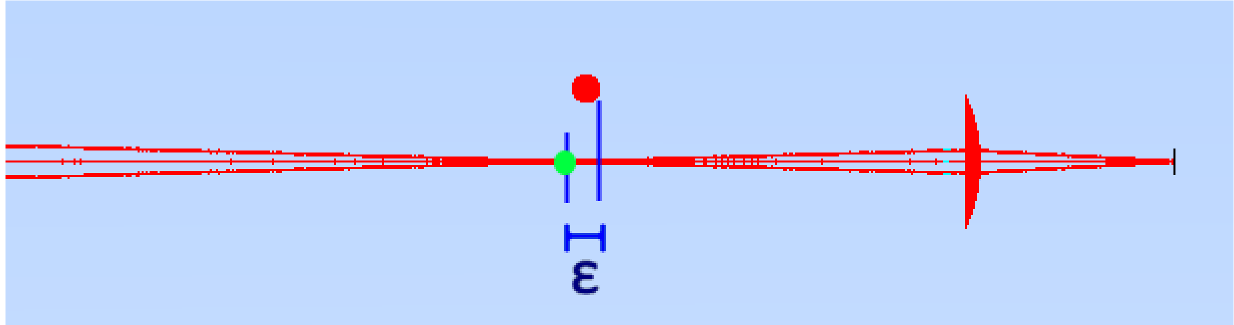
- For mesopic view, V and V‘ curves interpolated

Results

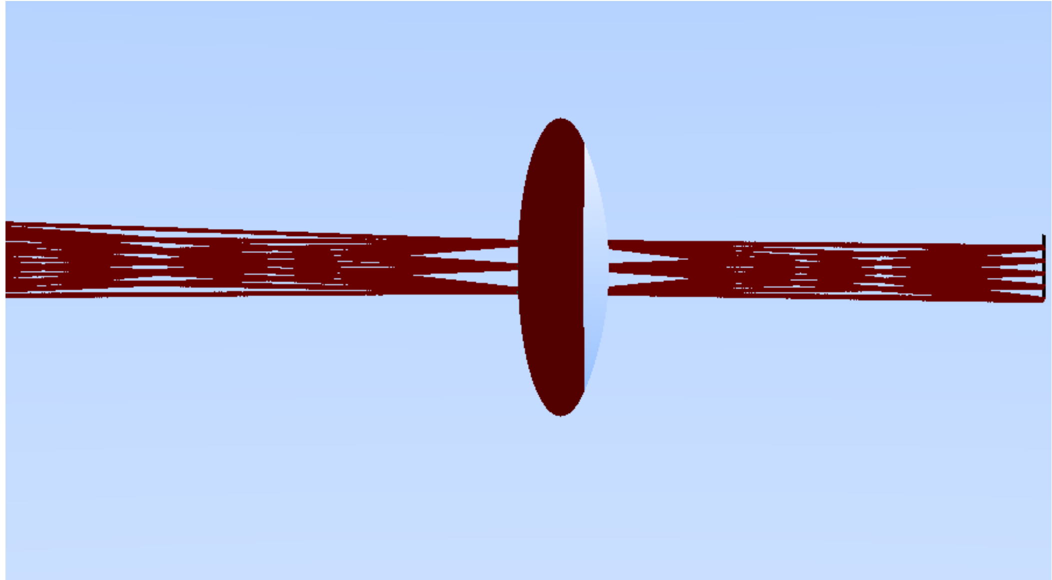
Results simulation eye model



Light Rays forming focal point



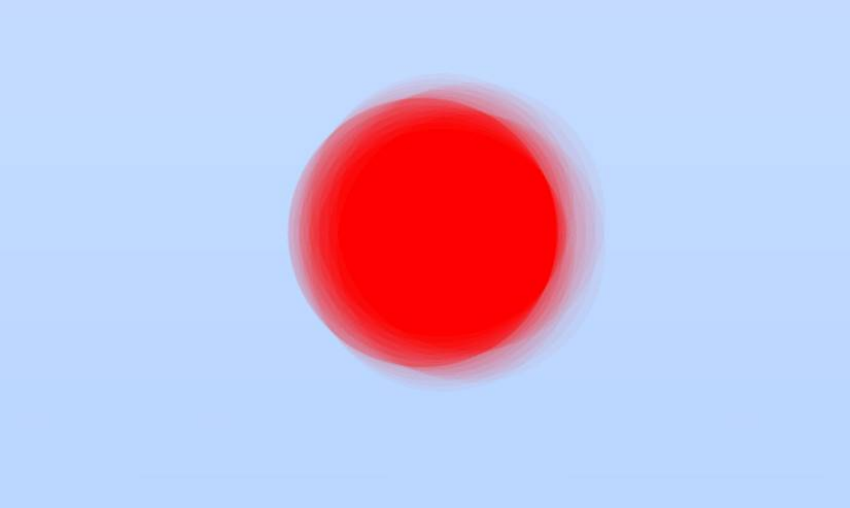
Side view with sphere at position of focal point (green). Small error visible(ε)



Ray bundles sorcing at retina falling through lens

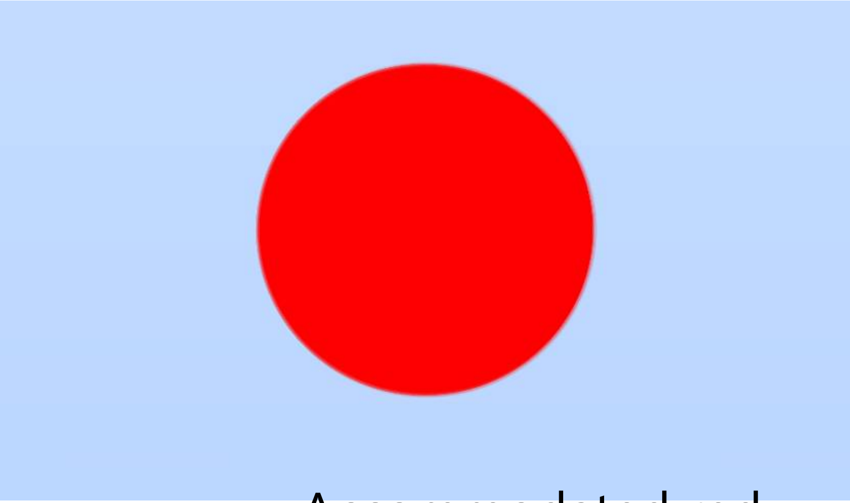
VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Results simulation eye model

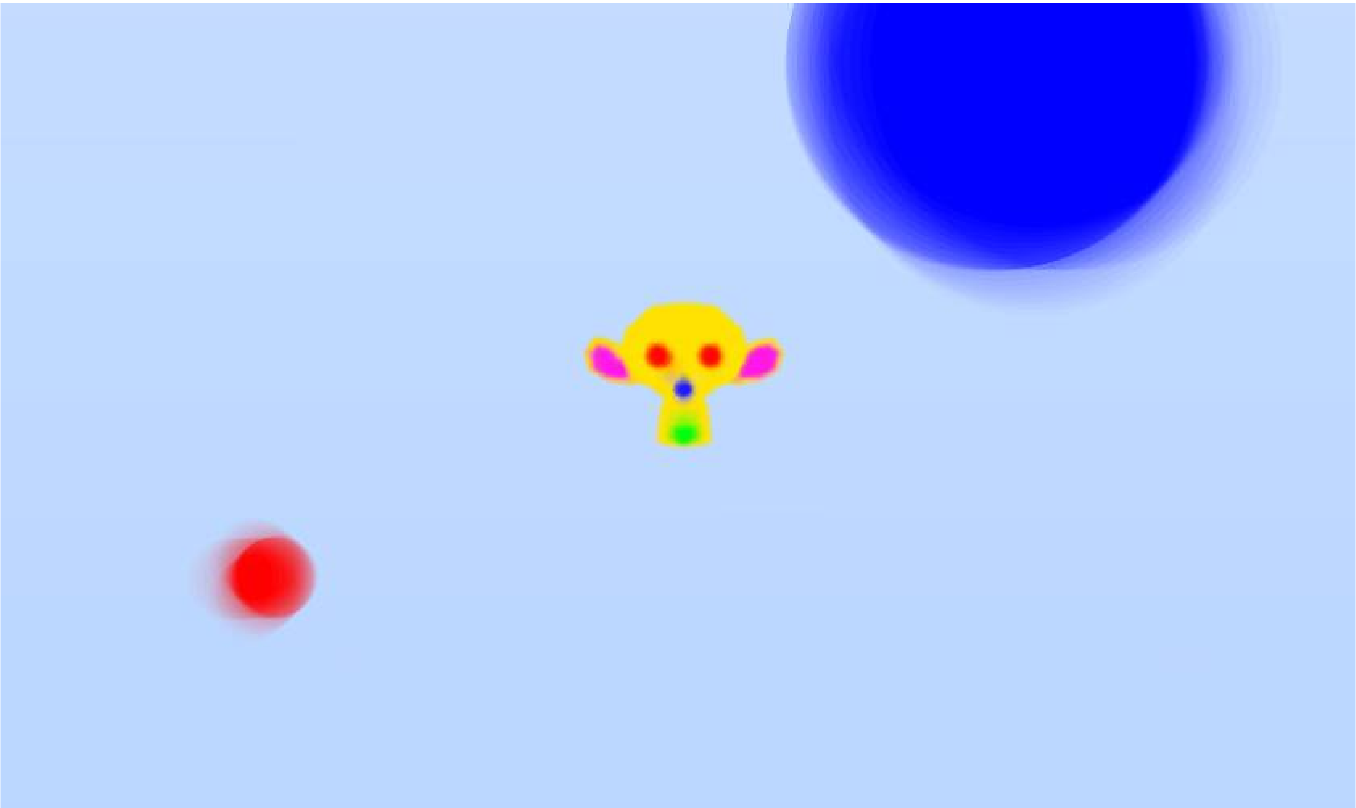


Unaccommodated red
Sphere

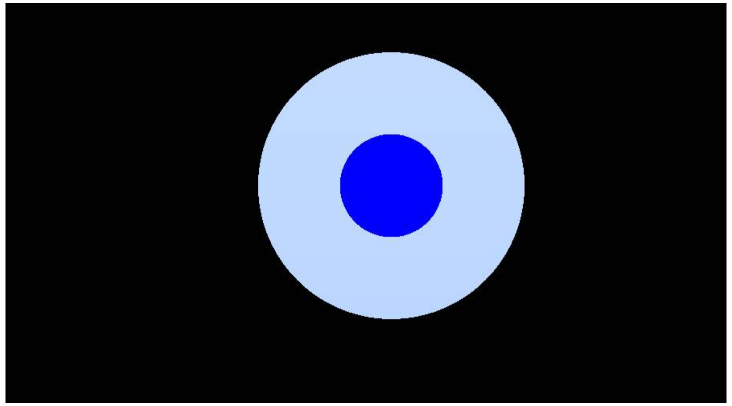
Unaccommodated red (far) and blue (near) spheres.
Monkey head Accommodated, distance between spheres.



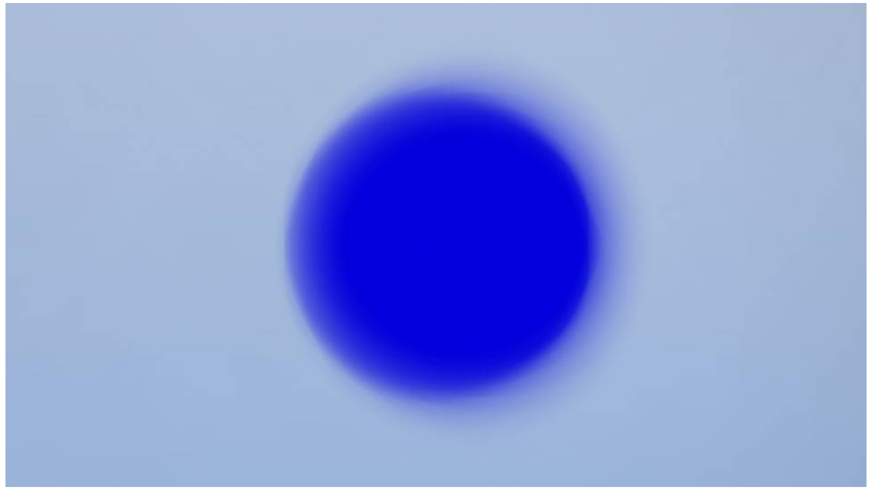
Accommodated red
Sphere



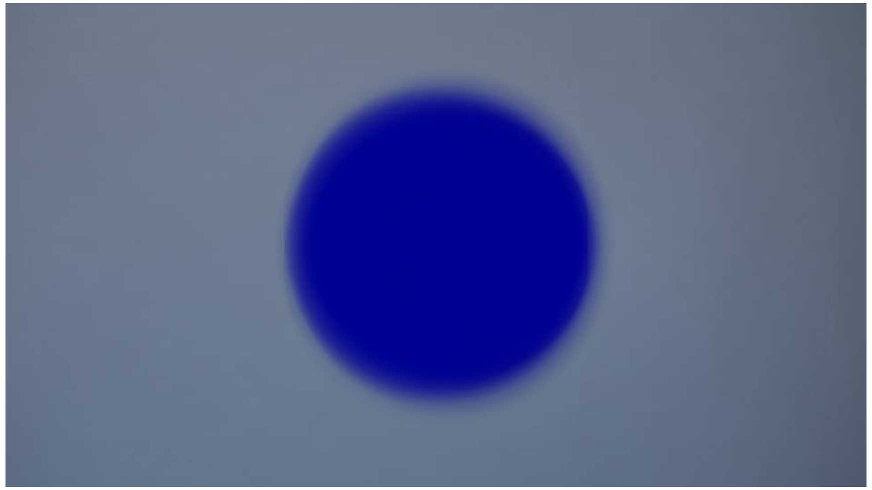
Results simulation light regulation



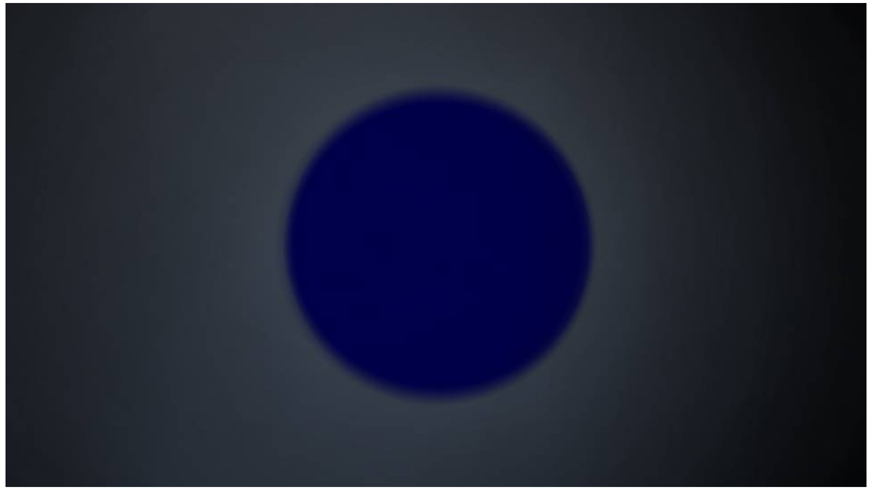
Almost closed iris, one ray per pixel



Wide opened iris, high resolution



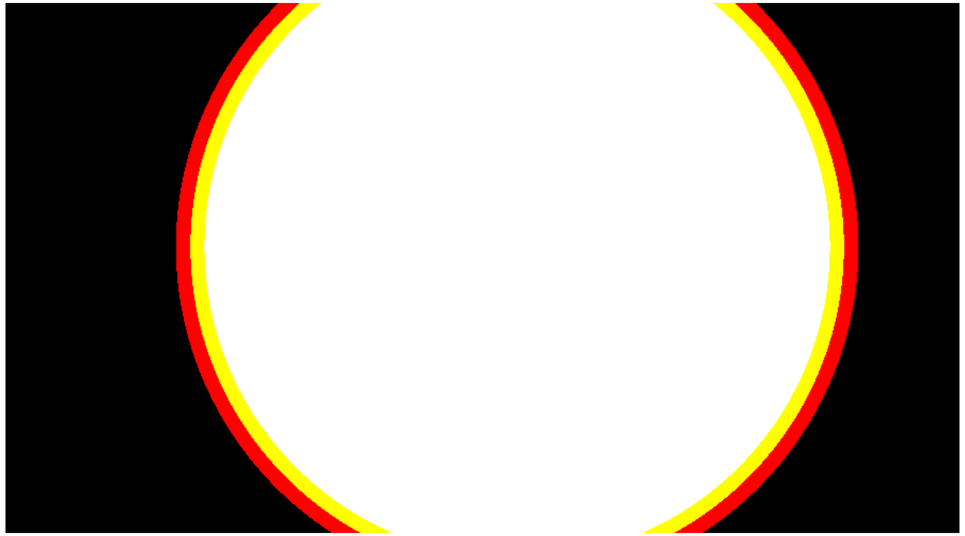
Half way opened iris, high resolution



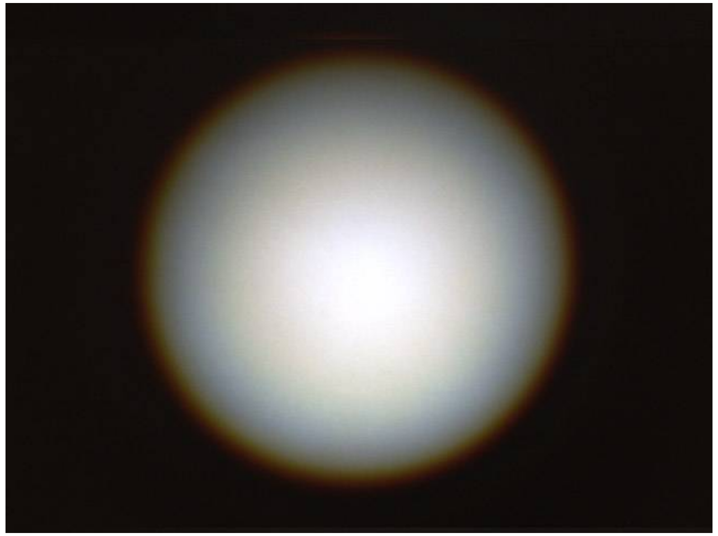
Almost closed iris, high resolution

VISUAL COMPUTING GROUP
HEIDELBERG UNIVERSITY

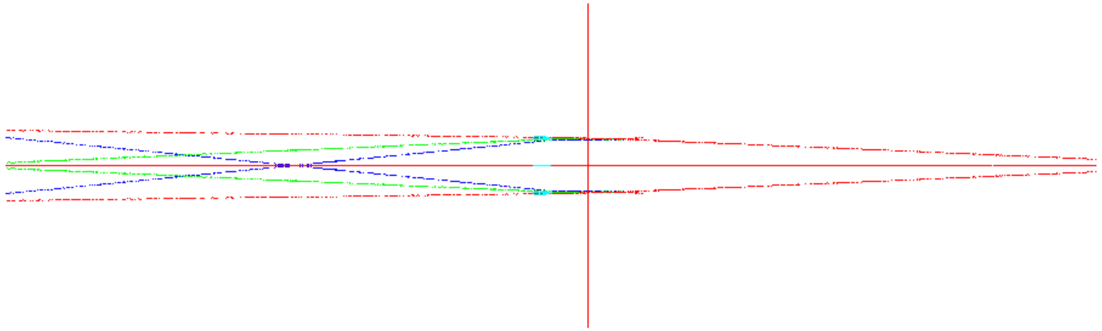
Results simulation chromatic aberration



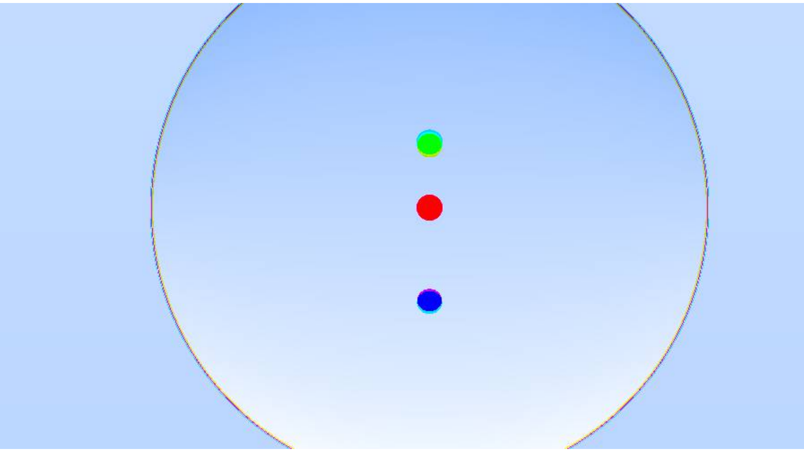
White Sphere with chromatic Aberration, only axial ray



Real World example of chromatic Aberration at white light cone



Side view of scene with light rays splitting by color at lens

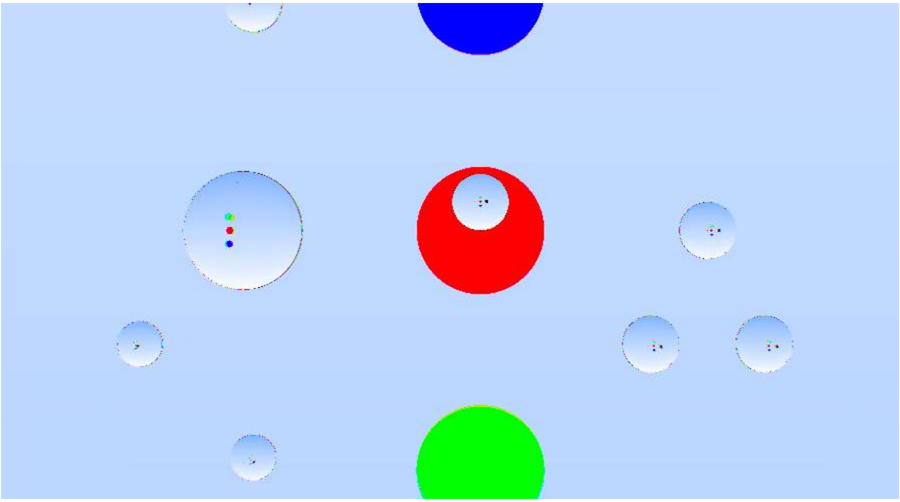


Chromatic Aberration at glass sphere and colored spheres

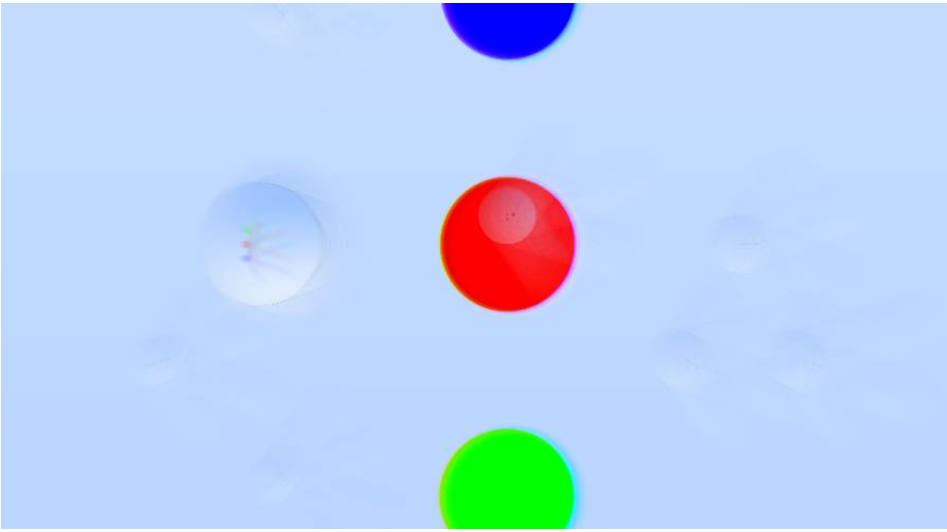
Results simulation soiled lens

Soiled lens, low resolution

Soiled lens, high resolution



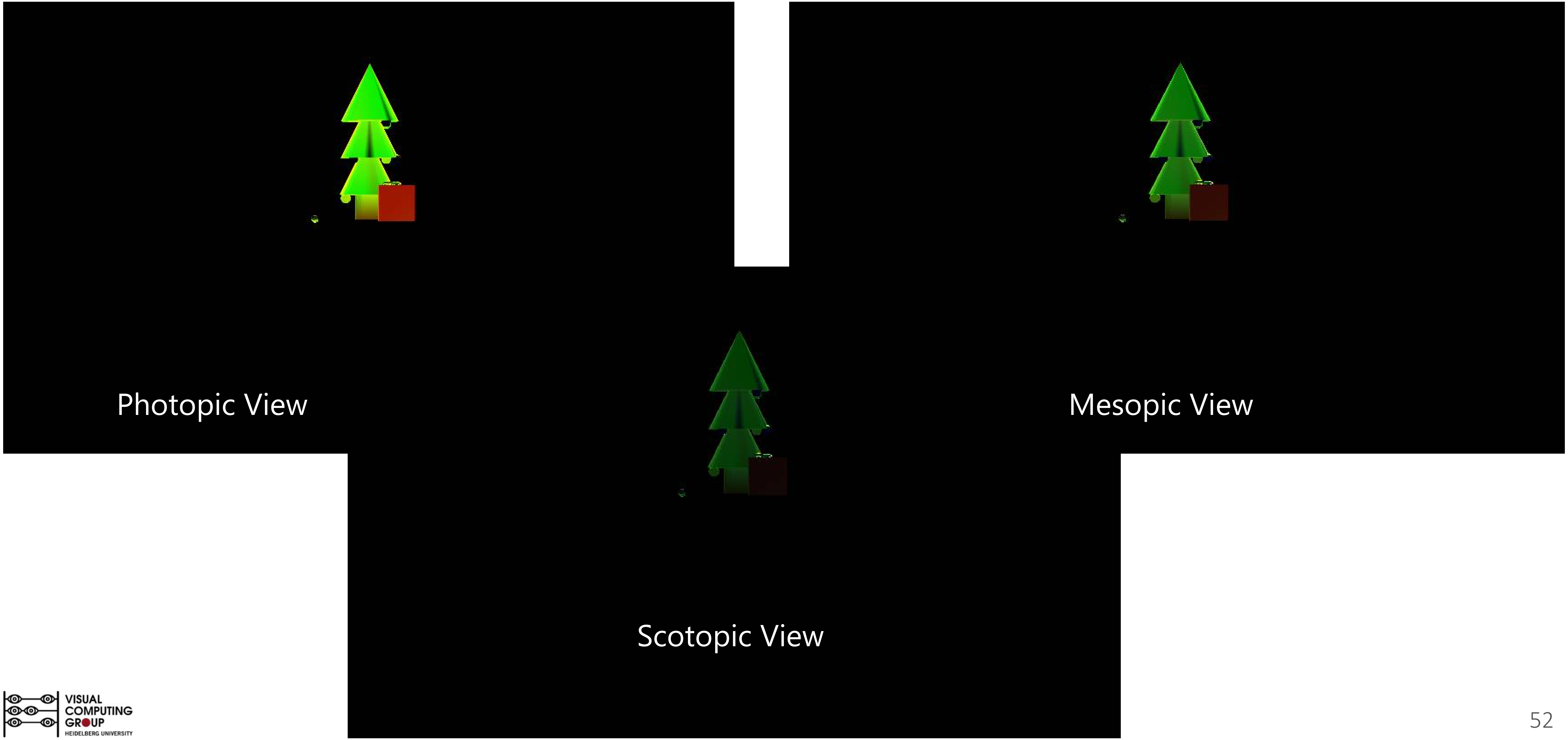
Water drops, low resolution



Water drops, high resolution

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Results simulation photopic, scotopic, mesopic vision



Limitations

- Incomplete Simulation of the eye
- Reduced part complexity
- Limitations by software

Conclusion

- Realized basic model for biological raytracing
  - Simulated parts of eye model
  - Introduced way to simulate ray traversal through eye

- Simulated visual effects on basis of model
  - Physical phenomena (e.g. Chromatic aberration)
  - Resulting by eye functions (e.g. Mesopic vision)

VISUAL COMPUTING GROUP HEIDELBERG UNIVERSITY

Outlook and Future Work

- More detailed Simulation of eye parts
  - Lens
  - Retina
  - Additional parts
- Methods to improve performance
  - Hardware acceleration (Vulkan,DirectX12)
  - Datastructures