

Rendering of Newton's Rings in Monochrome Light

Victor Debelov¹ and Nikita Dolgov²

¹ *Institute of Computational Mathematics and Mathematical Geophysics of Siberian Branch of Russian Academy of Sciences, 6, Ac. Lavrentieva ave., Novosibirsk, 630090, Russia*

² *Novosibirsk State University, 1, Pirogova str., Novosibirsk, 630090, Russia*

Abstract

Newton's rings are an interference pattern related to fringes of equal thickness. This image can be obtained on a simple experimental optical setup. Modern common renderers, based on zero-thickness ray tracing, calculate highly realistic images of complex 3D scenes, which are computer models of scenes from the real world. However, they do not allow you to reproduce such phenomena as interference, because they ignore even the polarization of light. Interference is studied by physical optics, and it is natural to assume that if the calculation is based on the "tracing" of waves in the scene, this problem can be solved. An algorithm is known when a solid beam of light is used instead of a light wave. The results of the calculations show images of the interference effects; Newton's rings are also calculated.

This is an acceptable solution for simple scenes involving a few objects. It is also good for optical design systems, when the result is important, and not the time spent. But not practical for universal renderers, which must calculate the image in an acceptable time for very complex scenes. In this paper, we propose an algorithm based on the traditional method of tracing paths consisting of zero-thickness rays. Only on the last ray of a path that crosses the picture plane is the modification made. It is assumed that these rays characterize spherical wavelets. In this paper, we consider the results of applying the mentioned heuristics to classical optical experiments.

Keywords

Interference, ray tracing, polarization, pixel color, fringes of equal thickness, Newton's rings

1. Introduction

Interference can significantly change the visual perception of a scene without any changes in its geometry and the optical properties of objects. It depends on the positions of the light sources and the camera or observer. In this paper, we will only consider ray tracing and monochrome stage lighting for the calculation. Monochrome lighting allows you to most clearly trace the effect of the calculation algorithm on the image without hindrance, which can be introduced by waves of different lengths. Recall that only coherent rays can interfere: two or more [1], i.e. rays that are emitted by a single point source or a small differential neighborhood of an extended source.

The interference can be calculated in the spectral rendering mode only, when the scene is rendered for each wavelength separately, and the intensity is associated with the ray. Alternatively, in monochrome lighting mode. Modern renderers do not allow you to reproduce such phenomena as interference, because they ignore even the polarization of light. In computer graphics, there are works devoted to algorithms for calculating more or less physically based images of interference effects. In the work [2] many of them were considered from different points of view, but here we will pay attention to only one characteristic: how these algorithms contribute to the intensity of the pixel of the picture plane.

In the works [3-7] and a number of others, the main attention was most often paid to the problem of calculating a photorealistic image of interference in thin films. Soap bubbles and varnish coatings are

GraphiCon 2021: 31st International Conference on Computer Graphics and Vision, September 27-30, 2021, Nizhny Novgorod, Russia

EMAIL: debelov@oapmg.sccc.ru (V. Debelov); nikitabrn1999@gmail.com (N. Dolgov)

ORCID: 0000-0002-7577-4700 (V. Debelov); 0000-0001-5622-3586 (N. Dolgov)



© 2021 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings (CEUR-WS.org)

especially popular. In these works, the interference is calculated by dividing the wave amplitude. In fact, the authors, within the framework of a common renderer, when a tracing ray hit an interference object, called a special additional module that performed the interference calculation and returned the ray to the renderer with a new intensity. Undoubtedly, this approach freed the module developers from the need to monitor the impact of the ray on the picture plane. Therefore, it is possible to include the interference calculation module in an existing renderer. It is worth noting that the authors, calculating the interference of two rays inside a special module, often did not specify how the intensity of the resulting output ray is obtained. And this is a task similar to combining two coherent rays in a pixel. However, Glassner [4] specified this process, moving at this stage from rays to spherical waves. The right-hand scene in Figure 1 shows the situation with the division of the amplitude, which can be processed by the mentioned algorithms. In the mentioned works, a special module calculates interference of the rays $r1$ and $r2$ inside itself, and outputs a ray $rOut$ in the renderer format. In the case when coherent rays fall on the picture plane, passing through different paths in the scene, this approach does not work, in Figure 1: The interference of two coherent rays is calculated on the picture plane (left) and two coherent rays obtained by dividing the amplitude (right) this situation is shown on the left scene. And, therefore, in such cases, the interference effect may be missed and will not be visible in the image.

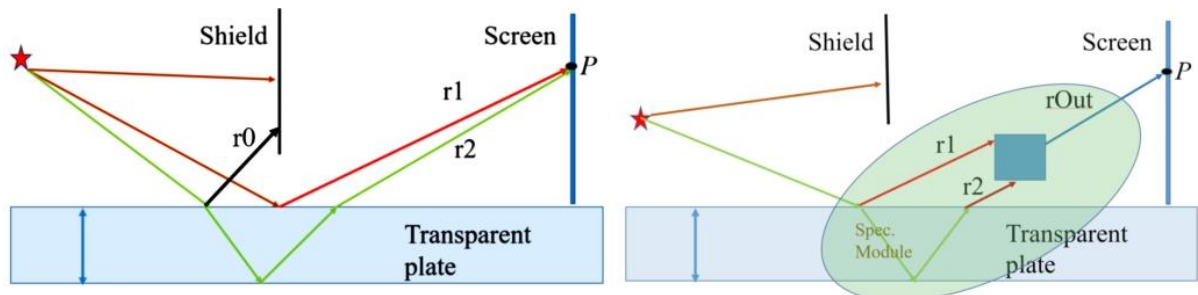


Figure 1: The interference of two coherent rays is calculated on the picture plane (left) and two coherent rays obtained by dividing the amplitude (right)

Interference is studied by physical optics, and it is natural to assume that if the calculation is based on the "wave tracing" in the scene, then this problem can be solved most physically correctly and accurately. For example, in the paper [8], instead of a light wave, a solid beam (or fat ray) of light is used, which has geometric dimensions that depend on the surrounding objects. The results of the calculations show images that realistically and physically correctly show the interference effects. For the first time, the results of rendering the photorealistic image of Newton's rings calculated under monochrome illumination are demonstrated. This is an acceptable solution for simple scenes involving a few objects. This is also acceptable for optical design systems, where the result is important, not the time spent. But it is impractical for universal renderers, which must calculate the image in an acceptable time for very complex scenes. Moreover, it is not yet clear whether it is possible to perform such beam tracing in any scene. Note that in the work [8] the emphasis is on optical design. Consequently, many scenes have a rather specific geometry similar to geometry of microscopes, which Bloss called the "optical tube" [9], in which it is quite convenient to trace the wave in the form of a solid beam.

In computer graphics, various tracing shapes were studied – a cone, a beam, a pencil (a beam with a central ray). But all of them are not used in practice [10], it is more practical to use Monte Carlo tracing with zero-thickness rays.

Newton's rings are an interference pattern referring to fringes of equal thickness. This image can be obtained on a simple experimental optical setup and photographed. We have not seen images of Newton's rings calculated on the basis of zero-thickness ray tracing in the literature. This was the motivation to choose fringes of equal thickness as the target test for the algorithm being developed, and especially the Newton rings. We propose an algorithm based on the traditional method of tracing paths consisting of zero-thickness rays. Only on the last ray of each path that crosses the picture plane is the modification made. It is assumed that these rays characterize spherical wavelets. In fact, the proposed approach is the next stage in the development of physically based algorithms for calculating photorealistic images of interference effects based on zero-thickness ray tracing.

The second section discusses the main elements of a virtual setup or 3D scene for numerical experiments. The third section is devoted to the description of the algorithm. The fourth section presents the results of numerical experiments. The fifth and sixth sections contain the conclusions and acknowledgements.

2. Virtual setup for numerical experiments

A virtual setup is a 3D scene that should contain a light source and a screen, and then we can trace the paths of the rays in the scene as a whole. The scene also contains a certain number of objects, from the interaction with which the necessary branching paths occur. We prefer optically isotropic transparent objects, from the meeting with which the rays are bifurcated. The optical literature deals with such scenes in the most detail, since most of these books are devoted to the description of the operation of interferometers, for example, see [1].

2.1. Light source

The light source L is a homogeneous rectangular area that emits linear polarized monochrome light with a λ wavelength. In the experiments, the source is represented by a rectangular grid of nodes that play the role of point sources of directional linear polarized light. Each node $L_{i,j}$ has its own unique identifier L_{id} and emits the light ray of intensity I strictly in the L_d direction. We set the linear polarization using the coordinate right-hand system associated with the ray:

$$\{\mathbf{X}, \mathbf{Y}, L_d\}.$$

In this implementation, we chose the coordinate system so that the oscillations of the electric vector occur along the \mathbf{X} axis.

In experiments, light can be polarized, partially polarized or unpolarized. A wave or ray of unpolarized light is modeled as the sum of two linearly polarized rays (waves) with different identifiers L_{ida} and L_{idb} with an intensity of $I/2$, in which the vectors of the electrical component oscillate in mutually perpendicular planes [1]. Note that both components of the decomposition of unpolarized light are independent and do not interfere, i.e., when rendering, they can be considered as independent point sources.

Partially polarized light can be represented as the sum of the unpolarized and polarized parts. In the monochrome version, there is no need to use spectral rendering, since only one wave of light is used, but the wavelength itself is necessary for performing calculations.

2.2. Scene objects

In our experiments, optically isotropic transparent objects play the main role, for which the refraction index is specified. The interaction of the tracing ray with them is calculated based on the application of the Fresnel equations [1, 11].

2.3. Screen & rendering

A rectangular screen S consisting of square pixels is located on a picture plane. Naturally, any image is represented by a matrix of intensities. The source light can be represented as the sum of up to three (in the case of partial polarization) independent linear polarized components. In this regard, it is possible to render three times for each component separately, getting three independent matrices of intensities, which are then simply summed element by element.

2.4. Problem

It is necessary to develop an algorithm: a) based on the traditional method of tracing paths consisting of rays of zero thickness; b) allows one to calculate images visually similar to photos; c) calculate

images visually similar to the calculated image of Newton's rings from the work [8]. Numerical experiments are performed for monochrome lighting in the scene.

Does this result meet the criteria of photorealistic computer graphics? Yes, quite. It is enough to recall Phong's formula for specular reflection [10], i.e. a phenomenological simulation approach. In our opinion, in the work [4] Glassner was quite accurate: "As almost always seems to be the case, writing a good shader seems to involve some judicious trading off of accuracy and realism with approximations and pragmatism. I mean, we could simulate all of this at the molecular or even atomic level, but it wouldn't show up in the results. The trick is to find a nice balance between simplicity, efficiency, and verisimilitude".

It is obvious that in this paper we do not give a complete solution to the problem. Here we are going to formulate new physically based approach to rendering interference phenomena, which is supported by the results of numerical experiments.

3. Proposed algorithm

Let us represent a linear polarized zero-thickness tracing ray with attributes based on the works [2, 11] in the form

$$R = \{P_0, \mathbf{d}, \mathbf{X}, \mathbf{Y}, I, \lambda, L_{id}, Op, \Sigma\}$$

where: $\{P_0, \mathbf{d}\}$ is the mathematical ray, $\{\mathbf{X}, \mathbf{Y}, \mathbf{d}\}$ is the associated right-hand coordinate system, P_0 is the origin of the ray, \mathbf{d} is the direction. The oscillations of the electric vector of electromagnetic wave occur along the \mathbf{X} axis, I is the intensity, λ is the wavelength of the light, L_{id} is the identifier of the point light source that generated the ray. Two rays are coherent if their source IDs are non-zero and match. Op is the optical path of a geometric path traveled from the source, which is used to calculate the current phase of the electromagnetic wave [1]. Σ is the phase jump accumulated during reflections from a denser medium, see [2, 11].

The generated rays (reflected and refracted) inherit some of the attributes, or they are recalculated during the contact of the incident ray with the scene surface. Note that the Fresnel equations apply.

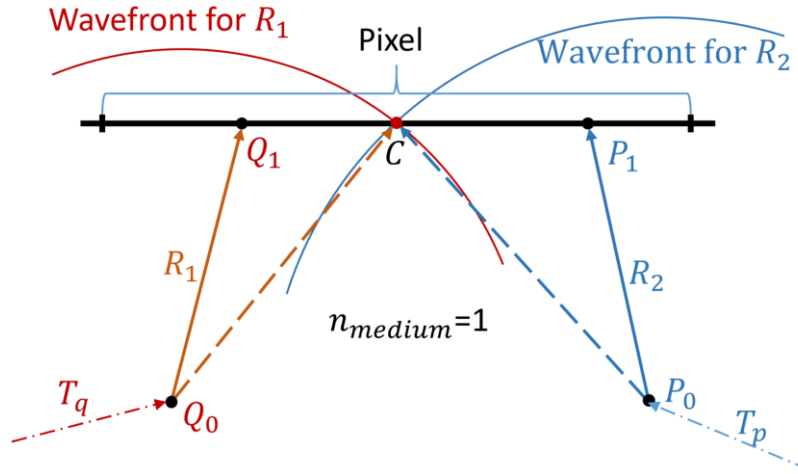


Figure 2: Correction of the optical path of the ray

One of the most important issues that arise when calculating the interference in a pixel is taking into account a path T of a ray from the light source to certain scene point P . Let us denote its optical path $OP(T, P)$. Let us consider the example shown in Figure 2: Correction of the optical path of the ray. Let the two tracing paths T_q and T_p end with coherent rays R_1 and R_2 , respectively, which fall into one pixel on the screen. In this case, the ray R_1 hits the point Q_1 ; the ray R_2 hits the point P_1 . The optical path of a single segment is equal to its length assuming that the refractive index of the scene space medium $n = 1$, see [1, 11].

In the general case, the points Q_1 and P_1 are different, but the interference calculation is only possible for coherent rays hitting one point of the screen. Both rays have traveled some unknown paths through the scene, and, in fact, the shapes of the fronts of the waves they represent are unknown.

We apply the hypothesis (heuristic) that both waves have a spherical front and assume that the coherent waves intersect at an intermediate pixel point $C = (Q_1 + P_1)/2$, which is defined as the intersection of circles with centers at Q_0 and P_0 , and interfere.

Thus, instead of the segment Q_0Q_1 , we consider the segment Q_0C , and instead of P_0P_1 , the segment P_0C . According to the heuristic we compute the new optical paths:

$$OP(T_q, C) = OP(T_q, Q_0) + Q_0C,$$

$$OP(T_p, C) = OP(T_p, P_0) + P_0C.$$

The monochromatic light source and the screen have grid representations. The grid nodes of the light source are the starting points of the paths, and the screen nodes are the pixel centers. As a rule, the experiments initially selected a pixel size of 0.01 mm. The grid step of the source is experimentally selected so that at least one pair of coherent rays fall into each pixel. If no pair of coherent rays hit any pixel on the screen, the source grid step decreased. Each pixel of the screen stores the following data:

- I_{pairs} is the total intensity introduced by pairs of coherent rays falling into this pixel of the screen,
- N_{pairs} is the number of pairs of such rays.

Then the resulting intensity in a pixel is equal to I_{pairs}/N_{pairs} . We considered to use the original values of the optical paths $OP(T_q, Q_1)$ and $OP(T_p, P_1)$ in the calculation. But after a series of experiments, we came to the solution described above. To calculate the interference term, it is necessary to know the difference between the optical paths $OP(T_q, Q_1)$ and $OP(T_p, P_1)$ in order to obtain the phase difference $d\phi$, which is calculated by the formula

$$d\phi = 2\pi(OP(T_q, Q_1) - OP(T_p, P_1)) / \lambda,$$

see [1, 11]. On the other hand, a pixel of 0.01 mm at a wavelength of 500 nm fits a hundred wavelengths, and there is too vague a spread. So we settled on the described solution.

Also in the first version of the proposed algorithm, the coherent rays crossing the picture plane in different pixels, we simply discarded from consideration. Therefore, the part of the energy emitted by the source was simply not taken into account. In the final version, only rays were discarded if the distance between the points of their intersection with the picture plane was greater than ε to reduce energy loss. In experiments, ε is equal to the pixel diameter. And the result of the interference is attributed to the pixel containing the point C . Thus, in many cases, it is possible to avoid reducing the source grid step due to empty pixels and not depend on borders between pixels.

4. Numerical experiments

Here we will present three installations that are described in sufficient detail in the literature on optics: an air wedge between two glasses, Newton's rings in transmitted light, and Newton's rings in reflected light. All glass objects are specified with a refractive index of 1.5. The refractive index of the medium is 1.0.

4.1. Air wedge

A similar installation is considered in [12], where a wedge is created by superimposing two thick, well-polished plane-parallel glass plates. A strip of thin paper is placed between the edges of these plates on one side. Thus, a small wedge-shaped air gap is formed between these plates, see Figure 3.

In the simulation of this experiment, it was decided to take into account not two, but four rays from one source, since all four rays carry intensities of about the same order. Rays R_1, R_2, R_3, R_4 are coherent. It was found experimentally that the location of the lens and the screen is that in all cases rays R_1, R_3 fall into one pixel, and the rays R_2, R_4 fall into another. It is impractical to consider more rays, because the rays obtained at a greater trace depth have a much lower intensity and do not significantly affect the interference pattern. Thus, the first two rays interfere in one pixel, and the second two rays in

the other. Figure 4 shows the variants of images that were computed at the same scale, i.e. at the same maximum intensity.

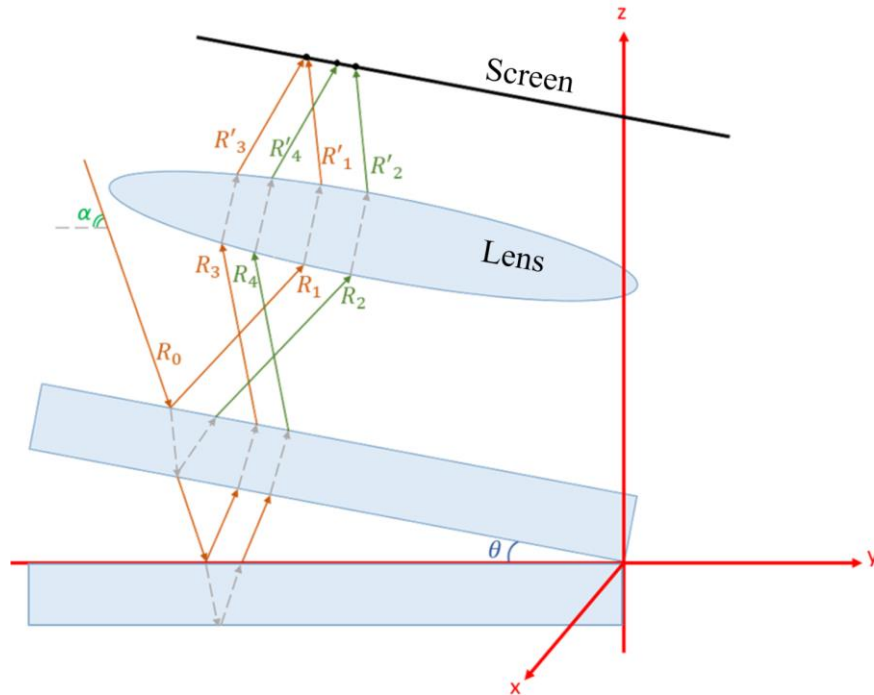


Figure 3: Diagram of the experimental scene and the path of the rays



Figure 4: Different rays are taken into account in the calculations of interference fringes: left) only rays R_1 and R_3 ; center) only rays R_2 and R_4 ; right) all four rays

4.2. Newton's rings in transmitted light

The setup for obtaining Newton's rings in transmitted light (see Figure 5 on the right) consists of a mercury lamp, a light filter, an installation for obtaining rings, a collecting lens, a shield, and a screen. The shield is an opaque circle in which a round hole is cut to limit the size of the image on the screen. Since we use monochromatic light, we do not need a filter. We also abandoned the shield in order to get an image of the rings on the entire screen, compare images shown in Figure 6. The virtual scene for obtaining Newton's rings in transmitted light is a glass plate, with which a plano-convex lens with a large radius of curvature $r_L = 2$ meters is in contact. Figure 5 on the left shows only the rays of interest in the scene. The ray of light R_0 falls normally on the flat surface of a plano-convex lens and is refracted. Then it passes through the lens and is refracted at the point A . Next, the ray passes through the air layer and falls on the glass plate at the point B .

R_1 is the ray that passed through the glass plate. The ray reflected from the plate at point B falls on the plano-convex lens at the point C . This ray, reflecting, falls on a glass plate and passes through it, this is the ray R_2 . The rays R_1 and R_2 are coherent and interfere.

The images shown in Figure 6 confirm the photorealism of the image obtained because of the calculation based on the zero-thickness ray tracing.

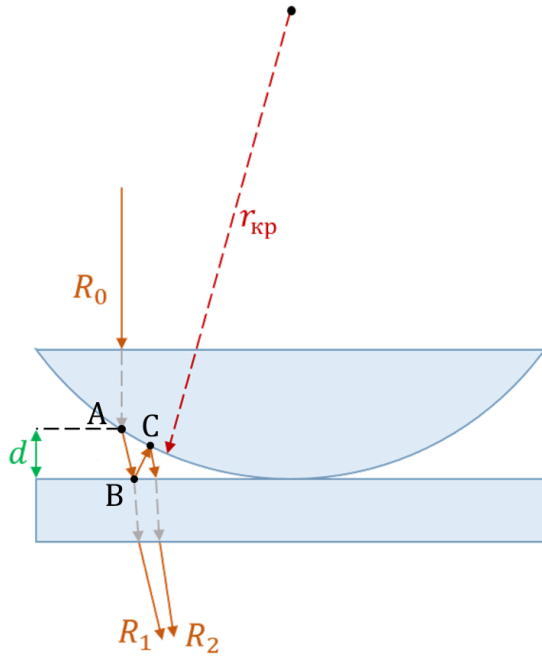


Figure 5: On the left: the passage of the rays through the ring-making installation; on the right: the scene of the experiment on obtaining Newton's rings in transmitted light [13]

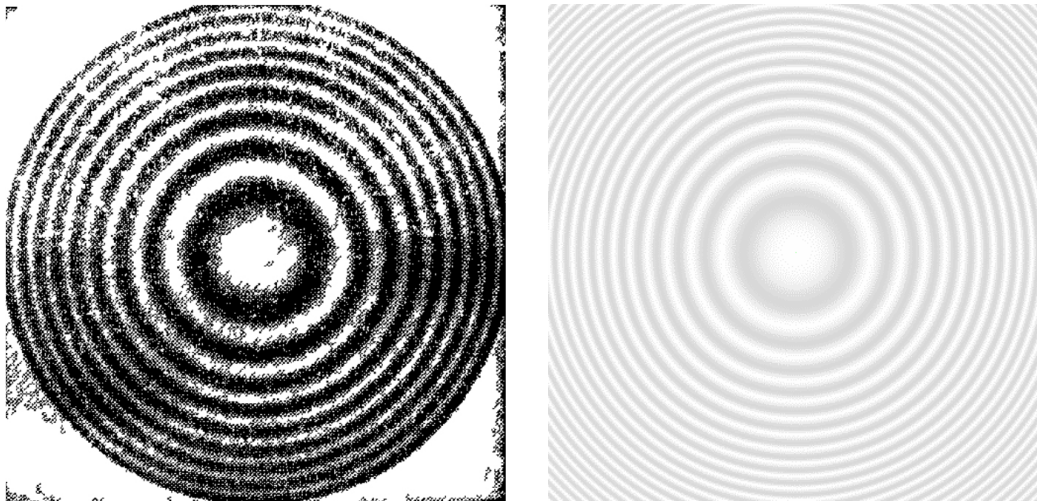


Figure 6: From left to right: the photo from [1] and the result of the zero-thickness ray tracing calculation, cut from the screenshot

In [12], a practical application of Newton's rings is given, namely: by the number m and the radius r_m of a ring, the radius of curvature r_L of the plano-convex lens can be determined using the formula:

$$r_L = \frac{2r_m^2}{(2m-1)\lambda}.$$

We checked this fact using the screenshot, see Figure 7. Given that $\lambda=500$ nm, and the pixel size is 0.005 mm, we perform the following calculations:

- If $m = 1$, $r_L = \frac{2r_1^2}{\lambda} = \frac{2((142-0.5) \cdot 0.005)^2}{0.0005} \approx 2002,2\text{mm} \approx 2\text{m}$,
- If $m = 6$, $r_L = \frac{2r_6^2}{11\lambda} = \frac{2((470-0.5) \cdot 0.005)^2}{0.0055} \approx 2003,9\text{mm} \approx 2\text{m}$.

The obtained values are fairly accurate estimates of the lens radius. Note that this fact also confirms the high photorealism of the calculated image. Obviously, the error is related to the pixel size.

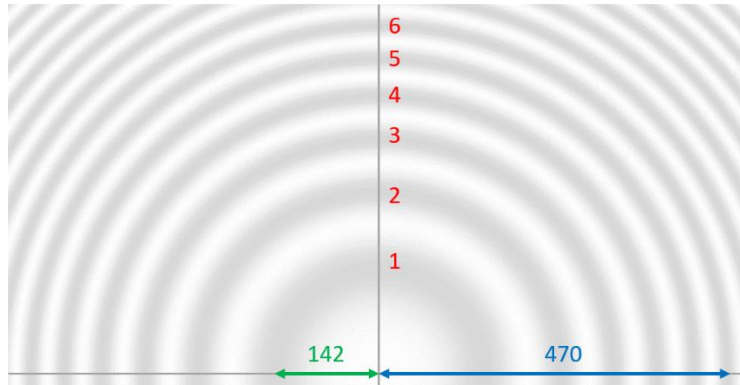


Figure 7: Numbers and radii of rings on a screen

4.3. Newton's rings in reflected light

As the initial prototype of the experiment, we took a demo clip [14], see Figure 8. The virtual scene itself is shown in Figure 9.



Figure 8: A frame of a clip (left). Image on the wall (right)

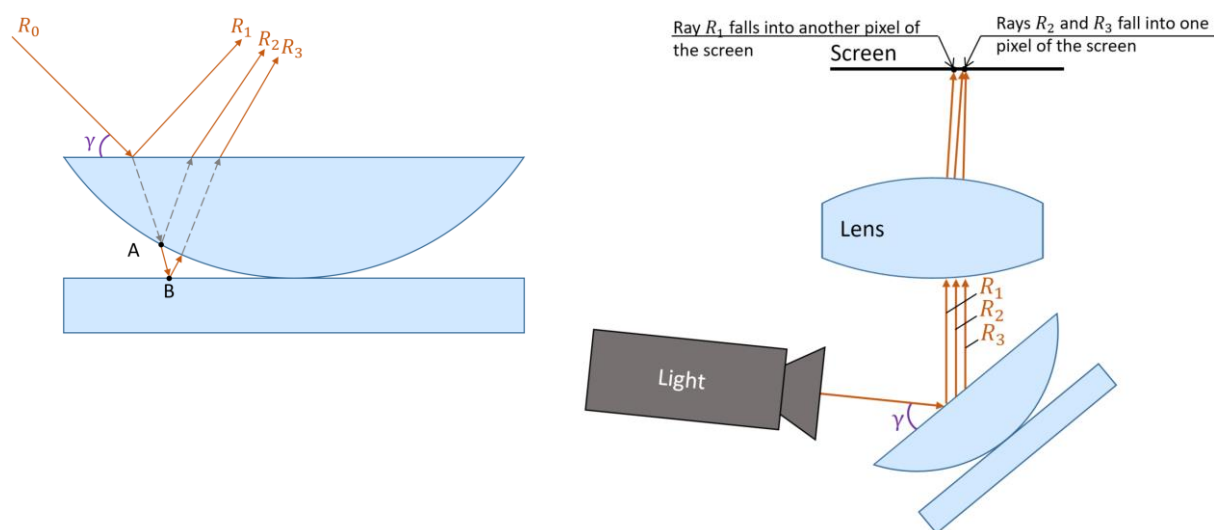


Figure 9: The passage of rays in the system and a setup

Figure 10 shows the results of calculations. The two images on the right contain more rings because there is no bounding shield in the scene.

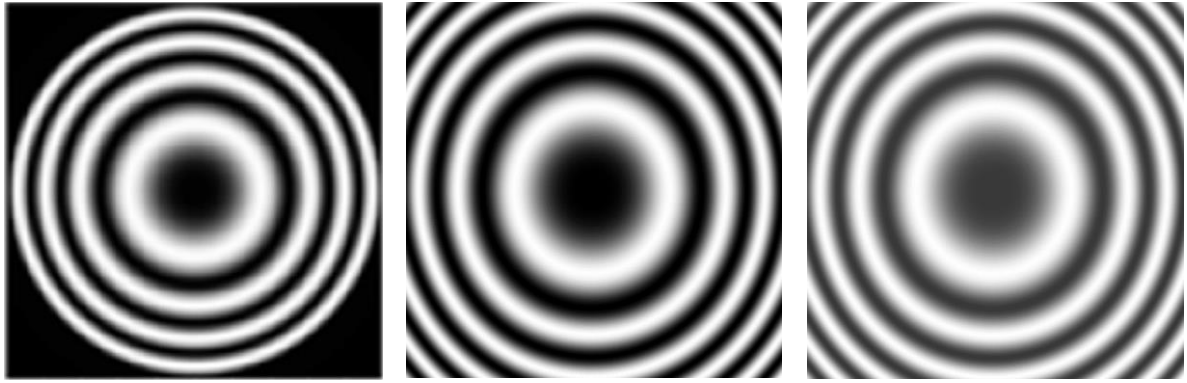


Figure 10: Newton's rings in reflected light. From left to right: image from the work [8]; calculated image taking into account the rays R_1 , R_2 ; calculated image taking into account the rays R_1 , R_2 , and R_3

5. Conclusions

We set the task to calculate images of Newton's rings, which are a fairly representative example for optical phenomena such as fringes of equal thickness. Basic requirement: the calculation must rely on zero-thickness ray tracing. Installations of this type are often called interferometers, they repeatedly use the method of dividing the amplitude of the electric component of the light wave. This corresponds to the division of the intensity associated with the ray. Newton's rings are convenient to use to confirm the visual validity of the algorithm, since there are available photos, see Figures 6 and 8. Sometimes there are additional theoretical tests, when the image can be used to check the values of certain quantities, based on the nature of the object. In our case, this is the determination of the radius of the lens by the numbers and radii of the rings in the image (see Figure 7).

We would like to note that for the first time, images of Newton's rings were calculated based on zero-thickness ray tracing. In general, the problem is solved positively, and the proposed approach can be used to visualize interference effects in 3D scenes. Further study of the problem is needed to clarify the limits of the applicability of the approach.

We have planned the development of the approach to other types of lighting in the scene, including natural light. Then in the future, this approach can serve as a base of modifications of renderers.

6. Acknowledgements

This work was carried out under state contract with ICMMG SB RAS (0251-2021-0001).

7. References

- [1] M. Born, E. Wolf. Principles of optics: Electromagnetic theory of propagation, interference and diffraction of light, Cambridge Univ. Press, 1980.
- [2] V. A. Debelov, L. F. Vasilieva. Visualization of interference pictures of 3D scenes including optically isotropic transparent objects, Scientific visualization, 12.3 (2020) 119–136. doi:10.26583/sv.12.3.11
- [3] M. L. Dias. Ray tracing interference color, IEEE Computer graphics & applications, 11.2 (1991) 54–60.
- [4] A. S. Glassner. Andrew Glassner's notebook soap bubbles: Part 2, IEEE Computer graphics and applications, 20.6 (2000) 99–109.
- [5] J. S. Gondek, G. W. Meyer, J. G. Newman. Wavelength dependent reflection functions, in: Proceedings of Siggraph'94, ACM Press, New York, NY, 1994, pp. 213–219.
- [6] Y. Sun, F. D. Fracchia, T. W. Calvert, M. S. Drew. A spectrally based framework for realistic image synthesis, Visual computer, 17.7 (2001) 429–444.

- [7] Y. Sun, F. D. Fracchia, T. W. Calvert, M. S. Drew. Deriving spectra from colors and rendering light interference, *IEEE Computer graphics and applications*, 4 (1999) 1–8.
- [8] J. E. Harvey, R. G. Irvin, R. N. Pfisterer. Modeling physical optics phenomena by complex ray tracing, *Optical engineering*, 54.3 (2015) 035105. doi:10.1117/1.OE.54.3.035105.
- [9] D. Bloss. An introduction to the methods of optical crystallography, Holt, Rinehartand, and Winston, NY, 1961.
- [10] J. D. Foley et al., *Computer graphics, principles and practice*, 2nd. ed. Addison-Wesley, Reading, Mass., 1996.
- [11] V. A. Debelov, K. G. Kushner, L. F. Vasilyeva. Lens for a computer model of a polarizing Microscope, *Mathematica Montisnigri*, 41 (2018) 151–165.
- [12] G. S. Landsberg, *Optics*, 6th. ed., Fizmatlit, Moscow, 2003. [in Russian]
- [13] Newton's Rings in transmitted monochromatic light, 2021. URL: <https://www.leybold-shop.com/vp5-3-3-1.html>.
- [14] Newton's rings, 2011. URL: <https://www.youtube.com/watch?v=zLzMEt0wJm0>.