

Interactive Rendering of Non-Constant, Refractive Media Using the Ray Equations of Gradient-Index Optics

Chen Cao[†] Zhong Ren^{†‡} Baining Guo[‡] Kun Zhou[†]

[†] State Key Lab of CAD&CG, Zhejiang University [‡] Microsoft Research Asia

Abstract

Existing algorithms can efficiently render refractive objects of constant refractive index. For a medium with a continuously varying index of refraction, most algorithms use the ray equation of geometric optics to compute piecewise-linear approximations of the non-linear rays. By assuming a constant refractive index within each tracing step, these methods often need a large number of small steps to generate satisfactory images. In this paper, we present a new approach for tracing non-constant, refractive media based on the ray equations of gradient-index optics. We show that in a medium of constant index gradient, the ray equation has a closed-form solution, and the intersection point between a ray and the medium boundaries can be efficiently computed using the bisection method. For general non-constant media, we model the refractive index as a piecewise-linear function and render the refraction by tracing the tetrahedron-based representation of the media. Our algorithm can be easily combined with existing rendering algorithms such as photon mapping to generate complex refractive caustics at interactive frame rates. We also derive analytic ray formulations for tracing mirages – a special gradient-index optical phenomenon.

Categories and Subject Descriptors (according to ACM CCS): Computer Graphics [I.3.7]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture

1. Introduction

The refraction of light when it passes through refractive media is an important optical phenomenon for computer graphics. It can generate many beautiful effects, ranging from complex caustics patterns of glass objects to a mirage in the sunset and heat shimmering over a hot road surface, which significantly enhance the realism of computer synthesized images. The rendering of refractive media has thus gained sustained interest from researchers, going back from the conventional off-line ray tracing [Whi80] to recent interactive techniques [SZS*08, WZHB09].

Existing methods can efficiently render refractive objects of constant refractive index using Snell's law at the boundaries of the objects. For non-constant media generated for example by heat gradients, variation of the growth condition of crystals, human engineering, or variation of biological tissue properties, most methods use the ray equation of *geometric optics* to compute piecewise-linear approximations of the non-linear rays. They assume a nearly constant index within each step, and often need to trace the rays in a large number of small steps, leading to high rendering cost.

In this paper, we present a new approach for efficiently



Figure 1: Rendering of non-constant, refractive media at 12 fps. The image resolution is 640×480 . Combined with photon mapping, our algorithm is able to render both the view ray distortion and caustics generated by the dynamic non-constant media.

tracing the non-linear rays in non-constant, refractive media. Based on the observation that most non-constant refractive media seen in everyday life have a smoothly varying index of refraction, we model the index as a piecewise-linear function, and use the ray equations of *gradient-index optics* to outline a general ray tracing algorithm. Compared with pre-

vious methods that use the ray equation of geometric optics, our algorithm can take larger tracing steps and achieves up to an order of magnitude speedups in our experiments. Specifically, we make the following technical contributions:

- An efficient algorithm for tracing refractive objects of constant index gradient. We show that the refracted rays in such objects have an analytic formulation. The algorithm makes use of this formulation to quickly locate the intersection points between the rays and the object boundaries.
- An efficient algorithm for tracing non-constant media of which the refractive index is a piecewise-linear function defined on a tetrahedral mesh. The algorithm can be extended to handle the voxel-based representation of media.
- Analytic ray formulations for tracing mirages, which we show is a special case of gradient-index optics.

We next briefly review related work in Section 2. The ray formulations and tracing algorithms for non-constant media are detailed in Section 3. In Section 4, we present results and show that our algorithms can be combined with photon mapping to render complex caustics and view ray distortions at interactive frame rates.

2. Related work

For objects of constant refractive index, refraction only occurs at the object boundaries. A majority of existing rendering algorithms, such as conventional ray tracing [Whi80], beam tracing [HH84] and photon mapping [Jen96], use Snell’s law to compute the refracted rays. To achieve interactive rendering performance, some algorithms restrict the refraction to happen only on one side of the object, and compute the refracted rays by indexing into a perturbed texture which describes nearby geometries [Oli00] or an environment map [LKM01]. Wyman [Wym05] proposed a two-pass, image-space approach to render refraction of a distant environment through two interfaces, and later extended this approach to render caustics [WN09]. Walter et al. [WZHB09] recently introduced a fast method to find paths that connect points inside and outside a refractive object with triangle mesh boundaries, assuming that paths cross the boundary only once.

Early methods [BTL90] render mirages (i.e., refraction in atmosphere) by refracting the ray according to Snell’s law at each interface of a series of homogeneous layers. Researchers subsequently proposed to use the ray equation of geometric optics [BW99], to trace the nonlinear rays inside a medium of varying refractive index [SL96, Grö95, GMAS05, GMAS06, MGS07]. Munoz [MGS07] analyzed the performance of different numerical schemes for solving the ray equation. The nonlinear ray tracing process has also been greatly accelerated in recent GPU algorithms [WSE04]. Zhao et al. [ZHF*07] proposed a interactive technique for simulating and rendering of heat shimmering and mirages. Ihrke et al. [IZT*07] introduced “eikonal rendering”, a wavefront-based propagation technique for single

scattering rendering. Sun et al. [SZS*08] described a GPU accelerated volume photon mapping technique based on a novel octree representation and an adaptive photon tracing scheme. All these techniques aim to find a piecewise-linear approximation of the nonlinear path. They assume that the refractive index of the medium is constant within each ray/photon tracing step. A relatively large number of steps are often needed for smoothly-varying indices, leading to high rendering cost.

Stam and Langenou [SL96] proposed to model the refractive index as a superposition of a set of weighted kernels, or blobs, and derived an analytical expression for the perturbation of the ray due to the superposition of blobs. This expression, however, is based on a first-order Taylor expansion of the perturbations, which would introduce significant errors when attempting to step over a large distance along the ray. Garg and Mitra [GM02] derived an analytic ray formulation for tracing mirages by assuming that the refractive index varies linearly with the height. According to the study of mirage formation [KTG77], the refractive index is an exponential function of the height. We derive analytic ray formulations for tracing this special non-constant medium.

This paper focuses on single scattering effects in refractive media. The media radiance due to multiple scattering can be computed using non-linear volume photon mapping [GMAS05]. Applying the ray equations introduced in this paper to multiple scattering remains as a future work.

Gradient-index optics is the branch of optics covering optical effects produced by a gradual variation of the refractive index of a material [Mar78]. Tracing rays in gradient-index materials has been extensively studied in the field of optics to help implement these materials into optical systems [Mar70, SKG82, Qia84]. For a constant index gradient, the solution of the ray equation has been proved to be an analytic curve [Qia84].

3. Tracing Non-Constant, Refractive Media

In this section, we first describe the analytic ray formulation and the ray tracing algorithm for refractive objects of constant index gradient – the simplest case of non-constant refractive media. Then we extend the algorithm to handle general non-constant refractive media and mirages.

3.1. Objects with Constant Index Gradient

Consider a closed triangle mesh, and suppose that the refractive index n inside the mesh is given by:

$$n(z) = n(0) + \alpha z,$$

where the z axis is aligned with the direction of the gradient and α is the magnitude of the gradient.

Ray Formulation According to the results in gradient-index optics [Mar78, Qia84], in a constant gradient-index

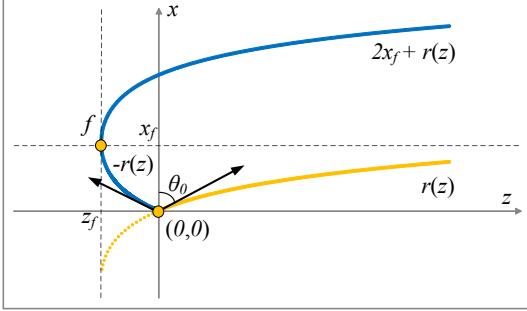


Figure 2: Non-linear rays in constant gradient-index media. Two rays with different incidence angles are shown.

medium, a ray travels in the *ray plane* formed by the gradient direction (i.e., the z axis) and \mathbf{v}_0 , the ray direction when the ray enters the medium. If the angle between \mathbf{v}_0 and the z axis is less than $\pi/2$, the ray path (the orange curve in Figure 2) can be described as an analytical curve on the ray plane:

$$x = r(z) = \frac{n(0) \cos \theta_0}{\alpha} \ln \left(2cz + b + 2\alpha \sqrt{A + bz + cz^2} \right) \Big|_0^z, \quad (1)$$

where the x axis is on the ray plane and perpendicular to the z axis, and θ_0 is the angle formed by \mathbf{v}_0 and the x axis. Other parameters are defined as

$$A = n^2(0) \sin^2 \theta_0, \quad b = 2\alpha n(0), \quad c = \alpha^2.$$

If the angle between \mathbf{v}_0 and the z axis is equal to or greater than $\pi/2$, the ray path (the blue curve in Figure 2) can be divided into two segments at a point $f = (x_f, z_f)$, where

$$z_f = \frac{n(0) \cos \theta_0 - n(0)}{\alpha}, \quad x_f = -r(z_f).$$

The first segment is described by

$$x = -r(z), \quad 0 \leq z \leq z_f, \quad (2)$$

and the second segment is given by

$$x = 2x_f + r(z), \quad z > z_f. \quad (3)$$

As shown in Figure 2, in all cases the ray's tangent gradually becomes parallel to the gradient direction as the ray travels in the medium.

Ray Intersection Test Given the analytic ray representation, the intersection test of the ray with the object's boundaries can be reduced to a 2D problem on the ray plane. A boundary triangle is first intersected with the ray plane, resulting in a line segment, which is then intersected with the ray.

The intersection of the ray with a line segment, however, cannot be analytically calculated due to the transcendental function \ln in the ray formulation. We instead resort to numerical solutions and use the bisection method to compute the intersection. Note that if the triangle size or the magnitude of the gradient is very large, the bisection method

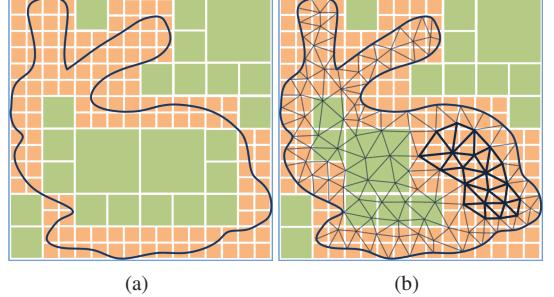


Figure 3: A 2D illustration of the octree construction for a triangle mesh (left) and a tetrahedral mesh (right). The refractive index function varies drastically in the region marked with bold lines, thus the nodes in the finest level there cannot be merged to nodes in the upper octree levels.

may mistakenly omit some intersection points. In practice, we never found this problem for all of our test scenes.

Ray Tracing Ray tracing of refractive objects with constant index gradient is just like conventional ray tracing – rays are generated starting from the eye (or light sources) and traced across the scene. Spatial hierarchies are used to prune intersection tests during ray traversal. Our algorithm uses octrees as the spatial acceleration structure.

Given a tree depth, an octree is first constructed for the boundary triangles of the refractive object (see Figure 3(a)). During ray tracing, when a ray enters the object's bounding box, we compute the intersection point of the ray with the bounding box as the starting point of the ray, and traverse the octree to find the leaf node where the point is located. For this node, we use the aforementioned ray formulation and intersection test method to compute the first intersection point of the ray with the six faces of the node and the triangles contained in the node. Note that for internal nodes that do not contain any triangles, only the six node faces need to be considered. The computed intersection point is set as the new starting point of the ray, and the ray's tangent direction at the point is taken as the new ray direction. The whole process is iterated until the ray travels out of the object.

We adopt the linear octree structure described in [SZS*08]: an octree is represented as a dense 3D array, where each element indicates the tree level of the leaf node that covers the voxel corresponding to the element. This structure is appropriate for fast construction and easy access of the octree on the GPU. The octree depth is determined so that the node size at the finest depth is comparable to the average edge length of the mesh.

3.2. General Non-Constant Refractive Media

We model the refractive index of general non-constant media as a piecewise-linear function defined on a tetrahedral mesh. Each vertex of the tetrahedral mesh is assigned an index value, and the refractive index inside each tetrahedron

is the linear interpolation of the indices of the tetrahedron's four vertices.

Suppose that the refractive index at the four vertices of a tetrahedron is $\{n_k, k = 1, \dots, 4\}$, respectively. It can be easily proven that the index gradient within the tetrahedron is constant and given by

$$\nabla n = \sum_{k=1}^4 \frac{A_k n_k}{T} \mathbf{N}_k, \quad (4)$$

where A_k, \mathbf{N}_k are the area and normal of the face opposite to vertex k , respectively, and T is the volume of the tetrahedron.

The ray formulation and intersection test method described in the preceding section can be directly applied to each tetrahedron, and the tetrahedral mesh can be simply traced one tetrahedron at a time until rays travel out of the mesh.

To fully utilize the smoothness of the index variation, we build an octree for the tetrahedral mesh (see Figure 3(b)). All tetrahedra are first voxelized into the bounding volume of the mesh. Each voxel is associated with a list of tetrahedra that it may intersect. Then, starting from the finest tree level and moving towards the root, we construct the octree one level at a time. At each tree level, for each node and its seven sibling nodes, we merge them into their parent node at the upper tree level if the following conditions are satisfied:

- None of them is associated with a boundary tetrahedron;
- The differences of the gradients of all of their associated tetrahedra are less than a threshold ϵ .

During ray tracing, we traverse the constructed octree to find the current leaf node a ray enters into. If the ray enters a leaf node that is not at the finest depth, the ray will travel along an analytic curve within the node since the index gradient within the node is approximately constant. And we only need to compute the intersection point of the ray with the six faces of the node. If the leaf node is at the finest depth, we need to trace the ray one tetrahedron at a time until the ray leaves the node and enters into other nodes, or travels out of the object.

We use an octree instead of a kd-tree as our acceleration structure due to the relatively lower cost of constructing a octree. This is important for interactive rendering of dynamic media since the acceleration structure needs to be updated at run time.

Voxel-based Representation Our algorithm can be easily adapted to handle the voxel-based representation of non-constant media, which is used by recent interactive techniques [IZT*07, SZS*08].

With the voxel-based representation, the refractive index is defined at each voxel. The index gradient is computed using the central difference. A triangle mesh is used to define the medium boundary.

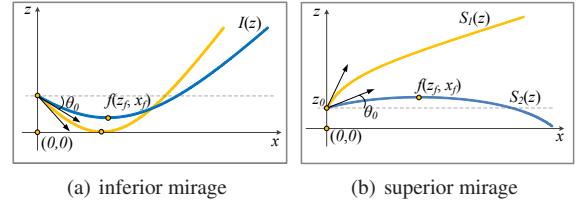


Figure 4: The ray paths of mirages.

We again use octrees as the acceleration structure. The mesh triangles are first rasterized into the voxels, each of which is associated with a list of triangles that the voxel may intersect. The voxels are taken as the finest-level nodes of the octree. Then, similar to what we do for the tetrahedron-based representation, a bottom-up process is carried out to merge small nodes into large nodes if two conditions are satisfied: 1) none of the nodes to be merged is associated with any triangles; 2) the differences of the gradients of the nodes to be merged is less than a threshold ϵ , or the differences of the indices of the nodes is less than a threshold δ .

The tracing algorithm is simple. When a ray enters a node associated with boundary triangles, it is first intersected with the triangles associated with the node to test if it travels out of the object. In other cases, the ray is only intersected with the six faces of the node.

3.3. Mirages

Mirage formation is known to be the result of the variation of refractive index of air mainly due to temperature gradients. According to the study by Khular et al. [KTG77], for an inferior mirage which occurs when the air above the line of sight is colder than that below, the refractive index is the following function of the height:

$$n^2(z) = \mu_0^2 + \mu_1^2 (1 - \exp(-\beta z)), \quad (5)$$

where z represents the height above the ground, $\mu_0 = 1.000233$ is the refractive index at the earth surface, and $\mu_1 = 0.4584$, $\beta = 2.303$ are constants.

This model assumes constant refractive index at a given altitude, and the direction of the index gradient is perpendicular to the earth surface. For a small region of the earth surface, which is approximately planar, the gradient direction can be assumed to be constant. According to the derivation in Appendix A, we can get the ray formulation:

$$x = I(z) = \frac{n(z_0) \cos \theta_0}{\beta \sqrt{B}} \ln \left(\frac{\sqrt{B} + \sqrt{B - C \exp(-\beta z)}}{\sqrt{B} - \sqrt{B - C \exp(-\beta z)}} \right) \Big|_{z_0}^z, \quad (6)$$

where

$$B = \mu_0^2 + \mu_1^2 - (n(z_0) \cos \theta_0)^2, \quad C = \mu_1^2.$$

Note that the origin of the ray plane is set to the projection of the ray origin on the earth surface, and z_0 is the height of the ray origin, as shown in the Figure 4(a).

For superior mirage the refractive index is given by:

$$n^2(z) = \mu_0^2 + \mu_1^2(\exp(-\beta z)). \quad (7)$$

The ray formulation can be derived as

$$x = S_1(z) = \frac{n(z_0) \cos \theta_0}{\beta \sqrt{D}} \ln \left(\frac{\sqrt{D + C \exp(-\beta z)} + \sqrt{D}}{\sqrt{D + C \exp(-\beta z)} - \sqrt{D}} \right) \Big|_{z_0}^z \quad (8)$$

for $\cos \theta_0 \leq \sqrt{\frac{\mu_0^2}{\mu_0^2 + \mu_1^2 \exp(-\beta z_0)}}$ (see the orange curve in Figure 4(b)), and

$$x = S_2(z) = \frac{-2n(z_0) \cos \theta_0}{\beta \sqrt{-D}} \arctan \left(\sqrt{-\frac{C}{D} \exp(-\beta z) - 1} \right) \Big|_{z_0}^z \quad (9)$$

for $\cos \theta_0 > \sqrt{\frac{\mu_0^2}{\mu_0^2 + \mu_1^2 \exp(-\beta z_0)}}$ (see the blue curve in Figure 4(b)), where

$$D = \mu_0^2 - (n(z_0) \cos \theta_0)^2.$$

The above ray equations assume the angle between v_0 and the gradient direction is less than $\pi/2$. If the angle is equal to or larger than $\pi/2$, the ray equations can be divided into two segments, as in the constant index gradient case. And the fold point is determined by solving $n(z_f) = n(z_0) \cos \theta_0$ for the corresponding type of mirage, which is:

$$z_f = -\frac{1}{\beta} \ln \left(1 - \frac{n^2(z_0) \cos^2 \theta_0 - \mu_0^2}{\mu_1^2} \right), \quad x_f = I(z_f).$$

for inferior mirages, and

$$z_f = -\frac{1}{\beta} \ln \left(\frac{n^2(z_0) \cos^2 \theta_0 - \mu_0^2}{\mu_1^2} \right), \quad x_f = S_2(z_f).$$

for superior mirages.

Musgrave [Mus90] pointed out that mirages are strong because of total internal reflection due to Brewster's critical angle of incidence. We analytically give the fold point, the exact point that the total internal reflection happens. The ray paths and fold points of mirages are plotted in Figure 4.

In rendering mirages, we assume the earth surface is represented by a set of quads within the range of sight. The quads are extruded along the lines connecting the earth center and the quad vertices to generate a set of frustums, which form the atmosphere volume. Within each frustum, the refractive index is described by equation 5 or equation 7, taking the normal of the quad as the gradient direction. Therefore, as long as we know the origin and direction of the ray when it enters a frustum, we can obtain the ray equations.

Tracing rays in frustums is as simple as that in tetrahedra. Based on the analytic formulation, the ray is intersected with the boundary faces of the current frustum to calculate the new ray origin and direction. This process continues until the ray shoots out of the atmosphere volume, or hits an object. The number of frustums used is typically less than ten. Acceleration structures are thus unnecessary.

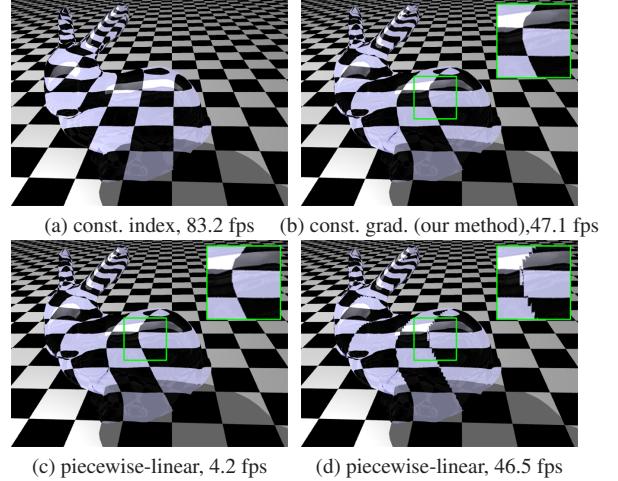


Figure 5: Comparison of our method (b) with the piecewise-linear approximation method with comparable quality (c) and similar performance (d). The constant refractive index result (a) is shown to reveal the influence of index gradient.

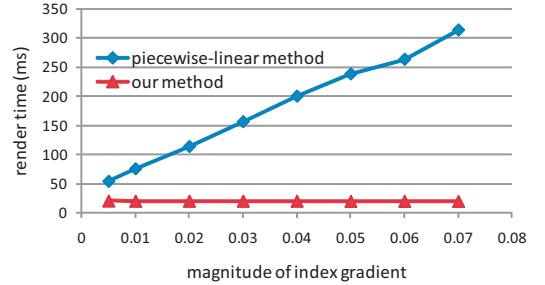


Figure 6: Render time for different magnitudes of index gradient.

4. Experimental Results

We have implemented the described algorithms on an NVIDIA GTX 285 graphics card using CUDA. Our GPU ray tracing implementation follows the per-ray persistent while-while traversal algorithm [AL09]. All images are generated at 640×480 .

Figure 5 shows the rendering results of a Bunny model of constant index gradient. Note that the image rendered by our algorithm (Figure 5(b)) is the ground truth as our algorithm is based on the precise, analytic ray formulation and does not contain any approximation. To compare our algorithm with the piecewise-linear approximation method, we carefully tuned the tracing step size to generate an image of comparable quality (Figure 5(c)). In this case, our algorithm is 10 times faster. The render time of the piecewise-linear method also increases with the magnitude of the gradient since it needs smaller step sizes to achieve accurate approximation. In contrast, our algorithm is insensitive to the gradient magnitude, as shown in Figure 6. Increasing the tracing step size in the piecewise-linear method can improve the ren-



Figure 7: A tetrahedral mesh (5,442 tetrahedra) with spatially-variant refractive indices is rendered at 25 fps.

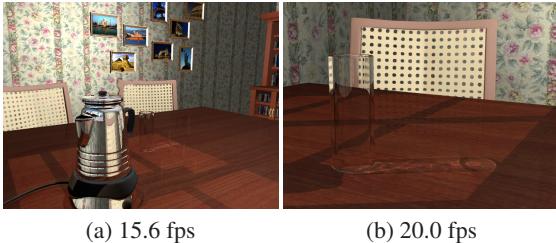


Figure 8: Rendering of heat shimmering with surface caustics. The effect is more obvious in the accompanying video.

dering performance but lead to severe artifacts, as shown in Figure 5(d).

Figure 7 shows the rendering results of a tetrahedral mesh with spatially-variant refractive index. The indices are obtained by sampling the luminance of two synthesized solid textures [KFCO*07] at the tetrahedron vertices. The 2D exemplar textures are shown as insets.

The number of tetrahedra required for plausible rendering results mainly depends on the spatial variation of the refractive index – more tetrahedra are required to faithfully represent significantly varying index functions. As demonstrated in Figure 7 and the accompanying video, with several thousands tetrahedra, our algorithm is able to reveal the interesting patterns in the synthesized textures via the distortion of view rays. The ray tracing time scales with the number of tetrahedra, K , roughly at the rate of $\sqrt[3]{K}$. This is because the average number of traversed tetrahedra of a ray is approximately proportional to $\sqrt[3]{K}$.

Figure 1 shows the rendering of a dynamic refractive medium represented by $128 \times 128 \times 128$ voxels. Our ray tracer is combined with the GPU-based photon mapping algorithm [ZHWG08] to render surface caustics at interactive frame rates. We carefully tuned the two thresholds when constructing octrees to guarantee the rendering quality. In this scene, $\epsilon = 0.005$ and $\delta = 0.01$ can give satisfactory results. Note that Sun et al. [SZS*08] also construct an octree describing the regions of space in which the refractive index is nearly constant, i.e., the gradient is nearly zero. Our algorithm merges nodes with similar gradients in addition to that with similar refractive indices. This typically leads to a coarser octree with larger nodes and superior performance. In this scene, we can get up to $3\times$ speedups over their approach.

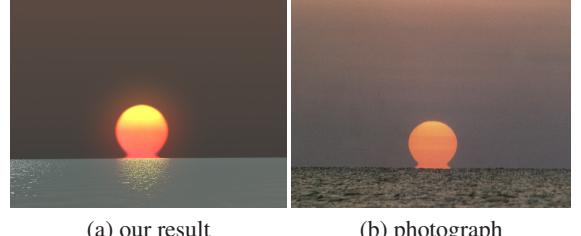


Figure 9: Rendering of a mirage. Our rendering result convincingly captures the so-called “Omega” sunset effect and is directly compared with a photograph taken at sunset.

Grid Size	32^3	64^3	128^3	256^3
const. grad. index (Figure 5(b))	0.63	4.1	29.6	224.1
voxel-based rep. (Figure 1, 8)	0.66	5.2	41.9	335.5
tetrahedra rep. (Figure 7)	1.1	5.2	33.0	232.7

Table 1: Memory consumption (in megabytes) for the octree voxelization under different grid sizes, including the linear octree structure and the array of the overlapped triangles/tetrahedra in each octree node. The latter array is not needed for the voxel-based representation (Figure 1, 8).

In Figure 8 and the accompanying video, we show the interactive rendering of heat shimmering. The shimmering is simulated by Perlin noise [Per02]. Combined with photon mapping, our algorithm is able to render both the view ray distortion and dynamic caustics generated by heat shimmering. This effect has not been observed in previous interactive applications. Note that the memory consumption for our octree voxelization is comparable with that of [SZS*08], as reported in Table 1.

Finally, Figure 9 shows the mirage image rendered by our algorithm, compared to a real photograph. We achieve very convincing results at real-time frame rates.

5. Conclusion

We have presented a new ray tracing approach for non-constant refractive media based on the ray equations of gradient-index optics. To our knowledge, these equations have not appeared before in the computer graphics literature. Our algorithm models the refractive index as piecewise-linear functions, traces rays along analytically-represented curves, and is thus more efficient than previous piecewise-linear approximation methods. We also derive analytic ray formulations for tracing mirages – a special gradient-index optical phenomenon in atmosphere. We believe our work will stimulate future research into the application of gradient-index optics in computer graphics.

One limitation of our method is the discontinuity of gradients at primitive boundaries – while the refractive index is guaranteed to be smooth in our piecewise-linear representation, the gradient may be discontinuous at the tetrahedron or voxel boundaries. This sometimes leads to visual artifacts

when the mesh is very coarse. The problem may be alleviated by adapting the mesh to the index distribution or developing higher order representations of the index function.

Acknowledgements We would like to thank the anonymous reviewers for their helpful comments. This research was partially funded by the NSF of China (No. 60825201).

References

- [AL09] AILA T., LAINE S.: Understanding the efficiency of ray traversal on gpus. In *HPG '09* (2009), pp. 145–149.
- [BTL90] BERGER M., TROUT T., LEVIT N.: Ray tracing mirages. *IEEE CG&A* 10, 3 (1990), 36–41.
- [BW99] BORN M., WOLF E.: *Principles of Optics: Electromagnetic Theory of Propagation, Interference and Diffraction of Light*. Cambridge University Press, 1999.
- [GM02] GARG G., MITRA N. J.: Nonlinear raytracing: Mirages, 2002. http://graphics.stanford.edu/courses/cs348b-competition/cs348b-02/mitra_garg.
- [GMAS05] GUTIERREZ D., MUÑOZ A., ANSON O., SERON F. J.: Non-linear Volume Photon Mapping. In *EGSR'05* (2005), pp. 291–300.
- [GMAS06] GUTIERREZ D., MUÑOZ A., ANSON O., SERON F. J.: Simulation of atmospheric phenomena. *Computer & Graphics* 20, 6 (2006), 994–1010.
- [Grö95] GRÖLLER E.: Nonlinear ray tracing: visualizing strange worlds. *The Visual Computer* 11, 5 (1995), 263–274.
- [HH84] HECKBERT P. S., HANRAHAN P.: Beam tracing polygonal objects. In *Proceedings of SIGGRAPH'84* (1984), pp. 119–127.
- [IHZ*07] IHRKE I., ZIEGLER G., TEVS A., THEOBALT C., MAGNOR M., SEIDEL H.-P.: Eikonal rendering: efficient light transport in refractive objects. *ACM Trans. Graph.* 26, 3 (2007), 59.
- [Jen96] JENSEN H. W.: Global illumination using photon maps. In *Rendering Workshop* (1996), pp. 21–30.
- [KFCO*07] KOPF J., FU C.-W., COHEN-OR D., DEUSSEN O., LISCHINSKI D., WONG T.-T.: Solid texture synthesis from 2d exemplars. *ACM Trans. Graph.* 26, 3 (2007), 2.
- [KTG77] KHULAR E., THYAGARAJAN K., GHATAK A. K.: A note on mirage formation. *American Journal of Physics* 45 (1977), 90–92.
- [LKM01] LINDHOLM E., KLIGARD M. J., MORETON H.: A user-programmable vertex engine. In *Proceedings of SIGGRAPH'01* (2001), pp. 149–158.
- [Mar70] MARCHAND E. W.: Ray tracing in gradient-index media. *Journal of the Optical Society of America* 60, 1 (1970), 1–7.
- [Mar78] MARCHAND E. W.: *Gradient Index Optics*. Academic Press, 1978.
- [MGS07] MUÑOZ A., GUTIERREZ D., SERÓN F. J.: Optimization techniques for curved path computing. *Vis. Comput.* 23, 7 (2007), 493–502.
- [Mus90] MUSGRAVE F. K.: A note on ray tracing mirages. *IEEE Comput. Graph. Appl.* 10, 6 (1990), 10–12.
- [Oli00] OLIVEIRA G.: Refractive texture mapping, part two, 2000. http://www.gamasutra.com/features/20001117/oliveira_01.htm.
- [Per02] PERLIN K.: Improving noise. In *Proc. of ACM SIGGRAPH* (2002), pp. 681–682.
- [Qia84] QIAO Y.: The solution of the ray-path differential equation in the axial gradient-index media. *Acta Optica Sinica* 4, 1 (1984), 89–92.
- [SKG82] SHARMA A., KUMAR D. V., GHATAK A. K.: Tracing rays through graded-index media: a new method. *Applied Optics* 21, 6 (1982), 984–987.
- [SL96] STAM J., LANGUENOU E.: Ray tracing in non-constant media. In *Rendering Workshop* (1996), pp. 225–234.
- [SZS*08] SUN X., ZHOU K., STOLLNITZ E., SHI J., GUO B.: Interactive relighting of dynamic refractive objects. *ACM Trans. Graph.* 27, 3 (2008), 35:1–35:9.
- [Whi80] WHITTED T.: An improved illumination model for shaded display. *Commun. ACM* 23, 6 (1980), 343–349.
- [WN09] WYMAN C., NICHOLS G.: Adaptive caustic maps using deferred shading. *Comput. Graph. Forum* 28, 2 (2009), 309–318.
- [WSE04] WEISKOPF D., SCHAFHITZEL T., ERTL T.: Gpu-based nonlinear ray tracing. *Comput. Graph. Forum* 23, 3 (2004), 625–633.
- [Wym05] WYMAN C.: An approximate image-space approach for interactive refraction. *ACM Trans. Graph.* 24, 3 (2005), 1050–1053.
- [WZHB09] WALTER B., ZHAO S., HOLZSCHUCH N., BALA K.: Single scattering in refractive media with triangle mesh boundaries. *ACM Trans. Graph.* 28, 3 (2009), 92:1–92:8.
- [ZHF*07] ZHAO Y., HAN Y., FAN Z., QIU F., CHUAN KUO Y., KAUFMAN A., MUELLER K.: Visual simulation of heat shimmering and mirage. *IEEE TVCG* 13 (2007), 179–189.
- [ZHWG08] ZHOU K., HOU Q., WANG R., GUO B.: Real-time kd-tree construction on graphics hardware. *ACM Trans. Graph.* 27, 5 (2008), 126.

Appendix A: Derivation of Ray Equations

We start the derivation from the well-known vector differential equation for the ray trajectories [BW99]:

$$\frac{d}{ds} \left(n \frac{d\mathbf{r}}{ds} \right) = \nabla n, \quad (10)$$

where n is the refractive index function, \mathbf{r} is the variable-position vector on the ray, and s is the arc length.

For now, let us assume that the direction of the index gradient of the media is constant, which is true for both the constant index gradient and mirage cases. And let us also take the z axis as the direction of the gradient. Equation 10 can be written with respect to the three components of \mathbf{r} :

$$\frac{d}{ds} \left(n \frac{dx}{ds} \right) = 0, \quad \frac{d}{ds} \left(n \frac{dy}{ds} \right) = 0, \quad \frac{d}{ds} \left(n \frac{dz}{ds} \right) = \frac{\partial n}{\partial z}, \quad (11)$$

and the direction cosines can be defined:

$$p_0 = n \frac{dx}{ds}, \quad q_0 = n \frac{dy}{ds}, \quad l = n \frac{dz}{ds}. \quad (12)$$

Equation 11 implies that the direction cosines along the x and y axes are constant along the ray, thus the notation p_0 and q_0 are used for these constants. The direction cosines and the refractive index are related by:

$$p_0^2 + q_0^2 + l^2 = n^2 \left(\left(\frac{dx}{ds} \right)^2 + \left(\frac{dy}{ds} \right)^2 + \left(\frac{dz}{ds} \right)^2 \right) = n^2. \quad (13)$$

Suppose the position of the ray origin is given by $Q_0 = (x_0, y_0, z_0)$, the position of any point $Q = (x, y, z)$ on the ray can be obtained by integration:

$$\begin{aligned} x &= x_0 + \int_{Q_0}^Q \frac{dx}{ds} ds = x_0 + p_0 \int_{Q_0}^Q \frac{ds}{n} = x_0 + p_0 \int_{z_0}^z \frac{dz}{l}, \\ y &= y_0 + \int_{Q_0}^Q \frac{dy}{ds} ds = y_0 + q_0 \int_{Q_0}^Q \frac{ds}{n} = y_0 + q_0 \int_{z_0}^z \frac{dz}{l}. \end{aligned} \quad (14)$$

Let us take the x axis as the orthogonalization of the initial ray direction with respect to z , thus we have $q_0 = 0$ and $y = 0$ for all points on the ray.

In the following, we derive the closed-form of the ray equation for the constant index gradient and mirage cases on the *ray plane* formed by the initial ray direction \mathbf{v}_0 and the gradient direction.

Constant Gradient-Index Media

In constant gradient-index media, the refractive index is given by:

$$n(z) = n(0) + \alpha z \quad (15)$$

where α is the magnitude of the gradient. Without loss of generality we put the origin of the ray plane at the origin of the ray, so that $x_0 = z_0 = 0$ in equation 14 and $n(0)$ is the refractive index at the ray origin.

Substituting equation 15 into equation 13 and assuming the angle between \mathbf{v}_0 and the z axis is less than $\pi/2$ so that $l = n \frac{dz}{ds} > 0$, we get

$$l = \sqrt{A + bz + cz^2}, \quad (16)$$

where

$$A = n^2(0) \sin^2 \theta_0, \quad b = 2\alpha n(0), \quad c = \alpha^2, \quad (17)$$

and θ_0 is the angle formed by \mathbf{v}_0 and the x axis. Note that $p_0 = n(0) \frac{dx}{ds} = n(0) \cos \theta_0$.

Substituting equation 16 into equation 14, we can obtain the closed-form of the ray equation [Qia84]:

$$x = r(z) = \frac{n(0) \cos \theta_0}{\alpha} \ln \left(2cz + b + 2\alpha \sqrt{A + bz + cz^2} \right) \Big|_0^z \quad (18)$$

If the angle between \mathbf{v}_0 and the z axis is greater than $\pi/2$, a case that is not discussed in [Qia84], $l = n \frac{dz}{ds}$ starts as negative value, thus $l = -\sqrt{A + bz + cz^2}$ and $x = -r(z)$. This equation holds until $l = 0$, which corresponds to the fold point $f = (x_f, z_f)$. After the ray passes the fold point, l becomes positive and $x = 2x_f + r(z)$. The shift of $2x_f$ is due to substituting (x_f, z_f) into equation 14 as the ray origin.

Mirages

Below we derive the ray equations for two kinds of mirages, the inferior mirage and the superior mirage, respectively. Note that we only derive the ray equations for the $l > 0$ case,

and the ray equations for the $l \leq 0$ case can be obtained similarly as above.

Inferior Mirage occurs when the air above the line of sight is colder than that below. Khular et al. [KTG77] gave an empirical representation of the refractive index:

$$n^2(z) = \mu_0^2 + \mu_1^2 (1 - \exp(-\beta z)), \quad (19)$$

where z represents the height above the ground, $\mu_0 = 1.000233$ is the refractive index at the earth surface, and $\mu_1 = 0.4584$, $\beta = 2.303$ are constants. We set the origin of the ray plane to the projection of the ray origin on the earth surface, so that $x_0 = 0$ and z_0 is the height of the ray origin.

Substituting equation 19 to equation 13, we get

$$l = \sqrt{\mu_0^2 + \mu_1^2 (1 - \exp(-\beta z)) - p_0^2}, \quad (20)$$

Substituting equation 20 into equation 14, we get:

$$x = \frac{n(z_0) \cos \theta_0}{\beta \sqrt{B}} \ln \left(\frac{\sqrt{B} + \sqrt{B - C \exp(-\beta z)}}{\sqrt{B} - \sqrt{B - C \exp(-\beta z)}} \right) \Big|_{z_0}^z, \quad (21)$$

where

$$B = \mu_0^2 + \mu_1^2 - (n(z_0) \cos \theta_0)^2, \quad C = \mu_1^2. \quad (22)$$

Superior Mirage occurs when the air below the line of sight is colder than that above, and the refractive index is given by [KTG77]:

$$n^2(z) = \mu_0^2 + \mu_1^2 (\exp(-\beta z)). \quad (23)$$

Substituting equation 23 to equation 13, we get

$$l = \sqrt{(\mu_0^2 - p_0^2) + \mu_1^2 (\exp(-\beta z))}. \quad (24)$$

The form of the analytical integration obtained by substituting equation 24 into equation 14 is dependant on the sign of $(\mu_0^2 - p_0^2)$, for the positive case or $\cos \theta_0 \leq \sqrt{\frac{\mu_0^2}{\mu_0^2 + \mu_1^2 \exp(-\beta z_0)}}$:

$$x = \frac{n(z_0) \cos \theta_0}{\beta \sqrt{D}} \ln \left(\frac{\sqrt{D + C \exp(-\beta z)} + \sqrt{D}}{\sqrt{D + C \exp(-\beta z)} - \sqrt{D}} \right) \Big|_{z_0}^z, \quad (25)$$

and for the negative case or $\cos \theta_0 > \sqrt{\frac{\mu_0^2}{\mu_0^2 + \mu_1^2 \exp(-\beta z_0)}}$:

$$x = \frac{-2n(z_0) \cos \theta_0}{\beta \sqrt{-D}} \arctan \left(\sqrt{-\frac{C}{D} \exp(-\beta z) - 1} \right) \Big|_{z_0}^z, \quad (26)$$

where

$$D = \mu_0^2 - (n(z_0) \cos \theta_0)^2. \quad (27)$$