

# Focus Range: Production Ray Tracing of Depth of Field

Edoardo A. Dominici Animal Logic Sydney, Australia Huawei Technologies Zurich, Switzerland edoaramis@gmail.com	Emanuel Schrade Animal Logic Sydney, Australia emanuels@al.com.au	Basile Fraboni Animal Logic Vancouver, Canada basilef@al.com.au	Luke Emrose Animal Logic Sydney, Australia lukee@al.com.au	Curtis Black Animal Logic Sydney, Australia curtisb@al.com.au
---	--	--	---	--



*Focus range* enables artists to define an in-focus depth range to visually emphasize key characters. The amount of blur in front and behind the range can be defined independently, allowing for greater artistic control (a, b). We trace three piece-wise linear ray segments which converge and diverge along the viewing direction. In contrast, in a thin-lens model the artists would have to choose either a large aperture with the desired defocus blur with some characters being out of focus (c), or a small aperture to ensure the characters are in focus which results in insufficient background blur (d). Our method can satisfy both constraints simultaneously (b). Image copyright © 2023 Animal Logic Pty Ltd. All Rights Reserved.

## ABSTRACT

Defocus blur adds realism to computer generated images by modelling the restricted focus range of physical camera lenses. It allows artists to draw the viewer’s attention to in-focus regions of the image by separating them from the background. The widely used thin lens model provides a simple approach to achieve this effect, allowing artists to control the frame’s appearance through the distance to the focus plane and the aperture size. Our proposed *focus range* model extends the focus distance to a range. This allows artists to keep characters fully in focus and independently define the out-of-focus blur. We demonstrate how to achieve this robustly in the presence of specular BSDFs, ray-oriented geometry and multiple light bounces. Additionally, we share our practical experience of integrating this model into our production renderer *Glimpse*.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

*SA Technical Communications ’23*, December 12–15, 2023, Sydney, NSW, Australia  
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 979-8-4007-0314-0/23/12...\$15.00  
<https://doi.org/10.1145/3610543.3626156>

## CCS CONCEPTS

- Computing methodologies → Ray tracing.

## KEYWORDS

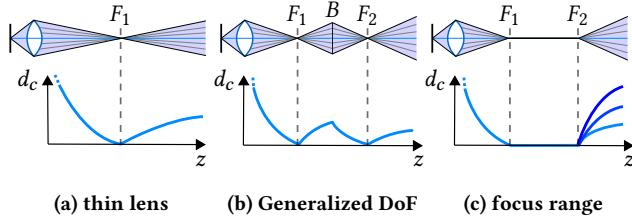
ray tracing, defocus blur, depth of field, thin lens

## ACM Reference Format:

Edoardo A. Dominici, Emanuel Schrade, Basile Fraboni, Luke Emrose, and Curtis Black. 2023. Focus Range: Production Ray Tracing of Depth of Field. In *SIGGRAPH Asia 2023 Technical Communications (SA Technical Communications ’23)*, December 12–15, 2023, Sydney, NSW, Australia. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3610543.3626156>

## 1 INTRODUCTION AND RELATED WORK

Defocus blur is the result of a point in the scene being imaged as a circle of confusion on the film plane. A portion of the scene is considered *in focus* if the circle of confusion is less than a pixel in size [Pharr et al. 2016] and the in focus depth range is called *depth of field*. The widely adopted thin lens model [Cook et al. 1984; Kolb et al. 1995] generates defocus by connecting points sampled on the aperture to the pixel position on a single focus plane located a specific *focus distance* away from the camera (see Figure 2a). Note that by construction, the extent of the depth of field depends directly on the aperture size and the focus distance is constant over the



**Figure 2: In the thin lens model (a) all lens samples for a sensor point converge in a single point on the focus plane at a focus distance  $F_1$ . Kosloff and Barsky [2009] split the camera ray into piecewise defined linear segments (b) to add another focus point along the ray direction. A parameter  $B$  controls the amount of defocus between focus points  $F_1$  and  $F_2$ . Our proposed focus range model (c) exposes the extent of the in focus part and a scaling factor for the background defocus to the artists. Image copyright © 2023 Animal Logic Pty Ltd. All Rights Reserved.**

whole image. Variations of the thin lens model have been proposed to allow for greater artistic control: Pixar achieved an effect known as *split diopter*, where a part of the image has a different focus distance than the rest, by allowing the focus distance to vary over the image plane [Pixar 2019]. Kosloff and Barsky [2009] add a second focus point along the ray direction by breaking up the camera ray into two linear segments such that the ray bundle converges in both focus points (see Figure 2b). Similar to Kosloff and Barsky, our key observation is that primary rays can be defined piece-wise. In fact, with  $B = 0$  their model would be similar to our *focus range* model. However, they merely mention the idea of having several focus points without giving details how this could be used in practice or discussing any special cases like  $B = 0$ . While these variations add more control over the in focus regions of the image and are certainly useful in their respective use-cases, our goal was to achieve large out of focus blur but keep a defined depth range completely in focus. To this end, we break up the rays into three linear segments (see Figure 2c): (i) the *foreground*, where rays converge towards the focus range, (ii) the *in-focus* part, (iii) the *background* where rays diverge again. To mitigate visual errors in the form of discontinuities in the circle of confusion on objects intersecting the focus planes, we use a smoothed shading ray for shading evaluation, similar to Chapiro et al. [2015], who use this approach to shade stereoscopic images with different shading rays for each eye.

In this paper we describe the construction of the three ray segments and discuss the following special cases that unavoidably occur in a production environment to ensure expected behaviour:

- glossy objects intersecting the focus planes
- refractive surfaces along the ray path
- ray-oriented geometry

We hope our practical experience will also inspire further development of user-controllable defocus effects based on ray-tracing.

## 2 THE FOCUS RANGE MODEL

We propose the *focus range* model, which extends the thin lens model by splitting the depth range into three intervals (see Figure 2c): *foreground* ( $z < F_1$ ), *in-focus* ( $F_1 \leq z < F_2$ ), and *background* ( $z \geq F_2$ ). This allows more control of depth of field by introducing a controllable range of perfect focus. Starting from the camera, the first ray segment is constructed like in the thin lens model by sampling a point  $l$  on the lens with diameter  $d_l$  and connecting it to the point on the focus plane  $F_1$  for the respective pixel, the position of which is determined by the focal length  $f$  or equivalently the field of view. This gives us the ray direction  $w_0$  for the first interval. We intersect the scene geometry with this ray segment, ignoring intersections behind the  $F_1$  plane. If no intersection is found in one segment, we update the ray direction for the next segment and test for intersections with the scene geometry. The direction change at interval transitions can be described as a shear transformation, which will prove useful in the following sections. Assuming the camera to be located at the origin and looking along the  $z$  axis, the updated ray direction  $w_{i+1}$  can be calculated from the previous normalized direction  $w_i$  as follows:

$$w_{i+1} = T(w_i, t) = \text{normalize} \left( \frac{w_i}{w_{i,z}} + tl \frac{S_i}{F_1} \right), \quad (1)$$

with shearing factors  $S_i$ , where  $S_0 = 1$  and  $S_1 \leq 0$  can be varied to choose the amount of background defocus blur. We also introduced a scalar  $t \in [0, 1]$  that we use for the interpolated shading ray, as we shall see in subsection 2.1. At  $F_1$  this shear maps all incoming rays onto the same line while at  $F_2$  rays are spread out again by taking the original lens sample offset into account. For an intersected scene point where  $z$  is its camera space depth, the circle of confusion is:

$$d_c = \begin{cases} \frac{df(F_1-z)}{z(f+F_1)} & z < F_1 \\ 0 & F_1 \leq z < F_2 \\ S_1 \frac{df(F_2-z)}{z(f+F_2)} & z \geq F_2 \end{cases} \quad (2)$$

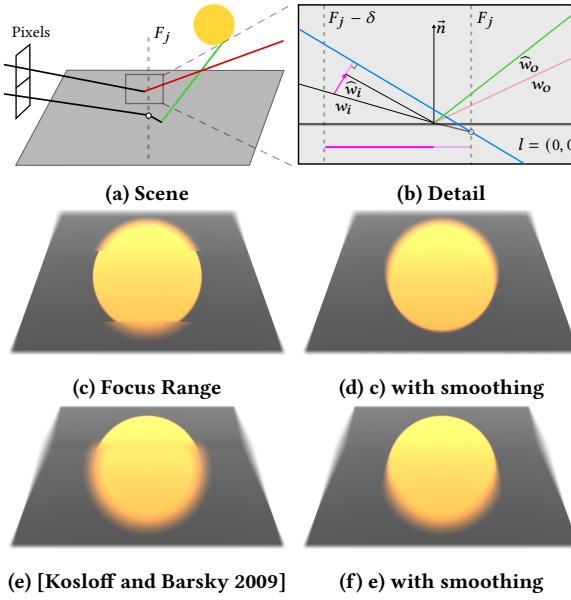
In the following sections, we describe the potential artifacts encountered when using this model, and the solutions we have implemented to address them.

### 2.1 Shading continuity

Glossy objects that intersect the focus planes are problematic, as the abrupt change in ray direction causes discontinuities in shading (cf. Figure 3). To mitigate this issue, we calculate an interpolated shading ray in a *smoothing* region  $\delta$  before an interval boundary. Although this parameter may depend on the scene scale and thus requires artist control, we found a value of  $\delta = 0.5$  scene units suitably addressed most cases. It is important to ensure that  $\delta$  is never greater than one of the intervals to avoid further visual errors:  $\delta = \min\{\delta, F_1, F_2 - F_1\}$ . The interpolated shading ray  $\hat{w}_i$  is computed using a partial shear of the ray direction  $w_i$ :

$$\hat{w}_i = T \left( w_i, \max \left( 0, 1 - \frac{F_{i+1} - z}{\delta} \right) \right), \quad (3)$$

where  $T$  is the transformation from Equation 1. Note that the smoothing is applied one-sided when approaching the interval boundary, such that shading rays have the expected direction at the focus planes  $F_j$ . The shading ray is then used for BSDF sampling

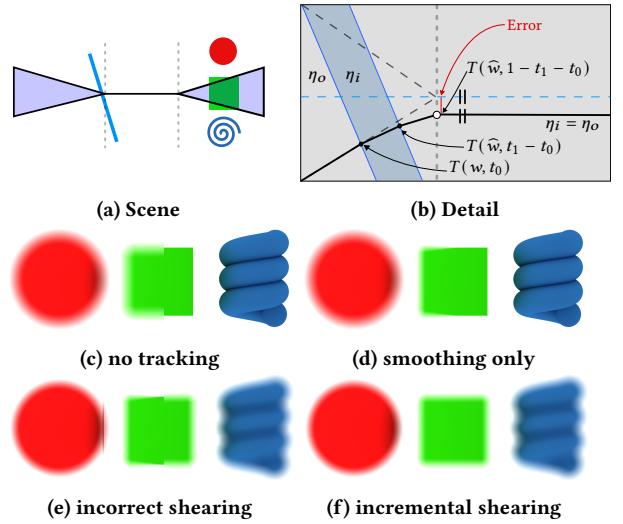


**Figure 3:** Shading artifacts on glossy objects intersecting the focus plane can be mitigated through smooth shading rays. (a) Rays for neighboring pixels can be reflected differently, as their direction changes at focus planes. (b) We use interpolated shading rays for material evaluations close to a focus plane. This improves transitions for both (c+d) focus range and (e+f) the model of [Kosloff and Barsky 2009]. Image copyright © 2023 Animal Logic Pty Ltd. All Rights Reserved.

and all quantities calculated in the context of next event estimation with multiple importance sampling.

## 2.2 Refractive objects

For diffuse geometry it may suffice to trace the three ray intervals of the focus range up to the first intersection. However, this leads to unexpected defocus effects for refractive objects such as glass. Consider, for example, a camera looking through a tilted glass sheet, as shown in Figure 4a. Choosing a refractive index of  $\eta_i = 1$ , we wouldn't expect the glass to have any influence on the rendered image. Note that the objects are in the third interval at an equal distance to the camera, so they should be similarly defocused. Figure 4c shows the result without special treatment for refractive objects, i.e. only tracing the focus range rays up to the first intersection. For the left half of the image, primary rays intersect the glass in the first segment and the resulting render is blurry, while the right half intersects the glass in the second interval resulting in a sharp image. The smoothing from the previous section hides the discontinuity but the unexpected defocus behaviour remains (Figure 4d). Simply ignoring the refractive interactions in the context of the focus range code and applying the shear at the focus plane as normal, results in the artifacts shown in Figure 4e. We address this issue by applying an incremental shear at each refractive interaction until the first rough interaction. The partial shear at refractive surfaces can be calculated similar to the smoothing through the parameter  $t$  in Equation 1. At each refractive interaction we choose  $t$  proportional



**Figure 4:** Tracking the focus range ray intervals only to the first refractive interaction causes artifacts, as the amount of defocus depends on the interval where the ray intersects the refractive object. We address this issue by applying partial shears at refractive interfaces, similar to the smoothing in subsection 2.1. Image copyright © 2023 Animal Logic Pty Ltd. All Rights Reserved.

to the distance traveled since the last intersection:

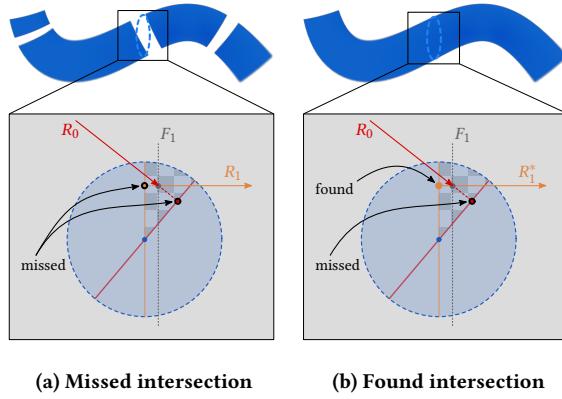
$$t = \max \left( 0, 1 - \frac{F_{i+1} - z}{\delta} \right) - t_{acc} \quad (4)$$

We keep track of the accumulated  $t_{acc}$  in the current focus range segment which allows us to apply the remaining shear when reaching the focus plane. At the first rough interaction we stop tracking the shearing and focus range interval and simply trace the BSDF sampled continuation ray. Figure 4f shows the results of this technique. As expected, the objects are slightly defocused and the glass plane has no visible influence on the defocus blur.

## 2.3 Ray-Oriented Geometry

Camera facing primitives such as discs or flat curves [Wald et al. 2014] are common in production rendering. Their intersections can be efficiently tested in 2D with a plane orthogonal to the ray direction. However, when such geometry overlaps any of the focus planes  $F_j$ , the change of ray direction causes missed intersections as illustrated in Figure 5a. The incoming ray  $R_0$  intersects the red line orthogonal to the ray. As the ray reaches the focus plane  $F_1$  before the intersection point, its direction gets changed and the new ray  $R_1$  intersects a different ray-oriented segment (orange). This intersection point lies behind the origin of  $R_1$ , and as a result we miss the intersection completely even though the ray clearly passes through the object.

In such degenerate configurations, we opt to catch the missed intersection as part of the intersection test of  $R_{i+1}$  by changing the curve intersection test for  $R_{i+1}$  only if  $R_i$  misses and  $R_{i+1}$  starts within the primitive. First, we need to skip the self-intersection test, because the ray origin is inside the primitive. Second, we need to



**Figure 5:** In (a) both rays miss the primitive when the change of ray direction due to the focus plane happens in the checkered region. In (b) we pass two boolean flags to  $R_1$  to allow (i) backward intersection of flat curves and (ii) the ray to start within the primitive to find the intersection. Image copyright © 2023 Animal Logic Pty Ltd. All Rights Reserved.

skip the near test, so that the intersection can be found backwards on  $R_{i+1}$ . This slight modification successfully recovers the missed intersections as depicted in Figure 5b.

### 3 DISCUSSION

*Limitations.* The smoothing technique of subsection 2.1 improves the look across discontinuities but it may not always meet aesthetic preferences, especially when  $\delta$  is too small compared to the size of the highlight. As production shaders are typically composed by a combination of complex lobes and the problematic specular BSDF isn't usually used in isolation, we couldn't observe visible artifacts in practice. However, a disparity between the amount of geometric defocus and shading defocus as well as the geometric error described in subsection 2.1 could theoretically lead to perceivable artifacts. We believe further subtle challenges lie in similar implementations to ours, as volume integration and differential calculations may necessitate modifications to produce consistent results with a path divided into multiple ray segments.

*Performance.* In Table 1 we compare the performance of our method to the two configurations of Figure 1c and 1d. For realistic workloads with complex materials and textures (ALab), the coherency of ray traversal is more relevant to the render time and the highly defocused thin lens tends to be slower. This is also visible in the Attic scene which contains color textures but just a principled BSDF. For the unfavourable case of outdoor (low-bounces) geometry only scenes (Moana) the overhead of additional BVH traversals for the focus range implementation is around 10%.

### 4 CONCLUSION

We have presented a set of techniques and production cases for implementing more complex defocus effects using piece-wise linear ray tracing. The techniques described can be applicable to other variants of generalized depth of field as they are visually pleasing solutions to ray discontinuities. It would be interesting to explore if



Ten bounces	(a)	(b)	(c)	(d)
focus range	1674.2s	1177.7s	796.2s	3234.0s
thin lens i)	1680.5s (-0.4%)	1192.5s (+1.3%)	811.0s (+1.9%)	3150.7s (-2.6%)
thin lens ii)	1570.5s (-6.2%)	1105.9s (-6.1%)	786.7s (-1.2%)	3138.5s (-2.3%)
direct only	(a)	(b)	(c)	(d)
focus range	451.3s	293.9s	108.4s	591.4s
thin lens i)	465.9s (+3.2%)	314.8s (+7.1%)	103.1s (-4.9%)	535.2s (-9.5%)
thin lens ii)	437.1s (-3.1%)	280.8s (-4.5%)	99.3s (-8.4%)	525.6s (-11.1%)

**Table 1: Render time (1024spp) for focus range and two thin lens configurations with i) comparable background defocus and ii) comparable in focus region. Copyright © 2023 Animal Logic Pty Ltd. All Rights Reserved; Animal Logic ALab Copyright 2022 Animal Logic Pty Limited. All rights reserved.**

artists prefer for the background defocus scale to be automatically adjusted to account for the length of the focus range, similar to the ray construction based on a per-pixel virtual lens described by Munkberg et al. [2012] in their Figure 9. Further exploration of our system after reflection events and compatibility with bidirectional algorithms are compelling areas for research.

### ACKNOWLEDGMENTS

Thanks to the Glimpse rendering team for the continued support, Jeff Renton, Steve Agland and Aniket Raj and the lighting/layout teams for help during the various stages of the project. NVIDIA Attic Scene published under CC BY 4.0. Moana Island Scene: Publicly available dataset courtesy of Walt Disney Animation Studios

### REFERENCES

- Alexandre Chapiro, Carol O'Sullivan, Wojciech Jarosz, Markus Gross, and Aljoscha Smolic. 2015. Stereo from Shading. In *Proceedings of EGSR (Experimental Ideas Implementations)*.
- Robert L. Cook, Thomas Porter, and Loren Carpenter. 1984. Distributed Ray Tracing. *SIGGRAPH Comput. Graph.* 18, 3 (1984), 137–145.
- Craig Kolb, Don Mitchell, and Pat Hanrahan. 1995. A Realistic Camera Model for Computer Graphics. In *Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '95)*, 317–324.
- Todd J. Kosloff and Brian A. Barsky. 2009. Three Techniques for Rendering Generalized Depth of Field Effects. In *Proceedings of the Fourth SIAM Conference on Mathematics for Industry: Challenges and Frontiers (MI09)*, 42–48.
- Jacob Munkberg, Robert Toth, and Tomas Akenine-Möller. 2012. Per-Vertex Defocus Blur for Stochastic Rasterization. *Computer Graphics Forum*, 31, 4 (2012), 1385–1389.
- Matt Pharr, Wenzel Jakob, and Greg Humphreys. 2016. *Physically Based Rendering: From Theory to Implementation* (3rd ed.).
- Pixar. 2019. The real fake cameras of Toy Story 4. <https://youtu.be/AcZ2OY5-TeM>.
- Ingo Wald, Sven Woop, Carsten Benthin, Gregory S Johnson, and Manfred Ernst. 2014. Embree: a kernel framework for efficient CPU ray tracing. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–8.