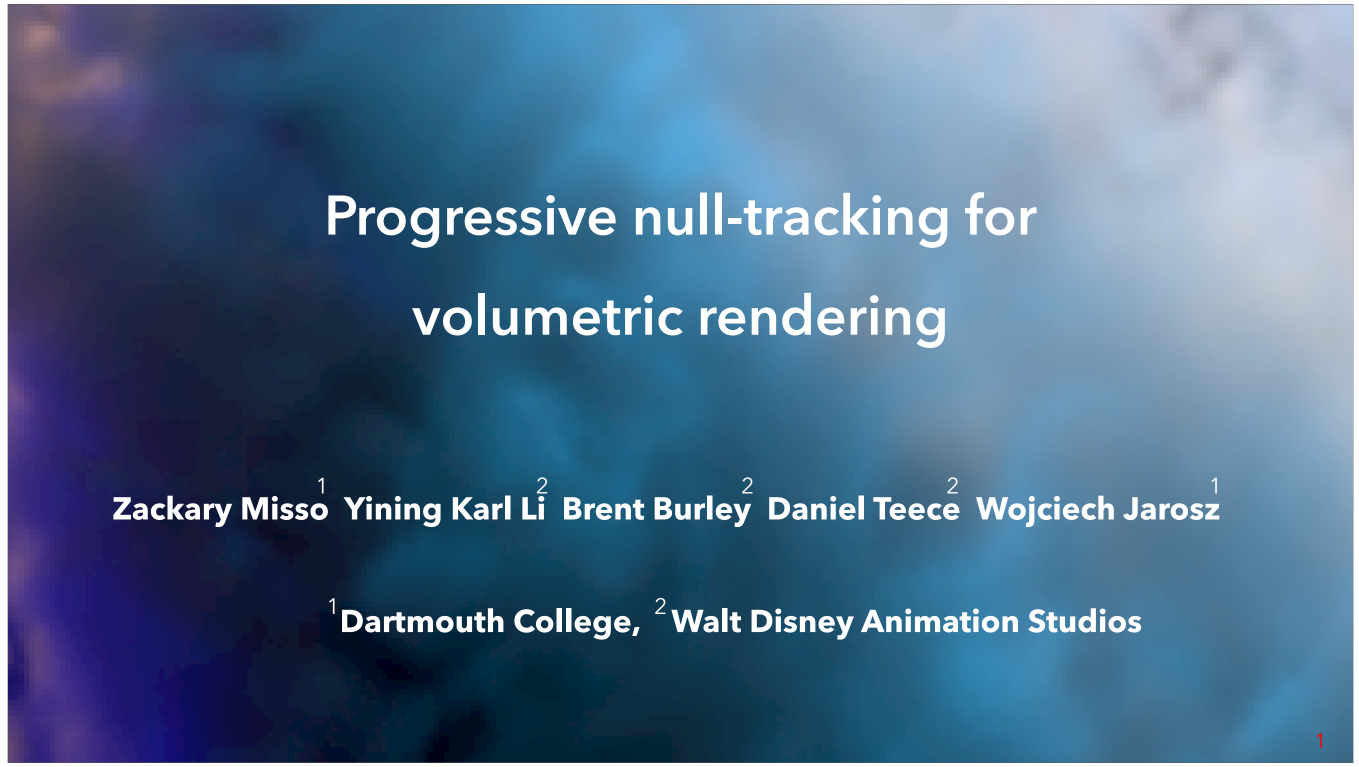
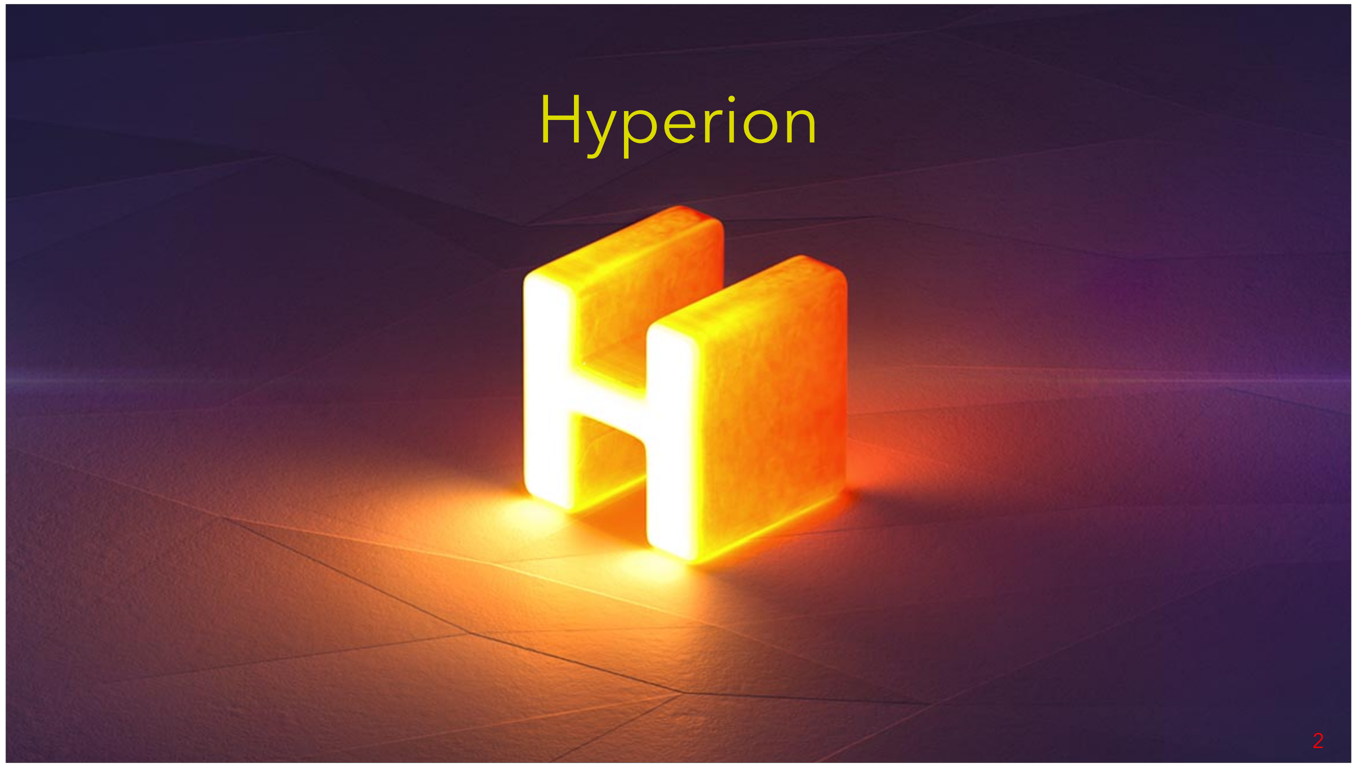
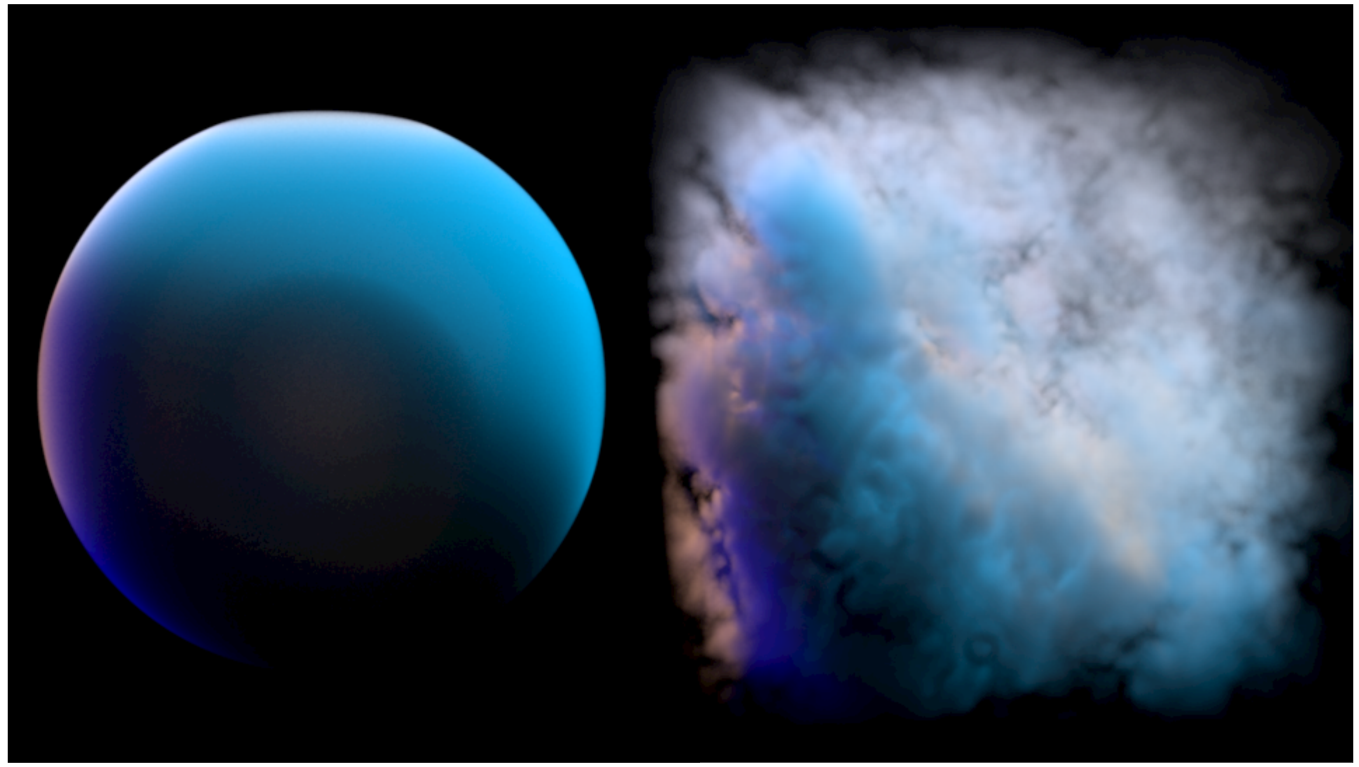



Good afternoon everyone! This project began early last summer during an internship at Walt Disney Animation,

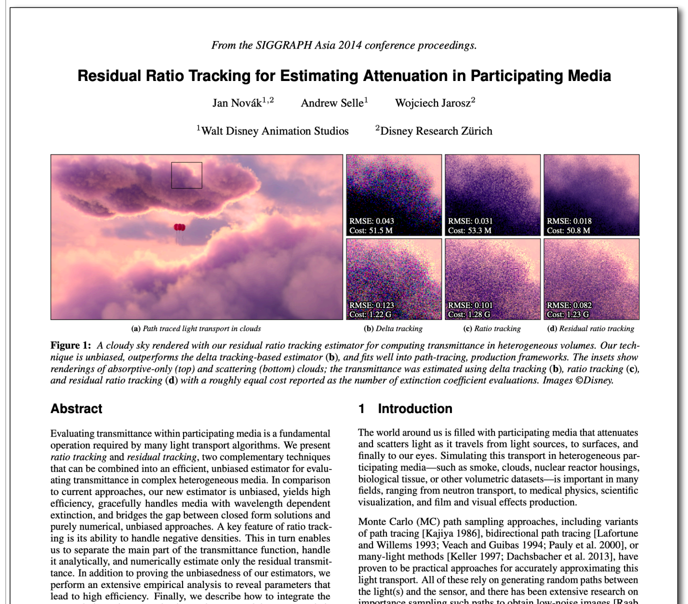



Where we initially met with a team of VFX artists see what features they wanted from the next generation of Disney's Hyperion rendering engine. One of the features which they requested



was for Hyperion to eventually support rendering general procedural media.


This would allow for supporting near infinite amounts of detail within the volumes used in their theatrical productions. However, due to algorithmic constraints this is not currently supported



*From the SIGGRAPH Asia 2014 conference proceedings.*

## Residual Ratio Tracking for Estimating Attenuation in Participating Media

Jan Novák[1,2] Andrew Selle[1] Wojciech Jarosz[2]

[1]Walt Disney Animation Studios [2]Disney Research Zürich

RMSE: 0.043 Cost: 51.5 M | RMSE: 0.031 Cost: 53.3 M | RMSE: 0.018 Cost: 50.8 M

RMSE: 0.123 Cost: 1.22 G | RMSE: 0.101 Cost: 1.28 G | RMSE: 0.082 Cost: 1.23 G

**(a)** *Path traced light transport in clouds* **(b)** *Delta tracking* **(c)** *Ratio tracking* **(d)** *Residual ratio tracking*

**Figure 1:** *A cloudy sky rendered with our residual ratio tracking estimator for computing transmittance in heterogeneous volumes. Our technique is unbiased, outperforms the delta tracking-based estimator* (**b**)*, and fits well into path-tracing, production frameworks. The insets show renderings of absorptive-only (top) and scattering (bottom) clouds; the transmittance was estimated using delta tracking* (**b**)*, ratio tracking* (**c**)*, and residual ratio tracking* (**d**) *with a roughly equal cost reported as the number of extinction coefficient evaluations. Images ©Disney.*

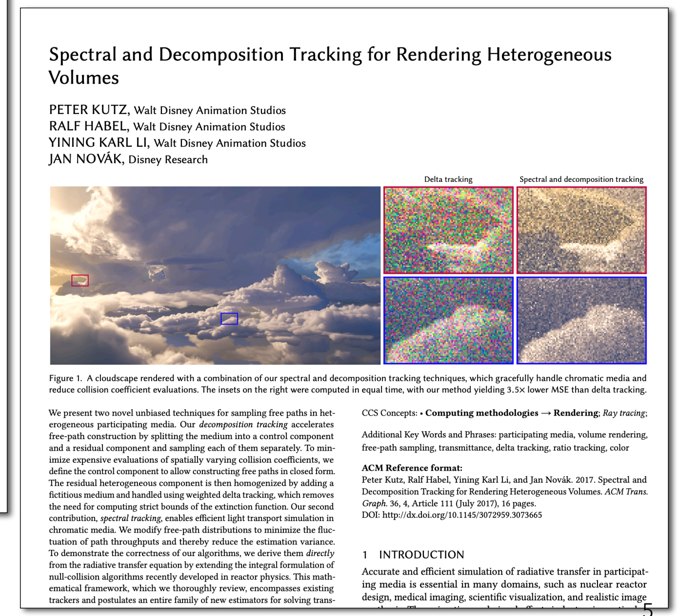
### Abstract

Evaluating transmittance within participating media is a fundamental operation required by many light transport algorithms. We present *ratio tracking* and *residual tracking*, two complementary techniques that can be combined into an efficient, unbiased estimator for evaluating transmittance in complex heterogeneous media. In comparison to current approaches, our new estimator is unbiased, yields high efficiency, gracefully handles media with wavelength dependent extinction, and bridges the gap between closed form solutions and purely numerical, unbiased approaches. A key feature of ratio tracking is its ability to handle negative densities. This in turn enables us to separate the main part of the transmittance function, handle it analytically, and numerically estimate only the residual transmittance. In addition to proving the unbiasedness of our estimators, we perform an extensive empirical analysis to reveal parameters that lead to high efficiency. Finally, we describe how to integrate the

### 1 Introduction

The world around us is filled with participating media that attenuates and scatters light as it travels from light sources, to surfaces, and finally to our eyes. Simulating this transport in heterogeneous participating media—such as smoke, clouds, nuclear reactor housings, biological tissue, or other volumetric datasets—is important in many fields, ranging from neutron transport, to medical physics, scientific visualization, and film and visual effects production.

Monte Carlo (MC) path sampling approaches, including variants of path tracing [Kajiya 1986], bidirectional path tracing [Lafortune and Willems 1993; Veach and Guibas 1994; Pauly et al. 2000], or many-light methods [Keller 1997; Dachsbacher et al. 2013], have proven to be practical approaches for accurately approximating this light transport. All of these rely on generating random paths between the light(s) and the sensor, and there has been extensive research on importance sampling such paths to obtain low-noise images [Raab



## Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes

PETER KUTZ, Walt Disney Animation Studios
RALF HABEL, Walt Disney Animation Studios
YINING KARL LI, Walt Disney Animation Studios
JAN NOVÁK, Disney Research

Delta tracking | Spectral and decomposition tracking

Figure 1. A cloudscape rendered with a combination of our spectral and decomposition tracking techniques, which gracefully handle chromatic media and reduce collision coefficient evaluations. The insets on the right were computed in equal time, with our method yielding 3.5× lower MSE than delta tracking.

We present two novel unbiased techniques for sampling free paths in heterogeneous participating media. Our *decomposition tracking* accelerates free-path construction by splitting the medium into a control component and a residual component and sampling each of them separately. To minimize expensive evaluations of spatially varying collision coefficients, we define the control component to allow constructing free paths in closed form. The residual heterogeneous component is then homogenized by adding a fictitious medium and handled using weighted delta tracking, which removes the need for computing strict bounds of the extinction function. Our second contribution, *spectral tracking*, enables efficient light transport simulation in chromatic media. We modify free-path distributions to minimize the fluctuation of path throughputs and thereby reduce the estimation variance. To demonstrate the correctness of our algorithms, we derive them *directly* from the radiative transfer equation by extending the integral formulation of null-collision algorithms recently developed in reactor physics. This mathematical framework, which we thoroughly review, encompasses existing trackers and postulates an entire family of new estimators for solving trans-

CCS Concepts: • **Computing methodologies** → **Rendering**; *Ray tracing*;

Additional Key Words and Phrases: participating media, volume rendering, free-path sampling, transmittance, delta tracking, ratio tracking, color

**ACM Reference format:**
Peter Kutz, Ralf Habel, Yining Karl Li, and Jan Novák. 2017. Spectral and Decomposition Tracking for Rendering Heterogeneous Volumes. *ACM Trans. Graph.* 36, 4, Article 111 (July 2017), 16 pages.
DOI: http://dx.doi.org/10.1145/3072959.3073665

### 1 INTRODUCTION

Accurate and efficient simulation of radiative transfer in participating media is essential in many domains, such as nuclear reactor design, medical imaging, scientific visualization, and realistic image

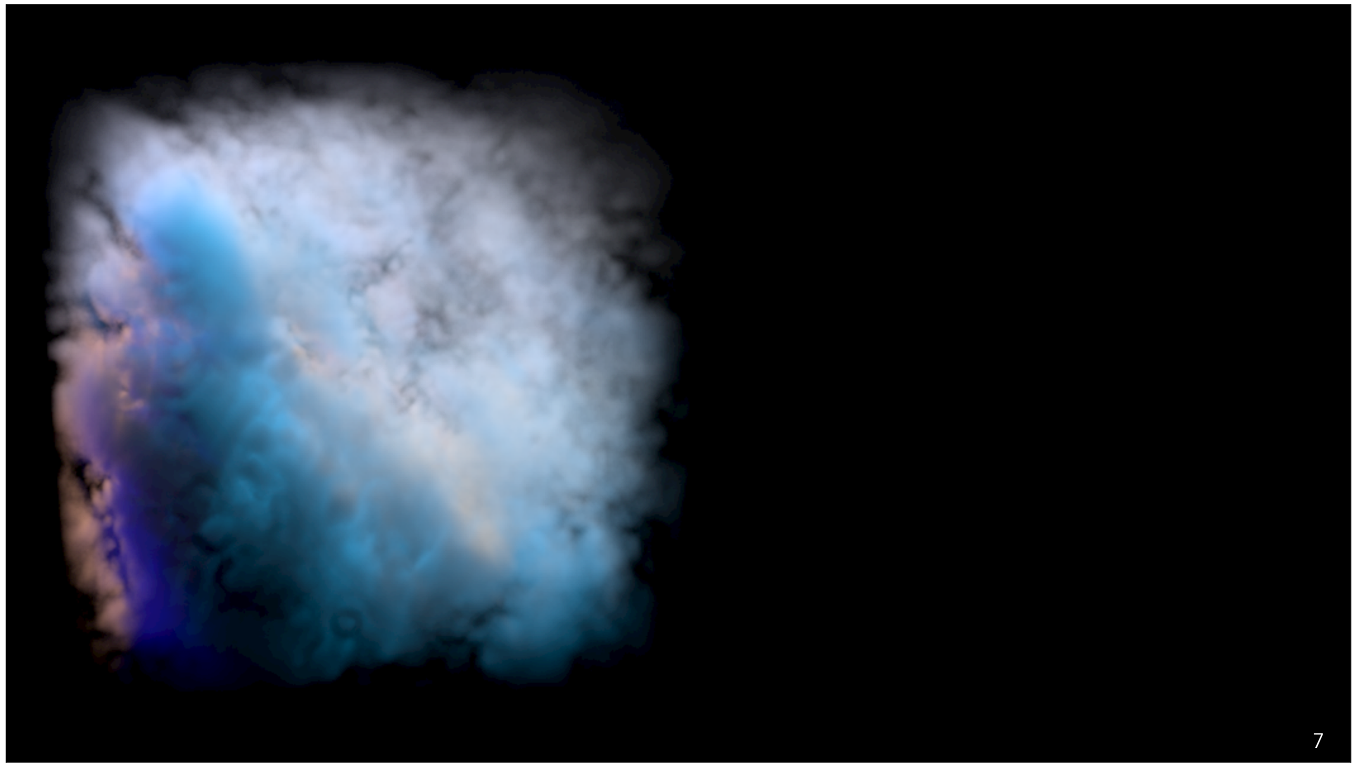


To render their productions, Hyperion currently relies on techniques based on the null-scattering paradigm,

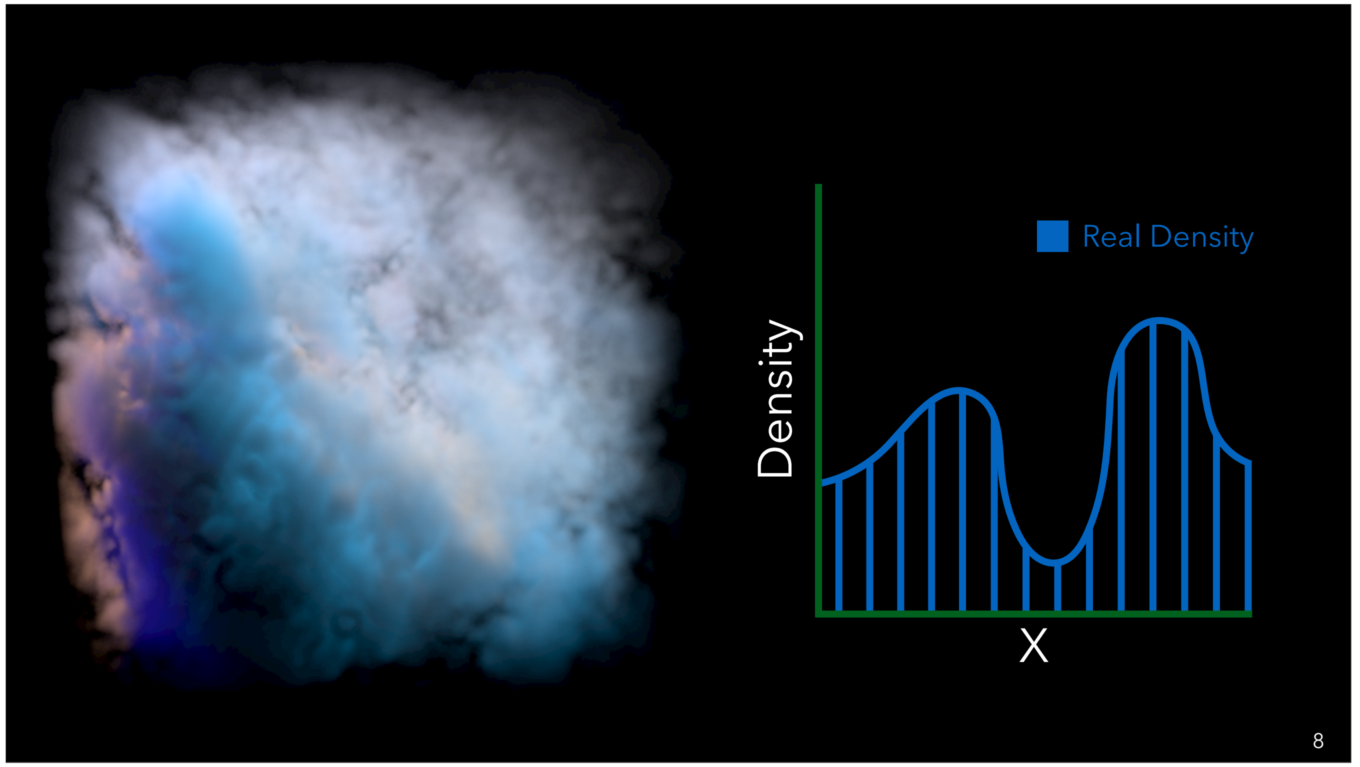
# Homogenization



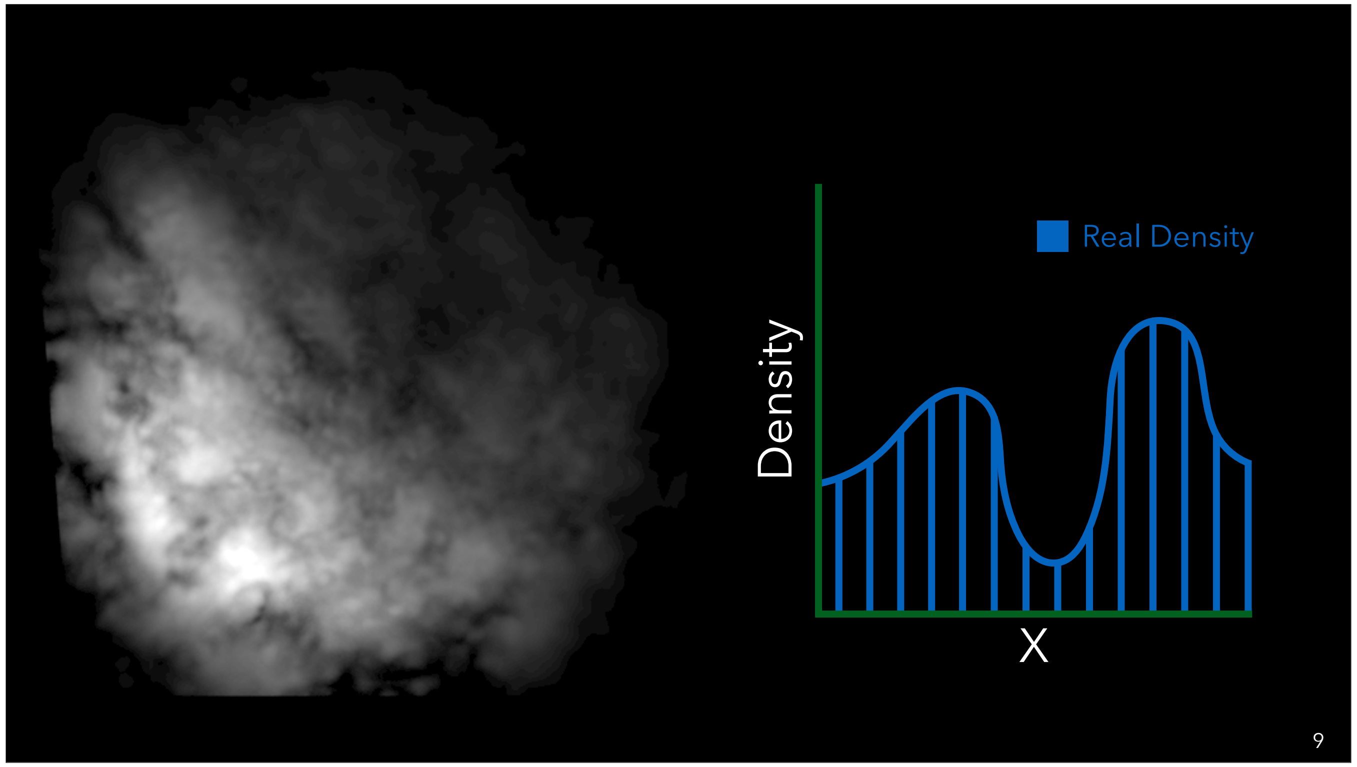
Which requires that we first homogenize all volumes.



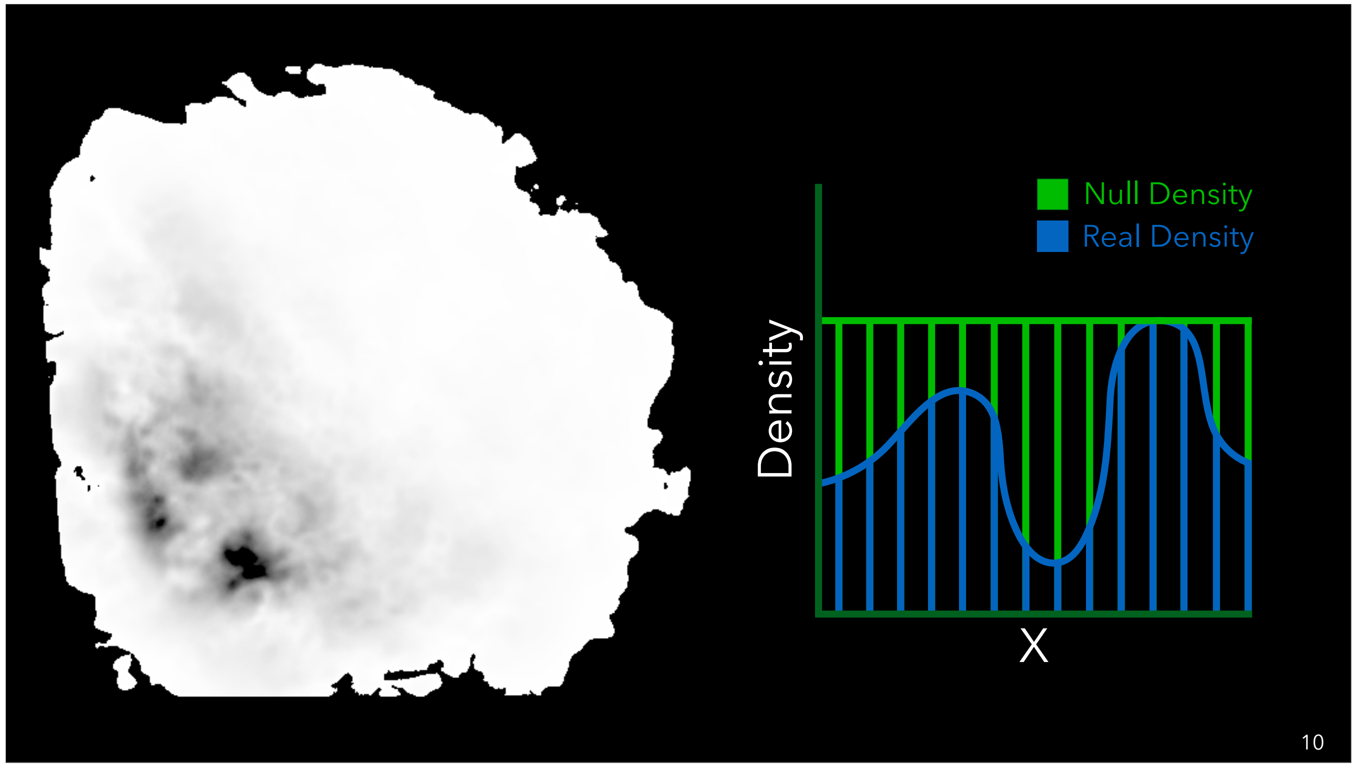

Take for example this heterogeneous volume,



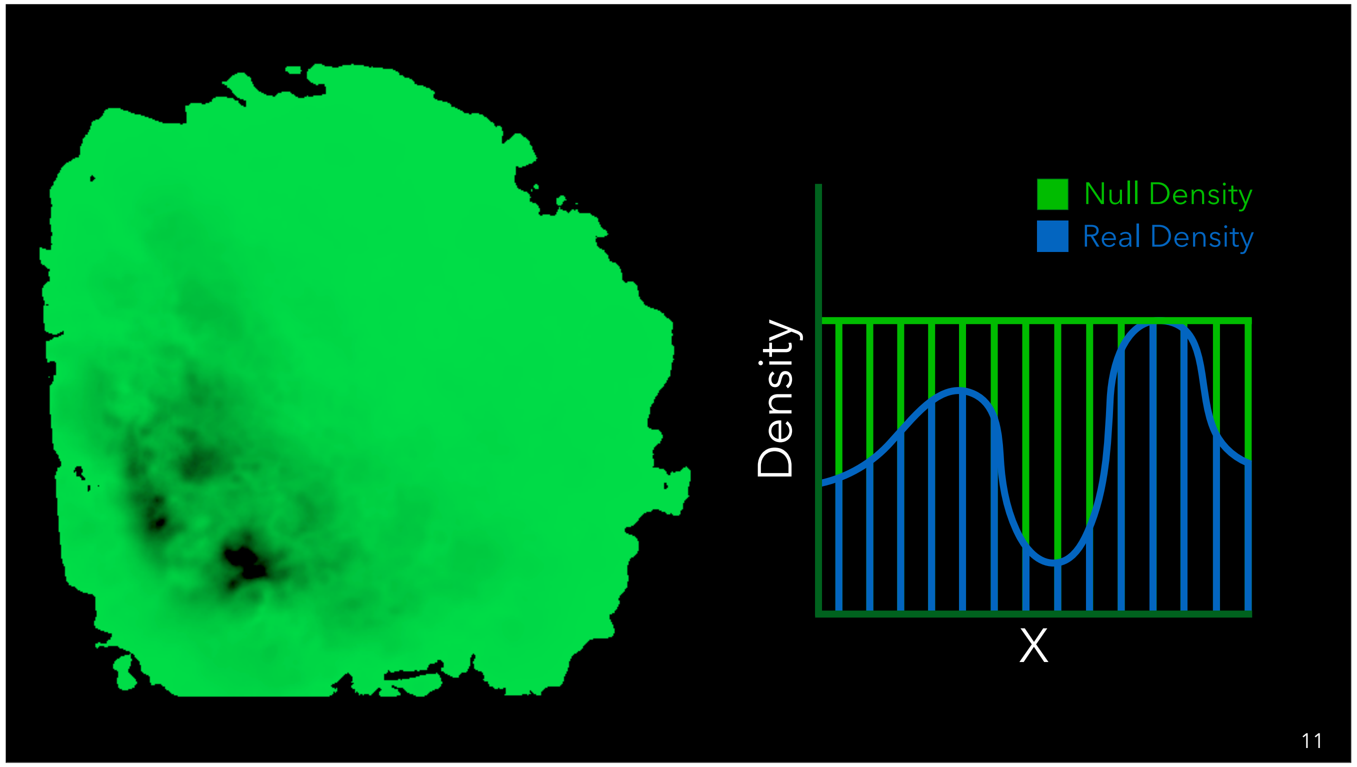

Which has a spatially varying density function. For visualization purposes, I am going to replace the rendering of the volume,



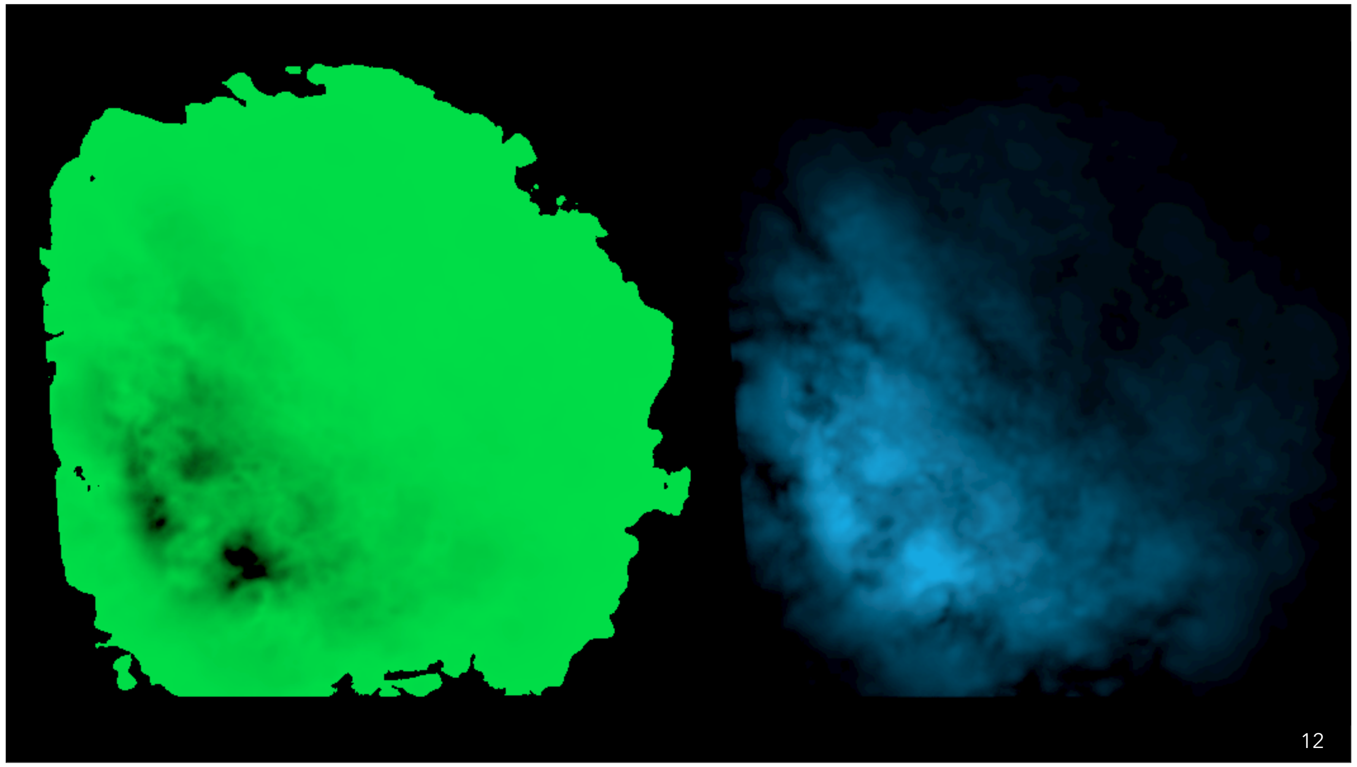

With a visualization of the average density looking straight through the medium. Null-scattering techniques require us



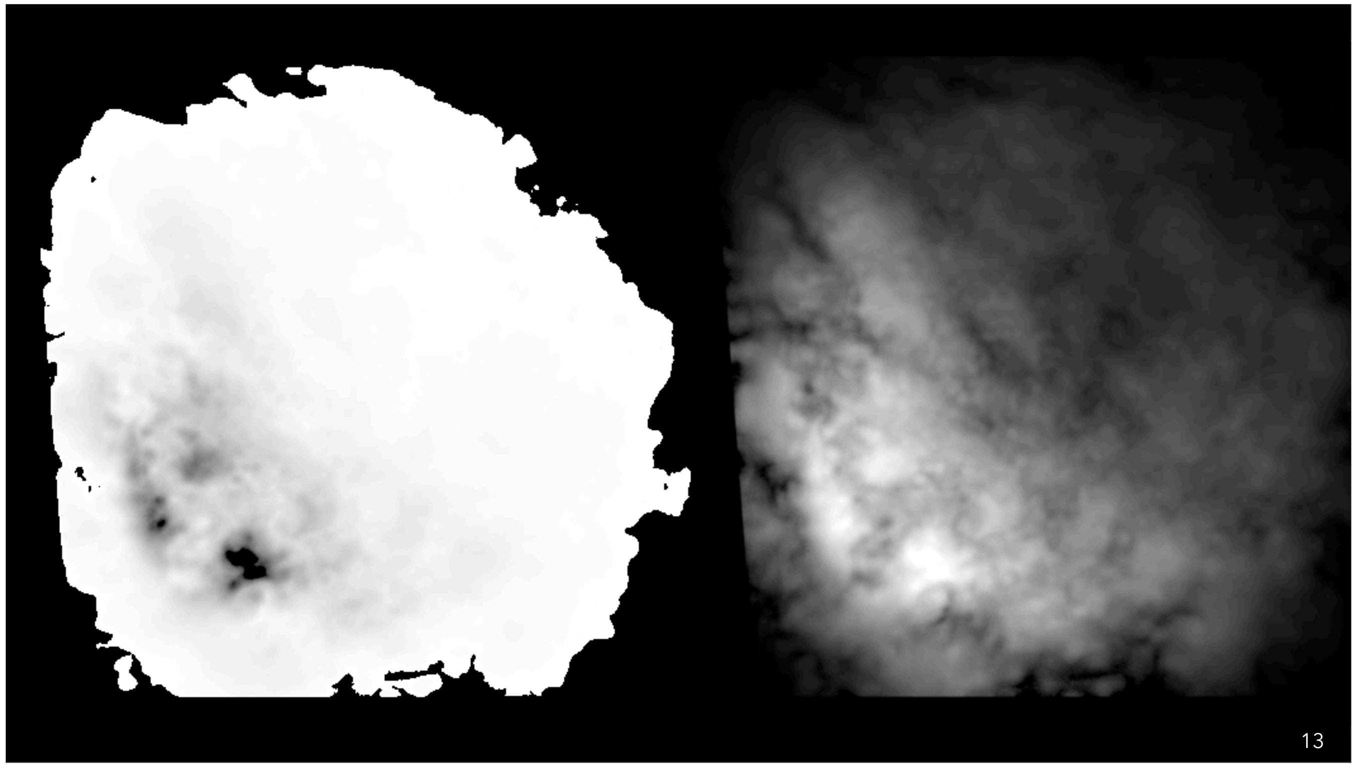

to inject ficticious, or null density into the medium, such that if we took



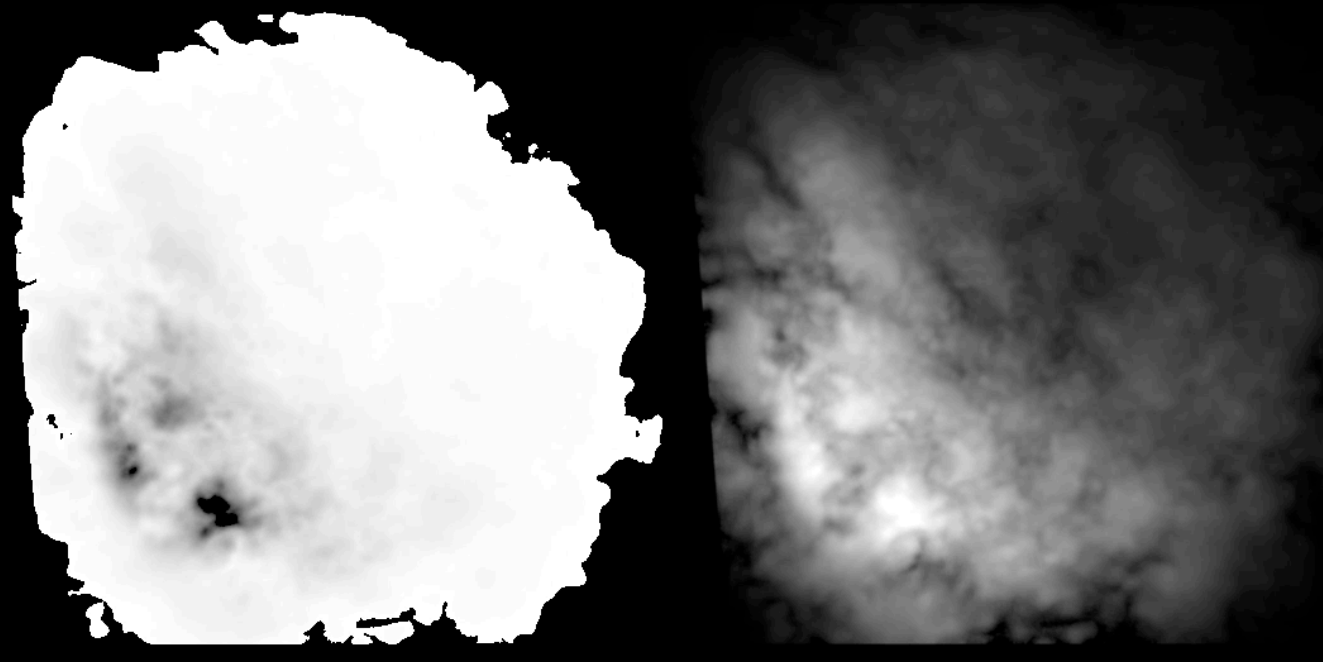

the null density,

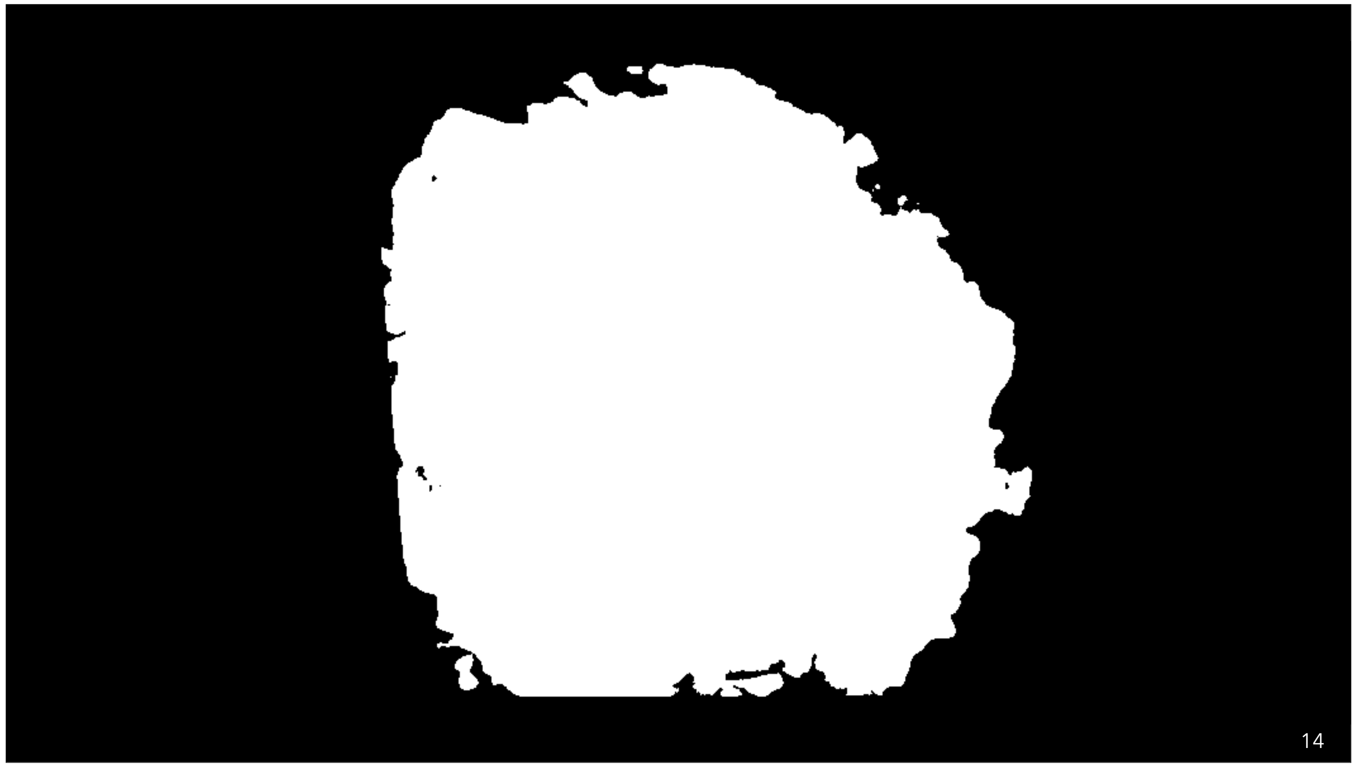



and real density

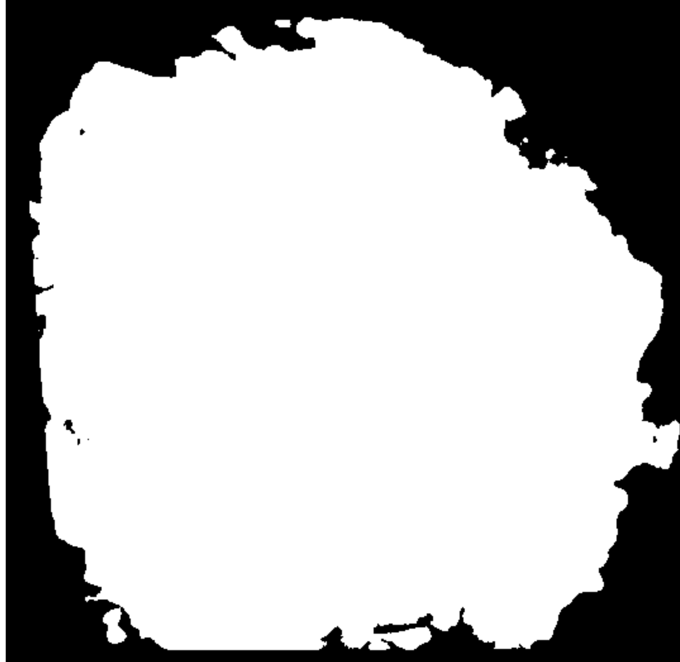


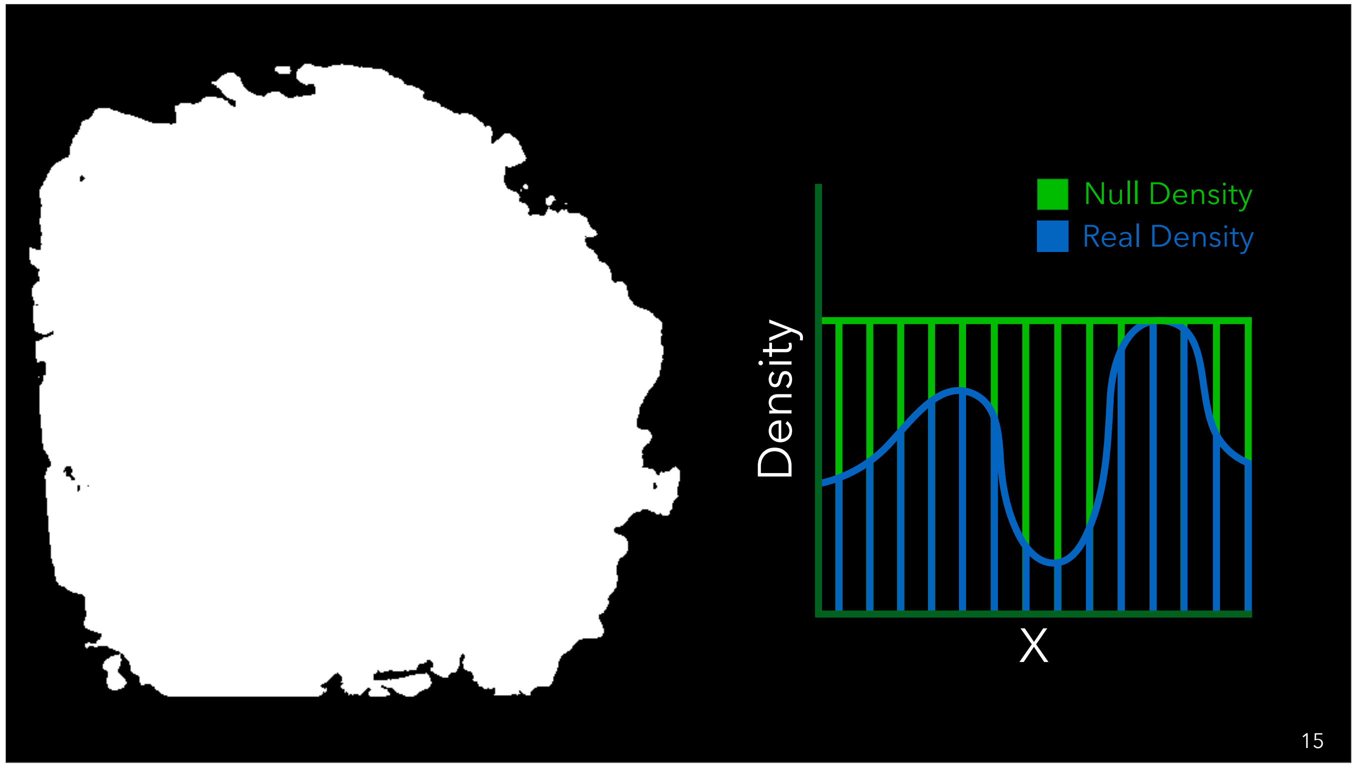

And then combined them by adding them together,



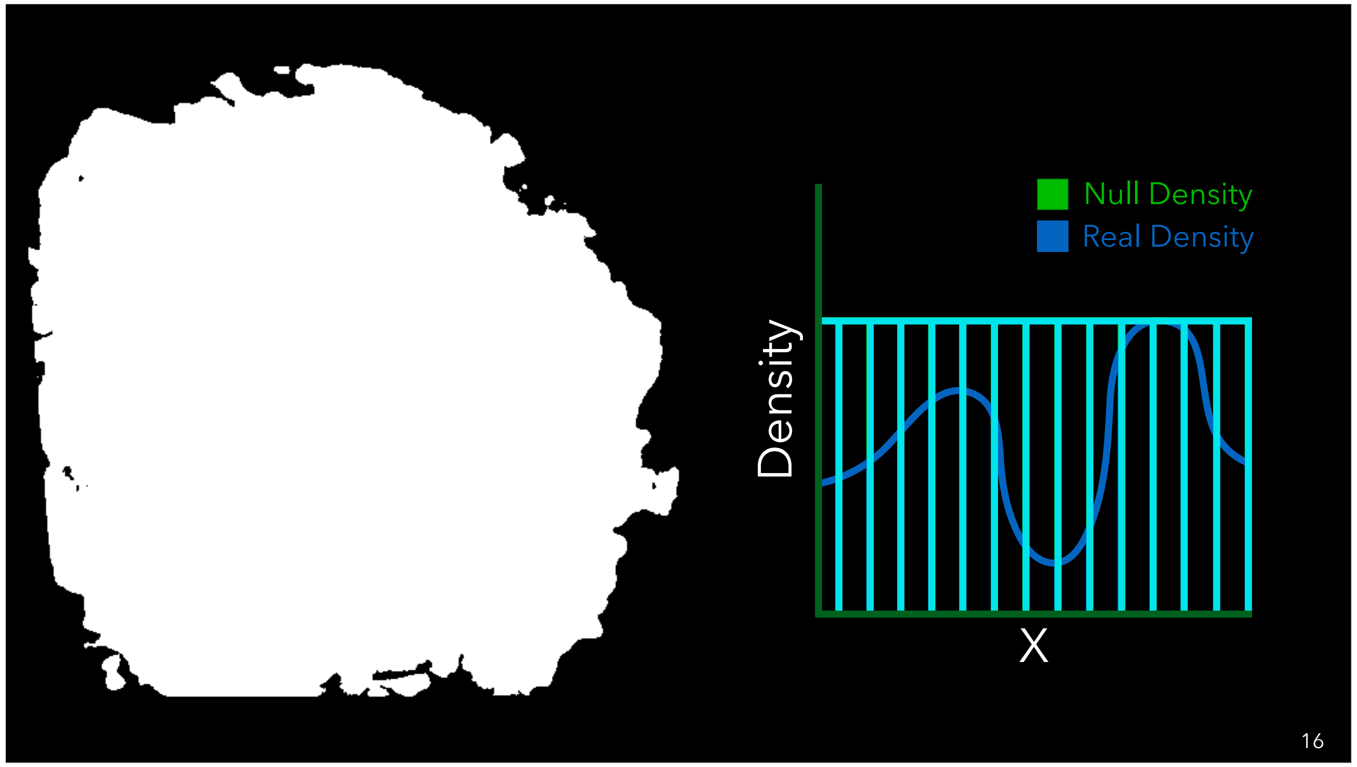



We would end up

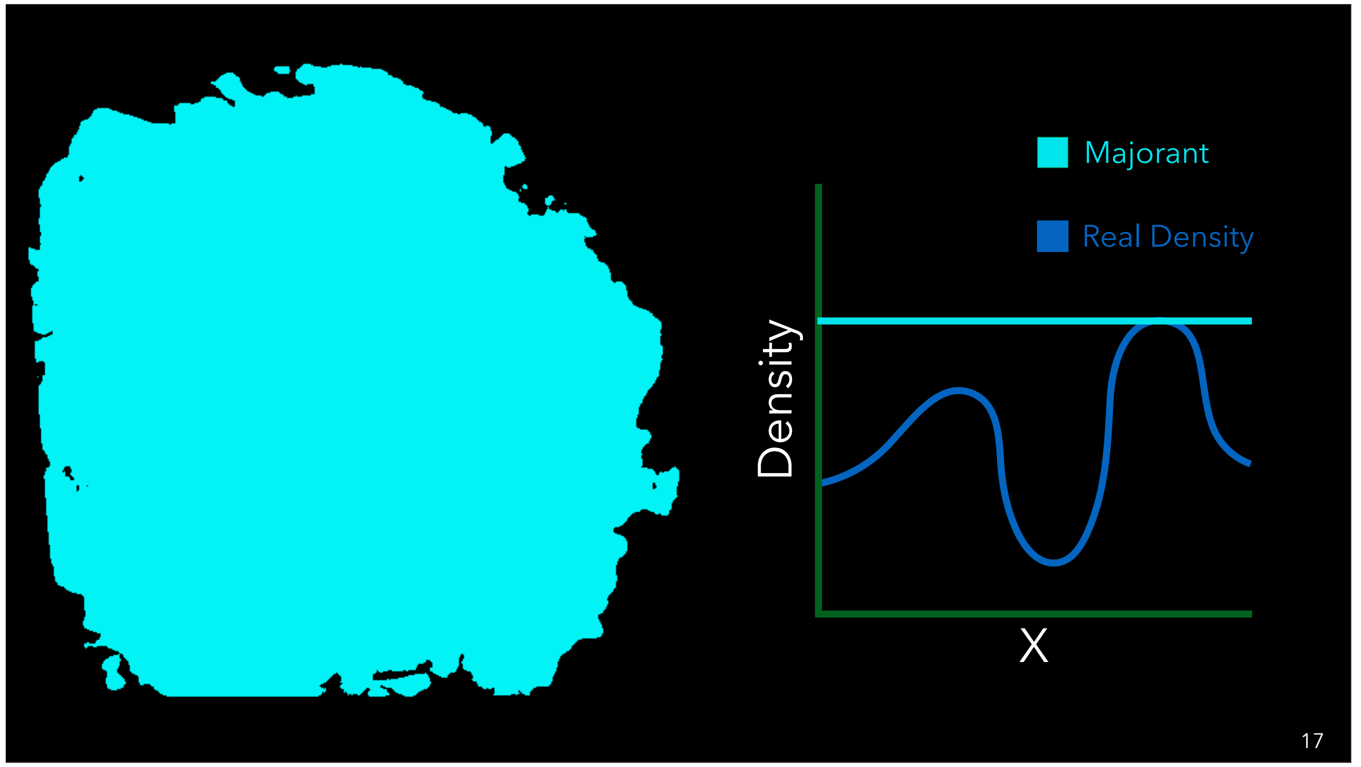




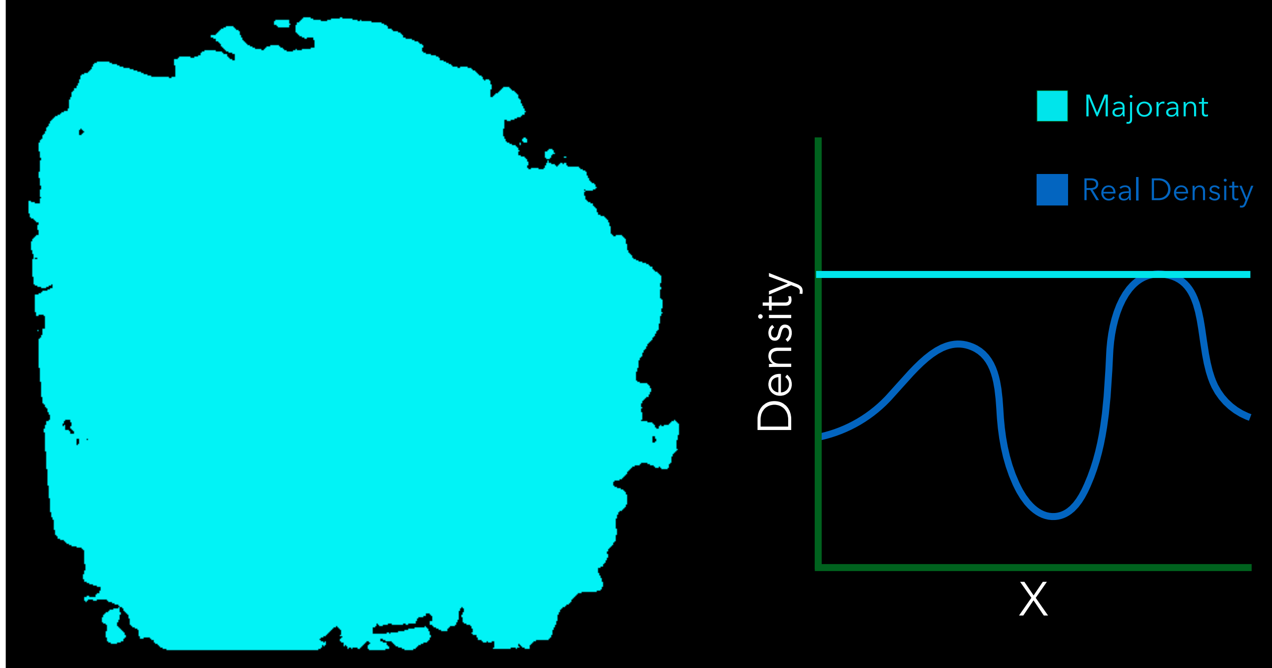

With a constant density medium whose

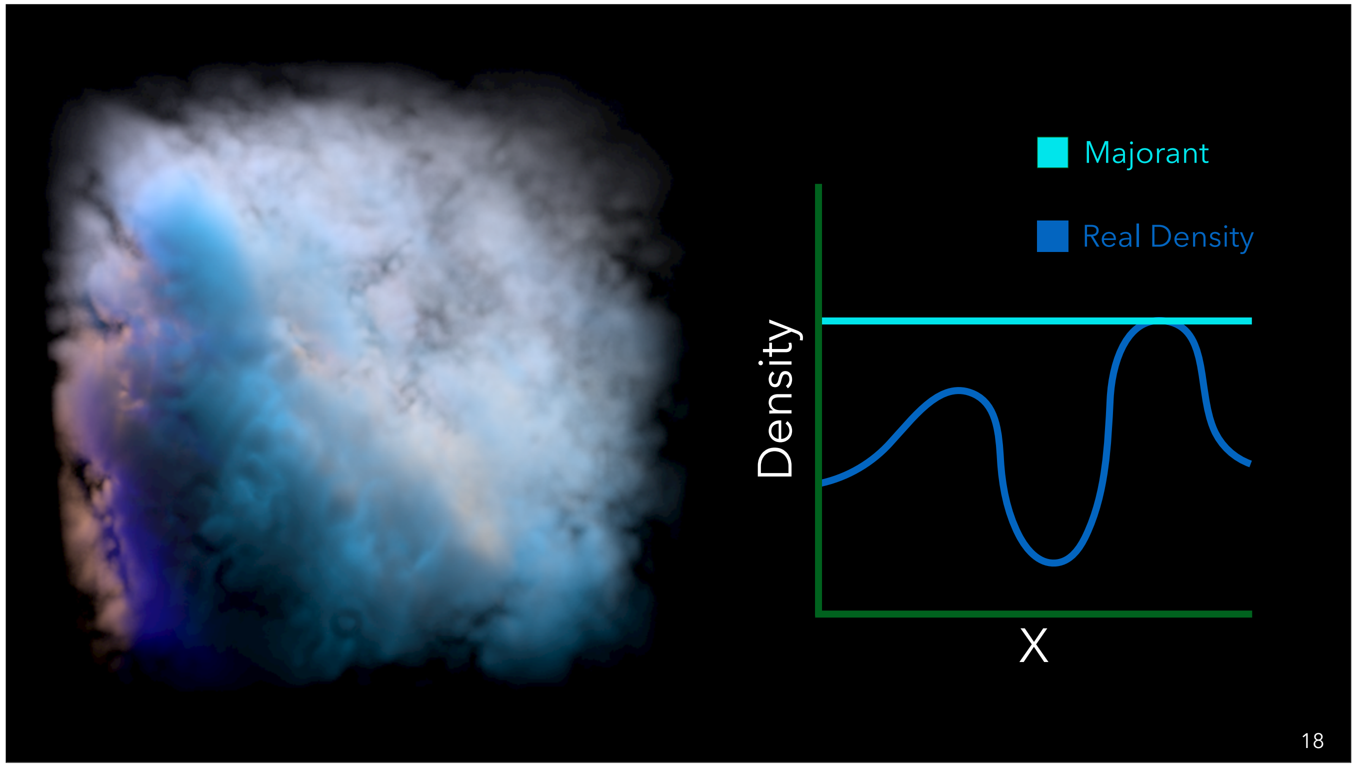



total combined density is often referred to

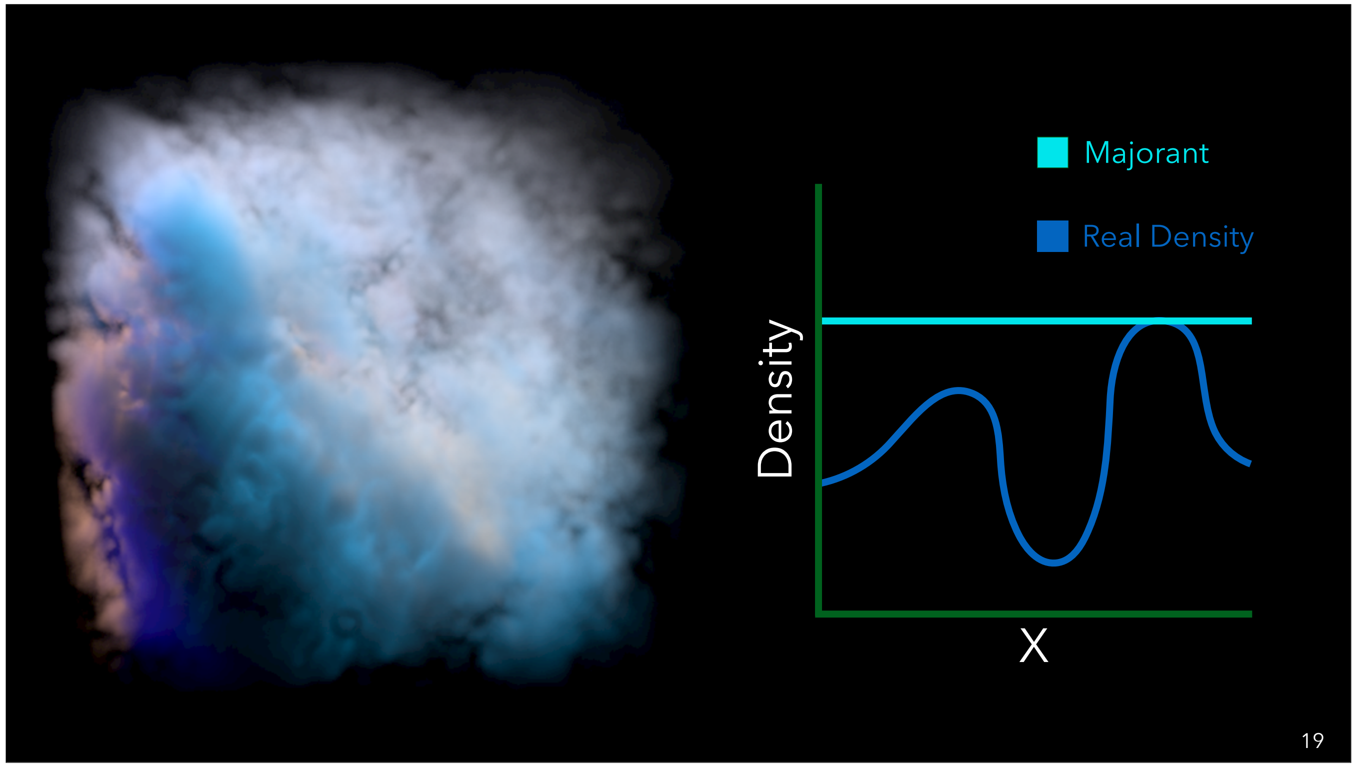



As the majorant. Now, most modern algorithms allow us to specify the majorant (click) as any non-zero positive value, and no matter what majorant we specify,

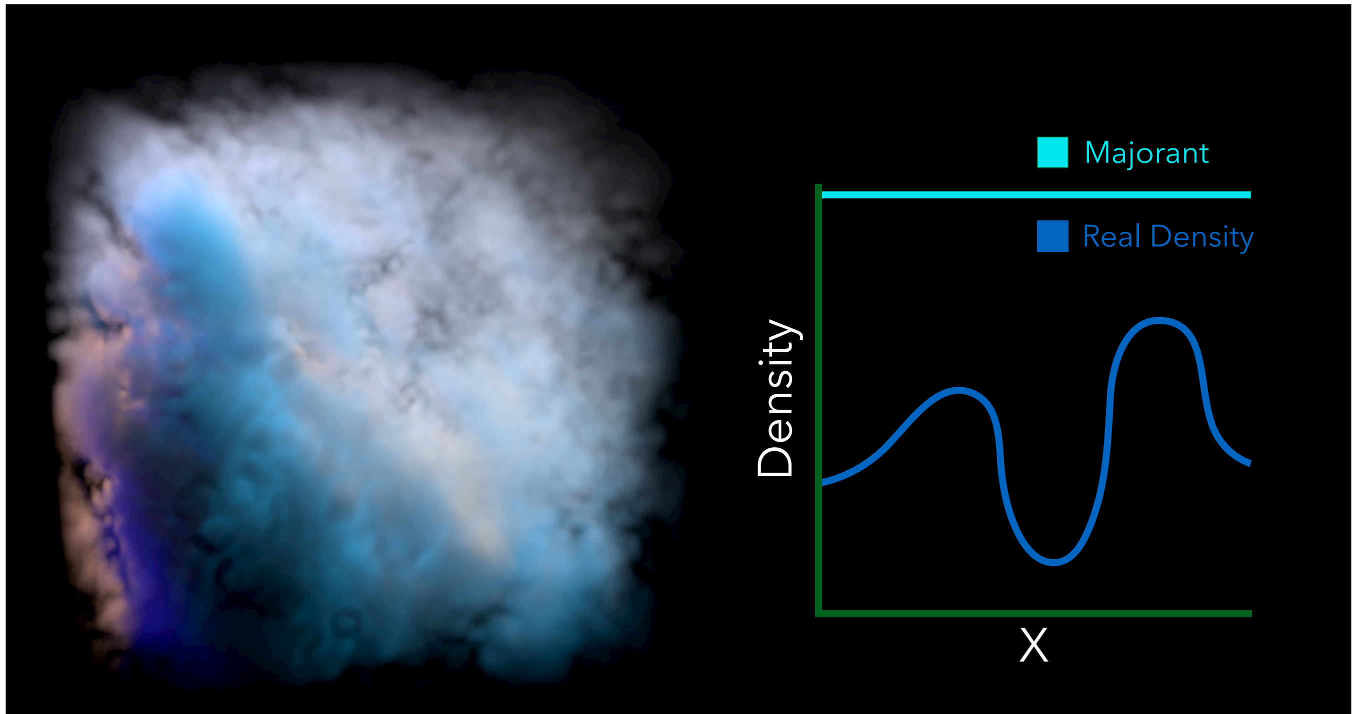

Majorant
Real Density
Density
X



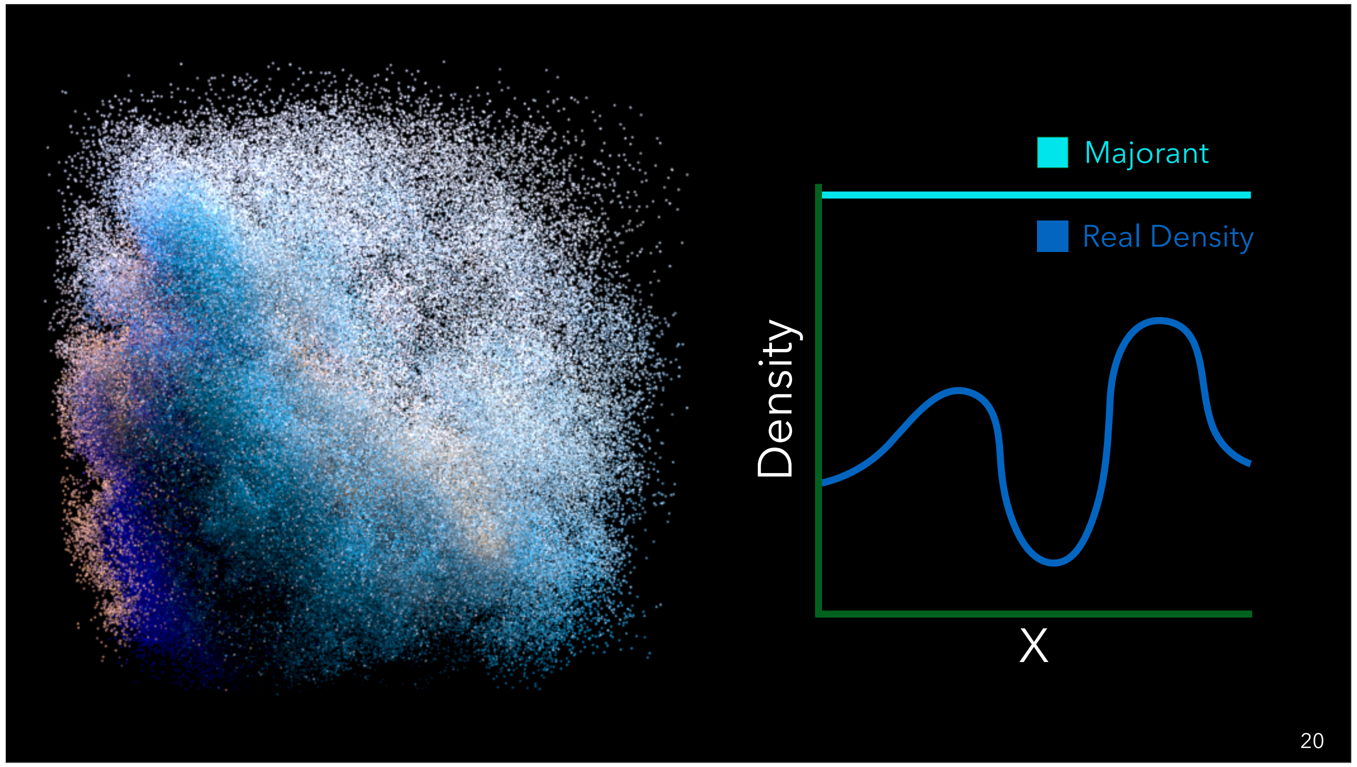

Modern null-scattering algorithms should still give us the correct expected result. However, the choice of the majorant directly impacts the performance of our renders.



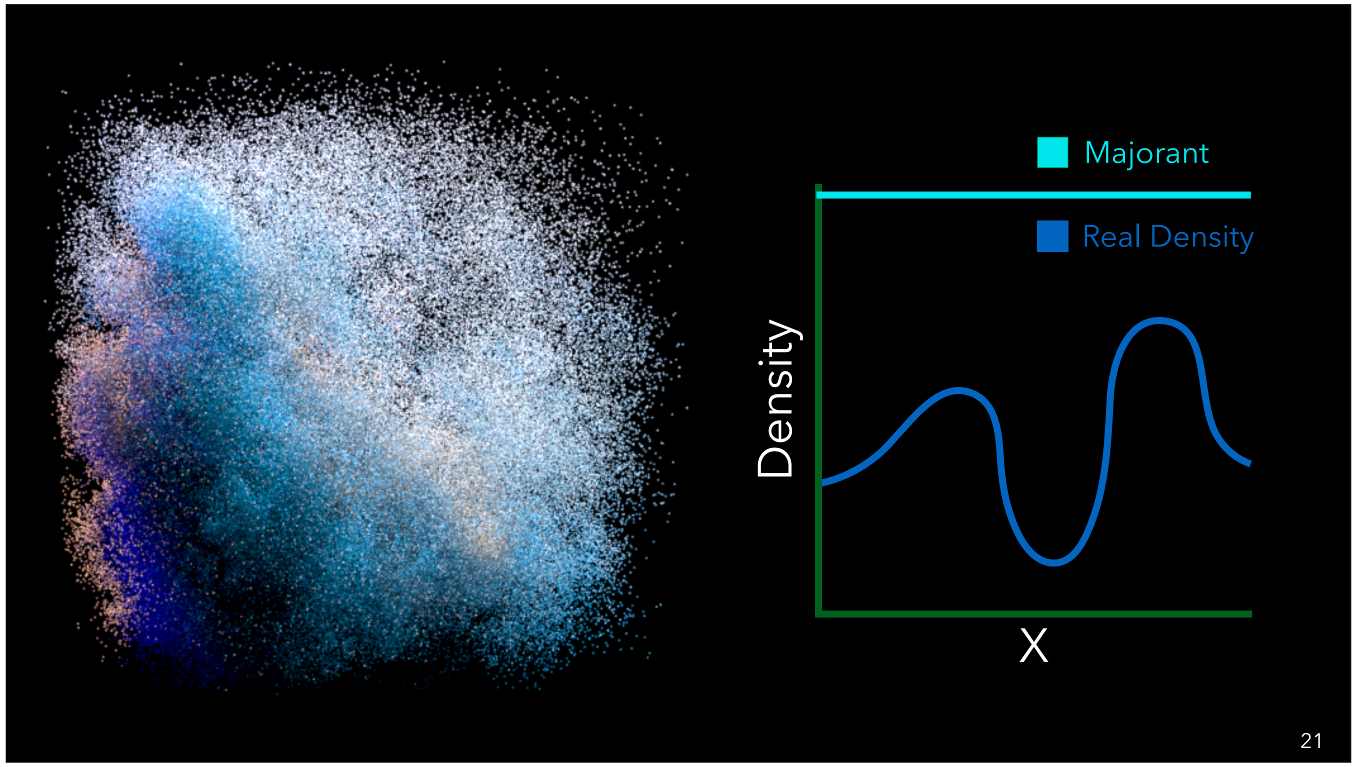

If the majorant is set too high (*click*),



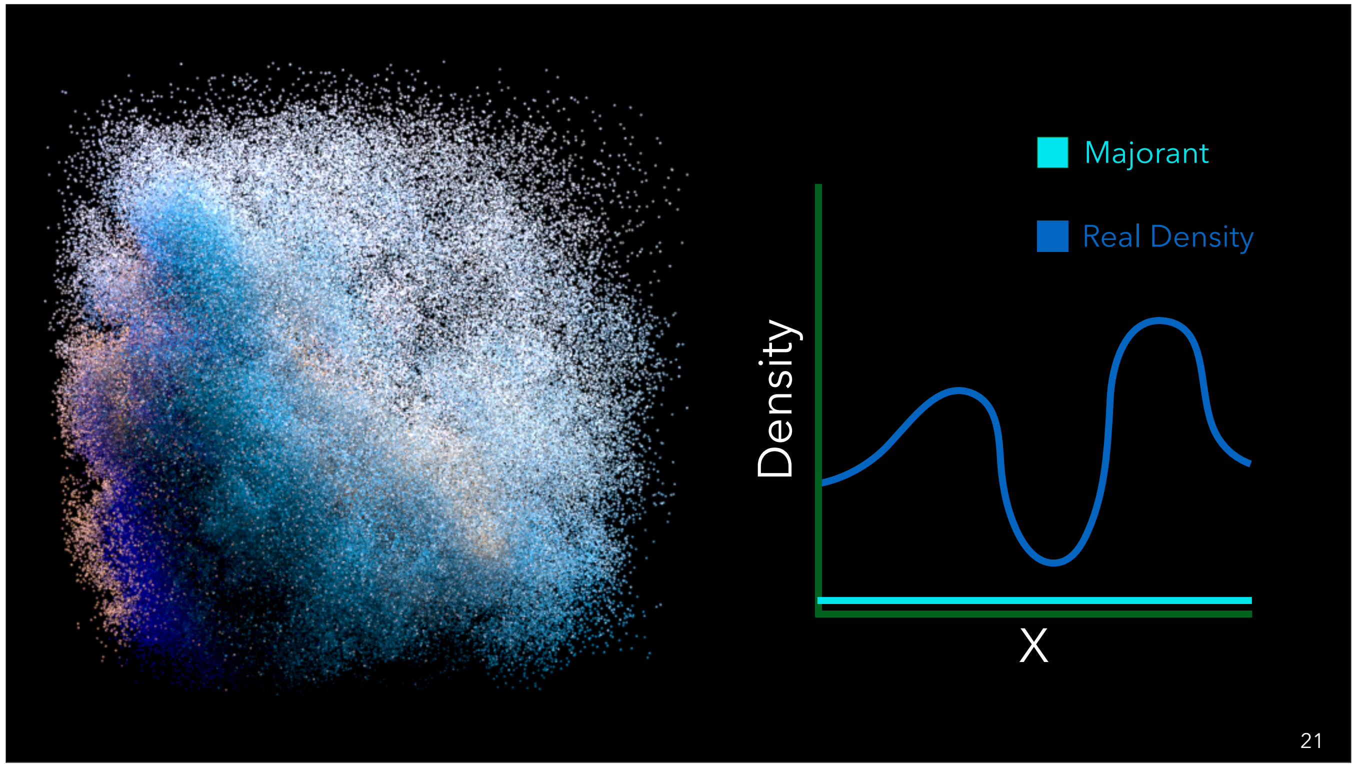
Majorant
Real Density
Density
X

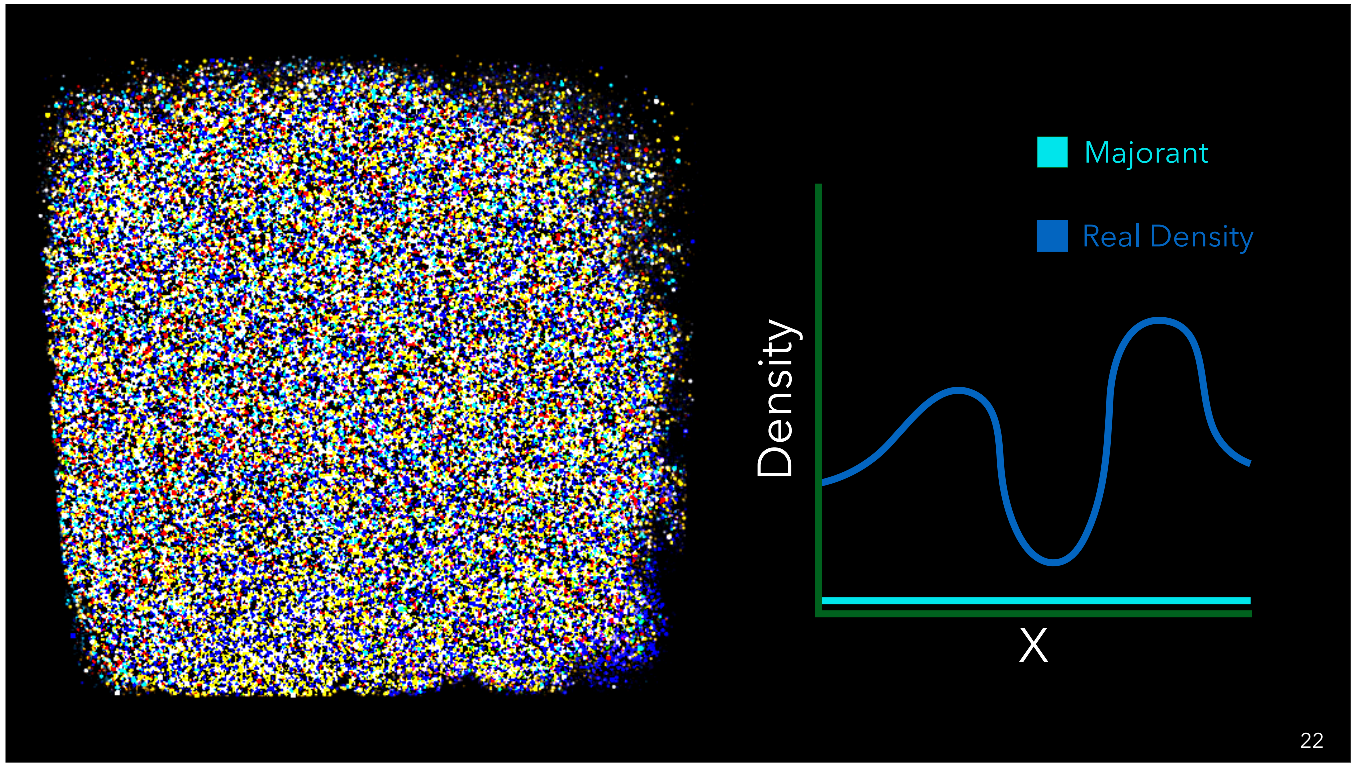



then the cost of our renders will increase, and may even become too costly to in production.

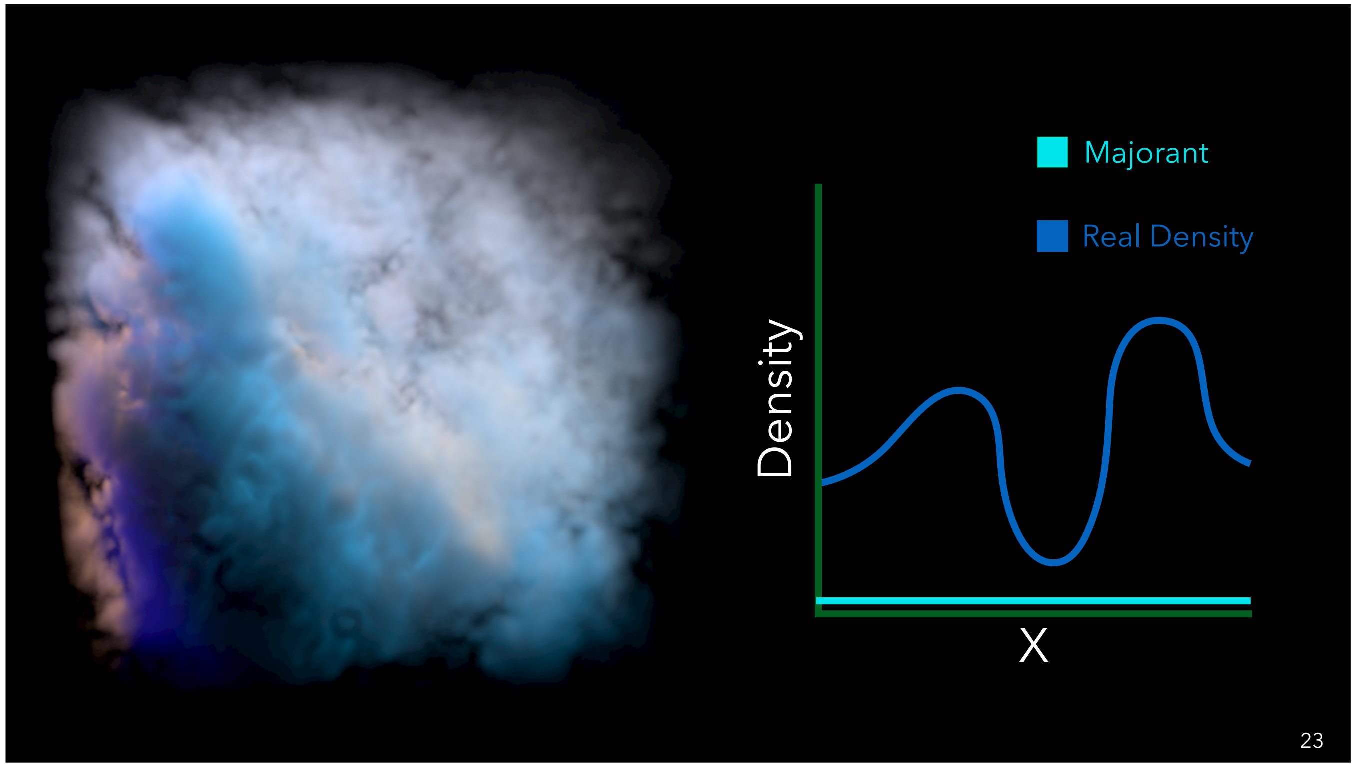



Alternatively, if the majorant is set too low (*click*) such that the majorant no longer bounds the density, the renders might become fast,

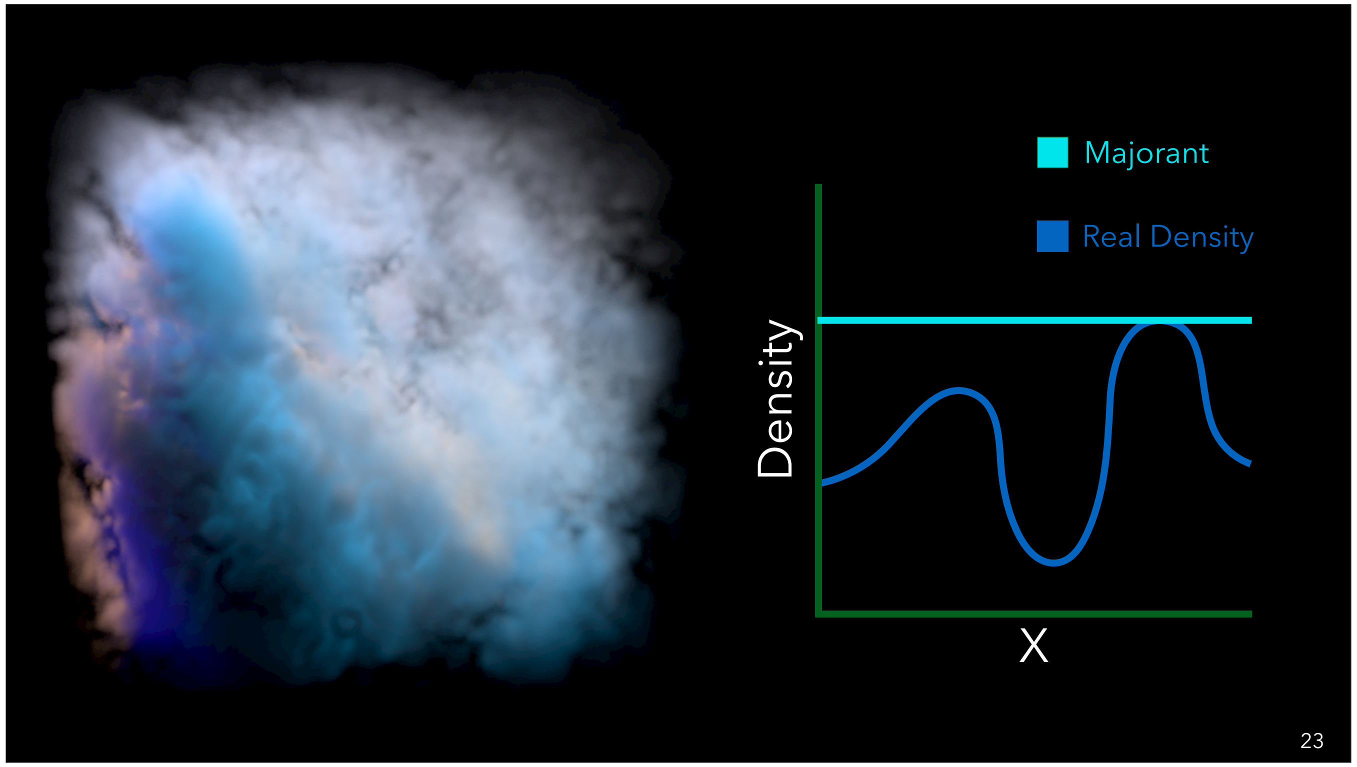

Majorant
Real Density
Density
X




But the variance of the renders will become so uncontrollable, that the renders will never converge in any reasonable amount of time. To have both low cost




and guarantee low variance, we ideally want a majorant which (*click*) bounds the density as tightly as possible. However, getting these tight majorants


Majorant
Real Density
Density
X

# Volume representation

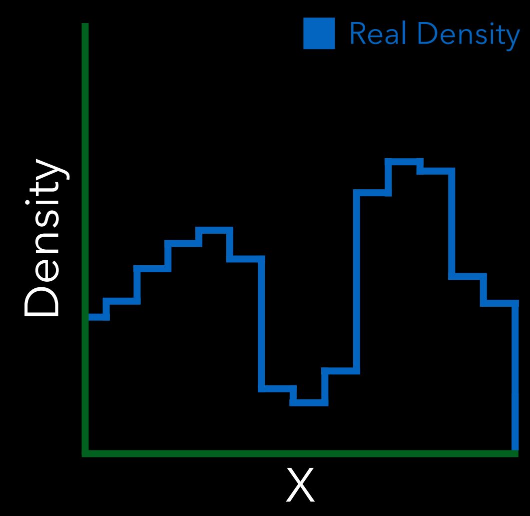


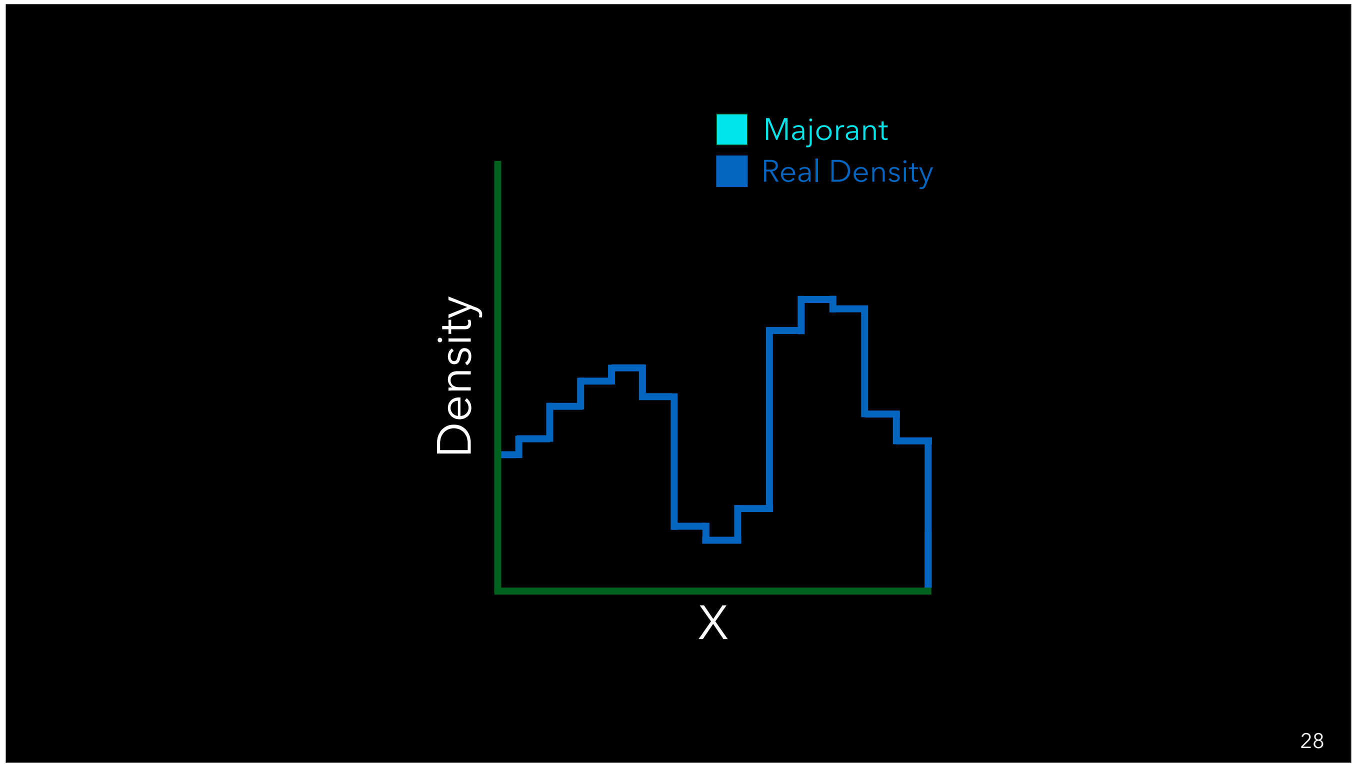
Depends on how we represent our volumes.


if we only render volumes like the Disney cloud,

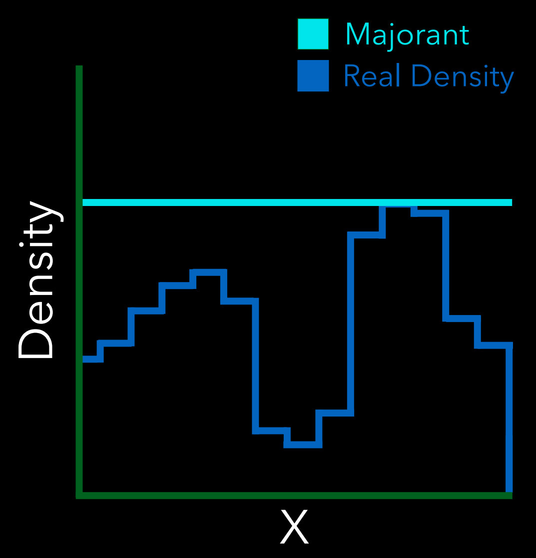

Which are explicitly stored as voxel density grids, then finding a tightly bounding majorant


Is as easy (click) as iterating through all of the voxels,

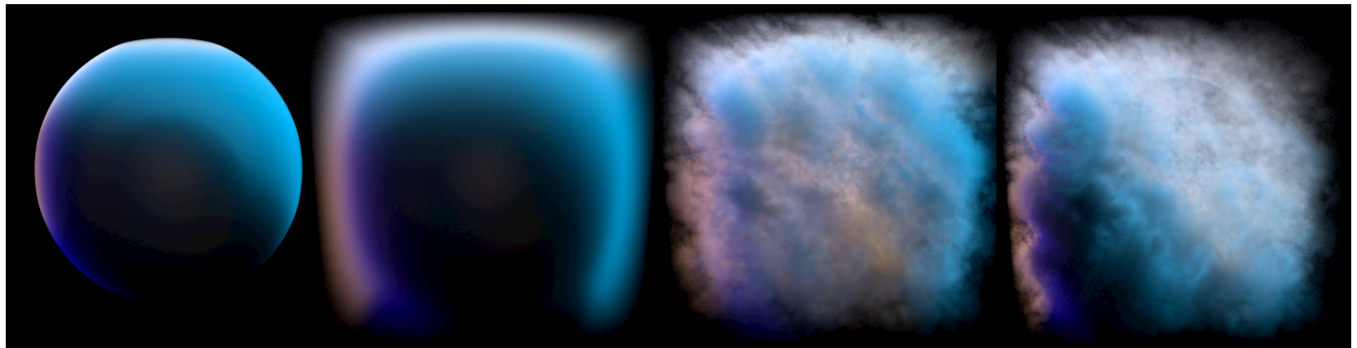

Real Density
Density
X

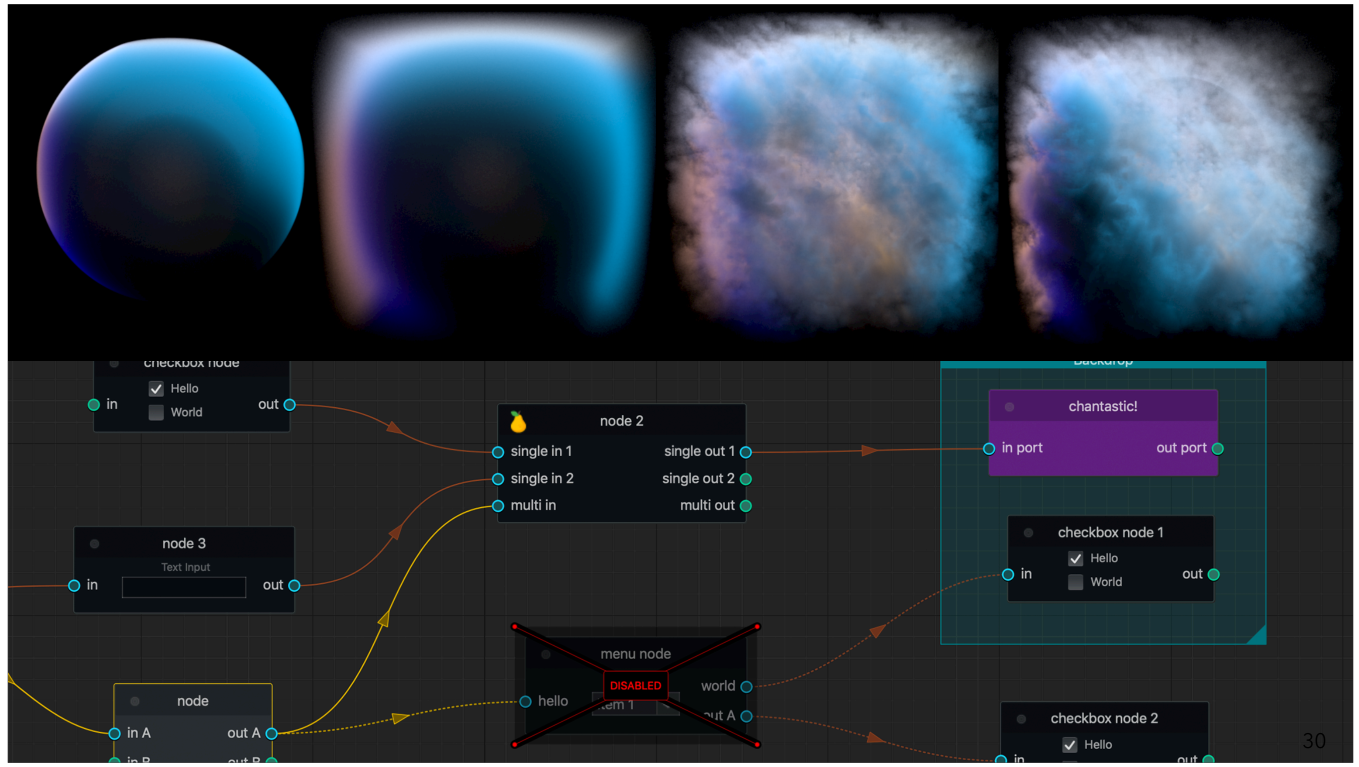



And setting the majorant to the largest found density. However,

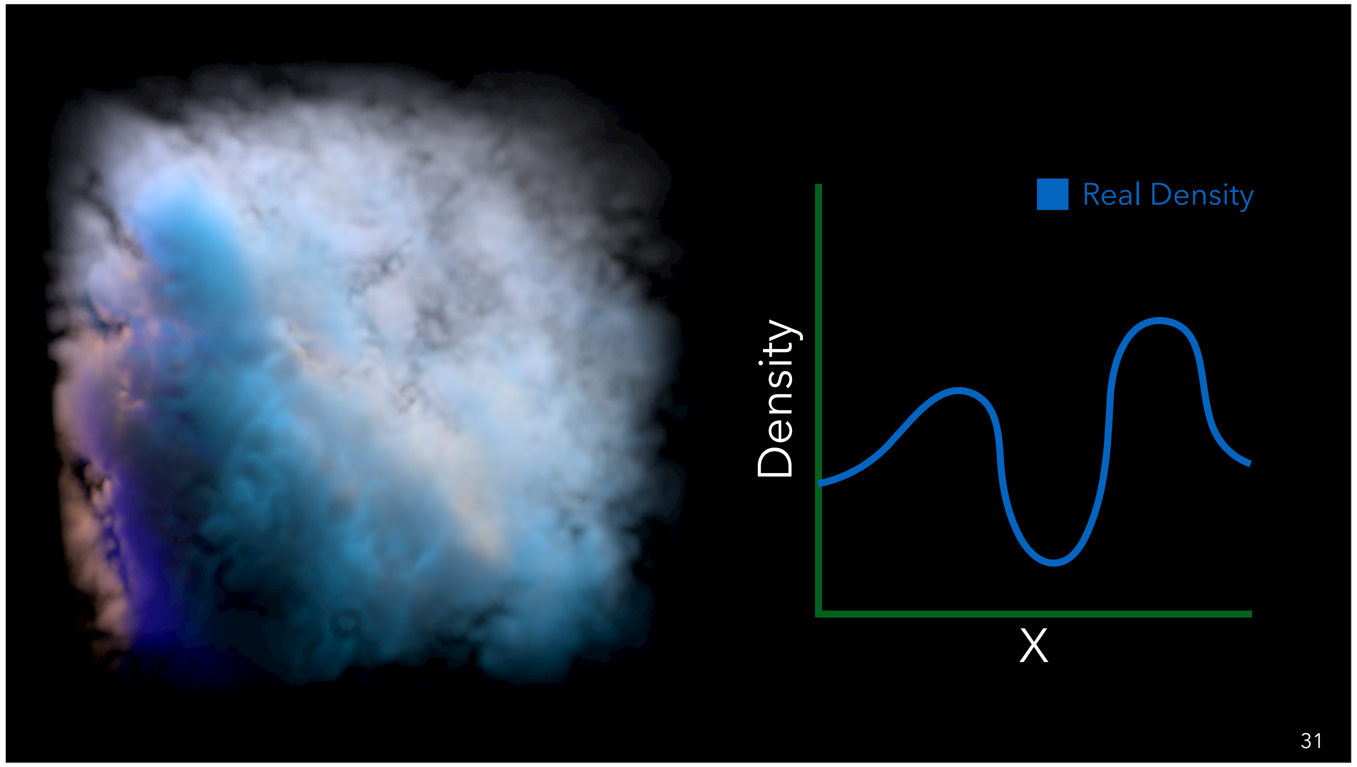

Majorant
Real Density
Density
X


when dealing with purely procedural volumes,

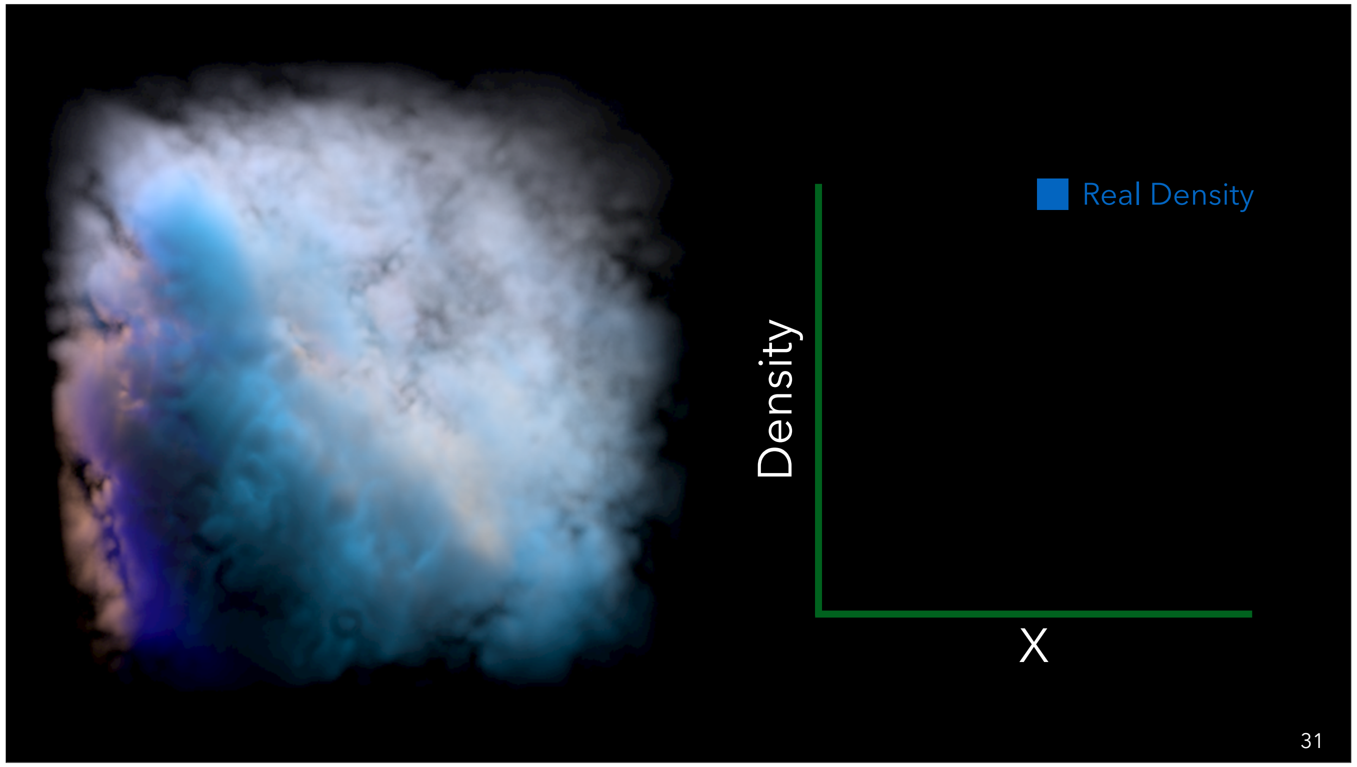


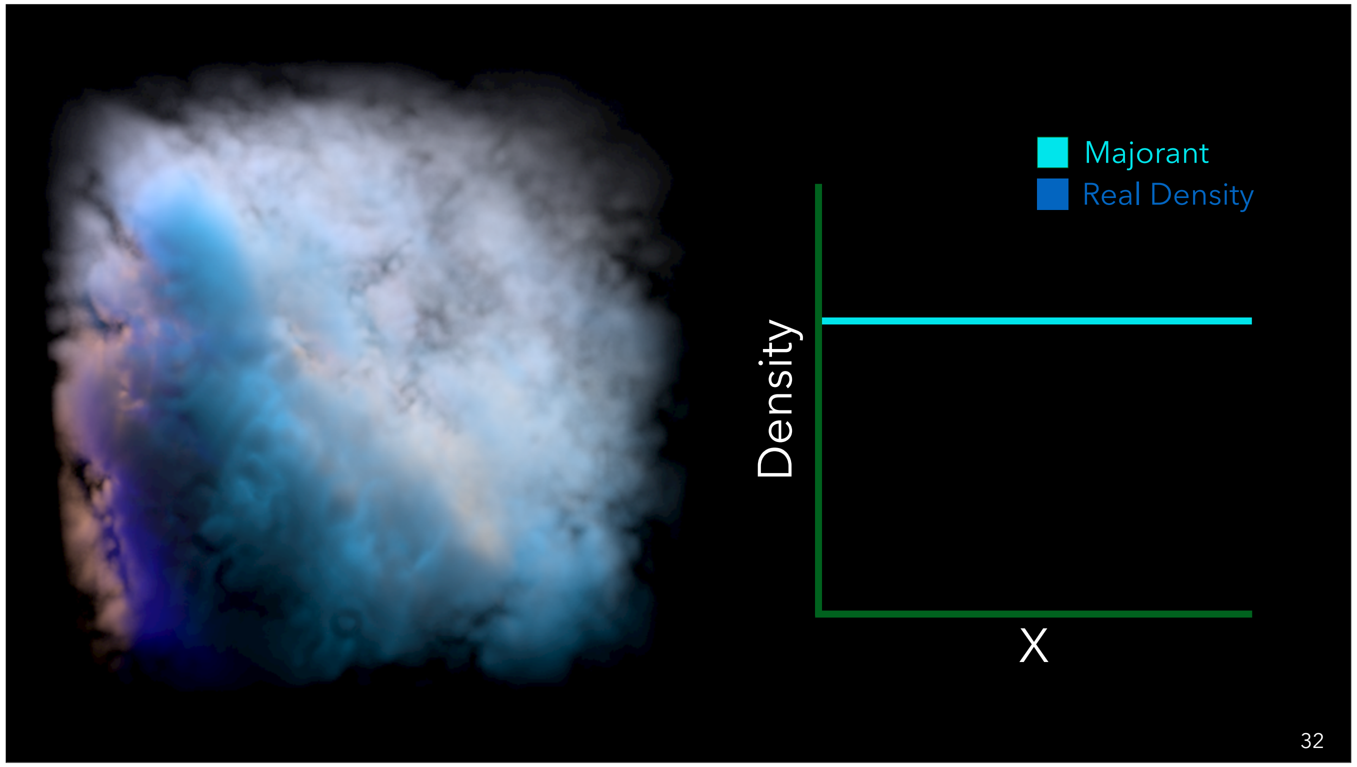



Or complicated production workflows,

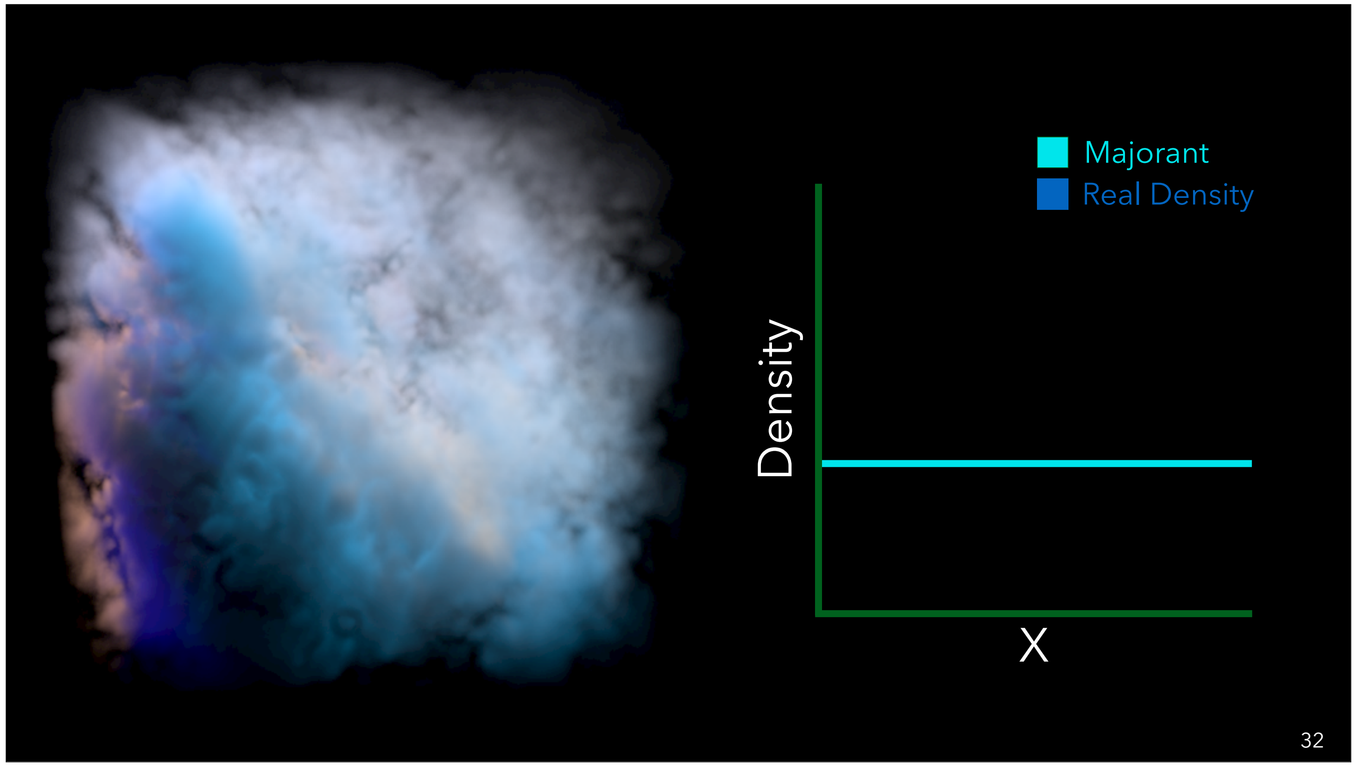


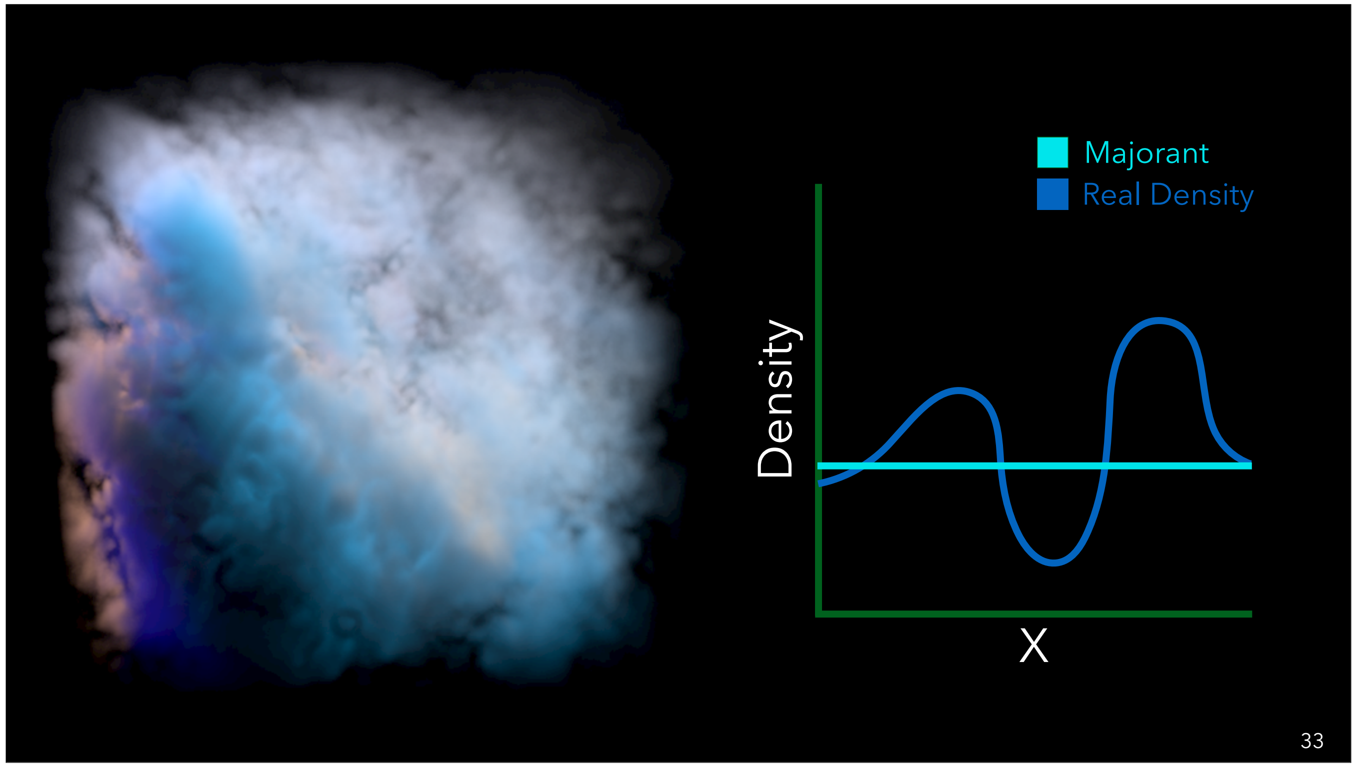

The underlying density (*click*) becomes a black box whose general shape is not known to the renderer.


Real Density
Density
X
31

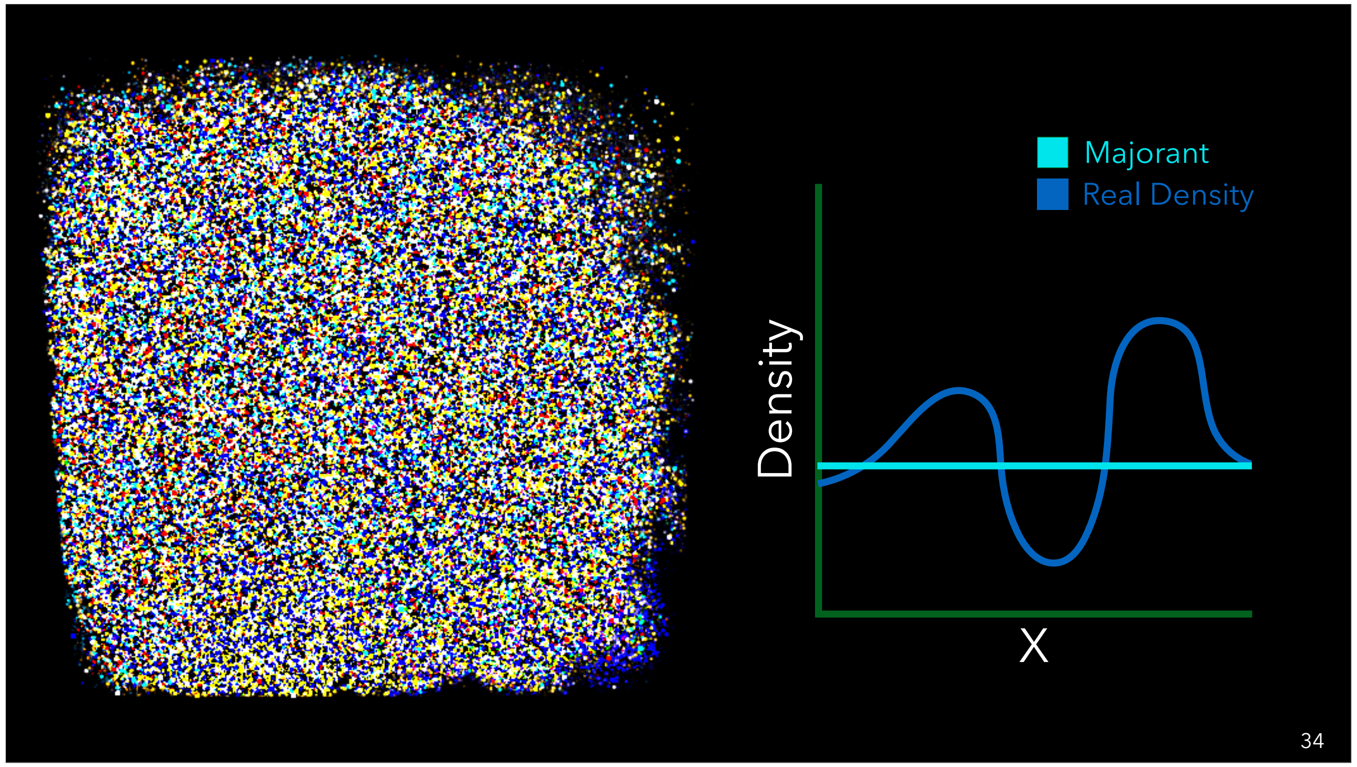



Any majorant which we specify, becomes an approximation for a bounding majorant,

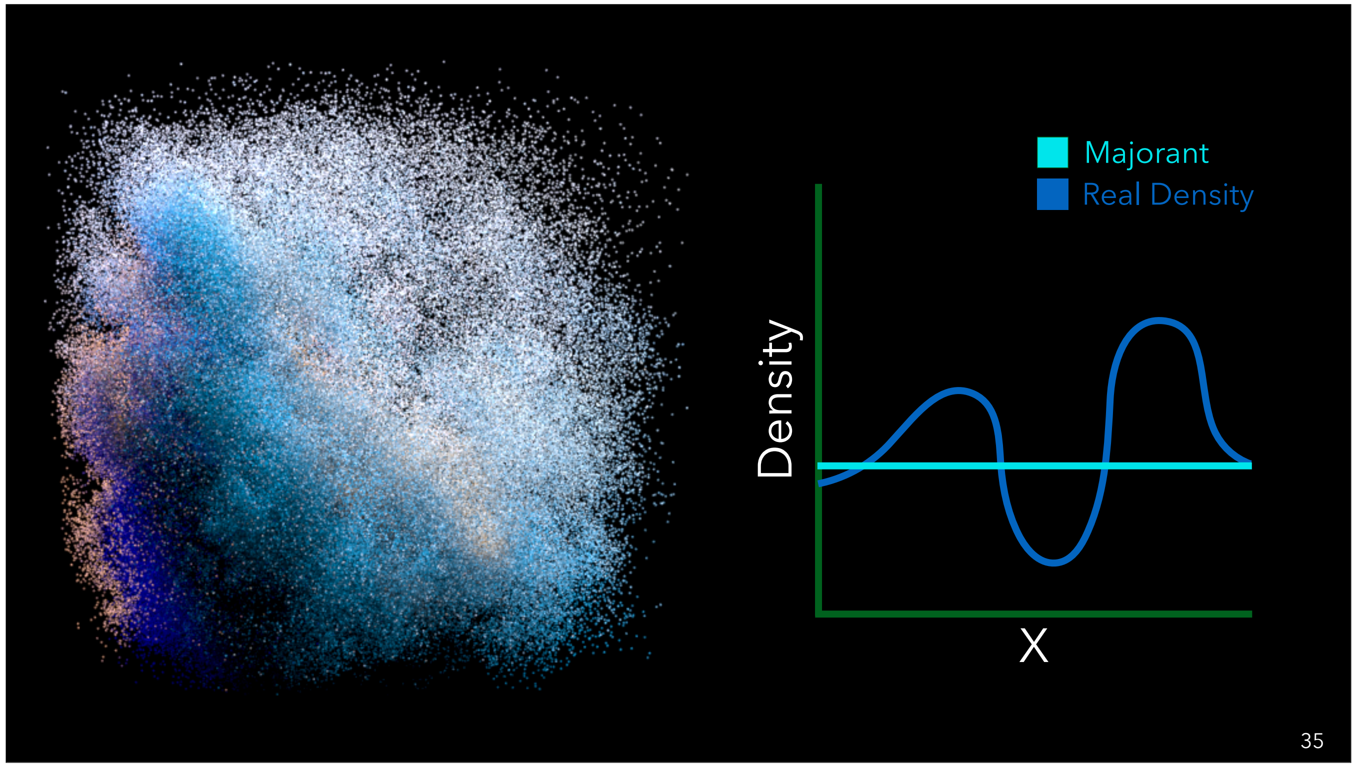

Majorant
Real Density
Density
X
32



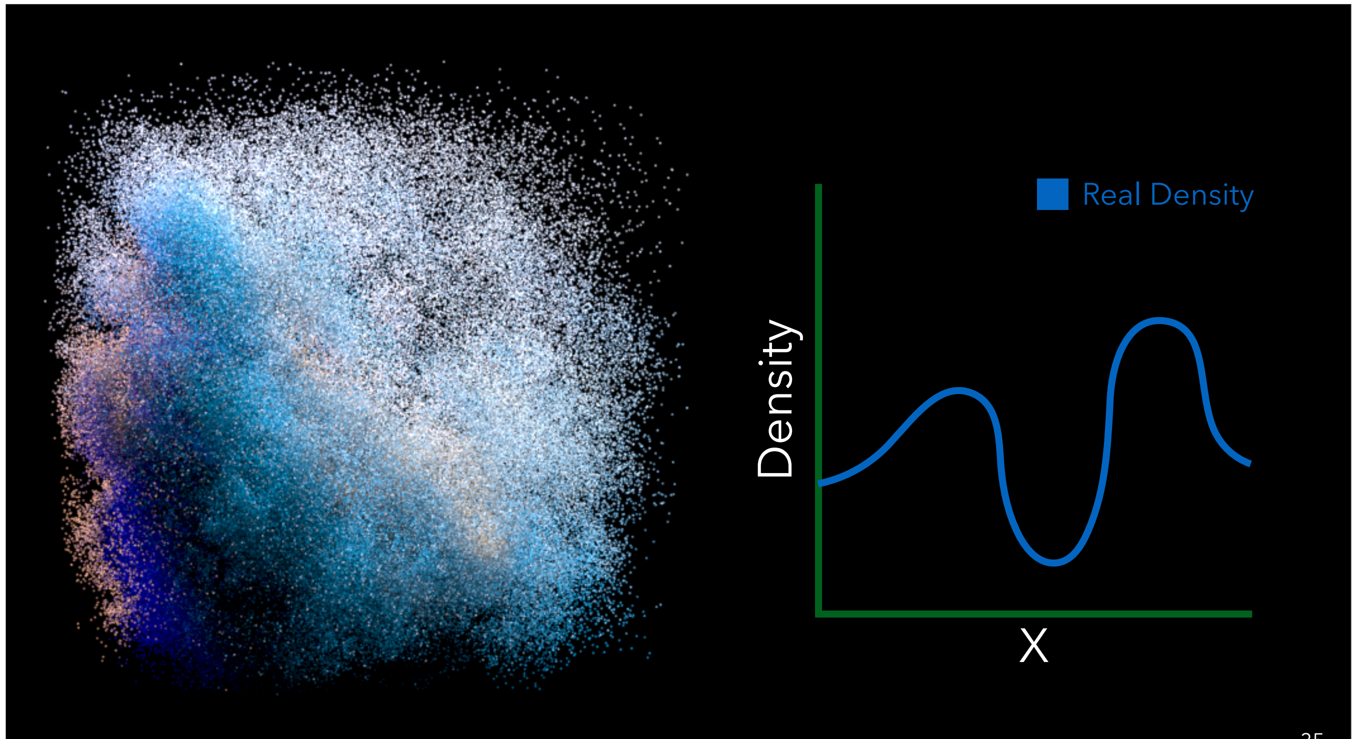

Becoming very difficult to guarantee that we will ever have a tightly bounding one. And, without a tightly bounding majorant we can’t be sure that we will avoid,

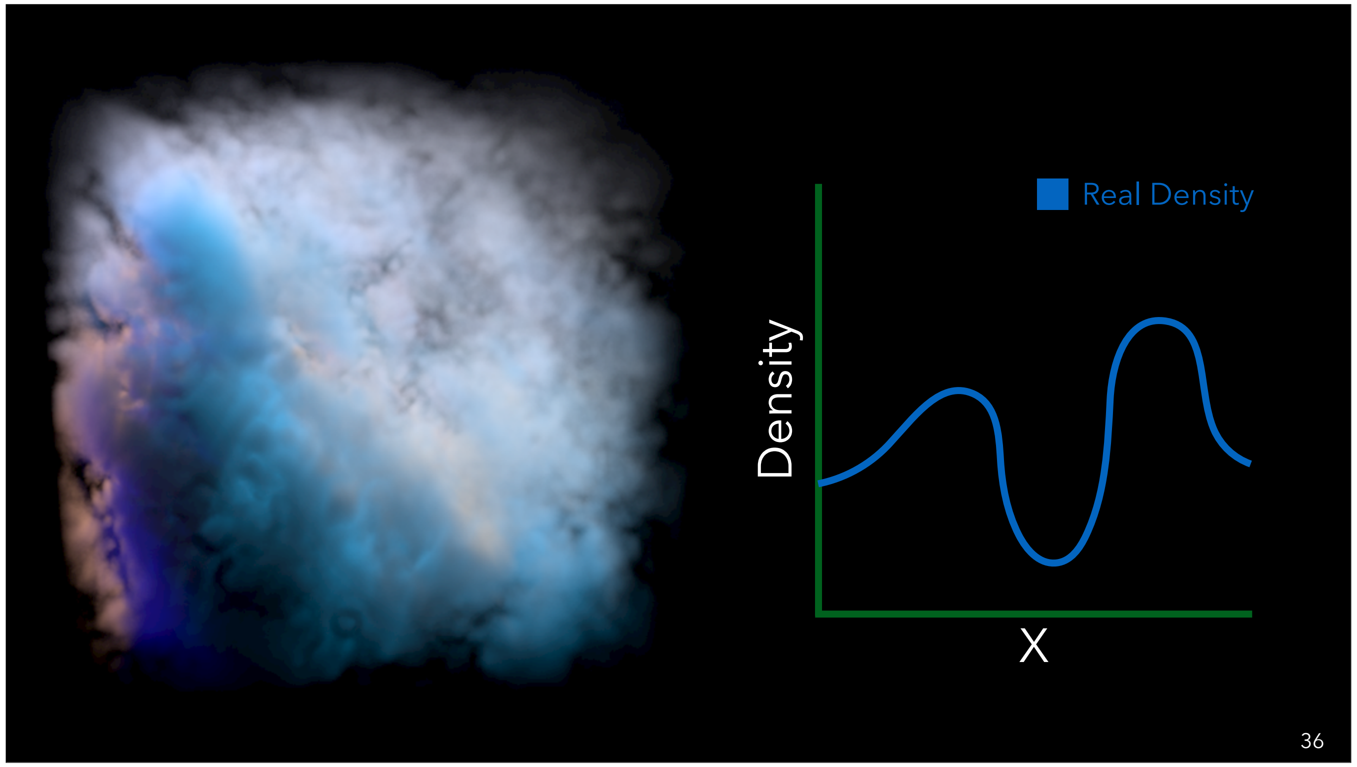



uncontrollable variance,

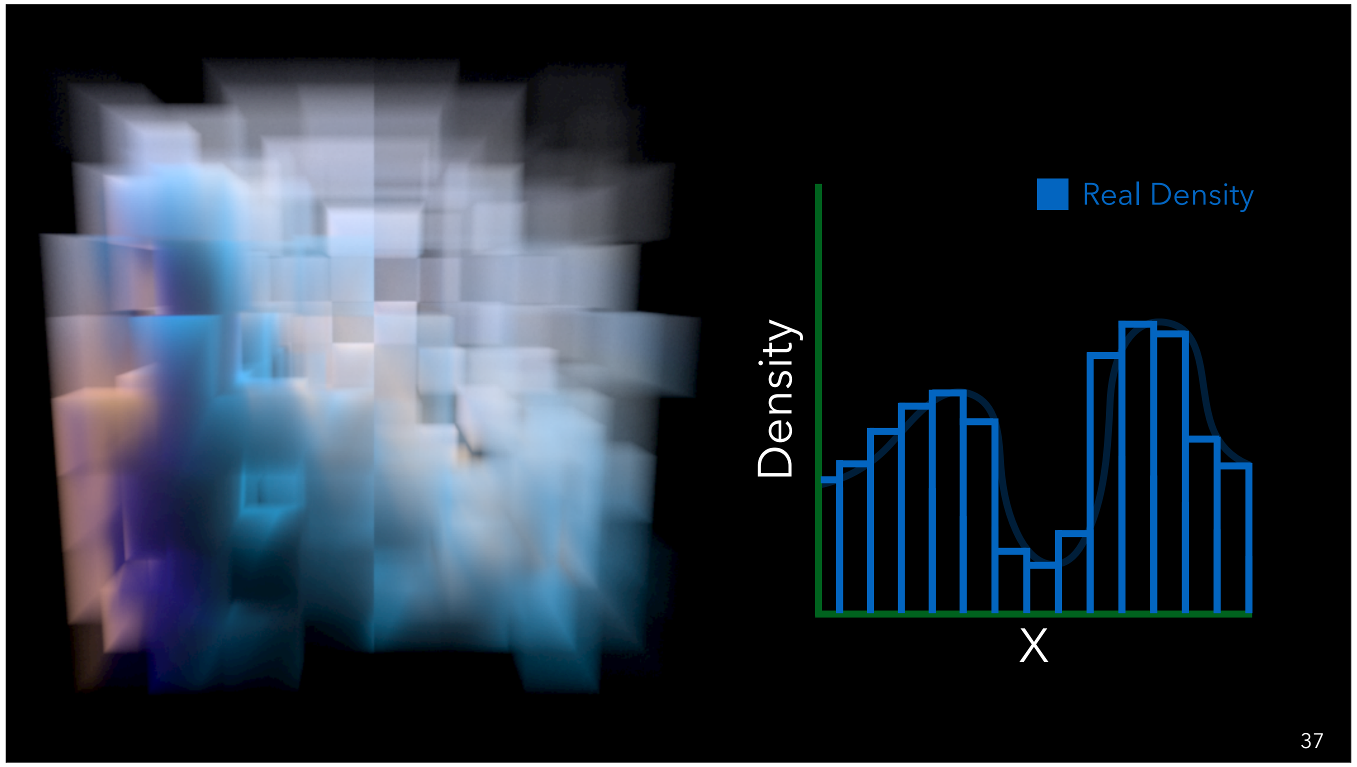



Or too inefficient of renders

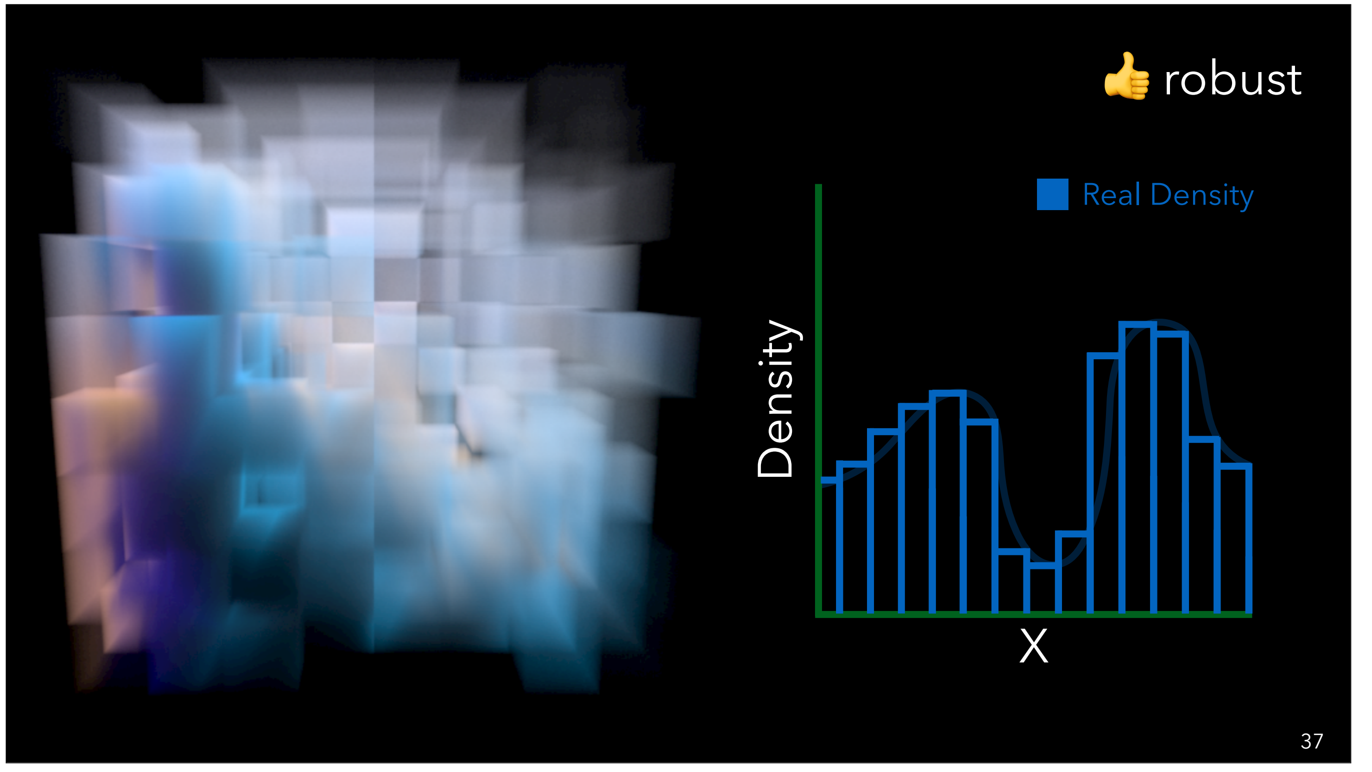

Real Density
Density
X



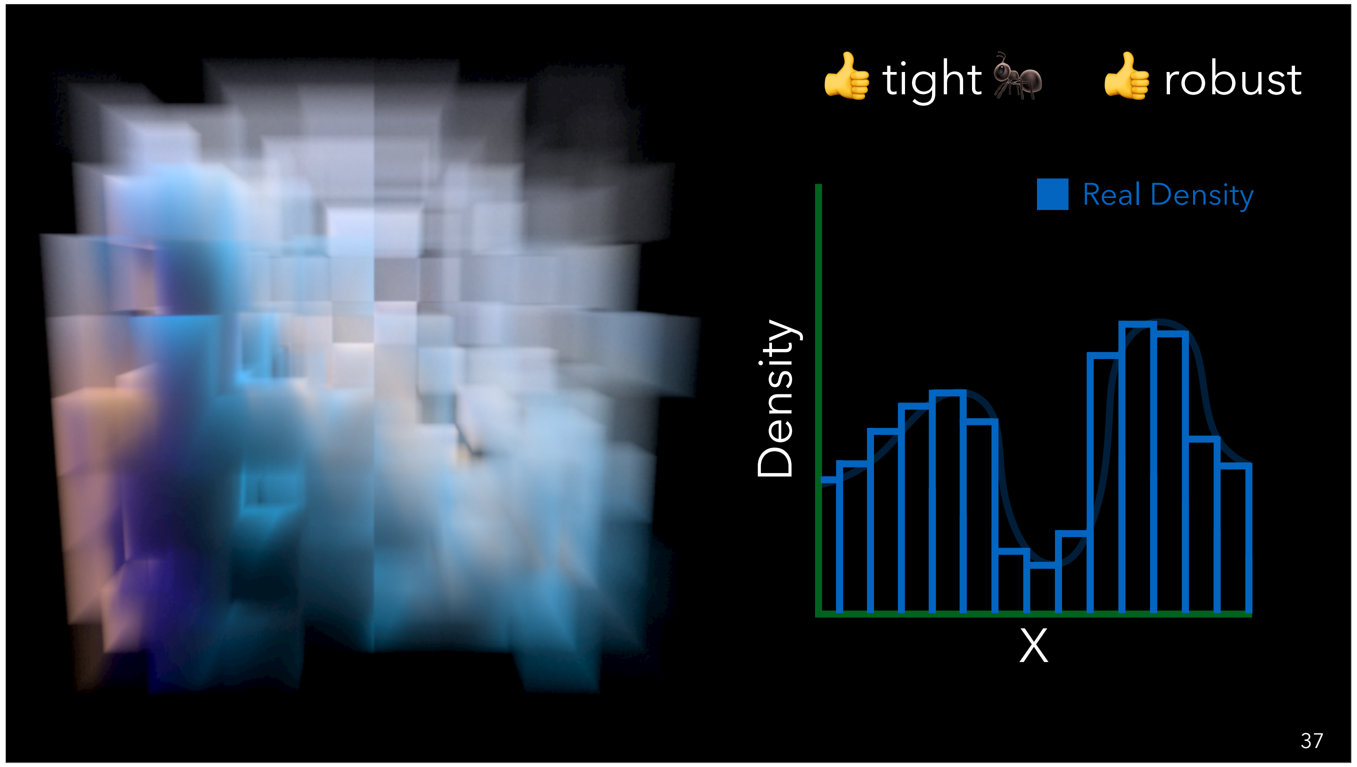

It is for this reason that Disney currently

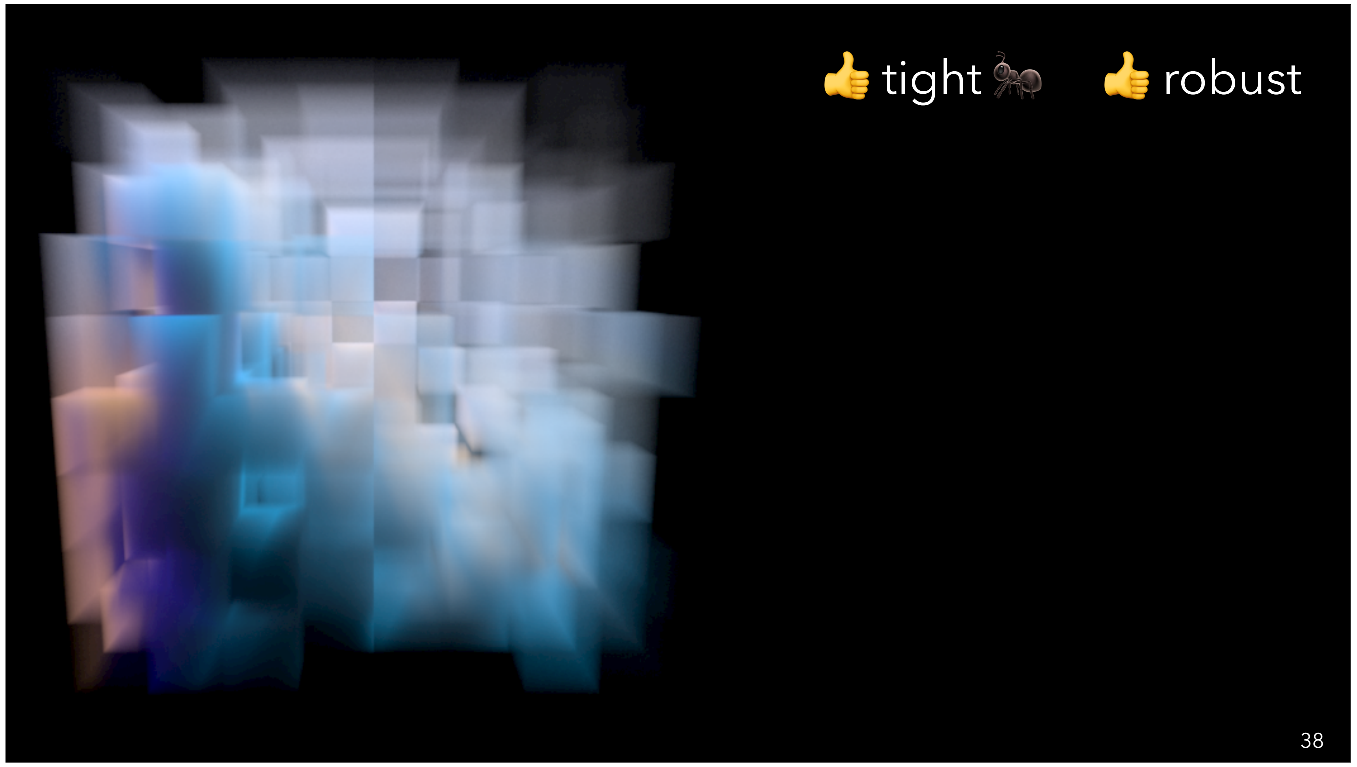



bakes all volumes into voxel density grids before rendering them. Representing all volumes as grids allows null-scattering techniques to be (*click*) robust, since, having a density grid (*click*) guarantees that tight majorants can always be found. However,

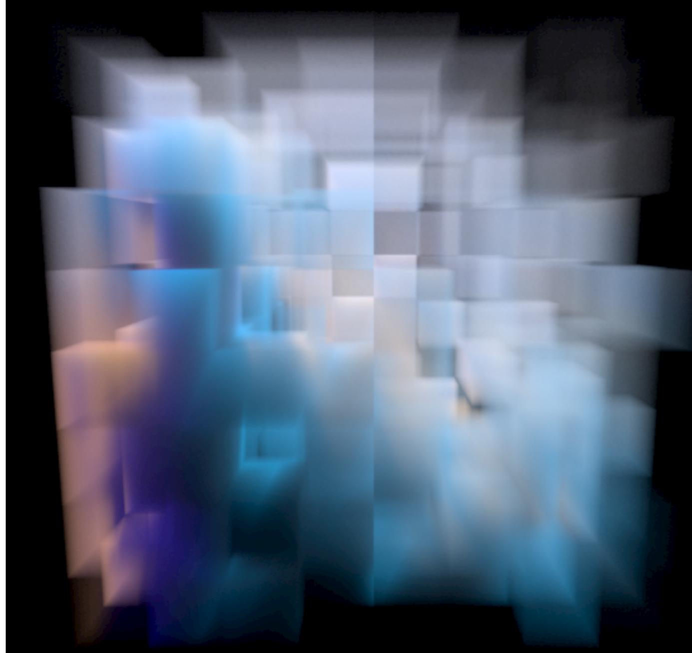


👍 robust
Real Density
Density
X


👍 tight 🐜 👍 robust
Real Density
Density
X
37

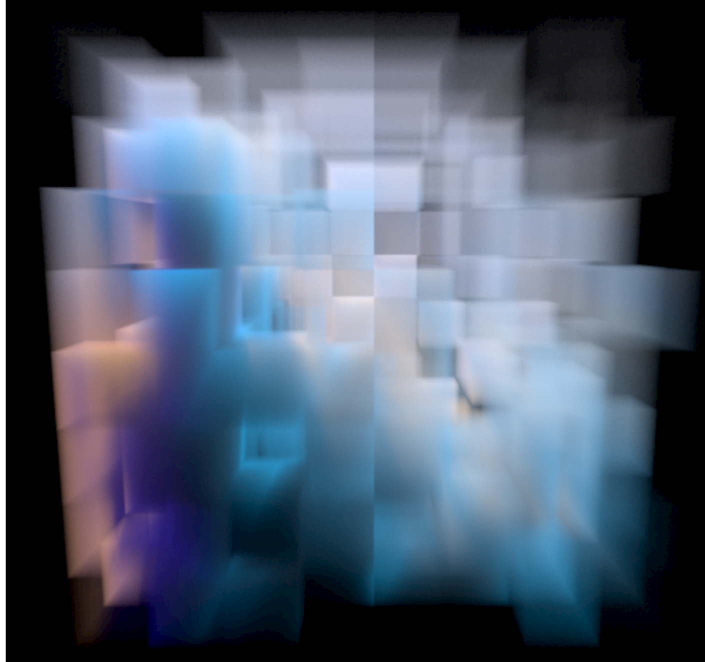



The process of baking reduces (*click*) the visual fidelity of all volumes, necessitates (*click*) preprocessing all volumes, and significantly increases (*click*) the storage requirements for all production scenes. We instead propose a solution which,



👍 tight 🐜 👍 robust

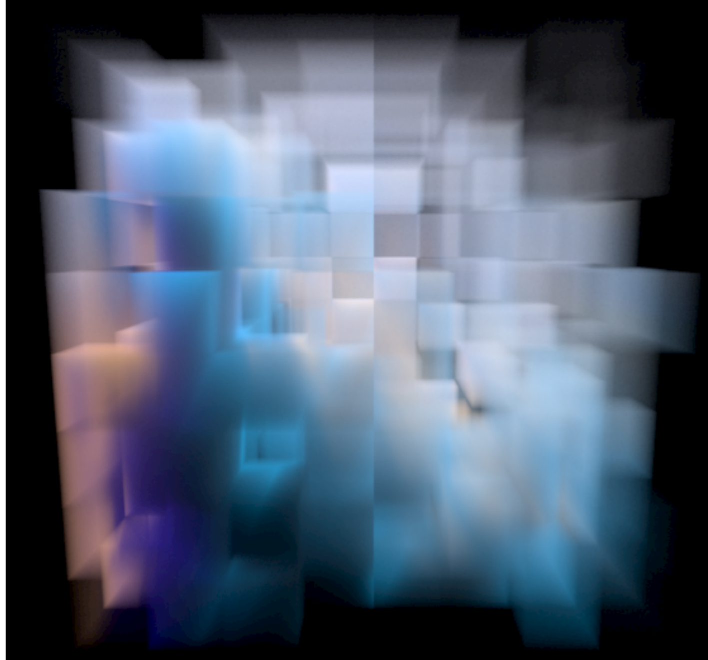
😢 visual quality



👍 tight 🐜 👍 robust

😢 visual quality

😢 preprocessing time

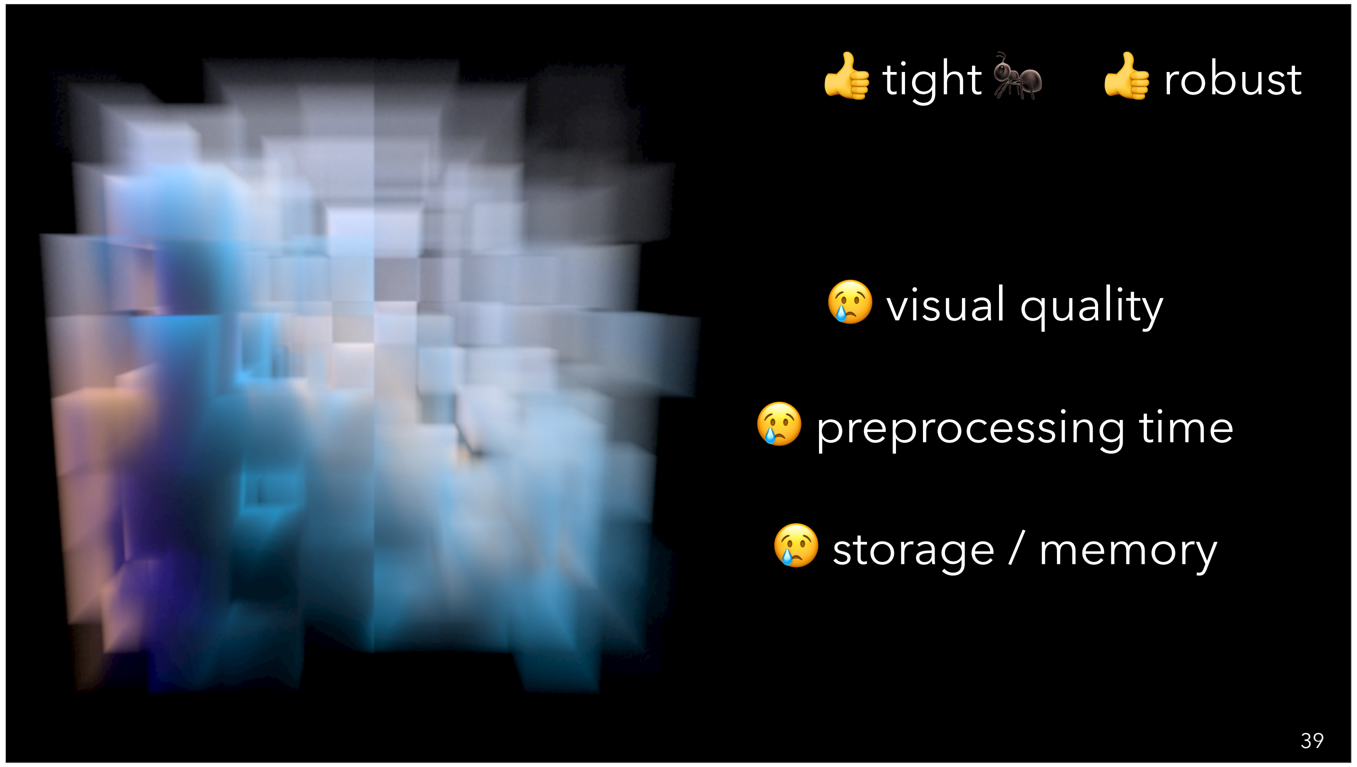


👍 tight 🐜 👍 robust

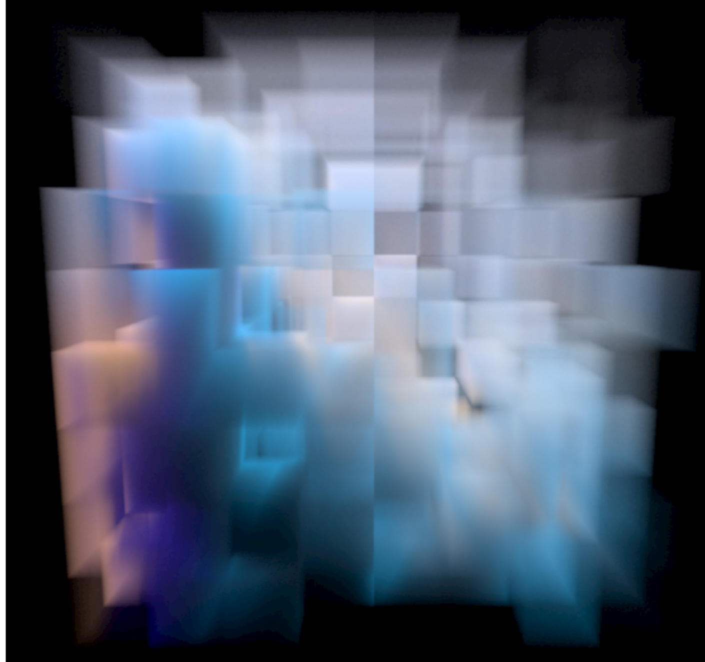
😢 visual quality

😢 preprocessing time

😢 storage / memory




Makes most null-scattering techniques resilient to non-bounding majorants. Our technique is (*click*) robust, discovers (*click*) tight majorants during render time,



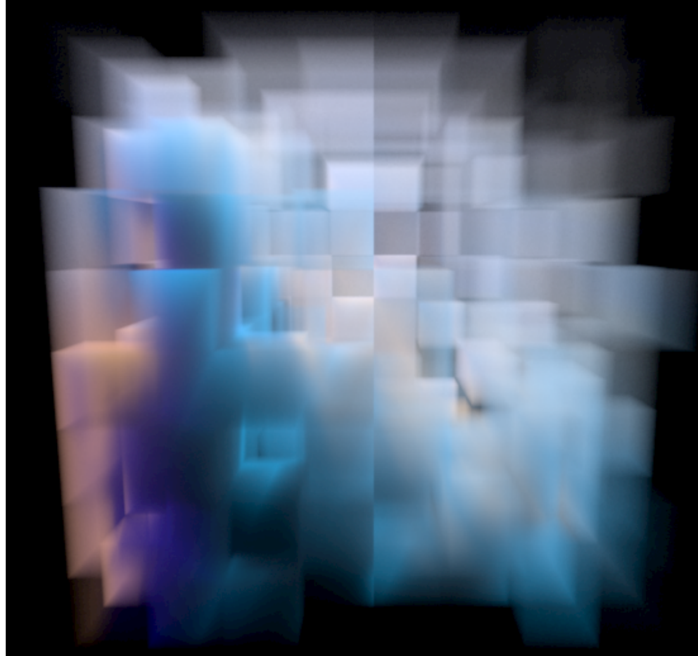
👍 tight 🐜 👍 robust

😄 works for any majorant

😢 visual quality

😢 preprocessing time

😢 storage / memory



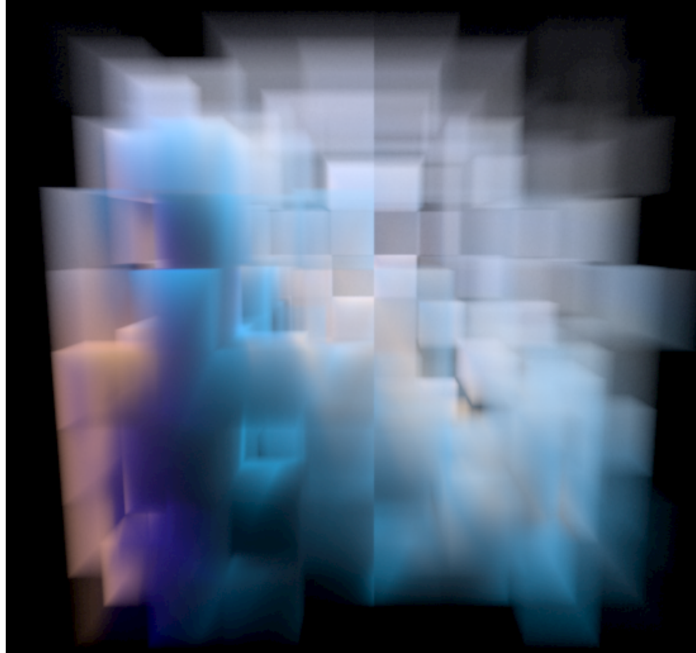
👍 tight 🐜 👍 robust

😄 works for any majorant

😢 visual quality

😢 preprocessing time

😢 storage / memory



👍 tight 🐜 👍 robust

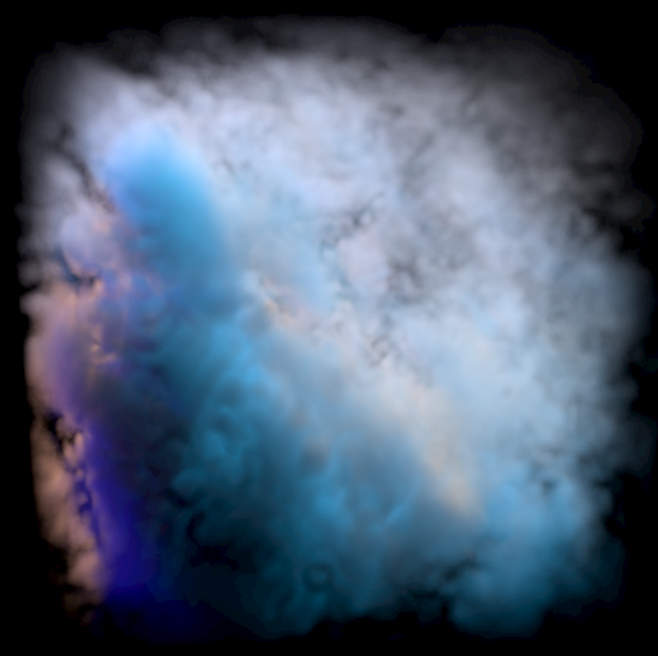
😁 works for any majorant

😢 visual quality

😢 preprocessing time

😢 storage / memory


maintains the same visual quality as using bounding majorants in the converged renders,



👍 tight 🐜 👍 robust

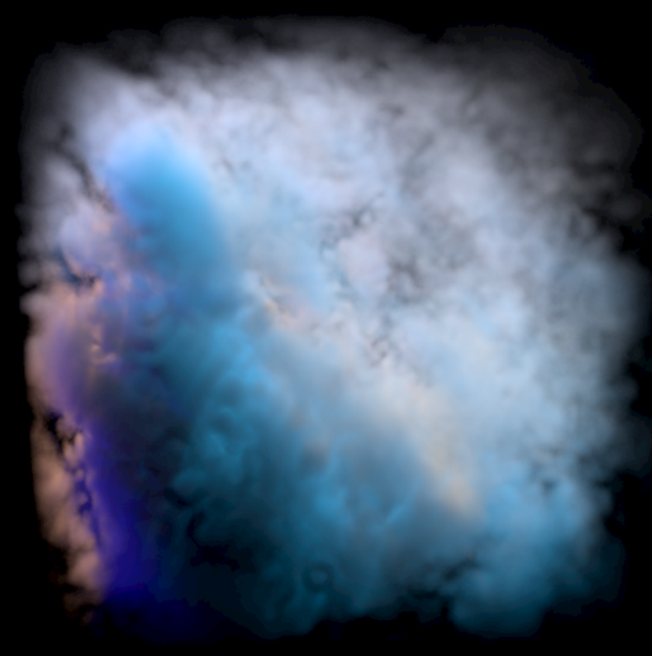
😄 works for any majorant

😄 visual quality

😢 preprocessing time

😢 storage / memory


require little to no preprocessing time



👍 tight 🐜 👍 robust

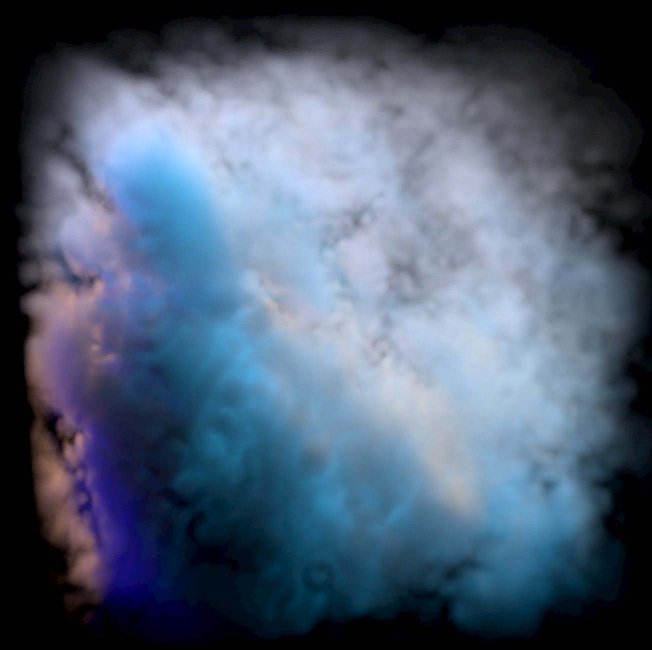
😁 works for any majorant

😁 visual quality

😁 preprocessing time

😢 storage / memory


And requires no extra storage on top of what is already needed in any practical implementation of null-scattering. However, in return for fixing all these prior issues,



👍 tight 🐜 👍 robust

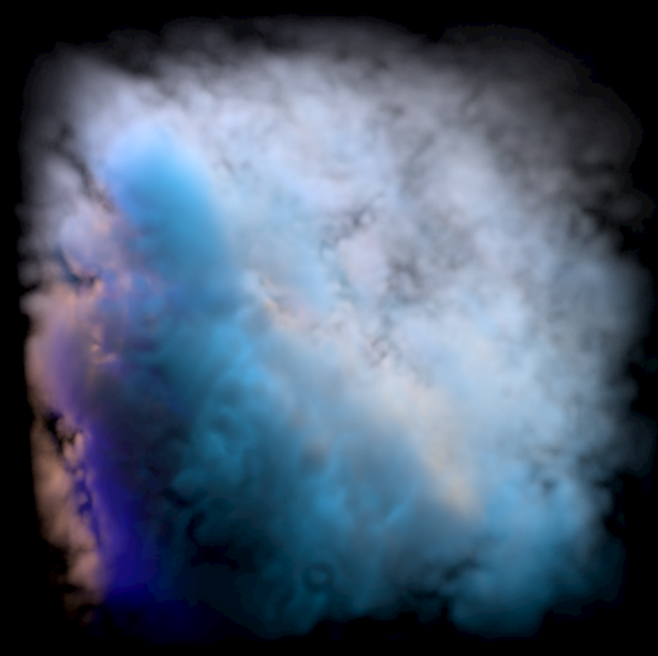
😁 works for any majorant

😁 visual quality

😁 preprocessing time

😁 storage / memory


Our technique requires that we relax the (*click*) unbiased property of most existing methods,



👍 tight 🐜 👍 robust

😁 works for any majorant

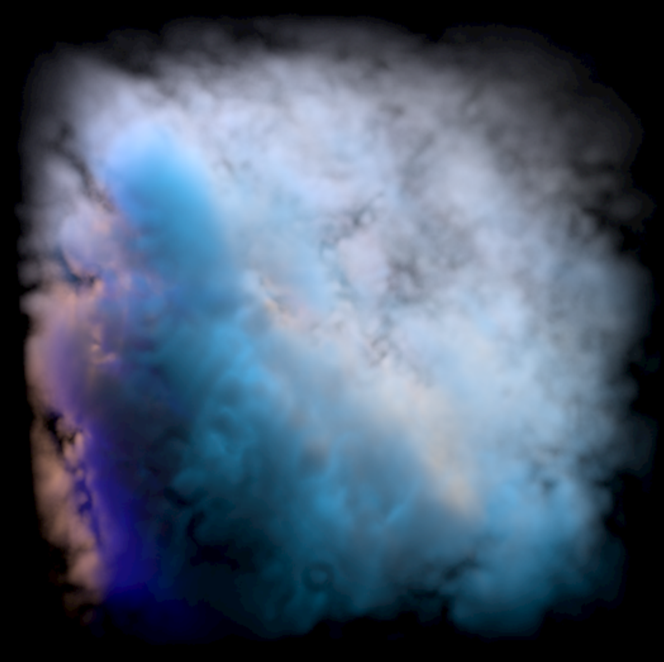
😁 visual quality

😁 preprocessing time

😁 storage / memory

😁 unbiased


To instead settle for only being consistent.



👍 tight 🐜 👍 robust

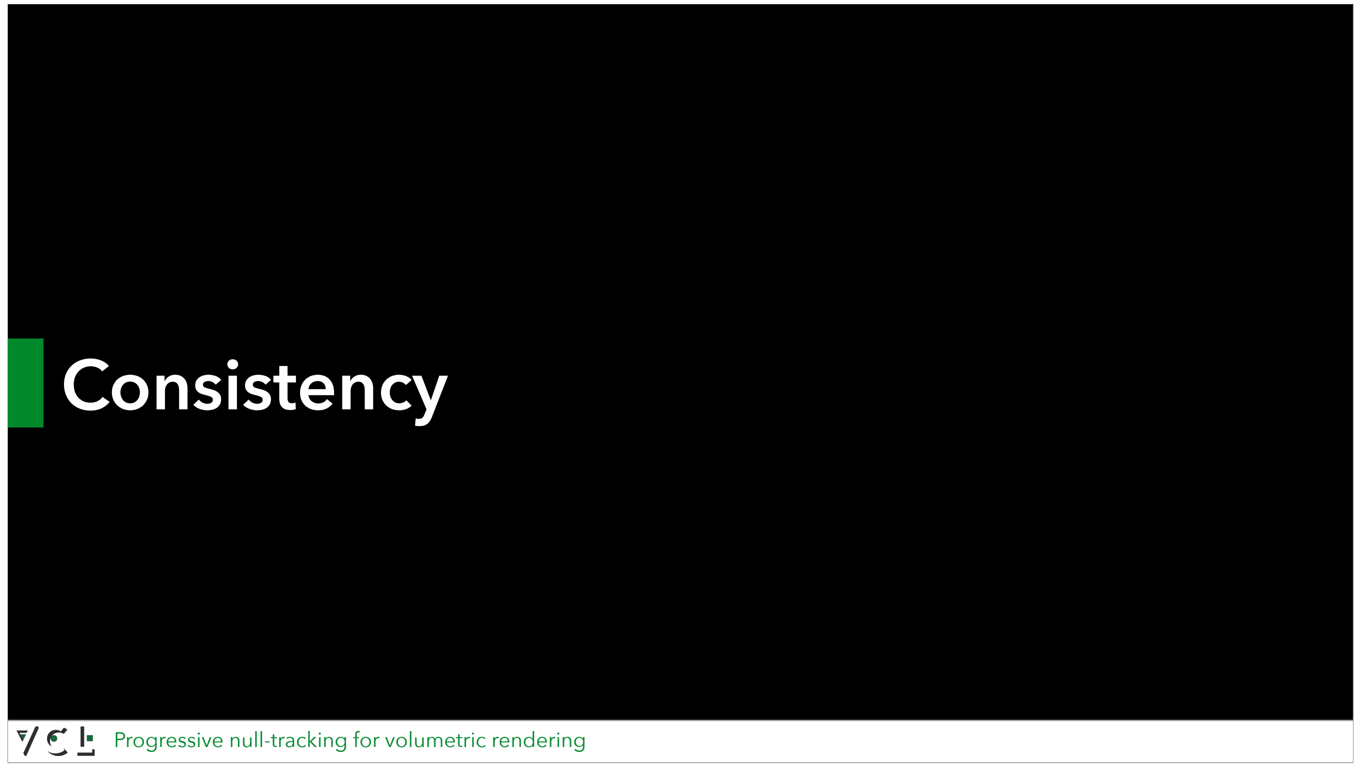
😁 works for any majorant

😁 visual quality

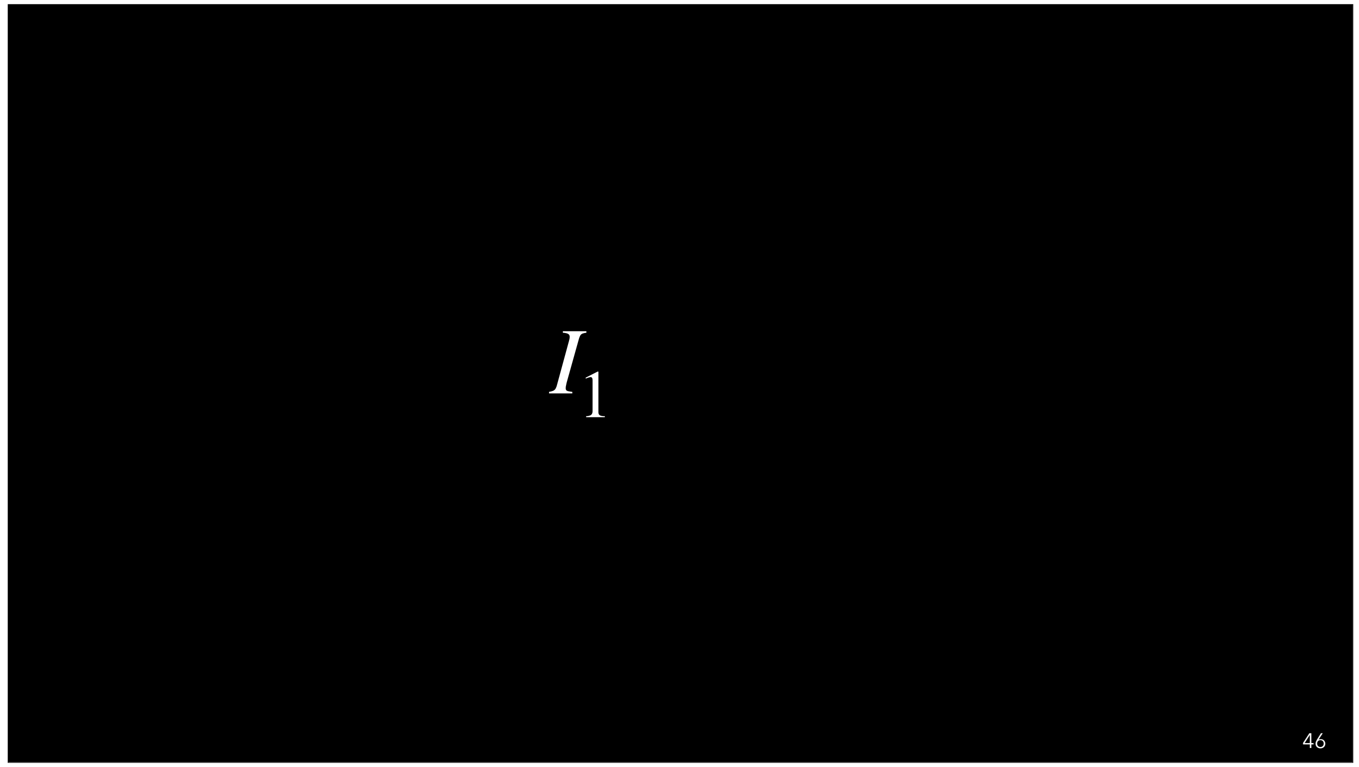
😁 preprocessing time

😁 storage / memory

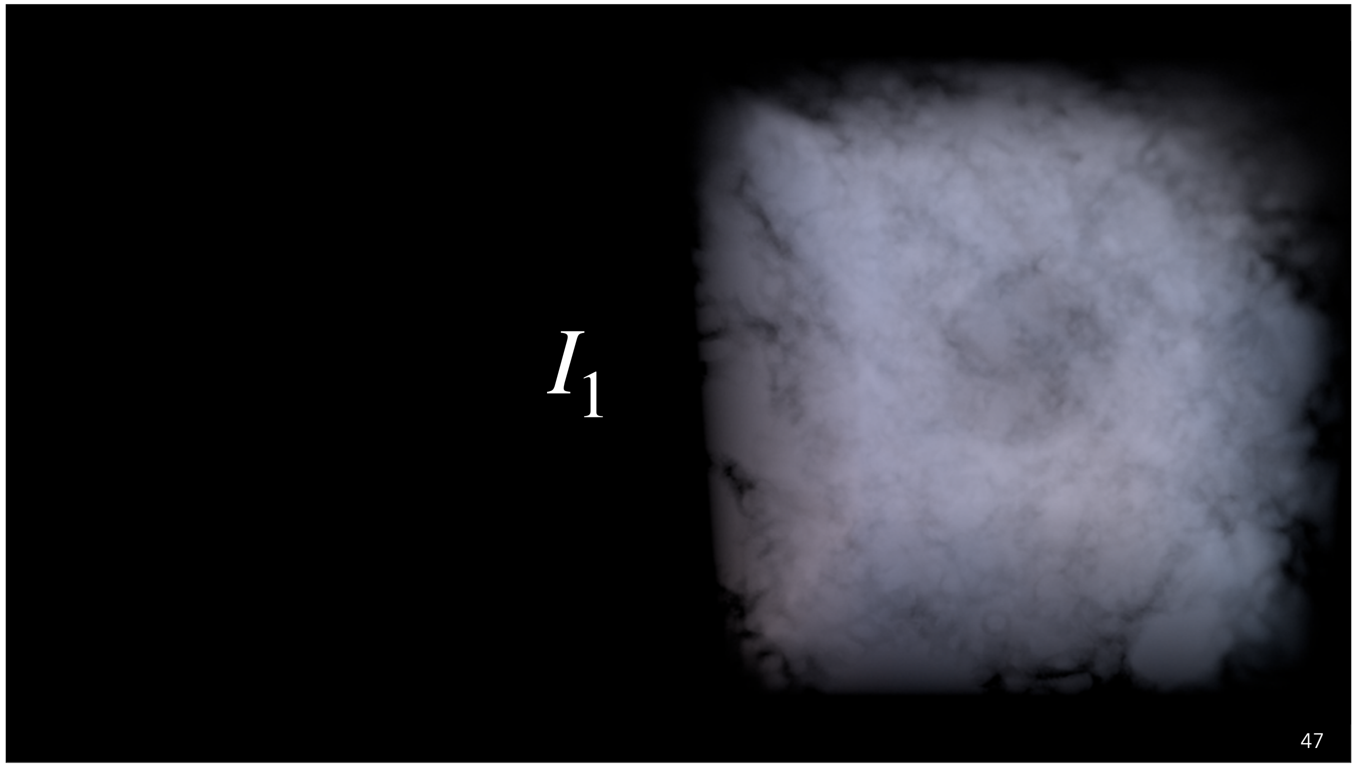
😐 consistent



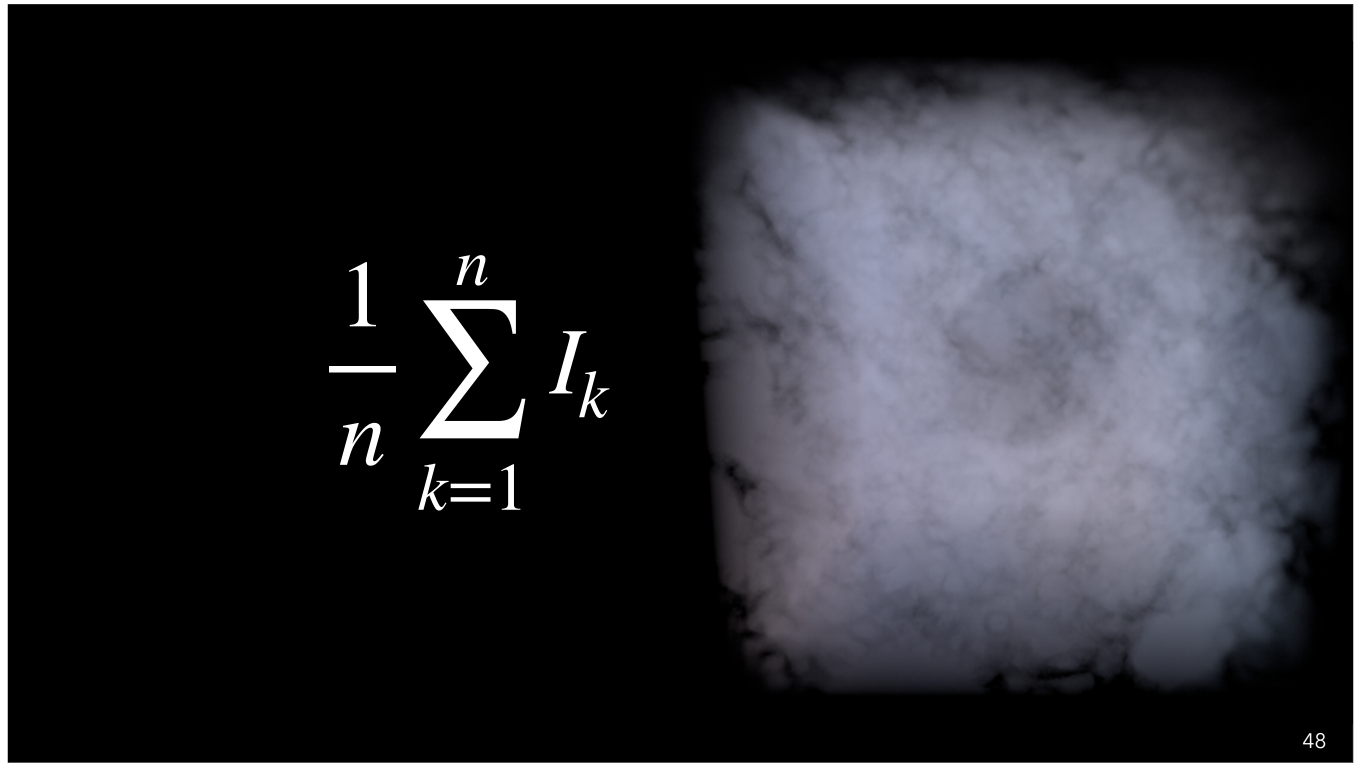

Now, let us define what we mean by consistency.



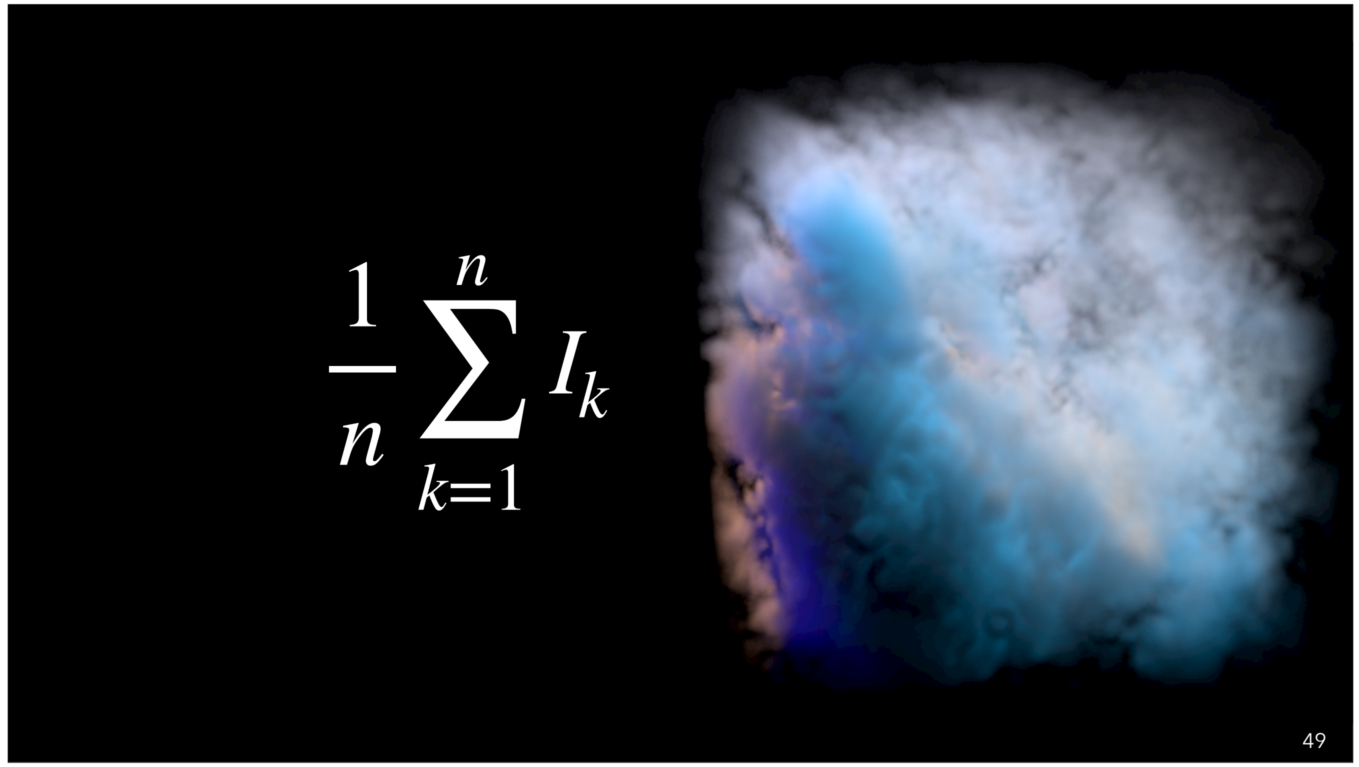

Let’s represent the first pixel sample in a render as I of 1, and let’s also assume that



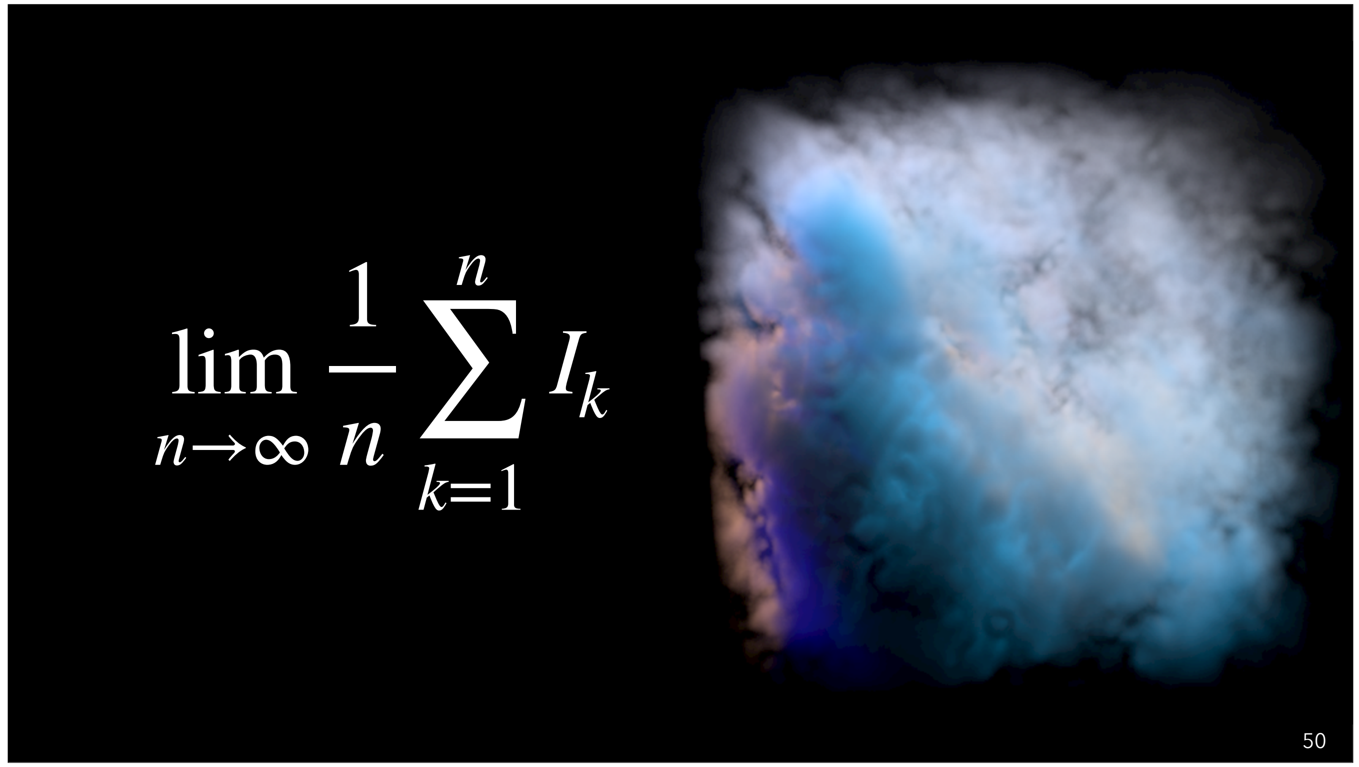

This first pixel sample is very biased.




A full render effectively takes the average across many different pixel samples. A consistent algorithm

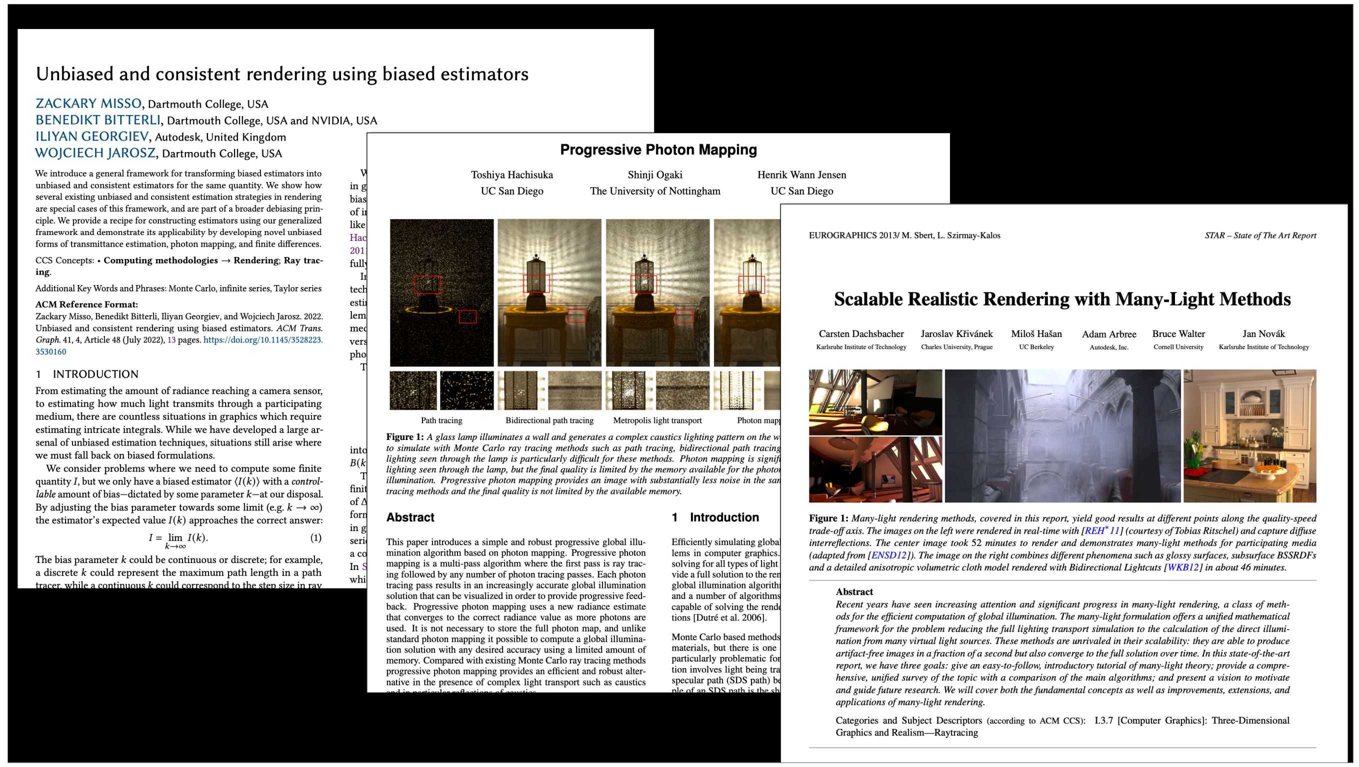



Is one which will guarantees that our render will eventually converge to the real solution,




in the limit. Regardless of how many individual pixel samples are biased.

Biased but consistent algorithms have appeared throughout graphics from photon mapping to many-light methods. And one idea, which has been used previously by virtual point lights,



# Unbiased and consistent rendering using biased estimators

ZACKARY MISSO, Dartmouth College, USA
BENEDIKT BITTERLI, Dartmouth College, USA and NVIDIA, USA
ILIYAN GEORGIEV, Autodesk, United Kingdom
WOJCIECH JAROSZ, Dartmouth College, USA

We introduce a general framework for transforming biased estimators into unbiased and consistent estimators for the same quantity. We show how several existing unbiased and consistent estimation strategies in rendering are special cases of this framework, and are part of a broader debiasing principle. We provide a recipe for constructing estimators using our generalized framework and demonstrate its applicability by developing novel unbiased forms of transmittance estimation, photon mapping, and finite differences.

CCS Concepts: • **Computing methodologies → Rendering; Ray tracing.**

Additional Key Words and Phrases: Monte Carlo, infinite series, Taylor series

**ACM Reference Format:**
Zackary Misso, Benedikt Bitterli, Iliyan Georgiev, and Wojciech Jarosz. 2022. Unbiased and consistent rendering using biased estimators. *ACM Trans. Graph.* 41, 4, Article 48 (July 2022), 13 pages. https://doi.org/10.1145/3528223.3530160

## 1 INTRODUCTION

From estimating the amount of radiance reaching a camera sensor, to estimating how much light transmits through a participating medium, there are countless situations in graphics which require estimating intricate integrals. While we have developed a large arsenal of unbiased estimation techniques, situations still arise where we must fall back on biased formulations.

We consider problems where we need to compute some finite quantity $I$, but we only have a biased estimator $\langle I(k)\rangle$ with a *controllable* amount of bias—dictated by some parameter $k$—at our disposal. By adjusting the bias parameter towards some limit (e.g. $k \to \infty$) the estimator's expected value $I(k)$ approaches the correct answer:

$$I = \lim_{k\to\infty} I(k). \quad (1)$$

The bias parameter $k$ could be continuous or discrete; for example, a discrete $k$ could represent the maximum path length in a path tracer, while a continuous $k$ could correspond to the step size in ray

# Progressive Photon Mapping

Toshiya Hachisuka, UC San Diego
Shinji Ogaki, The University of Nottingham
Henrik Wann Jensen, UC San Diego

Path tracing | Bidirectional path tracing | Metropolis light transport | Photon map

**Figure 1:** *A glass lamp illuminates a wall and generates a complex caustics lighting pattern on the w… to simulate with Monte Carlo ray tracing methods such as path tracing, bidirectional path tracing… lighting seen through the lamp is particularly difficult for these methods. Photon mapping is signif… lighting seen through the lamp, but the final quality is limited by the memory available for the photo… illumination. Progressive photon mapping provides an image with substantially less noise in the sa… tracing methods and the final quality is not limited by the available memory.*

## Abstract

This paper introduces a simple and robust progressive global illumination algorithm based on photon mapping. Progressive photon mapping is a multi-pass algorithm where the first pass is ray tracing followed by any number of photon tracing passes. Each photon tracing pass results in an increasingly accurate global illumination solution that can be visualized in order to provide progressive feedback. Progressive photon mapping uses a new radiance estimate that converges to the correct radiance value as more photons are used. It is not necessary to store the full photon map, and unlike standard photon mapping it possible to compute a global illumination solution with any desired accuracy using a limited amount of memory. Compared with existing Monte Carlo ray tracing methods progressive photon mapping provides an efficient and robust alternative in the presence of complex light transport such as caustics

## 1 Introduction

Efficiently simulating global… lems in computer graphics… solving for all types of light… vide a full solution to the ren… global illumination algorith… and a number of algorithms… capable of solving the rend… tions [Dutré et al. 2006].

EUROGRAPHICS 2013/ M. Sbert, L. Szirmay-Kalos — *STAR – State of The Art Report*

# Scalable Realistic Rendering with Many-Light Methods

Carsten Dachsbacher, Karlsruhe Institute of Technology
Jaroslav Křivánek, Charles University, Prague
Miloš Hašan, UC Berkeley
Adam Arbree, Autodesk, Inc.
Bruce Walter, Cornell University
Jan Novák, Karlsruhe Institute of Technology

**Figure 1:** *Many-light rendering methods, covered in this report, yield good results at different points along the quality-speed trade-off axis. The images on the left were rendered in real-time with [REH* 11] (courtesy of Tobias Ritschel) and capture diffuse interreflections. The center image took 52 minutes to render and demonstrates many-light methods for participating media (adapted from [ENSD12]). The image on the right combines different phenomena such as glossy surfaces, subsurface BSSRDFs and a detailed anisotropic volumetric cloth model rendered with Bidirectional Lightcuts [WKB12] in about 46 minutes.*

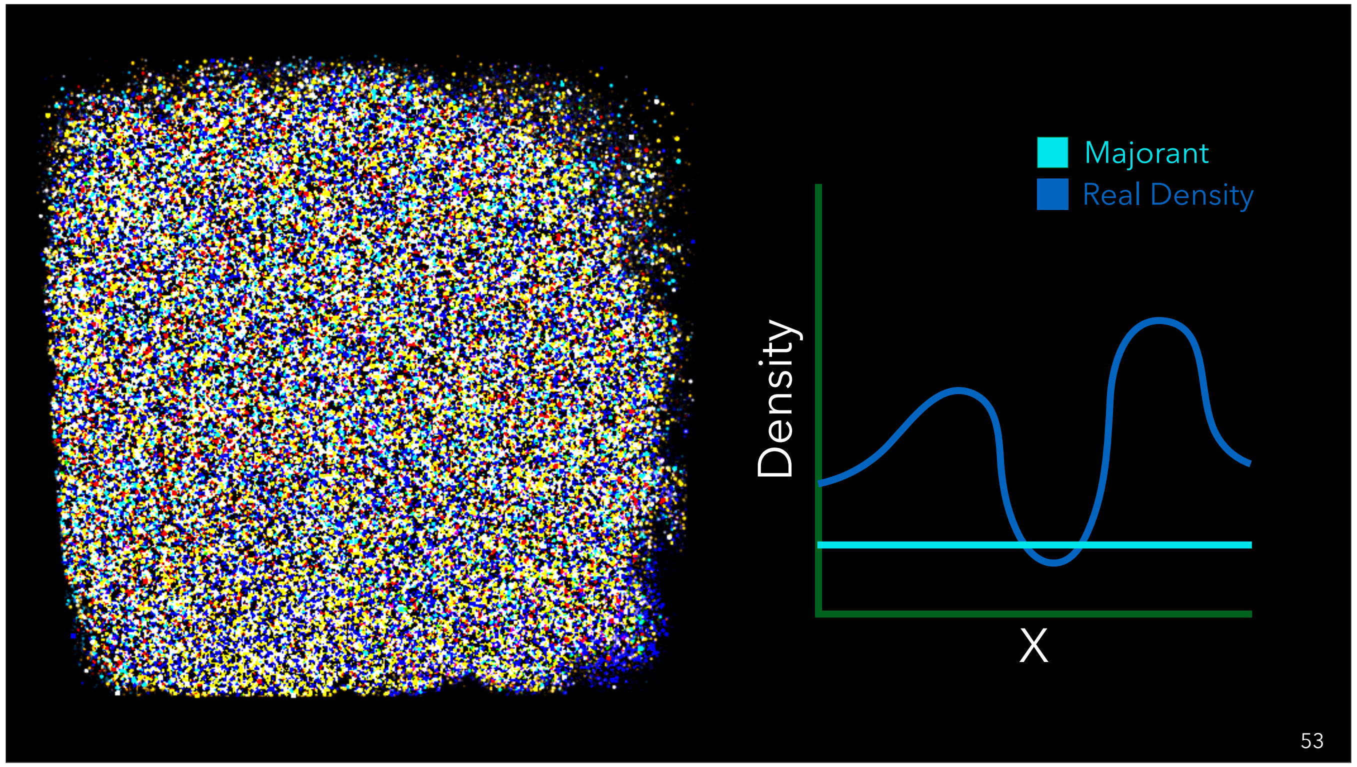
**Abstract**
*Recent years have seen increasing attention and significant progress in many-light rendering, a class of methods for the efficient computation of global illumination. The many-light formulation offers a unified mathematical framework for the problem reducing the full lighting transport simulation to the calculation of the direct illumination from many virtual light sources. These methods are unrivaled in their scalability: they are able to produce artifact-free images in a fraction of a second but also converge to the full solution over time. In this state-of-the-art report, we have three goals: give an easy-to-follow, introductory tutorial of many-light theory; provide a comprehensive, unified survey of the topic with a comparison of the main algorithms; and present a vision to motivate and guide future research. We will cover both the fundamental concepts as well as improvements, extensions, and applications of many-light rendering.*

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Raytracing

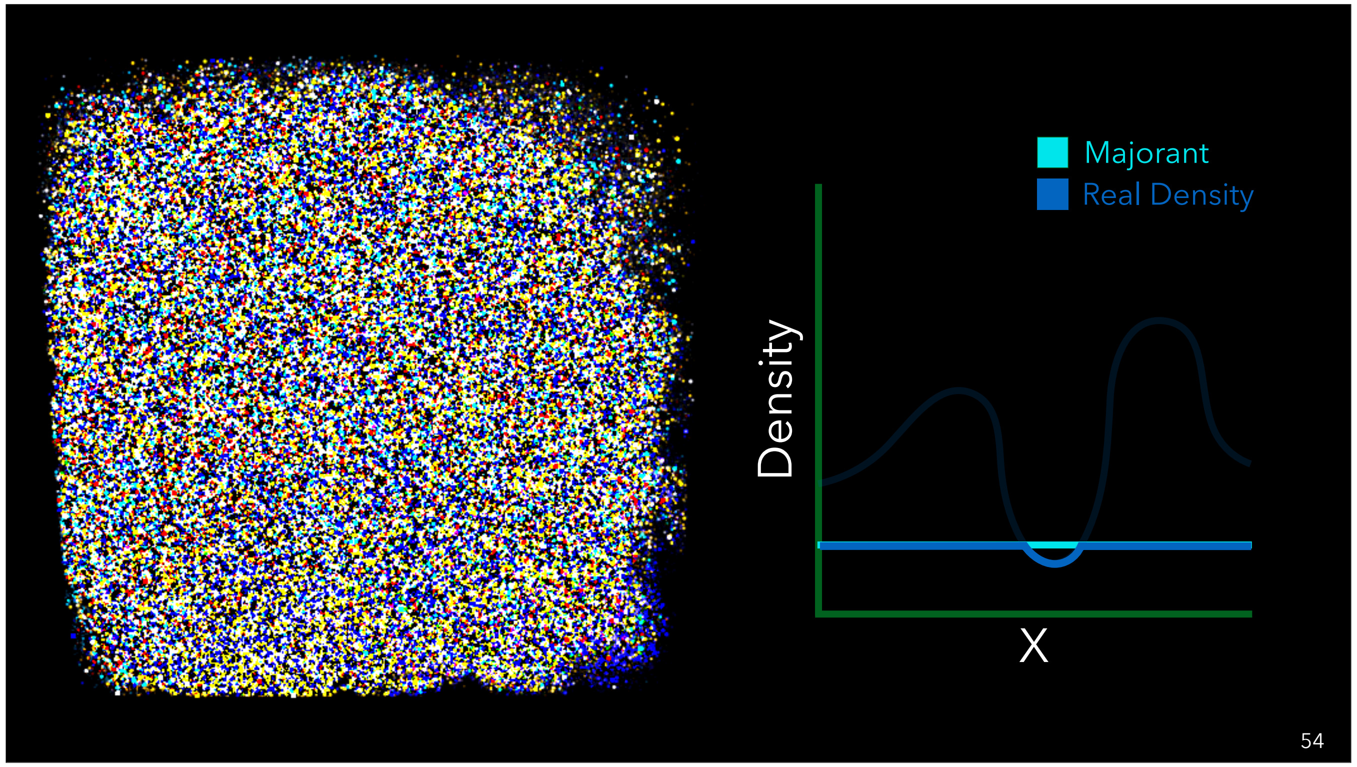
# Clamping to reduce variance



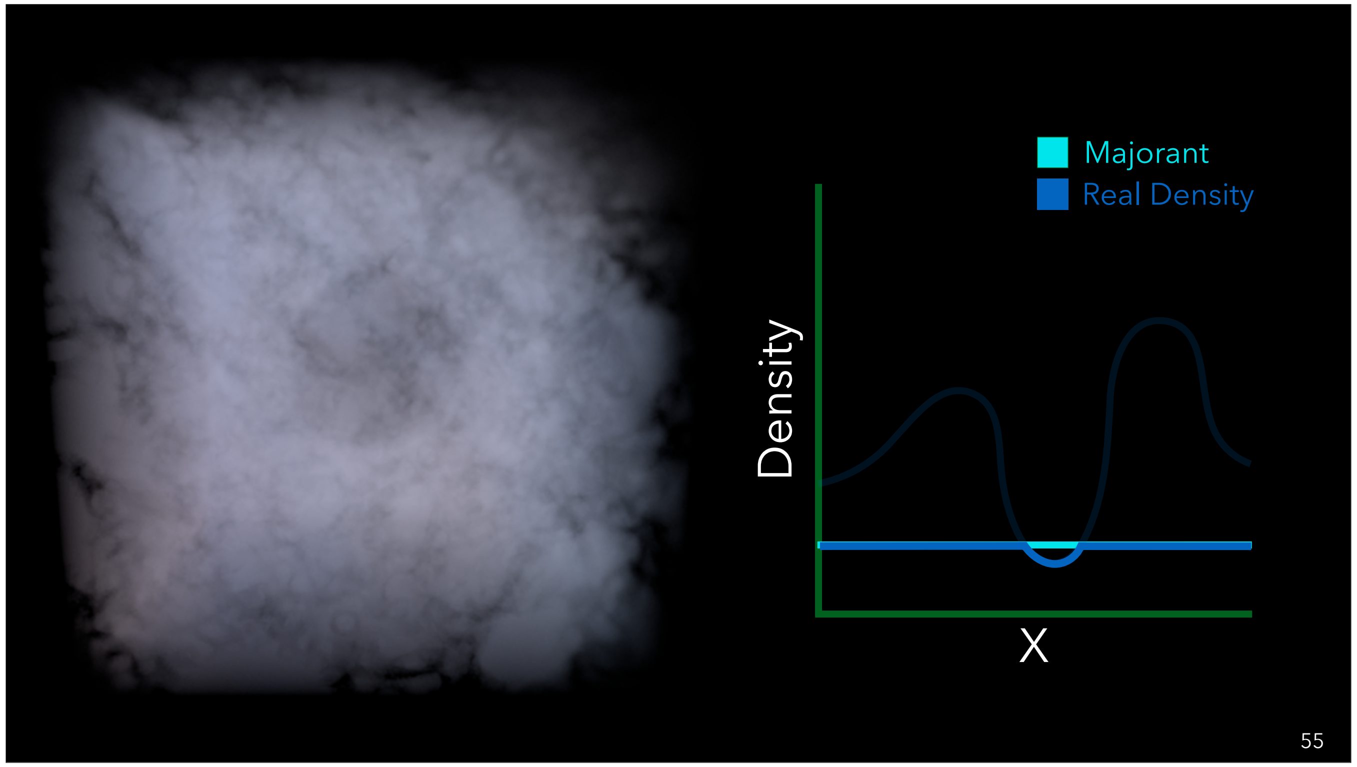
Is the idea of clamping to reduce variance. This is the first step in our technique.



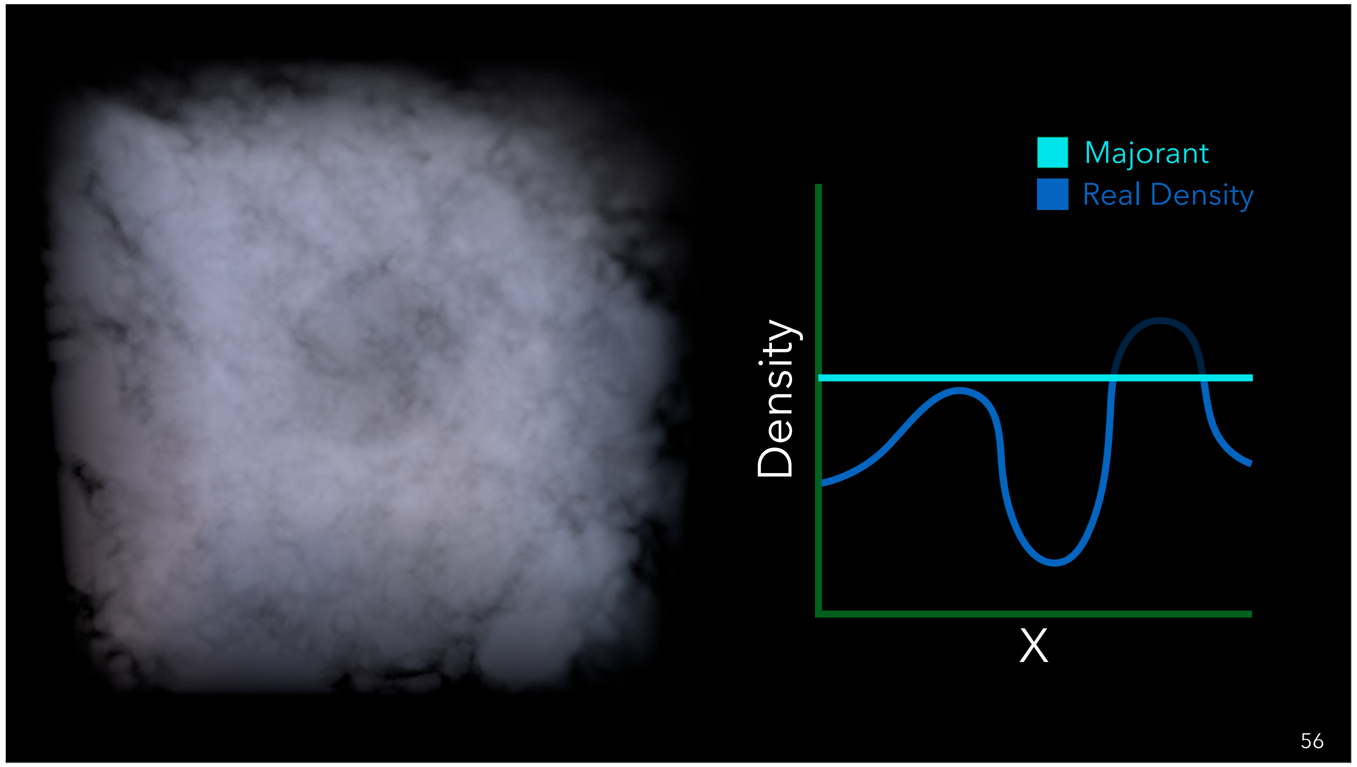

Significantly non-bounding majorants lead to uncontrollable variance, HOWEVER, we can enforce our majorants to be always bounding,



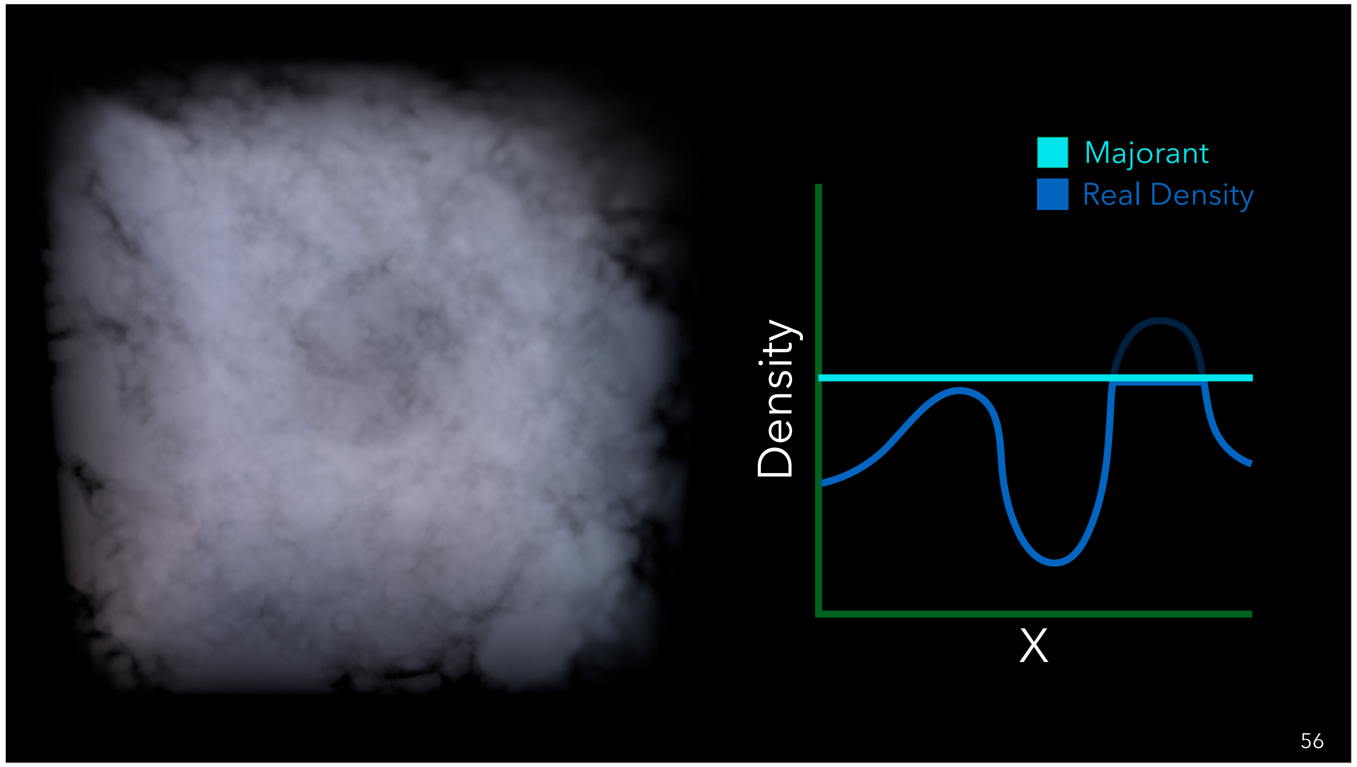

by clamping the medium density to the specified majorant. This process will obviously



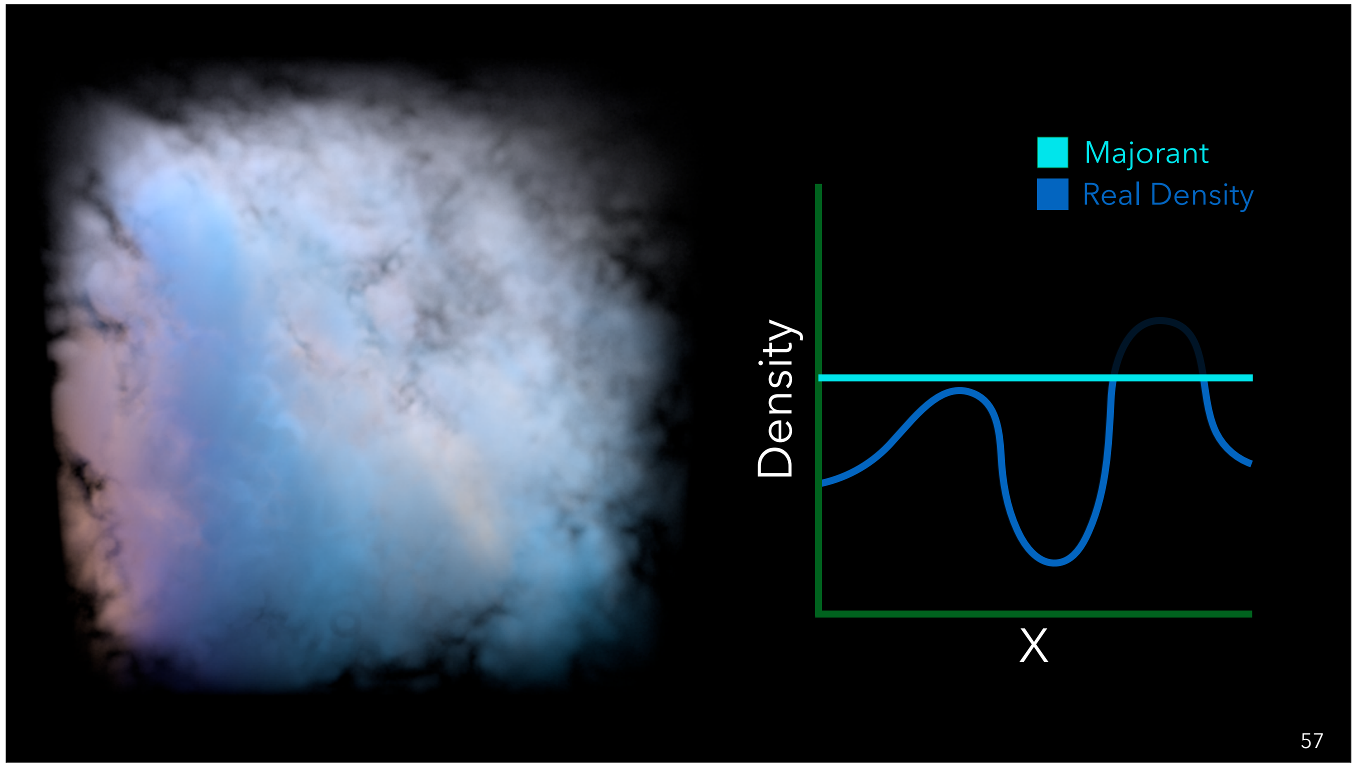

Make the medium itself biased. However, one thing to make note of is that if we



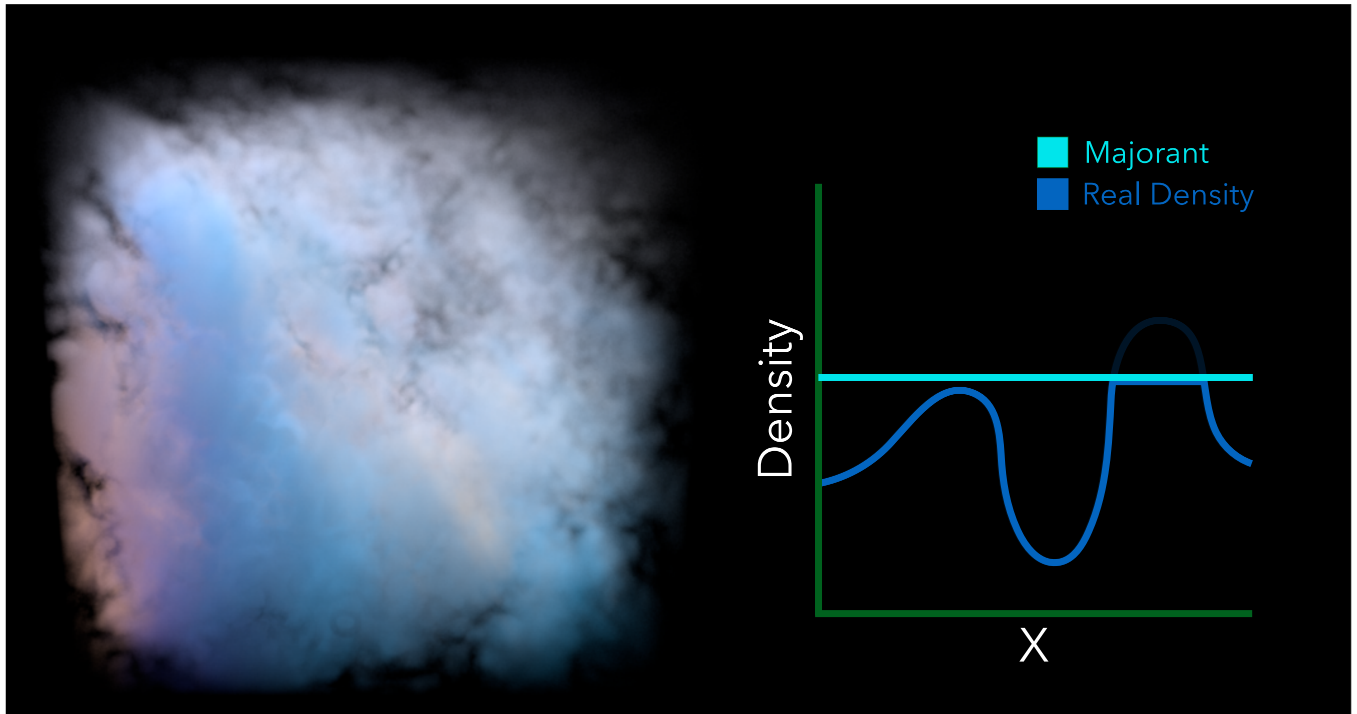

Increased the majorant, and thus clamped less of the density,



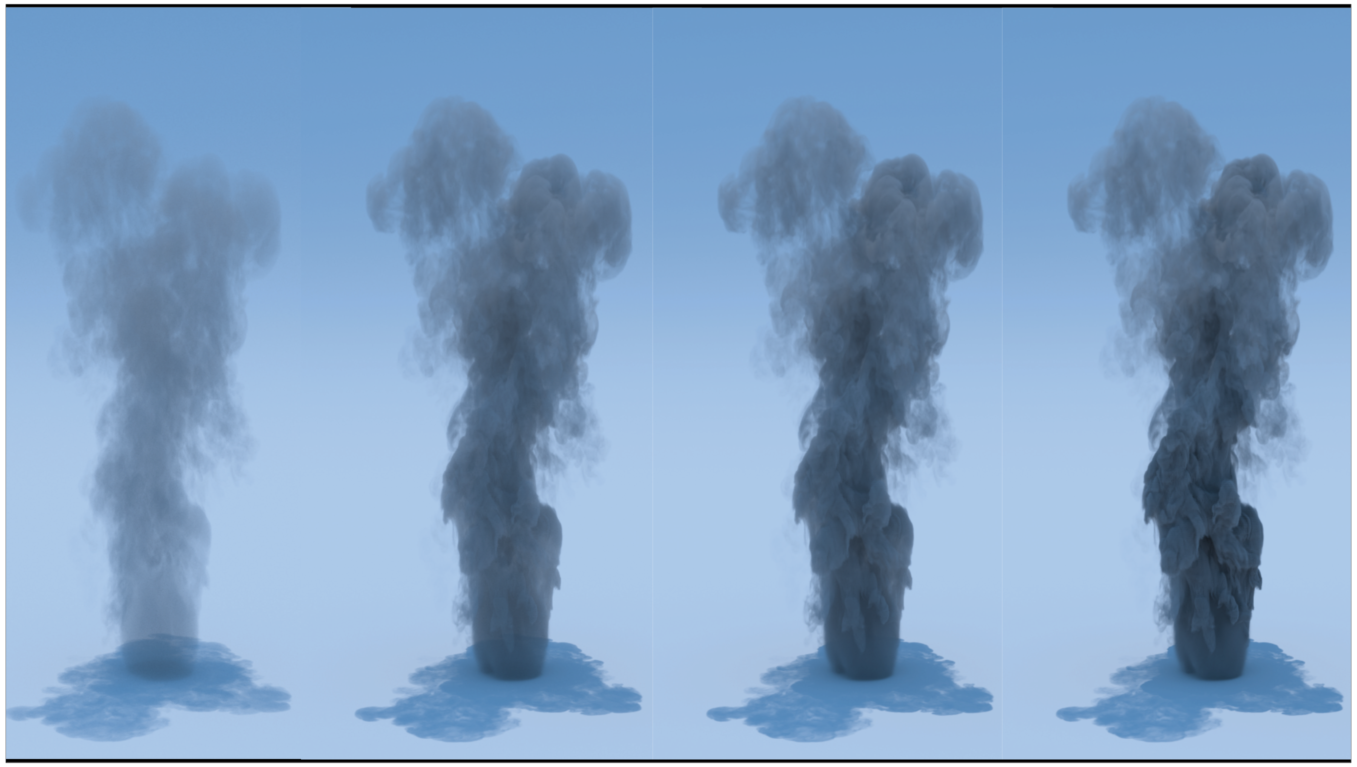
Majorant
Real Density
Density
X
56

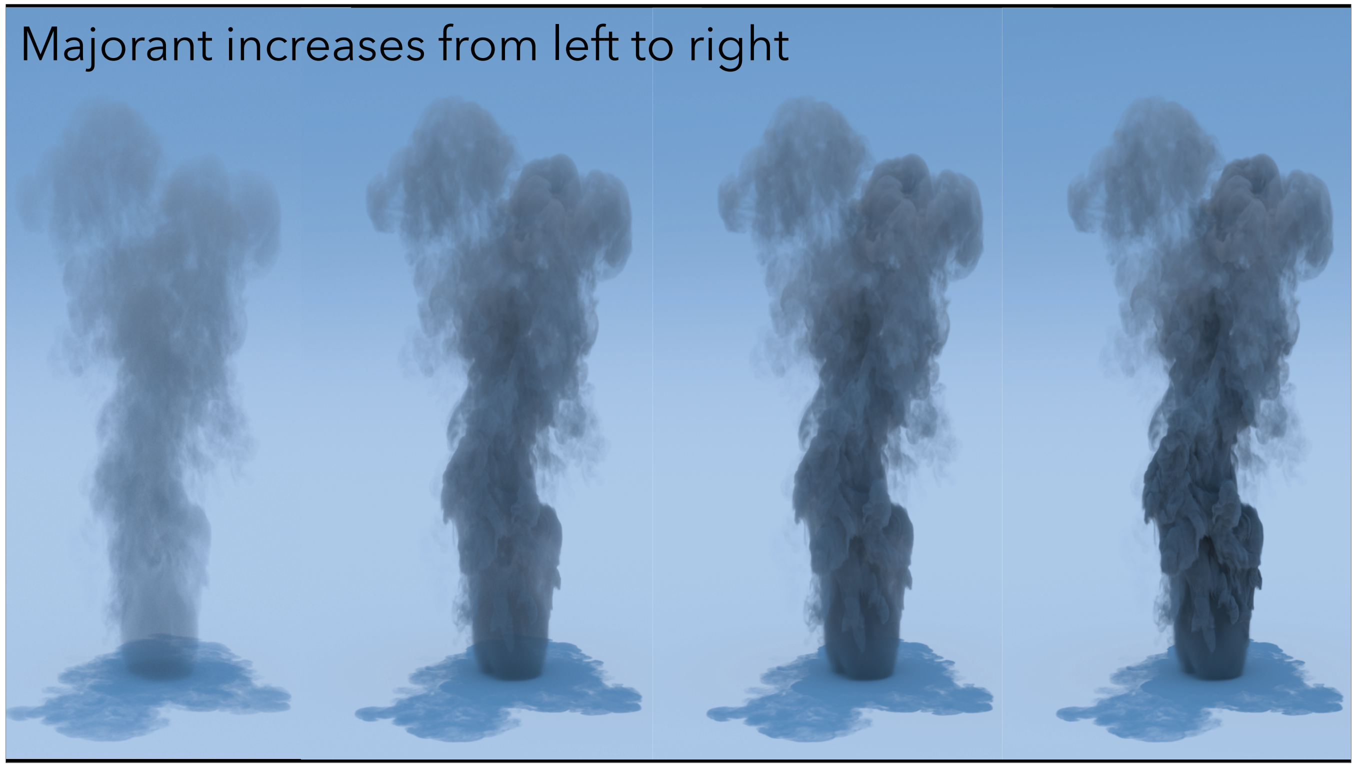



Our render would be less biased. And you can now start to see a thought experiment forming.

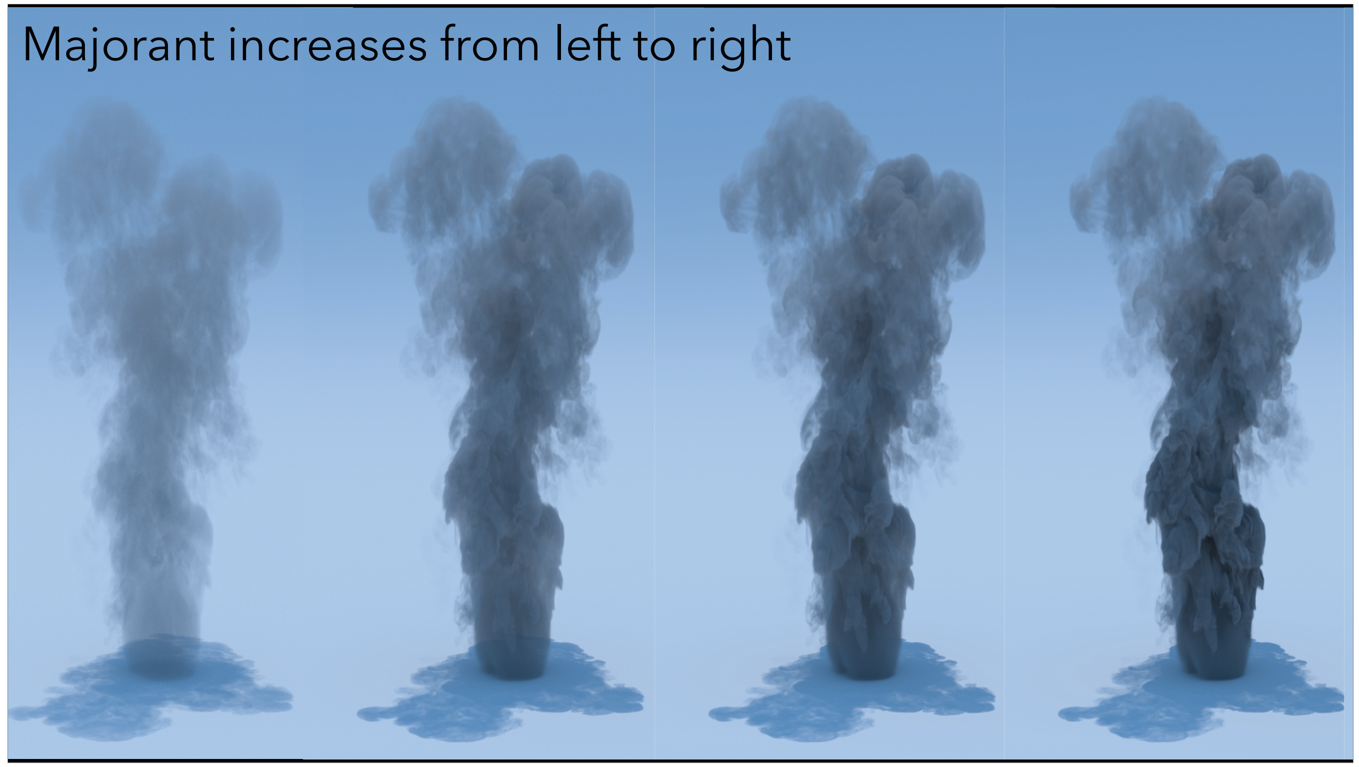

Majorant
Real Density
Density
X


What if we had a sequence of pixel samples,

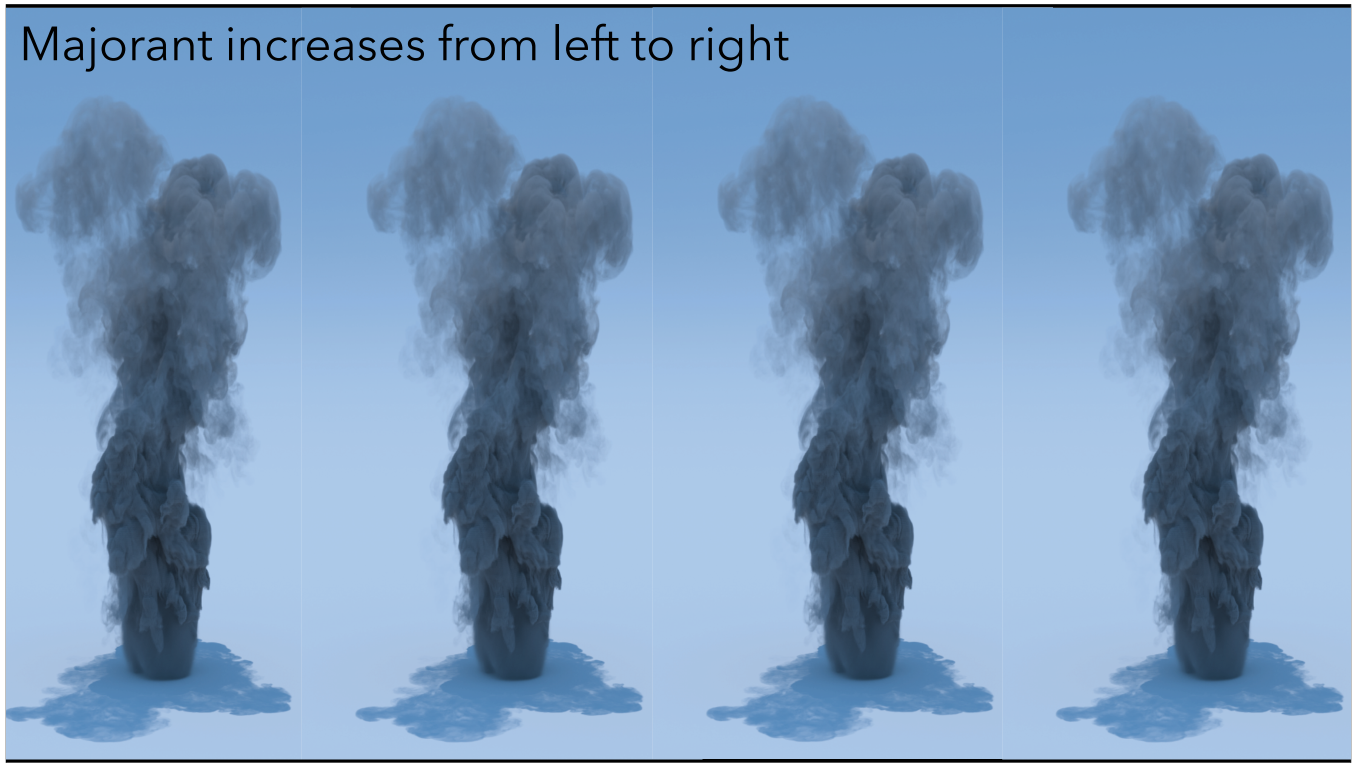





which all used monotonically increasing majorants. Meaning, the first few pixel samples will be biased,




But after some finite point a bounding majorant will be found and every subsequent pixel sample will be unbiased.


Majorant increases from left to right

$$\sum_{k=1}^{n} \frac{I_k}{n}$$

61

The entire render would be the average of all these images.

$$\sum_{k=1}^{j} \frac{I_k}{n} + \sum_{k=j+1}^{n} \frac{I_k}{n}$$



Which we can decompose into a finite number of (*click*) biased terms, while the remainder are all (*click*) unbiased.

$$\sum_{k=1}^{j} \frac{I_k}{n} + \sum_{k=j+1}^{n} \frac{I_k}{n}$$

Biased

$$\sum_{k=1}^{j} \frac{I_k}{n} + \boxed{\sum_{k=j+1}^{n} \frac{I_k}{n}}$$

Biased Unbiased

$$\lim_{n\to\infty}\left[\sum_{k=1}^{j}\frac{I_k}{n}+\sum_{k=j+1}^{n}\frac{I_k}{n}\right]$$

Biased Unbiased

63

In the infinite limit, the biased contribution is going to converge to

$$\lim_{n\to\infty}\left[\;0\;+\sum_{k=j+1}^{n}\frac{I_k}{n}\right]$$

Biased Unbiased



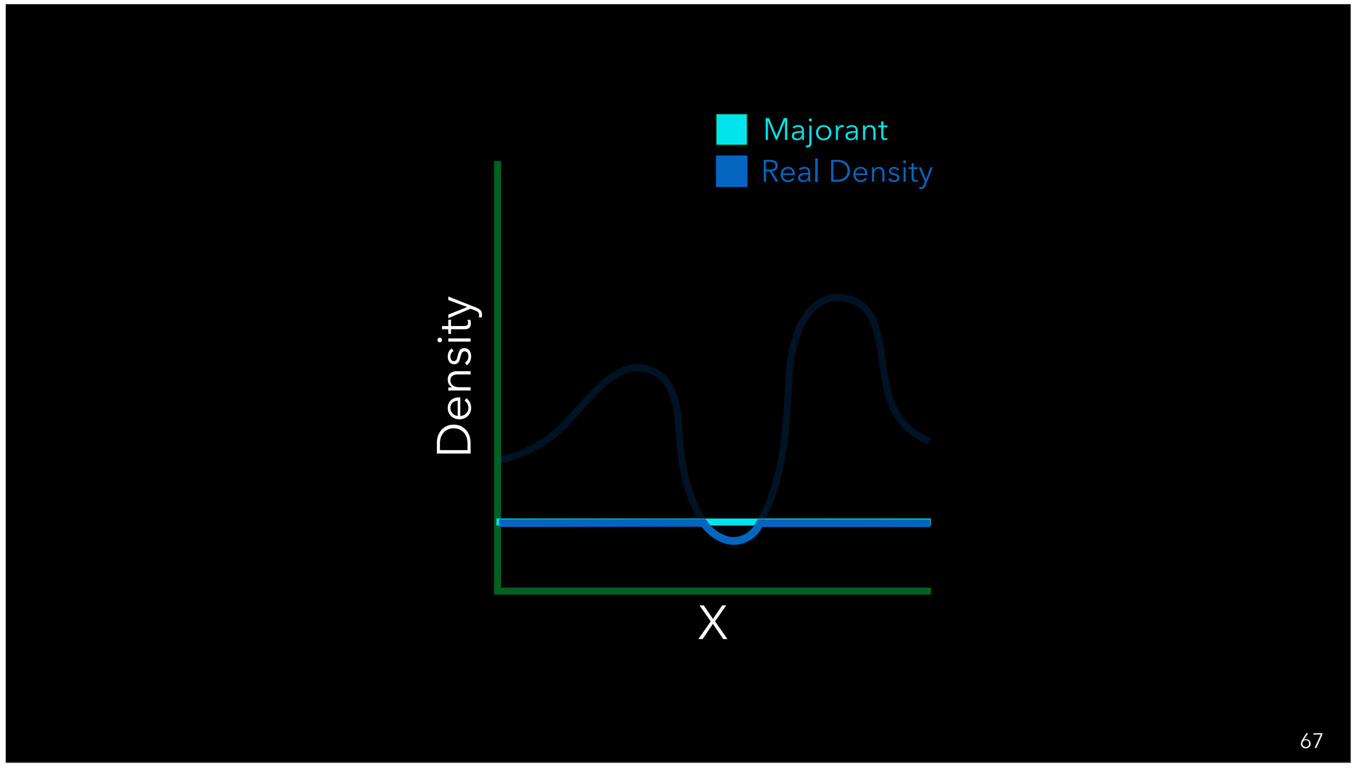
Zero, while the contribution from the infinite remaining unbiased terms is going to converge to


The true solution. So, if we discover a bounding majorant in finite time, we can make most null-scattering algorithms consistent while avoiding uncontrollable variance.

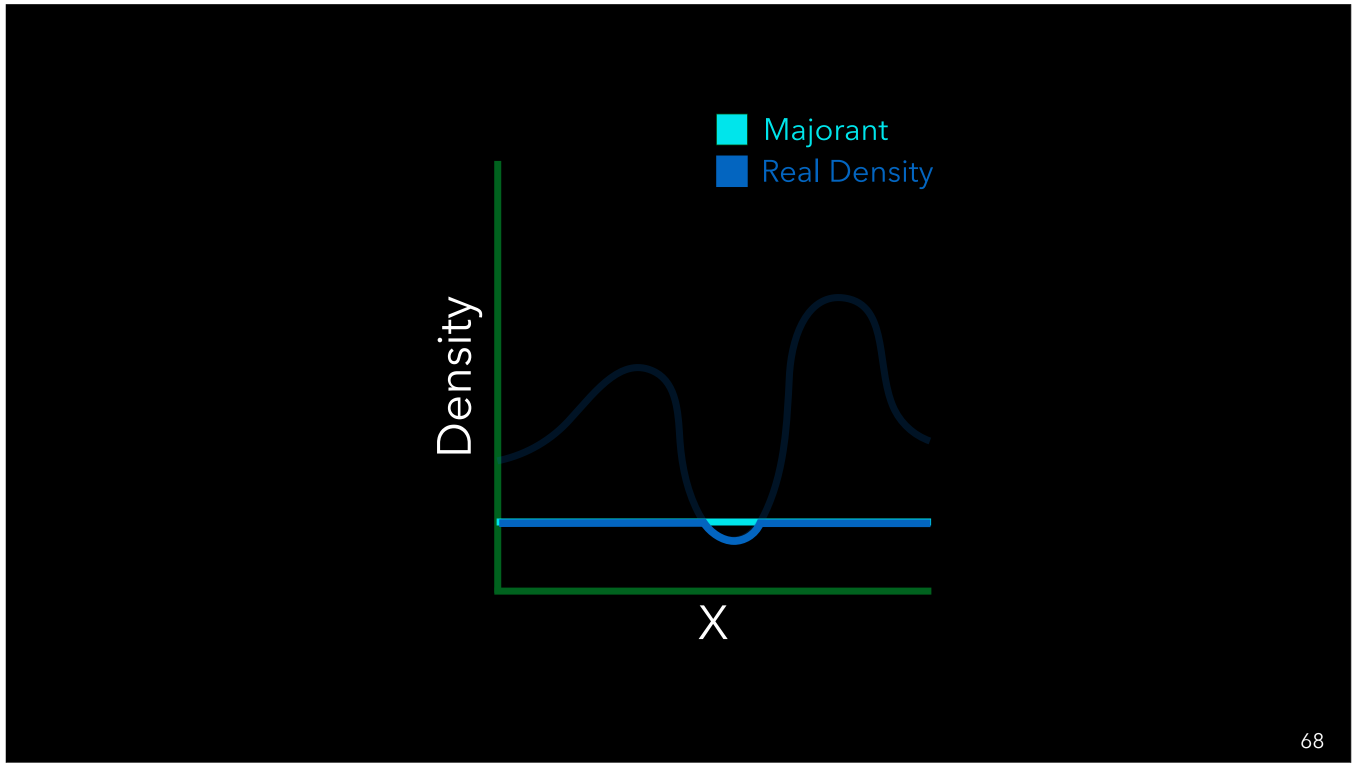
# Progressively update majorants



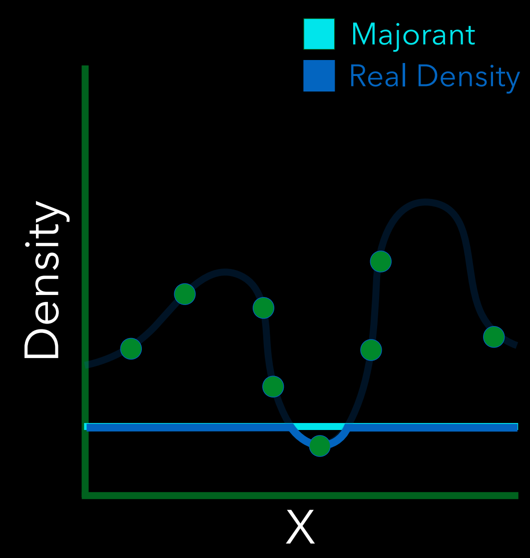
Which brings us to the second step in our technique. Progressively updating the majorants.



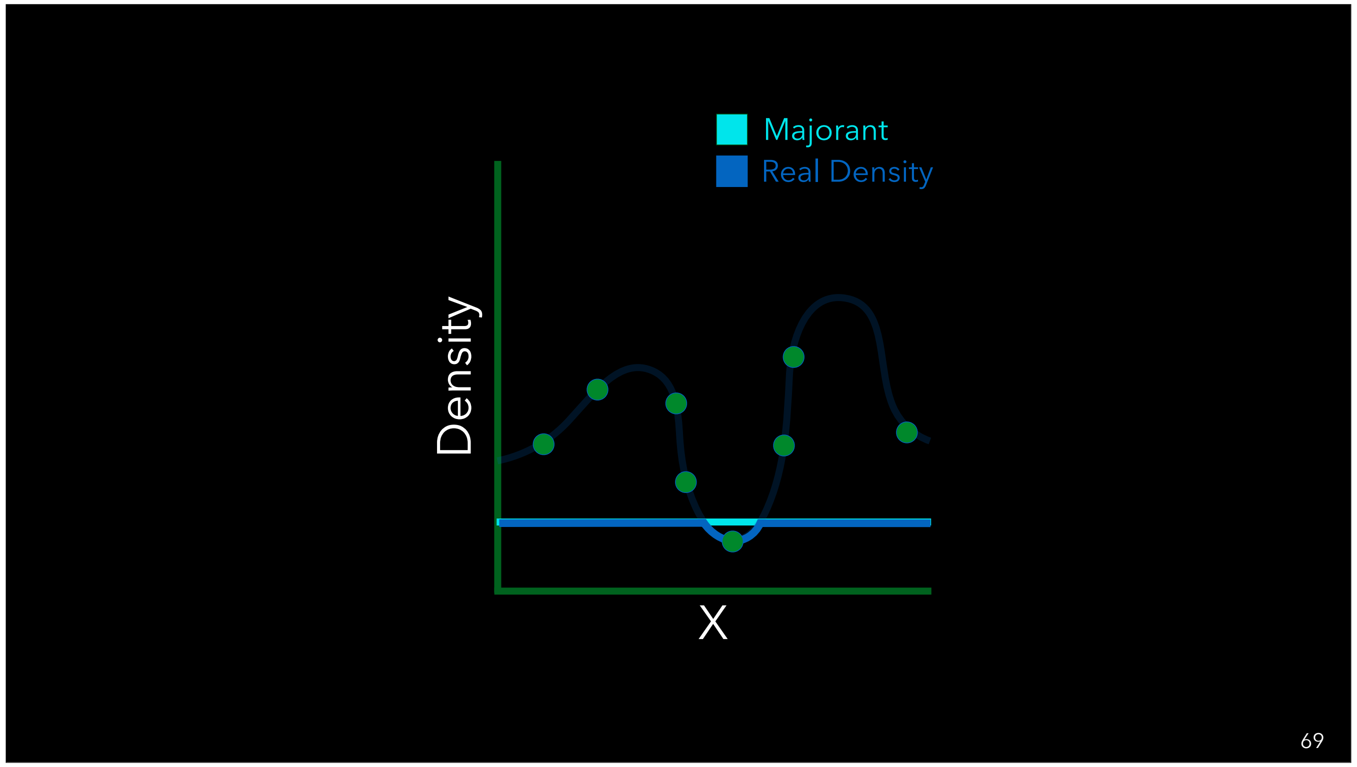

Over the course of rendering, we will naturally evaluate the density



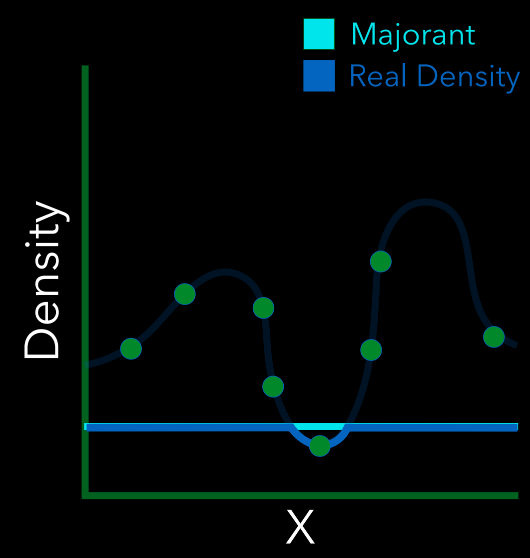

At many different points within the medium.

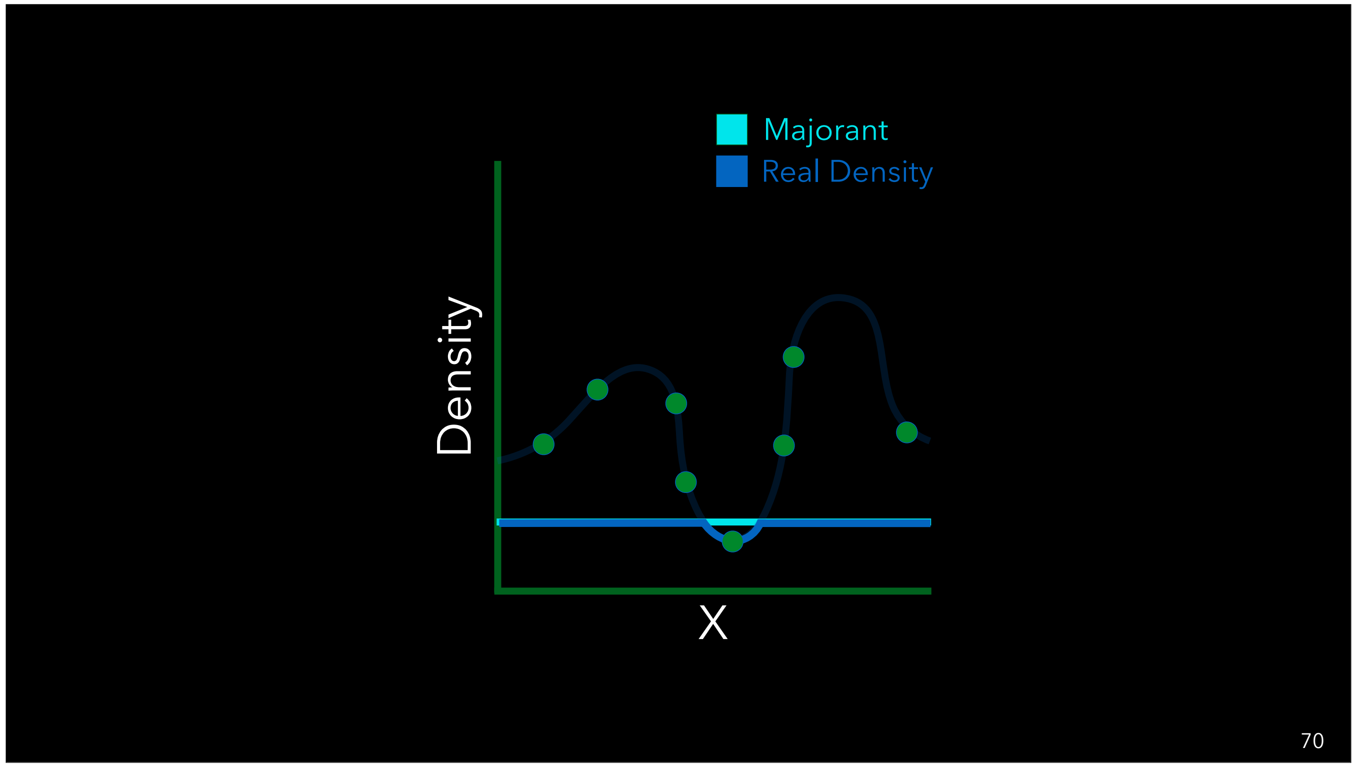


Majorant
Real Density
Density
X

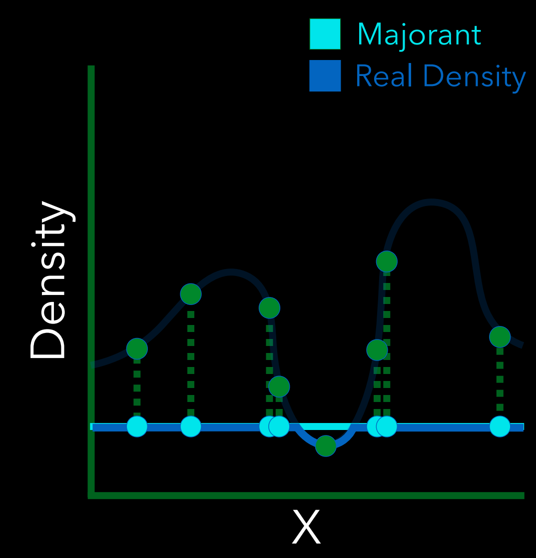



Every single one of these density evaluations may or may not

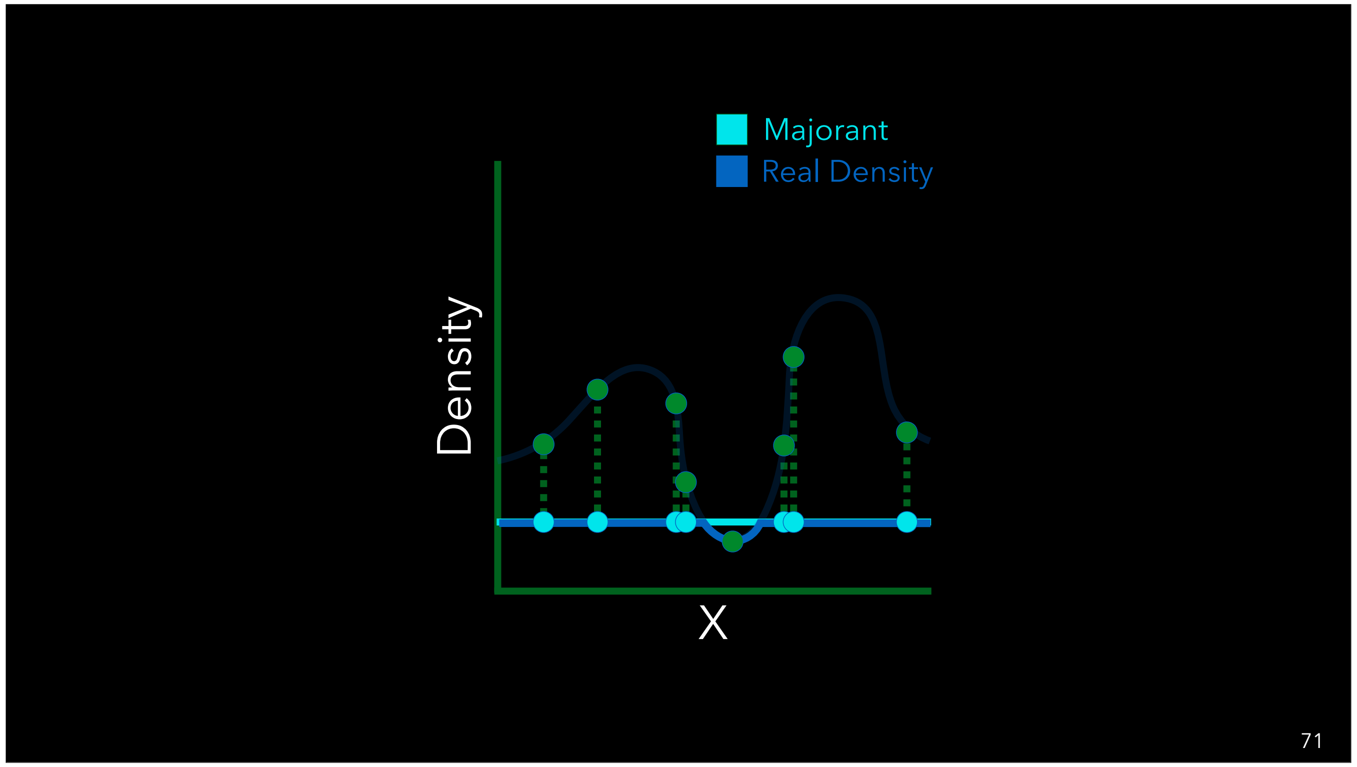

Majorant
Real Density
Density
X



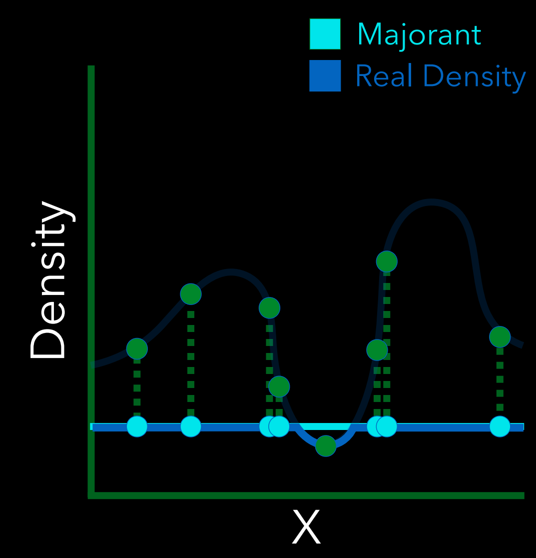

get clamped. However, All of these evaluations give us direct,



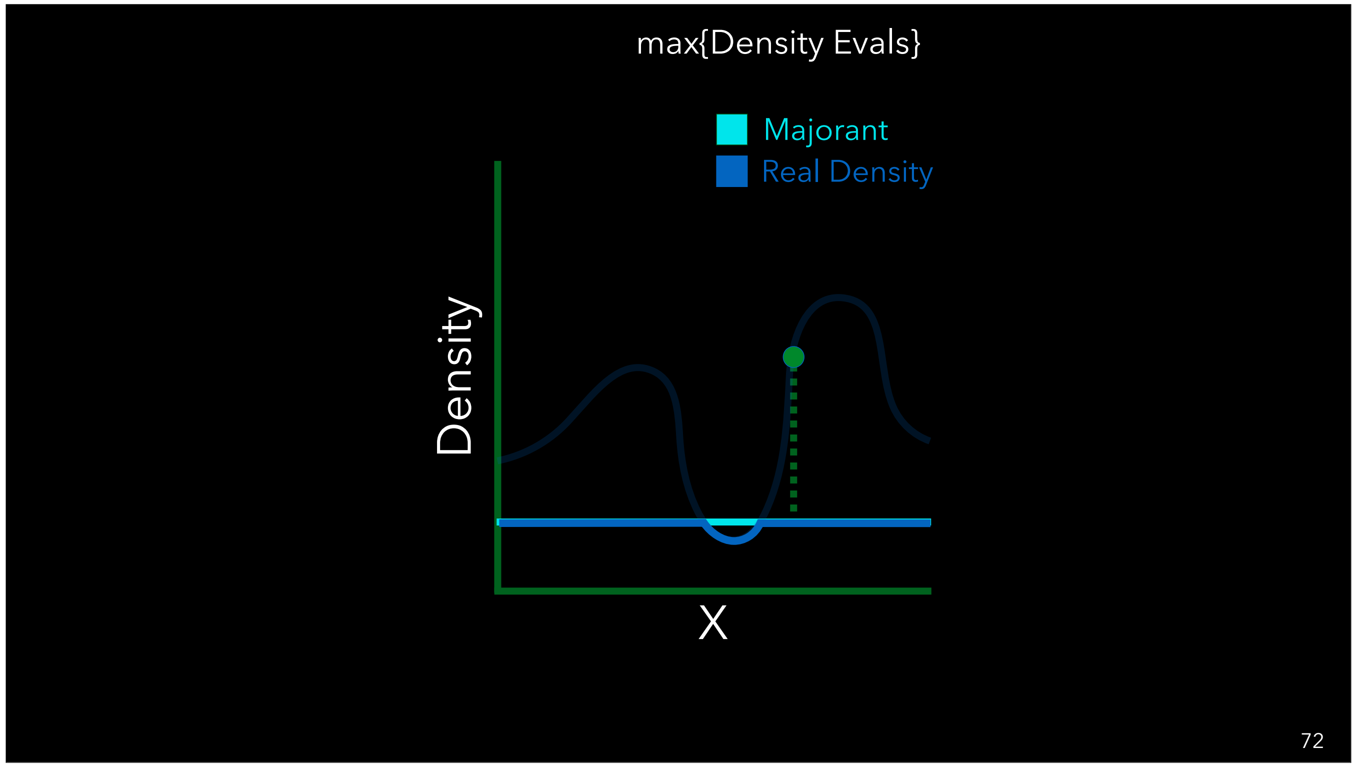
Majorant
Real Density
Density
X



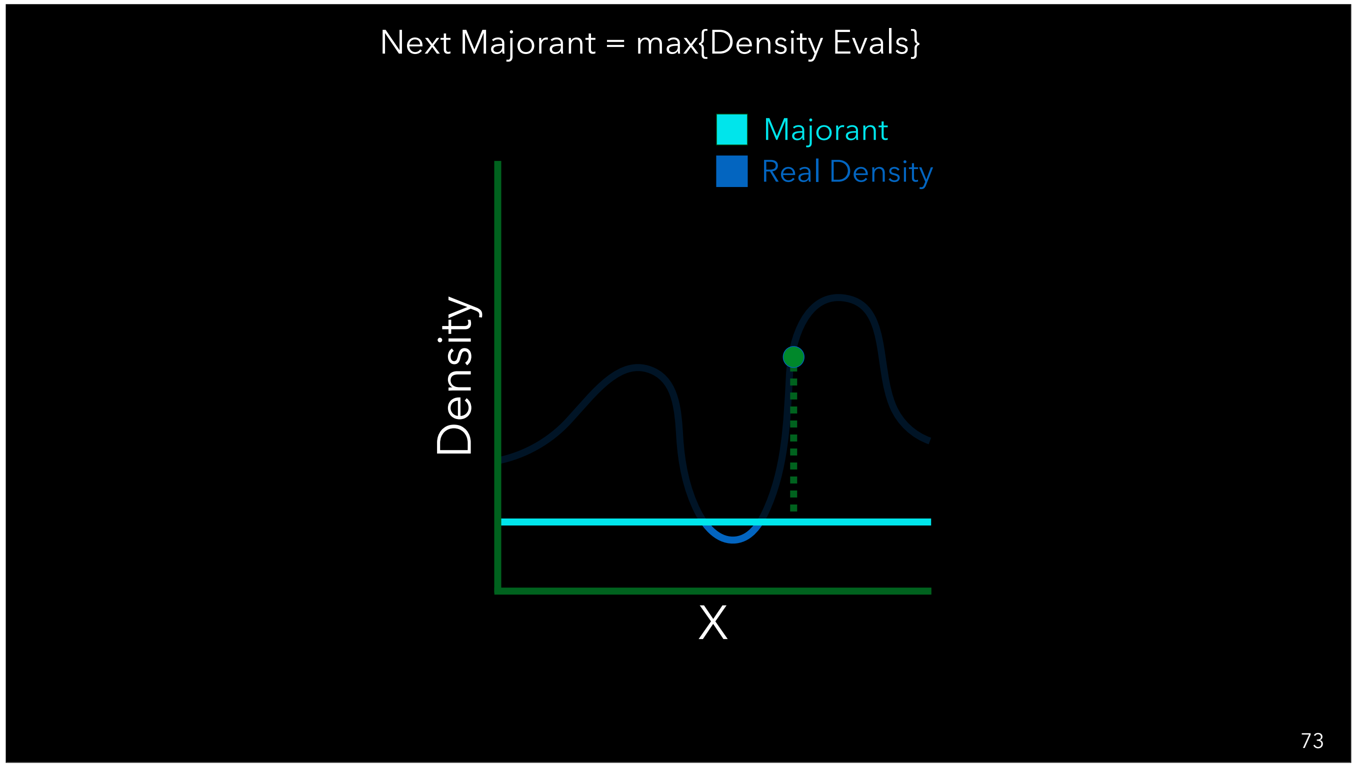

Estimates for how non-bounding our majorant actually is. We can then choose



Majorant
Real Density
Density
X

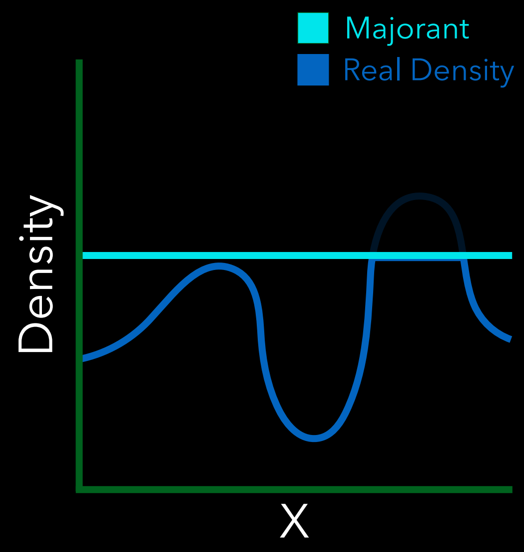


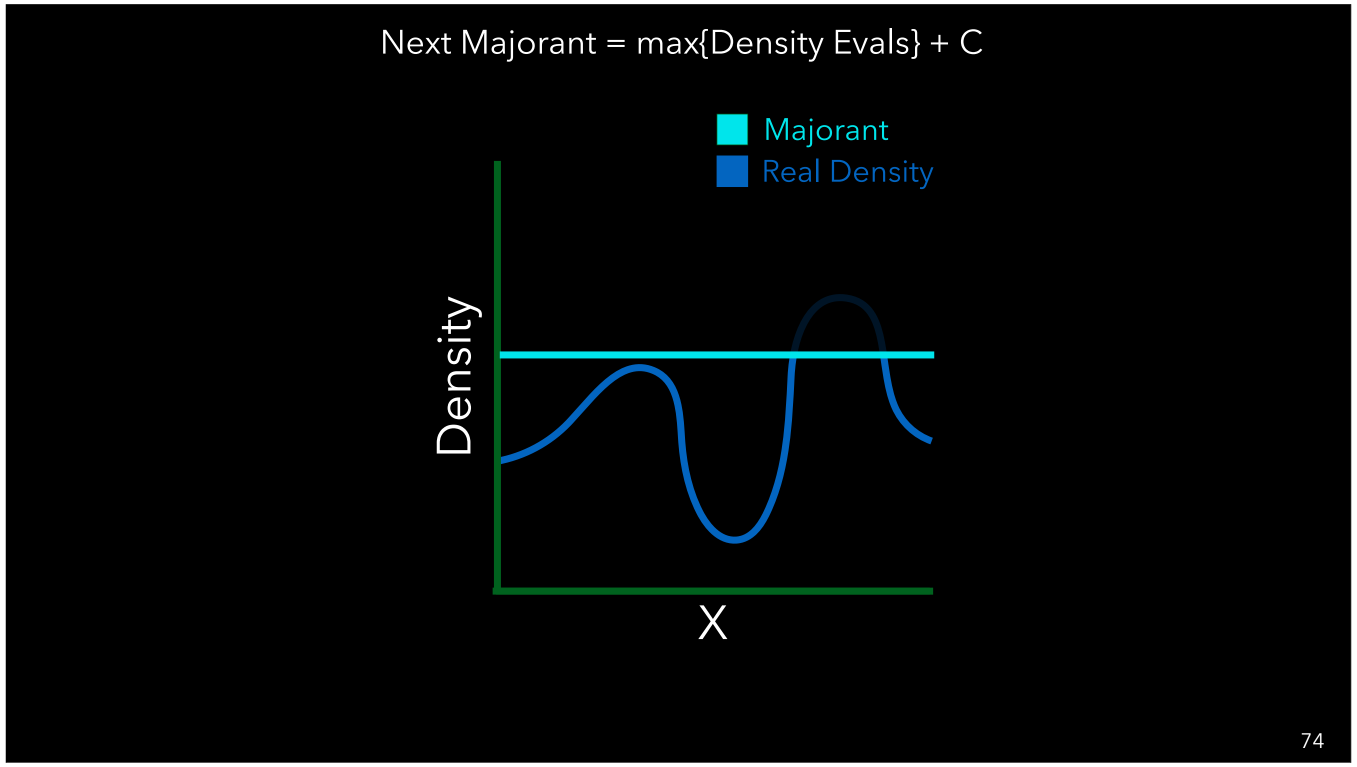

the largest difference between any of the density evaluations and our current majorant,




To directly set the majorant to use for the next render pass. We also add a small non-zero

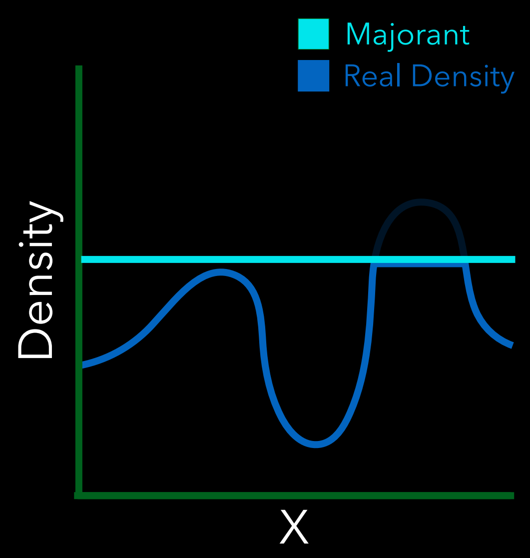
Next Majorant = max{Density Evals}






constant to the updated majorant to guarantee that we will discover a bounding majorant in finite time. For brevity, we refer you to the paper for our explanation regarding this. The combination of clamping then progressively updating majorants fully summarizes our progressive null-tracking technique.

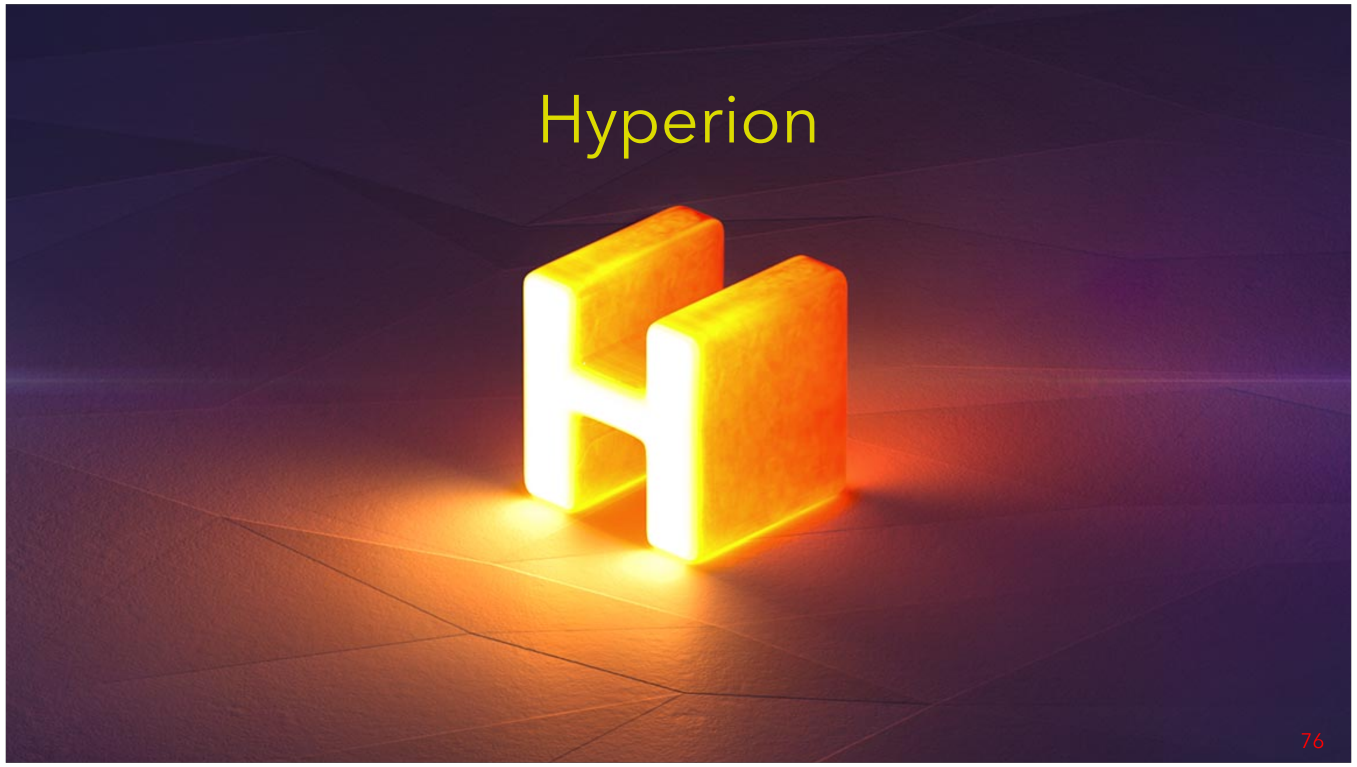
Next Majorant = max{Density Evals} + C



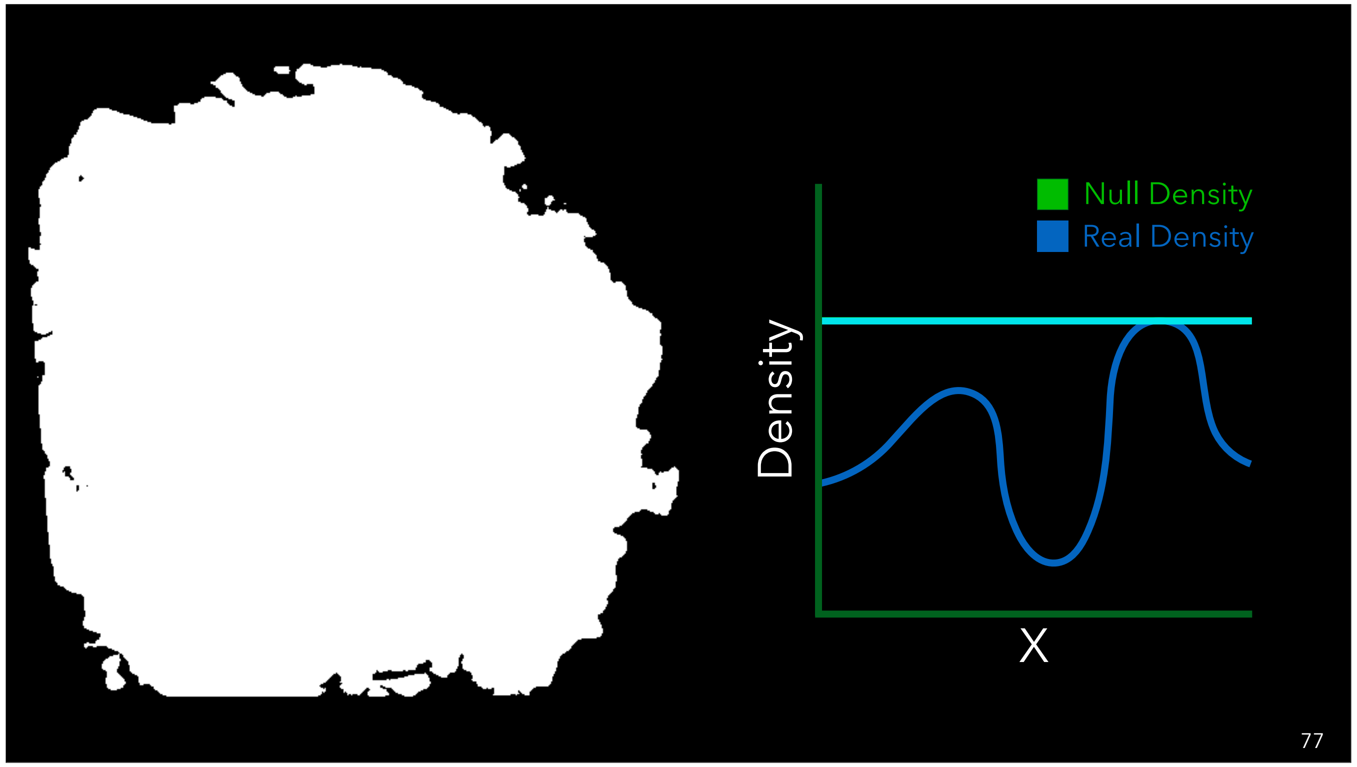
# Implementation



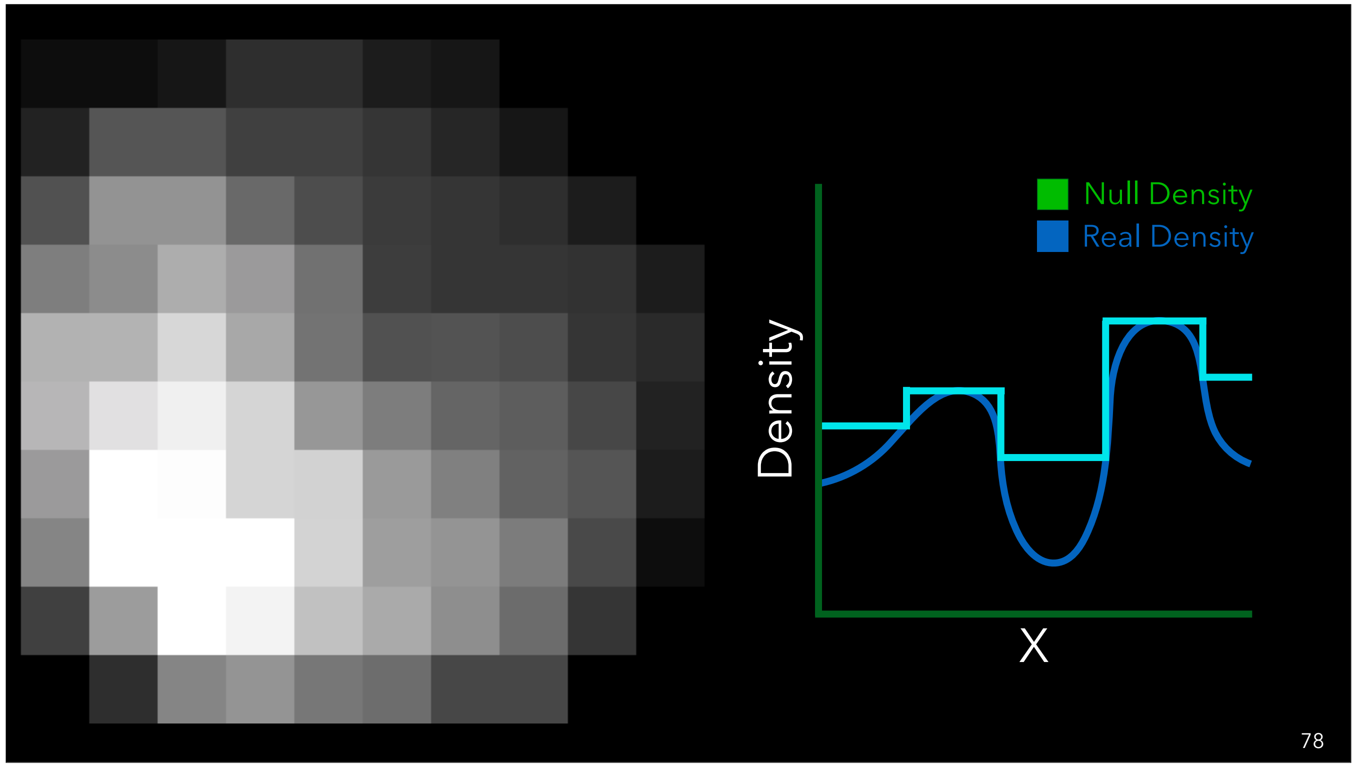
Before we move onto results. We need to mention that while it is our intention to eventually incorporate this technique




Into Hyperion, most of our implementations and results are from PBRT.




Additionally, while we introduced the idea of a majorant as if it were a singular global constant. In practice, we store it

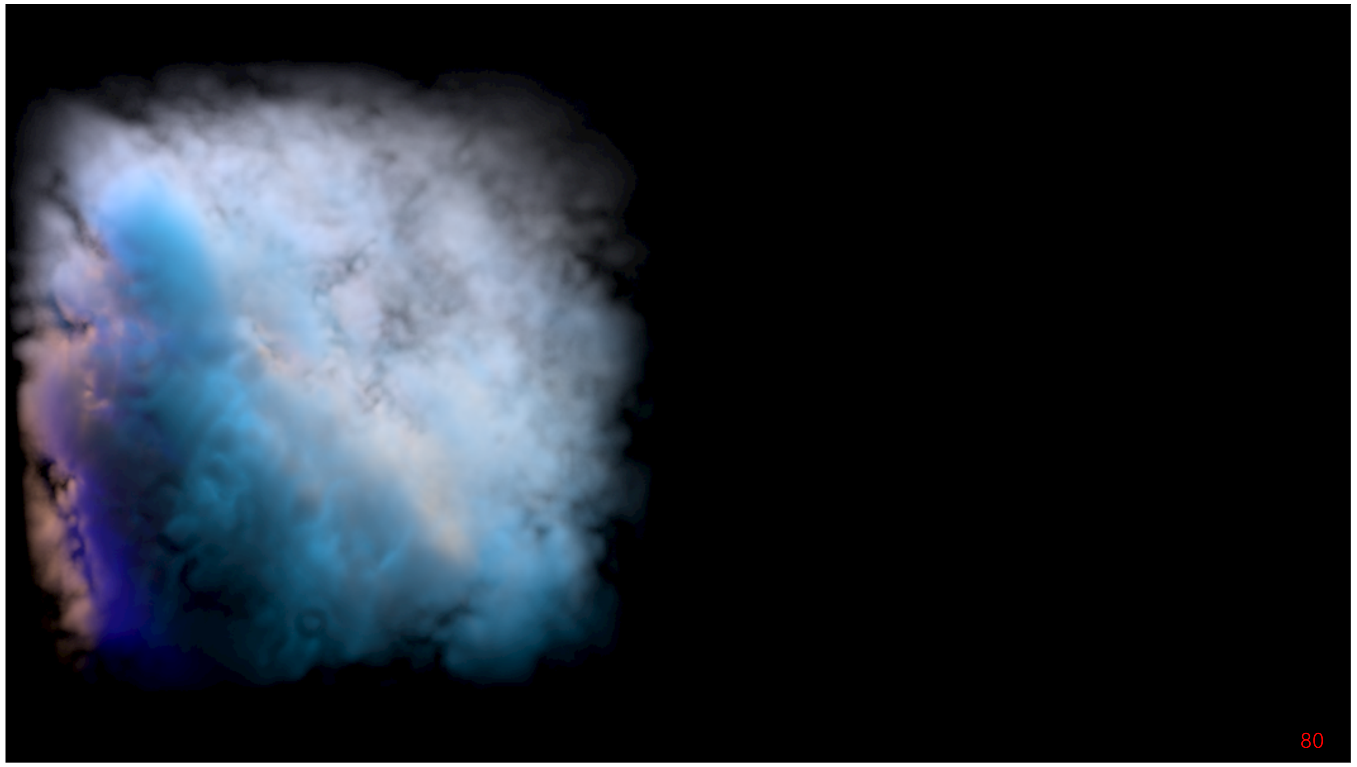



As a piecewise constant function to better locally fit the medium. Thus, we progressively update each majorant individually.

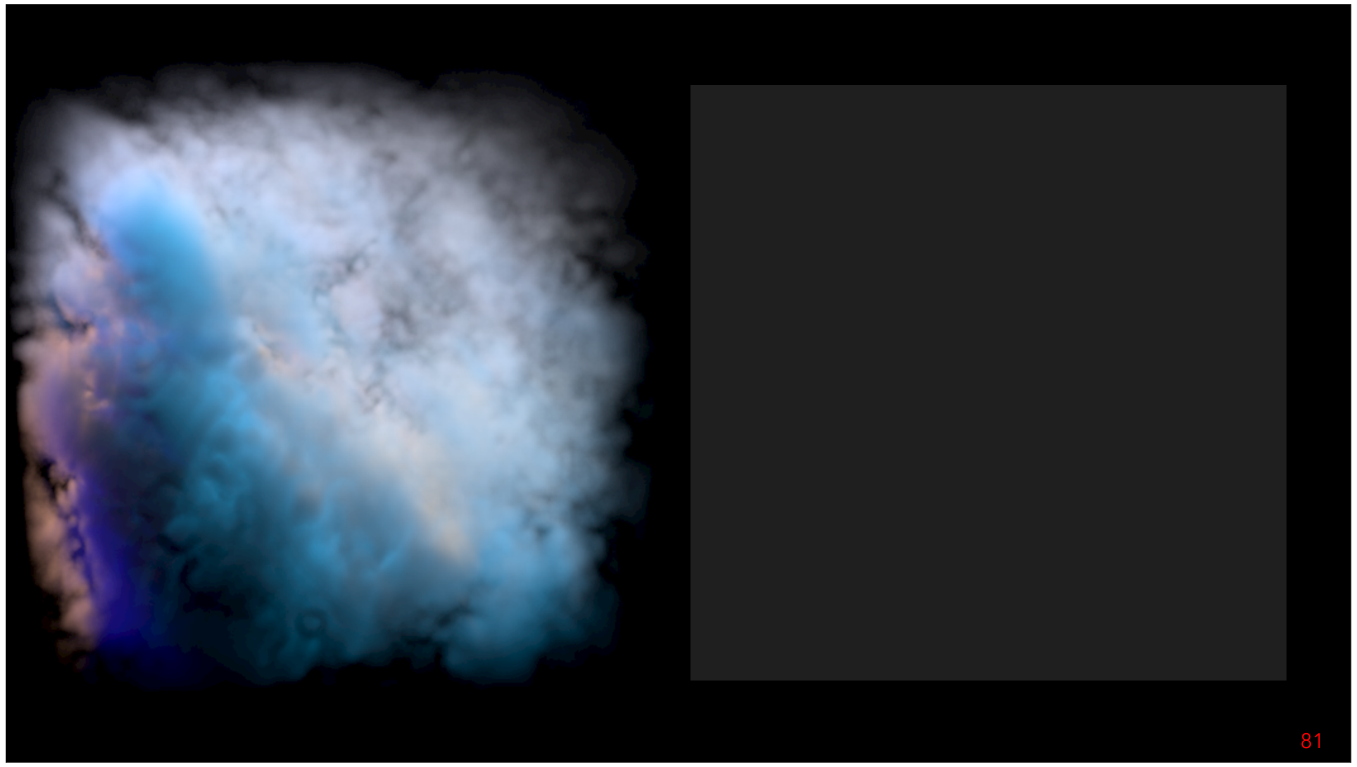
# Results



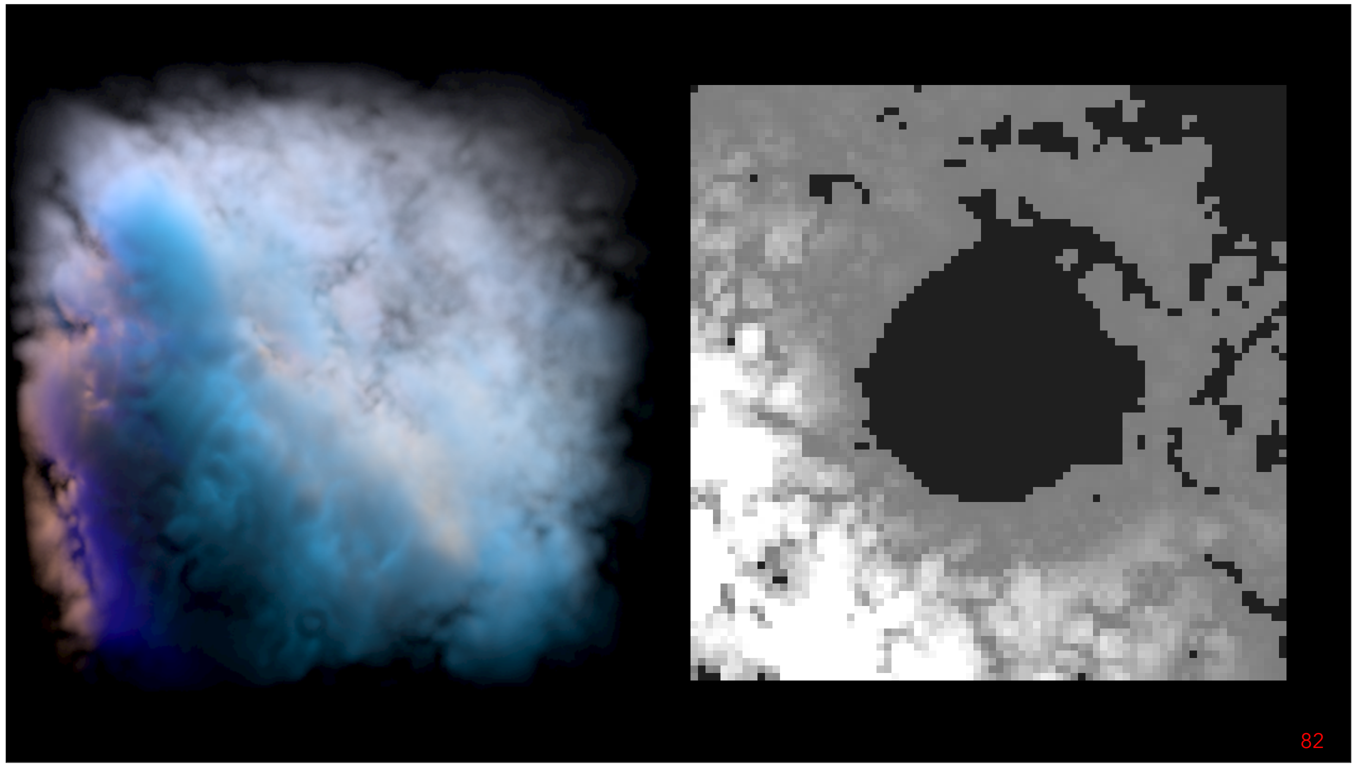
Now, on to some results.




For all results rendered using our technique,

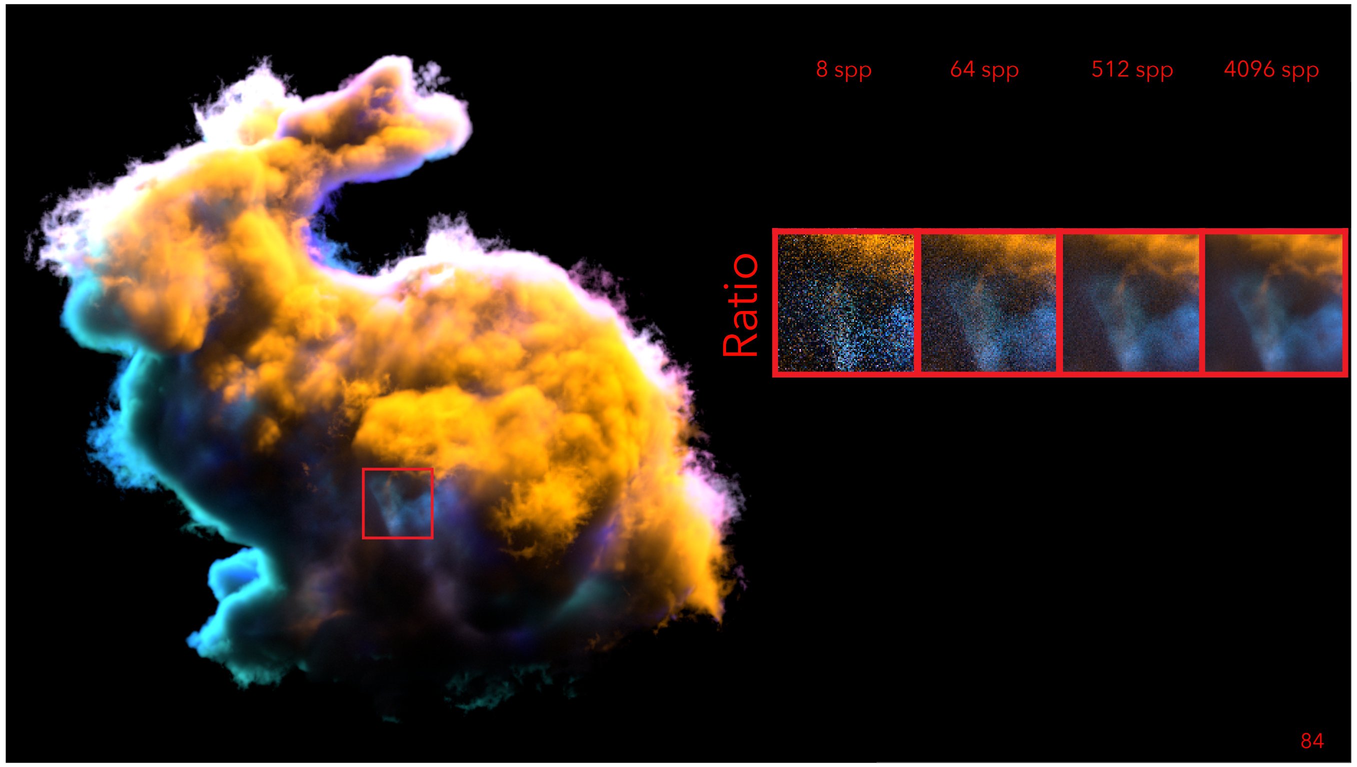



We initialize our majorants to be near-zero to convey the robustness of our technique in the worst case scenario. Our progressive method then updates

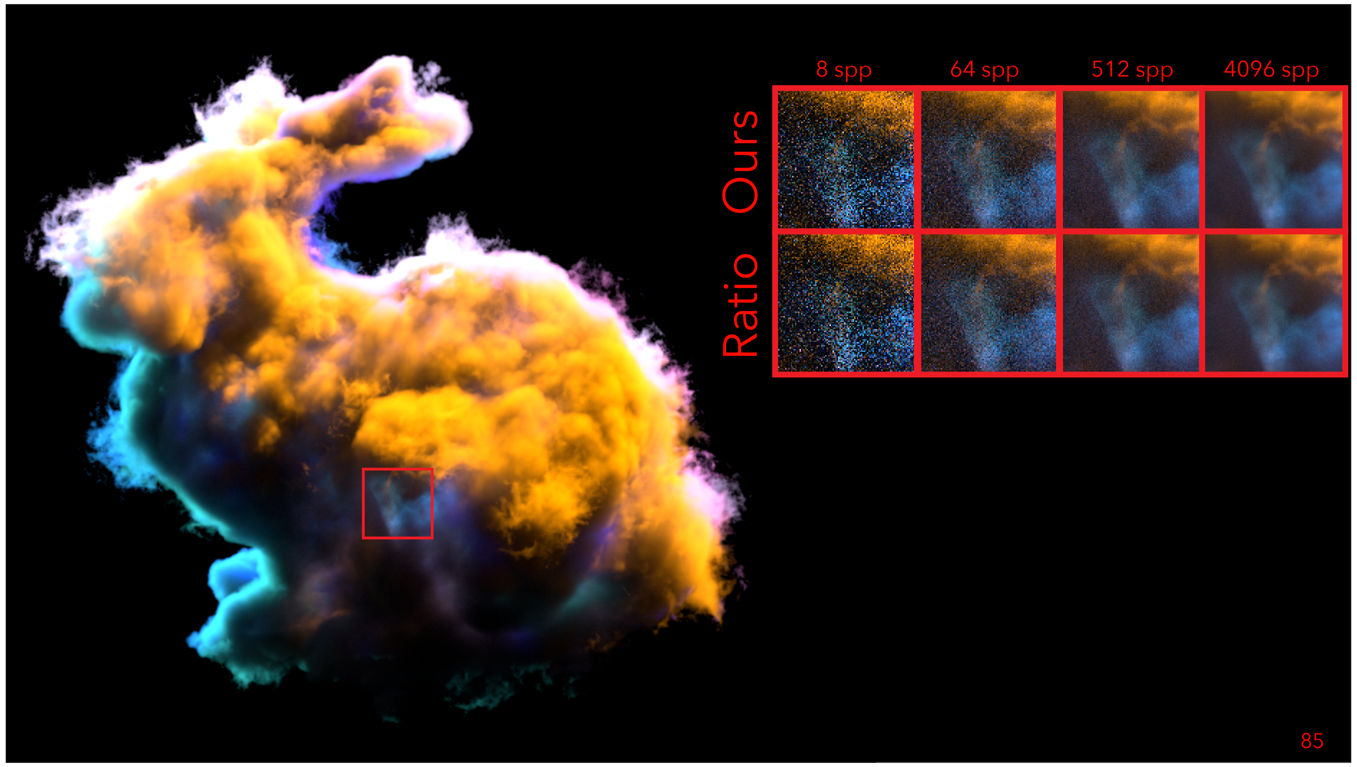



Those majorants over the course of a render.

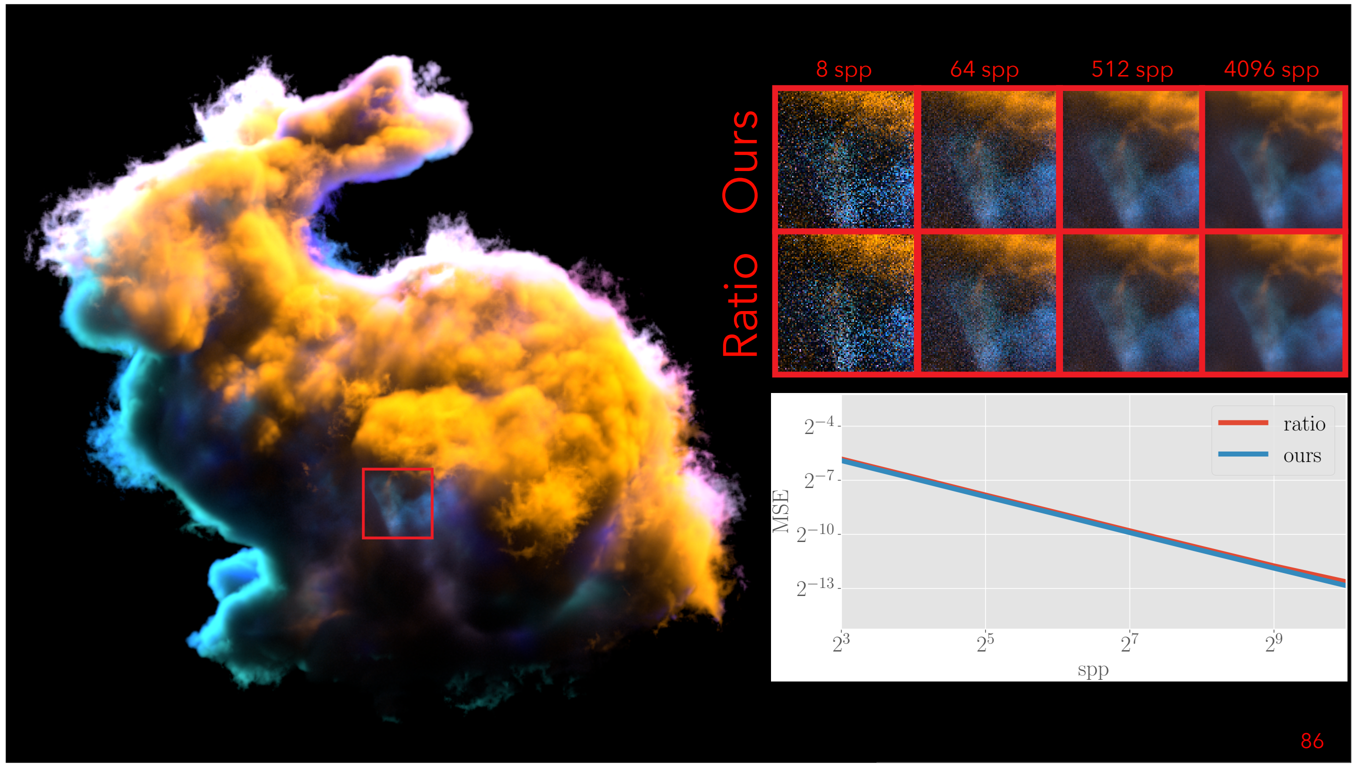

For this bunny scene, we compare




ratio plus weighted delta tracking which are given tightly bounding majorants ahead of time

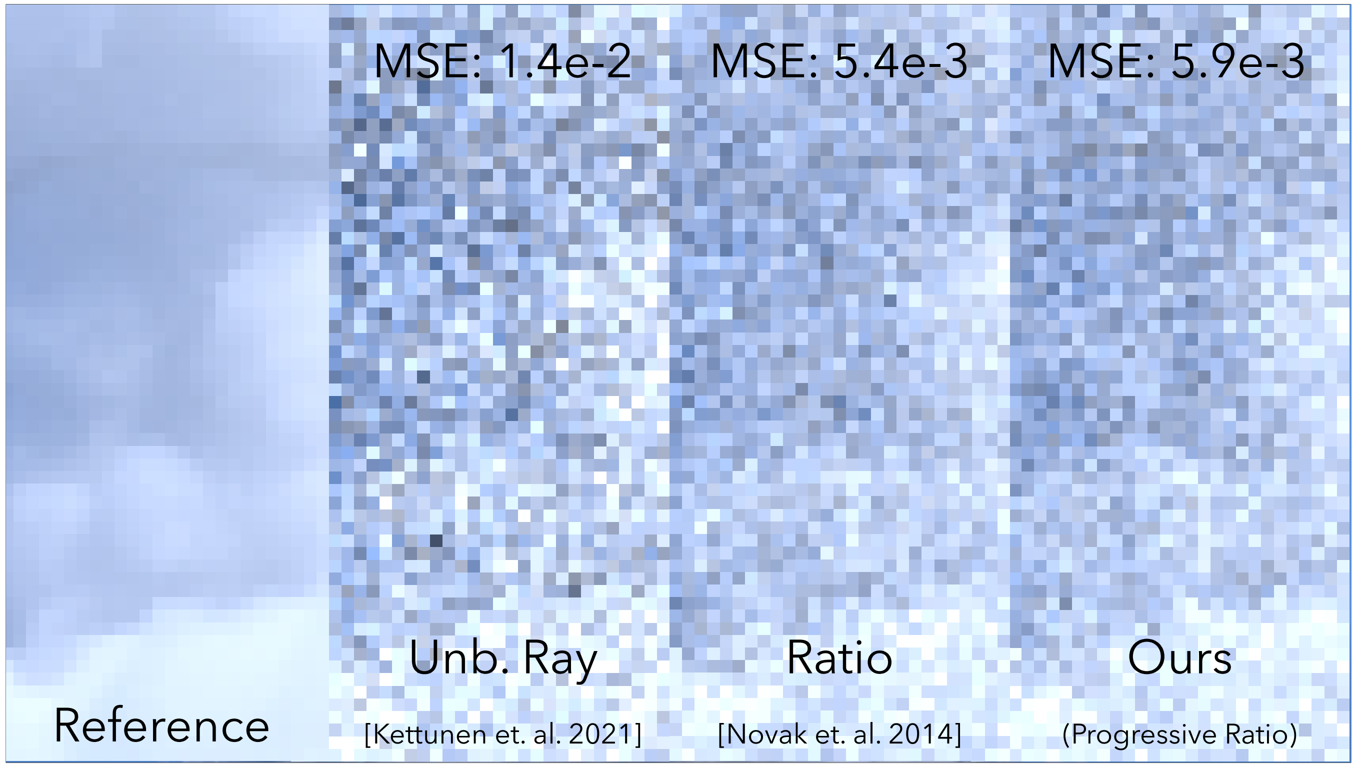



To our technique, which also uses ratio and weighted delta tracking, except with our progressive clamping and updating.  For this scene, our discovered majorants converge to become bounding very quickly so the bias seems visually imperceptible.




In terms of error, our technique converges fairly similarly to ratio tracking.

For the Disney cloud scene, which is a lot more dense, we performed




Equal extinction call comparisons between our method and a few state of the art transmittance estimators. In scenes like this where most of the variance comes from sources outside of transmittance estimation, low cost but higher variance estimators like ratio tracking are still preferable which is why we apply our progressive technique to ratio tracking. The point of this comparison is to show that even in these difficult scenes our progressive technique makes current methods resilient to non-bounding extinctions without taking a significant performance hit.

# Conclusion



In conclusion, we have introduced a progressive method for making most null-scattering techniques resilient to non-bounding majorants. Our method imposes no significant performance loss, requires no major modification to any existing null-scattering algorithm, and can be implemented as a simple abstraction layer on top of a renderers medium interface.

# In the paper

# In the paper

- Full analysis of explosive variance

# In the paper

- Full analysis of explosive variance
- Adaptive ratio tracking

# In the paper

- Full analysis of explosive variance
- Adaptive ratio tracking
- Proofs and convergence rates

# Future work

# Future work

- Residual

# Future work

- Residual
- Better majorant updating

# Future work

- Residual
- Better majorant updating
- Full incorporation into Hyperion


Thank you!