MOMENT TRANSPARENCY

BRIAN SHARPE, SENIOR SOFTWARE ENGINEER
WETA DIGITAL

Gazebo

- Gazebo is Weta Digital's general-purpose real-time renderer
  - OpenGL rasterizer
  - Consumes movie data directly without change.
- Data is often in non-ideal format for real-time rendering
  - Large data (lots of instances, lots of triangles)
  - Arbitrary transparency

Hardware Constraints at Weta Digital

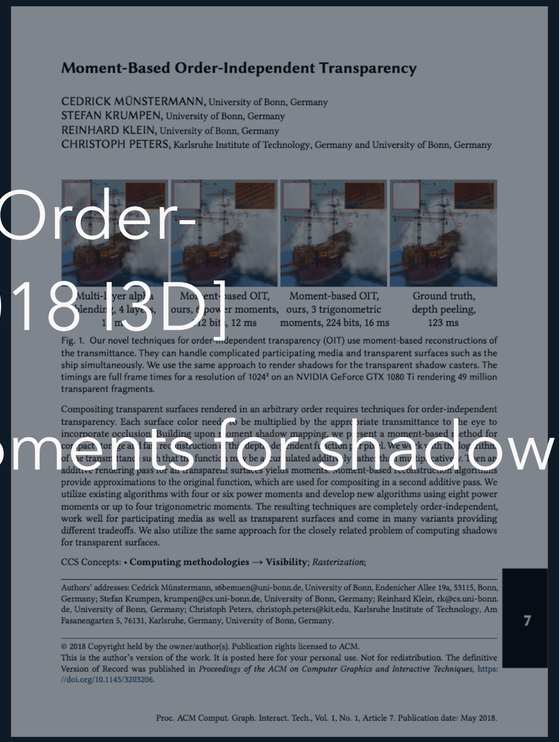
- A range of GPU's at Weta Digital's
- A lot of Kepler's (Desktop and RenderWall)
- Cannot use frame-interlock (raster order views)

Transparency Algorithms

- Hybrid Transparency [Maule et al. 2013]
  - Issues when camera inside particle system
- Weighted-Blended Order-Independent Transparency [McGuire and Bavoli 2013]
  - Overly transparency look
- Fourier Opacity Map Transparency [Jensen and Bavoli 2010]
  - Ringing issues
  - Precision in Z overly coarse (close objects blend together incorrectly)

Moments for OIT

- Decided to extend/improve “Weighted-Blended Order-Independent Transparency” with a per-pixel moments-based transmittance function.
- Déjà vu?
  - Work developed concurrently to “Moments-Based Order-Independent Transparency” [Münstermann et al. 2018 I3D]
  - From the same group that pioneered the use of moments for shadow mapping.
  - Go read their work, its awesome! :)



Moment-Based Order-Independent Transparency

CEDRICK MÜNSTERMANN, University of Bonn, Germany
STEFAN KRUMPEN, University of Bonn, Germany
REINHARD KLEIN, University of Bonn, Germany
CHRISTOPH PETERS, Karlsruhe Institute of Technology, Germany and University of Bonn, Germany

Moment-based OIT, ours, 3 trigonometric moments, 224 bits, 16 ms

Ground truth, depth peeling, 123 ms

Fig. 1. Our novel techniques for order-independent transparency (OIT) use moment-based reconstructions of the transmittance. They can handle complicated participating media and transparent surfaces such as the ship simultaneously. We use the same approach to render shadows for the transparent shadow casters. The timings are full frame times for a resolution of 1024² on an NVIDIA GeForce GTX 1080 Ti rendering 49 million transparent fragments.

Compositing transparent surfaces rendered in an arbitrary order requires techniques for order-independent transparency. Each surface color needs to be multiplied by the appropriate transmittance to the eye to incorporate occlusion. Building upon moment shadow mapping, we present a moment-based method for order-independent transparency such that the function in the accumulated additive pass. Then, an additive rendering pass for transparent surfaces yields moments. Moment-based reconstruction algorithms provide approximations to the original function, which are used for compositing in a second additive pass. We utilize existing algorithms with four or six power moments and develop new algorithms using eight power moments or up to four trigonometric moments. The resulting techniques are completely order-independent, work well for participating media as well as transparent surfaces and come in many variants providing different tradeoffs. We also utilize the same approach for the closely related problem of computing shadows for transparent surfaces.

CCS Concepts: • **Computing methodologies** → **Visibility**; *Rasterization*;

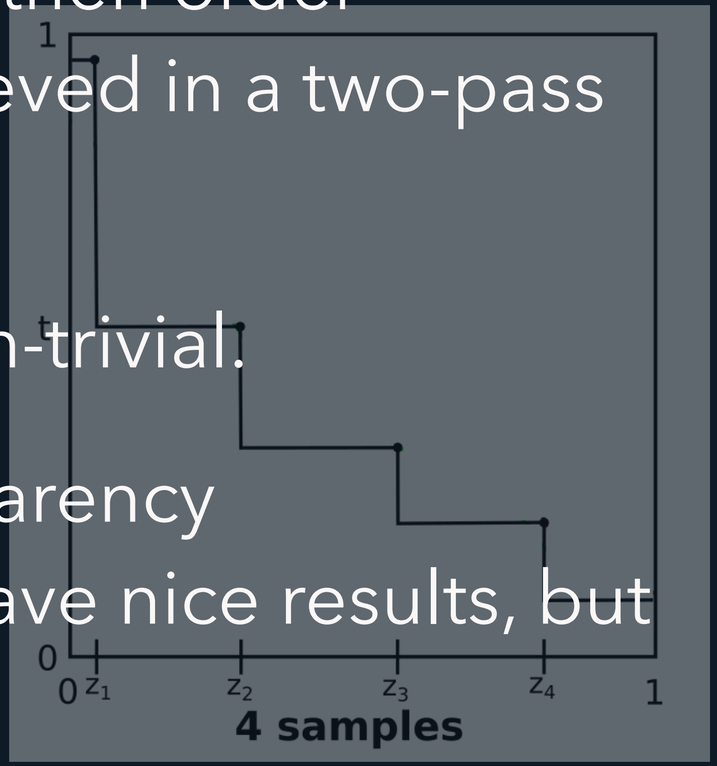
Authors' addresses: Cedrick Münstermann, s6bemuen@uni-bonn.de, University of Bonn, Endenicher Allee 19a, 53115, Bonn, Germany; Stefan Krumpen, krumpen@cs.uni-bonn.de, University of Bonn, Germany; Reinhard Klein, rk@cs.uni-bonn.de, University of Bonn, Germany; Christoph Peters, christoph.peters@kit.edu, Karlsruhe Institute of Technology, Am Fasanengarten 5, 76131, Karlsruhe, Germany, University of Bonn, Germany.

Proc. ACM Comput. Graph. Interact. Tech., Vol. 1, No. 1, Article 7. Publication date: May 2018.

7

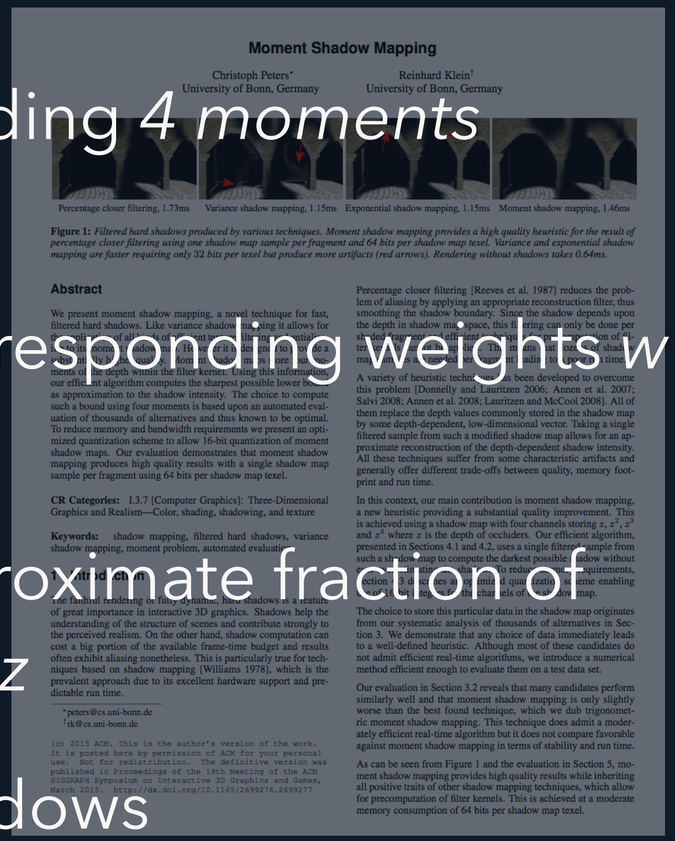
Transmittance Function for OIT

- Adaptive Transparency [Salvi et al. 2011] showed that if a per-pixel *transmittance function* could be precomputed, then order-independent transparency (OIT) could be achieved in a two-pass algorithm.
- Precomputing this transmittance function is non-trivial.
- Weighted-Blended Order-Independent Transparency approximated it with a constant curve, which gave nice results, but in most cases overly transparent.



Moment Shadow Mapping

- Moment Shadow Mapping [Peters and Klein 2015]
- For each pixel in shadow map, store the corresponding *4 moments* $(z, z^2, z^3, z^4)^\top$ for a single depth z
- A filtered sample over a set of these pixels with corresponding weights w is $b = \sum_{i=1}^{n} (z_i, z_i^2, z_i^3, z_i^4)^\top \cdot w_i$
- *Hamburger 4MSM* algorithm used to calculate approximate fraction of depth values closer to light than given input depth z
- Moment shadow mapping uses this for filtered shadows



Moment Shadow Mapping

Christoph Peters* University of Bonn, Germany

Reinhard Klein† University of Bonn, Germany

Percentage closer filtering, 1.73ms | Variance shadow mapping, 1.15ms | Exponential shadow mapping, 1.15ms | Moment shadow mapping, 1.46ms

Figure 1: *Filtered hard shadows produced by various techniques. Moment shadow mapping provides a high quality heuristic for the result of percentage closer filtering using one shadow map sample per fragment and 64 bits per shadow map texel. Variance and exponential shadow mapping are faster requiring only 32 bits per texel but produce more artifacts (red arrows). Rendering without shadows takes 0.64ms.*

Abstract

We present moment shadow mapping, a novel technique for fast, filtered hard shadows. Like variance shadow

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture

Keywords: shadow mapping, filtered hard shadows, variance shadow mapping, moment problem, automated evaluation

Percentage closer filtering [Reeves et al. 1987] reduces the problem of aliasing by applying an appropriate reconstruction filter, thus smoothing the shadow boundary. Since the shadow depends upon the depth in shadow map space, this filtering can only be done per

A variety of heuristic techniques have been developed to overcome this problem [Donnelly and Lauritzen 2006; Annen et al. 2007; Salvi 2008; Annen et al. 2008; Lauritzen and McCool 2008]. All of them replace the depth values commonly stored in the shadow map by some depth-dependent, low-dimensional vector. Taking a single filtered sample from such a modified shadow map allows for an approximate reconstruction of the depth-dependent shadow intensity. All these techniques suffer from some characteristic artifacts and generally offer different trade-offs between quality, memory footprint and run time.

In this context, our main contribution is moment shadow mapping, a new heuristic providing a substantial quality improvement. This is achieved using a shadow map with four channels storing z, z^2, z^3 and z^4 where z is the depth of occluders. Our efficient algorithm, presented in Sections 4.1 and 4.2, uses a single filtered sample from such a shadow map to compute the darkest possible shadow without

The choice to store this particular data in the shadow map originates from our systematic analysis of thousands of alternatives in Section 3. We demonstrate that any choice of data immediately leads to a well-defined heuristic. Although most of these candidates do not admit efficient real-time algorithms, we introduce a numerical method efficient enough to evaluate them on a test data set.

Our evaluation in Section 3.2 reveals that many candidates perform similarly well and that moment shadow mapping is only slightly worse than the best found technique, which we dub trigonometric moment shadow mapping. This technique does admit a moderately efficient real-time algorithm but it does not compare favorable against moment shadow mapping in terms of stability and run time.

As can be seen from Figure 1 and the evaluation in Section 5, moment shadow mapping provides high quality results while inheriting all positive traits of other shadow mapping techniques, which allow for precomputation of filter kernels. This is achieved at a moderate memory consumption of 64 bits per shadow map texel.

*peters@cs.uni-bonn.de
†rk@cs.uni-bonn.de

Moment-based transmittance function

- To adapt transmittance for use with moments and OIT, we move to log space, made easier using terminology from volumetric rendering.
- We can interpret a fragments alpha *a*, as *optical depth d*

$$d = -\log(1 - \alpha)$$

- Optical depth is additive, meaning composition of multiple fragments is a sum

$$d' = \sum_{i=1}^{n} d_i .$$

Moment-based transmittance function

- With this in mind, we can use *optical depth d* as the weight when creating a moments sample

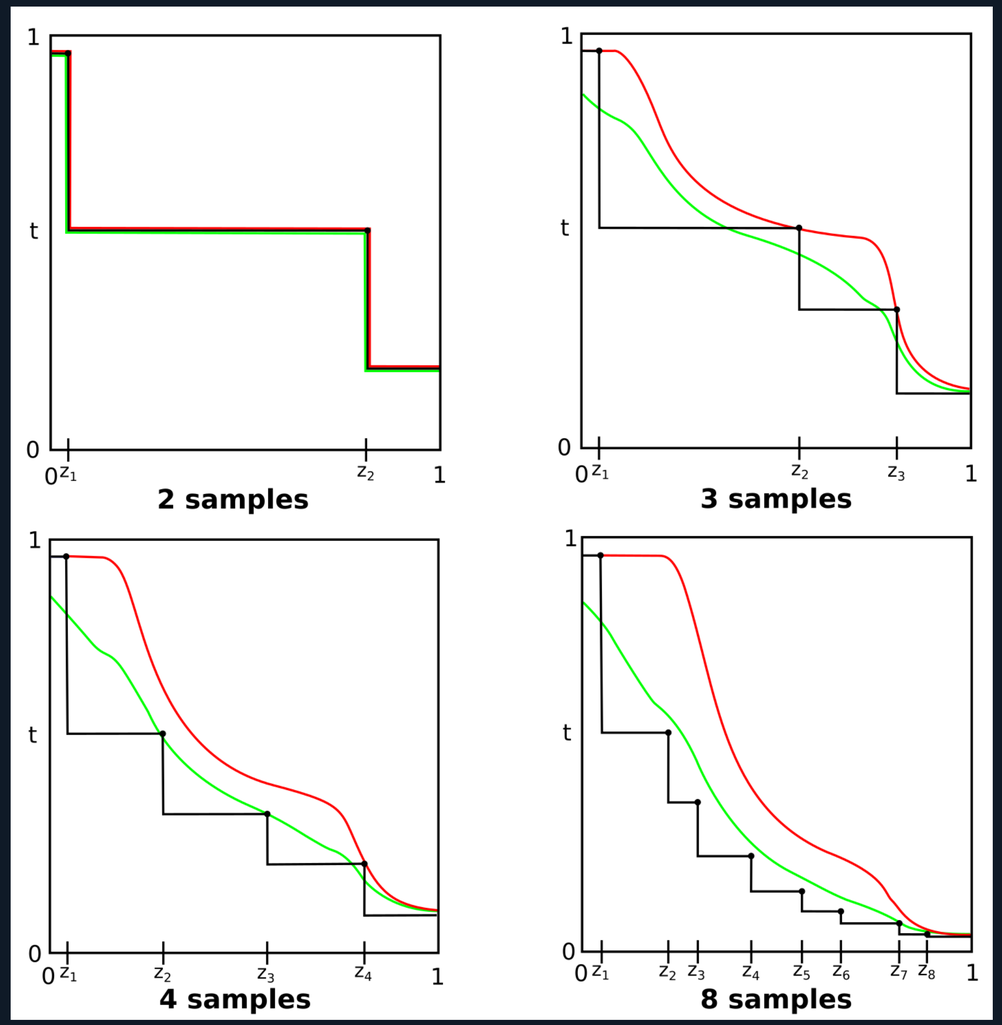
$$b = \sum_{i=1}^{n} (z_i, z_i^2, z_i^3, z_i^4)^\top \cdot d_i$$

- Noting that total optical depth d′ is needed for normalization, we now have a transmittance function stored as 5 floats (4 moments *b* and 1 total optical depth *d′*)
- Reconstruction is via the *Hamburger 4MSM* algorithm

$$T(z) = \exp(-\mathrm{Hamburger}(\frac{b}{d'}, z) \cdot d')$$

Moment-based transmittance function

- Results:
  - Transmittance function = black
  - Moment-based = red
  - Moment-based with overestimation = green
- Moments are a good match for 2 and 3 samples only.
- Overestimation [Peters et al. 2017] is an improvement (25% used in this case) although it overshadows in the front.



Moment Transparency

- Integrate into Weighted-Blended Order-Independent Transparency by replacing its empirically defined transmittance function, with new moments-based version.
- Requires an extra geometry pass
  - Allocate at screen resolution two new buffers
  - RGBA_F32 to capture the moments *b*
  - R_F32 to capture total optical depth *d'*
  - Initialize both to zero, and set blend mode to additive
- Replace *w(z,a)* with moments-based version

$$b = \sum_{i=1}^{n} (z_i, z_i^2, z_i^3, z_i^4)^\top \cdot -\log(1 - \alpha_i)$$

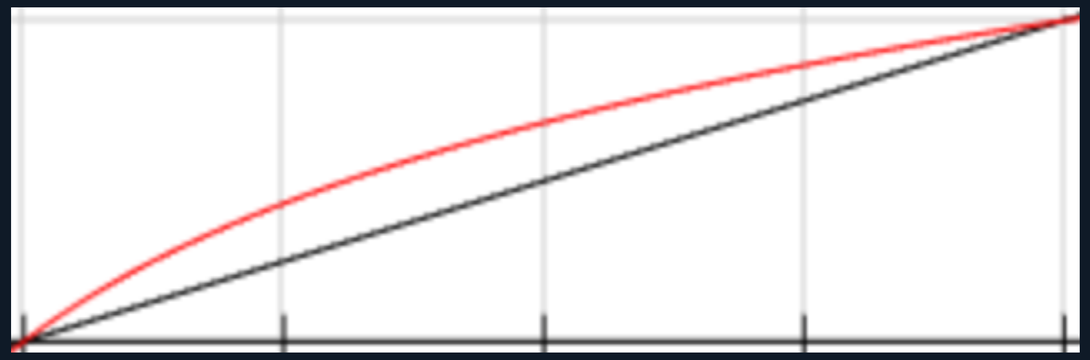
$$d' = \sum_{i=1}^{n} -\log(1 - \alpha_i)$$

$$T(z) = \exp(-\text{Hamburger}(\frac{b}{d'}, z) \cdot d')$$

Moment Transparency

- Large values in z can cause precision issues when accumulating moments.
- To fix this we apply the following logarithmic warp.

$$z' = \frac{\log(z/Z_{near})}{\log(Z_{far}/Z_{near})}$$



Moment Transparency

Weighted-Blended Order-Independent Transparency

Moment Transparency

Ground Truth

Moment Transparency

Alpha Blending

Weighted-Blended
Order-Independent Transparency

Moment Transparency

MOMENT TRANSPARENCY: Improvements

Low Resolution Moments

- Moments can be filtered
- This allows decoupling of the moments buffer resolution from that of the screen (ie low resolution transmittance function)
- Increases performance and reduces memory
- Results:
  - Thin surfaces can have issues (some bleeding, flickering with animation)
  - Large objects almost no issues at all
  - More forgiving than you might think
  - Recommend a little bit (at least 1/2, at Weta we use 1/8)

Low Resolution Moments

Moment Transparency

Moment Transparency LowRes 1/8

Different Formats for Write & Read

- To improve performance, it is possible to use a different format for moments capture (ie write) than used for moments reconstruction (ie read)
- Non-Linearly Quantized Moments (NLQM) [Peters 2017]
  - A special quantized format for storing 4 moments at 16bit
  - No loss observable loss in quality
  - Faster to reconstruct than regular moments (less bandwidth, less arithmetic instructions)
  - Convert captured 32bit moments to NLQM before reconstruction

Two Moments Passes

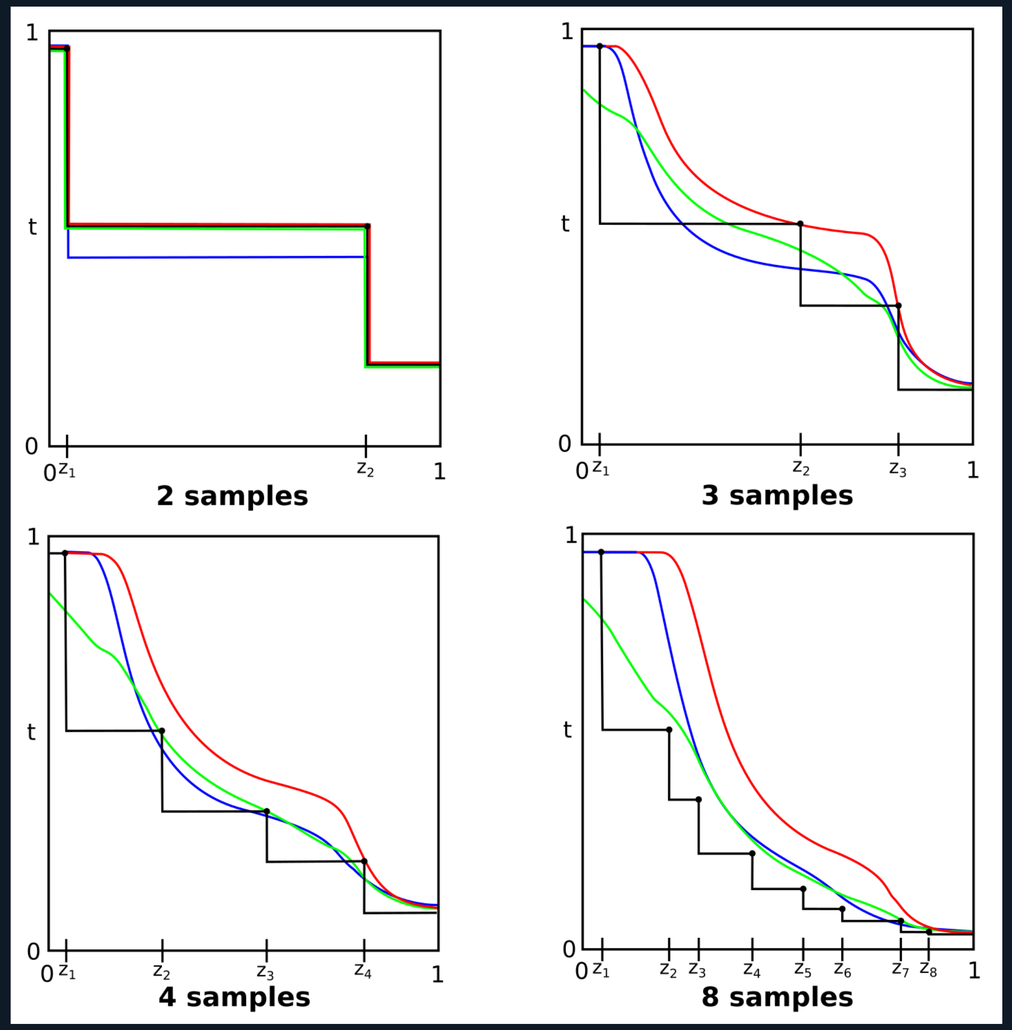
- As seen, regular *Hamburger* can be a poor fit for a transmittance curve with large fragment counts.
- A large part of the problem, is that we're giving equal weight to fragments in the back, as to those in the front, even though they'll contribute less to the final pixel.
- This results in poor compression.
- Overestimation can help, but can also overshadow in the front.

Two Moments Passes

- Capture a second set of moments, while using the first set of moments as an estimate for this occlusion.
- Adds bias in that each fragment occluded by twice the amount it should be.
- In practice bias is offset by the loss of detail of moments reconstruction, which tends to under occlude.

Two Moments Passes

- Results:
  - Transmittance function = black
  - Moment-based = red
  - Moment-based with overestimation = green
  - Two moments passes = blue



Two Moments Passes

Moment Transparency

Two Moments Passes

Ground Truth

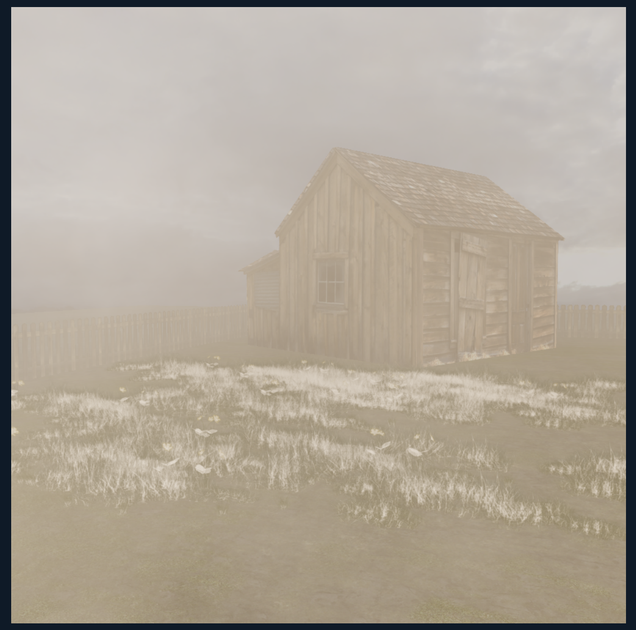
Two Moments Passes

Moment Transparency

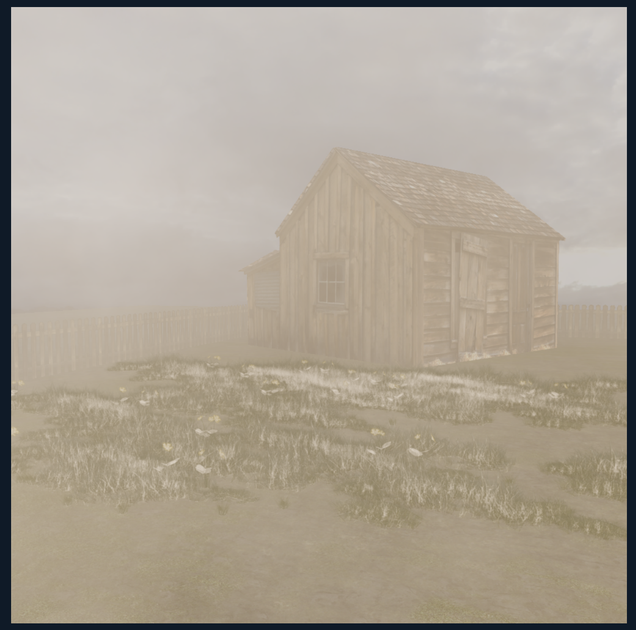
Two Moments Passes

MOMENT TRANSPARENCY: Challenging Cases

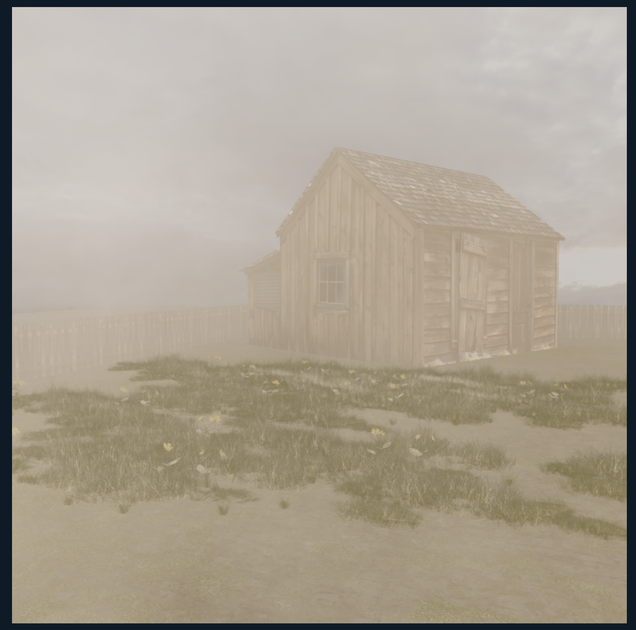
Challenging Cases Remain



Overestimation 25%



Overestimation 17%



Overestimation 10%

Ground Truth

THANK YOU