university of groningen / faculty of science and engineering

Tutorial 3: Ray Tracing, Sampling and (Anti-)Aliasing

Dewi Batista [1]
Dr. Christian Kehl [2]
Juan Diego Cardenas Cartagena, M.Sc.[3]
(j.d.cardenas.cartagena@rug.nl)

Signals and Systems
1B - 2024/2025

[1] Developed Educationa Material
[2] Developed Demo
[3] Revised Tutorial

Link to Demo: wezel.github.io/Virtual-Ray-Tracer/

How we perceive visually

Rays of light shoot out from a given light source, travelling through some environment (your room if the light source is your light bulb, the galaxy if the light source is a star, etc.) and bounce around before they reach your eye.

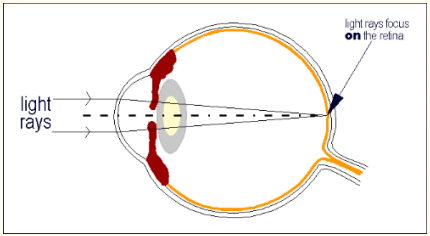

Figure: Ray of light reaching your eye.

Problems with simulating it

The vast majority of rays of light stemming from some light source will not end up reaching your eye - computationally expensive to simulate!

How we perceive visually

Rays of light shoot out from a given light source, travelling through some environment (your room if the light source is your light bulb, the galaxy if the light source is a star, etc.) and bounce around before they reach your eye.

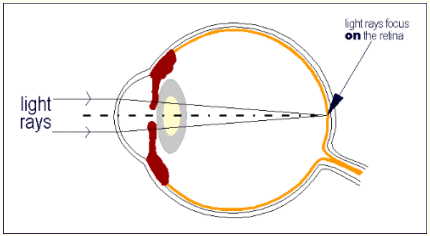

Figure: Ray of light reaching your eye.

Problems with simulating it

The vast majority of rays of light stemming from some light source will not end up reaching your eye - computationally expensive to simulate!

Problems with simulating it

The vast majority of rays of light stemming from some light source will not end up reaching your eye - computationally expensive to simulate!

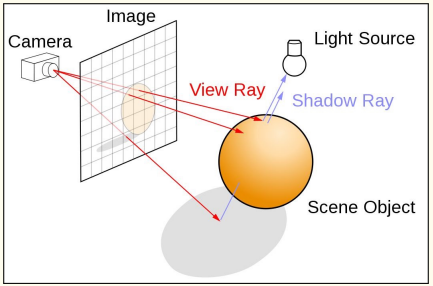

Figure: Ray tracing idea.

Ray tracing idea: Backwards-traverse rays of light from the camera (or eye) to the source of light. We can see which objects said rays of light bounce off (and how) and fill in the image that the camera (or eye) perceives accordingly.

Rendering an image via sampling

- For each pixel in the image, choose N distinct points within the pixel to send rays of light through.
- Each ray will return some colour and lighting information for the pixel according to what said ray bounced off before hitting the light source.
- Average those colour and lighting results to yield the colour and lighting information of the output pixel.

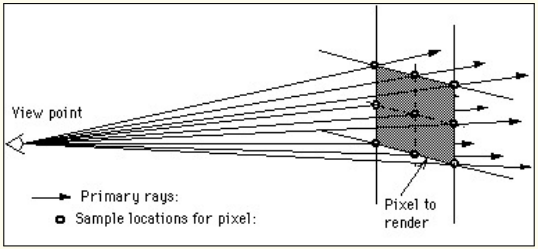

Figure: 9 uniformly sampled rays over a single pixel.

Rendering an image via sampling

- For each pixel in the image, choose N distinct points within the pixel to send rays of light through.
- Each ray will return some colour and lighting information for the pixel according to what said ray bounced off before hitting the light source.
- Average those colour and lighting results to yield the colour and lighting information of the output pixel.

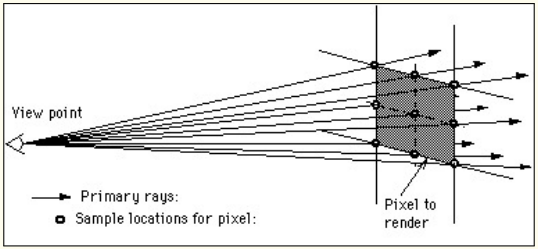

Figure: 9 uniformly sampled rays over a single pixel.

Rendering an image via sampling

- For each pixel in the image, choose N distinct points within the pixel to send rays of light through.
- Each ray will return some colour and lighting information for the pixel according to what said ray bounced off before hitting the light source.
- Average those colour and lighting results to yield the colour and lighting information of the output pixel.

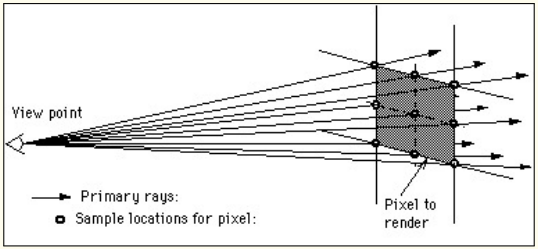

Figure: 9 uniformly sampled rays over a single pixel.

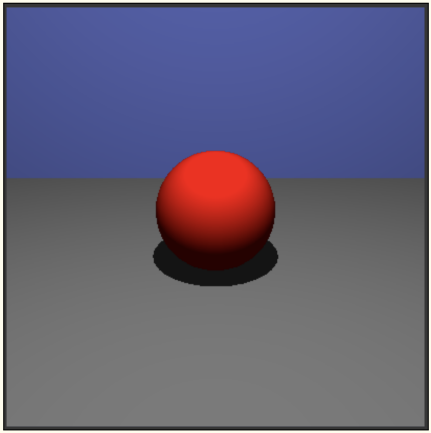

Figure: Image rendered with 1 sample per pixel using ray tracing.

Problem: The shadow below the ball and the ball's boundary are jagged - aliasing is occurring!

Wait, what exactly is aliasing and how does any of this relate to signals?

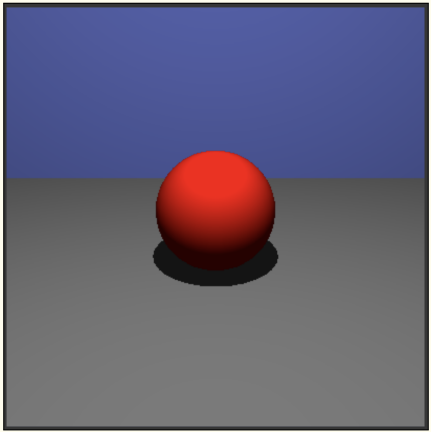

Figure: Image rendered with 1 sample per pixel using ray tracing.

Problem: The shadow below the ball and the ball's boundary are jagged - aliasing is occurring!

Wait, what exactly is aliasing and how does any of this relate to signals?

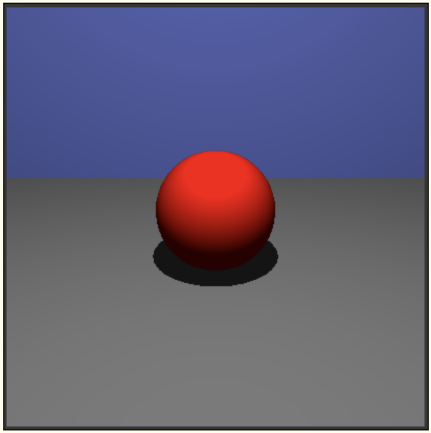

Figure: Image rendered with 1 sample per pixel using ray tracing.

Problem: The shadow below the ball and the ball's boundary are jagged - aliasing is occurring!

Wait, what exactly is aliasing and how does any of this relate to signals?

How does this relate to signals?

Images are signals, so they are subject to the Nyquist-Shannon sampling theorem:

For a continuous signal $f(t)$ with a maximum frequency $f_{\max}$ it can be completely reconstructed from its samples if it is sampled at a rate f_s such that

$$f_s \geqslant 2f_{\max}.$$

In our case, loosely speaking:

- f_s corresponds to the sampling rate - think of the number of rays we send through each pixel of the image
- $f_{\max}$ corresponds to the finest texture of the image - the edge of the shadow below the ball as well as the boundary of the ball itself

Aliasing

When $f_s < 2f_{\max}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

How does this relate to signals?

Images are signals, so they are subject to the Nyquist-Shannon sampling theorem:

For a continuous signal $f(t)$ with a maximum frequency $f_{\max}$ it can be completely reconstructed from its samples if it is sampled at a rate f_s such that

$$f_s \geqslant 2f_{\max}.$$

In our case, loosely speaking:

- f_s corresponds to the sampling rate - think of the number of rays we send through each pixel of the image
- $f_{\max}$ corresponds to the finest texture of the image - the edge of the shadow below the ball as well as the boundary of the ball itself

Aliasing

When $f_s < 2f_{\max}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

How does this relate to signals?

Images are signals, so they are subject to the Nyquist-Shannon sampling theorem:

For a continuous signal $f(t)$ with a maximum frequency $f_{\max}$ it can be completely reconstructed from its samples if it is sampled at a rate f_s such that

$$f_s \geqslant 2f_{\max}.$$

In our case, loosely speaking:

- f_s corresponds to the sampling rate - think of the number of rays we send through each pixel of the image
- $f_{\max}$ corresponds to the finest texture of the image - the edge of the shadow below the ball as well as the boundary of the ball itself

Aliasing

When $f_s < 2f_{\max}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

Aliasing

When $f_s < 2f_{\text{max}}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

Aliasing leads to undesirable results (jagged edges in images for example) so how do we get around it? **Anti-aliasing**.

Anti-aliasing

Techniques that minimise aliasing. The simplest example is increasing the sample rate f_s. This corresponds to taking more samples per pixel in our case.

Before we see the effect of super sampling: what are the drawbacks to anti-aliasing methods? Why not always apply them?

- Computational expense
- Particularly aggressive methods can result in loss of detail (e.g. blurring or softening where it is not desirable)
- Diminishing returns take effect very quickly in practice

Aliasing

When $f_s < 2f_{\text{max}}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

Aliasing leads to undesirable results (jagged edges in images for example) so how do we get around it? **Anti-aliasing**.

Anti-aliasing

Techniques that minimise aliasing. The simplest example is increasing the sample rate f_s. This corresponds to taking more samples per pixel in our case.

Before we see the effect of super sampling: what are the drawbacks to anti-aliasing methods? Why not always apply them?

- Computational expense
- Particularly aggressive methods can result in loss of detail (e.g. blurring or softening where it is not desirable)
- Diminishing returns take effect very quickly in practice

Aliasing

When $f_s < 2f_{\text{max}}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

Aliasing leads to undesirable results (jagged edges in images for example) so how do we get around it? **Anti-aliasing**.

Anti-aliasing

Techniques that minimise aliasing. The simplest example is increasing the sample rate f_s. This corresponds to taking more samples per pixel in our case.

Before we see the effect of super sampling: what are the drawbacks to anti-aliasing methods? Why not always apply them?

- Computational expense
- Particularly aggressive methods can result in loss of detail (e.g. blurring or softening where it is not desirable)
- Diminishing returns take effect very quickly in practice

Aliasing

When $f_s < 2f_{\text{max}}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

Aliasing leads to undesirable results (jagged edges in images for example) so how do we get around it? **Anti-aliasing**.

Anti-aliasing

Techniques that minimise aliasing. The simplest example is increasing the sample rate f_s. This corresponds to taking more samples per pixel in our case.

Before we see the effect of super sampling: what are the drawbacks to anti-aliasing methods? Why not always apply them?

- Computational expense
- Particularly aggressive methods can result in loss of detail (e.g. blurring or softening where it is not desirable)
- Diminishing returns take effect very quickly in practice

Aliasing

When $f_s < 2f_{\text{max}}$, different frequency components become difficult to distinguish in the sampled signal leading to a distorted reconstruction. This is **aliasing**.

Aliasing leads to undesirable results (jagged edges in images for example) so how do we get around it? **Anti-aliasing**.

Anti-aliasing

Techniques that minimise aliasing. The simplest example is increasing the sample rate f_s. This corresponds to taking more samples per pixel in our case.

Before we see the effect of super sampling: what are the drawbacks to anti-aliasing methods? Why not always apply them?

- Computational expense
- Particularly aggressive methods can result in loss of detail (e.g. blurring or softening where it is not desirable)
- Diminishing returns take effect very quickly in practice

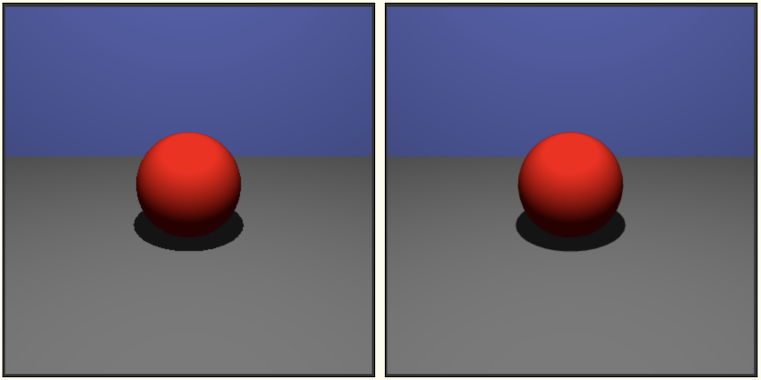

Figure: Rendering with 1 sample per pixel (left) and 3 samples per pixel (right).

Observation: The edge of the shadow beneath the ball as well as the boundary of the ball itself have been ‘smoothed’ out.

In case the difference can not be seen side-to-side:

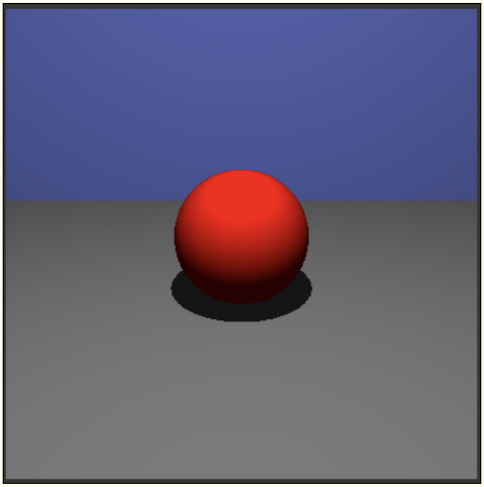

Figure: Rendering with 1 sample per pixel.

In case the difference can not be seen side-to-side:

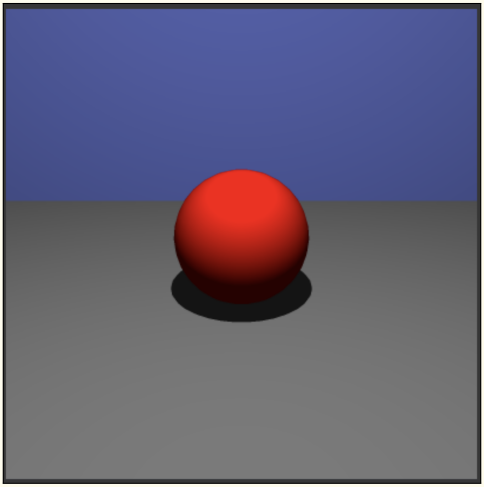

Figure: Rendering with 3 samples per pixel.