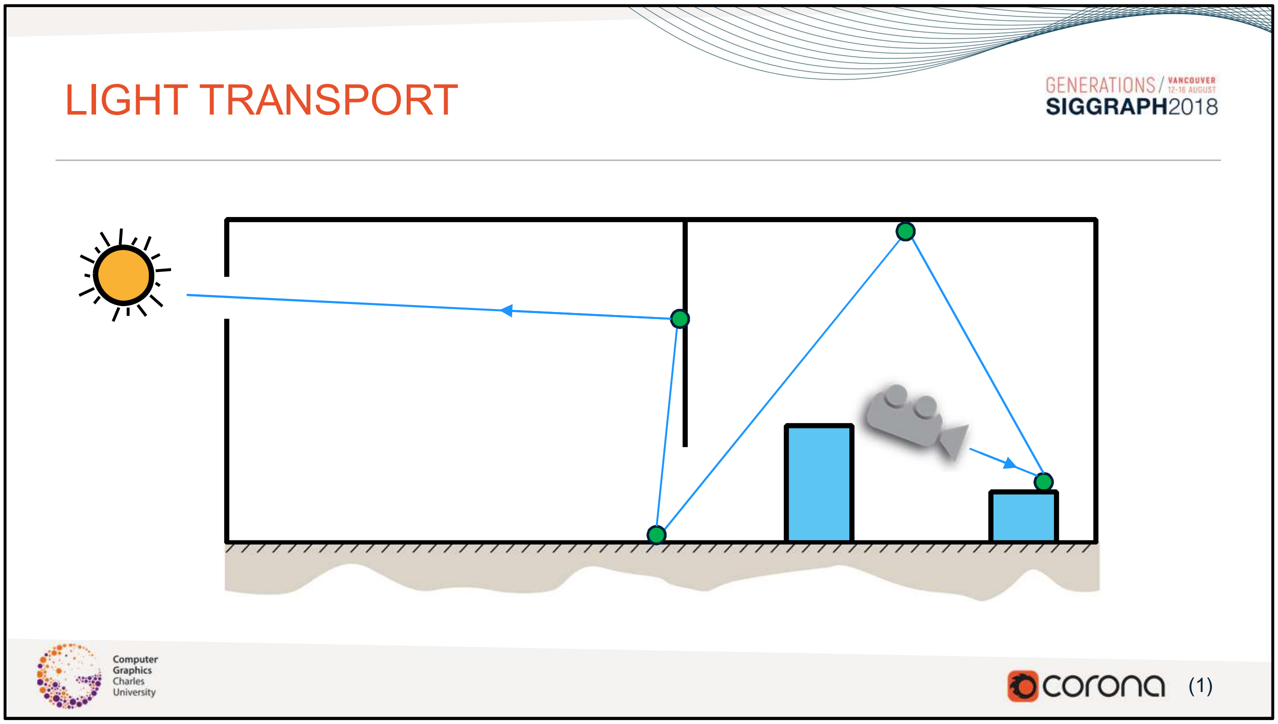

- The objective of this part of the course is to show how some of the fundamental tools from Machine/statistical learning can be used to substantially improve the performance of light transport simulation.




- The basic idea of Monte Carlo light transport simulation is to sample light transport paths that connect light sources to the camera sensors (pixels). Averaging the contributions of these paths then yields a progressively converging image.
- The various light transport simulation algorithms (path tracing, bidirectional path tracing, vertex connection and merging, etc.) differ mainly by the way in which the paths are generated (i.e. stochastically sampled).
- For instance, (unidirectional) path tracing starts from the camera and works by adding one vertex at a time until it reaches a light source. Light tracing, on the other hand, works in the opposite direction, while bidirectional path tracing generates sub-paths from both ends and connects them in the middle.

# TODAYS' RENDERING IS OLD NEWS

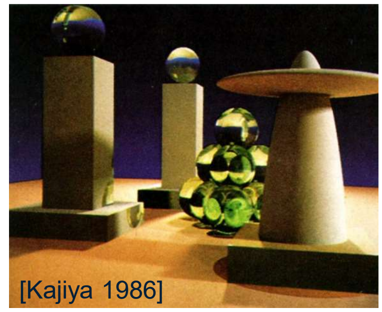
GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- From **Matt Pharr**'s editorial to ACM TOG special issue on production rendering [Pharr 2018]:

***"Today … renderers are … based on … path tracing. Introduced … by Jim Kajiya (1986)."***

*"Many advancements were made … including*
*- more effective light sampling algorithms (Shirley et al. **1996**),*
*- high-quality sampling patterns (Kollig and Keller **2002**), and*
*- multiple importance sampling (Veach and Guibas **1995**),"*

*"… the core ray tracing [got] more efficient (Wald et al. **2001**)."*


[Kajiya 1986]

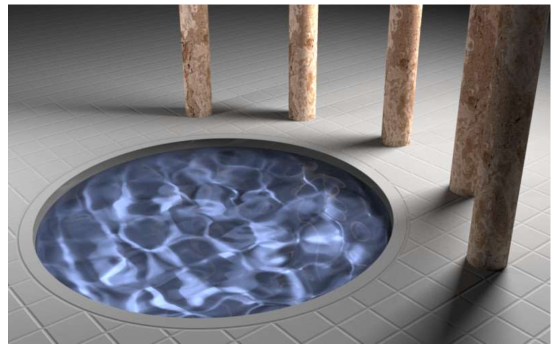


- So which of the many proposed light transport algorithms are actually used in practice?
- It turns out that most renderers are simple *unidirectional path tracers.* Indeed, the underlying light transport technology is 30 years old – with some other ingredients from around 2000.
- This can be nicely illustrated by looking at the references in the above quote from Matt Pharr's editorial to the ACM TOG special issue on production rendering.

# ADVANCED LIGHT TRANSPORT

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- Why are advanced light transport algorithms not used in practice?



Metropolis Light Transport [Veach and Guibas 1997]

Computer Graphics Charles University

corona (3)

- Why is it that none of the more advanced light transport methods, such as Metropolis Light Transport, bidirectional path tracing, or VCM, are used in practice? At least not very often?

## A GOOD LIGHT TRANSPORT ALGORITHM ...

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- … has to be
  - Fast in common scenes
  - Robust & reliable
  - Easy-to-use (no parameters)
  - Interactive & progressive
  - …


Computer Graphics Charles University

corona (4)

- To answer that question, let's assemble a basic checklist of properties that a good light transport algorithm should have, and match it to some of the existing algorithms.
- So here's the list. Of course, this is far from being exhaustive, but it's a good start:
  - Speed – no overhead over the currently accepted solution (path tracing).
  - Robustness – the algorithm must handle scenes with complex lighting and geometry reasonably fast and with no artifacts.
  - Ease-of-use – no technical user parameters are allowed. At most, there can be one speed-quality tradeoff slider.
  - Interactivity – time to first pixel on the screen should be minimized (at most a fraction of a second).
  - Progressivity – image quality steadily improves as the calculation progresses. One can inspect the image and resume rendering if needed at any point in time.

## THE GOOD ALGORITHM CHECKLIST

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- Fast in common scenes
- Robust & reliable
- Easy-to-use (no parameters)
- Interactive & progressive

Computer Graphics Charles University

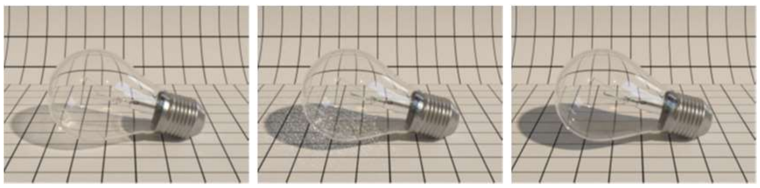
corona (5)

- So that's our checklist, let's see how the existing algorithms fare.

## PATH TRACING

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- [Kajiya 1986, Veach and Guibas 1995, Shirley 1996,…]



Reference rendering (VCM) | Path tracing (no clamping) | Path tracing (with clamping)

- Fast in common scenes
- ~~Robust & reliable~~
- Easy-to-use (no parameters)
- Interactive & progressive

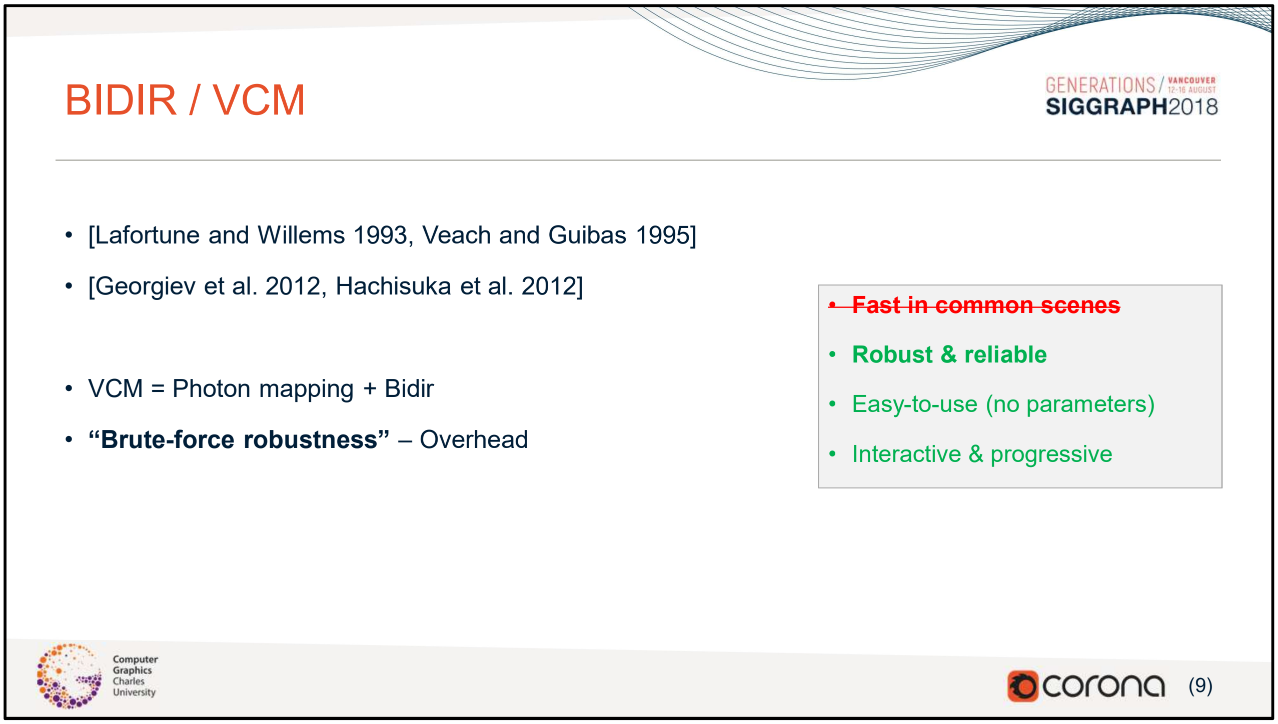
Computer Graphics Charles University

corona (6)

- Simple unidirectional path tracing – the technology used in most today's renderers – checks most of the boxes!
- The only issue is: The algorithm does not handle complex light transport, notably caustics and strong, concentrated indirect illumination.
- In other words, PT is not robust in the presence of complex light transport.
- Instead of a caustic, path tracing will generate just a bunch of fireflies (bright pixels).
- These can fortunately be removed by selective energy clamping (which is what everyone does in practice), so the image in the end looks ok (though it is far from being correct).
- And that's, I believe, almost the whole story of why path tracing is so incredibly popular in practice (code simplicity and maintainability also plays an important role).


- Ok, so path tracing is reasonably good, but it is not good enough.
- To motivate further development of light transport algorithms, let us define what constitutes the ‘ideal’ algorithm that we strive to achieve.
- The great challenge goes as follows: develop a light transport algorithm that renders a Cornell box at least as fast as a unidirectional path tracer (if not much faster).
- But at the same time, the same algorithm should be able to render complex scenes like the ones above in a reasonable amount of time, with no artifacts, without setting a single user parameter, interactively, and progressively.
- We are definitely not there yet as of today.


- One attempt at improving the robustness of path tracing is our Vertex Connection and Merging (VCM) algorithm [Georgiev et al. 2012, Hachisuka et al. 2012].
- VCM achieves robustness by combining bidirectional path tracing (BPT) with photon mapping.
- By doing that, it can now robustly render complex indirect illumination such as in the scene shown above.




- VCM achieves robustness by combining BPT with photon mapping, but it does that at the cost of introducing significant overhead in simple scenes.
- So it addresses one issue (robustness) at the cost of making another issue much worse (overhead).
- I call this approach “brute-force robustness” and I believe it is not the right way to go (“lightweight robustness” is what we should be striving for – i.e. robustness without unnecessary overhead).

# METROPOLIS LIGHT TRANSPORT

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

MLT + Manifold exploration [Jakob and Marschner 2012]

Reference

Computer Graphics Charles University

corona (10)

- Another attempt at handling complex light transport is the Metropolis Light Transport (MLT) algorithm.
- For some time, MLT kept the aura of and the promise of being the ‘ultimate light transport solution’.
- In fact, still around 2011/2012, we had hard time publishing our light transport research work because some reviewers believed that MLT had already solved the problem.
- Fortunately, Wenzel Jakob has released his MLT implementation in Mitsuba, which made it possible to see for oneself just how disappointing the performance of MLT really is in practice.
- MLT’s main problem is uneven convergence, lots of nasty image artifacts, temporal instability – illustrated in the above video.

# METROPOLIS LIGHT TRANSPORT



- [Veach and Guibas 1997, …]
- Uneven convergence, temporal instability

- ~~Fast in common scenes~~
- ~~Robust & reliable~~
- Easy-to-use (no parameters)
- Interactive & progressive


# DESIGNING THE ULTIMATE PRACTICAL ALGORITHM

# THE ULTIMATE LIGHT TRANSPORT ALGORITHM

GENERATIONS / VANCOUVER 12-16 AUGUST SIGGRAPH2018

- Start off from PT
  - because it ticks most of the boxes
- Address its problems
- Root of the problem: **lack of information** in sampling decisions

- Fast in common scenes
- Robust & reliable
- Easy-to-use (no parameters)
- Interactive & progressive

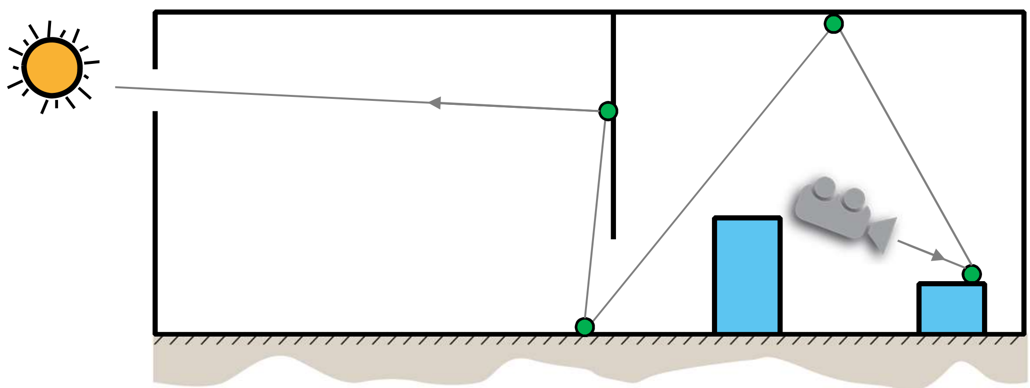
Computer Graphics Charles University

corona (13)

- Ok, so we've seen that the previous attempts at making path tracing robust were not completely successful.
- Let us make our own attempt.

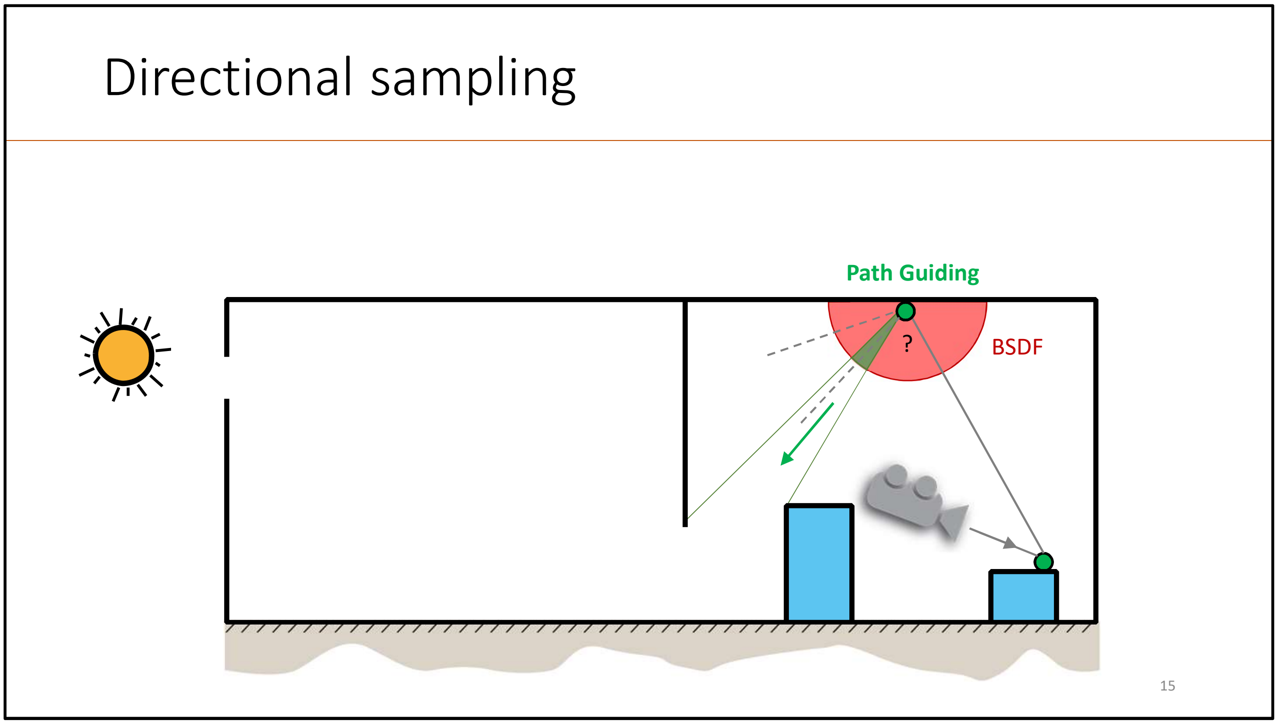
- Looking at the checklist of the desirable properties, the one algorithm that is the closest to having all of them checked is the simple unidirectional path tracing.
- So let's use path tracing as the basis for designing the `ultimate solution'. Let us identify the root of the path tracing's problems and design a solution that addresses specifically these problems, without introducing other issues.
- And with only a little bit of simplification, we can say that the root of the problem in path tracing is the **lack of relevant information for the sampling decisions** used to construct light transport paths. Let us elaborate…

# Path sampling in unidirectional path tracing



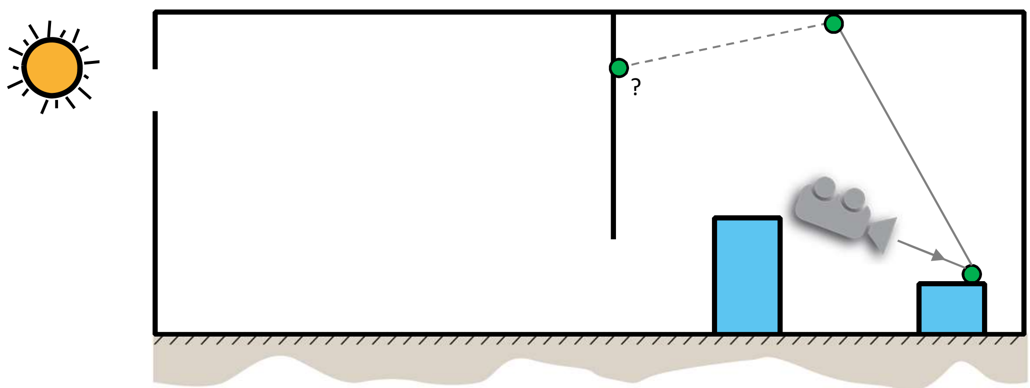
14

- As mentioned earlier, Monte Carlo light transport algorithms rely on random sampling of transport paths connecting the light sources to the camera sensors.
- Path tracing does this by starting from the camera, adding one vertex at a time, **hoping** this process it will eventually allow to reach the light source.



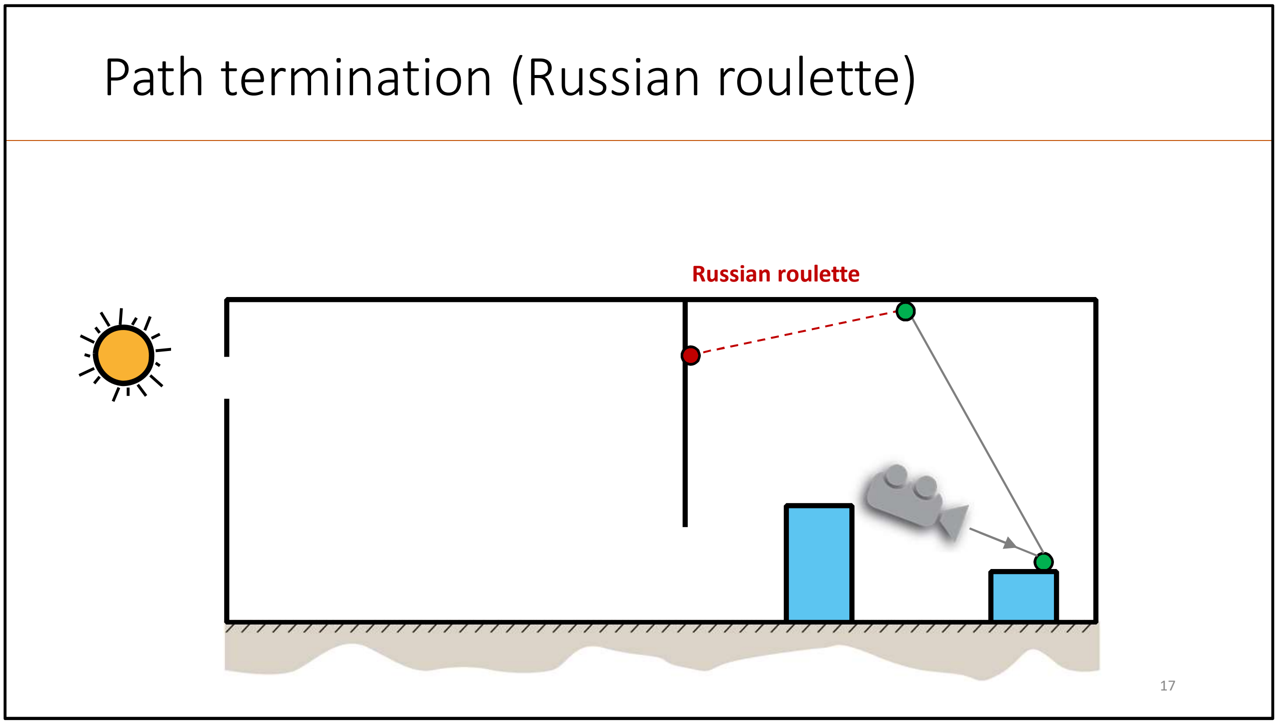

- Path construction in path tracing consists of several types of randomized (stochastic) sampling decisions.
- The first fundamental sampling decision is the **direction sampling**: given the last vertex of a partially constructed path, we add a new path by randomly sampling a direction and shooting a ray.
- In a vanilla path tracer, the sampling distribution would be proportional to the bidirectional scattering distribution function (BSDF), which describes the reflection profile of a material. The use of such a sampling distribution means to shoot rays preferably to directions corresponding to a large throughput of the local BSDF.
- Whether or not such directions actually lead toward the light sources, the path tracer does not know – it can only hope it is the case. (3D artists are aware of this limitation, so they make sure to construct their scenes in such a way that it will indeed often be the case. But not always.)
- Consider the path vertex highlighted on the slide.
- Since the BSDF at that point is diffuse, we have a fairly small chance to sample a reasonable direction (that will lead toward a light source), because the subset of “good” directions lies within a small solid angle depicted in green.
- But the path tracer has no way of knowing that light will be coming from that cone – it is **lacking the relevant information**.

# Path termination (Russian roulette)

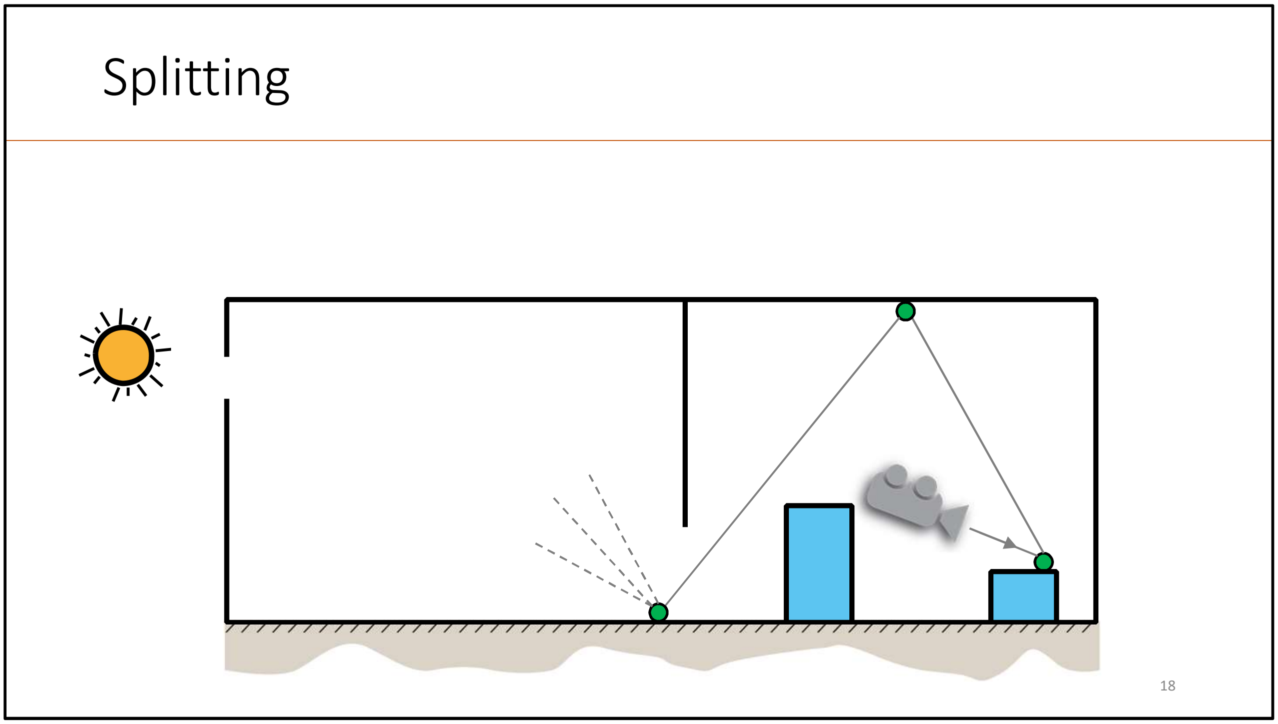



16

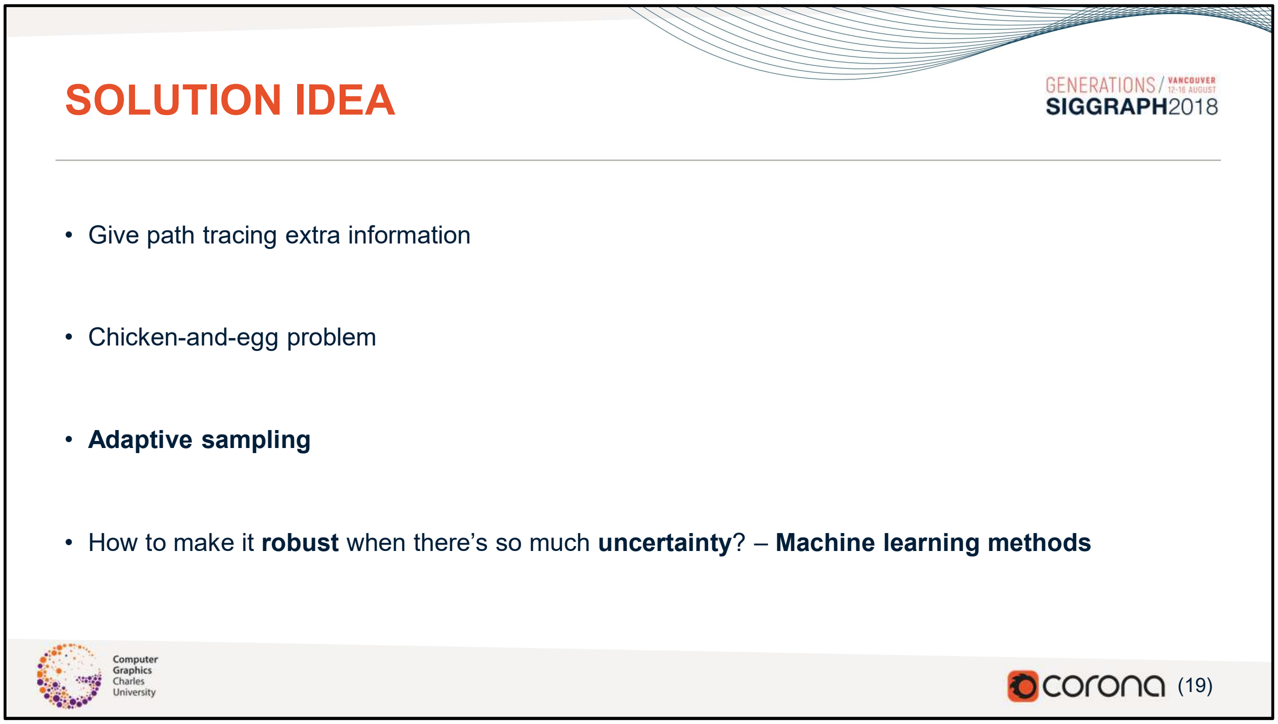
- Another important sampling decision in the path tracer is randomized **path termination** using Russian roulette.



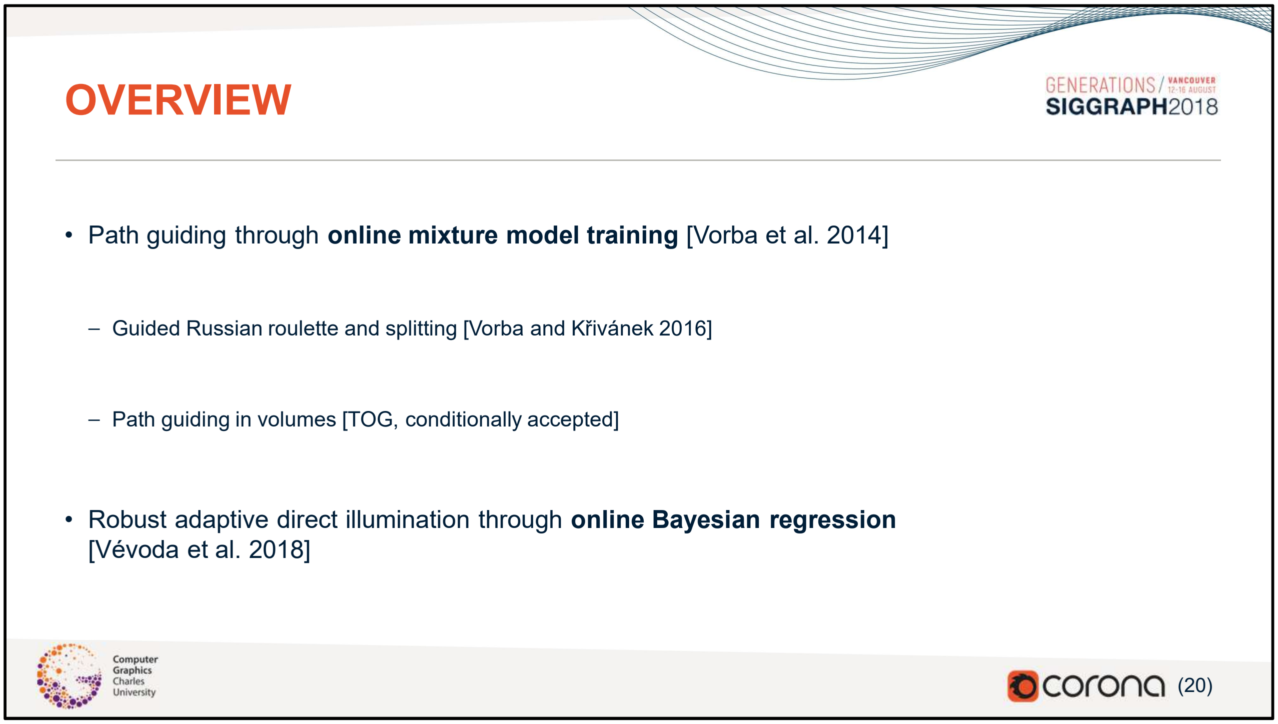

- The intuitive idea is that if the partially generated path happens to reach a dimly illuminated region of the scene, the path is unlikely to contribute significantly to the resulting image and we should not waste time on its tracing.
- Once again, the path tracer cannot make that intuitive decision because it does not know upfront how much light will reach which scene region (that's what the poor old path tracer is trying to compute in the first place).




- Similarly, if the path construction reaches a well-illuminated region of the scene, it is more likely that the path will be able to reach the light source and it pays off to **split** it into several independent trajectories.
- And once again, the path tracer needs some information about illumination – that it normally does not have – to make that decision.




- Ok, so hopefully by now, the reader is convinced that the root of the problem in path tracing is sub-optimality of sampling decisions due to the lack of information about the illumination of the scene.
- The solution of the problem seems trivial: let's just give the path tracer the missing information.
- But computing the distribution of light in the scene is what the path tracer is trying to do in the first place. And if we knew it, there would be nothing left for the path tracer to do.
- How can we break this chicken-and-egg problem?
- One answer is adaptive sampling – we can gather the relevant information from the samples used in the rendering itself.
- The fundamental problem in this context is: How can we extract reliable information from the Monte Carlo samples, given that they are contaminated by so much variance, i.e. **uncertainty**.
- Fortunately, **Machine learning** has been dealing with extraction and generalization of information from **uncertain and noisy data** for several decades (cf. [Bishop 2006]).
- And we can take advantage of these tools in light transport as well.

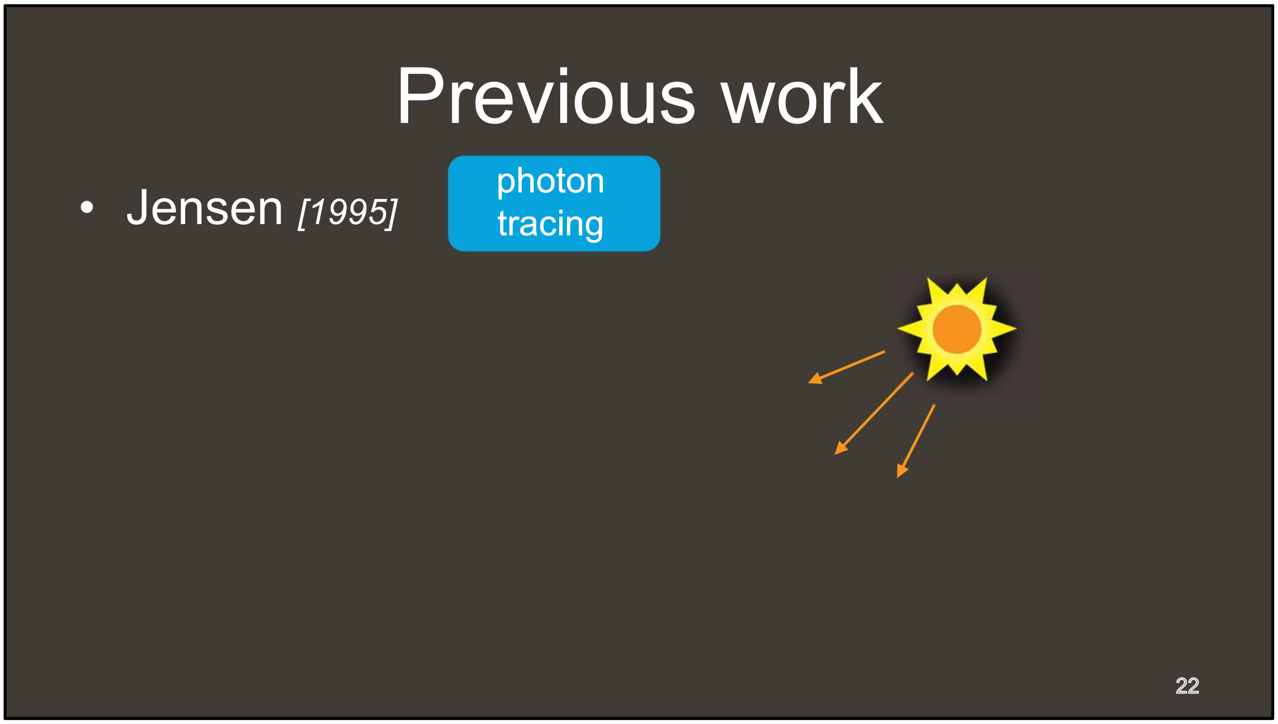



- In the rest of the talk, I present several of our works that aim at improving path sampling through Machine learning (ML) methods.
- First, we have introduced the use of **online training of mixture models** and applied it to **path guiding** (i.e. directional sampling) on surfaces [Vorba et al. 2014], for Russian roulette and path splitting [Vorba and Křivánek 2016], and recently also for path guiding in volumes (participating media) [Herholz et al. 201?].
- Another tool from the ML repertoire is **Bayesian regression**, which we have recently applied to the problem of robust adaptive direct illumination sampling [Vévoda et al. 2018].

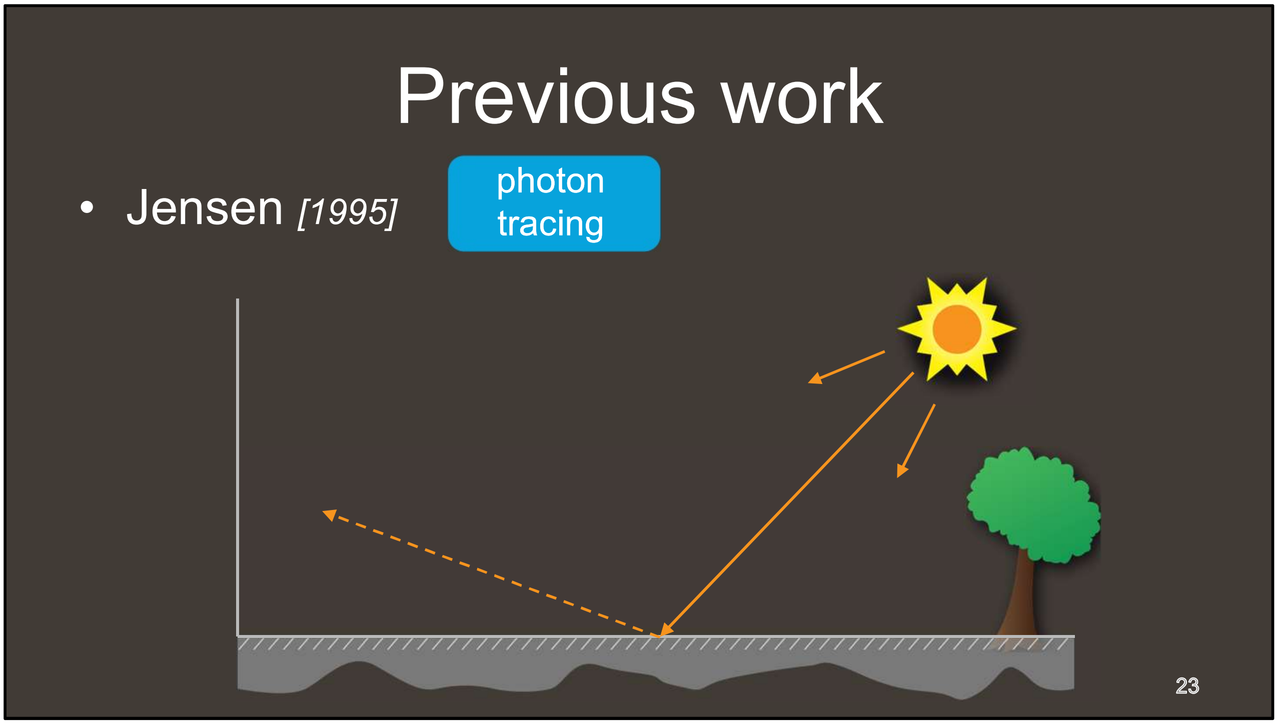

# PATH GUIDING

Vorba et al. – ACM SIGGRAPH 2014

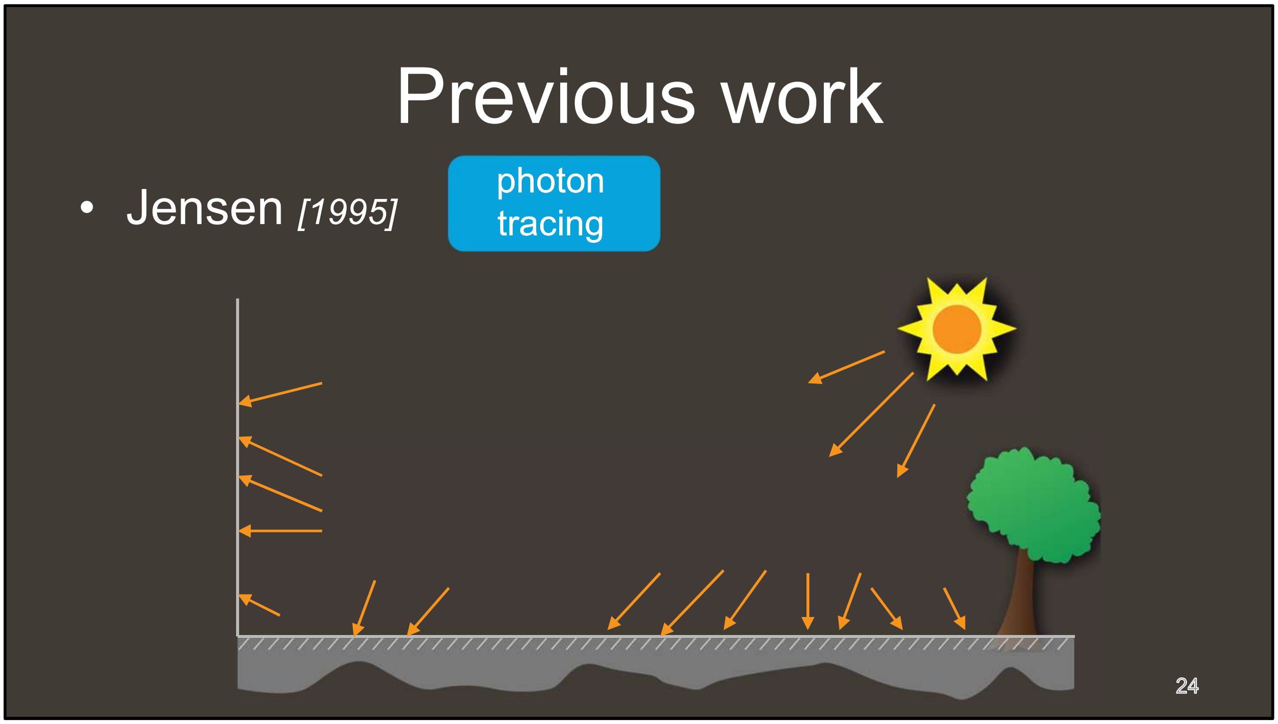
- Let us start by our work on path guiding through parametric mixture model learning.



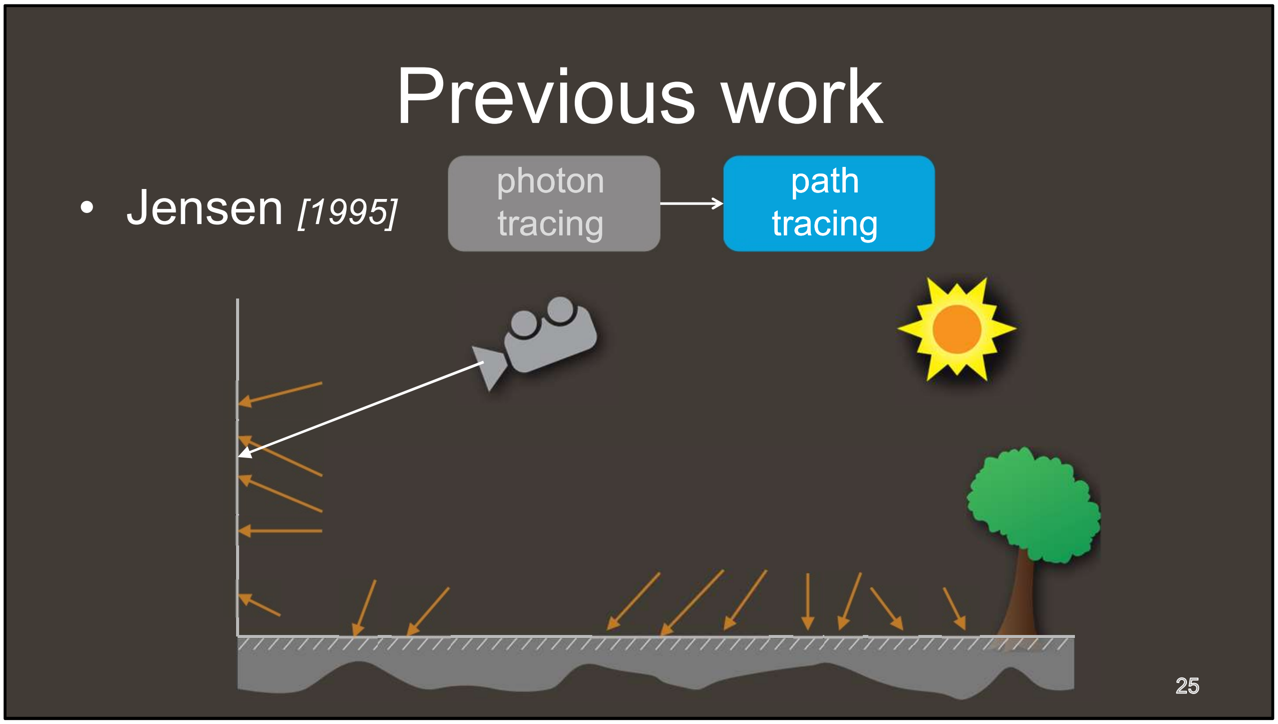

- We build on the idea of path guiding, first proposed by Henrik Wann Jensen in his 1995 paper.
- In his work, he uses light particles – that is photons – to reconstruct the distribution of incoming radiance ($L_i$) at a point in the scene.



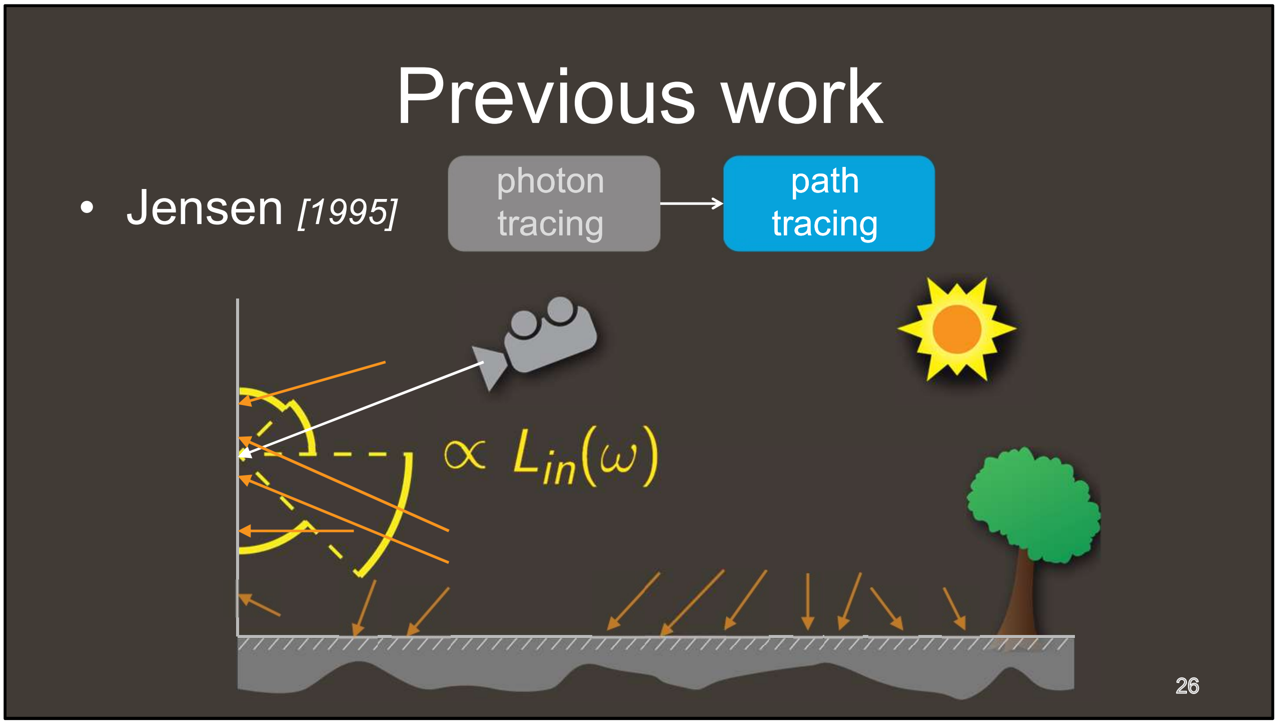

- He traces the photons in a preprocessing phase…



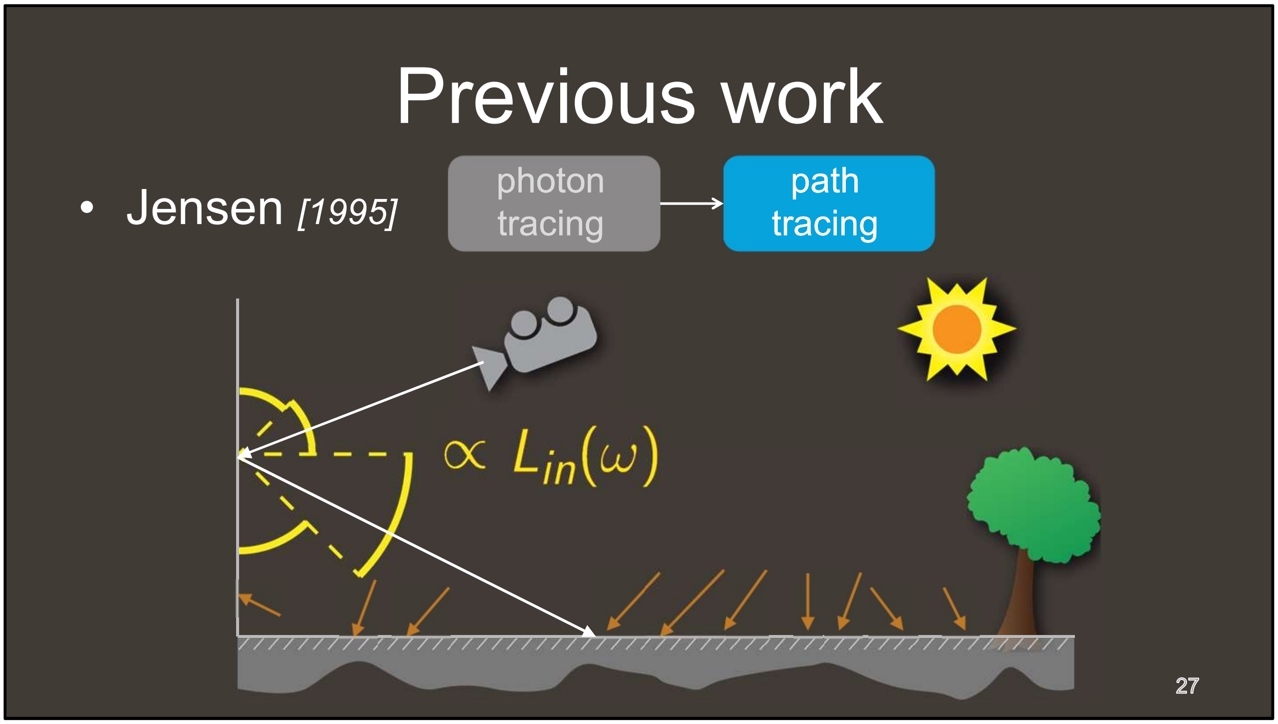

• … and stores them on surfaces for later use for path guiding in path-tracing.




• During the path-tracing, …



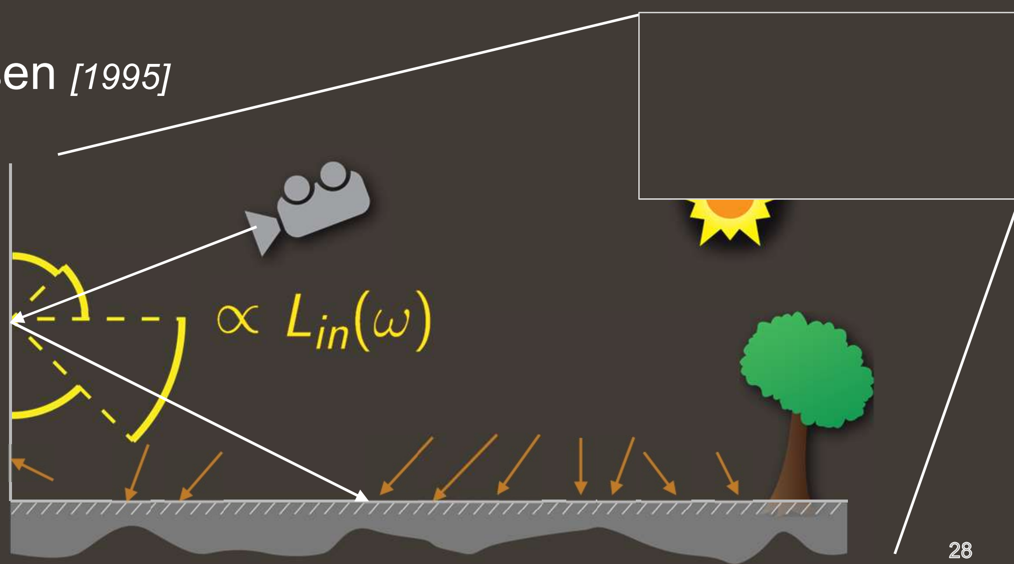

- …, he reconstructs the directional distributions of radiance from nearest photons…




- … and uses the distributions for sampling of reflected directions.
- Note that contrary to using the photons directly to estimate the scene radiance, using them for path guiding produces an unbiased image.

# Previous work

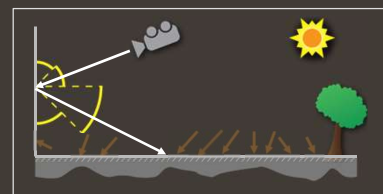
- Jensen *[1995]*



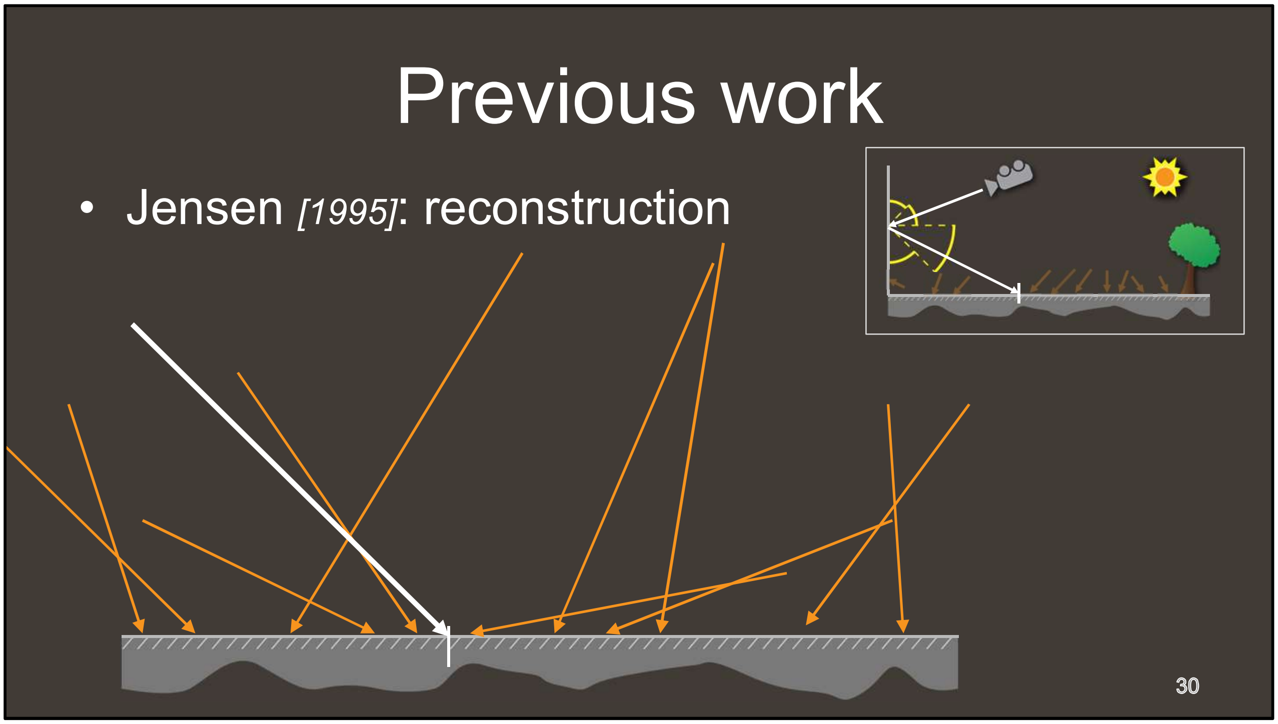

28

# Previous work

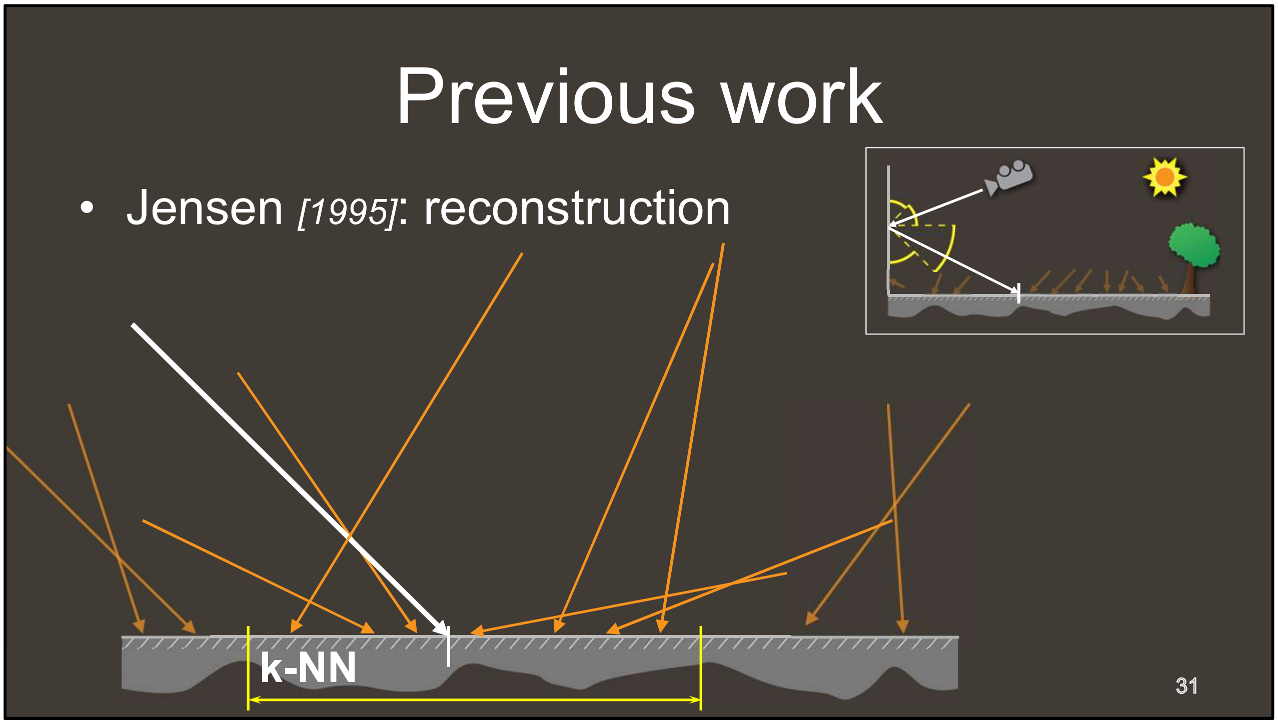
- Jensen *[1995]*



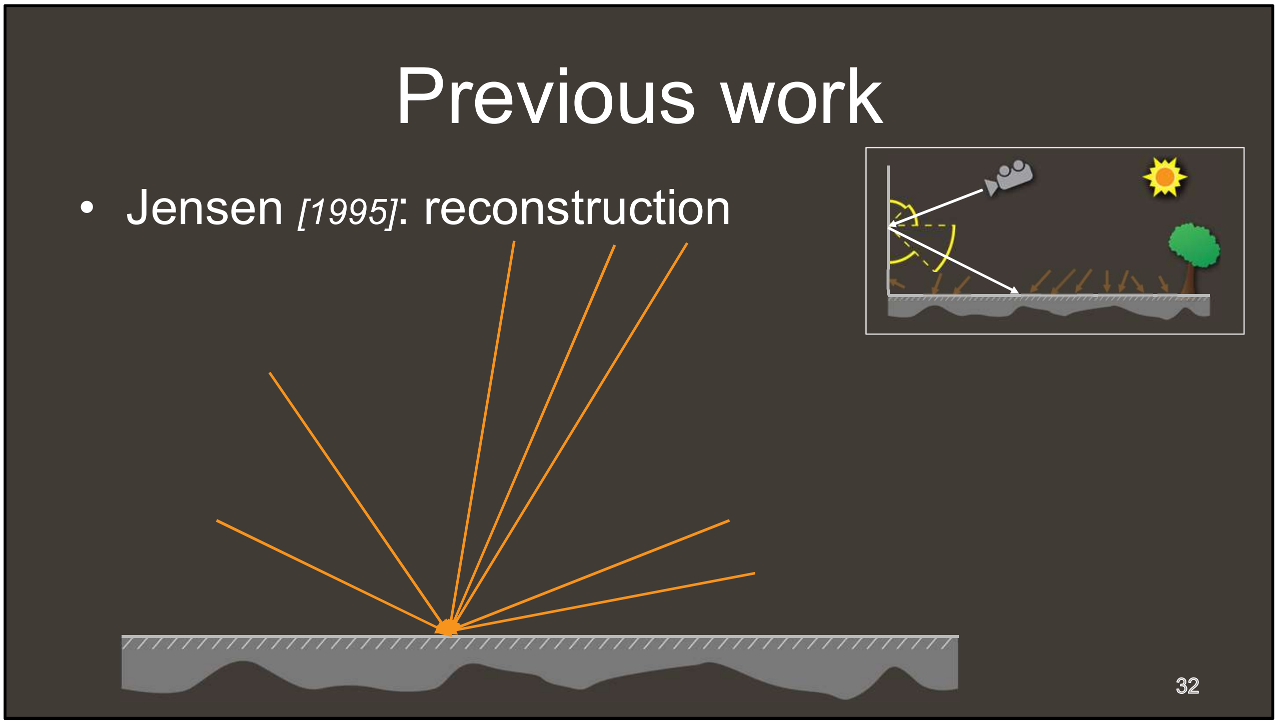
29



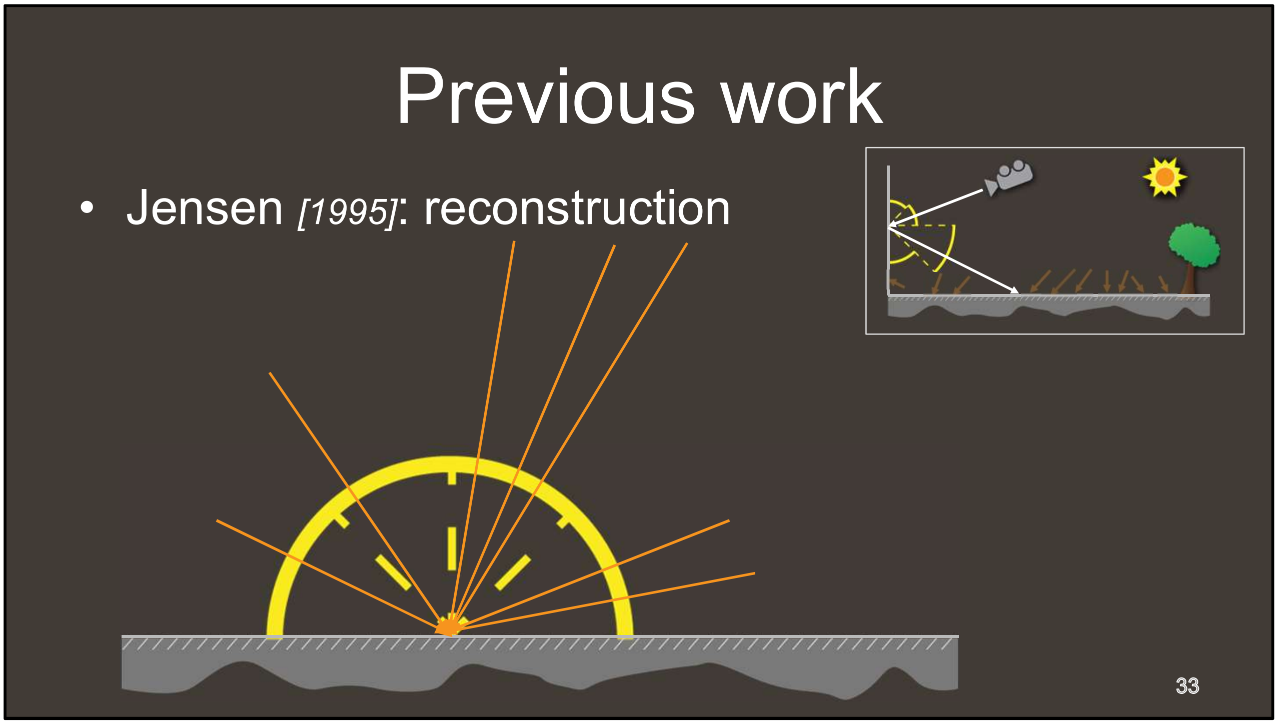

- The reconstruction of a directional sampling distribution at a given point starts with a search for nearest photons.



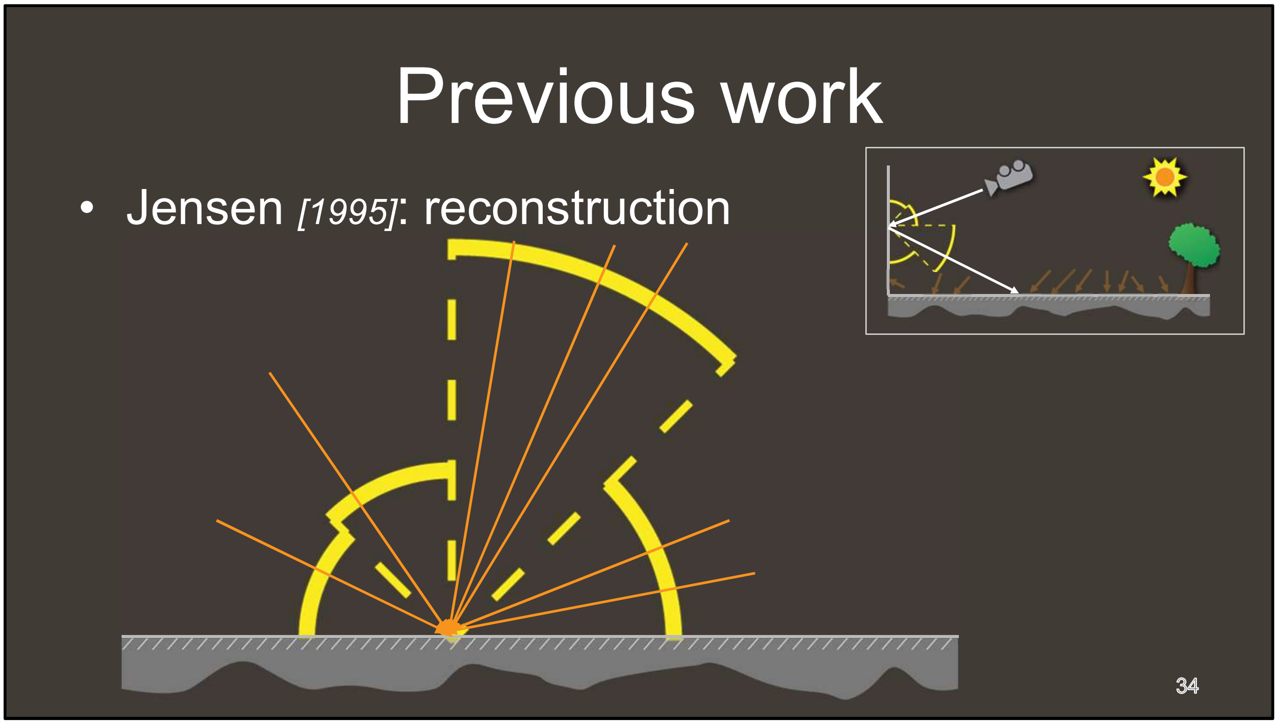

- Since he is only interested in reconstructing a directional distribution, he makes the assumption that all the photons hit the surface at the point of interest.




- To reconstruct the directional probability distribution …




- … they discretize the hemisphere into equal-sized bins and count the number of particle directions falling into each bin.



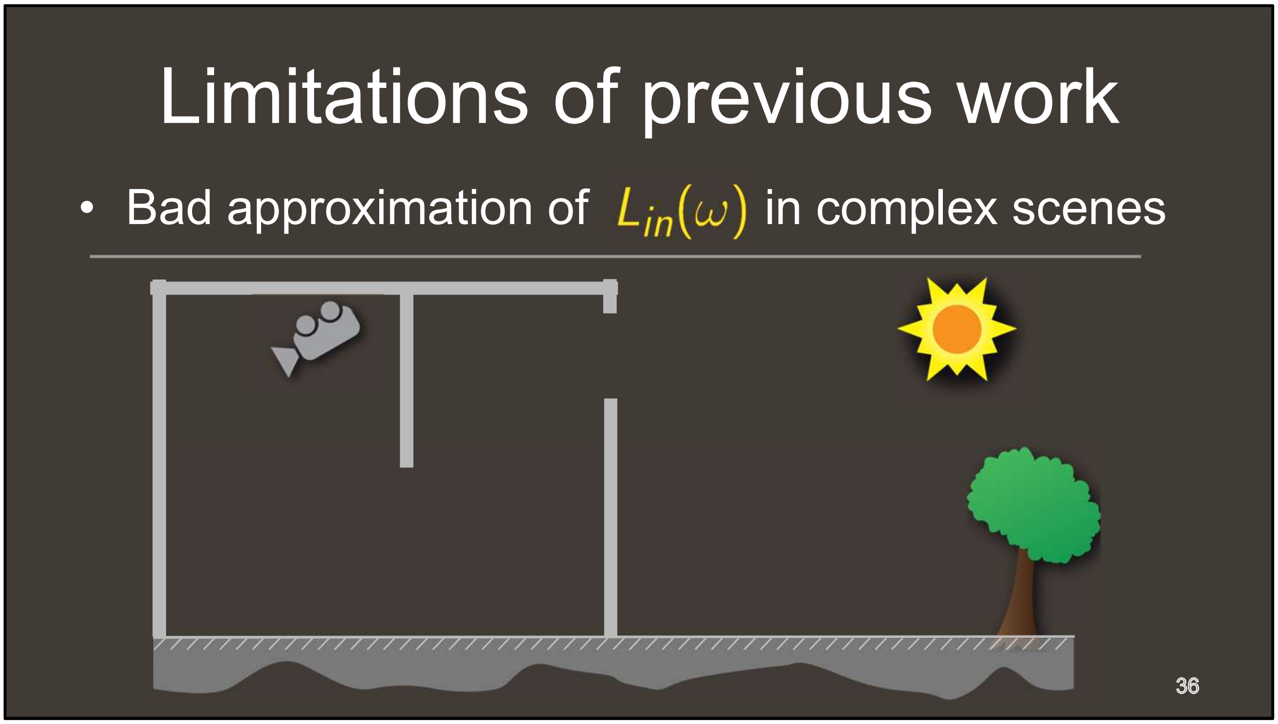

- The result is a histogram over the hemisphere.
- However, histogram is known to be a poor density estimation method prone to over and under fitting.

# Limitations of previous work

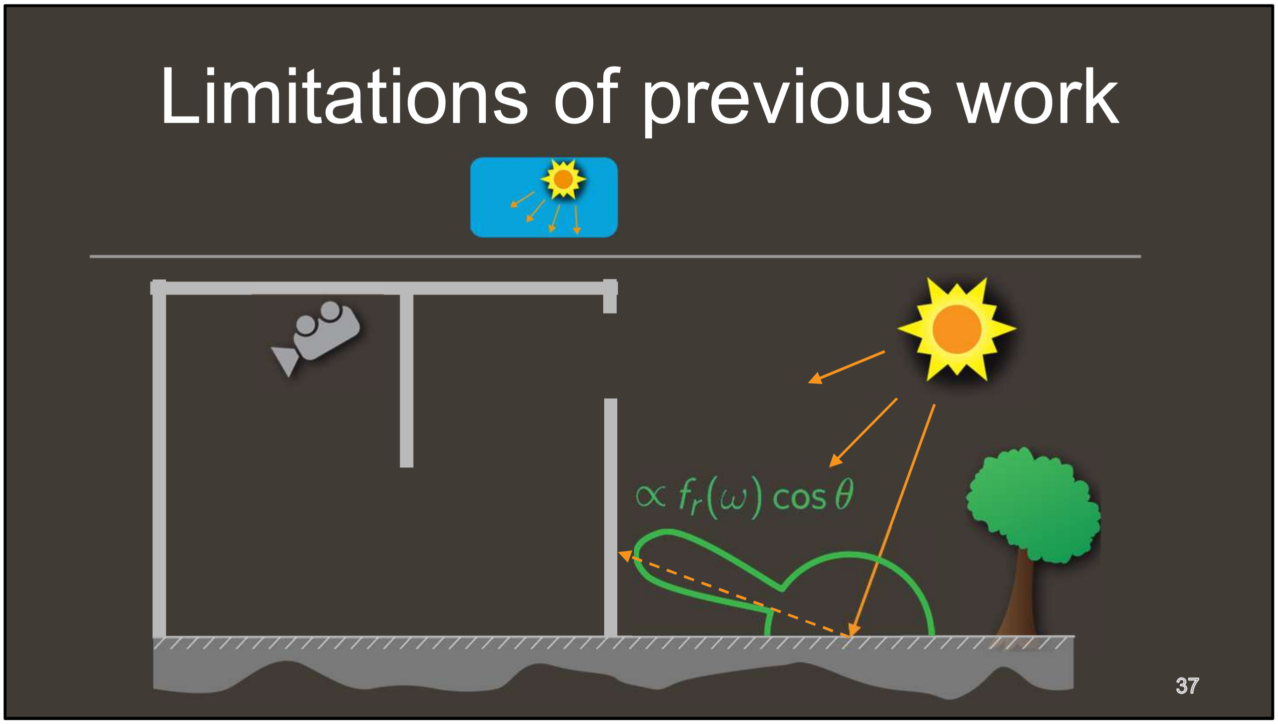
- Bad approximation of $L_{in}(\omega)$ in complex scenes

35

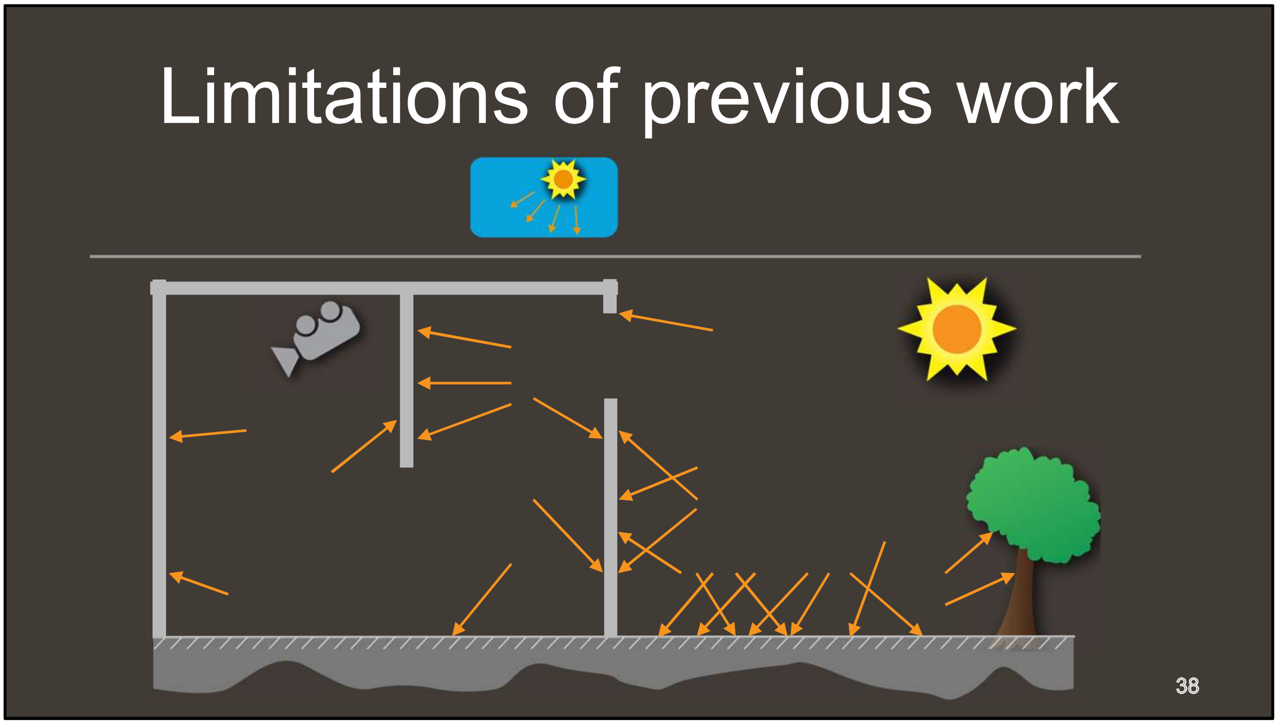
- Although Jensen's method can use the incident radiance term for path guiding, the method is still not robust and indeed, it struggles in a wide range of scenes.



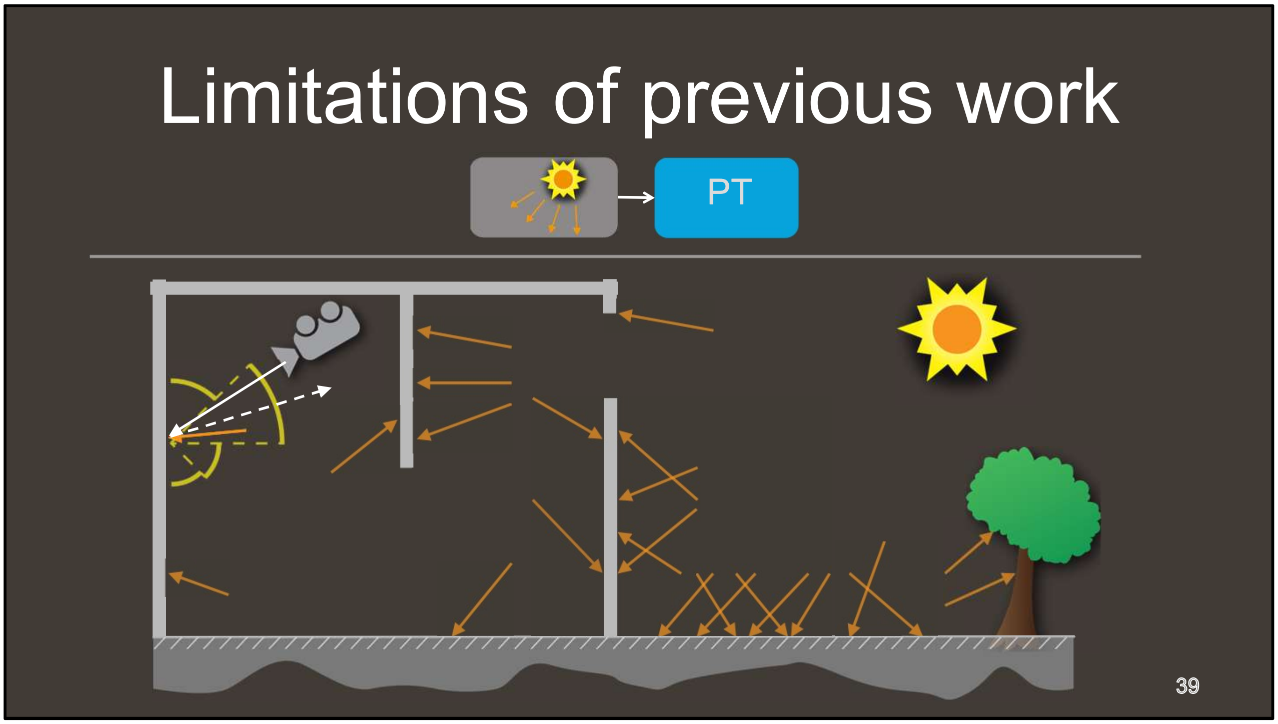

- Let’s consider an interior scene where the camera is in a dim room and light enters form the outside.



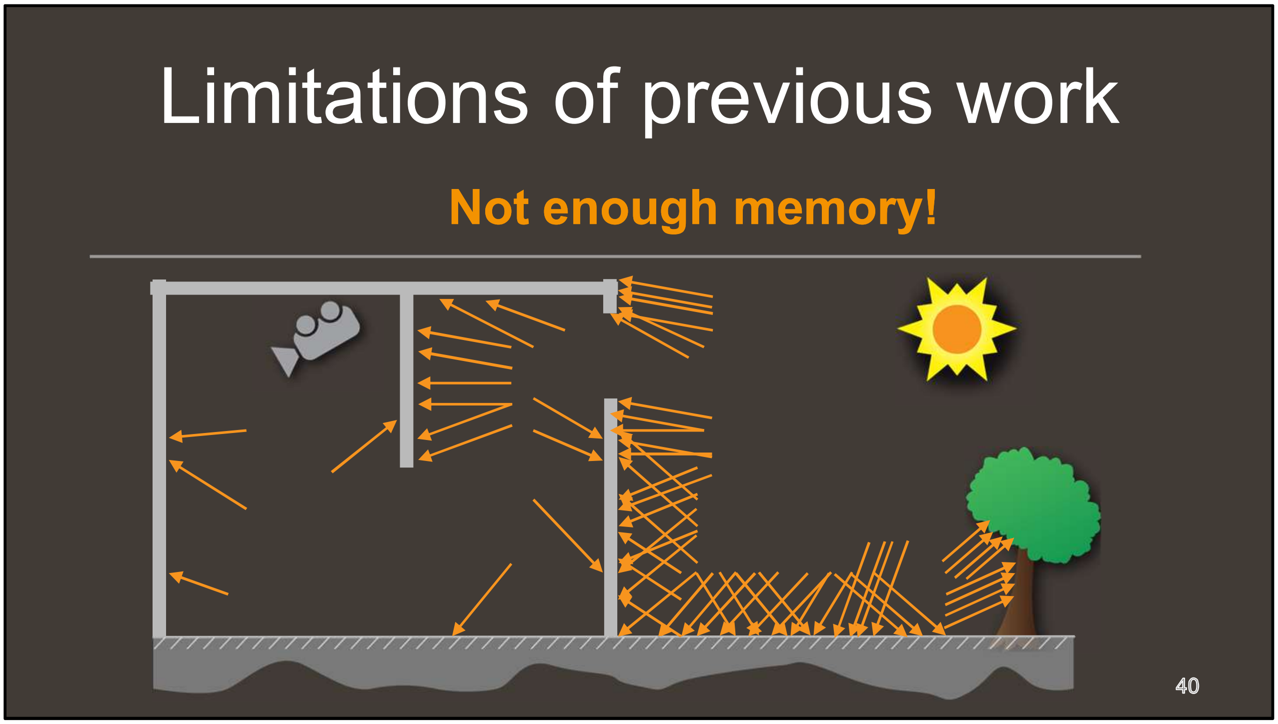

- As the particles (photons) are scattered through BRDF sampling in the photon tracing phase at the beginning, it is often not possible to obtain a sufficient number of particles everywhere for a good-enough reconstruction of incident radiance.




- For example, in this illustration, we have only one photon in front of the camera.




- Using such a poor reconstruction in path-guiding would even increase the noise level in the rendered image.
- What we need is to get many more particles into the dim room, …




- ... but the problem is that we are limited by memory where all the particles need to be stored (including the particles outside of the room, where we do not need them at all).

# Our solution

**GMM**

- The Gaussian mixture model (GMM)

41

- We follow the basic ideas of using photons as the source of information for better sampling (i.e. path guiding).
- But instead of the histogram, we propose the use of a Gaussian mixture model to represent the directional distributions.

# Our solution

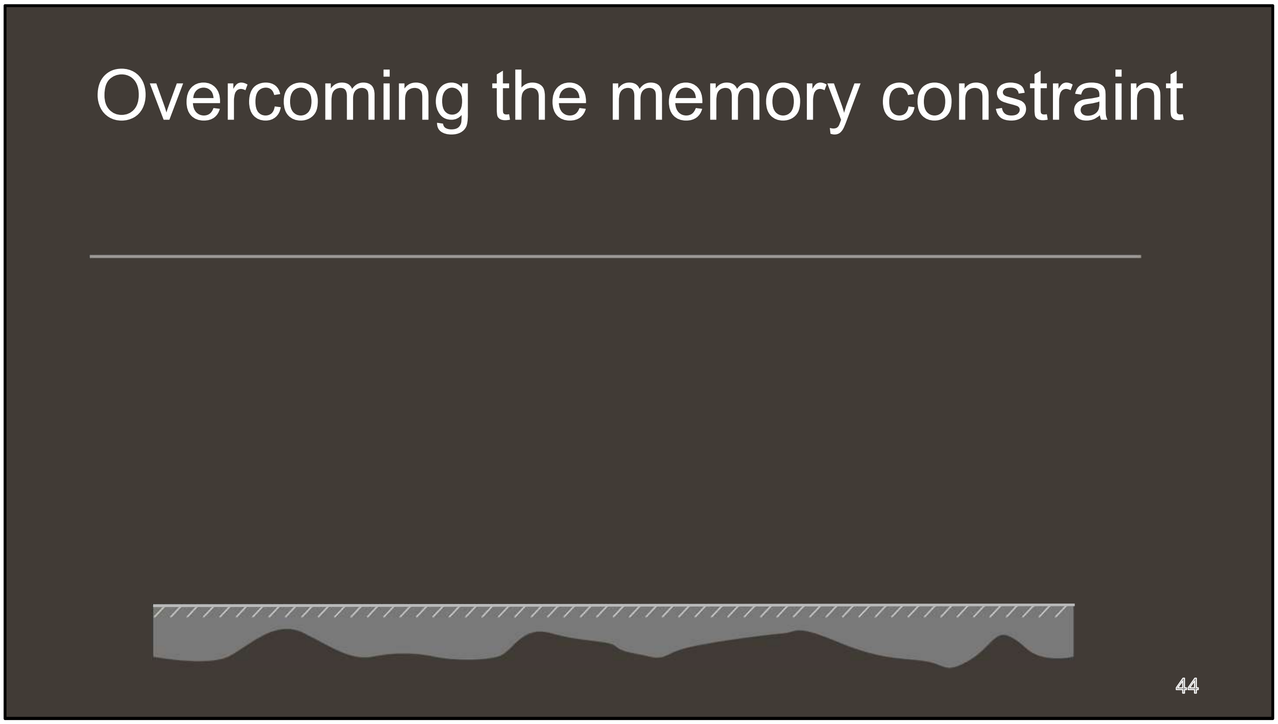
**GMM** ⇒ **on-line learning**

- The Gaussian mixture model (GMM)

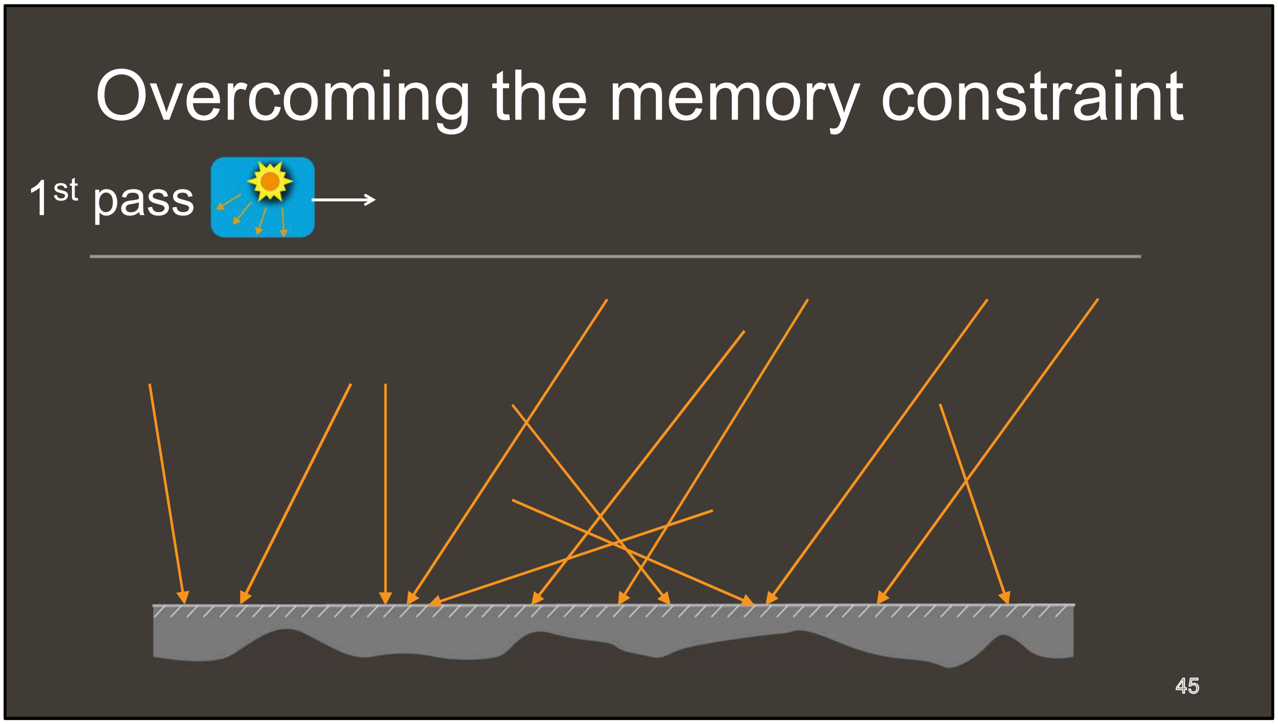
42

- Furthermore, we propose an approach for progressive (online) training of the model from a potentially infinite stream of particles.
- The important thing is that the particles do not need to be stored in memory at once (which they would have to in Jensen’s method).

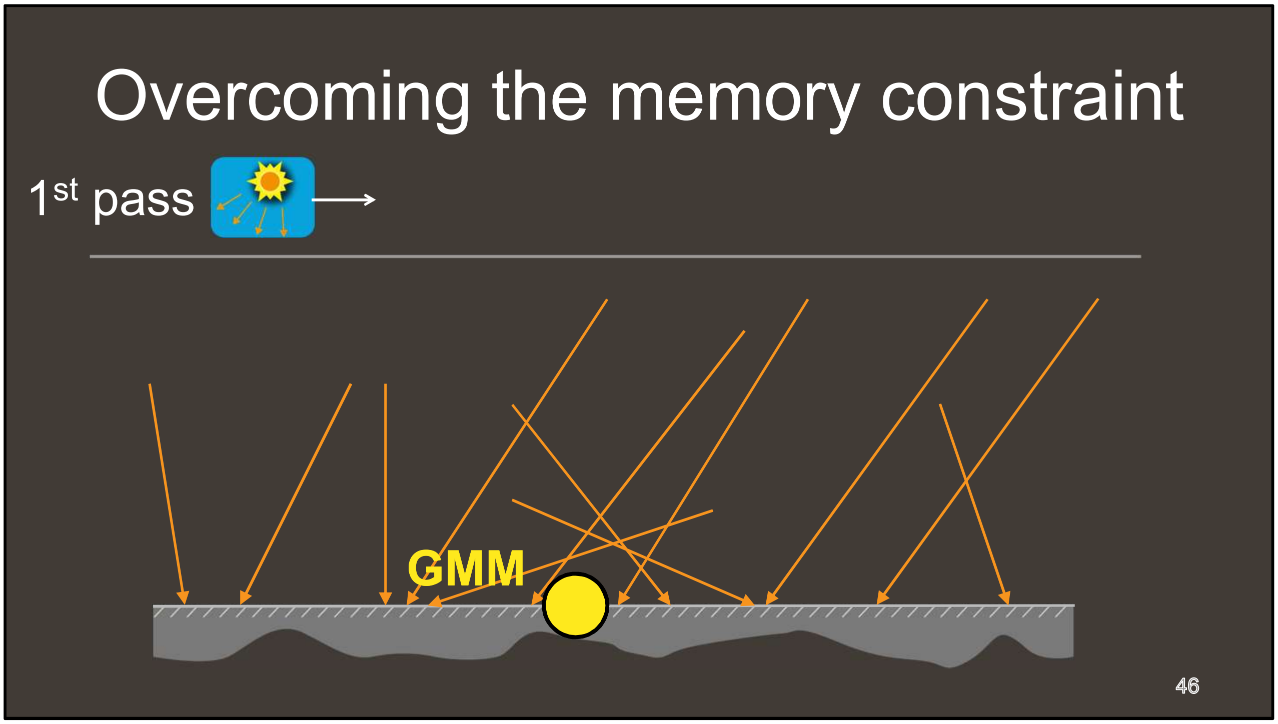

- So in this way our on-line learning allows to overcome the memory constraint.



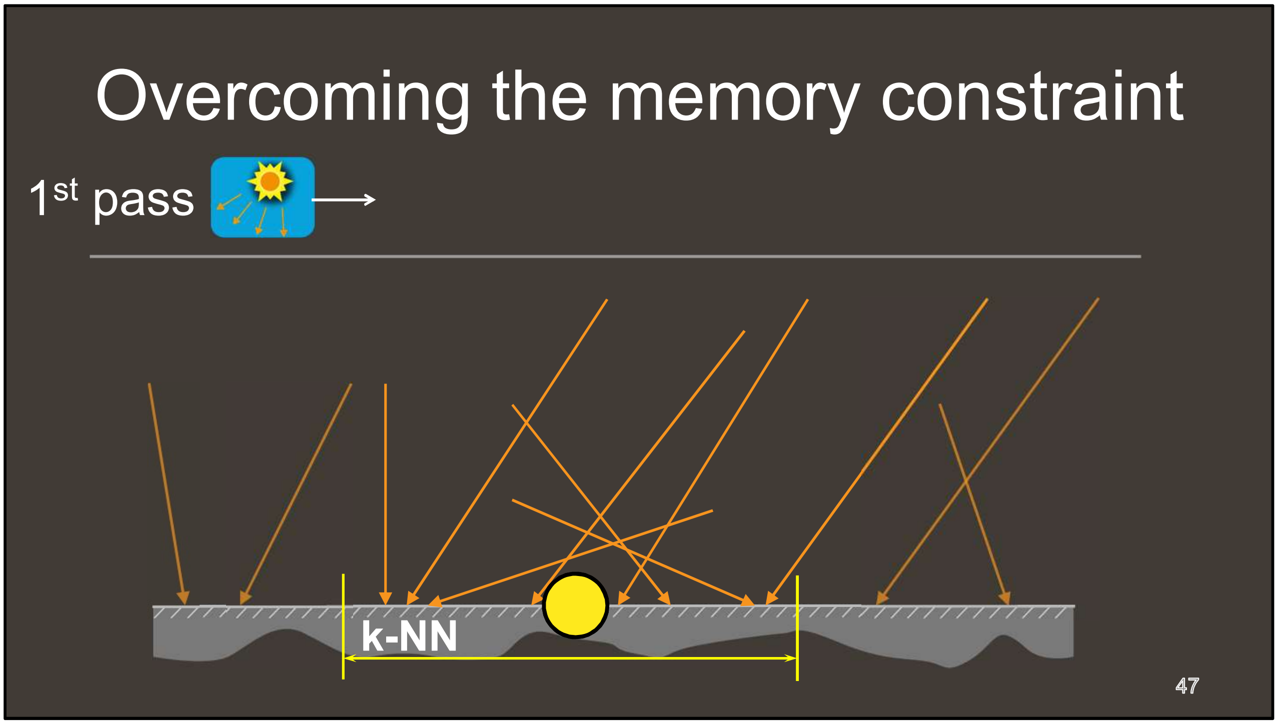

- This is how it works.



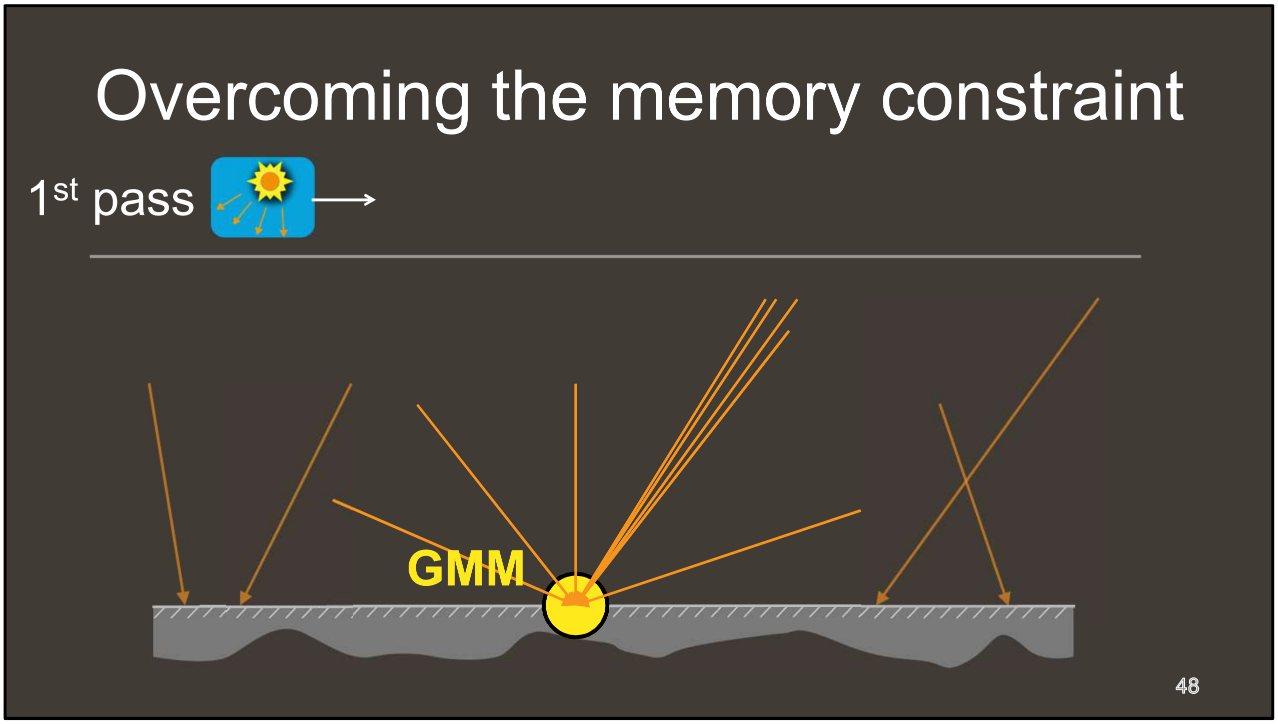

- We trace particles in batches that can fit into memory.
- So we trace a first batch of photons, …

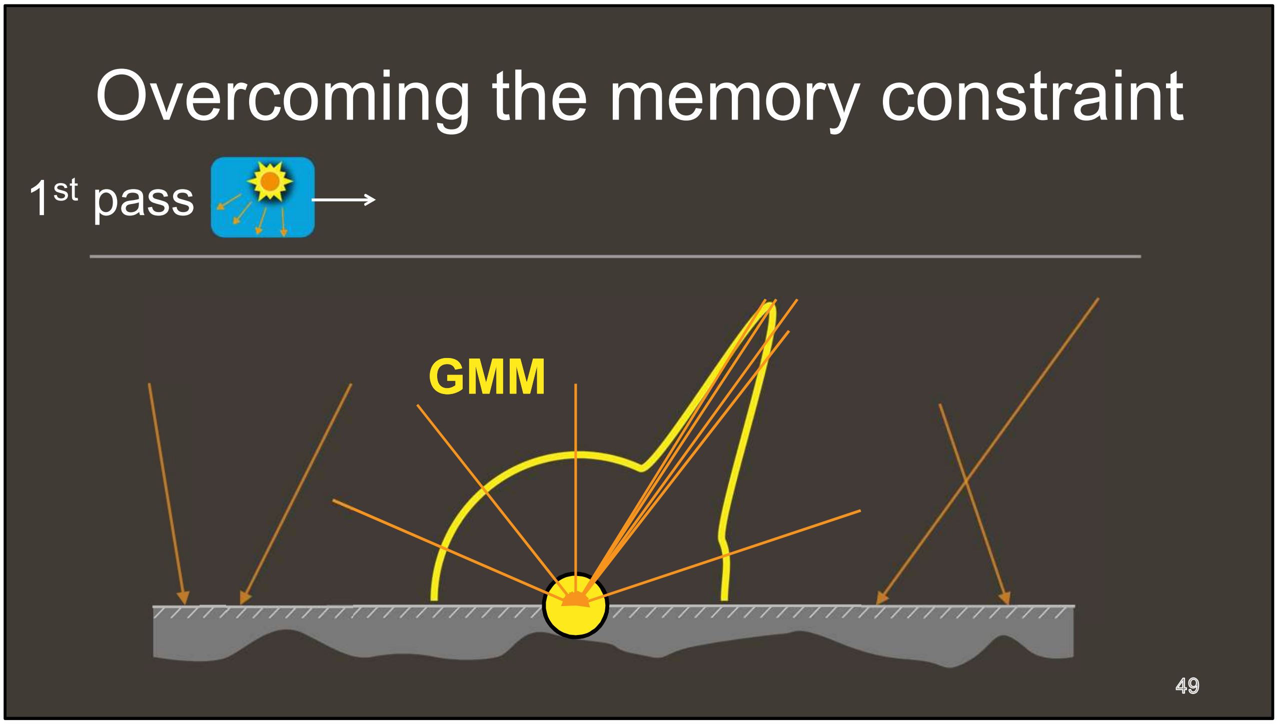



- … we create a distribution on a scene surface…

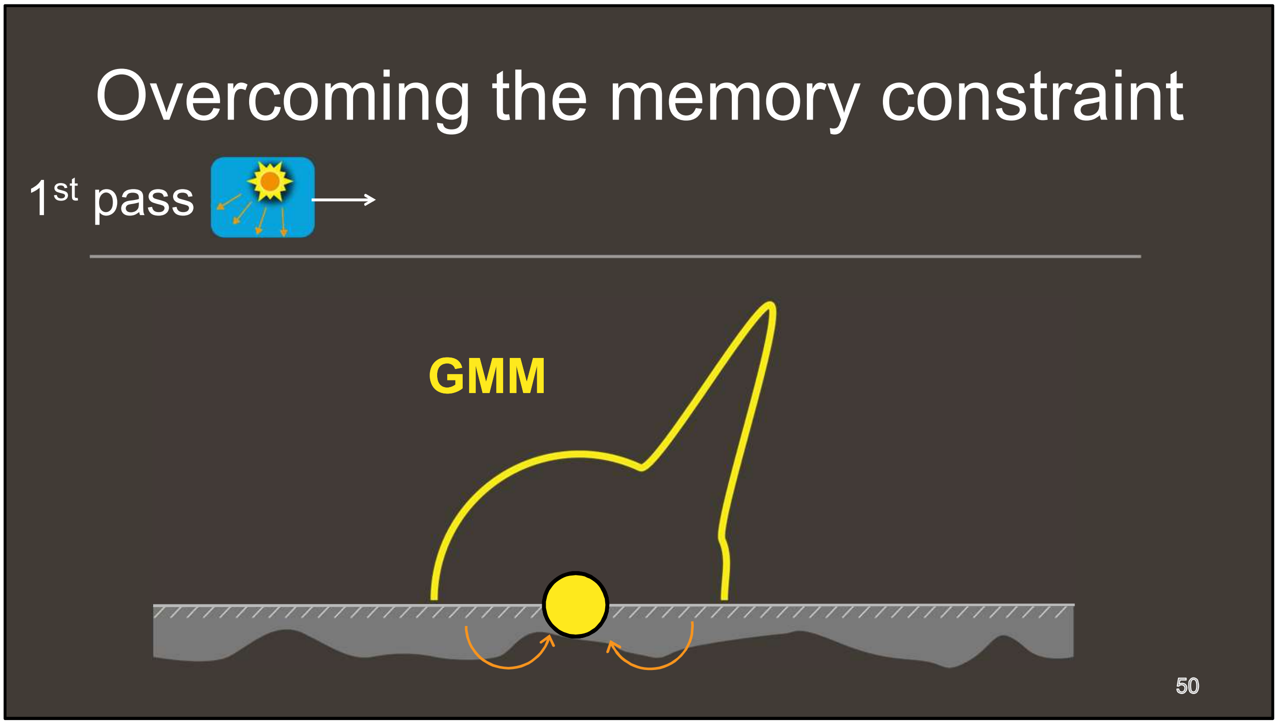



- … and we use the nearest photons for its initial training.

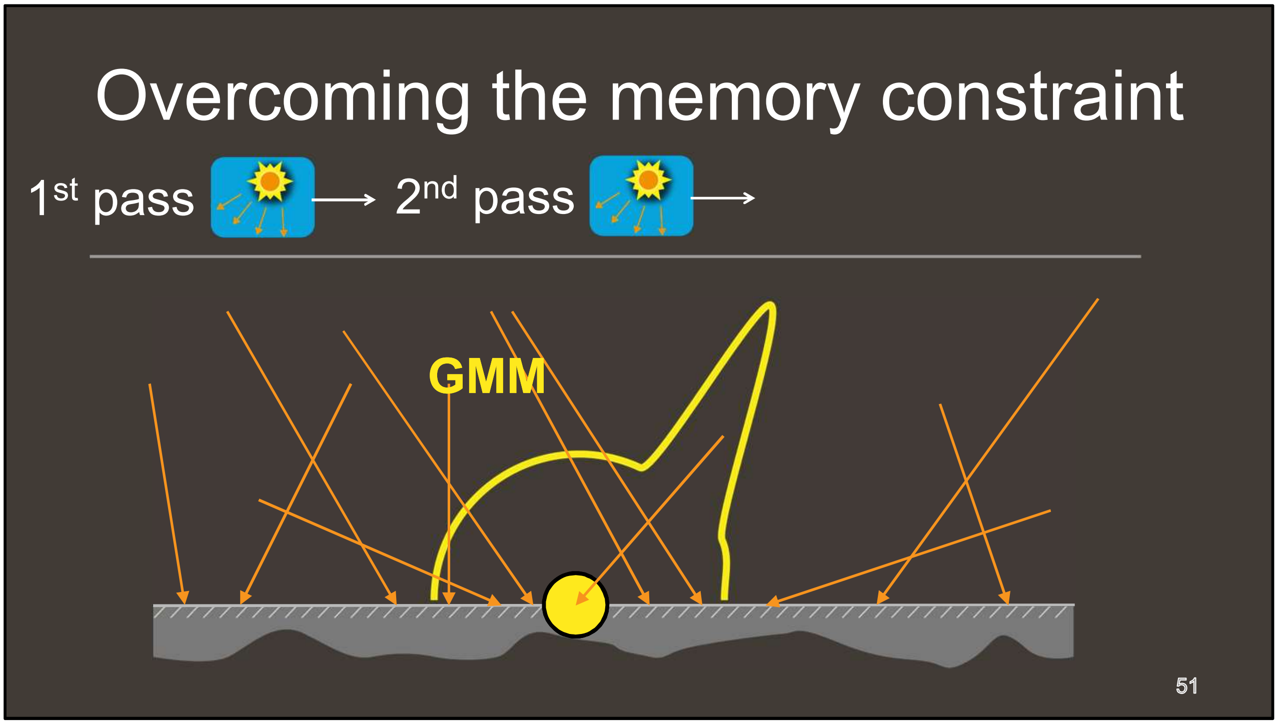

Overcoming the memory constraint
1st pass
GMM
48



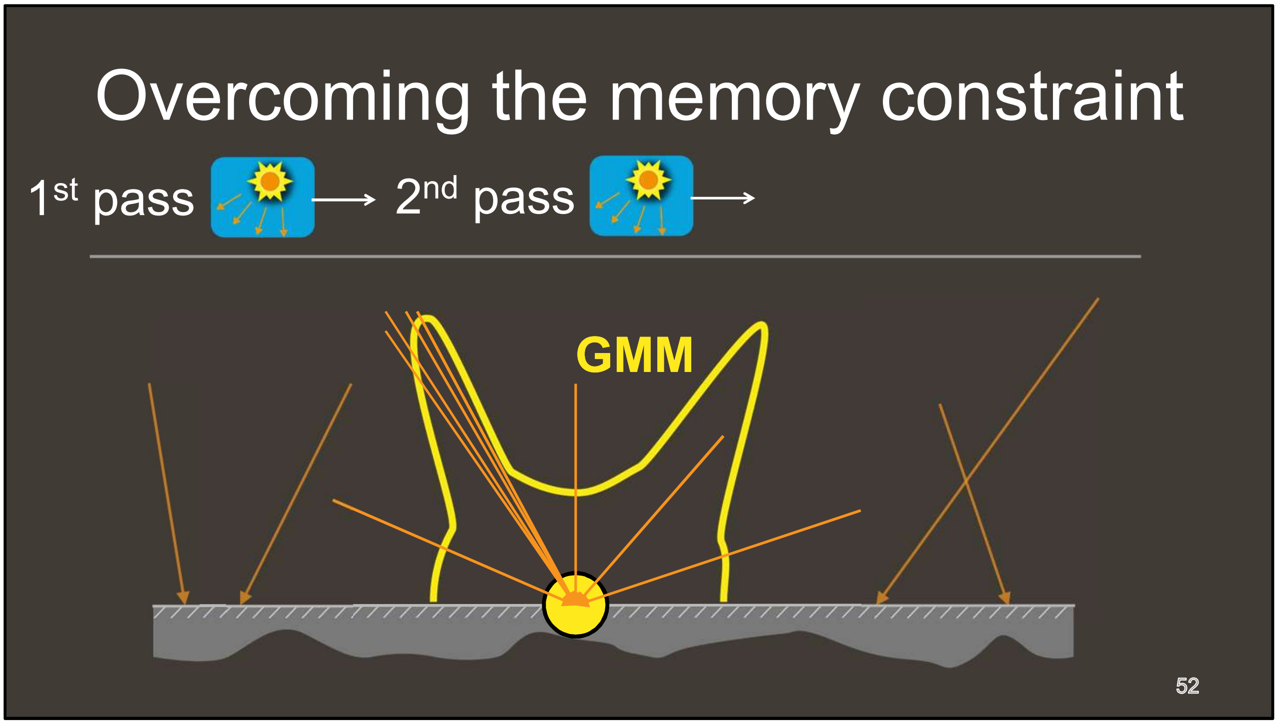

- Now the particles can be removed …



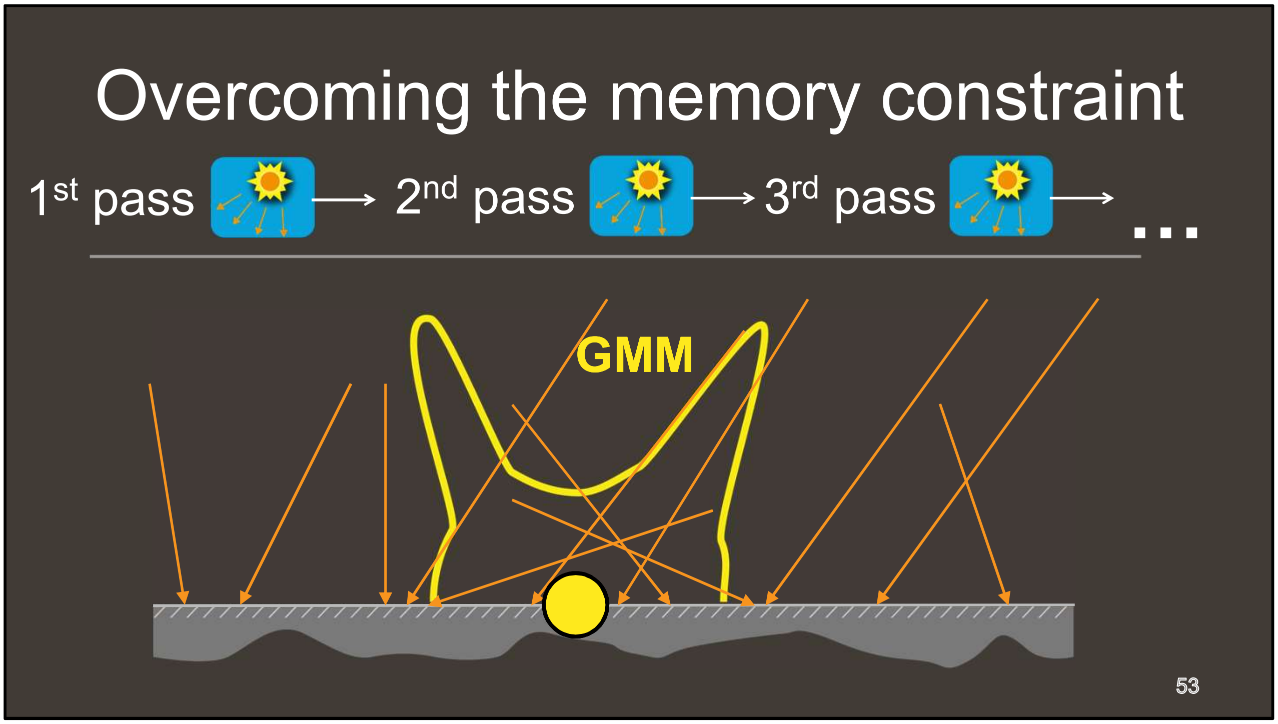

- … because the information has been absorbed in the distribution.



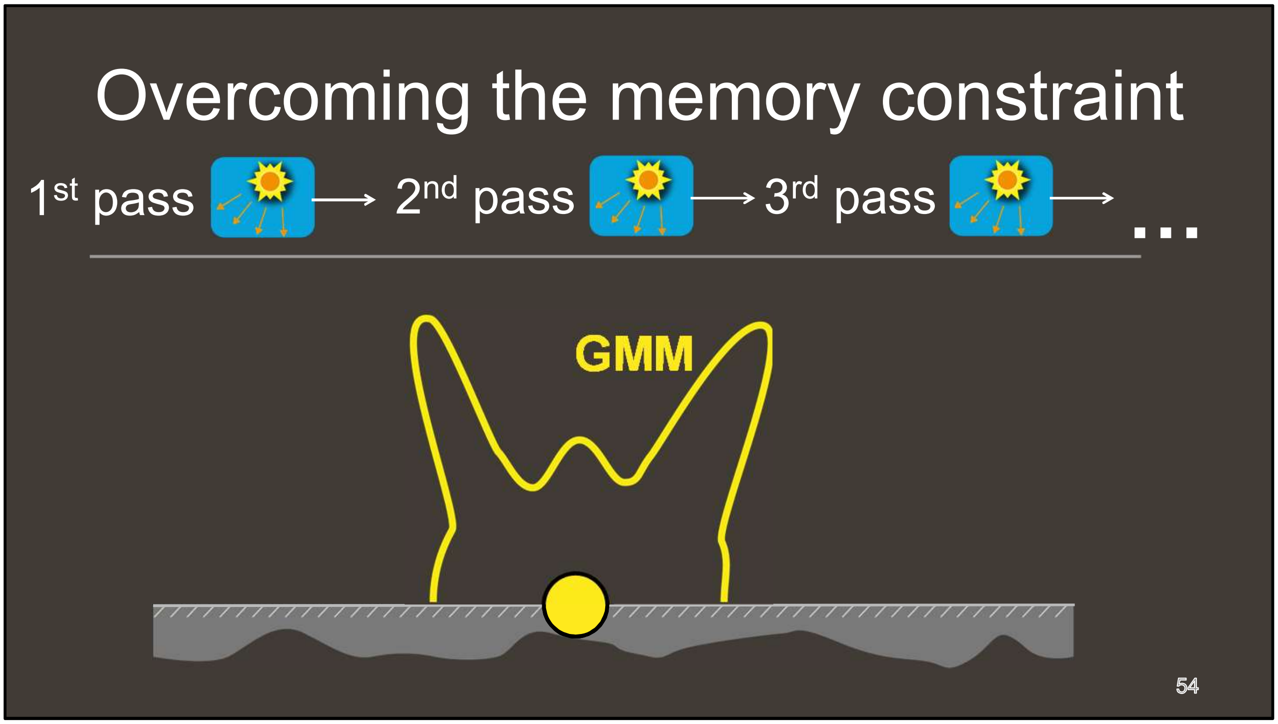

- Then we trace another batch of particles,



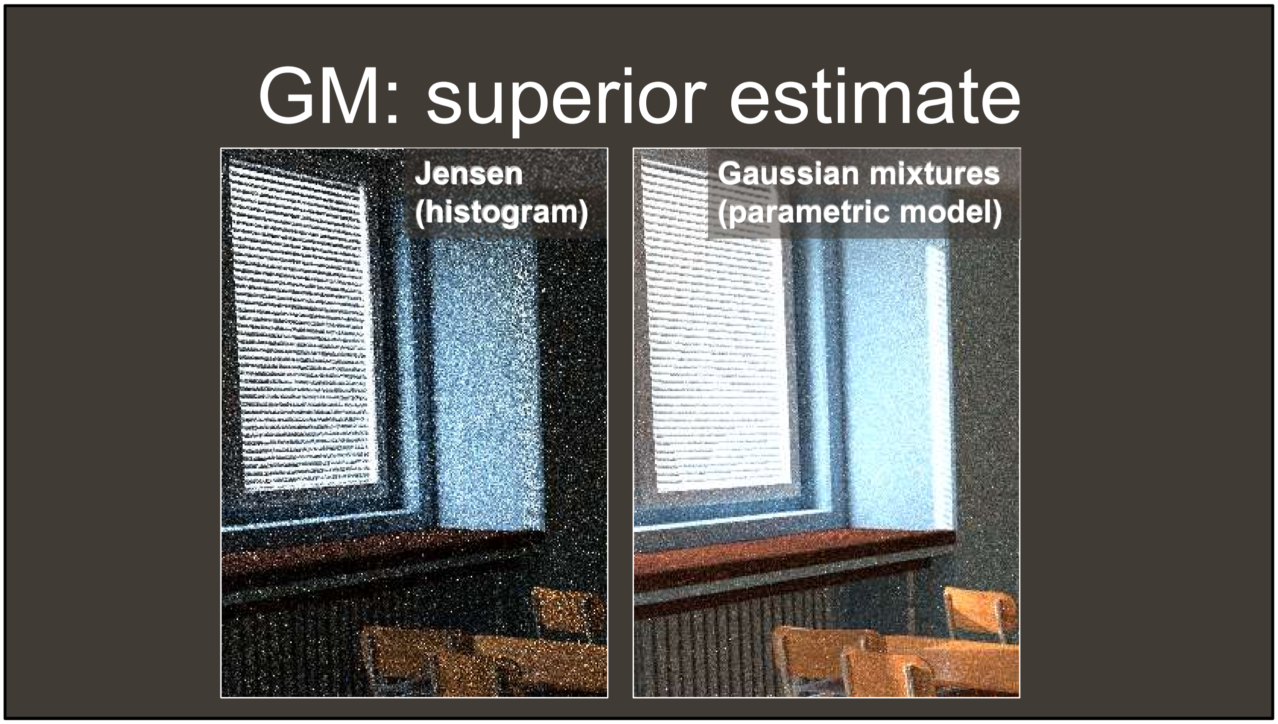

- … and we use the nearest ones to progressively update the distribution.



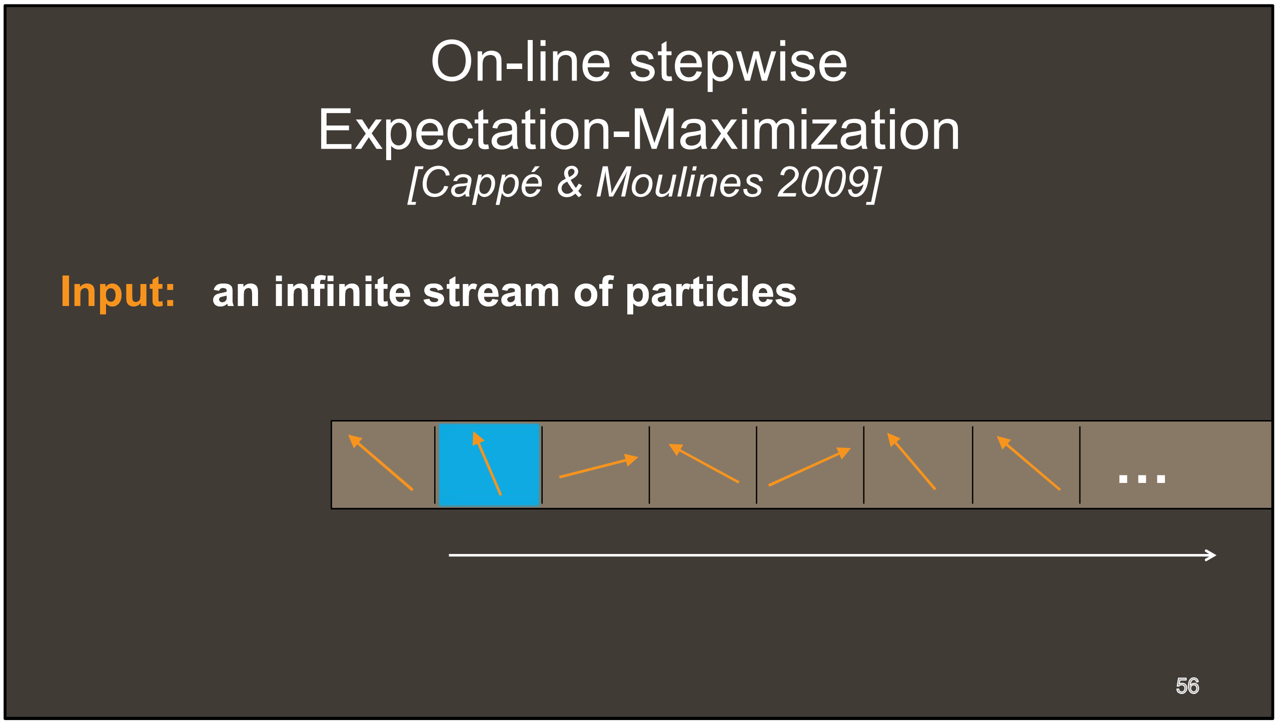

- We can repeat this process until the distribution is fully trained.



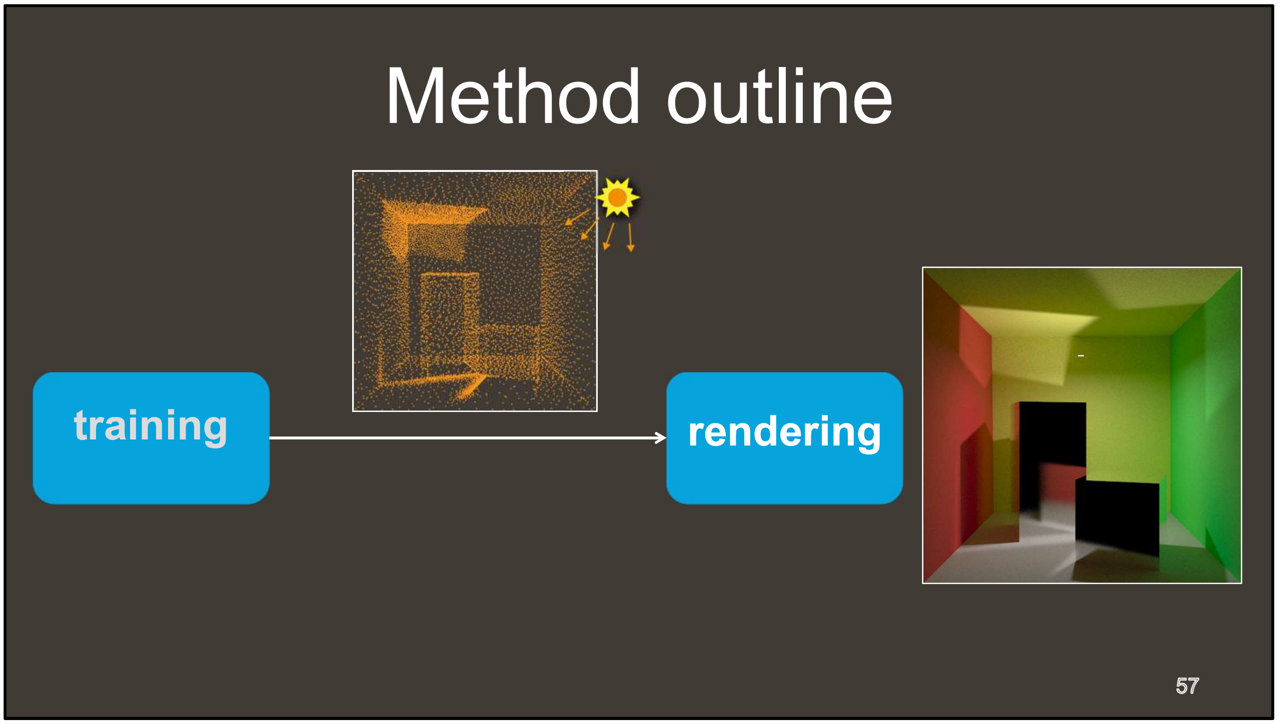
Overcoming the memory constraint
1st pass
2nd pass
3rd pass
...
GMM
54



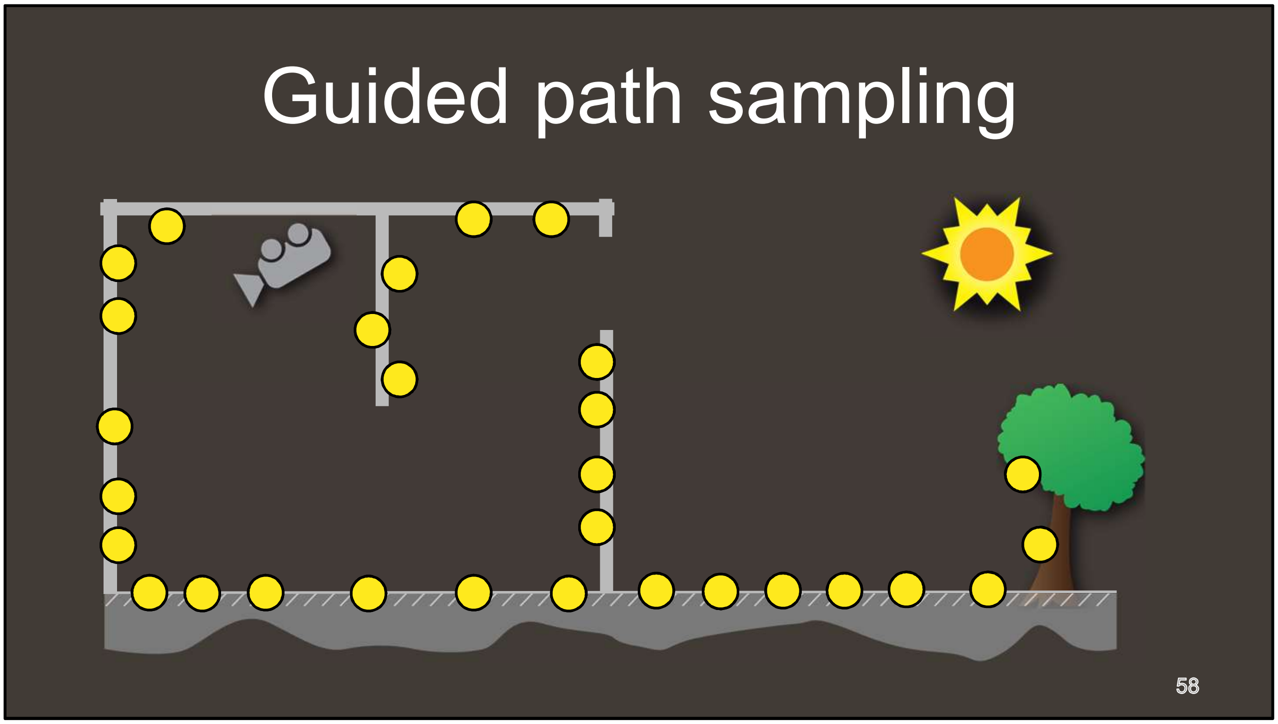

- Is it a good idea to use the (admittedly more complex) Gaussian mixture model to represent the incident radiance, as opposed to Jensen’s histograms?
- The above comparison shows that this is indeed the case – even without taking advantage of the online learning aspect.
- In this experiment, we trained Jensen’s histograms and our GMMs using the same number of photons.
- The amount of noise after 1h of rendering suggests that the Gaussian mixture model is clearly superior to the histograms.




- We saw that the GMMs are a good model guiding distributions, but they can only be useful if we can keep training them form the stream of photons as they come during rendering.
- We want to read particles once from a potentially infinite stream and never return to previously used particles.
- And here, the **machine learning** literature is of a great help, since **online learning methods** are available that can achieve exactly that.
- In our case, we rely on the **online formulation of the famous EM algorithm**.
- Note that online training would not be possible with the usual (batch) EM algorithm previously used in graphics – because the batch EM needs to repeatedly iterate over all the available particles.

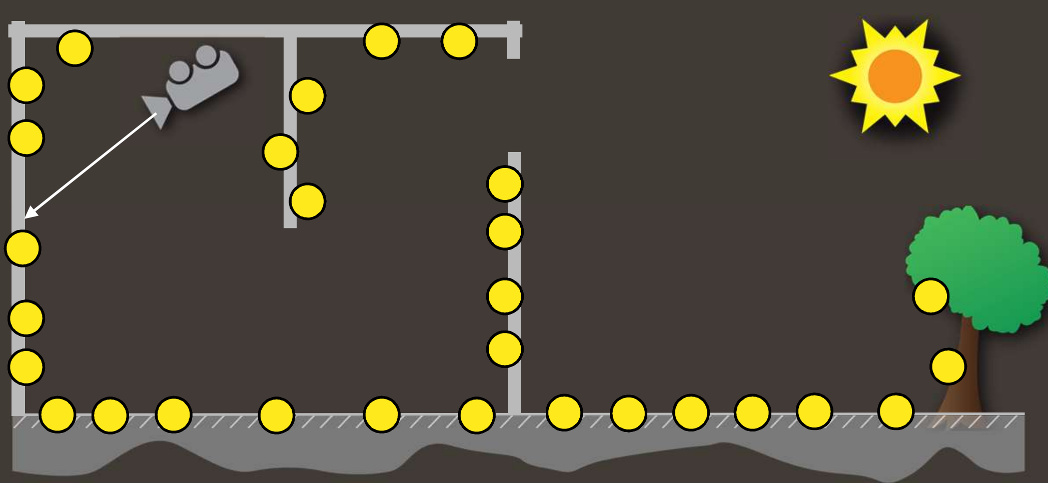



- Let us now summarize the entire rendering method.
- Before rendering, we train our Gaussian mixtures in a training phase.
- The result of this phase is a spatial cache of path guiding distributions. The distributions are trained from photons and thus contain directional distributions of radiance.
- These distributions are used to guide path-sampling during the later rendering phase.



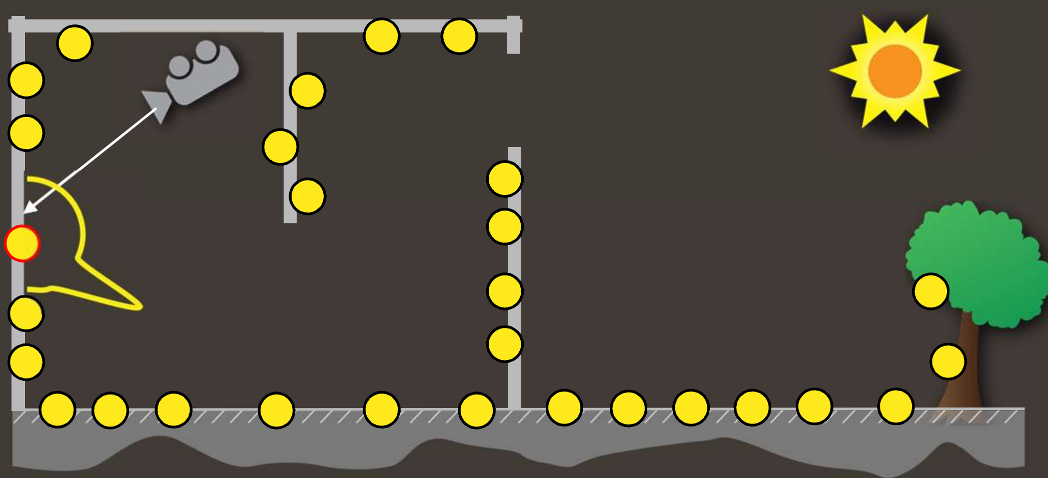

- Let us now explain the guided path-sampling on an illustration.
- Here, we trace a path from the camera and we use the nearest radiance distributions for its guiding towards the light sources.

# Guided path sampling

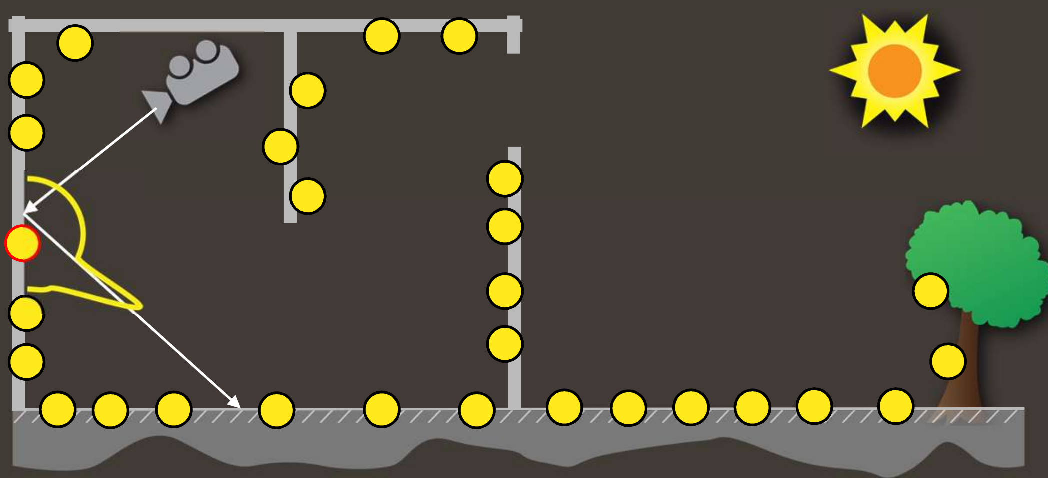


59

# Guided path sampling

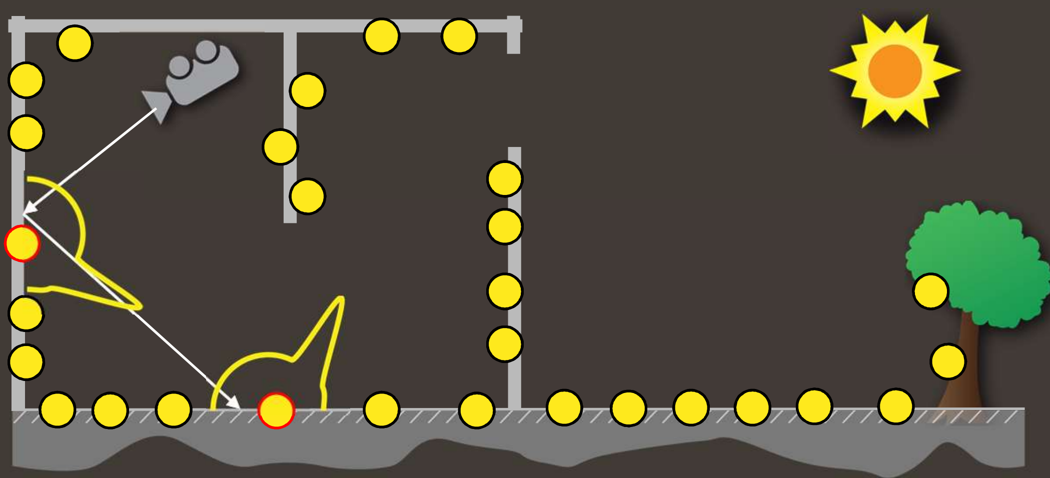


# Guided path sampling



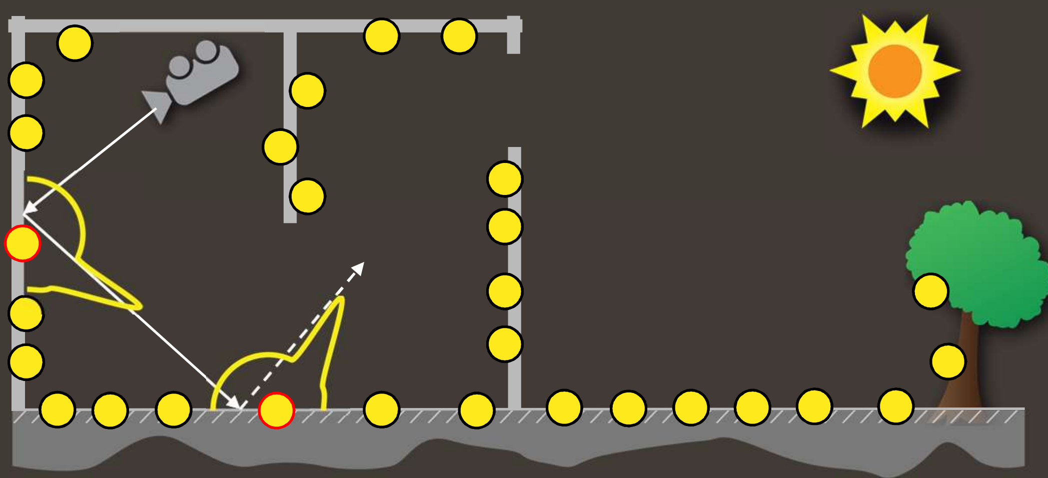
61

# Guided path sampling

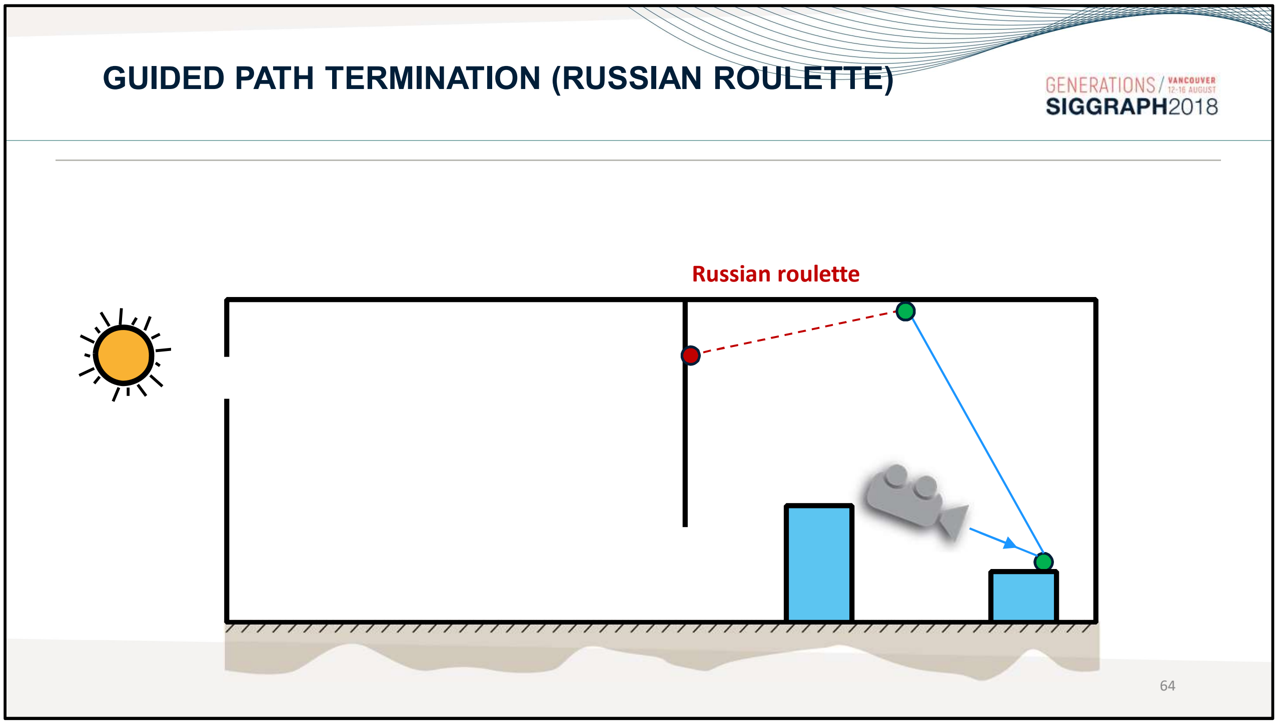


62

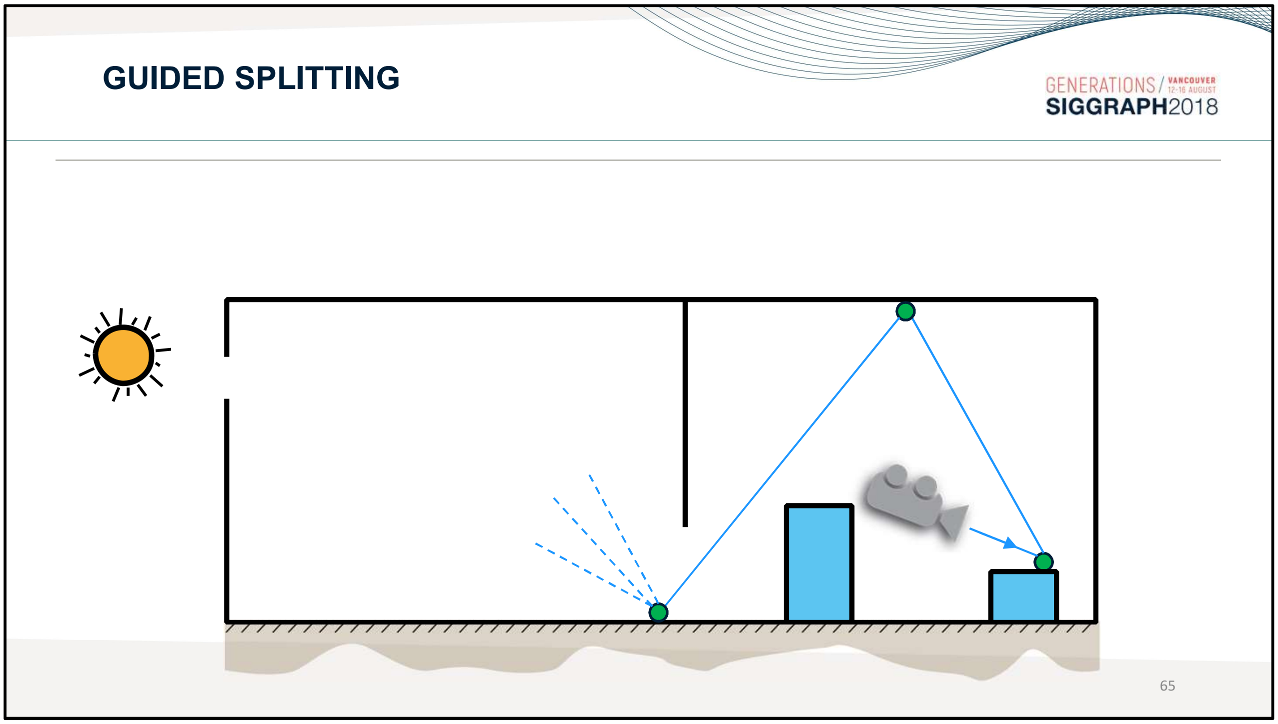
# Guided path sampling



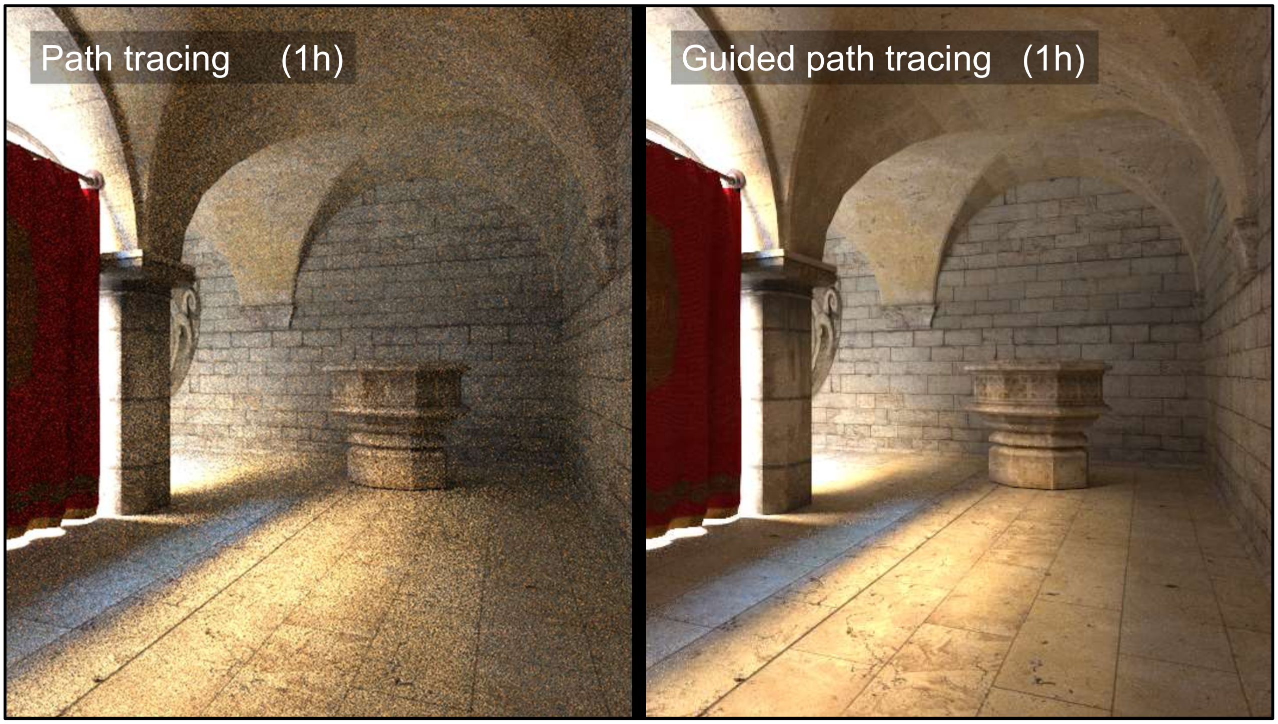
63



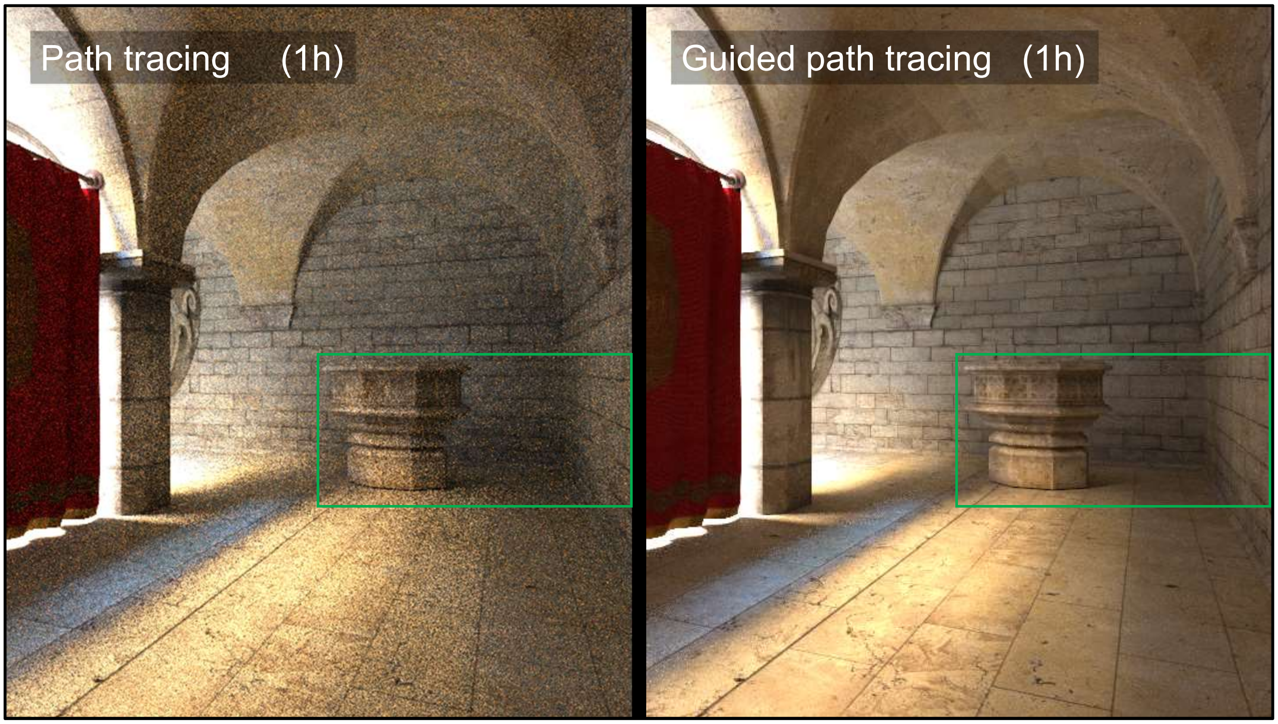

- The information stored in the trained Gaussian mixtures can be additionally used to inform the path tracer about the expected amount of illumination in different regions of the scene.
- And thanks to this information, the path tracer can now make a much more intelligent decisions about path termination: whenever there is not much light is some scene region, a path reaching that region is unlikely to contribute significantly to the resulting image and we should not waste time on its tracing -> terminate.

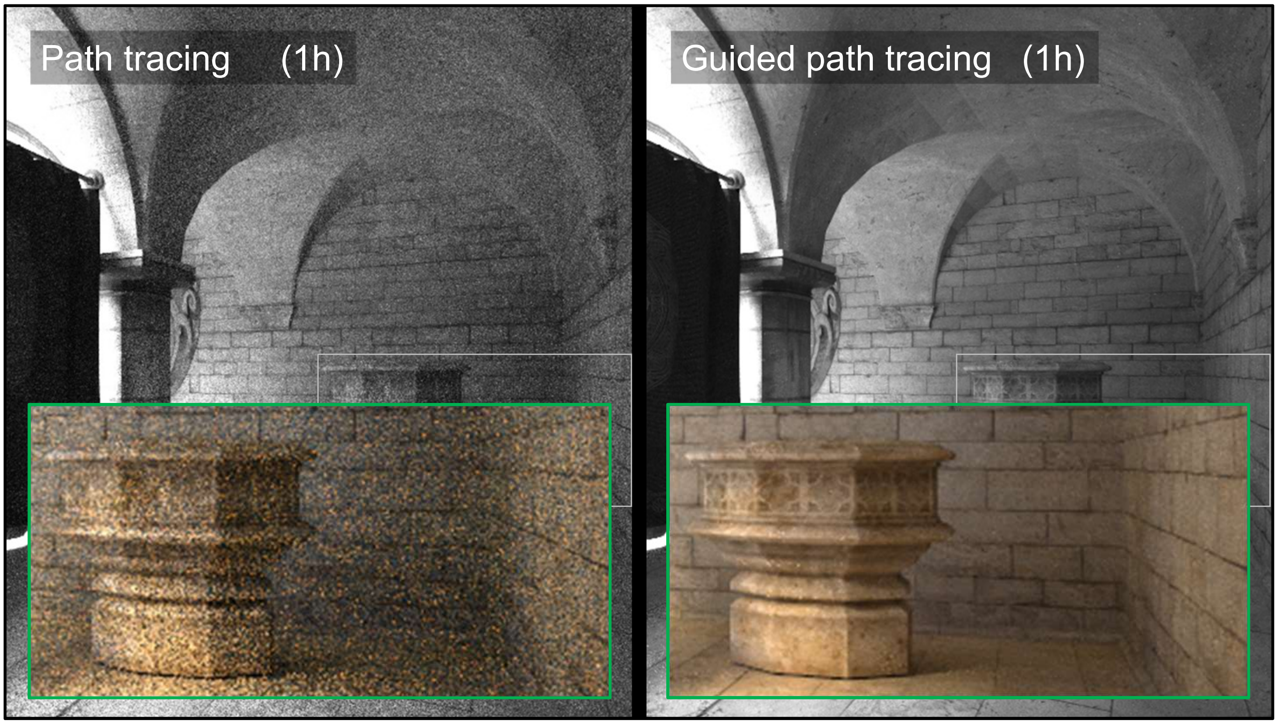



- And vice versa, if the path construction reaches a well-illuminated region of the scene, it is more likely that the path will be able to reach the light source and it pays off to split it into several independent trajectories.
- And once again, the gaussian mixtures provide a good source of information for making such an intelligent sampling decision.



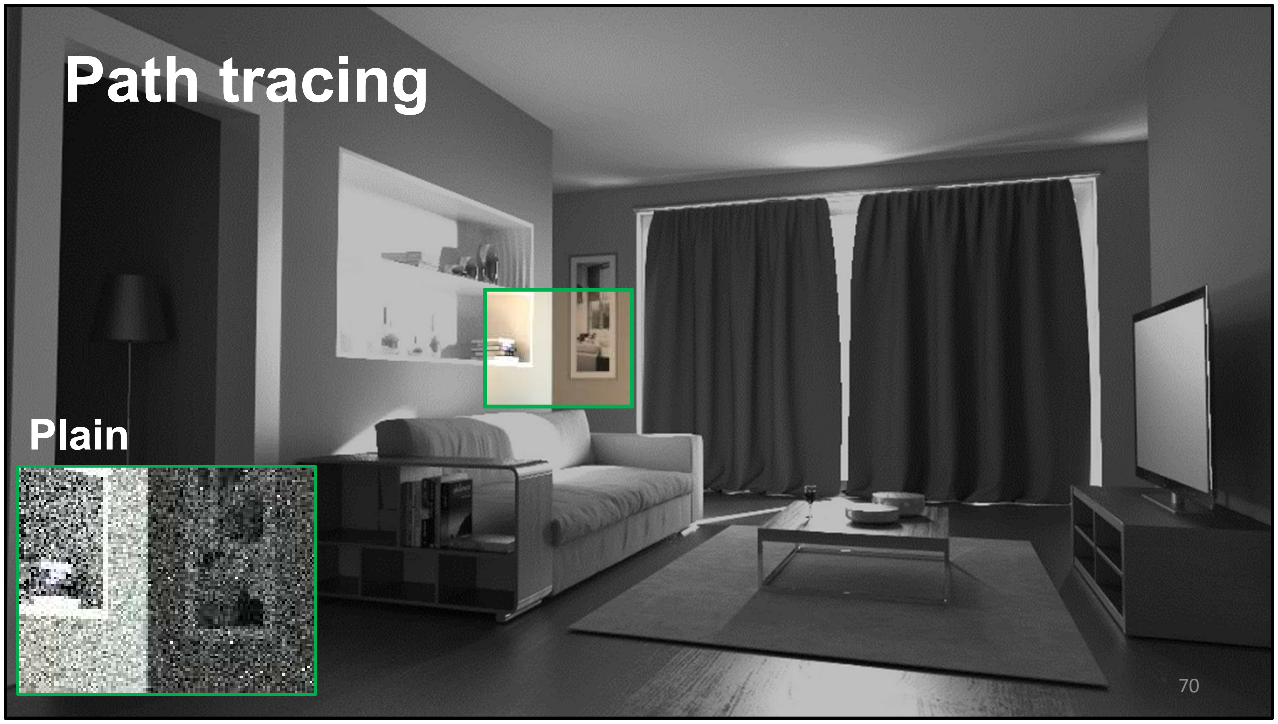

- Let’s have a look at some path guiding results.
- Both images above have been rendered using a path tracer in 1 hour.
- The left image is a classic, uninformed path tracer.
- While the path tracer on the right uses path guiding with guided Russian roulette and splitting. This is based on guiding distributions that the method automatically *learns* for each given scene.


Path tracing (1h)
Guided path tracing (1h)

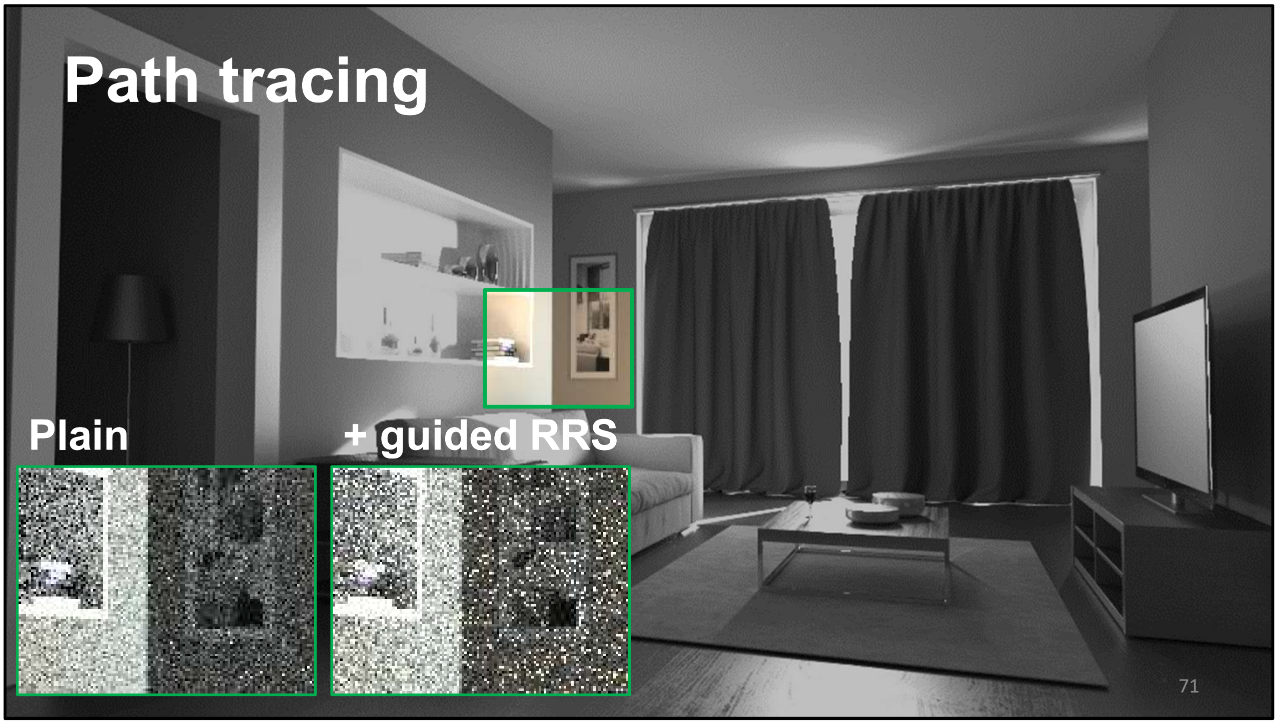



- This is a detail of a part of the image that is especially problematic for the plain path tracing.
- The problem is that the well is illuminated indirectly and light must bounce many times before it reaches the camera.
- Plain path tracing is not able to sample these important paths with sufficient probability because it does not use any global information about light distribution in the scene.

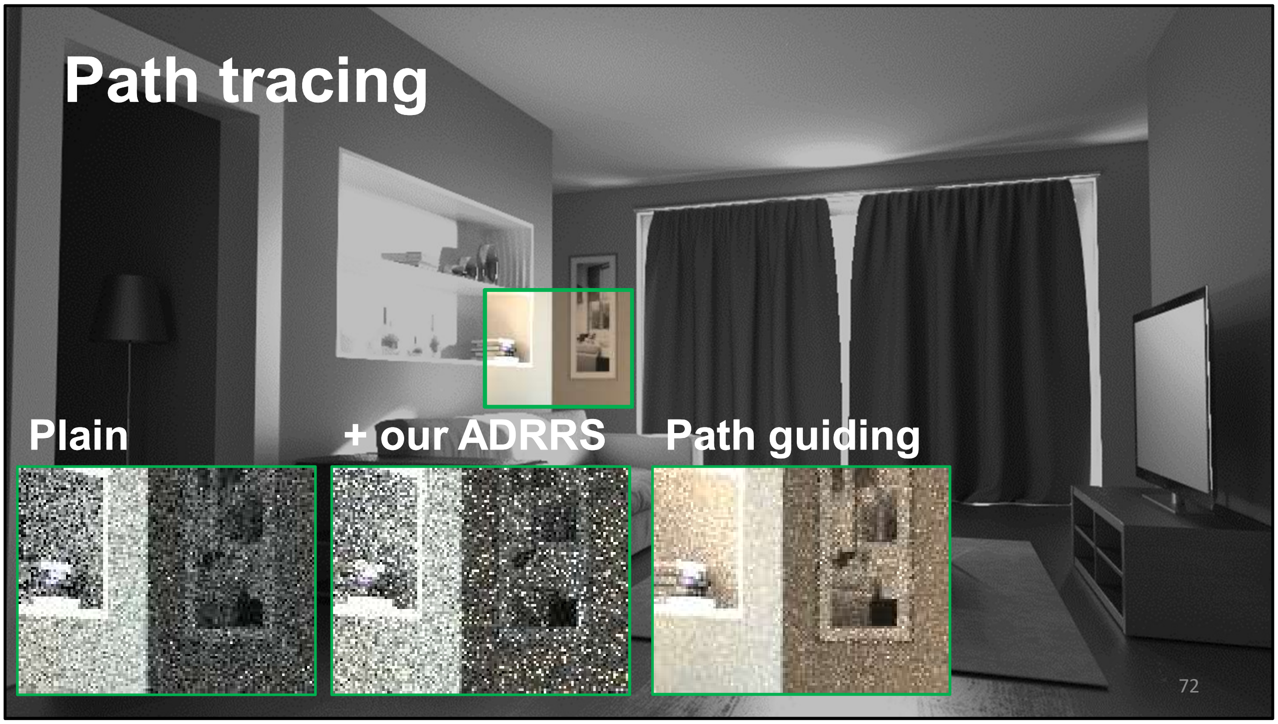

- In this scene, the living room is illuminated by sun and sky.
- Light comes only through the glass window and a small gap between the curtains.



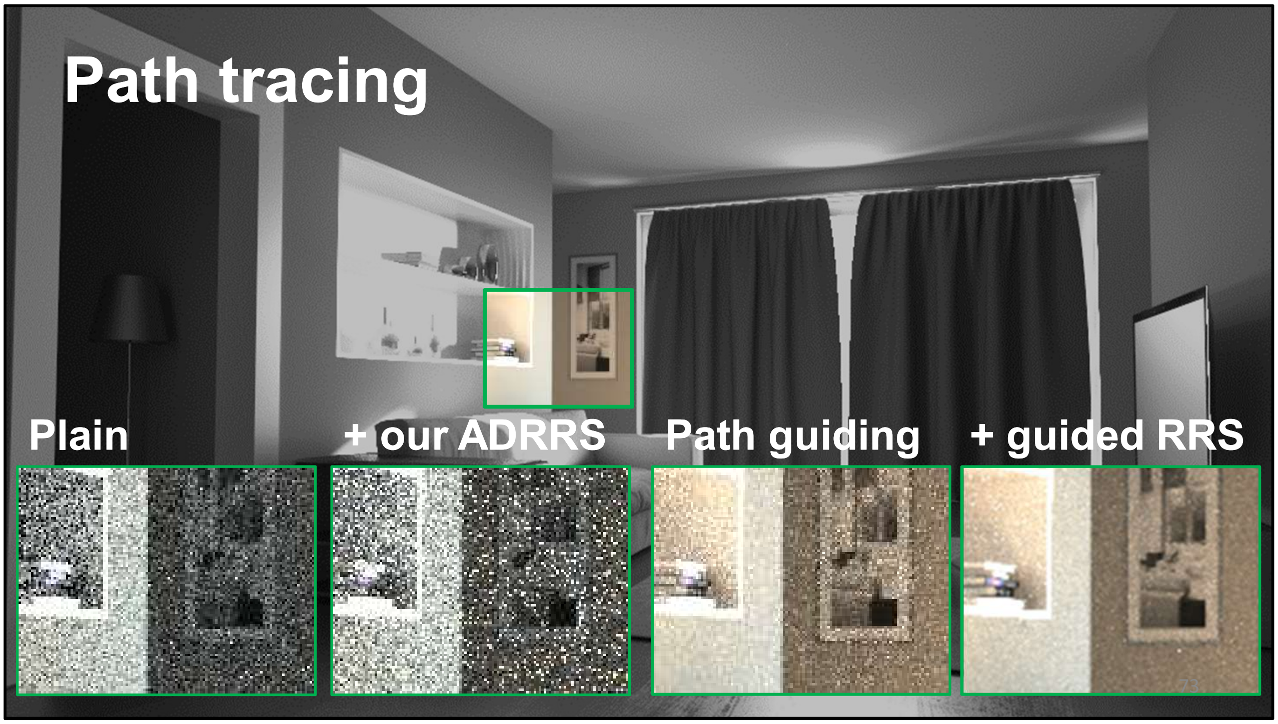

- These difficult conditions are a showstopper for plain path tracing.




- The guided Russian roulette and splitting by itself doesn’t make any striking difference.
- The main problem is that the light paths can reach the light source only through very narrow set of directions.
- Our guided RR&S can detect that we highly undersample these directions at the moment when path hits the window.
- But this is already too late for splitting because all rays would be refracted in the same direction and end in the Sun.




• While the guided RR&S does not work very well, path guiding (i.e. guided directional sampling) achieves very good results.



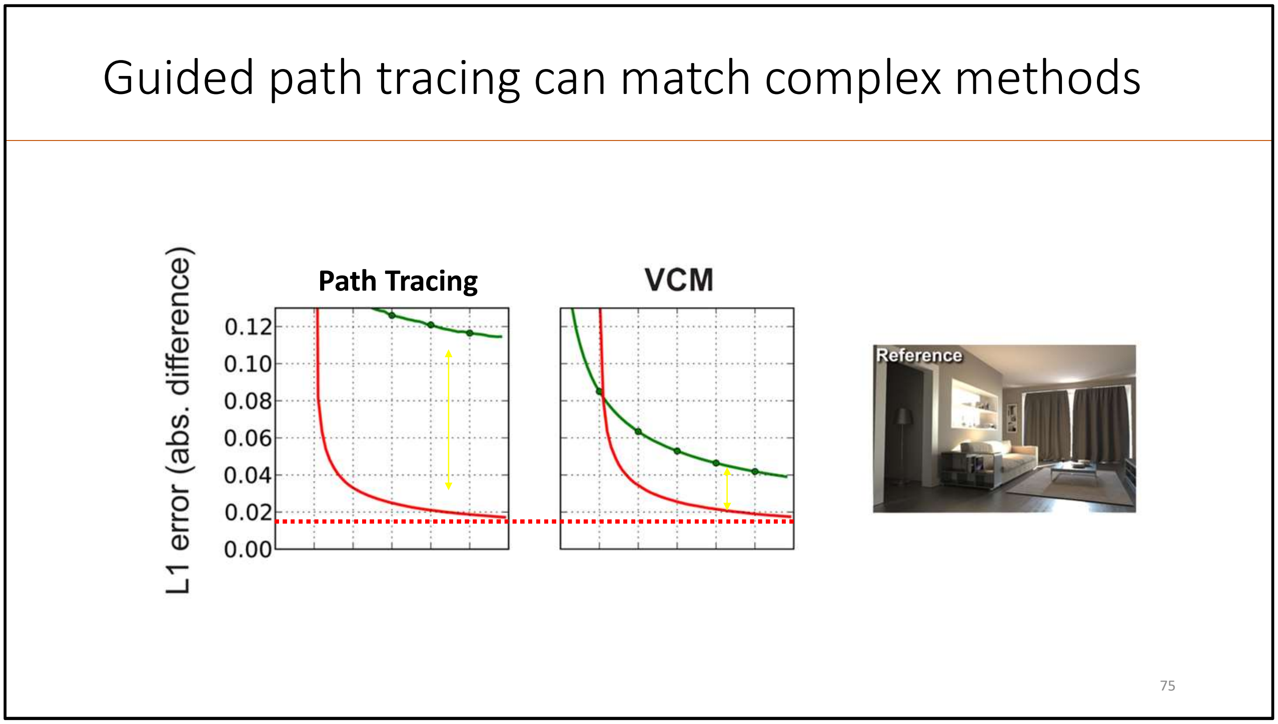

- However, we can see that with the synergic use of path guiding and guided RR&S together, we can achieve far better results than the plain uniformed path tracer.

## Complex Bidirectional Methods (VCM)



74

- Let's now go back to our motivation. We saw that the VCM algorithm was able to achieve robustness in complex scenes, but at the cost of significant overhead.
- Let us now have a look how a guided version of simple unidirectional path tracing compares to a guided version of the advanced VCM algorithm.




- We applied our path guiding in unidirectional path tracing and in bidirectional VCM.
- Here we can see their respective convergence plots in a scene with difficult visibility.
- The green curves correspond to each algorithm without path guiding, while the red curves correspond to improved path sampling with our path guiding.
- Although our guiding improved both methods in this challenging scene, there is almost no advantage of the complex VCM over path tracing when we use our path guiding.
- This suggests that in unidirectional guided path tracing we do not need to introduce any extra interconnections with light paths because it would not make things any better.
- This is an important implication – it confirms that guided path tracing is at least as good as the much more complex bidirectional methods.

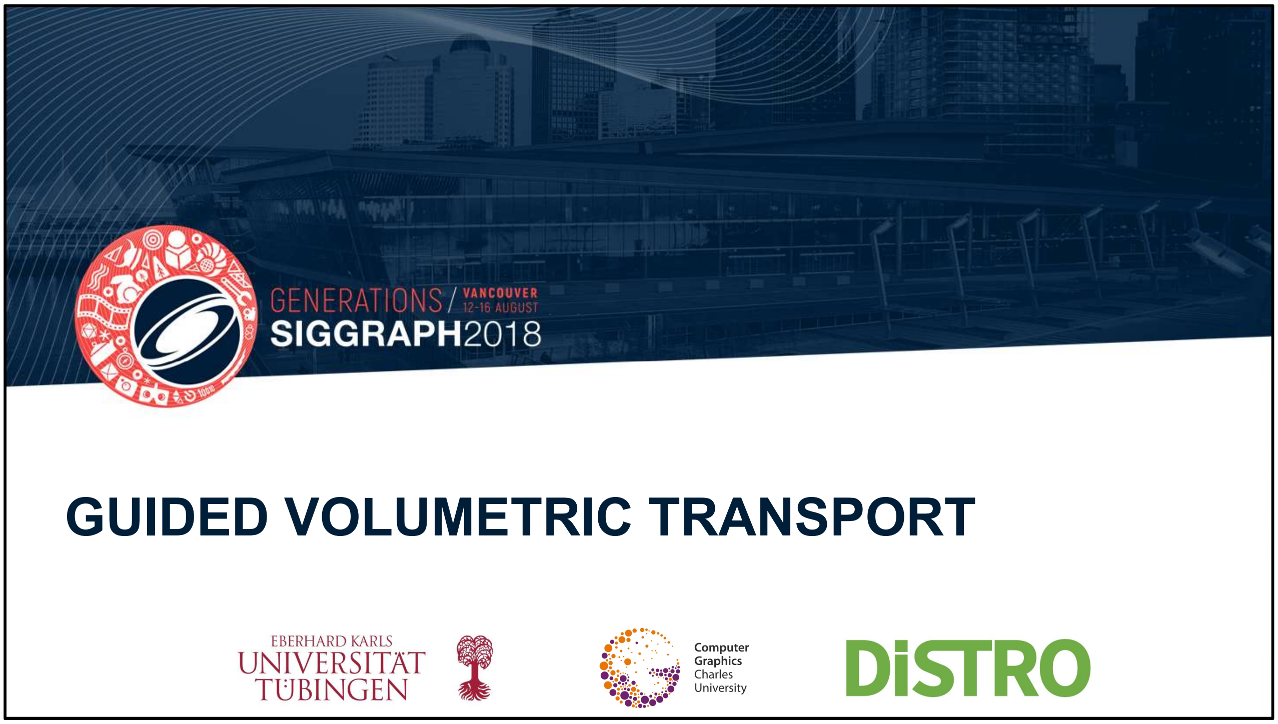
## Practical Implication

- Providing path tracer with information makes it much more robust
  - Machine learning is the key (online step-wise EM formulation)
- Step towards a simpler ultimate algorithm
- Path guiding applicable in production

76

- To bring the point home: Providing the path tracer with reliable information that it can use to make ‘smarter’, or ‘guided’ sampling decisions makes the algorithm tremendously more robust.
- A key aspect here is **the use of machine learning methods** – parametric density estimation using mixture models and online Expectation-Maximization – to be able to extract reliable information from the noisy Monte Carlo samples.

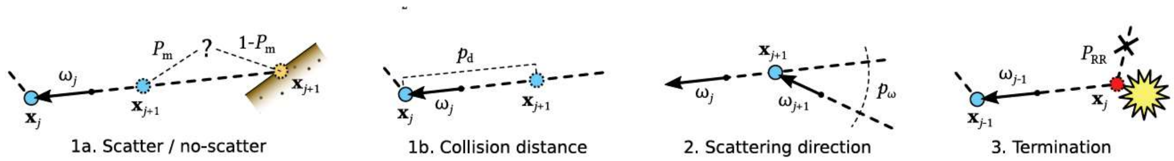
- This is indeed a step towards a simpler ultimate algorithm that can simulate all kinds of light transport.
- The simplicity is important for production environments that btw. also requires simulation of various non-physical phenomena.
- Such a non-physical simulation is usually not easy with bidirectional methods.
- So our method that we develop throughout the thesis is immediately applicable to production renderers.




- Recently, we have applied the path guiding idea also to volumetric light transport.

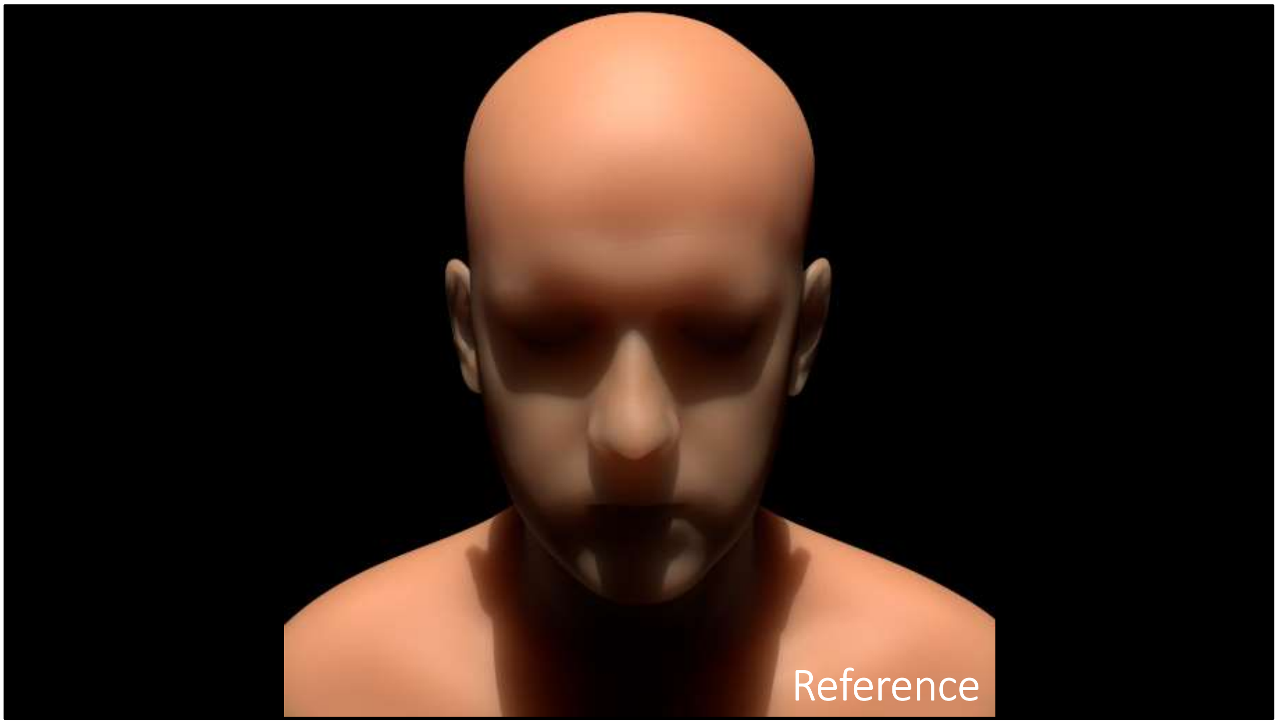
## Volume path guiding

- All events importance sampled
- Product sampling for collision distance

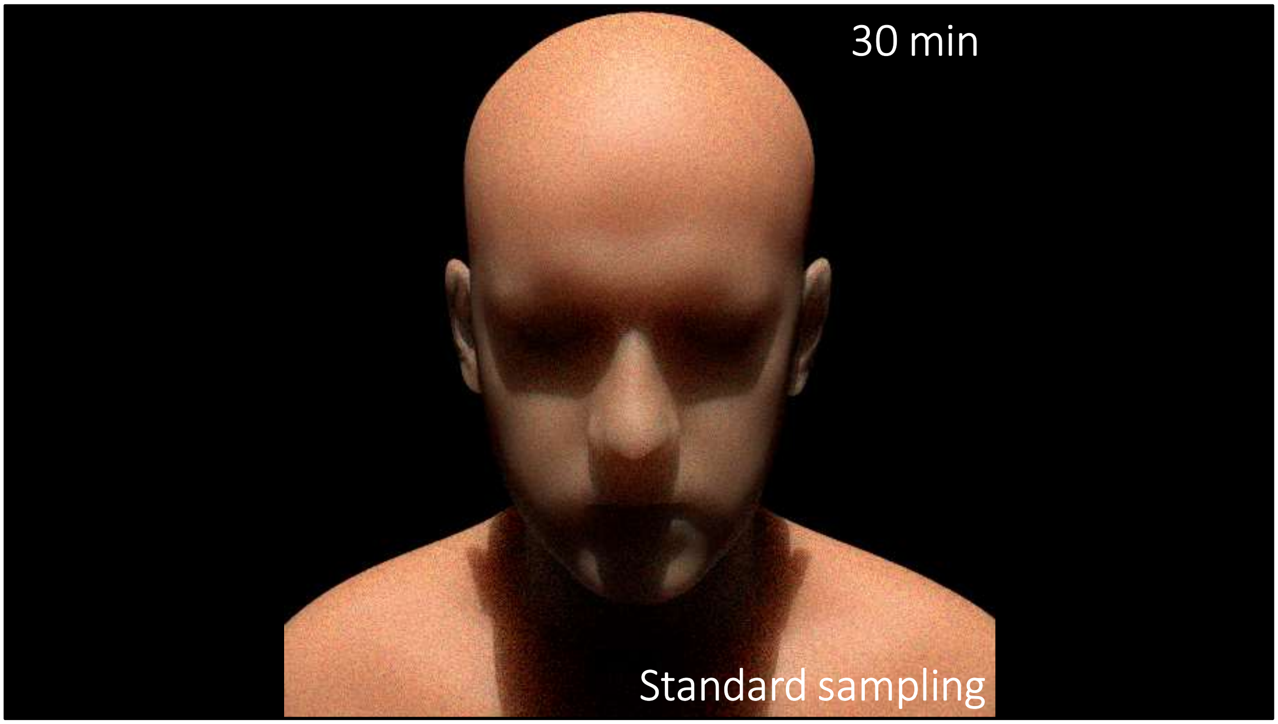




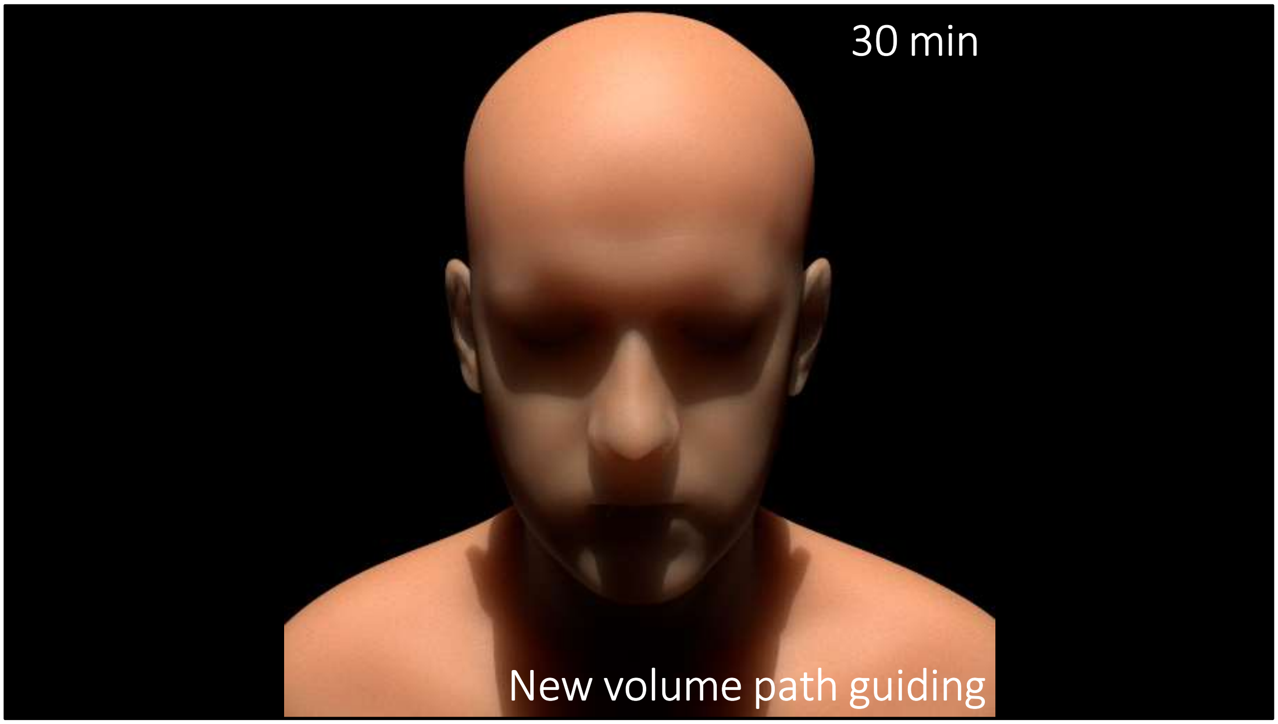

- In order to apply path guiding to volumetric transport, all the various random decisions used when constructing a light transport path need to be appropriately importance sampled (`guided').
- This includes the selection of scattering distance along a ray, and the decision whether the scattering should occur in the volume or at the next surface interaction. These decisions are unique to volumetric light transport and do not appear in surface transport.
- Furthermore the decisions shared with surface transport include the choice of the scattering direction and random termination/splitting of the paths.



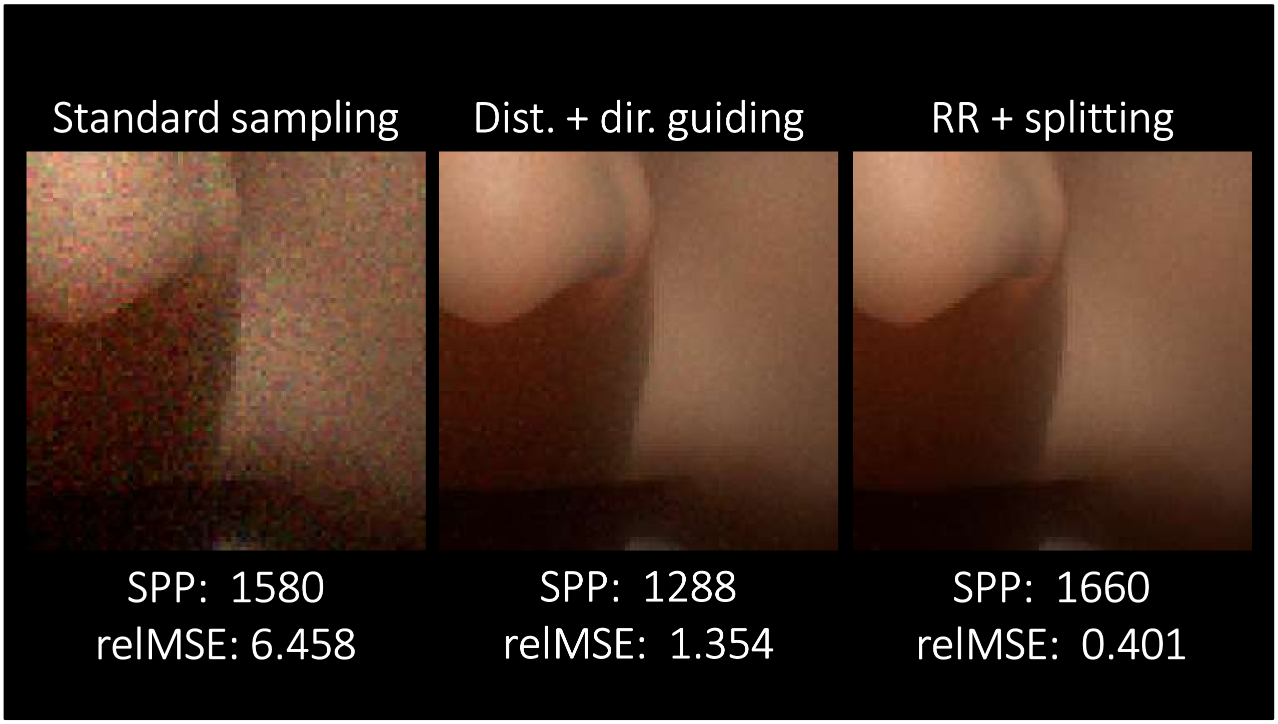

- Without giving any further details, let’s have a look at some results.
- This is a homogeneous medium with scattering properties approximating these of a Caucasian skin.
- We use Monte Carlo path tracing to render the scene.




- With standard path sampling, we can see that even after 30 minutes of rendering, the image shows a significant amount of noise.




- Our volumetric path guiding based in the radiance distribution learned from Monte Carlo samples yields a nearly clean image in the same rendering time.

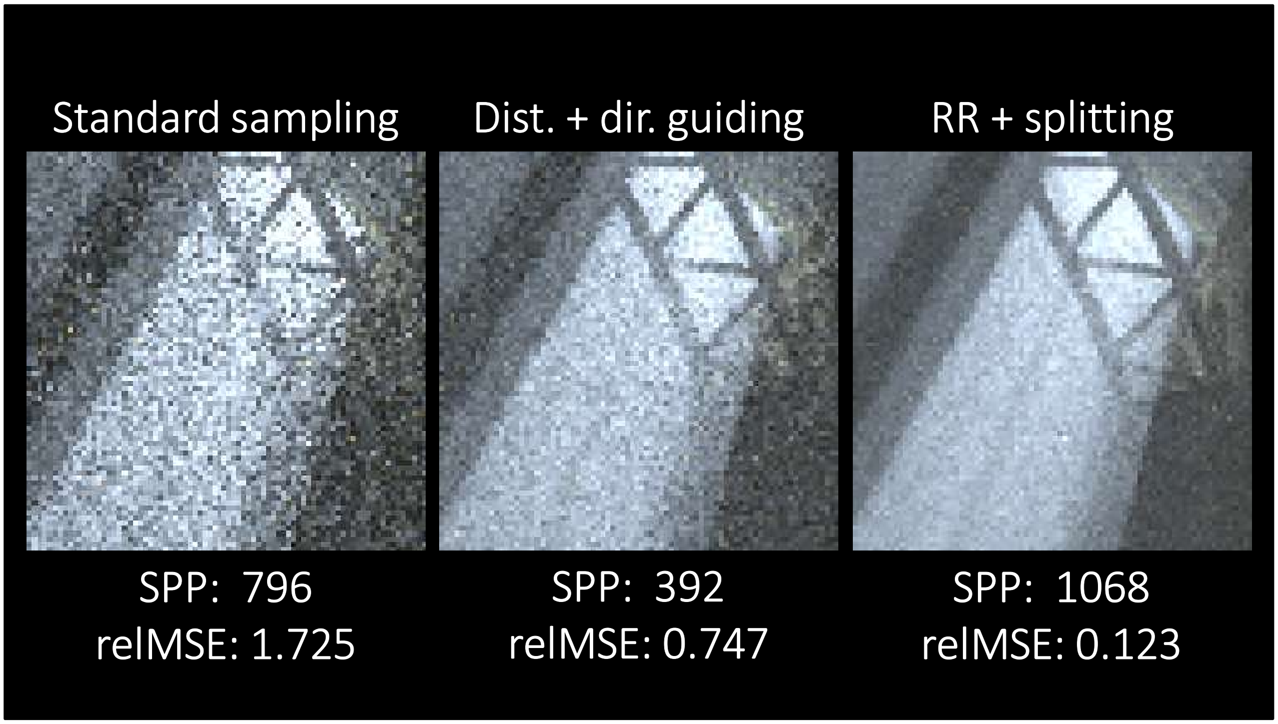



- This slide shows that the different random sampling decisions complement each other and together they yield the desired variance reduction.


- Here, we show the same technique applied to a very different scene – this time a natural history museum filled with thin haze and illuminated by light shafts.

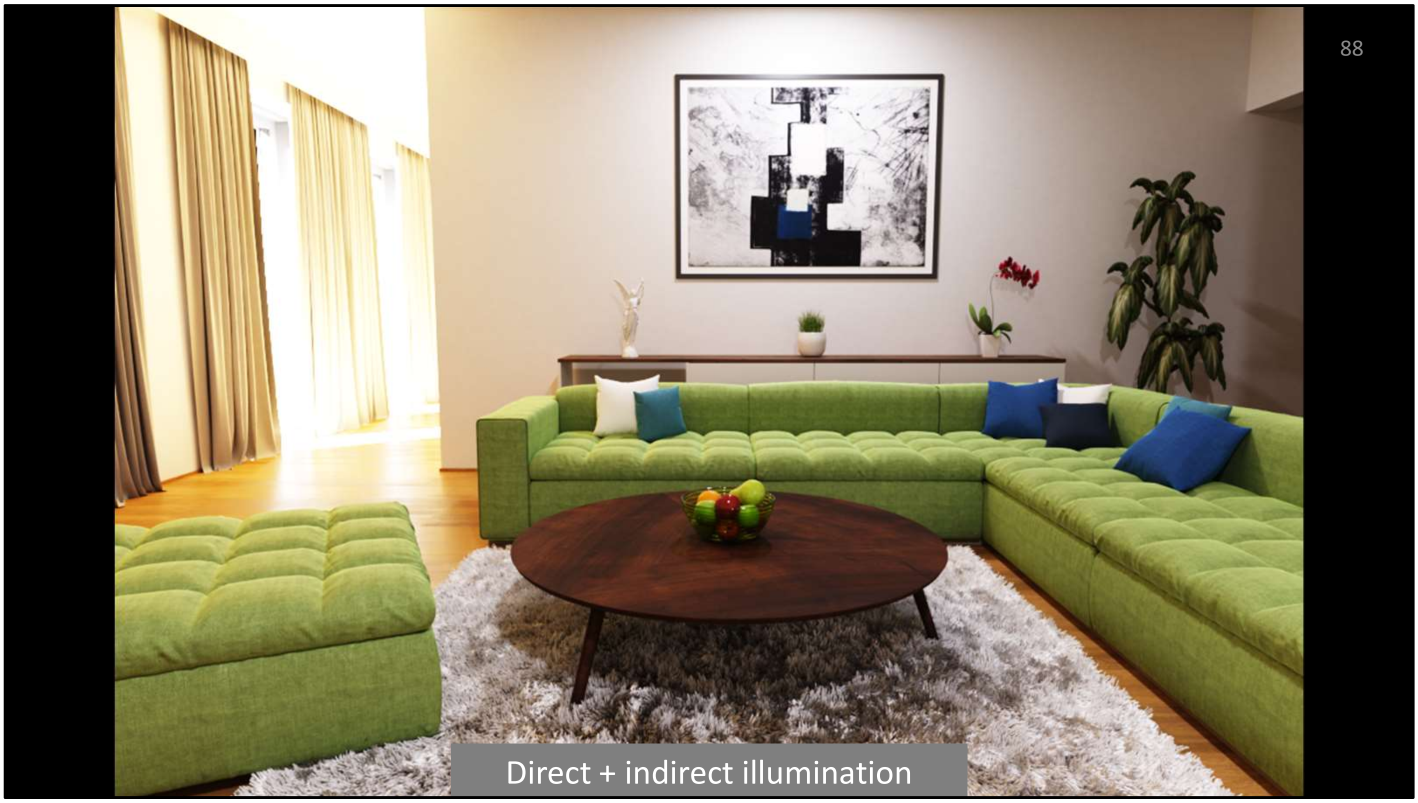

- Again, the standard sampling shows a significant amount of noise…


- While the path guiding based on the extra information provides a significant variance reduction.

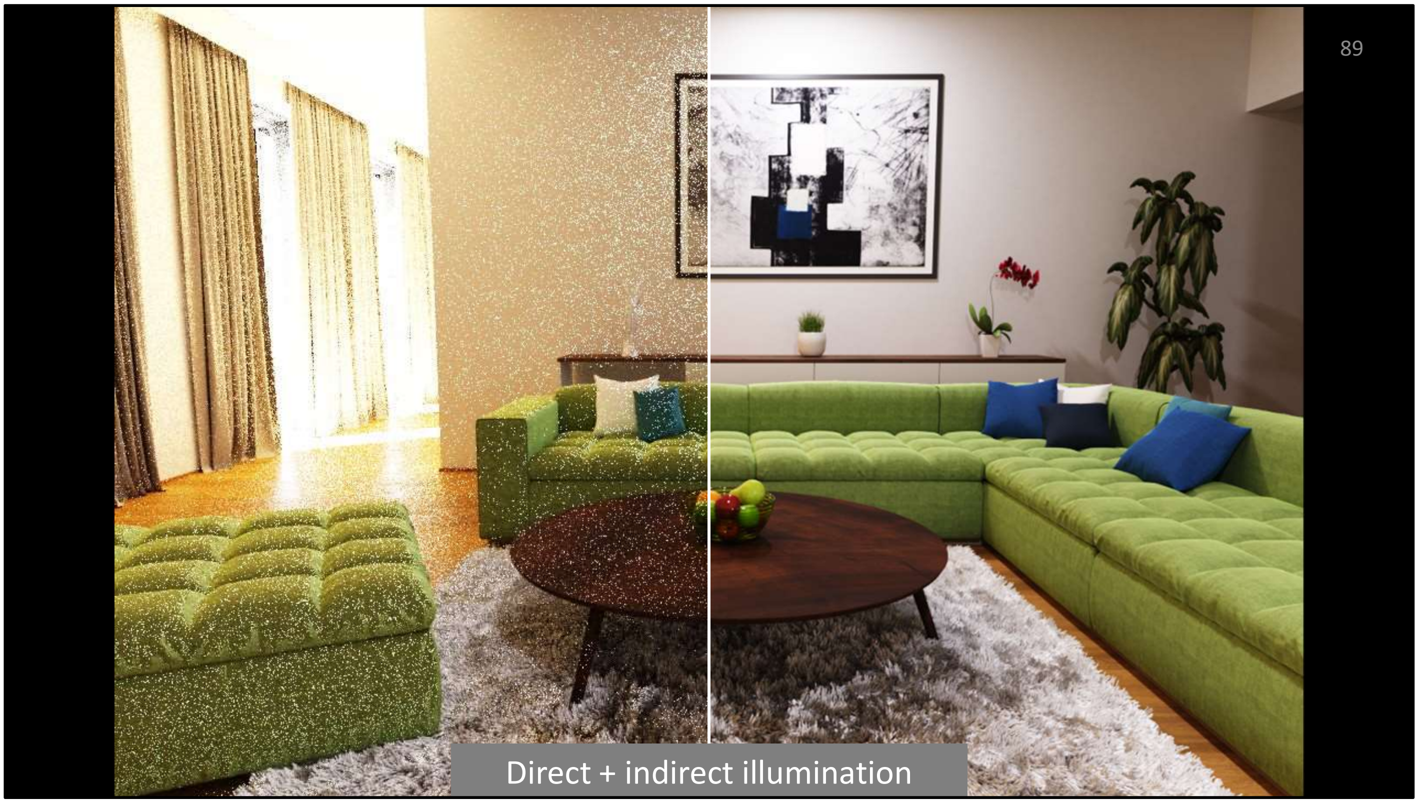



- Once again, we can see that the different random decisions add up to yield the final solution.

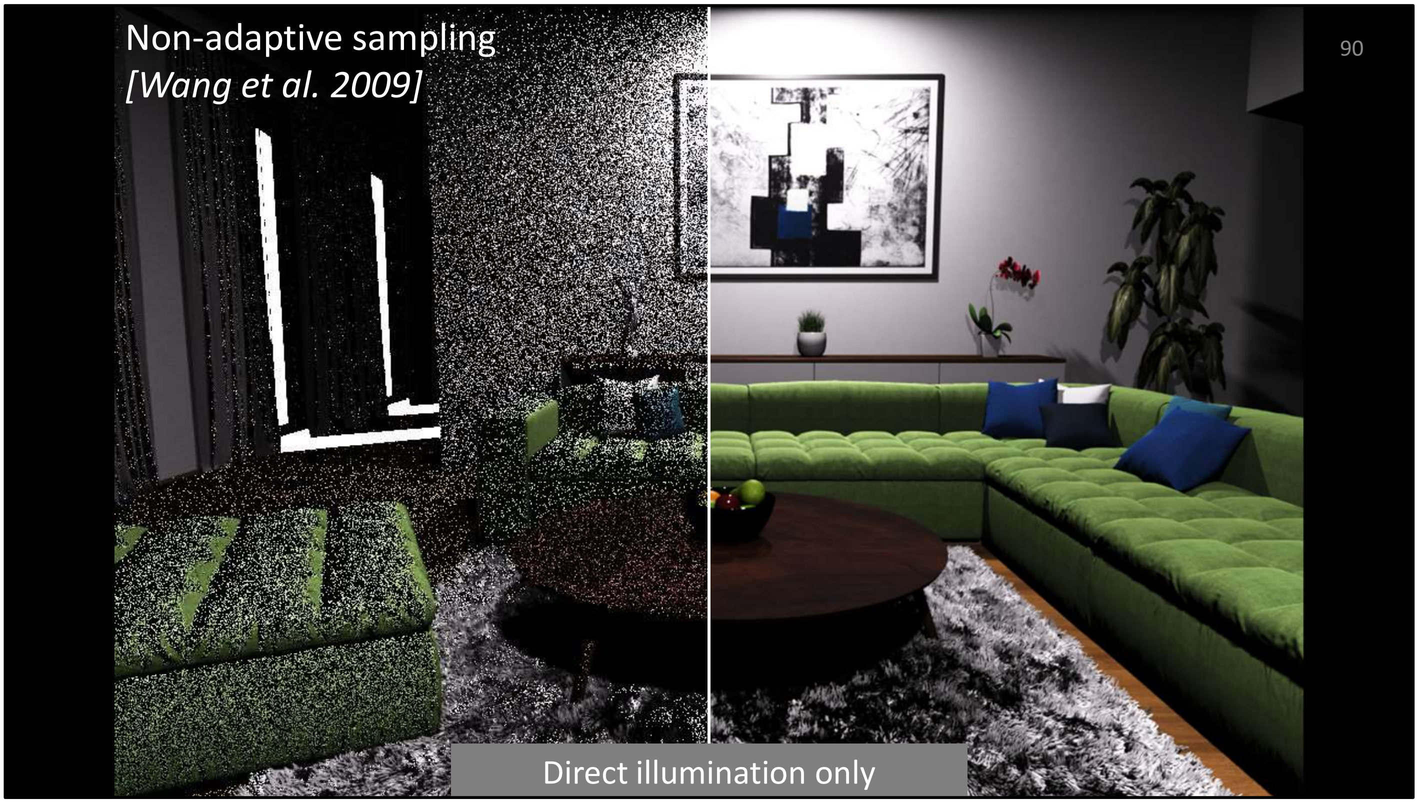

- We now move to our recent work, where we apply a different tool from the Machine learning toolbox (this time **Bayesian regression**) to the problem of adaptive direct illumination sampling.



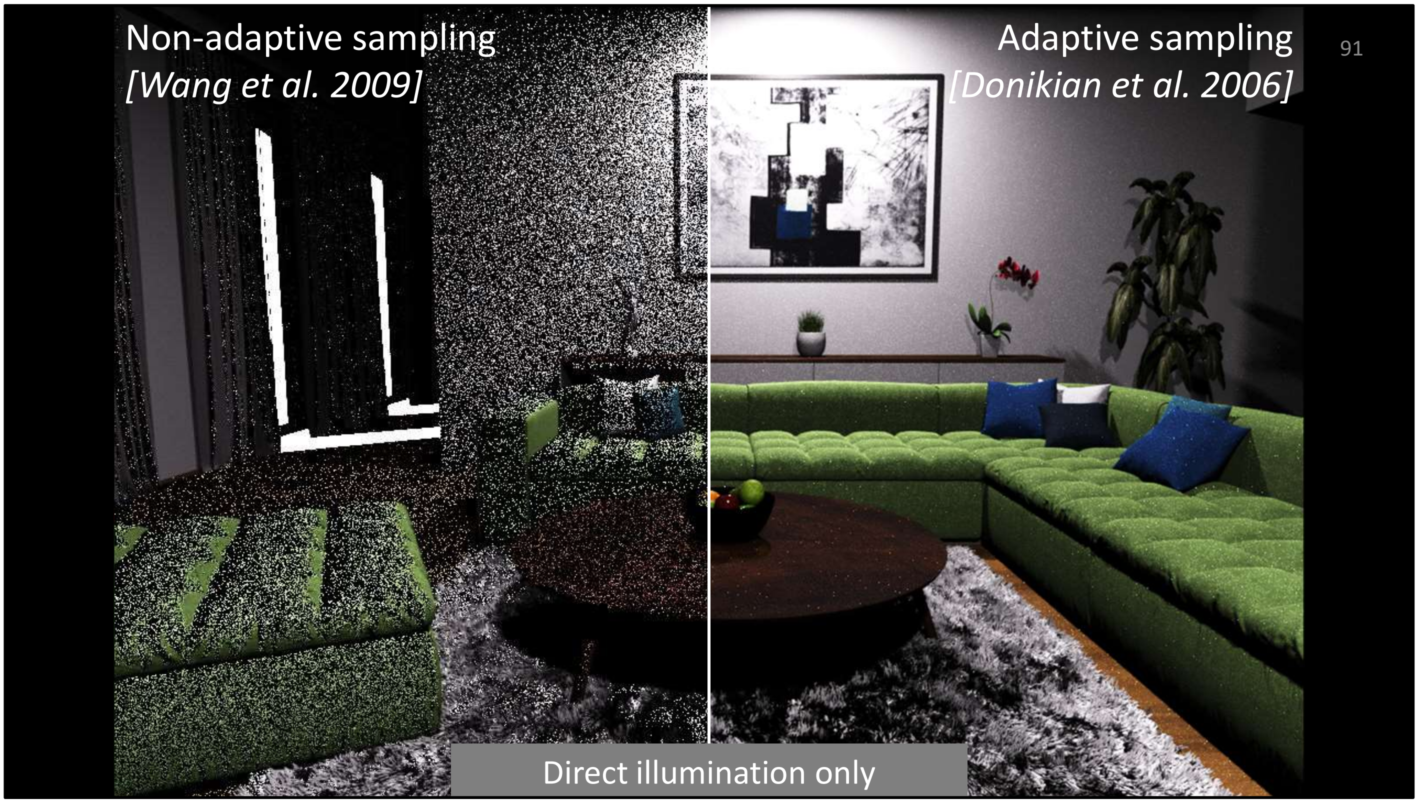

- MC rendering algorithms – including path tracing – suffer from noise.



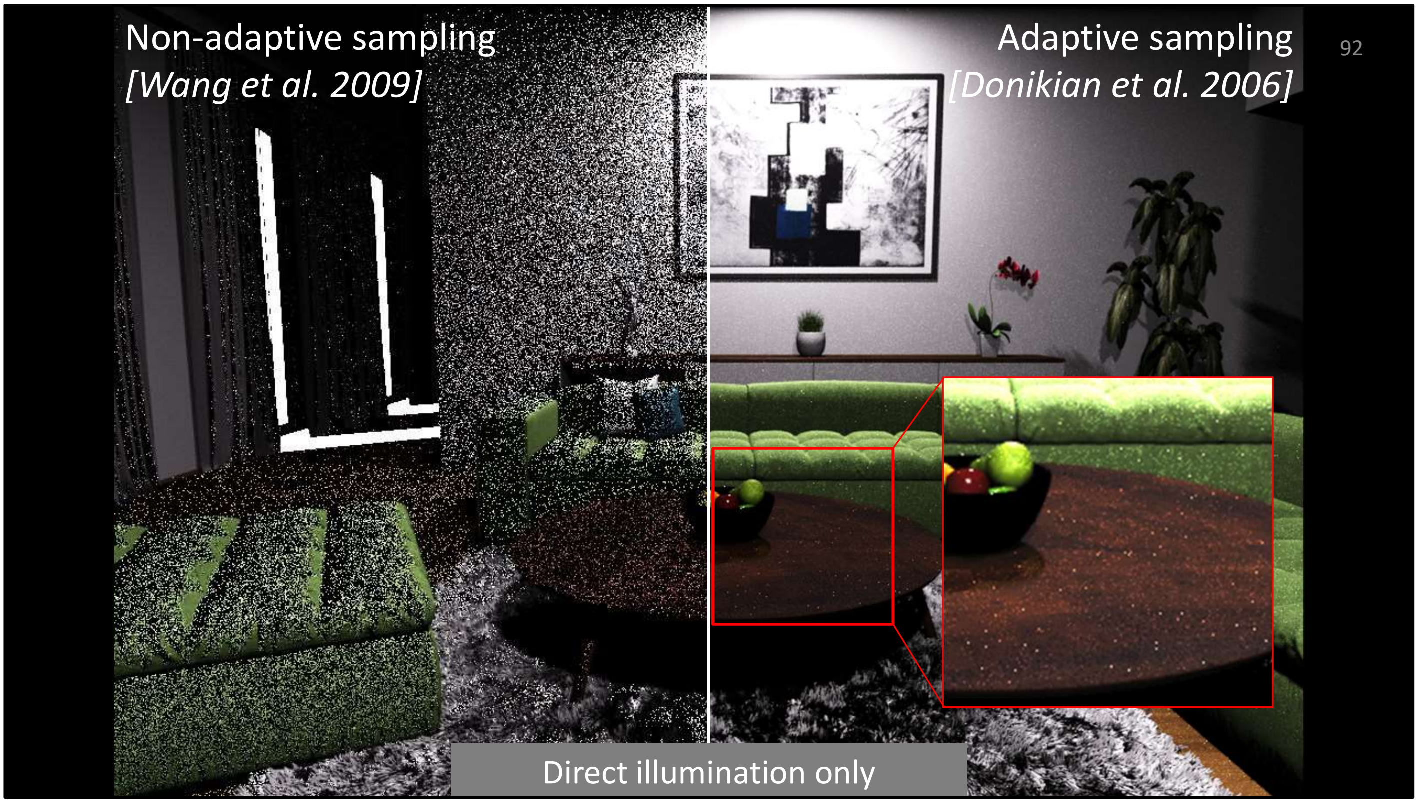

- Traditionally, the *indirect illumination* component has been considered as the main source of noise, and it’s been subject to lot of research, … but in this scene it is actually the direct component which causes the trouble.




- In this image with direct illumination only we can see it clearly. The non-adaptive method shown on the left struggles to work efficiently, because it wastes lot of samples on strong but completely occluded sun.

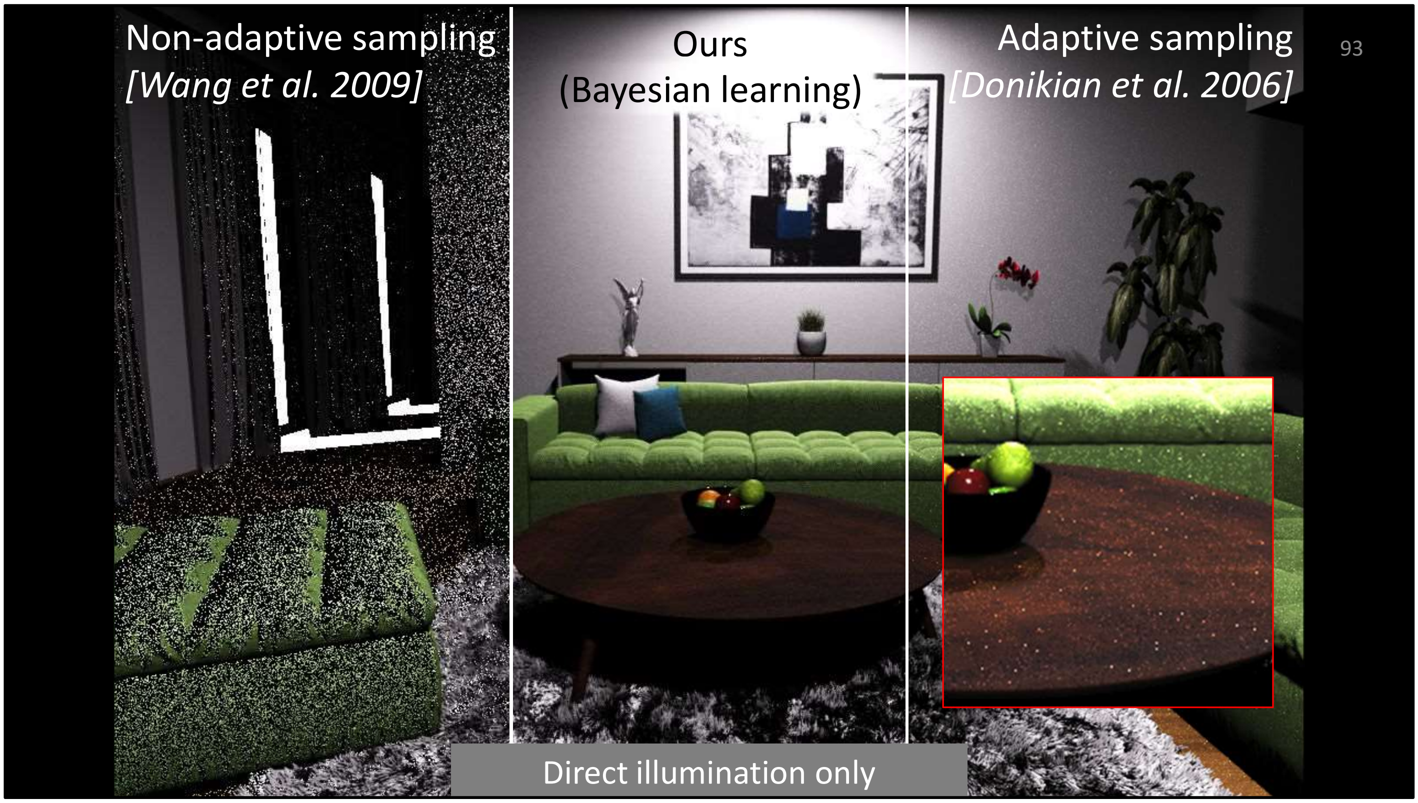



- Possible solution are *adaptive methods* which try to improve sampling based on past samples. But while they can decrease the amount of noise significantly, …

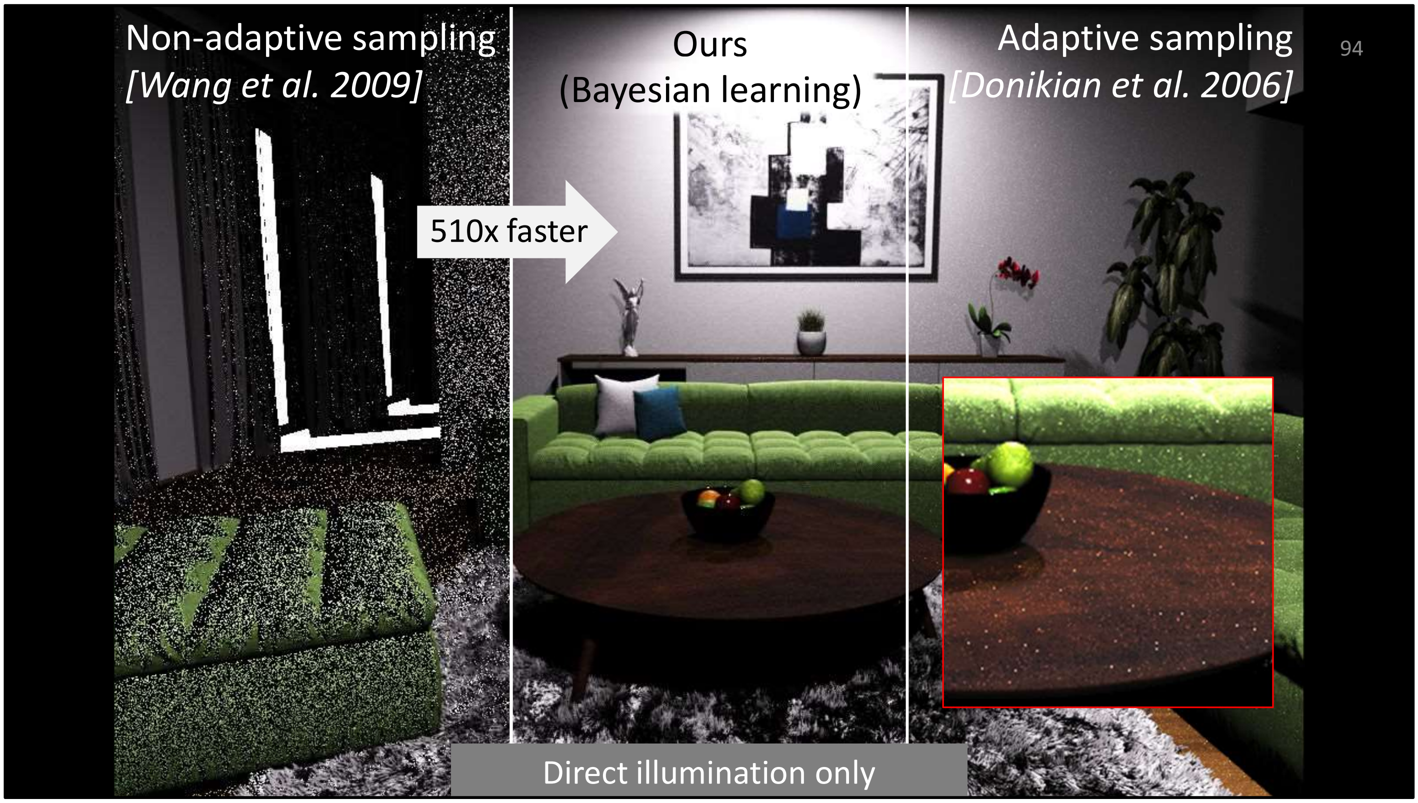



- they can also introduce artifacts or spiky noise because they are based on adhoc solutions and they tend to overfit.

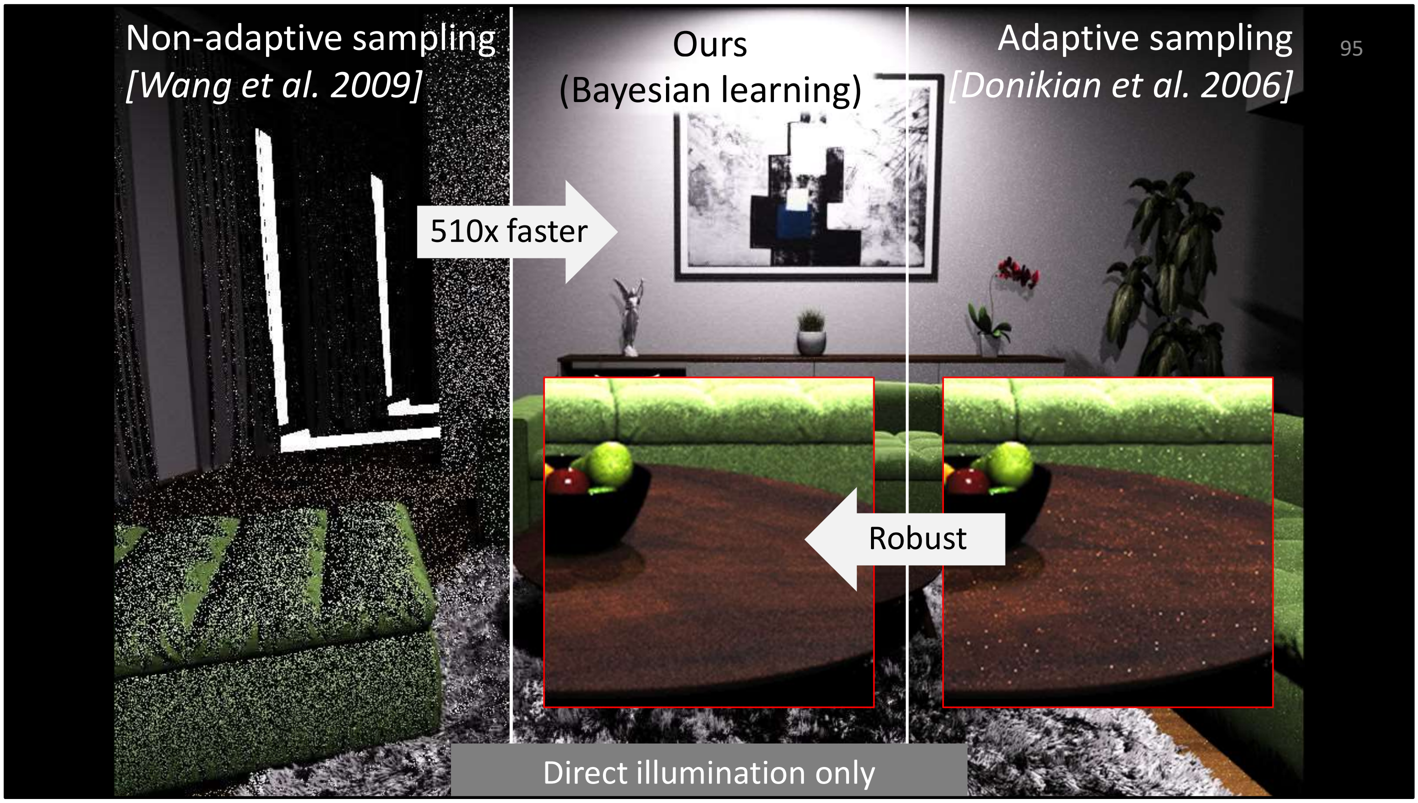
- This lack of *robustness* is a consequence of adhoc solutions to crucial questions in adaptive sampling: when is it safe to use the samples and how they should be combined with any previous knowledge?




- Therefore we propose a *first* solid theoretical framework for robust adaptive sampling in rendering.
- We draw on the Machine learning work and coin the problem as **Bayeasian regression**.




- In this scene, our solution is more than 500 times faster than the non-adaptive solution…




- … and we can achieve much better robustness.

- The new framework is not limited to the direct illumination. We are certain that other applications of adaptive sampling will benefit from it as well.

# Previous work



In the context of Monte carlo simulation there is a lot of work related to ours.

# Adaptive sampling

- General Monte Carlo
  - Vegas algorithm
    - [*Lepage 1980*]
  - Population MC
    - [*Cappé et al. 2004, ...*]
- Rendering
  - Image sampling
    - [*Mitchell 1987, ...*]
  - Indirect illumination (path guiding)
    - [*Dutre and Willems 1995*, *Jensen 1995*, *Lafortune et al. 1995, ...*]
    - [*Vorba et al. 2014, Muller et al. 2017*]
  - Direct illumination
    - [*Shirley et al. 1996*, *Donikian et al. 2006, Wang et al. 2009*]



- Adaptivity in Monte Carlo simulation is not a new concept.

- There is lot of work in the context of general Monte Carlo as well as in rendering, for example works dealing with image sampling, Indirect illumination and also in direct illumination.

- One of the oldest adaptive algorithms for Monte carlo estimation is Vegas by Lepage which works by histogramming integrand and using these histograms for sampling in next steps.
- Another example are population Monte carlo algorithms, which use population of particles and track how well they sample the integrand and based on that they keep the best individuals.
- In the context of rendering we can find a lot of work in image space sampling, we just mention the old work by Mitchell which deals with allocation of more samples into image parts with high-frequency content.
- In Indirect illumination computation first works were by Dutre and willems, where authors adaptively shot particles from lights (TODO: ziskat paper, nebo se zeptat Jardy), Jensen who used photon maps to construct sampling densities and Lafortune et al. who applied Vegas algorithm onto Monte carlo simulation of light transport.
- Along the years there were several other works dealing with this topic, but most recently

there was work by Vorba et al. which uses gaussian mixtures for guiding and Muller et al. which is a revamped version of algorithm by lafortune from 1995.

- Regarding direct illumination we mention a pioneering work by Shirley et al. who adaptively classified lights into important and unimportant ones and more recent work by Donikian et. al. which is closely related. Wang et al. sampled lights adaptively based on surface reflectance and estimates of lights' contributions.
- *None of these works deal with a problem of determining when and how to incorporate new information into the current sampling model.*

# Bayesian methods in rendering

- Filtering
  - NonLocal Bayes [*Boughida and Boubekeur 2017*]

- Global illumination
  - Bayesian Monte Carlo [*Brouilat et al. 2009, Marques et al. 2013*]
  - Path guiding [*Vorba et al. 2014*]



- Bayesian methodology on the other hand is not used very much in rendering.

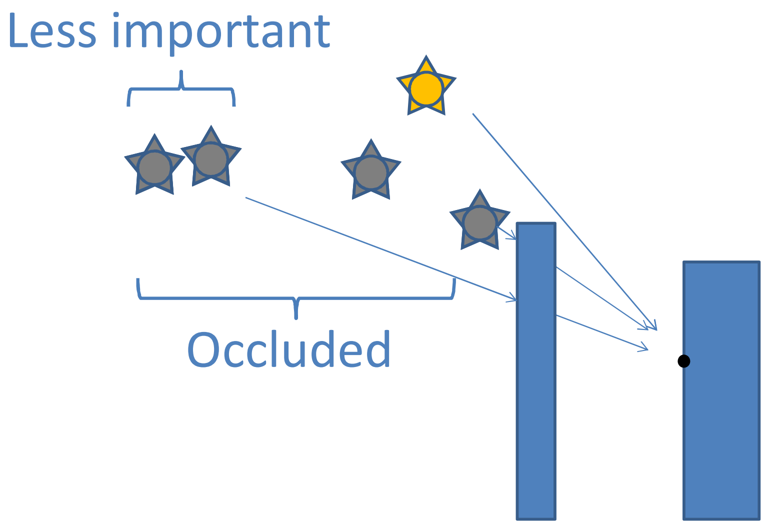
- There are just few methods in filtering and Global illumination.

# Background



Now let me give you some background related to direct illumination problem

# Direct illumination

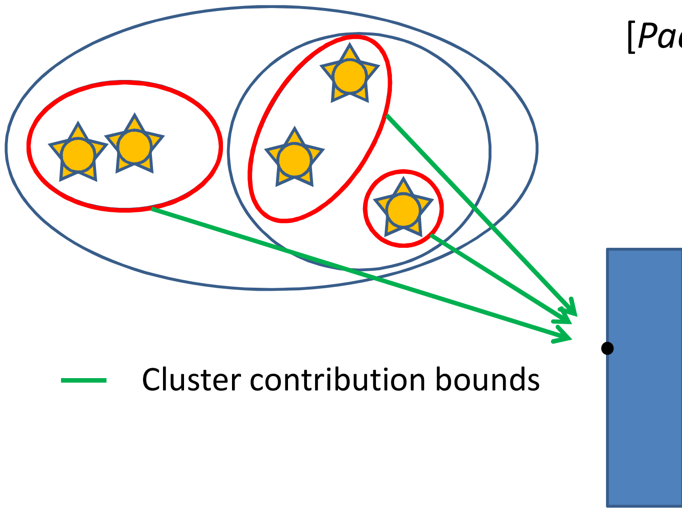





- What exactly is the direct illumination?
- We have a scene with several lights and some geometry
- And we estimate the ***direct*** contribution of each ligth onto each point in the scene. That is still a complex task ...
- due to uneven luminaire importances
- and due to occlusion

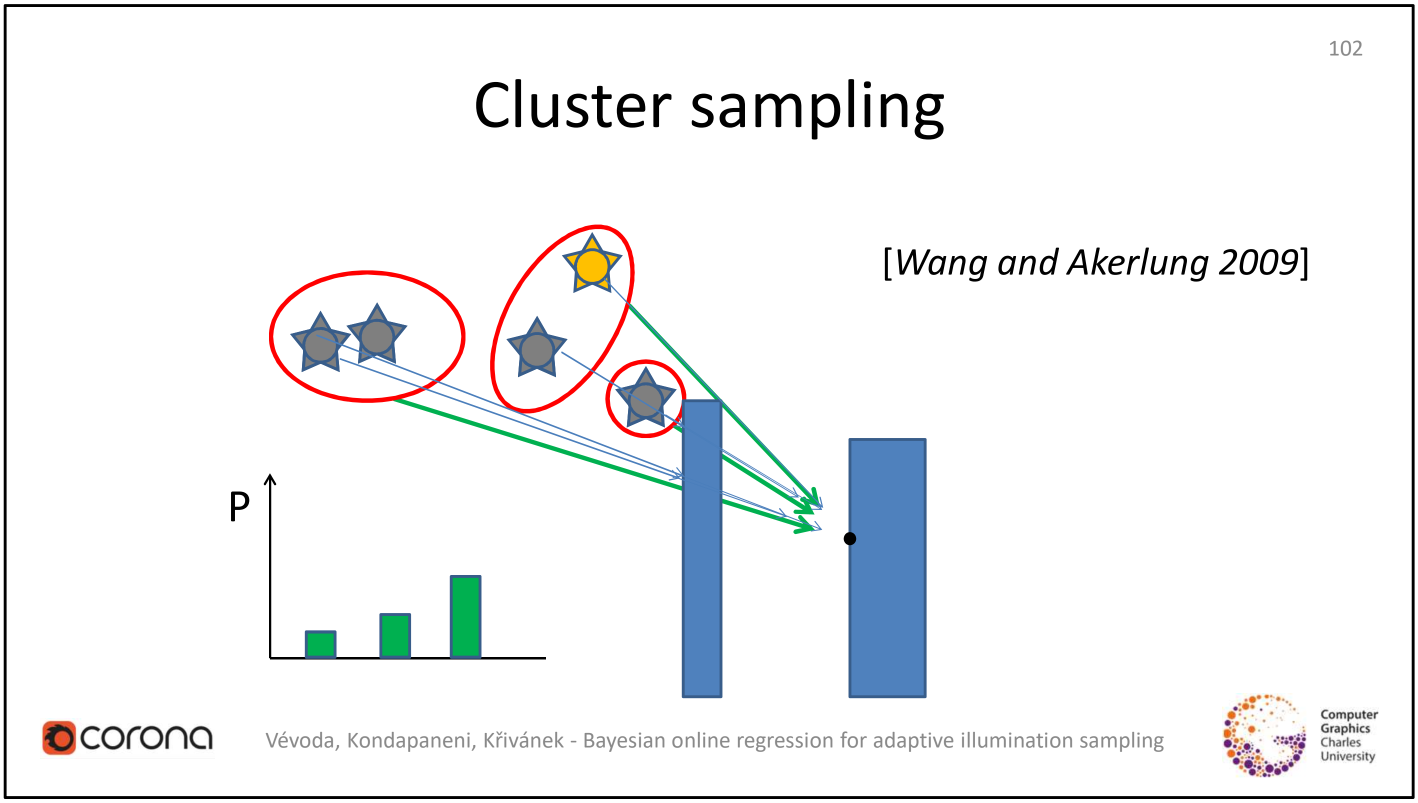
# Clustering (Lightcuts)

*[Paquette et al. 1998,*
*Walter et al. 2006]*

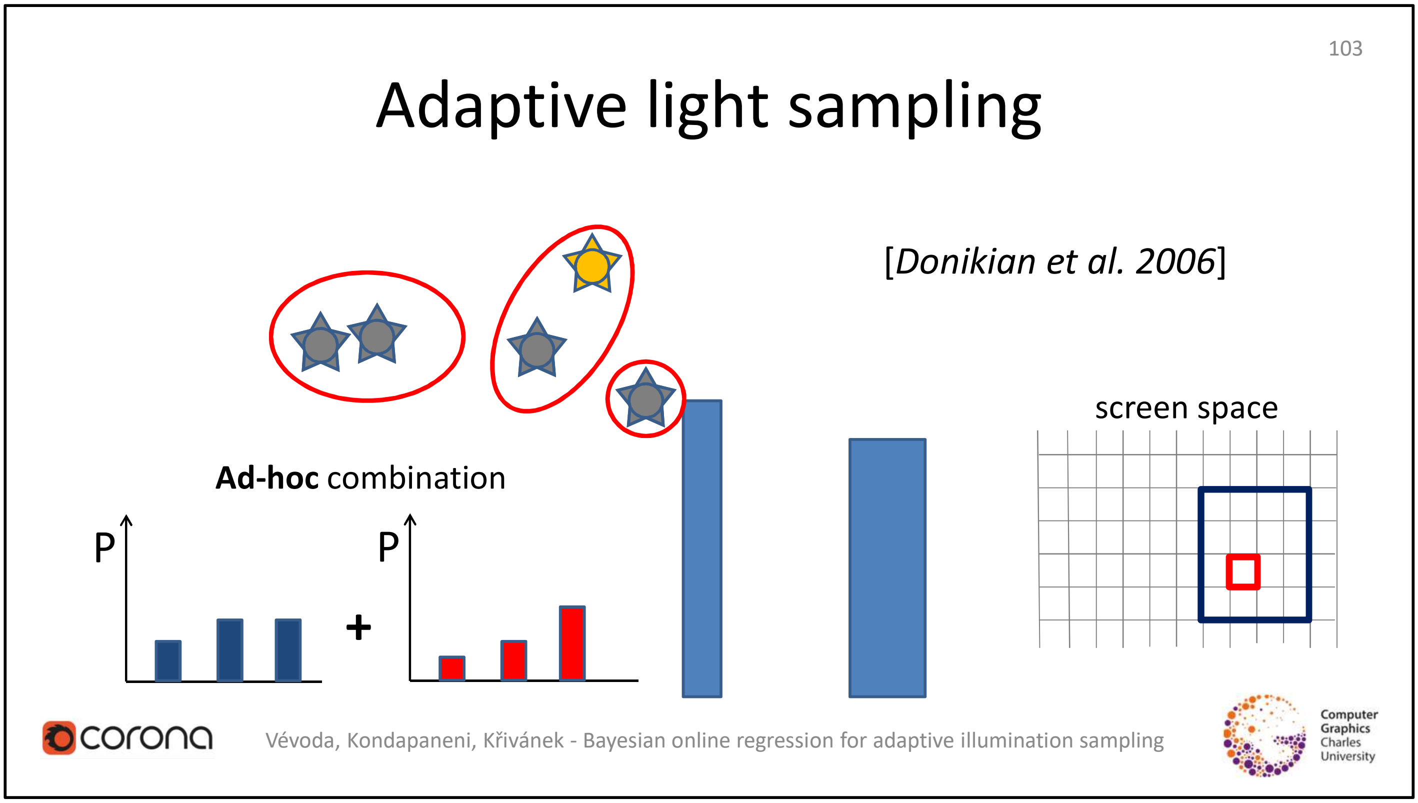




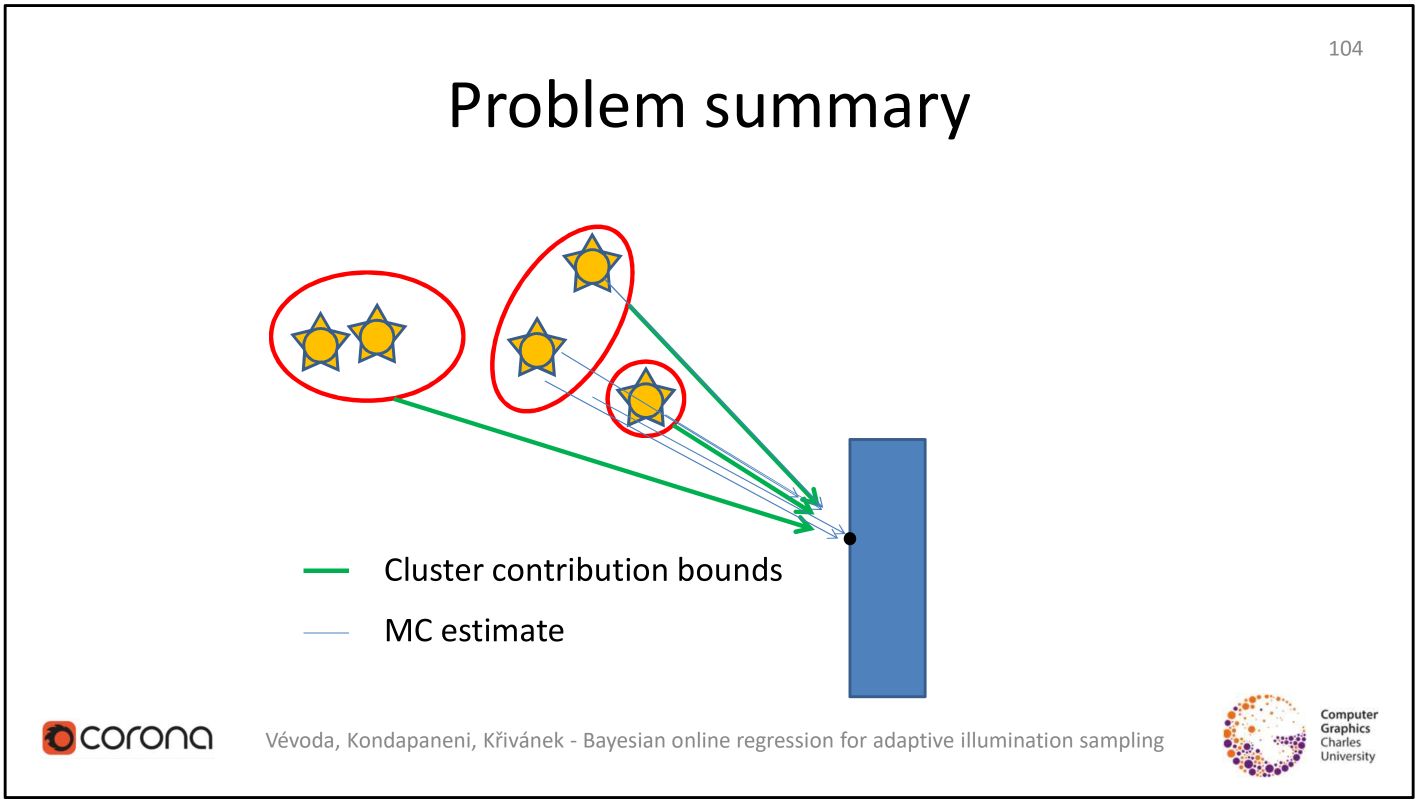

- One way to improve direct light estimation is to improve scalability when there is a lot of lights in the scene
- We can do so by hierachically clustering the lights
- and for each point in the scene we can choose clustering with the lowest approximation error.
- Each cluster gives us a conservative bound on its contribution to the point.




- Then we could use cluster contribution bounds to
- build a sampling distribution over clusters
- and then use it for getting MC estimates.
- At this point, we still wouldn't have adaptivity, and we wouldn't be able to capture complex visibility.




- We could achieve adaptivity as Donikian et al. [2006] did.
- They gather statistics about clusters in screen space and
- They do so for each pixel and from the statistics they build sampling over clusters. But these estimates are very noisy.
- Therefore they do the same also over the whole block of pixels and then
- they mix both information distributions, but in an ad-hoc way which may sometimes result into overfitting and artifacts in a picture.




- So our goal is to compute direct illumination by means of Monte Carlo, and for that we need to find **optimal** sampling distribution over clusters. We want to have **adaptive solution** because adaptivity has huge potential, but we strive for a **robust solution**.

- In order to achieve these goals, we have two kinds of information about clusters at hand:

  - We have cluster contribution bounds towards a point, which are conservative, noise-free.  We have them right from the beginning so they can serve as our prior information.

  - And we have Monte-carlo estimates, which are noisy, and we get them over time.

- For a question how to combine these two sources of information together in a robust way we've found a good answer (as we explain shortly)

# Our approach



Let us now introduce our approach ...

106

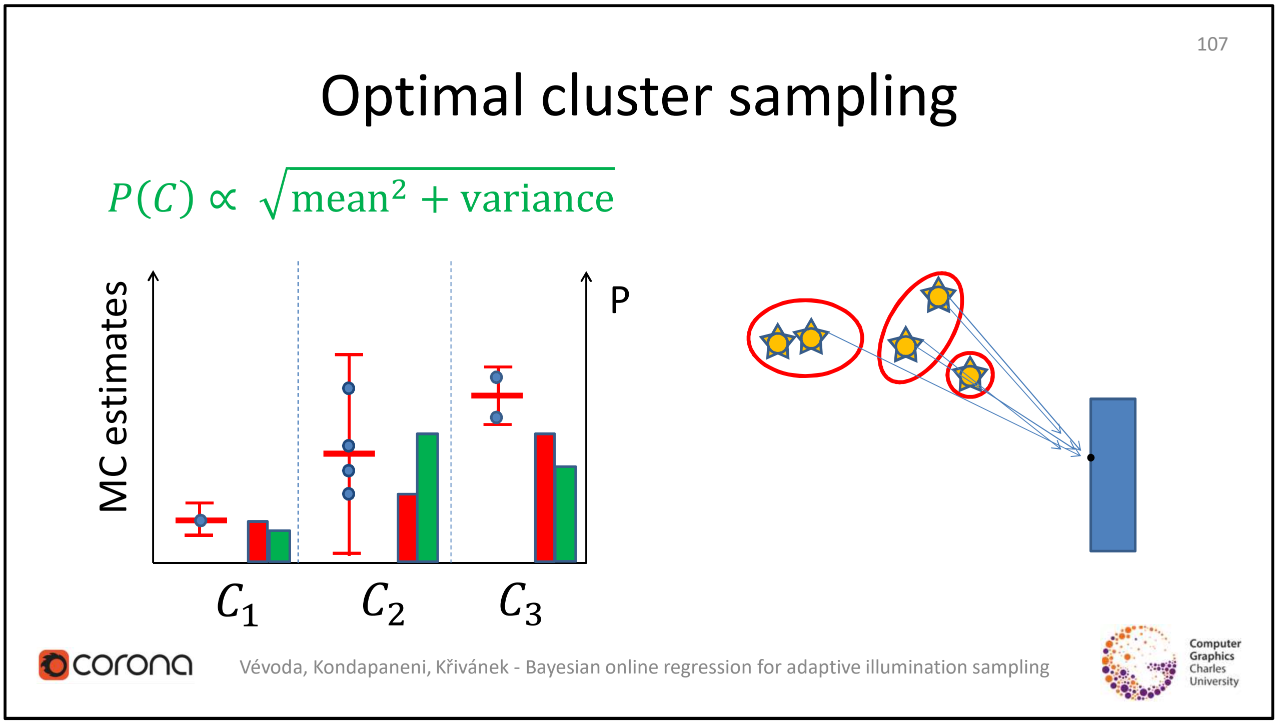
# Contributions

- Optimal sampling of clusters
- Adaptive sampling by Bayesian inference

corona Vévoda, Kondapaneni, Křivánek - Bayesian online regression for adaptive illumination sampling

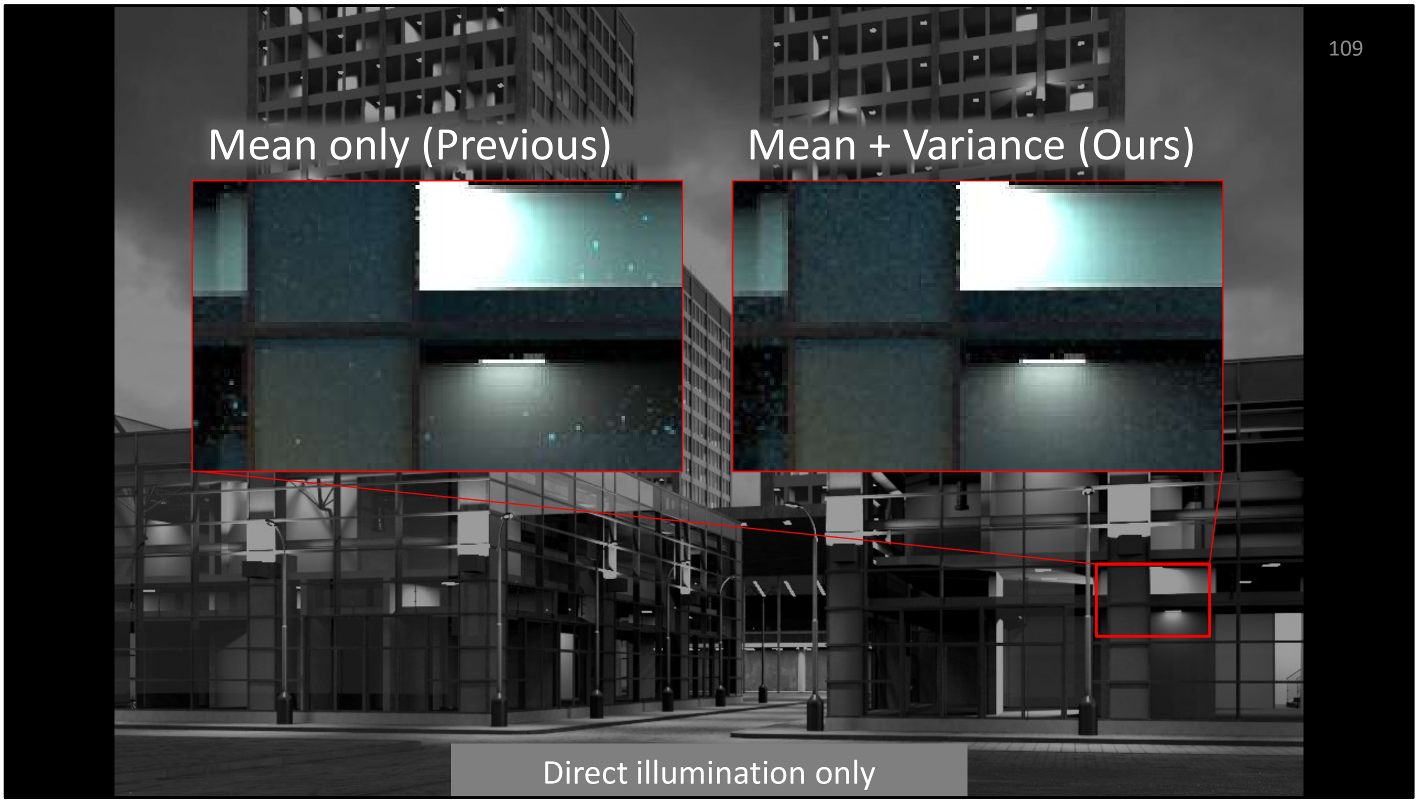
Computer Graphics Charles University

- Our main cotributions are two:
- We found the optimal sampling scheme of clusters
- And we do adaptivity with a help of Bayesian inference giving us a more robust solution and allowing us to combine Monte carlo samples with cluster contribution bounds in a principled way.
- Lets start with optimal sampling of clusters




- Given a scene, let’s have a look at how to derive sampling probabilities from MC samples
- We are sampling the clusters and getting our MC samples
- with some mean and variance.
- And normally, and what various approaches did in the past, the cluster's sampling probability would be proportional to the **mean only**.
- But we found out that variance of samples from each cluster has a big impact. The higher variance means that more samples should be allocated to it.
- Therefore optimal sampling should be proportional to the square root of the second moment of the samples.
- The sampling probability then can change drastically as is depicted by the green bars.


- Let me show you the practical example: this scene contains more than 5000 light sources so the clusters can be large and complicated….




- On the left we see an inset showing how sampling according to a mean performs. It undersamples some tricky cluster which leads to spiky noise.
- And on the right we see that sampling according to both the mean and the variance eliminates this issue.

110

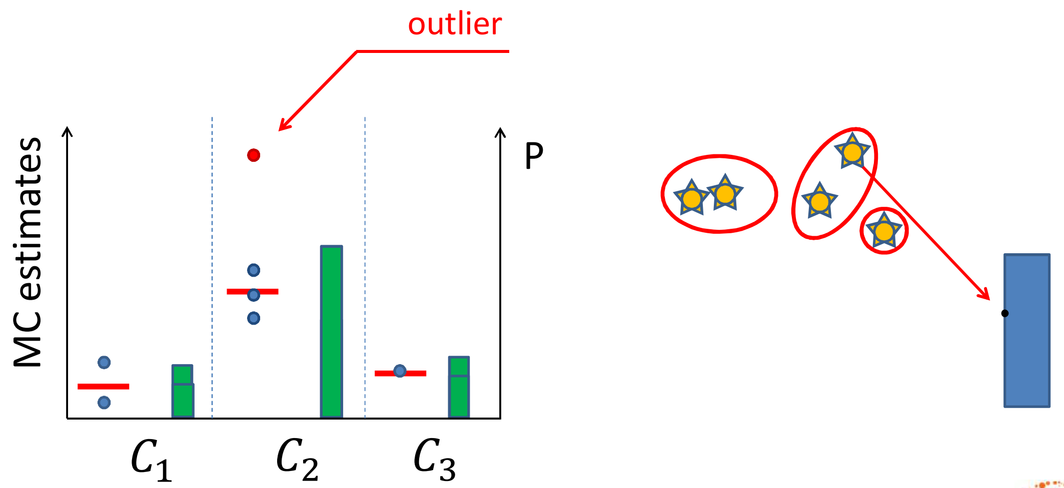
# Contributions

- Optimal sampling of clusters
- Adaptive sampling by Bayesian inference

corona Vévoda, Kondapaneni, Křivánek - Bayesian online regression for adaptive illumination sampling Computer Graphics Charles University

- Having explained the optimal sampling,
- we will show you now how to do the adaptivity the Bayesian way ...
- The issue is that the mean and the variance needed for the optimal sampling are not know upfront and need to be learned during rendering.

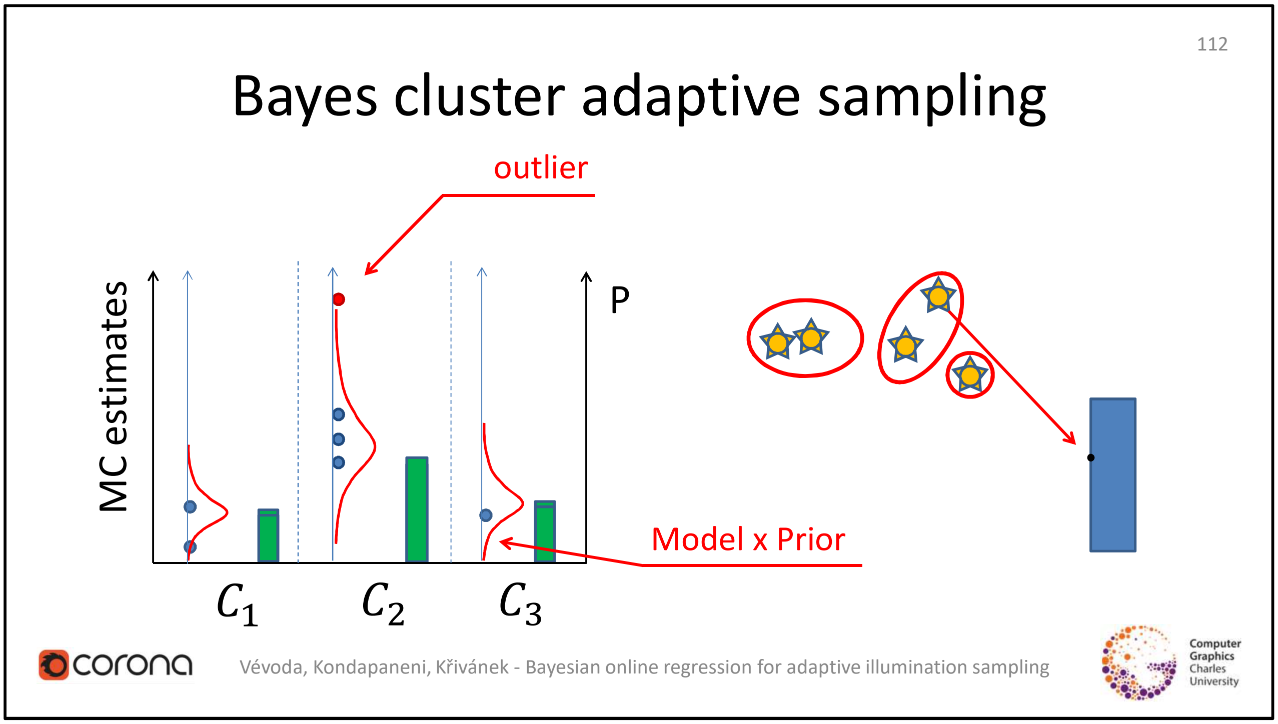
# Naive adaptive cluster sampling



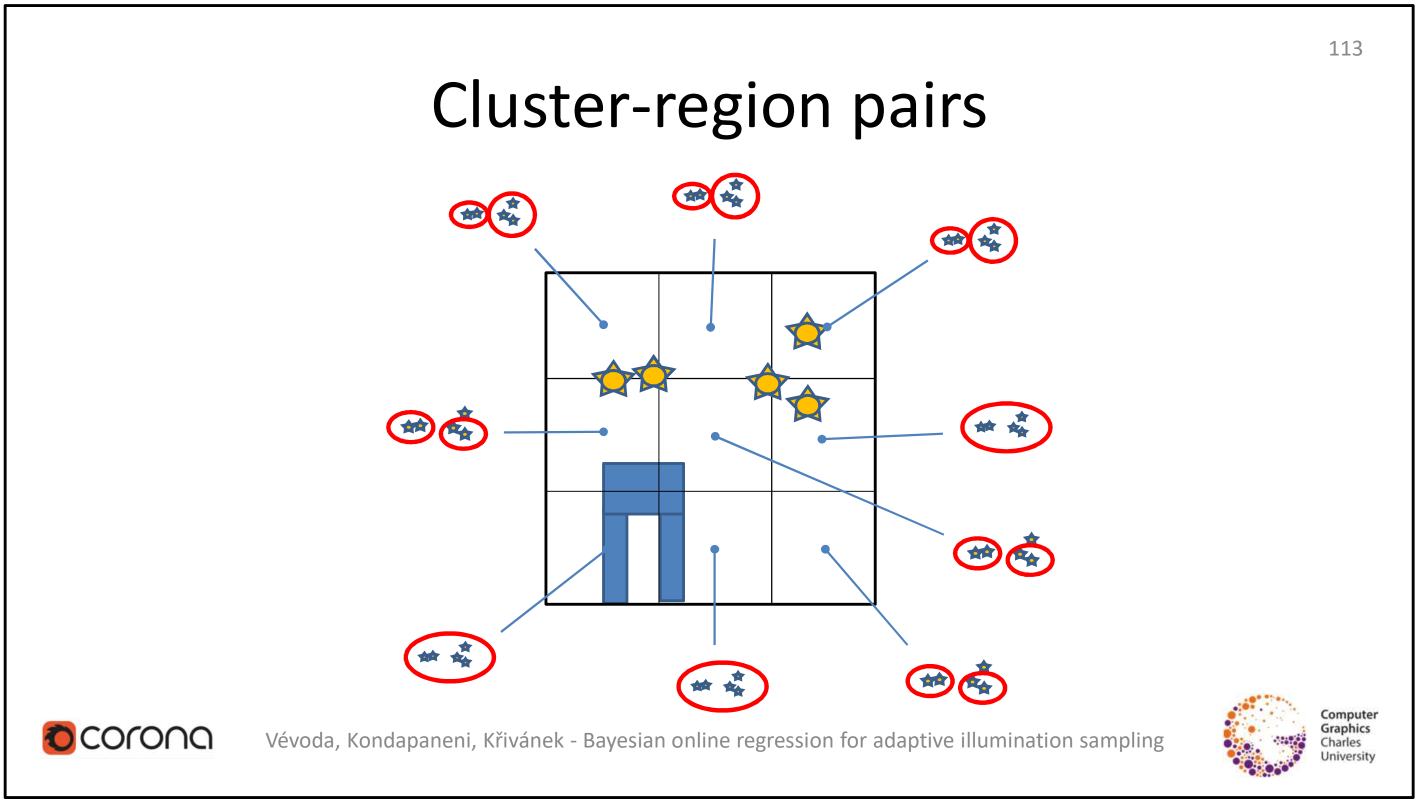

corona Vévoda, Kondapaneni, Křivánek - Bayesian online regression for adaptive illumination sampling

Computer Graphics Charles University

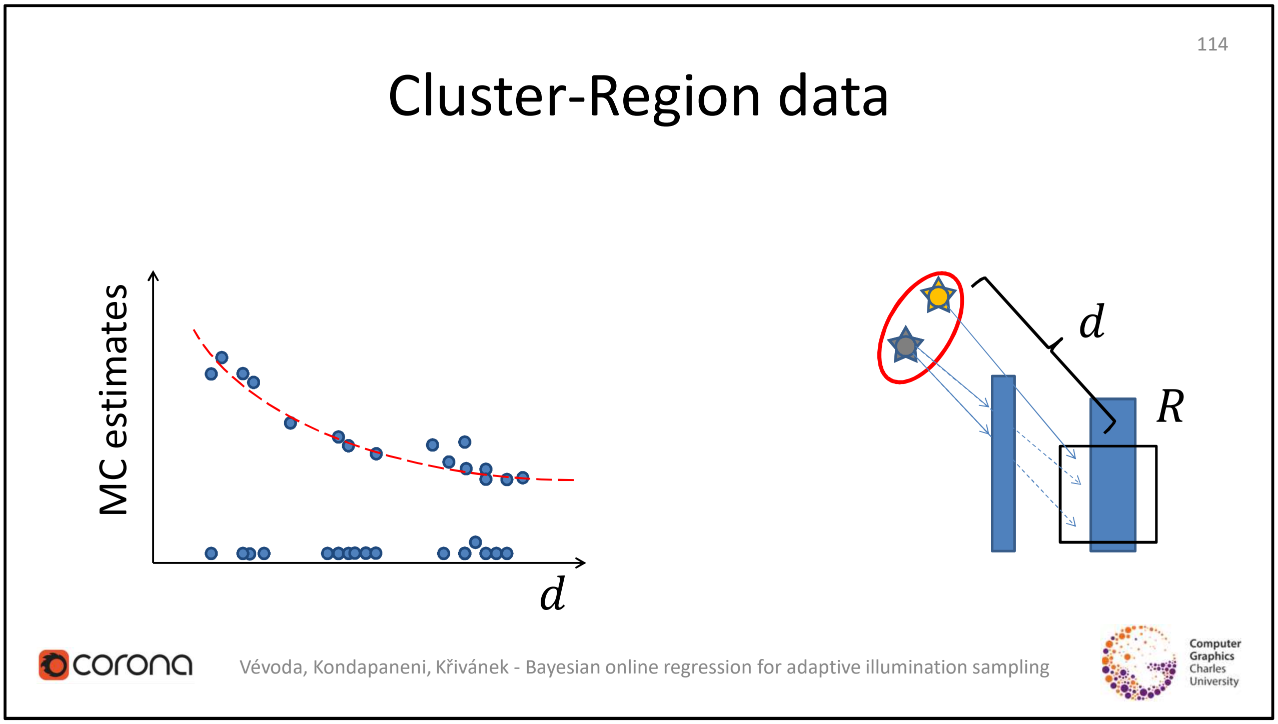
- First, lets have a look how naive adaptivity looks like. We have some samples from clusters and some cluster sampling probabilities
- And suppose we have gathered a new data point which happens to be an outlier
- If we estimate the sample means directly, our estimates can change abruptly and that will have a strong effect on further cluster sampling.
- (It might then take a long time to fix that decision. More probably it will cause an artifact in the final picture.)




- If we estimate means in a Bayesian way, we model the **distributions** of MC estimates seen so far while we also have some prior information about parameters of that distribution,
- Therefore, when we get a new sample,
- Our distribution changes less abruptly as well as cluster sampling probabilities derived from it, which yields a more robust solution.

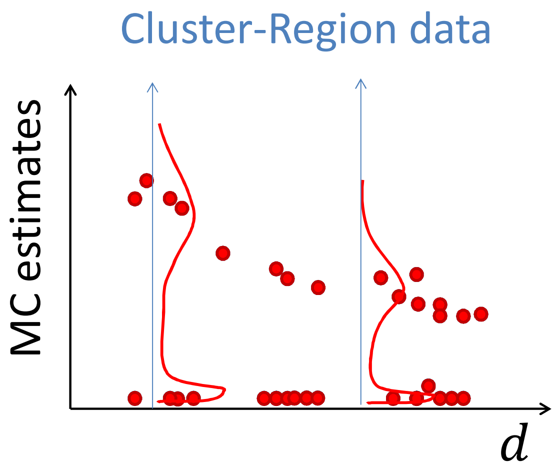



- Before we explain how we model the data we need to explain some basic context regarding clusters and scene subdivision.
- Contrary to the previous approaches, we split the scene into fixed **Regions**
- for each region we compute exactly one light clustering and keep it cached for that region. That speeds up clustering retrieval.




- Now we focus on samples collected from one cluster in one region in the scene.
- And we keep track of the distance *d* in the geometry factor in each estimate.
- We can then plot the data (i.e. MC estimates) for a cluster in a region which reveals relation of estimate values to distance. You may notice the inverse squared falloff with the distance and a number of zero-valued samples.

# Regresion Data model




Parameters:

$k, h$ - normal distr. parameters

$p_0$ - probability of occlusion

$$(1 - p_0) \times \; N(\text{est.} \,|\, \frac{k}{d^2}, \frac{h}{d^4})$$

$$p_0 \;\; \times \; \delta(\text{est.})$$

corona

Vévoda, Kondapaneni, Křivánek - Bayesian online regression for adaptive illumination sampling

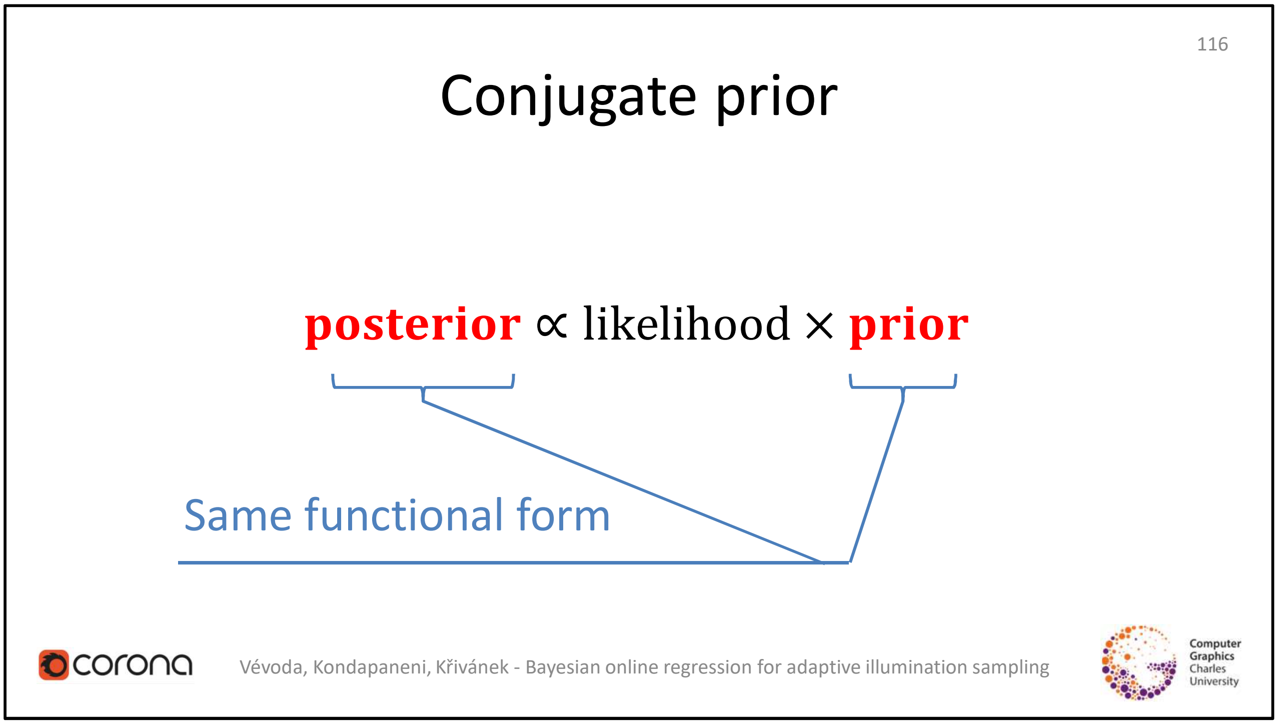
Computer Graphics Charles University

- Our model is therefore a parametric regression model, which for a distance yields a distribution of MC estimates. We design it as follows:

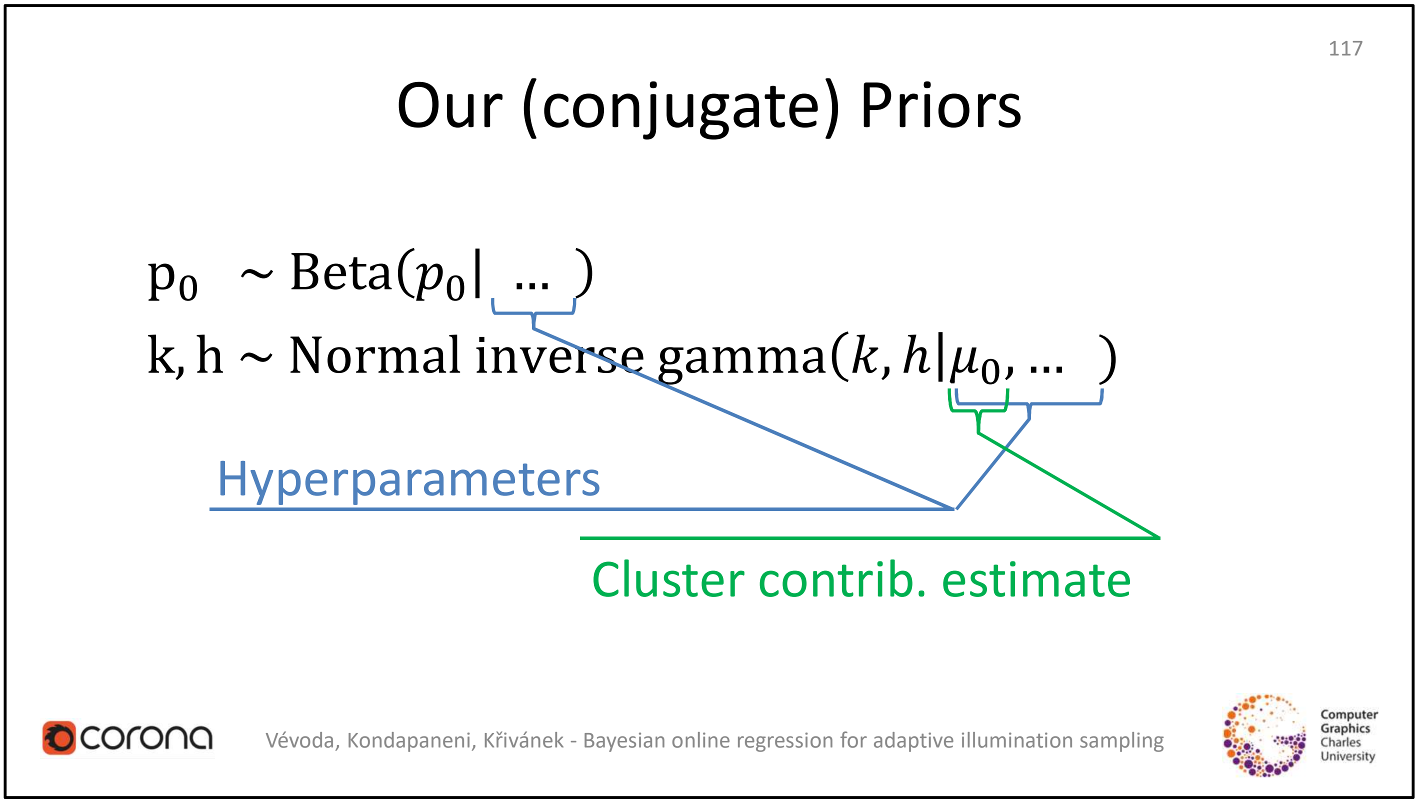
- Non-zero samples are modeled by a normal distribution with mean and variance being a function of a distance associated with samples. (This part has two parameters *k* and *h*.)

- The zero valued samples are incorporated by mixing the inverse-square distance falloff model with a delta function (and it is controled by $p_0$, which has a meaning of occlusion probability.)

- Now having designed our data model, we need to define prior for the model parameters




- The model we have just defined has parameters $k$, $h$ and $p_0$ and for them we use so called conjugate priors
- conjugate prior is such that when combined with likelihood it has the same functional form as the posterior.




- We proved that conjugate prior in our case is Beta distribution for $p_0$ and
- normal-inverse-gamma distribution for the parameters *k* and *h*.
- There are various hyperparameters in the equation, but one parameter which stands out is $m_0$, for which we use the conservative cluster contribution bound

118

# Algorithm summary

- Light preprocess (clustering)
- During each Next event estimation:
  – Obtain clustering (Cut) cached in a region
  – Compute distributions of estimates for each cluster in Cut
    -> mean, variance
  – Build distribution over clusters
  – Sample direct illumination
  – Record new data for sampled cluster

corona Vévoda, Kondapaneni, Křivánek - Bayesian online regression for adaptive illumination sampling Computer Graphics Charles University

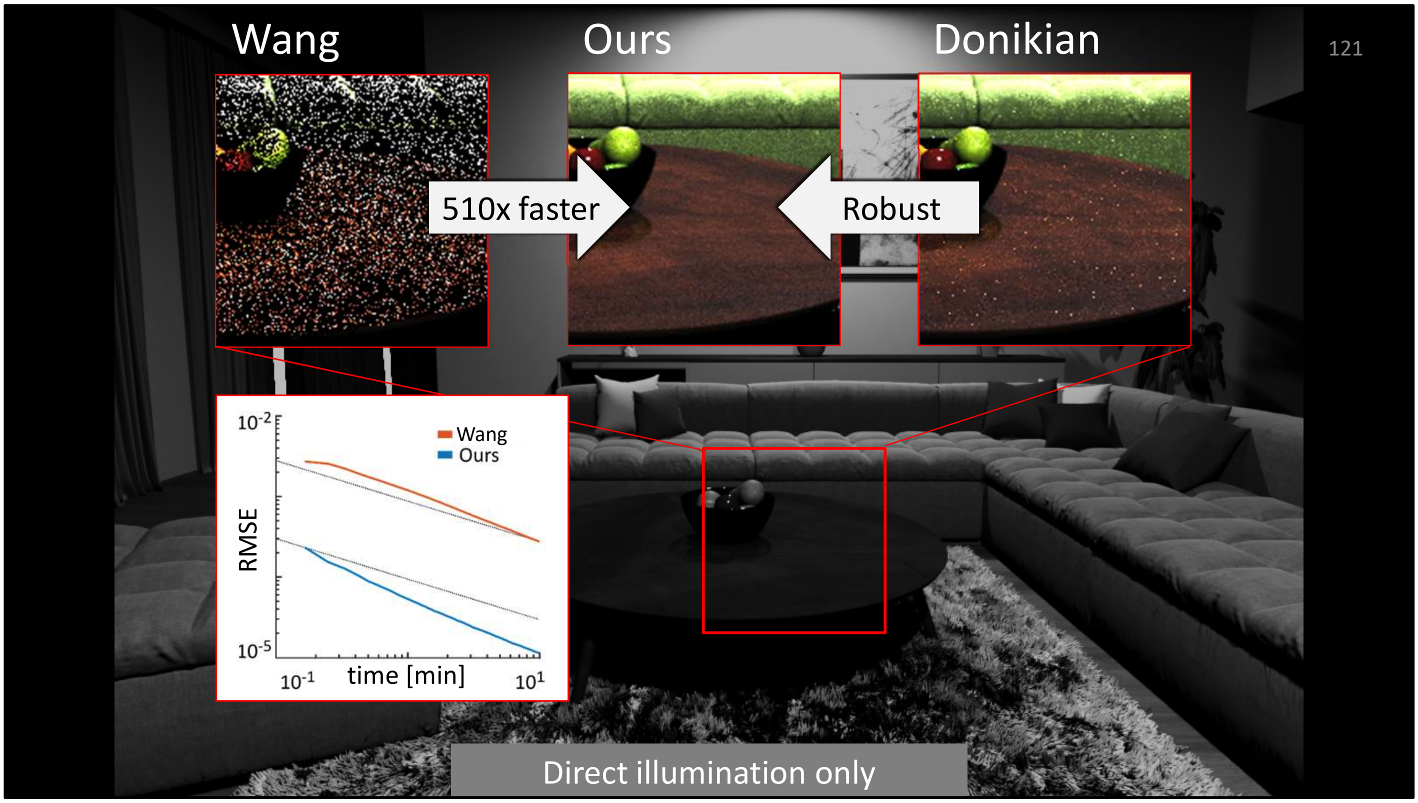
- To wrap it up, our algorithm is following:

# Results

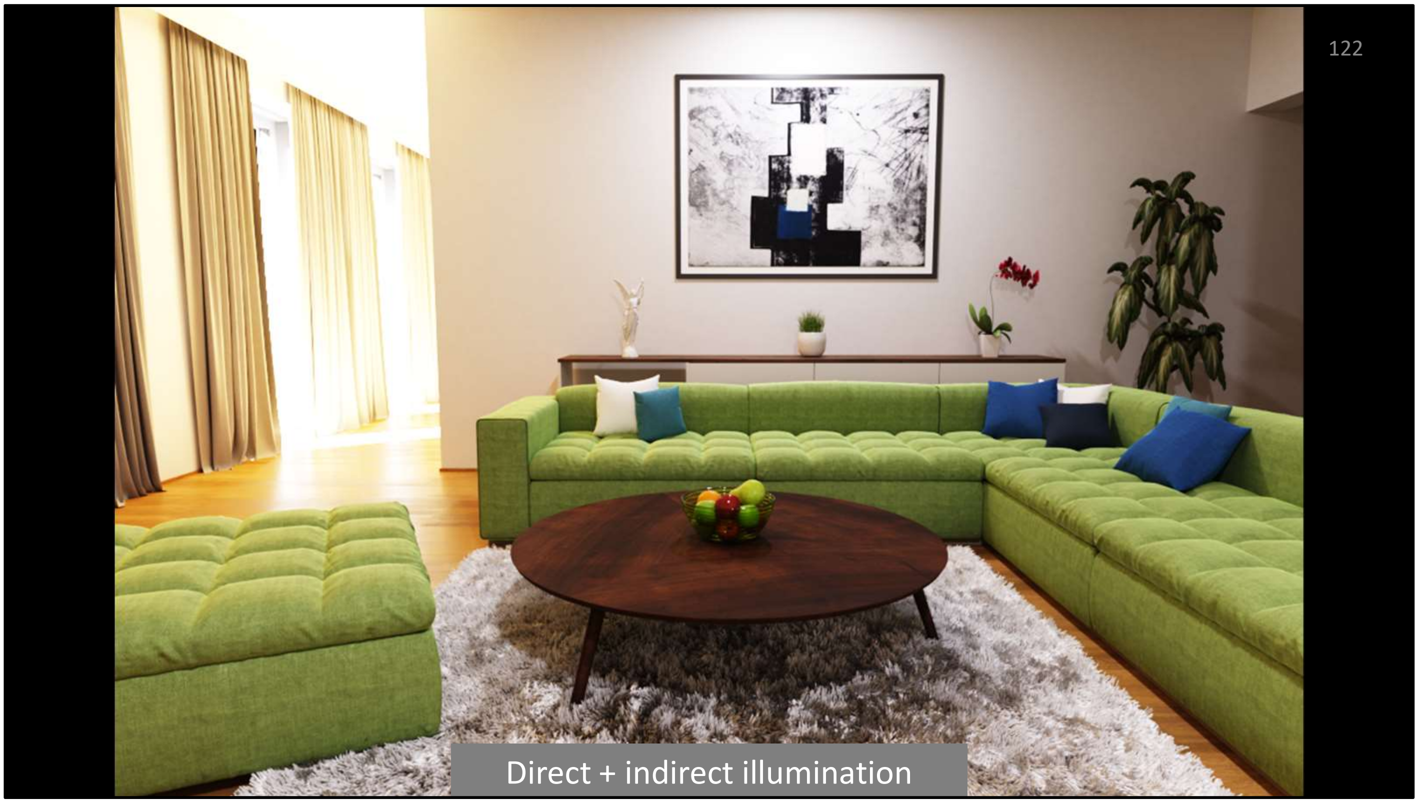


- Let us now demonstrate our solution in practice.

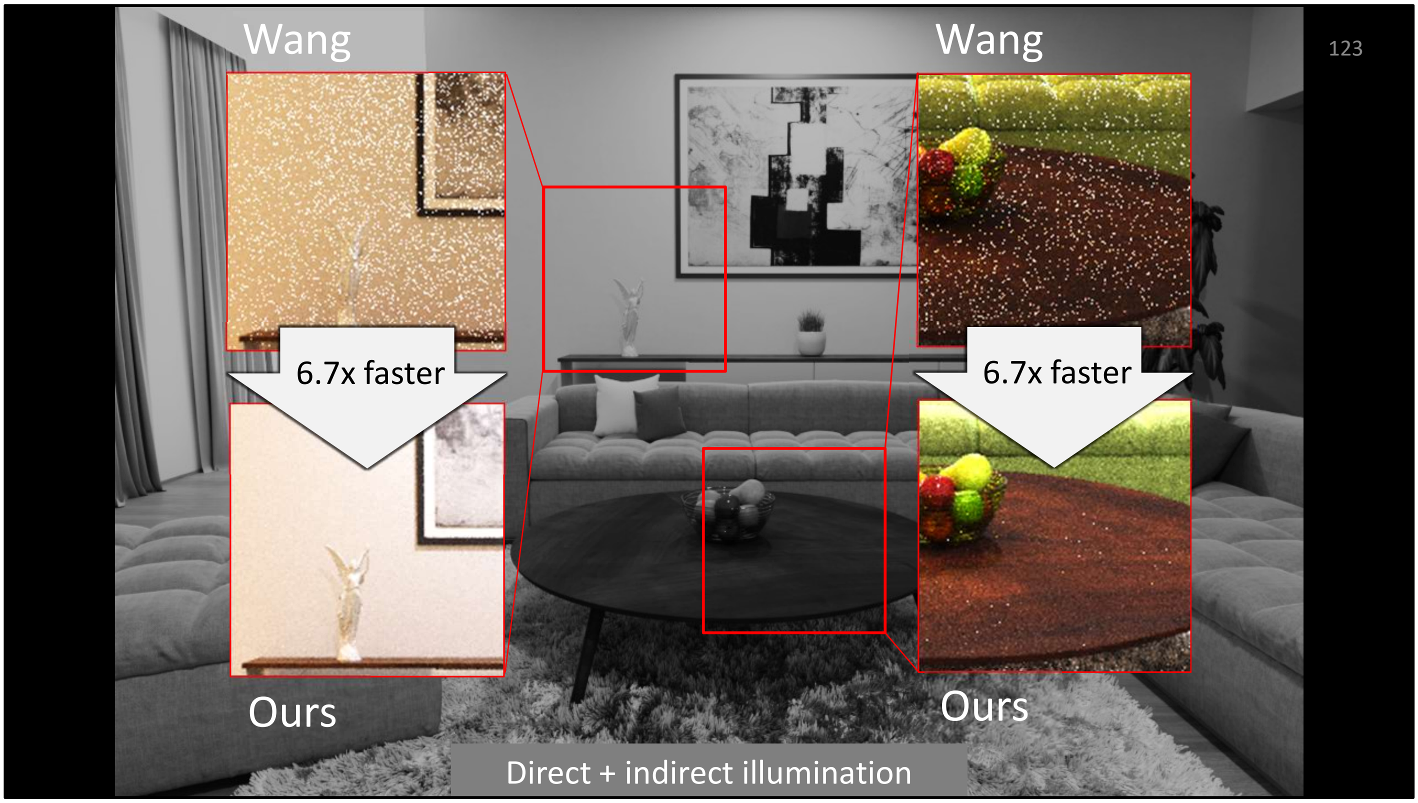

- Let us start with performance testing in a scene with simple occlusion in direct illumination only setting.
- It is the living room scene from the beginning of our presentation.
- It is lit mostly by a few small area lights on the ceiling, only in the left part sunlight is coming through the windows.



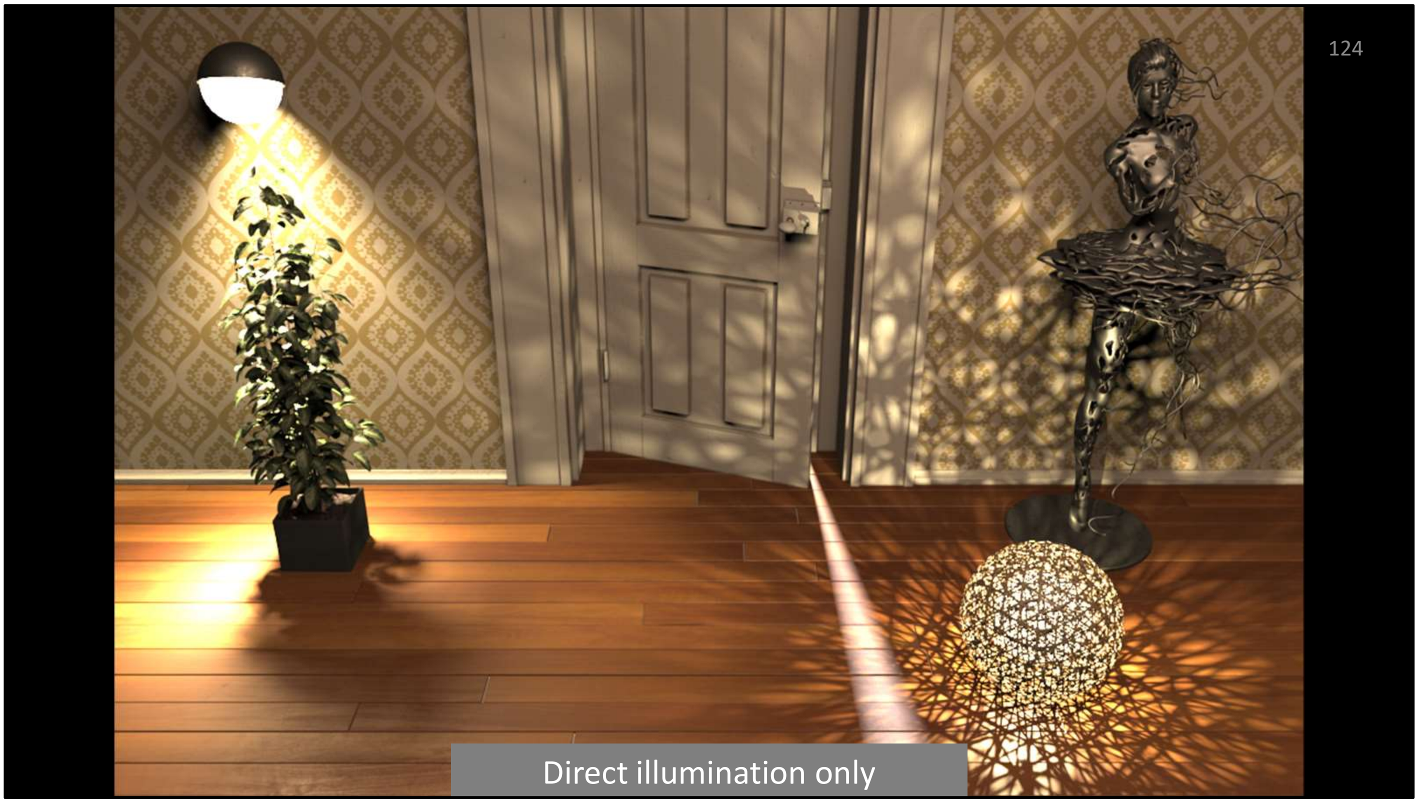

- As you could already see, non-adaptive sampling of Wang at al. does not perform well in this scene.
- The sun is much stronger than the ceiling lights and is therefore sampled much more often even if it is actually occluded and so most of the samples are wasted.
- <click>
- Donikan’s algorithm improves the result significantly, as it quickly learns the sun occlusion.
- On the other hand, it struggles with the ceiling lights. They are covered by shades which block some of the samples. The method gives such samples too big weights, undersamples these lights and introduces spiky noise.
- <click>
- Our method also quickly learns the sun occlusion…
- <click>
- … and converges more than 500× faster than Wang.
- <click>
- We can even observe higher empirical convergence rate.
- At the same time, thanks to the Bayesian treatment, our method is robust, does not get confused by the occluded samples and avoids the spiky noise.



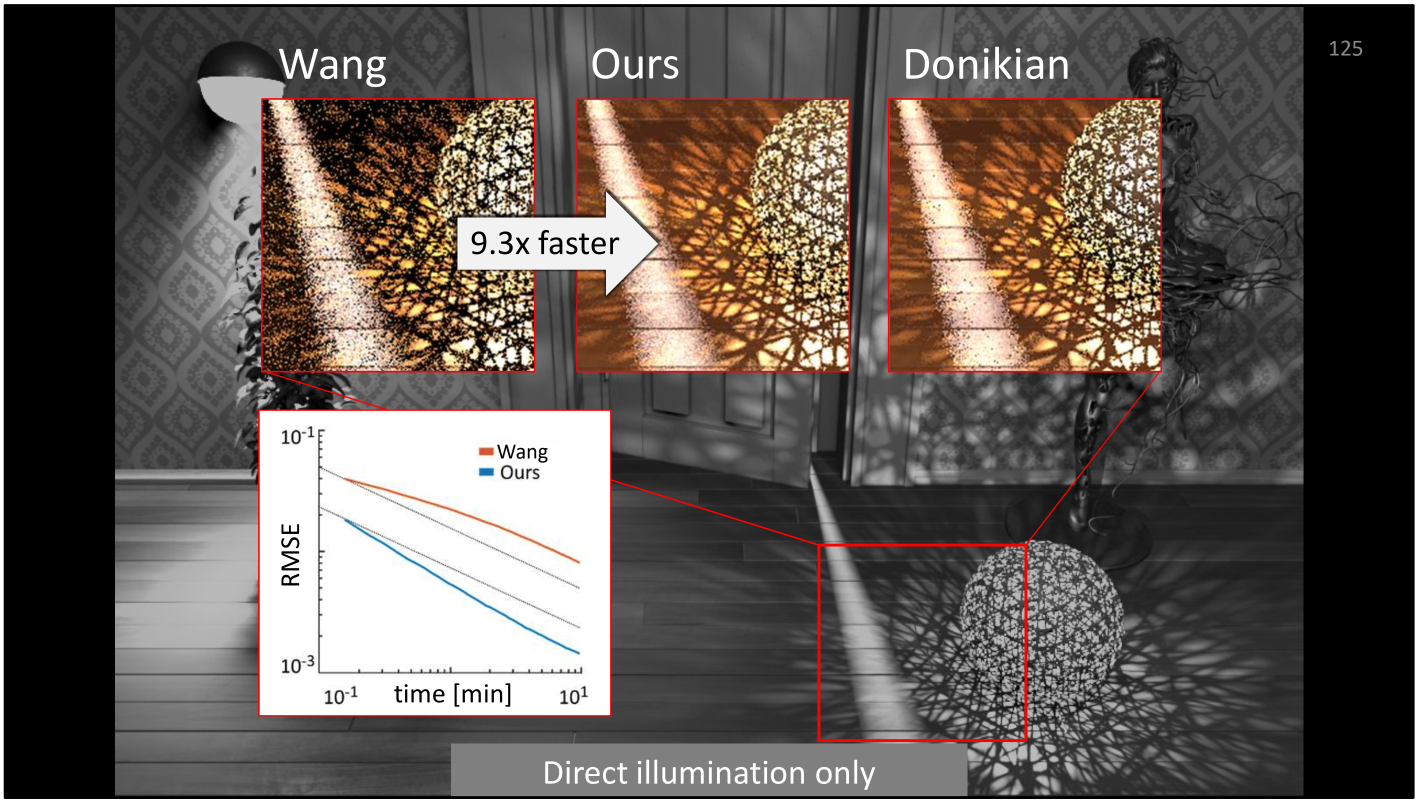

- So, that was the direct illumination.
- However, in practice one is usually more interested in images containing both direct and indirect components.



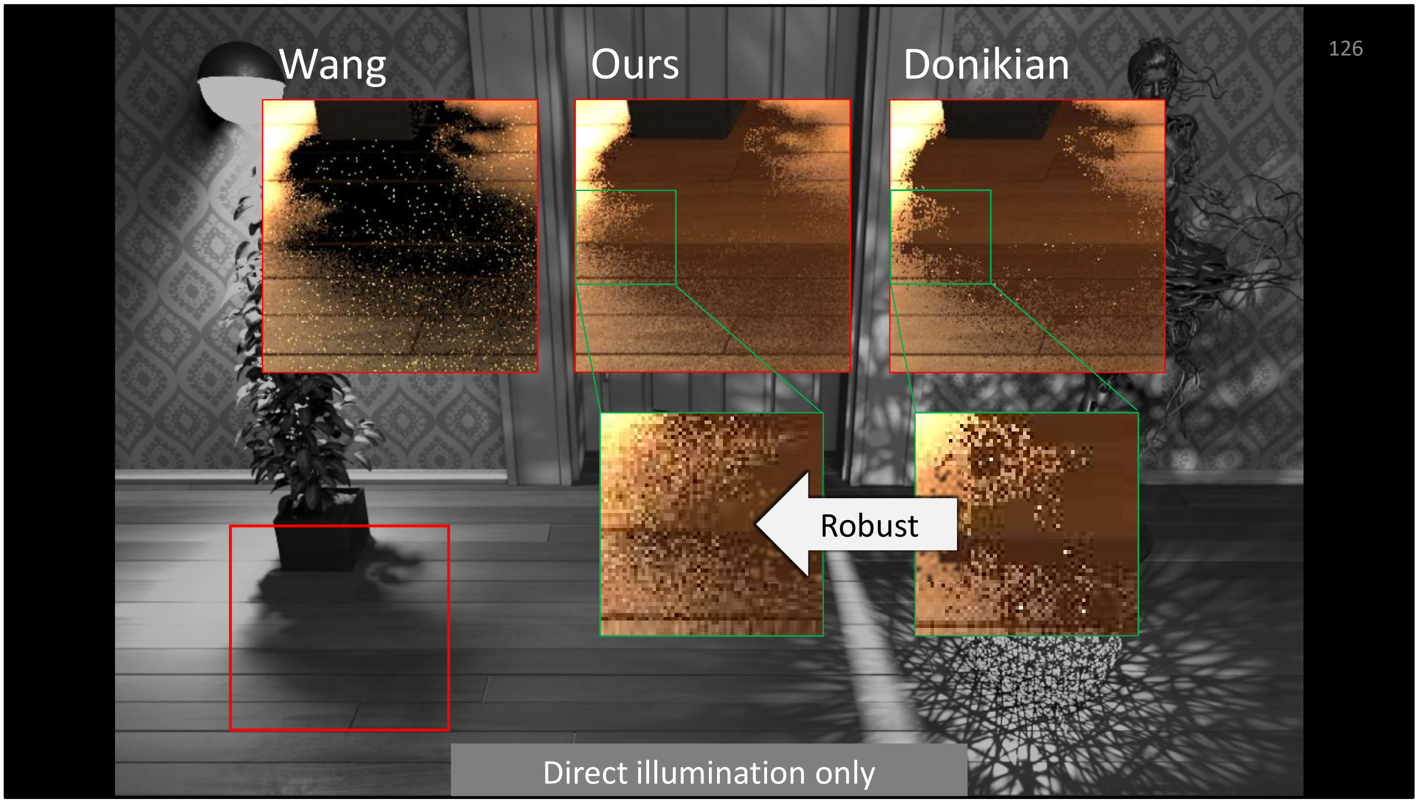

- We can see that the strong direct illumination noise of Wang dominates also in the complete image.
- The direct component is definitely the main source of noise in this scene.
- <click>
- By using our method in the next event estimation in path tracing we are able to improve the light sampling on every path vertex…
- <click>
- …and get more than 6x times speedup.
- Note that the remaining noise in the bottom right image is caused solely by the indirect component and cannot be influenced by our method.




- Now it is time to stress test the algorithm’s robustness robustness in a scene with complex occlusion.
- This scene presents a real challenge due to its highly structured illumination plus there are lights in the other room behind the door.

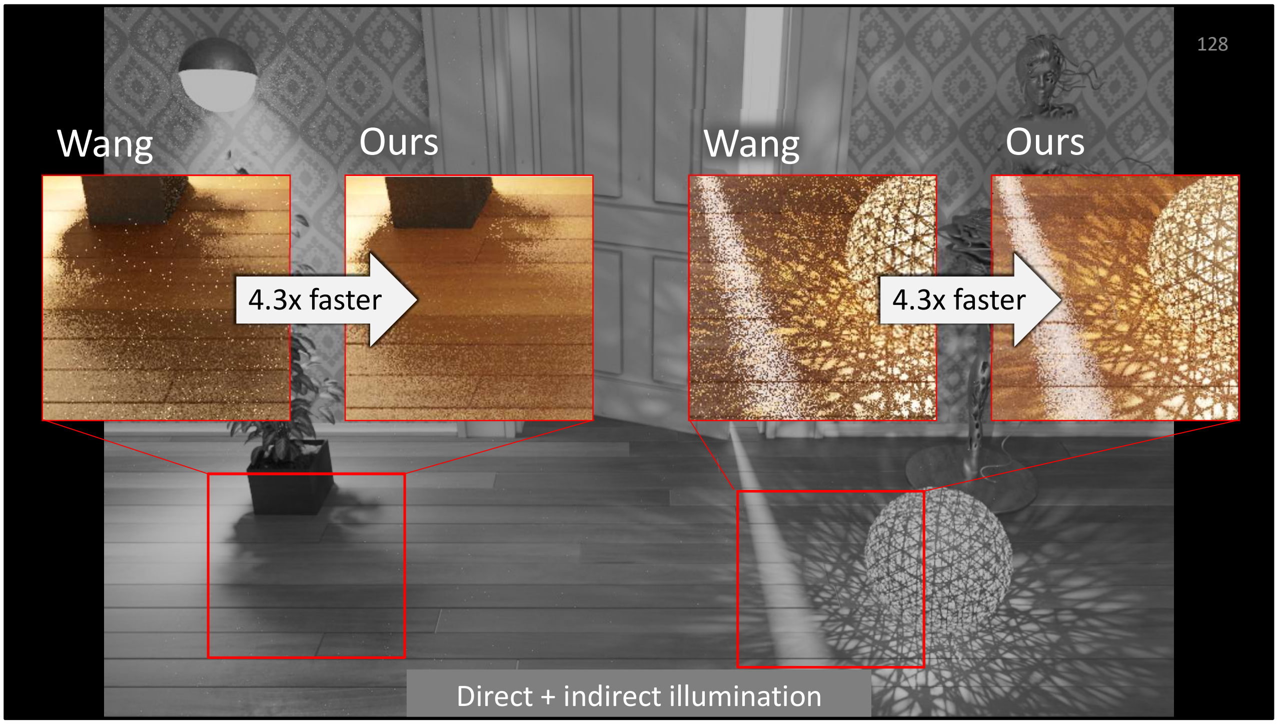



- In this part…
- <click>
- … Wang's method produces a lot of noise again as it wastes samples on the lights behind the door.
- <click>
- On the other hand, our method performs great.
- <click>
- It is more than 9 times faster,
- <click>
- And again we can observe higher empirical convergence rate. And all that without introducing any artifacts in such a complicated illumination setting.
- <click>
- Donikian's method at first also seem to perform well but further inspection would discover small blocky artifacts in the shadows.

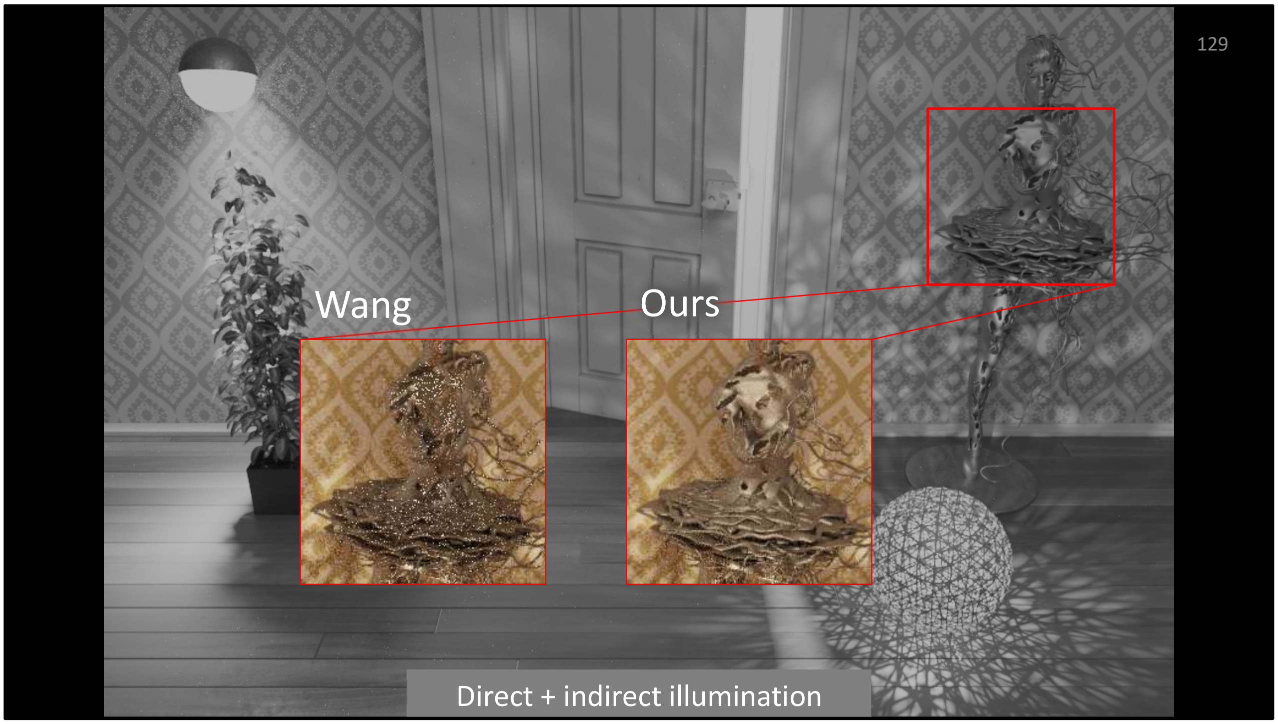



- However, in this part …
- <click>
- Wang does not perform well again, but this time also the Donikian’s method fails.
- <click>
- The illumination coming through the leaves of the plant is too complex for the ad-hoc learning to handle, the method overfits and produces square artifacts. This is exactly the problem of previous adaptive methods. They can provide substantial speedup but they don’t fail gracefully.
- <click>
- Bayesian learning makes our method much more robust and artifact-free.


- Finally, let’s test the complex occlusion also with the indirect component.
- If we take a look at the same scene…

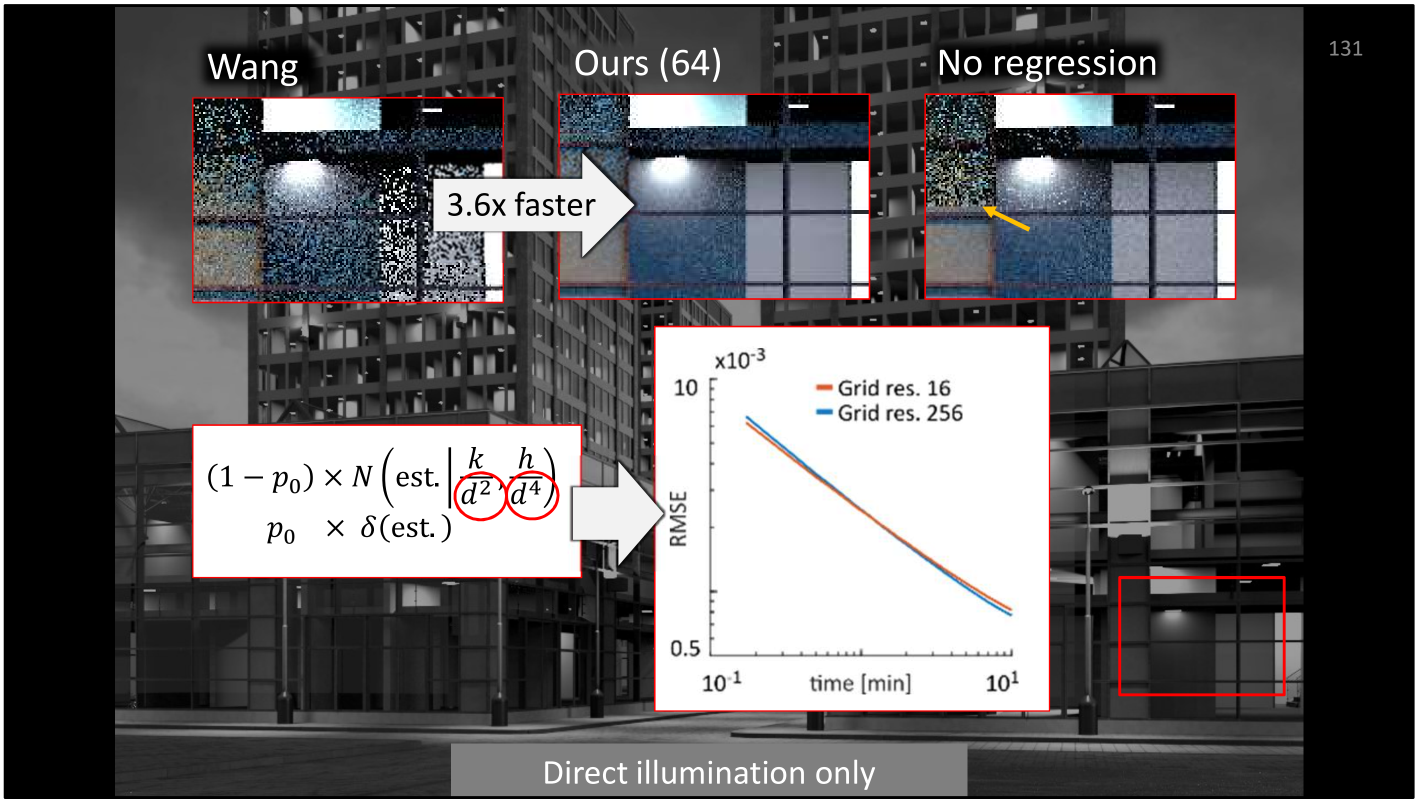



- …we will see the direct illumination noise dominates the complete image the same way as in the previous scene.
- <click>
- Our method eliminates it
- <click>
- and renders the complete image more than 4 times faster and without any artifacts.




- There is one more interesting place in this scene.
- <click>
- It is this statue.
- It is made of glossy metal and even though our method does not take the surface BRDF into account, it performs significantly better even there.


- Since we divide a scene into regions by a uniform grid of a fixed resolution, we need to test how this resolution affects the performance.
- For that we will use this large scene containing a lot of lights.




- With our default choice of 64 regions per the shortest grid dimension
- <click>
- Our method performs more than 3 times faster than Wang.
- So what about other resolution?
- <click>
- The regression modeling of the distance falloff makes our method rather insensitive to the actual grid resolution.
- <click>
- And so even much smaller as well as much higher resolutions all perform roughly the same.
- <click>
- Without the regression we would have to use much higher resolution otherwise we would see sudden noise transitions between regions.

# Contribution

- **Bayesian framework for robust adaptivity**
- Optimal cluster sampling
- Algorithm for direct illumination
  - Unbiased, adaptive, robust
  - Easy to integrate into a path tracer





- The main contribution of our work is creating a **Bayesian framework for adaptive Monte Carlo quadrature**.
- It enables exploiting the big potential of the adaptive approach while avoiding the biggest weakness of previous attempts – the lack of robustness.
- We applied this framework on the problem of direct illumination sampling.
- In the process we derived the optimal sampling of clusters and developed an unbiased adaptive direct illumination algorithm with online learning of light sampling distributions. It is easily integrable into a path tracer and suitable for interactive rendering.
- Our **new framework is not limited to the direct illumination** though and we are certain that other applications of adaptive sampling will benefit from it as well and **it opens the path for many other tools of statistical machine learning (such as full Bayes or variational Bayes)**.

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

# CONCLUSION

# CONCLUSION

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- **Path guiding**
  - Makes complex bidirectional method unnecessary
  - Potential for wide adoption practice

- Machine learning methods = principled way to achieve **robust, online adaptive sampling**
  - Path guiding – **online learning of parametric mixture models**
  - Direct illumination sampling – **Bayesian online regression**

- **Online learning methods** compatible interactive rendering workflows & progressivity
- **Bayesian methodology** can provide the necessary **robustness**

Computer Graphics Charles University

corona

# FUTURE WORK

GENERATIONS / VANCOUVER 12-16 AUGUST
SIGGRAPH2018

- Bayesian model selection
- Full Bayesian inference – Variational Bayes?
- Adaptive decision based on reinforcement learning
- Deep learning for light field reconstruction for path guiding
- Can this be that one missing piece to make MCMC methods useful in practice?

Computer Graphics Charles University

corona

# THANK YOU!


- **Acknowledgments**
  - Colleagues and students from Charles university & Corona
  - Funding: Czech Science Foundation (16-18964S), Charles University Grant Agency project GAUK 1172416, by the grant SVV-2017-260452


- While you may think that rendering is science, remember that first and foremost, **rendering is magic**.


Computer Graphics Charles University

corona