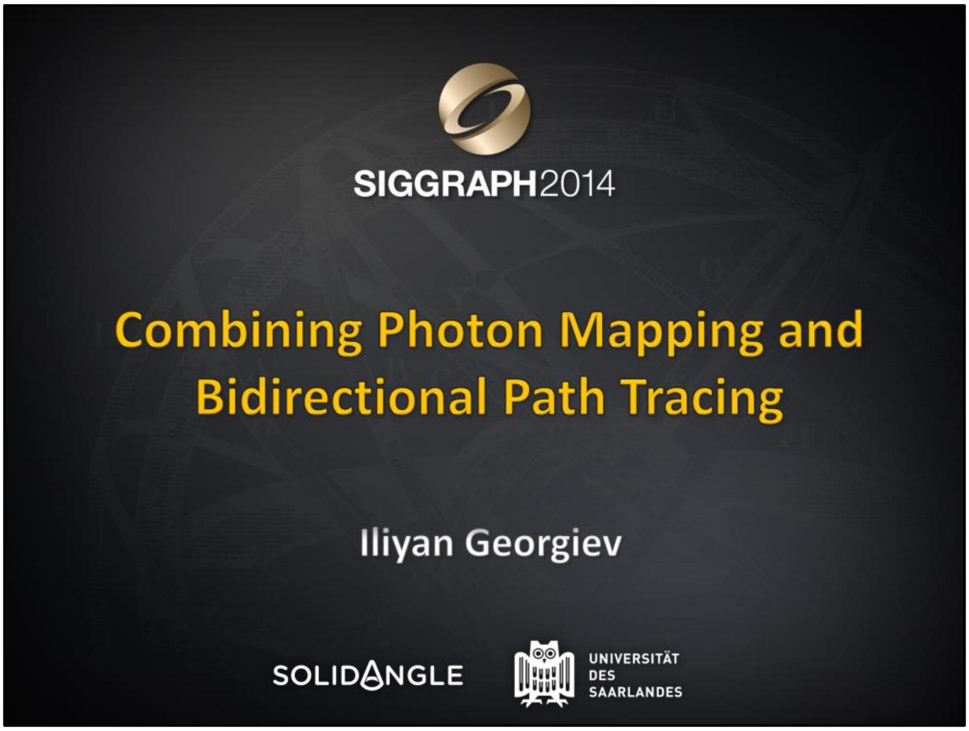
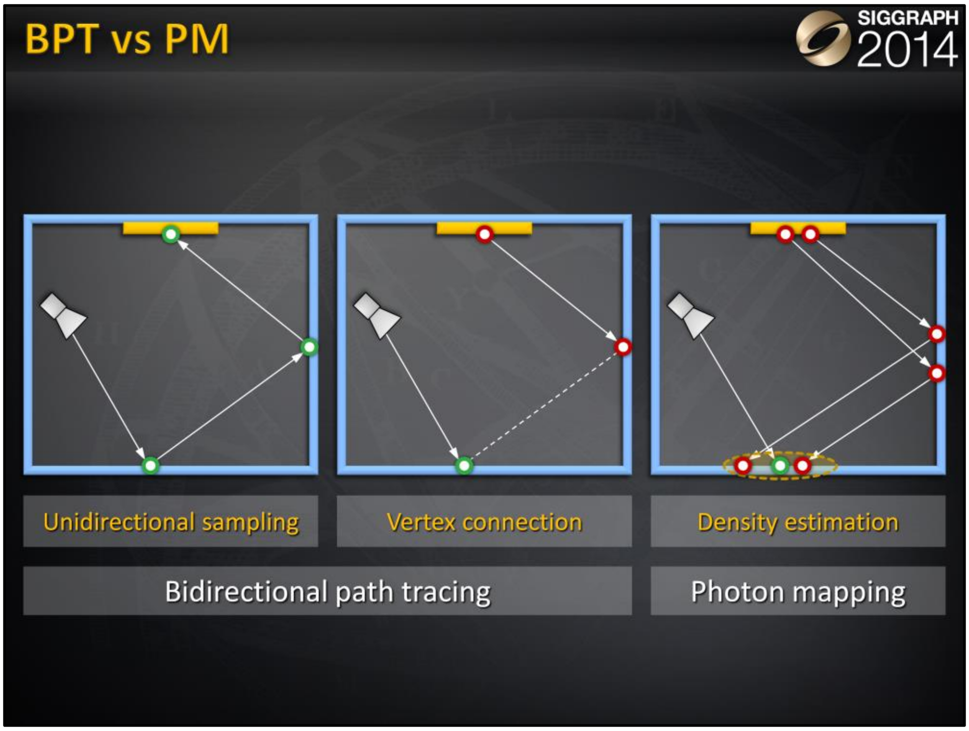


In the previous talk, Jaroslav discussed the path integral formulation of light transport and demonstrated its conceptual simplicity and flexibility. I will now show how we can leverage this framework to seamlessly combine photon mapping and bidirectional path tracing via multiple importance sampling.

Bidirectional path tracing is one of the most versatile light transport simulation algorithms available today. It can robustly handle a wide range of illumination and scene configurations, but is notoriously inefficient for specular-diffuse-specular light interactions, which occur e.g. when a caustic is seen through a reflection/refraction.

On the other hand, photon mapping (PM) is well known for its efficient handling of caustics. Recently, Hachisuka and Jensen [2009] showed a progressive variant of PM that converges to the correct result with a fixed memory footprint. Their stochastic progressive photon mapping (PPM) algorithm captures the reflected caustics in our scene quite well. However, it has hard time handling the strong distant indirect illumination coming from the part of the scene behind the camera.

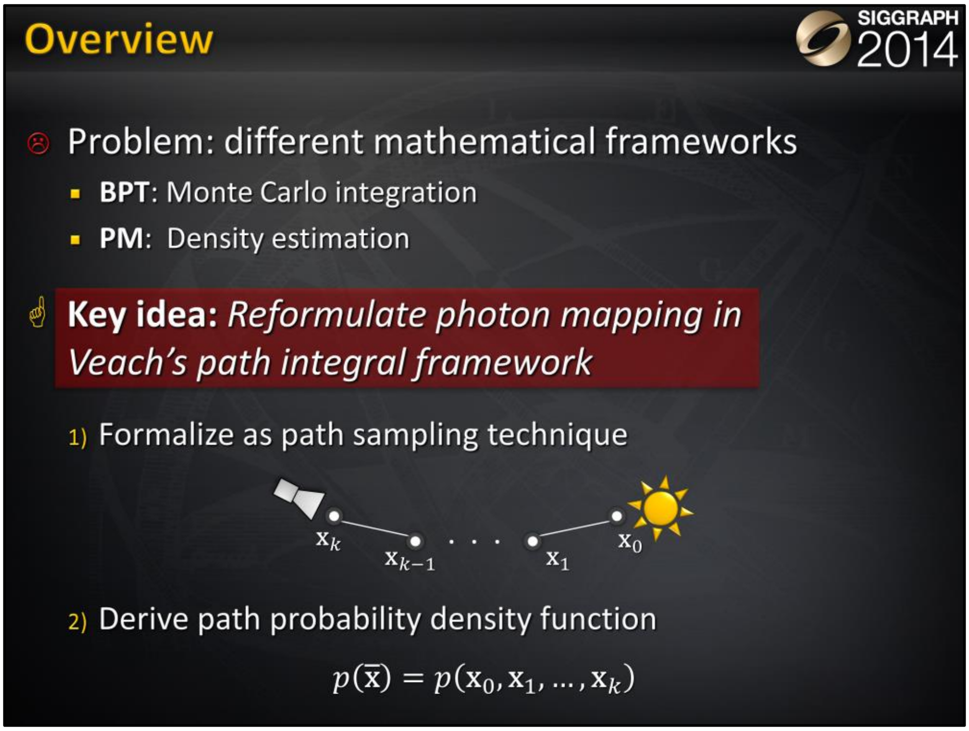
By using multiple importance sampling to combine estimators from bidirectional path tracing and photon mapping, the algorithm I will talk about today automatically finds a good mixture of techniques for each individual light transport path, and produces a clean image in the same amount of time.



Let us start by reviewing the techniques BPT and PM use to construct light transport paths connecting the eye and the light sources.

The BPT techniques can be roughly categorized *to unidirectional sampling* (US) and *vertex connection* (VC). US samples a path by starting either from a light source or the eye, and performs a random walk until termination. On the other hand, VC traces one sub-path from the eye and another sub-path from a light source, and then connects their endpoints.

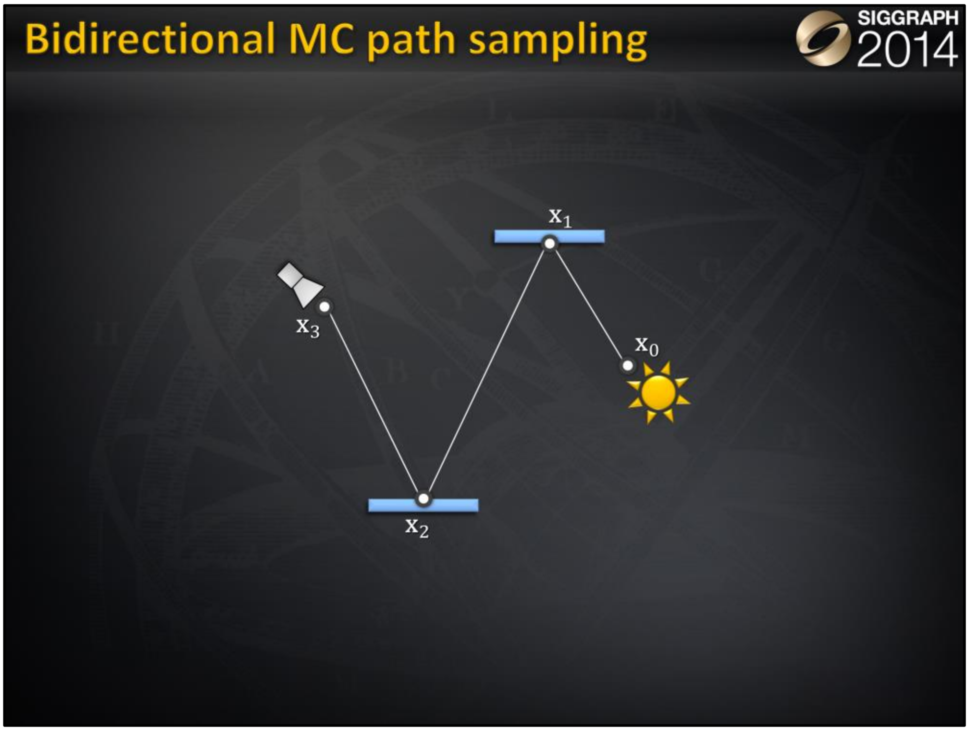
In contrast, photon mapping first traces a number of light sub-paths and stores their hit-points (a.k.a. photons). It then traces sub-paths from the eye and uses photon density estimation to compute the outgoing radiance at the hit-points.



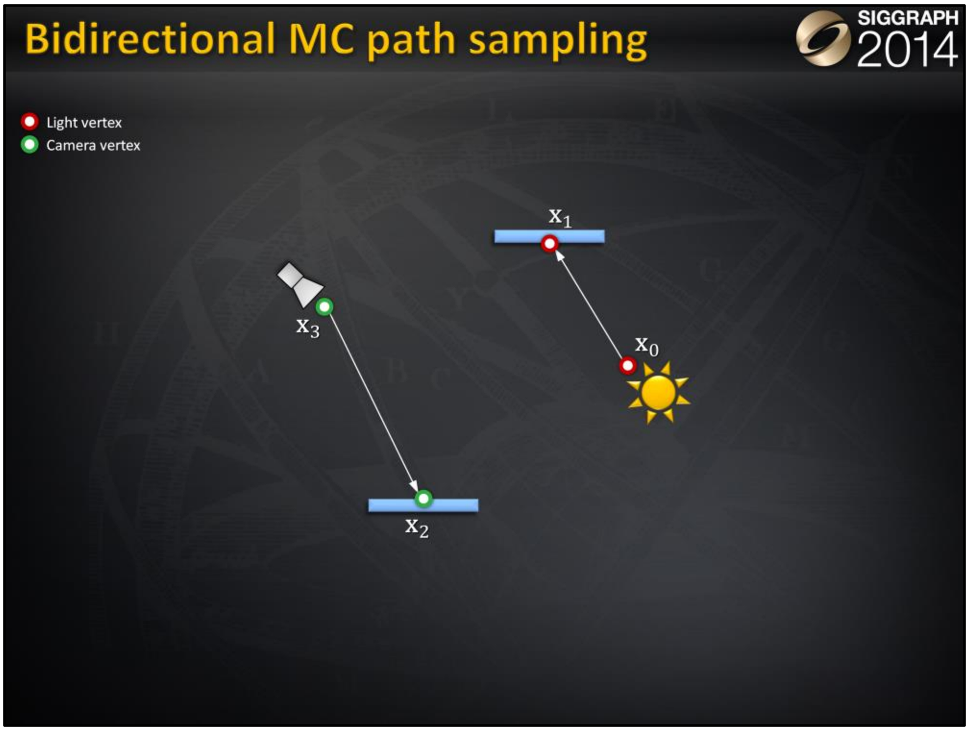
It has been long recognized that bidirectional path tracing (BPT) and photon mapping (PM) complement each other in terms of the light transport effects they can efficiently handle. However, even though both methods have been published more than 15 years ago, neither a rigorous analysis of their relative performance nor an efficient combination had been shown until very recently. The reason for this is that BPT and PM have originally been defined in different theoretical frameworks – BPT as a standard Monte Carlo estimator to the path integral, and PM as an outgoing radiance estimator based on photon density estimation.

The first step toward combining these two methods is to put them in the same mathematical framework. We choose Veach's path integral formulation of light transport, as it has a number of good properties (which Jaroslav discussed) and also because BPT is already naturally defined in this framework.

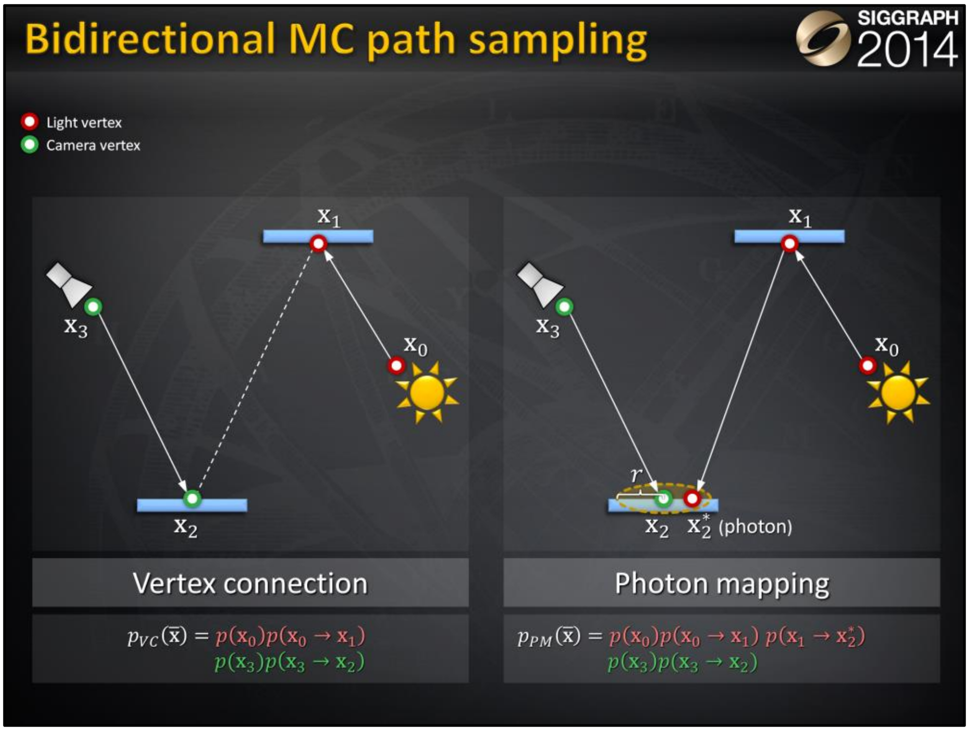
We need two key ingredients: (1) express PM as a sampling technique that constructs light transport paths that connect the light sources to the camera, and (2) derive the probability density function for paths sampled with this technique. This will give us a basis for reasoning about the relative efficiency of BPT and PM. And more importantly, it will lay the ground for combining their corresponding estimators via multiple importance sampling.



Let us start by taking a simple length-3 path and see how it can be constructed bidirectionally.



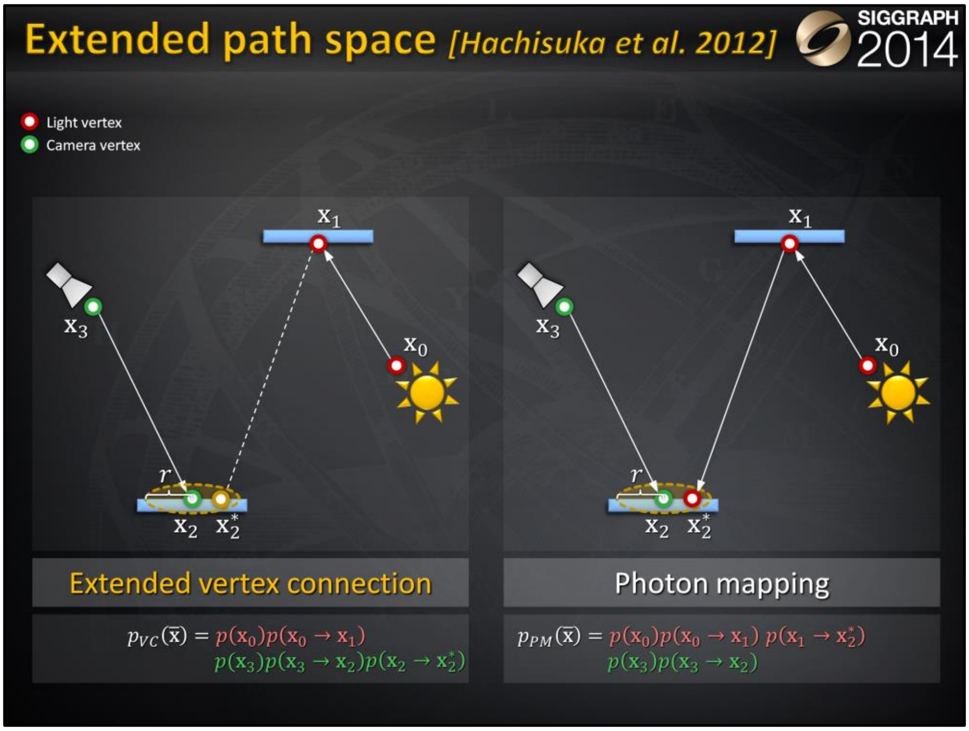
We first trace one subpath from the camera and another one from a light source.



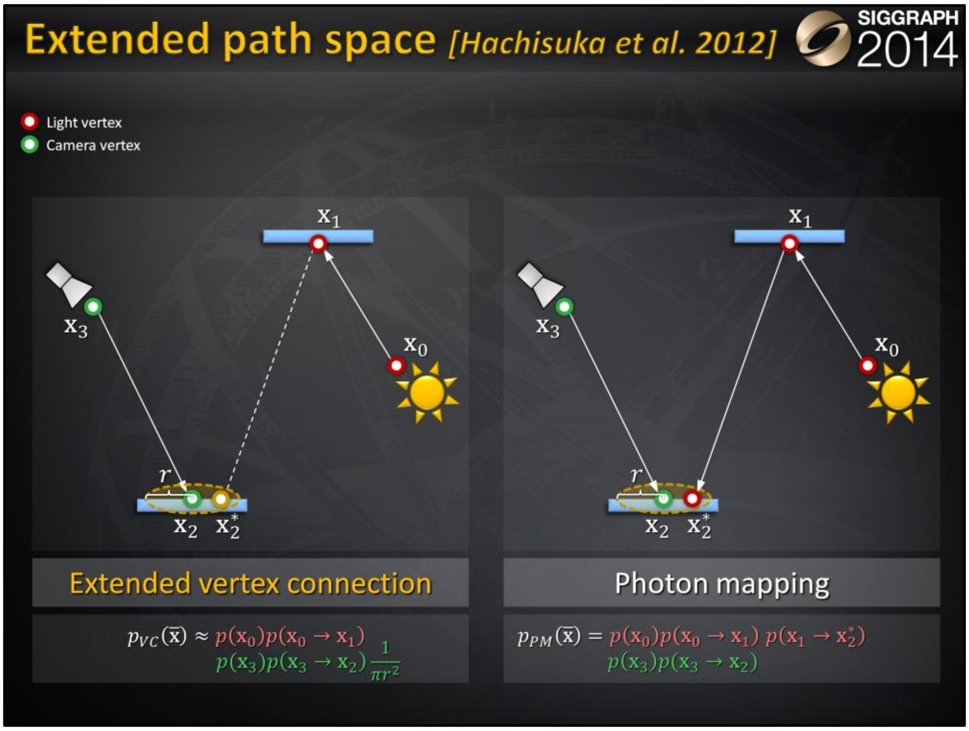
Now let's see how we complete a full path in BPT and PM.

Bidirectional path tracing (BPT) connects the subpath endpoints deterministically. We call this technique *vertex connection*. The PDF of the resulting full path is simply the product of the PDFs of two independently sampled subpaths.

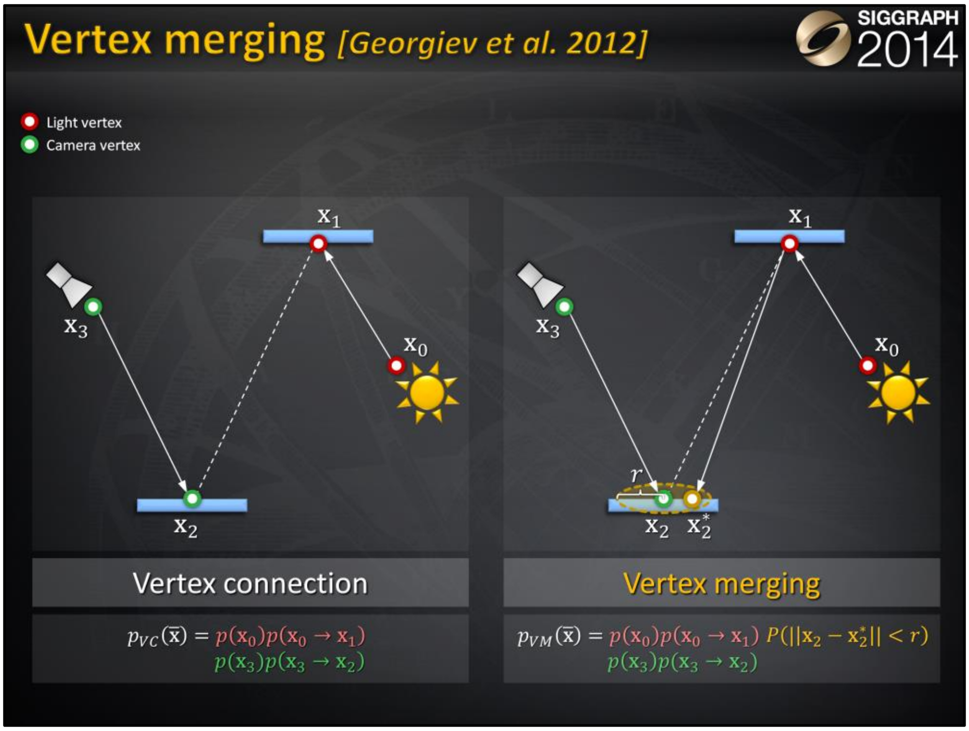
Photon mapping (PM), on the other hand, extends the light subpath by sampling one more vertex from $\mathbf{x}_1$, and makes a contribution if the photon hit-point $\mathbf{x}_2^*$ lies within a distance r from $\mathbf{x}_2$. And we can derive a similar path PDF for PM. However, we see that the two methods sample paths with a different number of vertices, and consequently their PDFs have different units. Plugging these PDF into MIS wouldn't produce a meaningful result, because the heuristics expect all PDFs to be expressed w.r.t. the same measure. To obtain a meaningful MIS combination, we need to express these two PDFs w.r.t. the same measure.



Hachisuka et al. [2012] express the vertex connection PDF in the higher-dimensional space of photon mapping by considering the sampling of vertex $\mathbf{x}_2^*$ via a random perturbation of the eye vertex $\mathbf{x}_2$ within an r-neighborhood.



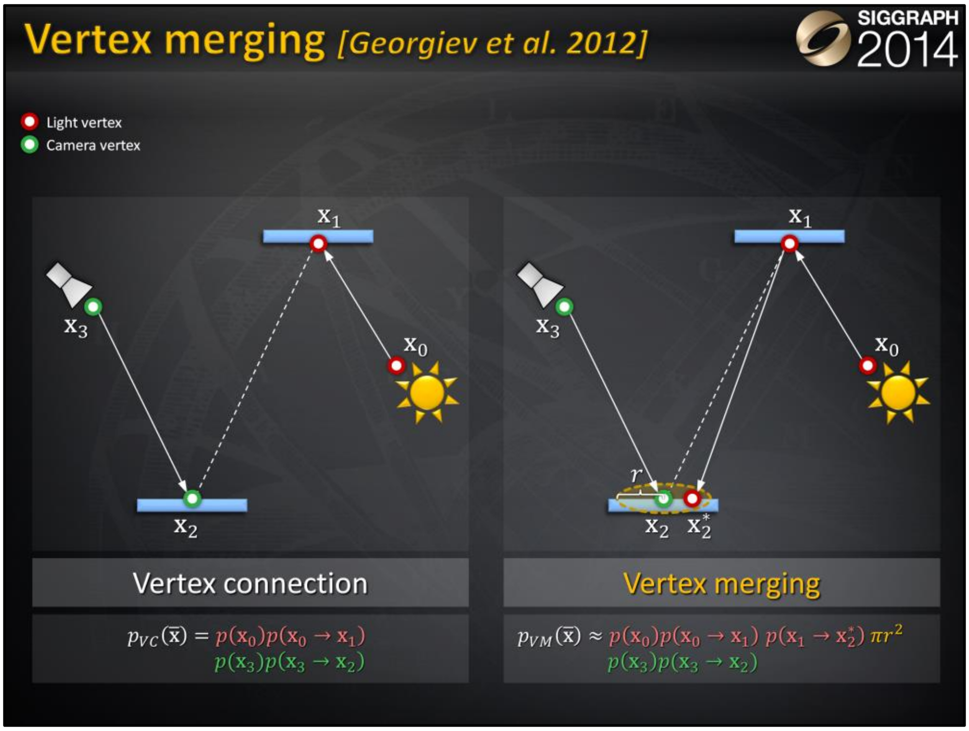
Assuming that the surface in this neighborhood is locally flat, i.e. that the region is a disk, the PDF of $\mathbf{x}_2^*$ is $^1/_{\pi r^2}$.



Alternatively, we can keep the vertex connection PDF in its original form, and express the PDF of photon mapping in the lower-dimensional space of BPT.

To do this, we can interpret the sampling process as establishing a regular vertex connection between $\mathbf{x}_1$ and $\mathbf{x}_2$, but conditioning its acceptance on the random event that a vertex $\mathbf{x}_2^*$ sampled from $\mathbf{x}_1$ lands within a distance r to $\mathbf{x}_2$. This probabilistic acceptance is simply a Russian roulette decision.

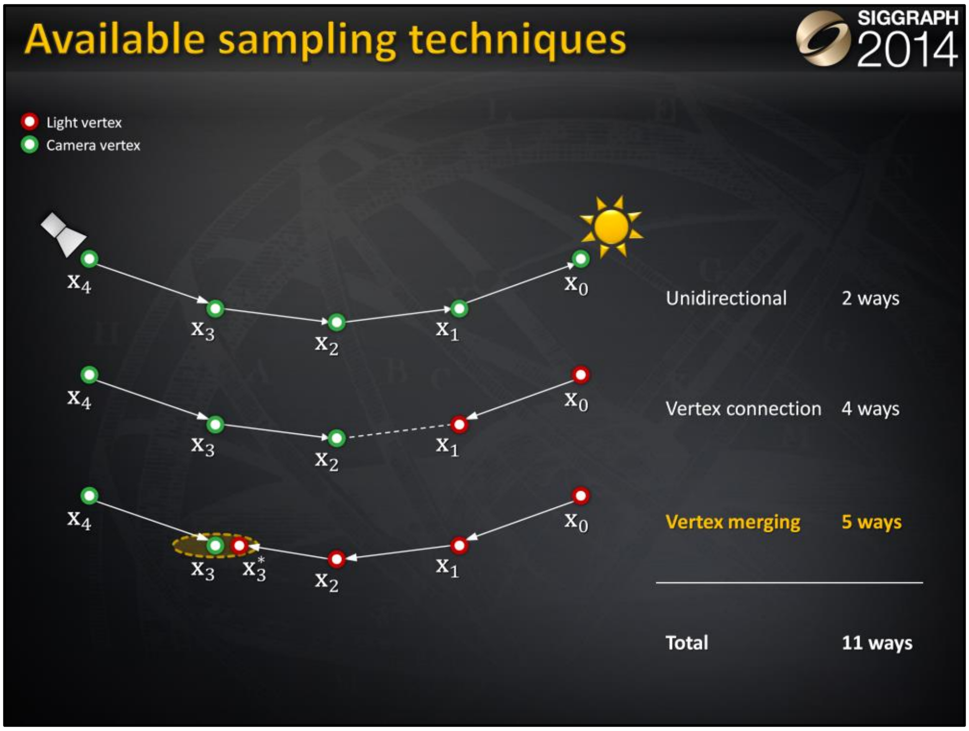
The full path PDF is then the product of the subpath PDFs as on the left, but in addition multiplied by the probability of sampling the point $\mathbf{x}_2^*$ within a distance r of $\mathbf{x}_2$. This acceptance probability is equal to the integral of the PDF of $\mathbf{x}_2^*$ over the r-neighborhood of $\mathbf{x}_1$.



Again, assuming that this neighborhood is a disk, and also that the density of $\mathbf{x}_2^*$ is constant inside this disc, the integral can be approximated by the PDF of the actually sampled point $\mathbf{x}_2^*$, multiplied by the disc area πr^2.

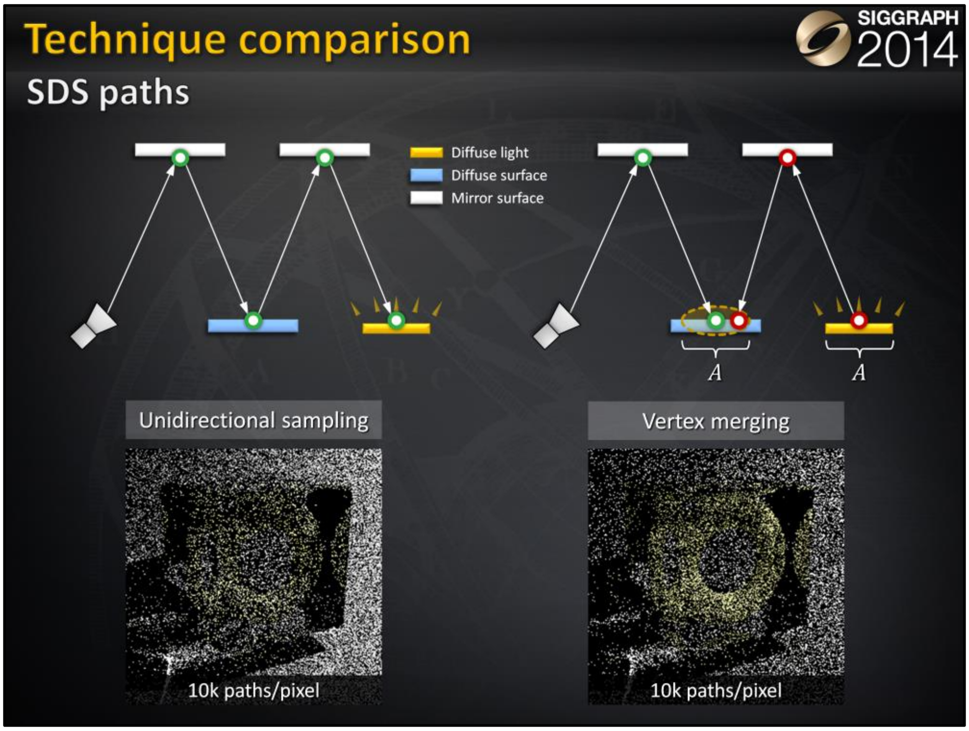
We label this technique *vertex merging*, as it can be intuitively thought to weld the endpoints of the two subpaths if they lie close to each other.

Note that while in the interpretation of Hachisuka et al. we had πr^2 in the vertex connection PDF denominator, it appears in the nominator of the VM interpretation. In the remainder of the discussion I will use the vertex merging interpretation, but the final combined algorithm I will present is identical with both interpretations.



Now that we have formulated the vertex merging path sampling technique, we can put it side by side with the already available techniques in BPT. There are two ways to sample a length-4 path unidirectionally, and four ways to sample it via vertex connection. Vertex merging adds five new ways to sample the path, corresponding to merging at the five individual path vertices. In practice, we can avoid merging at the light source and the camera, as directly evaluating emission and sensitivity is usually cheap.

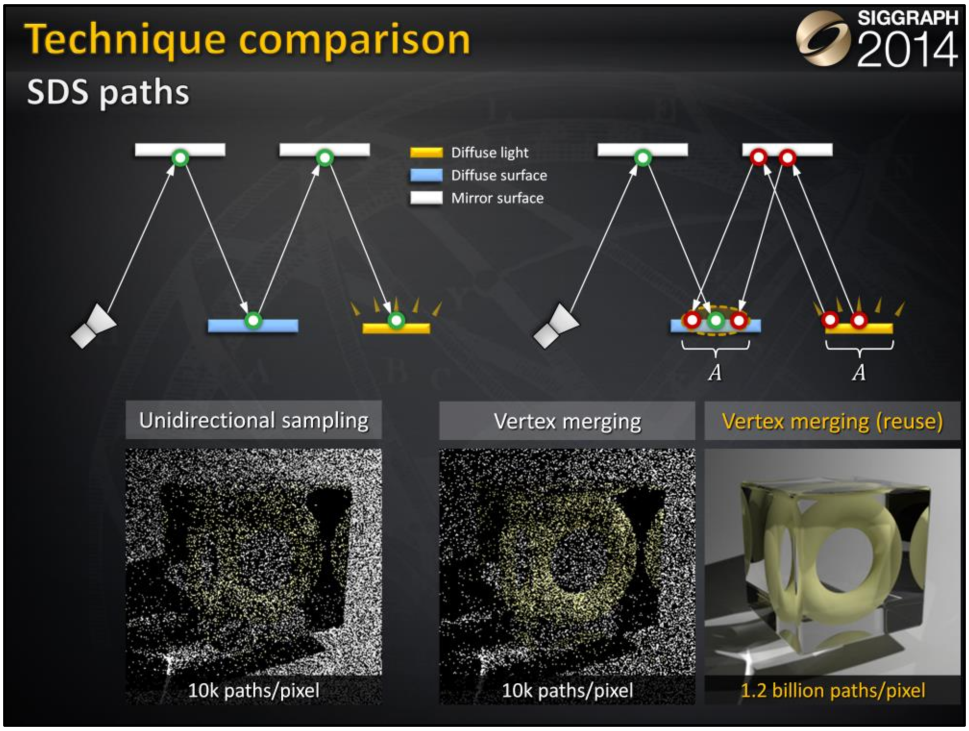
But with so many ways to sample the same light transport path, a question naturally arises: which technique is the most efficient for what types of paths?



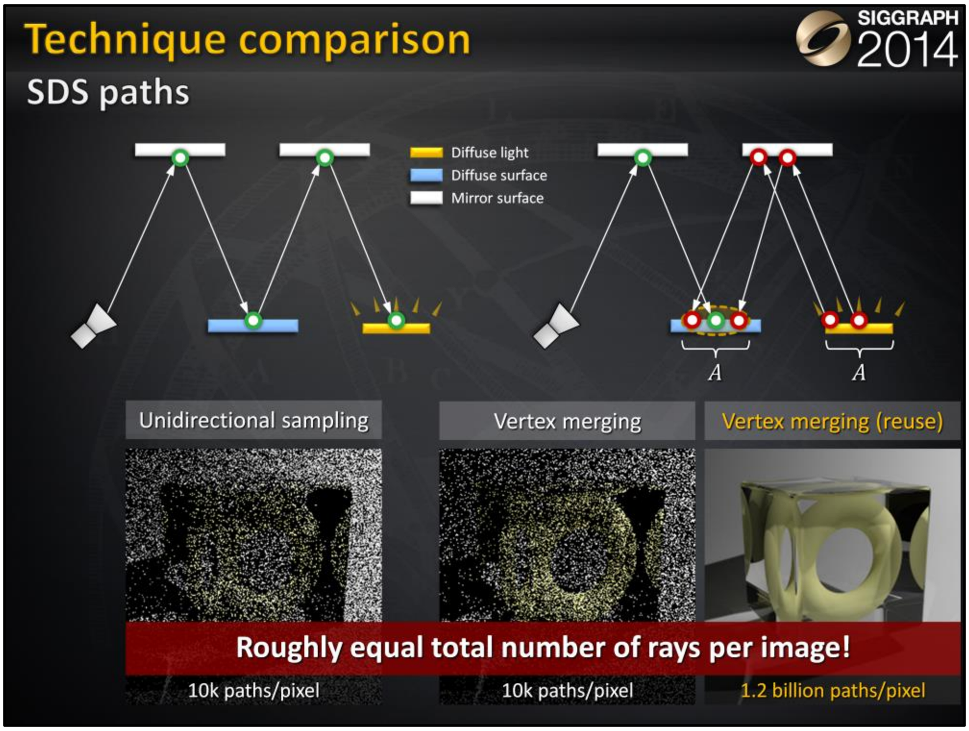
To answer this question, let us first take a look at specular-diffuse-specular (SDS) paths. Here, bidirectional path tracing can only rely on unidirectional sampling: it traces a path from the camera hoping to randomly hit the light source. With vertex merging, we can trace one light and one camera subpath, and merge their endpoints on the diffuse surface.

It can be shown that if the light source and the merging disk have the same area A, then unidirectional sampling and vertex merging sample paths with roughly the same probability density. This means that we should expect the two techniques to perform similarly in terms of rendering quality.

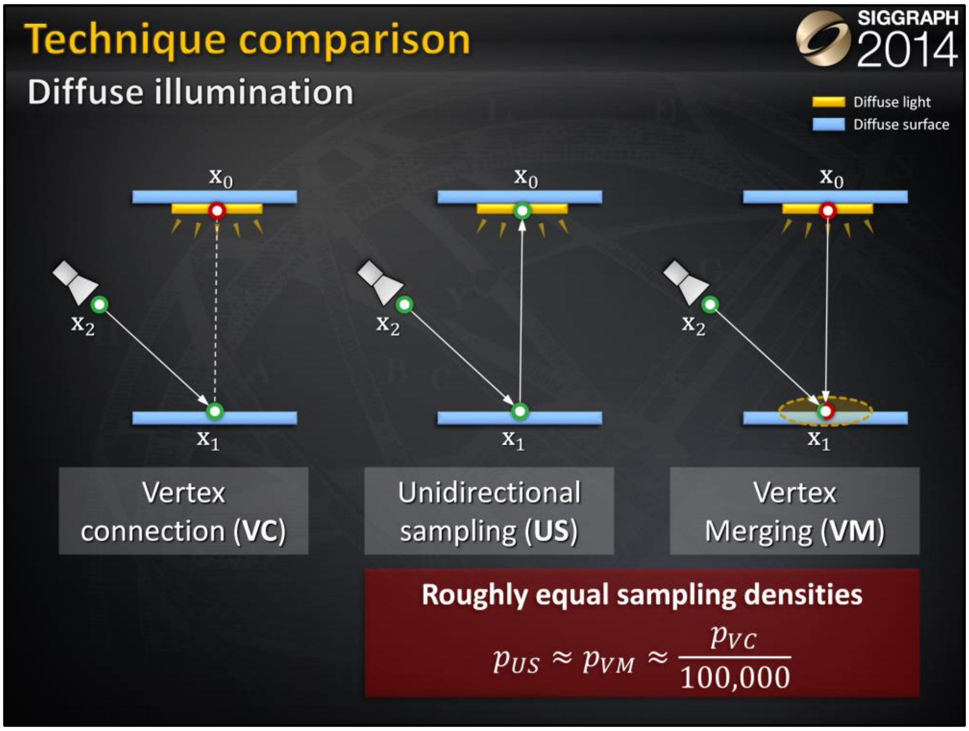
We render these two images progressively, sampling one full path per pixel per iteration. For the left image we trace paths from the camera until they hit the light. For image on the right, we trace subpaths from both ends, and merge their endpoints if they lie within a distance $r = \sqrt{A/\pi}$ from each other. Both images look equally noisy, even after sampling 10,000 paths per pixel. This confirms that vertex merging, and thus photon mapping, is *not* an intrinsically more robust sampling technique for SDS paths than unidirectional sampling.



However, the strength of vertex merging is computational efficiency – we can very efficiently reuse the light subpaths traced for *all* pixels at the cost of a single range search query. This allows us to quickly construct orders of magnitude more light transport estimators from the same sampling data, with a minimal computational overhead, resulting in a substantial quality improvement.



For all these three images we have traced roughly the same number of rays, and the only difference between the one in the center and the one on the right is that the for right image we have enabled path reuse, by storing, and looking up, the light subpath vertices in a photon map at every rendering iteration.

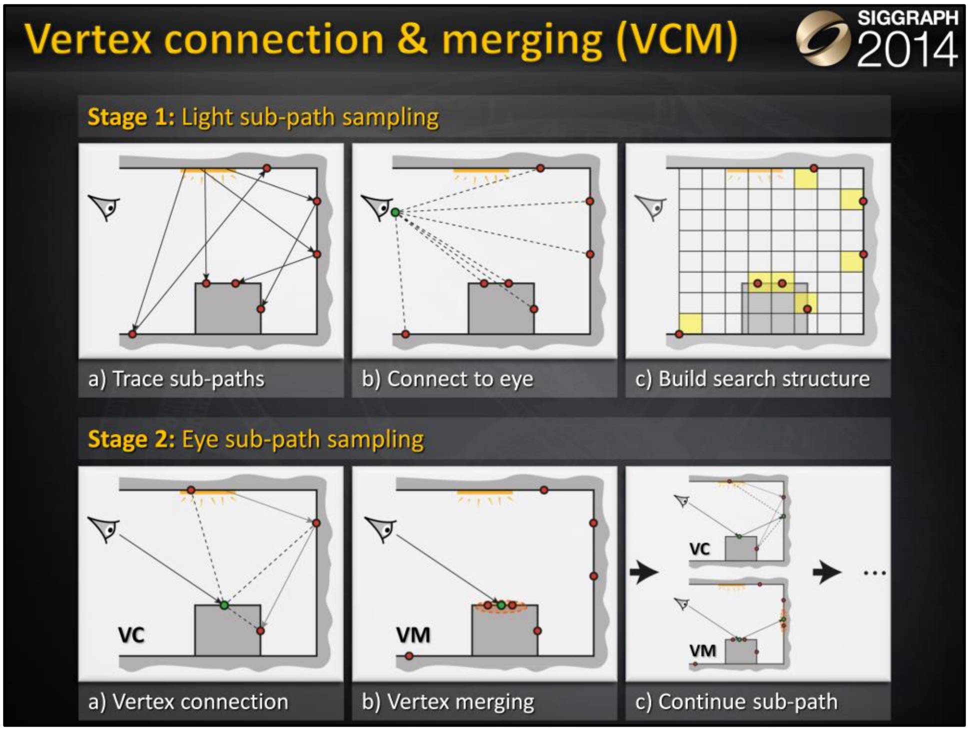


Now let's look at another extreme example – diffuse illumination. Note that vertex connection (VC) constructs the edge between $\mathbf{x}_1$ and $\mathbf{x}_2$ deterministically, while unidirectional sampling (US) and vertex merging (VM) both rely on random sampling.

Once again, it can be shown that if the light source and the merging disk have the same area, then US and VM sample this path with roughly the same probability density.

For the specific case shown on this slide, this density is about 100,000 lower than that of VC. This demonstrates that VM is not an intrinsically more robust sampling technique than VC either. This is not surprising – if we recall the expression for the VM path PDF, we see that it can only be lower than that of the corresponding VC technique, as their only difference is the probability factor in the VM PDF, which is necessarily in the range $[0; 1]$. Still, by reusing paths across pixels, vertex merging, and thus photon mapping, gains a lot of efficiency over unidirectional sampling.

All these useful insights emerge from the reformulation of photon mapping as a path sampling technique.



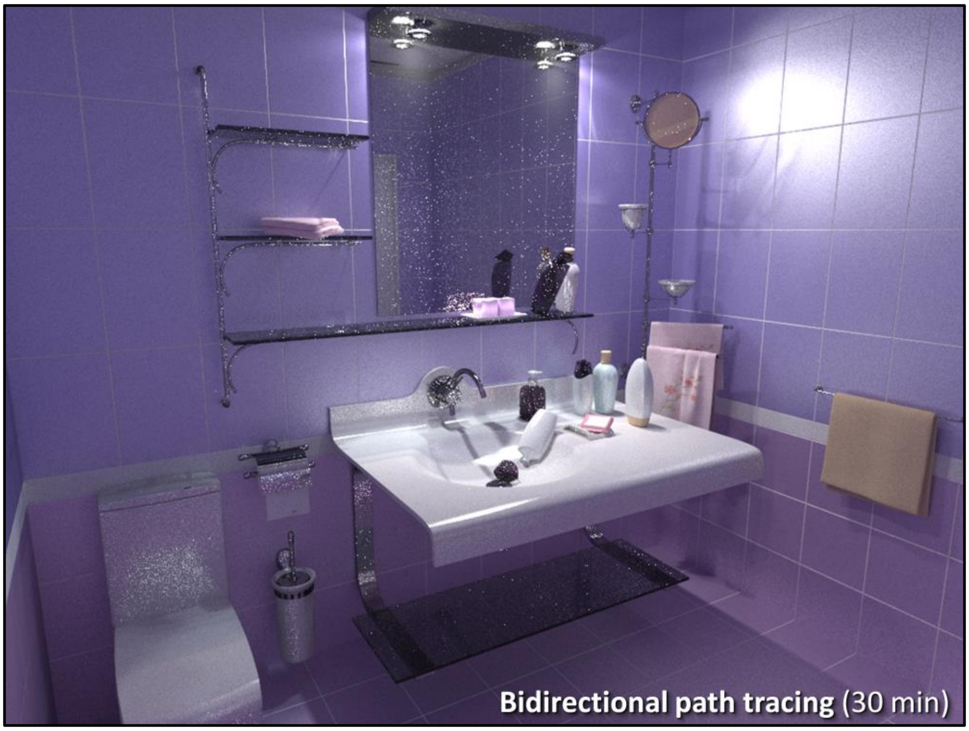
Even more usefully, we now have the necessary ingredients for combining photon mapping and bidirectional path tracing into one unified algorithm. The vertex merging path PDFs tell us how to weight all sampling techniques in multiple importance sampling, and the insights from the previous two slides command to strive for path reuse.

The combined algorithm, which we call *vertex connection and merging* (VCM), operates in two stages.

1. In the first stage, we
   a) trace the light subpaths for all pixels,
   b) connect them to the camera, and
   c) store them in a range search acceleration data structure (e.g. a kd-tree or a hashed grid).

2. In the second stage, we trace a camera subpath for every pixel.
   a) Each sampled vertex on this path is connected to a light source (a.k.a. next event estimation), connected to the vertices of the light subpath corresponding to that pixel, and
   b) merged with the vertices of *all* light subpaths.
   c) We then sample the next vertex and do the same.

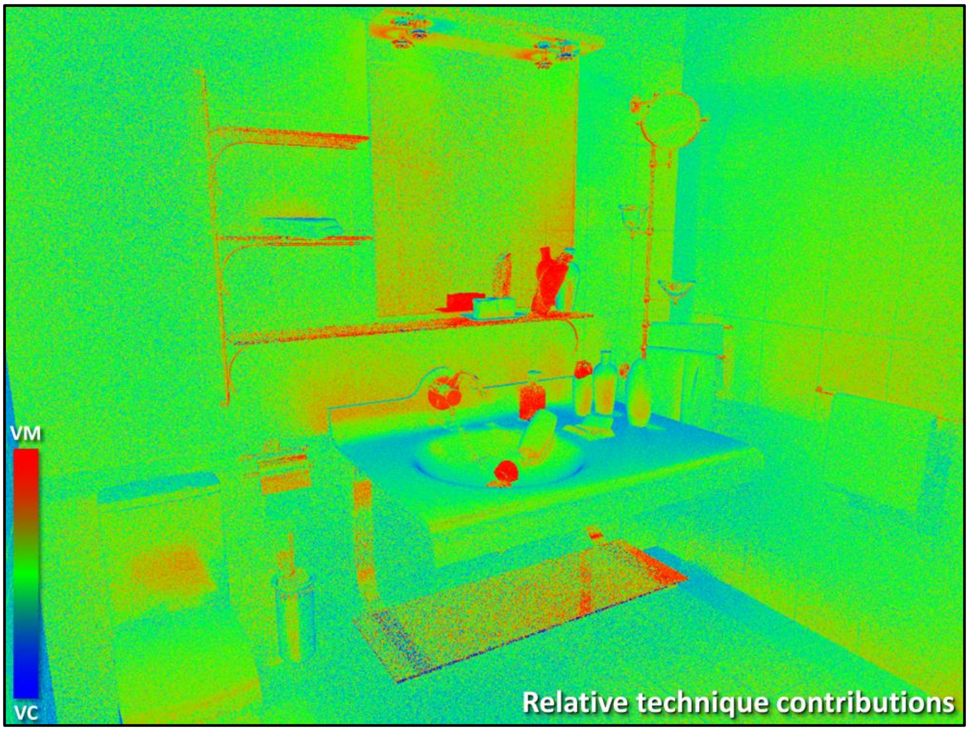
In a progressive rendering setup, we perform these steps at each rendering iteration, progressively reducing the vertex merging radius . For details on this, please refer to the cited papers below for details.



Let us now see how this combined algorithm stacks up against bidirectional path tracing and stochastic progressive photon mapping on a number of scenes with complex illumination.


Stochastic progressive photon mapping (30 min)


Vertex connection and merging (30 min)



Here, we visualize the relative contributions of VM and VC techniques to the VCM image from the previous slide. This directly corresponds to the weights that VCM assigned to these techniques.


AA-340
Bidirectional path tracing (30 min)

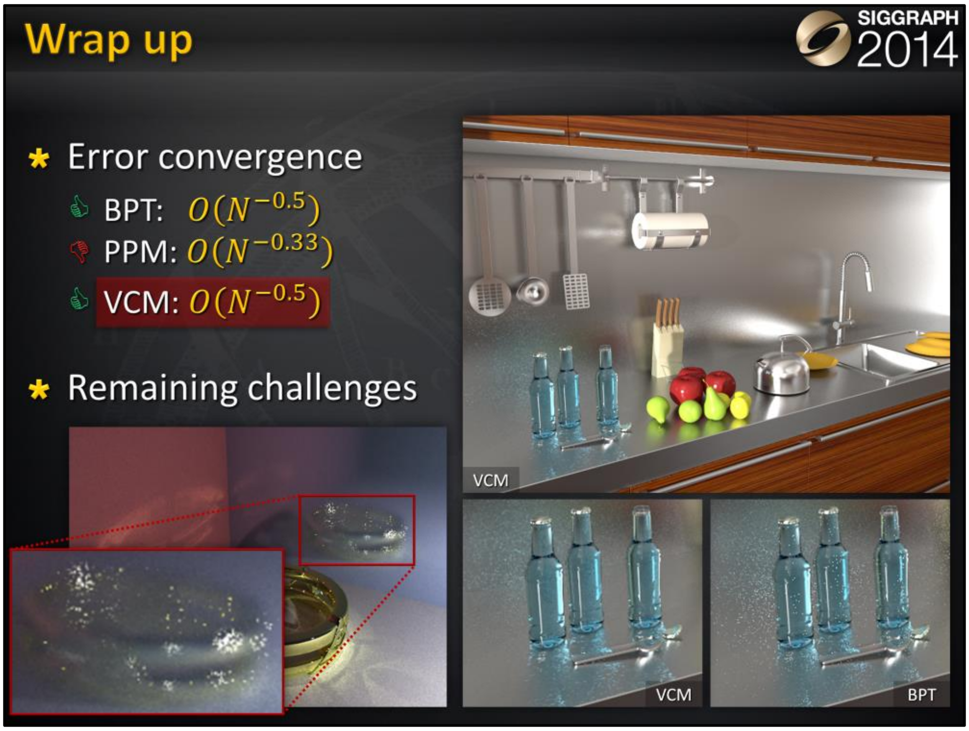

AA-340
Stochastic progressive photon mapping (30 min)


AA-340
Vertex connection and merging (30 min)


VM
VC
AA-340
Relative technique contributions

In summary, vertex connection and merging tries to combine the best of bidirectional path tracing (BPT) and (progressive) photon mapping. An important property of the algorithm is that it retains the higher order of convergence of BPT, meaning that it approaches the correct solution faster than PPM as we spend more computational effort (i.e. sample more paths). The asymptotic analysis can be found in the VCM paper.

Even though VCM is a step forward in Monte Carlo rendering and has proven very useful in practice, it doesn’t come without limitations. Specifically, it cannot handle more efficiently those types of light transport paths that are difficult for both BPT and PM to sample. A prominent example are caustics falling on a glossy surface.



And on this kitchen scene, even though VCM brings practical improvements over BPT, there is still a lot to be desired from the caustics on the glossy surface.

One more thing…

We have recently extended this work to support efficient rendering of various participating media, by augmenting it with various point- and beam-based volumetric light transport estimators. Please refer to the paper for more details.