

Adaptive LightSlice for Virtual Ray Lights

Supplementary Material

R. Frederickx*, P. Bartels, and Ph. Dutré†

Department of Computer Science, KU Leuven, Belgium

1 Deriving the estimate V_i of the final pixel variance

The final pixel variance is the result of three types of stochastic processes that were introduced in the main paper: the VRL tracer \tilde{X} , the VRL integration \tilde{X}_{ij} and the selection of the representatives for each cluster. The resulting combined stochastic process determines the final distribution of the sampled values for a pixel. In principle, when regarding this highly complex combined process, the three subprocess share correlations with each other which, for instance, depend on the specific clustering method that was used. However, to derive a practical algorithm, we treat the subprocesses as independent. Based on our tests, this approximation still yields satisfactory results.

1.1 Variance due to the VRL tracer

We first estimate the variance due to averaging the contribution of N samples from the VRL tracer \tilde{X} to the pixel i , i.e. we want to estimate $\text{Var}[C(\tilde{X}, i)]/N$. This is the expected variance that would be observed in the pixel value in the case that the contribution of each of the N sampled VRLs to the eye ray would be integrated exactly (for the case of VPLs, this corresponds to simply rendering the full set of point lights, as each VPL evaluation is already exact). It describes a base line variance that cannot possibly be improved without tracing more VRLs. An estimate of this variance would be given by

$$V_i^{\text{trace}} = \frac{1}{N(N-1)} \sum_j \left(C(X_j, i) - \sum_k \frac{C(X_k, i)}{N} \right)^2. \quad (1)$$

However, the only values that we have at our disposal are the transfer matrix elements \mathbf{A}_{ij} (which are samples from $\tilde{X}_{ij} = C(X_j, i)/N + \tilde{\xi}_{ij}$) and the estimated VRL integration variances ξ_{ij}^2 . Let us start with some tentative quantity that estimates the variance in

*roald.frederickx@cs.kuleuven.be

†phil.dutre@cs.kuleuven.be

the observed sum of the transfer matrix elements $\sum_j \mathbf{A}_{ij}$:

$$\frac{N}{N-1} \sum_j \left(\mathbf{A}_{ij} - \sum_k \frac{\mathbf{A}_{ik}}{N} \right)^2. \quad (2)$$

It is a realisation of the estimator

$$\frac{N}{N-1} \sum_j \left(C(X_j, i)/N + \tilde{\xi}_{ij} - \sum_k \frac{C(X_k, i)/N + \tilde{\xi}_{ik}}{N} \right)^2 \quad (3)$$

and describes the expected variance of pixel i if we were to render the full set of VRLs without clustering and with Monte Carlo VRL – eye ray integration. Taking the expectation value $\langle \cdot \rangle$ over the VRL integration noise $\tilde{\xi}_{ij}$ and rearranging yields

$$\left\langle \frac{N}{N-1} \left[\frac{1}{N} \left(C(X_j, i) - \sum_k \frac{C(X_k, i)}{N} \right) + \left(\tilde{\xi}_{ij} - \sum_k \frac{\tilde{\xi}_{ik}}{N} \right) \right]^2 \right\rangle, \quad (4)$$

which, after some manipulation, evaluates to

$$\frac{1}{N(N-1)} \sum_j \left(C(X_j, i) - \sum_k \frac{C(X_k, i)}{N} \right)^2 + \sum_j \text{Var}(\tilde{\xi}_{ij}). \quad (5)$$

The first term is exactly the estimate V_i^{trace} of Equation (1) and the second term is estimated by $\sum_j \xi_{ij}^2$. In fact, we could have just written down this estimate at once, by reasoning that the observed variance in the samples is the independent result of the VRL tracer variance and the VRL integration variance. Hence, we have the following estimate of V_i^{trace} in terms of readily available values¹

$$V_i^{\text{trace}} = \frac{N}{N-1} \sum_j \left(\mathbf{A}_{ij} - \sum_k \frac{\mathbf{A}_{ik}}{N} \right)^2 - \sum_j \xi_{ij}^2. \quad (6)$$

1.2 Variance due to the VRL integration and VRL undersampling

The undersampling of the VRLs by selecting only a single representative for each cluster adds additional variance. Moreover, due to the weighting factors of the representative, the inherent VRL integration noise gets amplified as well.

Because we neglect correlations with the VRL tracer, we can assume that the VRLs X_j are simply given, fixed quantities. Within a single cluster q that consists of the VRLs with indices j_1, \dots, j_n , sampling from $\sum_l \tilde{X}_{ij_l}$ is then replaced by sampling from

¹As an aside, note that this estimate can potentially become negative in rare, pathological cases where the VRL integration variance is comparable or larger to the actual VRL tracer variance. Then, variations due to the VRL tracer and integration could happen to cancel each other out in the observed variation of the samples \mathbf{A}_{ij} . However, in such noisy cases, no significant information can be extracted from the sparse pixel samples anyway, and no informed clustering can be made. Moreover, such cases do not cause the algorithm to fail as the total pixel variance estimate V_i will always be positive.

$\tilde{X}_{ijm}/p_m = [C(X_{jm}, i)/N + \tilde{\xi}_{ijm}]/p_m$ with some probability p_m . We use the general formula $\text{Var}(x) = E(x^2) - E(x)^2$ to arrive at the estimate V_{iq}^{und} of the variance of this process:

$$V_{iq}^{\text{und}} = \left\langle \sum_l p_l \left[\frac{C(X_{jl}, i)/N + \tilde{\xi}_{ijl}}{p_l} \right]^2 \right\rangle - \left[\sum_l C(X_{jl}, i)/N \right]^2. \quad (7)$$

Note that the last term could be written down immediately because the process is unbiased. The total variance with contributions from all clusters is simply $V_i^{\text{und}} = \sum_q V_{iq}^{\text{und}}$. Because the samples are independent between clusters, we can just focus on the contribution of a single cluster q .

Performing the average over $\tilde{\xi}$ in Equation (7) and estimating $C(X_{jl}, i)/N$ by \mathbf{A}_{ijl} gives an estimate of the undersampling variance V_{iq}^{und} of this cluster in terms of known quantities:

$$V_{iq}^{\text{und}} = \sum_l \frac{1}{p_l} [\mathbf{A}_{ijl}^2 + \xi_{ijl}^2] - \left[\sum_l \mathbf{A}_{ijl} \right]^2. \quad (8)$$

Because the cost of evaluating a cluster is constant (evaluating the contribution of its representative VRL), the convergence constant c is minimized by choosing p_l such that the variance due to sampling the representative is minimized. If we write $p_l = w_l / \sum_k w_k$ for some unnormalized weights w_l , then taking the derivative of Equation (8) with respect to w_m yields

$$\sum_l \frac{1}{w_l} [\mathbf{A}_{ijl}^2 + \xi_{ijl}^2] - \frac{\sum_k w_k}{w_m^2} [\mathbf{A}_{ijm}^2 + \xi_{ijm}^2], \quad (9)$$

which equals zero for the optimal weights $w_m = \sqrt{\mathbf{A}_{ijm}^2 + \xi_{ijm}^2}$. Note that, if $\xi_{ij}^2 = 0$ (as would be the case for LightSlice for VPLs), then $p_m = \mathbf{A}_{im} / \sum_l \mathbf{A}_{il}$, the value returned from the discrete sampling is $\mathbf{A}_{ijm}/p_m = \sum_l \mathbf{A}_{il}$ and $V_{iq}^{\text{und}} = 0$. Indeed, if we have exact information of all contributions to a pixel, we can just always return the exact total contribution for that pixel. However, the sampling of the representative VPLs/VRLs happens for the entire slice and the combined undersampling variance of *all* pixels in that slice can in general not be made equal to zero by a single choice of p_m . When combining the variance over such a set of P (or P^{repr}) pixels by summing over the corresponding i indices in Equation (8), the resulting optimal weights read

$$w_m = \sqrt{\sum_i [\mathbf{A}_{ijm}^2 + \xi_{ijm}^2]}. \quad (10)$$

2 Minimizing c

For each slice, we start from a single cluster that holds all the VRLs and repeatedly split the cluster with the highest contribution to $\sum_i^{P^{\text{repr}}} V_i$ until the minimum of the convergence constant c is reached. In order to do so, we need to efficiently calculate this contribution for all tentative cluster splittings (Sec. 2.1) and we need to know when the minimum of c has been reached (Sec. 2.2).

2.1 Computing the cluster contribution to $\sum_i V_i$

The estimate V_i^{und} depends on the cluster under consideration and can be computed in a numerically robust on-line manner. This allows the optimal cluster split to be found in linear time in the number of VRLs in the original cluster by accumulating the variances of both resulting clusters. If the VRLs are sorted so that j_1, \dots, j_k and j_{k+1}, \dots, j_n are the tentative sub clusters for $k = 1, \dots, n - 1$ (through projecting the VRLs on a P^{repr} -dimensional line as explained in the main text), then the total resulting variance of the two sub clusters for each of the $n - 1$ possible splits can be found by accumulating V_i^{und} in a forward (j_1, j_2, \dots) and backward pass (j_n, j_{n-1}, \dots).

2.2 Finding the minimum of c

During the initial phase of the cluster splitting, when there are still few clusters (N^{repr} is small and $N^{\text{repr}}P \ll NP^{\text{repr}}$), the effect of $\sum_i V_i$ dominates the behaviour of c and we have $c \sim 1/N^{\text{repr}}$. As the number of clusters increases, the variance V_i converges to the unclustered variance $V_i^{\text{trace}} + \sum_j \xi_{ij}^2$ and one has $c \sim N^{\text{repr}}$. We can use this knowledge to derive a lower bound on the possible convergence constants that can be observed after further cluster splitting: no future convergence constant can be smaller (i.e. better) than

$$\bar{c} = (NP^{\text{repr}} + N^{\text{repr}}P) \sum_i^{P^{\text{repr}}} \left(V_i^{\text{trace}} + \sum_j \xi_{ij}^2 \right). \quad (11)$$

Hence, as soon as we arrive at a number of clusters N^{repr} for which $c \geq \bar{c}$, we know with certainty that the optimal clustering that was observed thus far is in fact the global optimum.

3 Number of VRL evaluations

Figure 1 shows false color images pertaining to the number of evaluated VRLs during rendering.

Box For the unclustered rendering of the Box scene, each VRL that was traced is evaluated exactly once for the direct eye ray, and a second time for the reflected ray if the reflective sphere is directly visible. When adaptive clustering is enabled, pixels that are near the light source (or for which the light source is visible, as in the reflection in the sphere) are assigned the most number of VRLs (around 3% of the total), whereas pixels in the volumetric shadow regions of the sphere and box are evaluated with only around 1% of the total number of VRLs. The heterogeneity of the scene allows us to significantly outperform a fixed clustering rate by locally adapting the number of VRLs.

Brain The unclustered rendering of the Brain scene shows more structure compared to that of the Box scene, due to extra rays spawned from reflections and refractions in the dielectric material (Russian roulette keeps the number of evaluated VRLs finite). The ‘Relative’ image cancels out this effect and cleanly shows the relative number of evaluated VRLs. The majority of the work is performed for pixels in the center of the brain, as rays entering this region and reflecting and refracting on the inner lobe structures of the model get scattered in all directions and hence get contributions from many VRLs. Rays entering the model near the edges meet less internal structure and get contributions from a smaller set of VRLs, allowing more aggressive clustering. Overall, there is less variability that can be exploited compared to the Box scene, and we are merely on par with the ideal fixed clustering fraction.

Glass The grape juice in the Glass scene shares the behaviour of the dielectric participating medium of the Brain scene, but the extra dielectric medium of the glass itself now leads to the observed less-than-unity fraction of unclustered VRLs. Rays entering the glass near the back surface at the top or the solid glass portion at the bottom now first experience refraction and reflection events before entering the participating medium itself. A fraction of these rays will have been terminated by Russian roulette before they even meet a single VRL. Again, the ‘Relative’ image cancels out this effect. In the bulk of the grape juice, similar relative fractions are observed as for the other scenes (a reduction to around 1%). Near the top and bottom, in the areas where many internal reflections occur before the eye rays enter the grape juice, there is much less reduction in the number of evaluated VRLs compared to unclustered rendering: due to variations in the number of internal reflections, the rays accumulate contributions from a diverse set of VRLs. The adaptive clustering detects this and only reduces the number of VRLs to roughly 10%, here. Overall, the relative homogeneity of the lighting in the bulk of the medium makes it harder for the adaptive clustering to beat an a priori fixed ideal clustering rate, as it has to base itself on a sparse and noisy sampling. As a consequence, the ideal fixed clustering rate outperforms the adaptive clustering by around 3% for this scene.

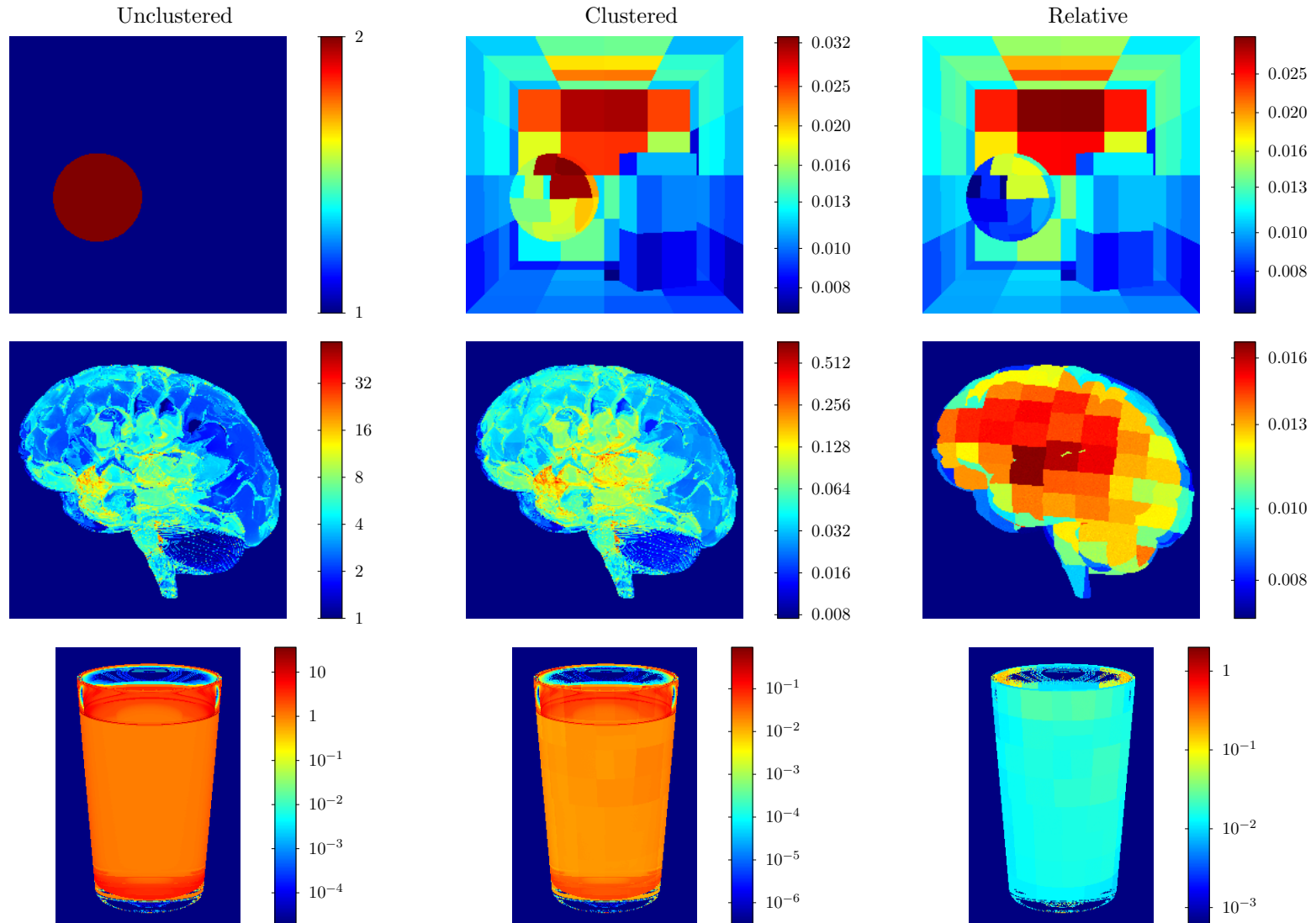


Figure 1: Number of evaluated VRLs during rendering (all images in logarithmic scale for clarity). For every pixel, the total fraction of VRLs that are evaluated during rendering, relative to the number of VRLs that were traced is visualized in the ‘Unclustered’ and ‘Clustered’ columns when rendering without clustering and with our adaptive clustering method, respectively. The ‘Relative’ column shows the fraction of evaluated VRLs with clustering, compared to the number of evaluated VRLs without clustering for each pixel (with an equal number of VRLs traced), i.e. it is the ‘Clustered’ column divided by the ‘Unclustered’ column. See Section 3 for an in-depth discussion.