Ensemble Metropolis Light Transport: Supplemental Material

THOMAS BASHFORD-ROGERS, University of the West of England, UK
LUÍS PAULO SANTOS, Universidade do Minho / INESC TEC, Portugal
DEMETRIS MARNERIDES, University of Warwick, UK
KURT DEBATTISTA, University of Warwick, UK

ACM Reference Format:
Thomas Bashford-Rogers, Luís Paulo Santos, Demetris Marnerides, and Kurt Debattista. 2021. Ensemble Metropolis Light Transport: Supplemental Material. *ACM Trans. Graph.* 1, 1, Article 1 (January 2021), 4 pages. https://doi.org/10.1145/3472294

In this document, we provide further details on:

- the selection of α and the ensemble size
- the data structure used to store the complementary ensemble to allow fast queries
- how EMLT scales with the number of threads
- a visualization of the perturbations on the image plane for all other scenes in the main paper
- why normals are not included in the similarity measure via an example
- using EMLT for rendering caustics

1 IMPACT OF α

To assess the impact of the α parameter for selecting perturbation sizes when using the Guided Lens perturbation we investigated using a static lower and upper bound for a region of the DOOR AJAR where the impact of this is especially apparent. This shows that using a constant small value for α leads to low-frequency noise, a too high value results in jumps on the image plane which are too large and and leads to perturbations being rejected. The sigmoid weighting between the two extremes leads to fewer rejected perturbations while exhibiting lower frequency noise.

2 ENSEMBLE SIZE

To determine the size of the ensemble of paths used, we performed an experiment to assess how MSE varies with ensemble size. We computed an image at 64 SPP for each scene used in this paper, and varied the ensemble size from 1204 paths to 131072 paths. As the MSE varies for each scene, we normalized each MSE by average scene luminance to be able to compare values across scenes. Figure 2 shows the results of this analysis, where the black line is an average across scenes. The minimal MSE occurs at 16384 paths. We found that fewer paths provide insufficient information to guide sampling in some scenes, whereas using more paths leads to inefficient exploration of the space in a finite rendering time. The reason for this latter point is that if there are N paths being computed over $T = \text{width} \times \text{height} \times \text{SPP}$, each path can perform $\frac{T}{N}$ mutations and perturbations. As N increases, each path has fewer steps to explore the space, leading to a decrease in convergence.

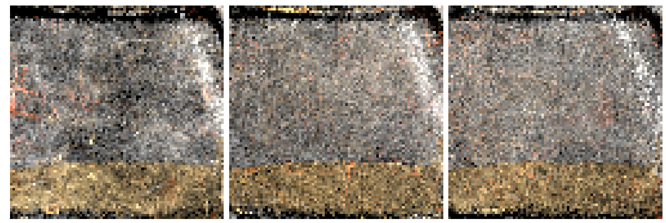


Fig. 1. Different values for the α parameter shown for an inset of the glass teapot in the DOOR AJAR scene. The left image shows a small constant $\alpha = 0.05$ leading to clumping of paths, the middle shows the adaptive value for α used in this paper which balances between extremes, and the right image shows $\alpha = 0.5$ where the size is too large leading to a higher chance of sampling invalid paths especially when lighting varies in a small region such as the reflection in the glass.

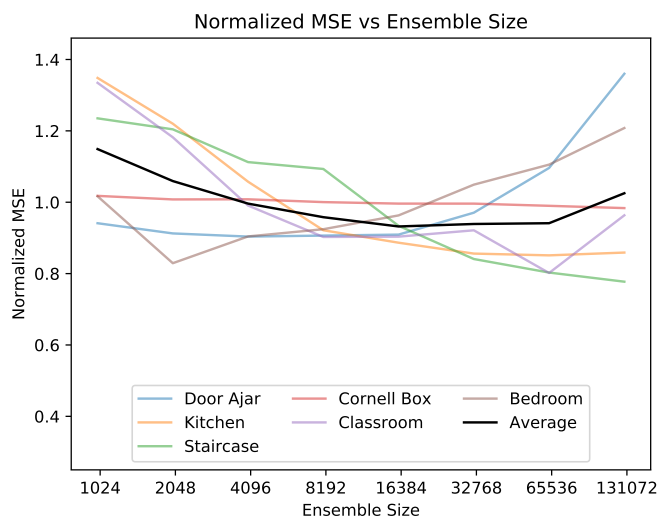


Fig. 2. MSE of different ensemble sizes across all scenes used in this paper. The black line shows an average, with a minimum at 16384 paths.

Authors' addresses: Thomas Bashford-Rogers, University of the West of England, Coldharbour Ln, Bristol, BS16 1QY, UK; Luís Paulo Santos, Universidade do Minho / INESC TEC, Campus de Gualtar, Braga, 4710-057, Portugal; Demetris Marnerides, University of Warwick, Gibbet Hill Road, Coventry, CV4 7AL, UK; Kurt Debattista, University of Warwick, Gibbet Hill Road, Coventry, CV4 7AL, UK.

0730-0301/2021/1-ART1 $15.00
https://doi.org/10.1145/3472294

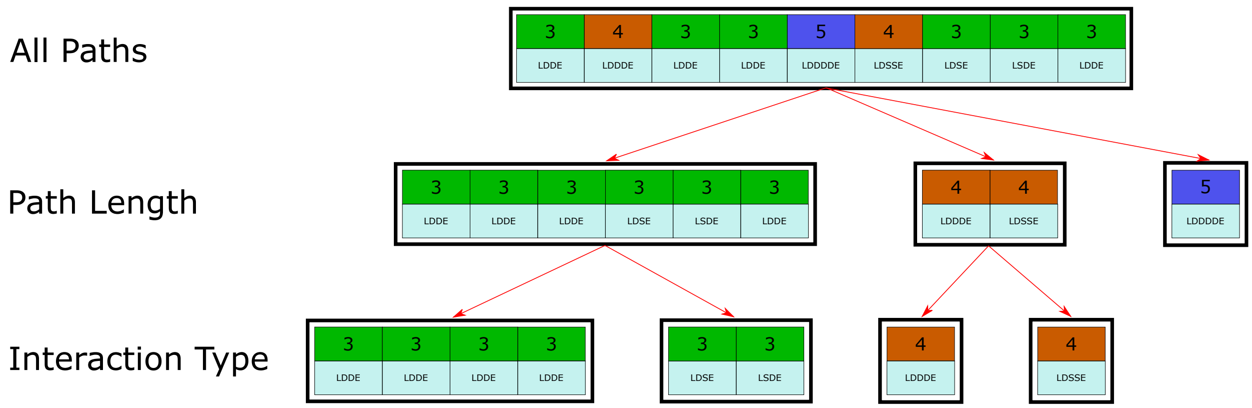


Fig. 3. Overview of the pool structure used to accelerate lookups of paths from the complementary ensemble. This diagram shows the three level structure we use: the root contains all paths in the ensemble, each node on the first level contains paths of the same number of path vertices, then the final level clusters paths with the same interaction types. Perturbation strategies which rely on sampling paths of the same length and interaction type can be quickly found by traversing the tree.

3 POOL ACCELERATION STRUCTURE

The data structure used to efficiently query paths with similar properties from $\mathbf{Y}$ (the complementary ensemble) is implemented as a three level tree of pointers to light paths. The root contains a list of all light paths in the complementary ensemble $\mathbf{Y}$. The second level of the tree contains nodes storing a list of paths of equal length, and the third level contains all paths with the same interaction type for all vertices in the path. Figure 3 illustrates the structure of the tree. This tree is fast to construct with $O(|\mathbf{Y}|)$ complexity, and uses minimal memory due to storing lists of pointers.

Lookups can be performed efficiently based on the criteria specified by the transition kernel by traversing the branches of the tree which match the criteria. The proposed transition kernels require either one random path from the ensemble, or a set of M deterministically chosen paths. In either case, the tree is first traversed through nodes which match the criteria from the kernel until all criteria are matched, or no matching criteria are found and the traversal returns no paths. If the criteria are matched, and one random path is required, then a path from the list of paths in the node is returned at random. If a set of deterministically chosen paths is required, then we propose a deterministic selection procedure. Similar to many hashing schemes, this takes in an integer value ID and returns a set of M integer values, corresponding to indices of paths within the list of paths in the node. This mapping $H : \mathbb{Z} \mapsto \mathbb{Z}^M$ can be implemented through any deterministic mapping. We implement H as a function which returns $(ID + o) \mod |Node|, o \in [1..M]$, where $|Node|$ is the number of paths in the node. If this is less than M then we return all paths in the node. Traversal therefore has $O(1)$ complexity. These paths are stored in the set $\Upsilon = \{\bar{v}^1..\bar{v}^M\}$ which contains the M paths resulting from the tree lookup. ID is set to a random integer for each path in $\mathbf{X}$ and $\mathbf{Y}$, and is incremented every lookup.

For example, if a transition kernel requires a set of paths of length 4, with interaction types *LDSE*, then the root node does not satisfy the criteria, so the tree is followed to the set paths of length 4, then the node containing the set of paths with interaction types *LDSE* is located, and finally H is applied to the node returning a set of paths. If no paths of the required type are found, this then backtracks in the tree and returns a set of paths of the matching length, again through the application of H.

4 MULTITHREADING SCALABILITY

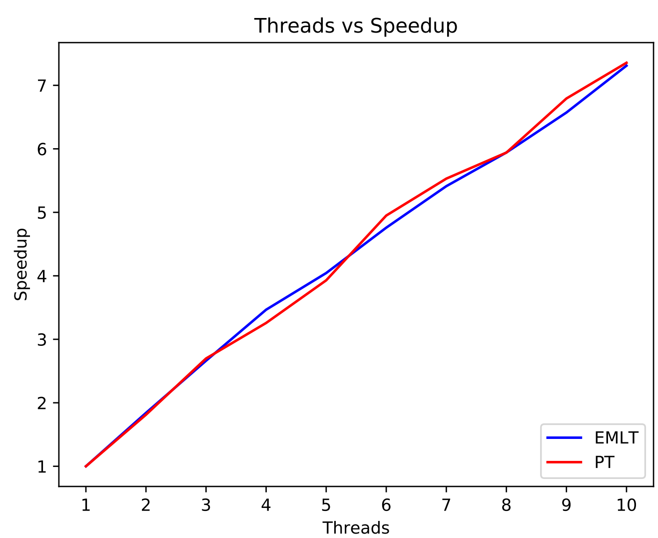


Fig. 4. Graph showing how the performance of EMLT scales with the number of threads. We have also included the scalability graph for Path Tracing (PT) for reference.

Figure 4 shows how the performance of EMLT scales with the number of threads. We implemented all methods in the paper using a job queue where each job consists of updating each path. This leads to fine grained load balancing and efficient utilization of available threads. The only synchronization point is updating data structures swapping between ensembles, but this involves creating a tree of

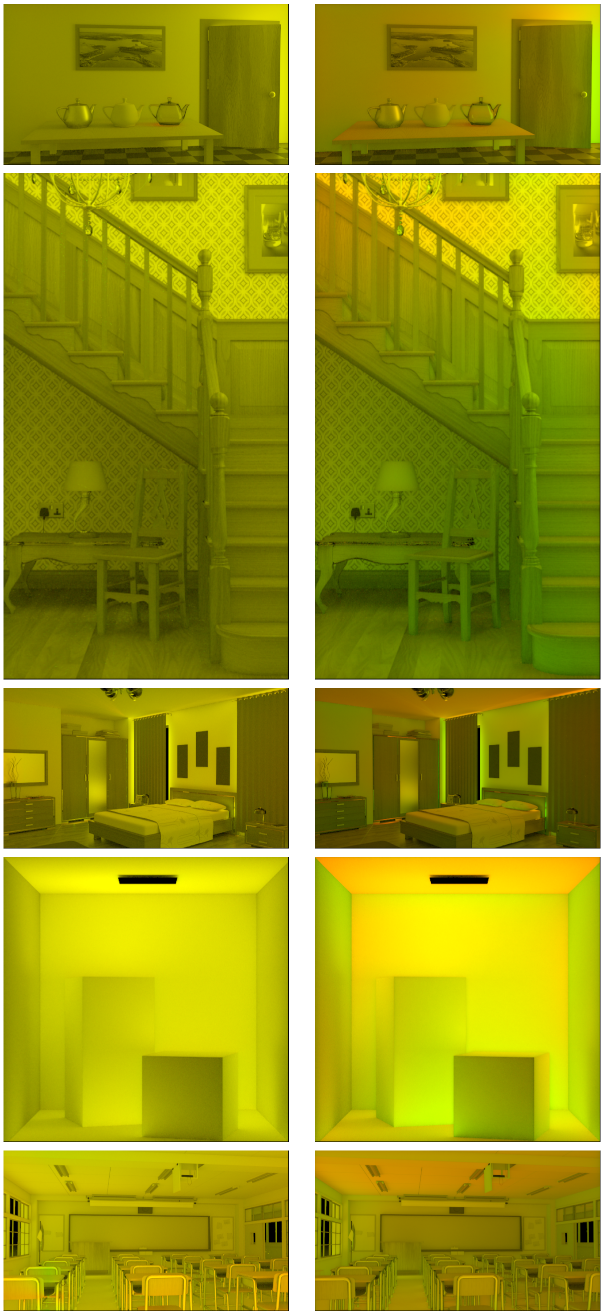


Fig. 5. Visualization of perturbations on the image plane for the remainder of the scenes used in this paper (the KITCHEN scene is shown in Figure 12 in the main paper). Green colors mean perturbations were predominately vertical, red mean predominately horizontal, and yellow means perturbations were predominately isotropic. The left column shows perturbations from MLT and the right shows our method.

paths in $O(|\mathbf{Y}|)$ time, which is small as $|\mathbf{Y}|$ is small, see Section 2 of this document. Therefore, Figure 4 illustrates a linear scaling in performance with the number of threads as expected. We also show the scaling for path tracing implemented in the same system for comparison, also exhibiting the linear scaling as expected.

5 MUTATION MAPS

Figure 5 visualizes the anisotropic perturbations for the remainder of the scenes in the main paper. Similar to Figure 12 in the main paper, green represents perturbations which lead to vertical shifts of path position on the image plane, red shows horizontal, while yellow are isotropic. Results for MLT are in the left column and our method in the right. This shows that the perturbations proposed by our approach adapt to both geometry and lighting information.

6 IMPACT OF NORMALS ON THE SIMILARITY MEASURE

The similarity measure as described in the paper does not include normals. If normals are included, then this measure leads to lower similarity scores for paths which may be spatially local and useful for guiding perturbations. Figure 6 shows results using a similarity measure with and without normals for the DISPLACED WALL scene, a wall with high frequency displacements lit by an environment map. The image without normals, right, shows a reduction in MSE of around 5% compared to including normals.

7 CAUSTICS RESULTS

The scenes in the main paper have shown lighting consisting of all types of surface interaction. However, to investigate the performance of EMLT for caustics in particular, we examine an example scene containing glass shards lit from above, resulting in caustics on the ground, and multiple specular interactions between the shards. In Figure 7 we show results for this scene rendered at 8 mutations per pixel to illustrate the differences between the methods. EMLT is able to capture thin features of the caustics compared to MLT, as shown in the insets, and is able to better explore the lighting on the glass fragments.

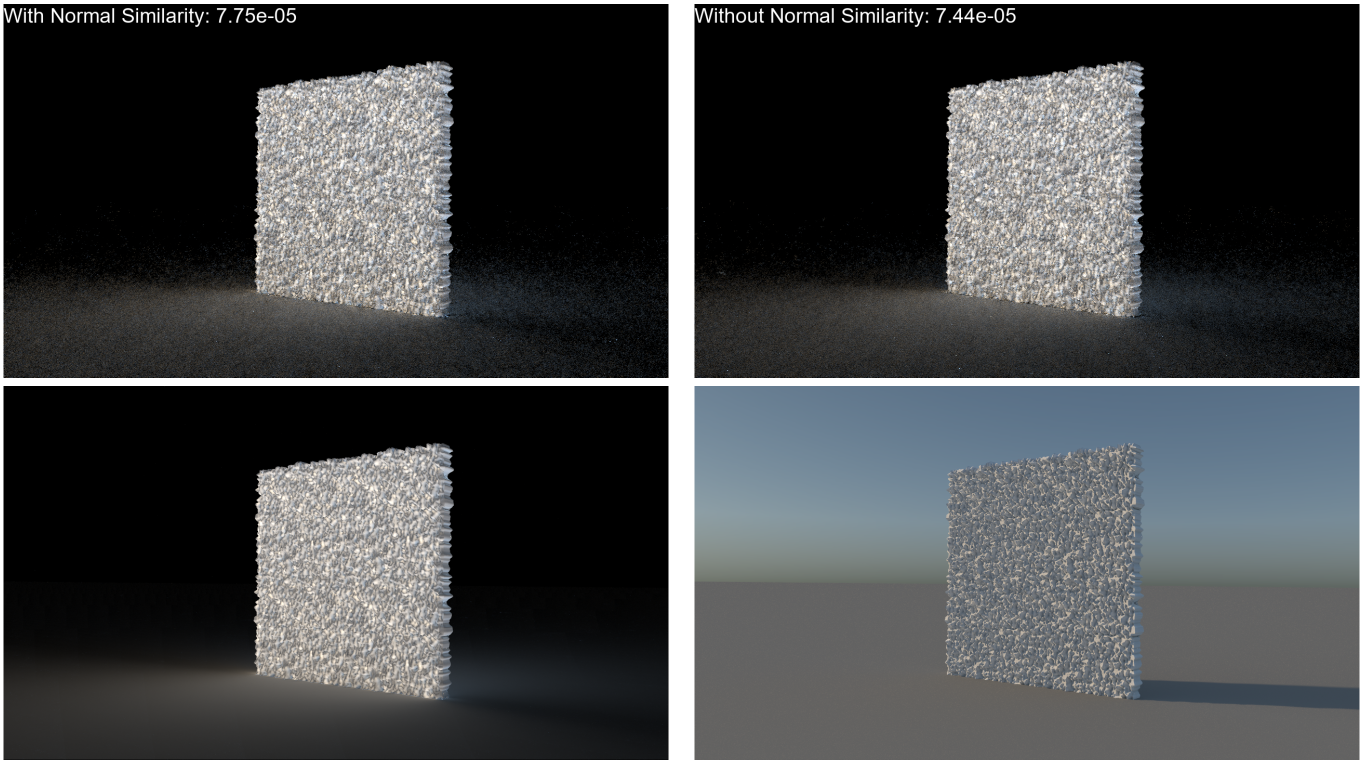
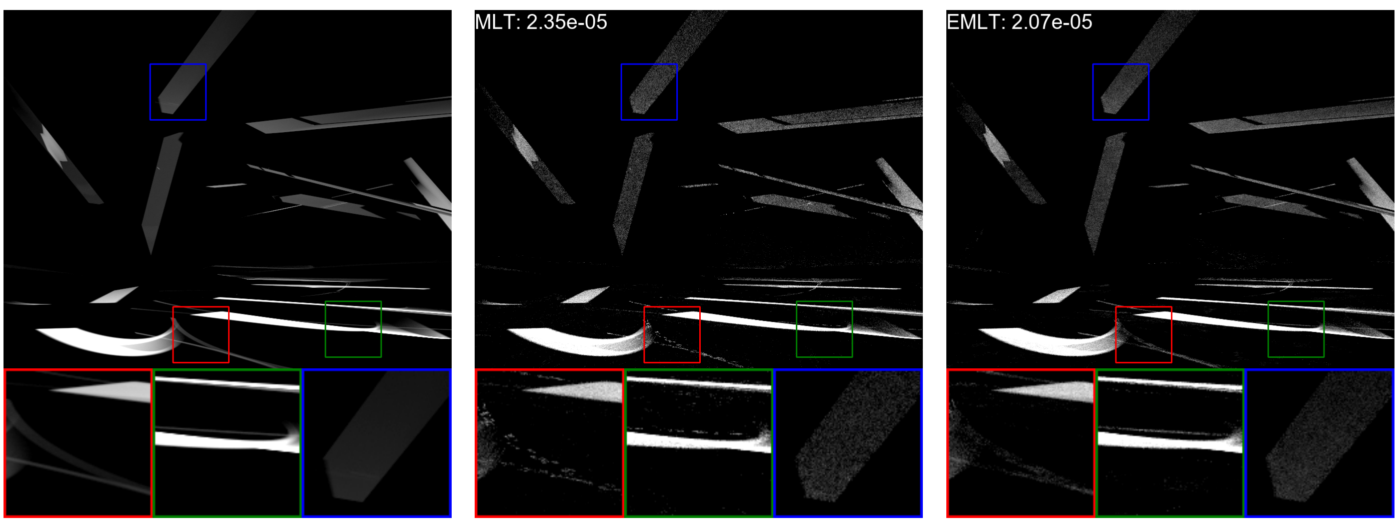


Fig. 6. The DISPLACED WALL scene showing the impact of using normals when computing the similarity measure for indirect lighting. The top left image includes normals, the top right excludes normals, the bottom left is a reference, and the bottom right shows the lighting environment for reference.



Reference MLT EMLT

Fig. 7. Results for the shards scene showing a reference rendering on the left, MLT in the center, and EMLT on thr right.