Additional Material: Fourier Opacity Optimization for Scalable Exploration

Irene Baeza Rojo, Markus Gross, and Tobias Günther

1 Further Results

In the following, we provide further examples of the comparisons shown in the main paper.

1.1 Comparison of Importance Approximation

In Section 5.2 of the main paper, we concentrated on the approximation of the importance sum using Fourier series. To isolate the effect of the importance approximation from the subsequent smoothing steps, we directly displayed the fragment opacities without any object-space smoothing. Fig. 1 shows more results.

1.2 Comparison with Blending Approximations

In Section 5.3 of the main paper, we compared multiple blending approximations with each other. Fig. 2 shows another example of a HELICOPTER in forward flight close to the ground. Close-ups of the front vortex and the wake turbulence show that our method preserves the order of lines best.

1.3 Comparison with Decoupled Opacity Optimization

In Section 5.4 of the main paper, we compared our method with decoupled opacity optimization [2], where we show results of the complete pipeline. Fig. 5 adds two more examples for the TORNADO and the ECMWF data set.

1.4 Subset for Fourier Approximation

In Section 5.5.3 of the main paper, we showed that it is possible to use a small percentage of the entire geometry, that is, a subset $\frac{1}{n}$ of the data, to approximate the coefficients for the squared importance approximation without losing important structures in the visualization. Fig. 3 shows a close-up of the effect that the skipping of points can produce, here in the DARK SKY simulation and the VISCOUS FINGERS data sets. The larger n, the more the context geometry (blue) fades away. Note, however, that the important constellation network (red) is still always visible.

2 Additional Experiments

To shed further light on the behavior under more restricted parameter constraints, we performed two additional experiments.

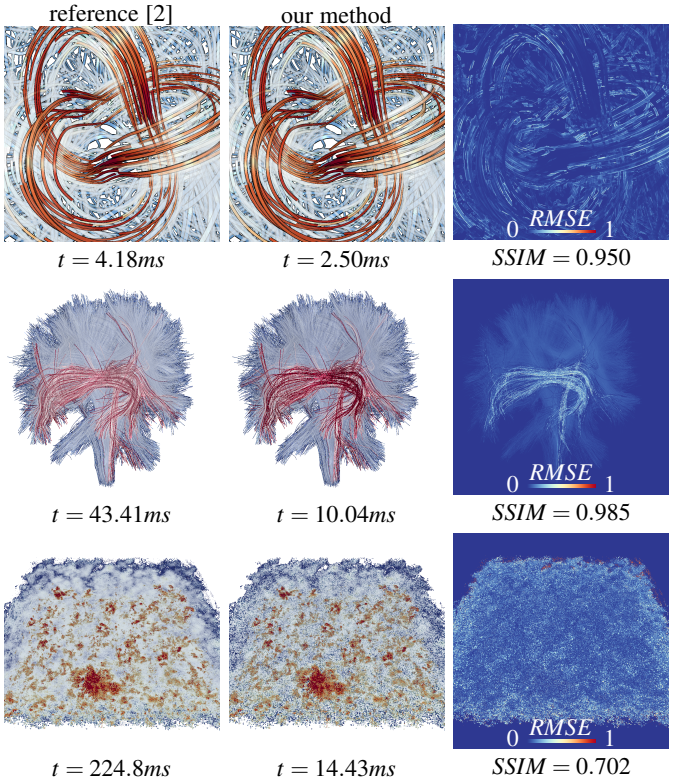

Fig. 1: Further comparisons of the importance approximation between our method and decoupled opacity optimization [2], showing the RMSE, the SSIM index and the GPU render time (t) in *ms* for the same energy weights in the TREFOIL ($q = 50$, $r = 200$, $\lambda = 2$), DT-MRI ($q = 300$, $r = 200$, $\lambda = 0.26$) and CTBL ($q = 100$, $r = 500$, $\lambda = 3$) data sets. The segment opacity computation via atomic min and the per-segment smoothing (end of step 2 and step 3 of section 4.2 of the paper) are skipped. In both cases, the transmittance is computed directly using fragment linked lists.

2.1 Restricted Time Budget

In Section 5.3 of the main paper, we compared our method with other color blending approximations [1], [3] that are faster than our method. Our method, however, achieves better results due to the additional representation of the depth order. Still, our method would be slower if the number of bands m is set to a very small number (1 or 2), which might give unsatisfactory results due to the coarse depth approximation. We found that using $m = 5$ is a good trade-off between performance and accuracy. Using this number in

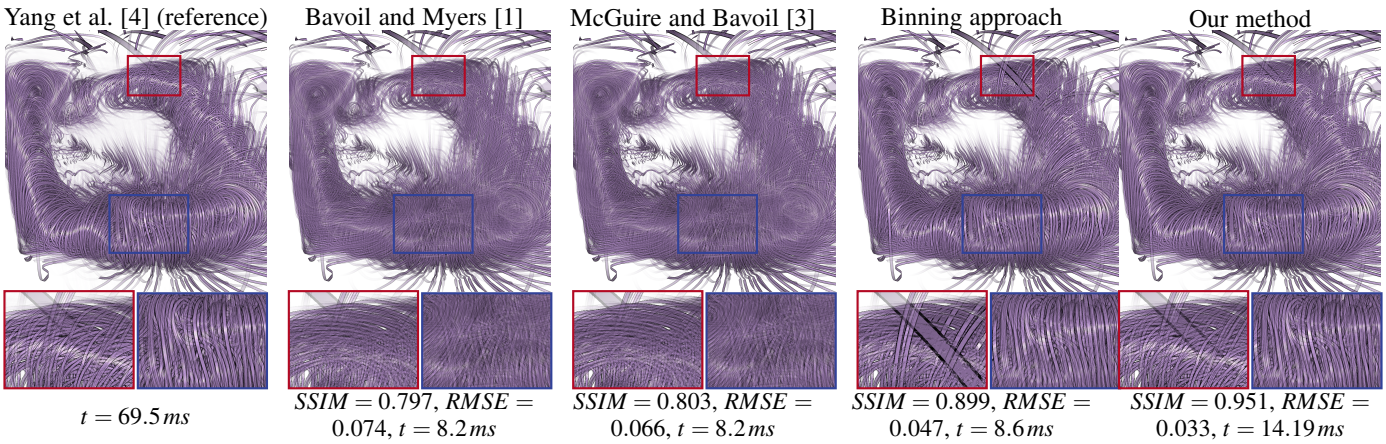

Fig. 2: In this figure, we show an additional comparison of multiple blending approximations, from left to right: fragment linked lists (reference solution) by Yang et al. [4], weighted averaging by Bavoil and Myers [1], the extension with exact background term by McGuire and Bavoil [3], binning of $\tau(d)$ into 11 bins, and our Fourier-based method. Here, the HELICOPTER ($q = 10k$, $r = 6k$, $\lambda = 0.38$) data set is shown. We list the structural similarity index (SSIM), the root-mean-square error (RMSE) and the GPU time (t) in *ms* for each data set. All other methods are faster than our method, but ours retains the order of fragments best, as shown in the zoom-in images, which makes a correct perception of the layer order possible and results in the lowest approximation errors.

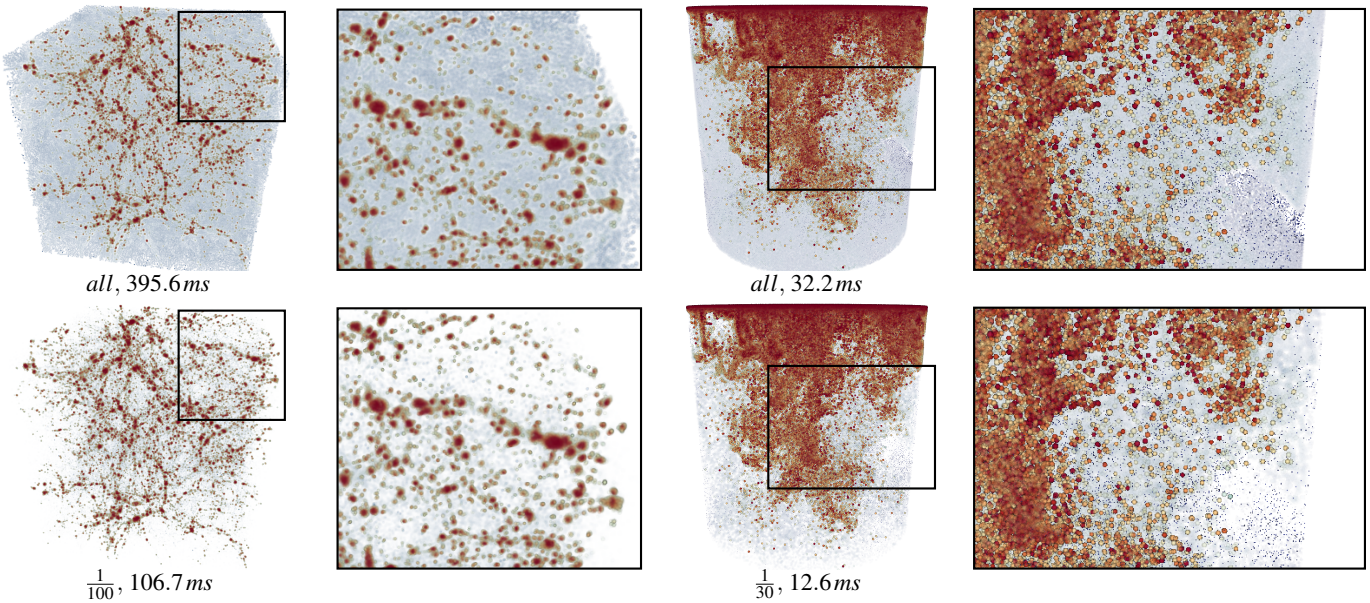

Fig. 3: Results and total GPU render time when using varying subset sizes for the construction of $G(d)$ and $\tau(d)$ in the DARK SKY (left, $q = 100$, $r = 20$, $\lambda = 1.3$) and the VISCOUS FINGERS data set (right, $q = 100$, $r = 500$, $\lambda = 3$). While the importance of the context is underestimated when using only a subset of the geometry, the important structures always remain visible.

the experiments in the main paper, showed the advantages of using our method (correct perception of layer order) compared with faster but too simple previous methods. There might be cases where the user has a restricted time budget and would prefer to drastically lower the number of bands. In order to provide an intuition of how our method would behave, Fig. 6 shows a repetition of the experiment in the main paper using only $m = 2$ frequency bands in the Fourier series. Note that visually, it is difficult to perceive the difference, and the most noticeable effect is a slight increase or decrease of transparency in the geometry due to the low-frequency approximation, which leads to a decrease of the SSIM and an increase of the RMSE.

2.2 Binning approach

For the binning approach shown in Section 5.3 of the main paper, we used a linear interpolation between the bins to approximate $\tau(d)$. For completeness, Fig. 4 shows a comparison of the results when using piecewise constant bins or linear interpolation. The piecewise constant approach results in very noticeable depth discontinuities at the bin boundaries, which are very apparent during camera navigation.

3 PROOF OF NORMALIZATION

To cope with overshooting and undershooting in the Fourier approximation of the optical depth $\tau(d)$, we introduced a normalization in the main paper in Eq. (19) that is based on the following theorem.

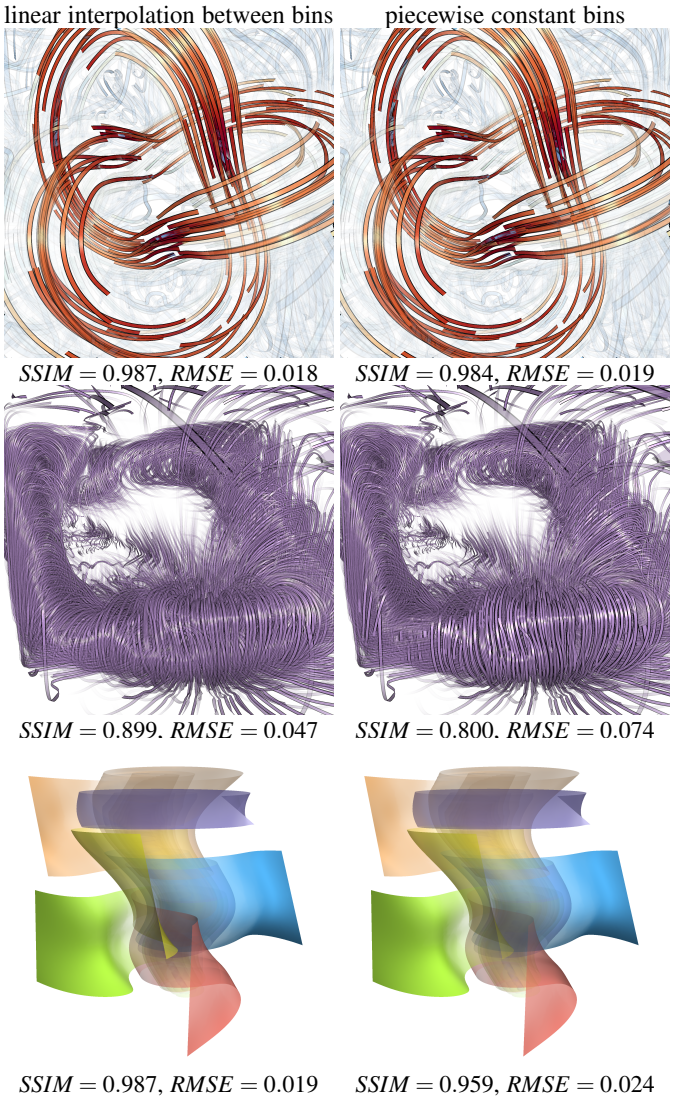

Fig. 4: Comparison of results using the binning approach with linear interpolation (left) or piecewise constant bins (right) for the approximation of $\tau(d)$. The piecewise constant approximation leads to noticeable artifacts at bin boundaries.

Theorem 1. *For any* $n \in \mathbb{N}$: $\sum_{i=1}^{n} \frac{\alpha_i}{1-\alpha_i} e^{-\tau(d_i)} = 1 - e^{-\tau(1)}$.

Proof. By Eqs. (14) and (18) of the main paper, Theorem 1 is equivalently expressed as:

$$\sum_{i=1}^{n} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) = 1 - \prod_{i=1}^{n}(1-\alpha_i) \tag{1}$$

The equivalence of Eq. (1) is shown by induction. For the base case $n = 1$, we have $\alpha_1 = 1 - (1-\alpha_1)$, which is true. Assuming that Eq. (1) holds up to n, we show that it holds for $n+1$, i.e.:

$$\sum_{i=1}^{n+1} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) \stackrel{!}{=} 1 - \prod_{j=1}^{n+1}(1-\alpha_j) \tag{2}$$

First, the left hand side of Eq. (2) is split:

$$\sum_{i=1}^{n+1} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) = \sum_{i=1}^{n} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) + \alpha_{n+1} \prod_{j=1}^{n+1-1}(1-\alpha_j) \tag{3}$$

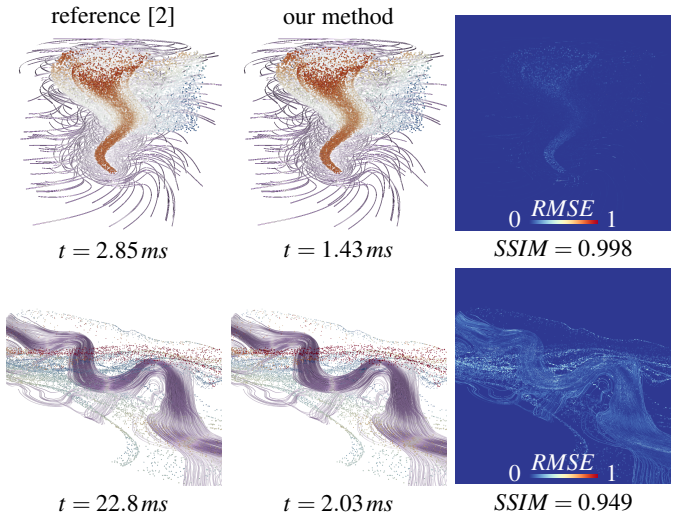

Fig. 5: Further comparisons with decoupled opacity optimization [2], showing RMSE, SSIM index and GPU time in *ms* for the same energy weights in the TORNADO ($q = 60$, $r = 500$, $\lambda = 2$) and ECMWF ($q = 400$, $r = 50$, $\lambda = 4$) flow.

Next, we insert the inductive hypothesis, i.e., Eq. (1), into Eq. (3):

$$\sum_{i=1}^{n+1} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) = 1 - \prod_{j=1}^{n}(1-\alpha_j) + \alpha_{n+1} \prod_{j=1}^{n}(1-\alpha_j) \tag{4}$$

By factoring out $\prod_{j=1}^{n}(1-\alpha_j)$ on the right of Eq (4) we obtain:

$$\sum_{i=1}^{n+1} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) = 1 - (1-\alpha_{n+1}) \prod_{j=1}^{n}(1-\alpha_j) \tag{5}$$

which is equivalent to Eq. (1) after joining $(1-\alpha_{n+1})$ into the product:

$$\sum_{i=1}^{n+1} \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j) = 1 - \prod_{j=1}^{n+1}(1-\alpha_j) \qquad \square$$

REFERENCES

[1] L. Bavoil and K. Myers. Order independent transparency with dual depth peeling. Technical report, NVIDIA Research, 2008.
[2] T. Günther, H. Theisel, and M. Gross. Decoupled opacity optimization for points, lines and surfaces. *Computer Graphics Forum (Proc. Eurographics)*, 36(2):153–162, 2017.
[3] M. McGuire and L. Bavoil. Weighted blended order-independent transparency. *Journal of Computer Graphics Techniques (JCGT)*, 2(2):122–141, 2013.
[4] J. C. Yang, J. Hensley, H. Grün, and N. Thibieroz. Real-time concurrent linked list construction on the GPU. *Computer Graphics Forum*, 29(4):1297–1304, 2010.

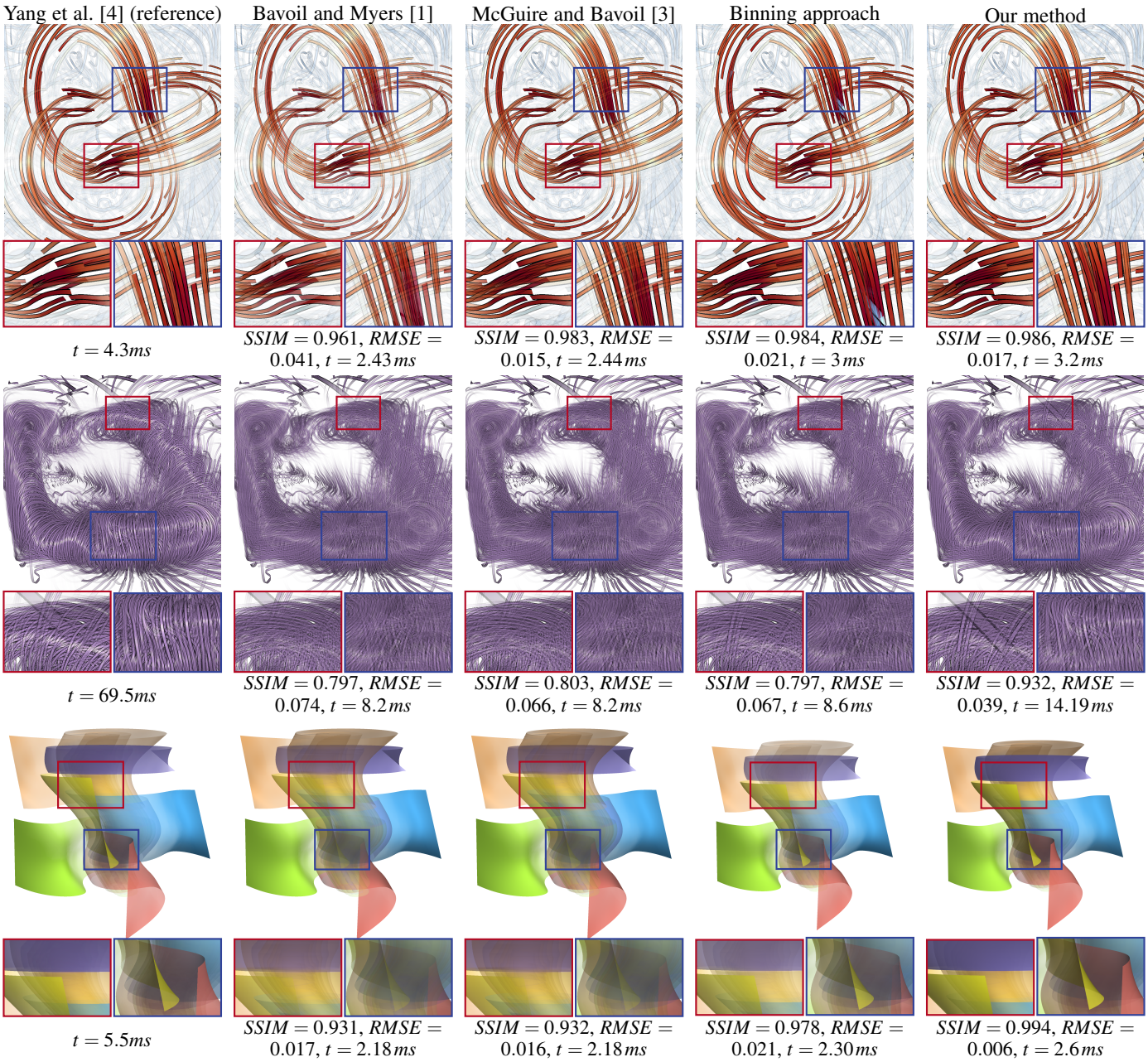

Fig. 6: In this figure, we compare multiple blending approximations with each other, from left to right: fragment linked lists (reference solution) by Yang et al. [4], weighted averaging by Bavoil and Myers [1], the extension with exact background term by McGuire and Bavoil [3], binning of $\tau(d)$ with 3 bins (equal-memory comparison to our method) and our Fourier-based method using only $m = 2$ frequency bands. The rows show the TREFOIL (top, $q = 50$, $r = 500$, $\lambda = 2$), the HELICOPTER (middle, $q = 10k$, $r = 6k$, $\lambda = 0.38$) and the TORNADO (bottom, $q = 10$, $r = 500$, $\lambda = 2$) data sets. We list the structural similarity index (SSIM), the root-mean-square error (RMSE) and the GPU time (t) in *ms* for each data set.