Spatio-Temporal Sampling of Dynamic Environment Sequences

Liang Wan, *Member, IEEE,* Shue-Kwan Mak,
Tien-Tsin Wong, *Member, IEEE,* and Chi-Sing Leung, *Member, IEEE*

Abstract—Environment sampling is a popular technique for rendering scenes with distant environment illumination. However, the temporal consistency of animations synthesized under dynamic environment sequences has not been fully studied. This paper addresses this problem and proposes a novel method, namely spatio-temporal sampling, to fully exploit both the temporal and spatial coherence of environment sequences. Our method treats an environment sequence as a spatio-temporal volume and samples the sequence by stratifying the volume adaptively. For this purpose, we first present a new metric to measure the importance of each stratified volume. A stratification algorithm is then proposed to adaptively suppress the abrupt temporal and spatial changes in the generated sampling patterns. The proposed method is able to automatically adjust the number of samples for each environment frame and produce temporally coherent sampling patterns. Comparative experiments demonstrate the capability of our method to produce smooth and consistent animations under dynamic environment sequences.

Index Terms—Spatio-temporal sampling, dynamic environment sequences, temporal consistency, importance metric, adaptive volume stratification

1 Introduction

Realistic rendering with real-world illumination [1], [2], [3] is a challenge in computer graphics. A brute-force way is to integrate the contributions of all incoming radiances to each surface element, in which an HDR environment map [4] is commonly used to capture the natural distant illumination. However, the large amount of light sources in a high-resolution environment map makes the radiance computation intractable. To simplify the computation, we can approximate an environment map with a finite number of directional light sources by using the environment sampling techniques [5], [6], [7], [8], [9], [10]. As a practical solution to production pipelines, it is possible to perform sophisticated sampling algorithms during preprocessing and then to apply different environment lighting in rendering. In addition, environment sampling can greatly save the storage space, especially when dynamic environment sequences are used for illumination.

Existing environment sampling techniques can be roughly classified into two categories, depending on whether the environment is static (i.e. a single environment map) [5], [6], [7], [11] or dynamic (i.e. a dynamic environment sequence) [9], [10]. Although most static sampling methods can achieve high rendering quality, directly applying them to sample dynamic environment sequences in a frame-by-frame manner may cause severe flickering artifacts in the rendered animations. In our previous work [9], we proposed spherical q^2-tree, whose adaptive nature suppresses the temporal inconsistency to some extent. Although it utilizes the temporal coherence of consecutive frames, the spherical q^2-tree samples each frame independently. Havran et al. [10] developed a postprocessing method to improve the temporal consistency, in which temporal filtering is applied to the power and position of light samples. Their method is effective to alleviate fluctuations in the environment lighting, while the rendered results might not conform to the ground truth.

In this paper, we propose spatio-temporal sampling, a novel method to simultaneously exploit the temporal and spatial coherence of dynamic environment sequences. Our method represents an environment sequence as a spatio-temporal volume, in which consecutive frames are stacked up along the temporal dimension. Environment sampling is then performed through stratifying the spatio-temporal volume, with more subdivisions in more important regions. To measure the importance of a stratified volume, we present a volumetric importance metric that considers both the illumination and the temporal duration of the volume. Based on this metric, we perform stratification not only spatially but also temporally. This is achieved by an adaptive stratification scheme that employs a quadtree splitting in the spatial dimensions and a binary tree splitting in the temporal dimension. Additionally, we present a subdivision cost, according

- *L. Wan is with School of Computer Software, Tianjin University; the Department of Electronic Engineering, City University of Hong Kong; and was with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. E-mail: lwan@ieee.org*
- *S.-K. Mak and C.-S. Leung are with City University of Hong Kong. E-mail: s.k.mak@student.,eeleungc@cityu.edu.hk*
- *T.-T. Wong is with The Chinese University of Hong Kong. E-mail: ttwong@cse.cuhk.edu.hk*

to which we select a particular dimension to split.

The proposed method can automatically determine the number of light samples for each environment frame, reflecting the variation of illumination over time. It can also suppress abrupt changes in the generated sampling patterns. As a result, we are able to obtain smooth and temporally coherent rendered animations. Readers may skip to Figures 12 and 13 for two examples demonstrating the effectiveness of our method.

The rest of the paper is organized as follows. Section 2 briefly reviews related work. Section 3 presents the way of constructing the spatio-temporal volume from an environment sequence. In Section 4, we describe our spatio-temporal sampling method in detail. Experimental results are reported in Section 5, followed by conclusions in Section 6.

2 Related Work

Consider the reflected radiance at point x on an object:

$$I(x,\vec{s}) = \int_{\Omega} L_{in}(\vec{\omega})\rho(x,\vec{\omega},\vec{s})v(x,\vec{\omega})(\vec{\omega}\cdot\vec{n})d\vec{\omega}, \qquad (1)$$

where $\vec{\omega}$, $\vec{s}$ and $\vec{n}$ represent the lighting direction, viewing direction, and surface normal, respectively; L_{in} is the incident illumination; ρ is the BRDF function; and v is the binary visibility function. It is apparent that when an environment map is used for illumination, computing the above integral by brute force is very time consuming.

To improve the computing efficiency, several effective techniques, including sampling of the BRDF, sampling of the environment lighting, and sampling of the product of both, have been proposed. The BRDF importance sampling [12], [13], [14] performs better for high-frequency BRDFs but low-frequency illuminations. When the environment map contains high-frequency lighting features, the environment importance sampling performs better in reducing rendering bias. In comparison, the product importance sampling [15], [16], [17], [18], [19] achieves the best rendering quality for various BRDFs and illuminations. Note also that it could be reduced to BRDF sampling or environment sampling. In the following, we review the existing techniques that can be used for environment sampling.

2.1 Sampling from Static Environment Maps

Gibson and Murta [20] generated light samples from an environment map by minimizing rendering errors in shadows. Their method requires a reference image of shadows cast by the environment map. Kollig and Keller [6] developed an iterative method. In each iteration, a new sample is inserted near the sample with the maximum intensity, and all samples are refined according to the associated Voronoi tessellation.

Agarwal et al. [5] thresholded an environment map into separate regions, and then assigned each region a different number of samples based on an importance metric. Afterwards, the regions are hierarchically stratified into disjointed strata, with one sample centered at each stratum. Ostromoukhov et al. [7] proposed a fast sampling algorithm that generates light samples by constructing a Penrose tiling structure on top of an environment map. Devebec [8] developed a median cut algorithm to partition an environment map hierarchically into rectangular regions with similar energies. Annen et al. [21] approximated an environment map with a set of rectangular area lights. Differing from other methods, their approach may generate overlapping area lights.

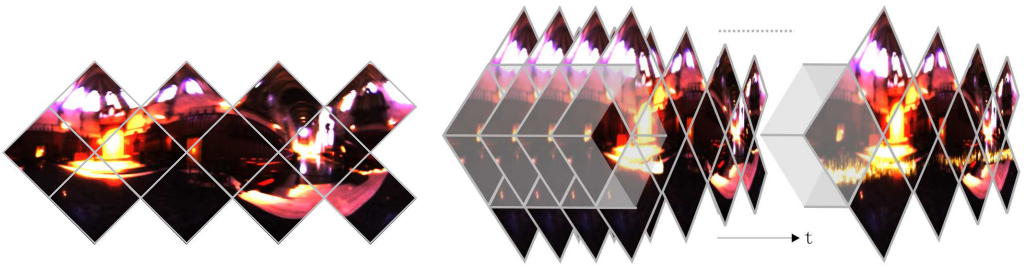
Clarberg et al. [16] presented a technique for product sampling using wavelets, which can generate high quality sampling patterns. Their method hierarchically warps an input point set to match the distribution of the product of the BRDF function and the environment lighting. Subsequently, they proposed a more practical method [18], which avoids using precomputed BRDF function by sampling it on-the-fly. Jarosz et al. [19] further extended the hierarchical sample warping for importance sampling functions represented as spherical harmonics.

2.2 Sampling from Dynamic Environment Sequences

For dynamic environment sequences, sampling each frame separately is prone to flickering artifacts in the rendered animations. In our previous work [9], we addressed the temporal consistency problem by building an adaptive spherical q^2-tree. The temporal coherence of consecutive frames is used to generate samples rapidly. Due to its local adaptive nature, the spherical q^2-tree can suppress abrupt changes in the generated sampling patterns to some extent. Although it makes use of the coherence of environment sequences, it still relies on the strategy of sampling the sequence frame by frame.

Havran et al. [10] proposed an efficient algorithm for sampling hemispherical environment sequences. For each frame, samples are distributed according to the associated goniometric diagram mapping from a unit square to the hemisphere. Temporal filtering by an FIR filter is then applied to the power and position of the samples within a time sliding window. Such temporal filtering can reduce the fluctuations in the environment lighting, e.g. when a sequence is captured in an interior with fluorescent lighting. However, by redistributing the light energy, the rendered animations might not conform to the ground truth. Note also that their method handles hemispherical environment maps and is not directly applicable for spherical environment sequences.

Ghosh et al. [17] introduced a sequential sampling



(a) (b)

Fig. 1. Spatio-temporal volume construction: (a) shows the unrolled view of one environment frame represented in the HEALPix format, which consists of 12 base patches. (b) by stacking up the frames of an environment sequence in the chronological order, we obtain 12 base volumes. Here one base volume is visualized in light gray for illustration.

approach for efficiently sampling the product of dynamic environment lighting and BRDFs. They generated samples for the first frame of the rendered animation by bidirectional importance sampling. The samples are then propagated to the following frames using sequential importance sampling. They further employed a Markov Chain Monte Carlo transition kernel to redistribute the samples according to the product distribution at each frame. This method does not require the knowledge of future environment frames and hence can deal with dynamically captured environment data. Another work aiming to decrease temporal flickers is from Hasan et al. [22]. They clustered lights in a tensor defined over lights, rendered pixels, and the indices of the rendered frames. In contrast, our spatio-temporal volume is composed of environment lighting only, which leads to a simple sampling method.

Note that most aforementioned methods consider the temporal coherence either between two consecutive frames or among a block of continuous frames. In our work, we generate light samples using all the frames in a dynamic environment sequence. We deem that dynamic environment data is to be used in offline rendering so that exploiting the spatio-temporal coherence globally is possible and practical. Although our spatio-temporal sampling cannot be implemented in real time, it maintains sampling consistency better than the previous methods.

3 Spatio-Temporal Volume Construction

Figure 1 illustrates the construction process of a spatio-temporal volume. Given a dynamic environment sequence, we stack up all the frames in the chronological order. As a result, each voxel, i.e. the smallest unit in a volume, corresponds to a unique pixel of one frame. Therefore, both temporal and spatial information of the sequence are embedded in the volume. For simplicity, we use the term, time domain, to refer to the set of frame indices, and spatial domain to denote the image space. This volumetric representation allows us to utilize both the temporal and spatial coherence of the entire environment sequence simultaneously.

In our construction process, we have an underlying assumption that the spherical environment is represented in a rectangular structure. Following our previous work [9], we adopt an equal-area spherical mapping, namely HEALPix [23], because it effectively maps a sphere to a rectangular structure, owns high sampling uniformity and introduces less distortion in the mapping. Figure 1(a) shows the HEALPix map for one environment frame. Since the HEALPix partitions a sphere into 12 base patches, stacking an environment sequence yields 12 base volumes, as shown in Figure 1(b).

4 Spatio-Temporal Sampling

After constructing spatio-temporal volumes, our goal is to perform sampling in the volumes such that the generated samples can be used to render high-quality animations. We consider the rendering quality from two aspects: the rendering accuracy of each rendered frame, and the temporal consistency of the rendered animation. Note that in a practical sense, we are more likely to notice abrupt temporal changes (i.e. flickers) when we have no prior knowledge of the ground truth. Intuitively, a decent sampling algorithm has to meet three requirements: (1) for each environment frame, the important regions should be assigned with more samples; (2) more samples should be given to the frames with higher importances; and (3) the generated sampling patterns should change smoothly.

Based on these concerns, we develop a spatio-temporal sampling algorithm that stratifies the spatio-temporal volumes both spatially and temporally. Its general framework is as follows:

1. For a dynamic environment sequence, construct the 12 base volumes at initialization.
2. Evaluate the importances for all the volumes and choose the one with the highest importance.

3. For the selected volume, determine which domain is to be split and perform an appropriate splitting. The splitting partitions a volume into several non-overlapping volumes.
4. Define a temporal slice of each volume as a stratum. If the average number of strata for each frame does not reach a pre-defined value, return to step 2.
5. For each volume, place a sample at the centroid of each stratum. Hence, a resulted volume consisting of T temporal slices will have T samples in total.

This framework can suppress the abrupt changes in the generated sampling patterns due to its adaptive nature. Its performance is affected by two factors, the importance metric of a volume and the splitting scheme. The importance metric determines which regions should be assigned with more samples, and the splitting scheme controls the coherence in the generated sampling patterns. We will discuss these two issues in the following subsections.

4.1 Importance Metric for Spatio-Temporal Volumes

In importance sampling, we would like to assign more samples at brighter regions and fewer samples at darker regions. Note that the regions are not restricted to one environment frame, but may come from different frames.

Let us begin by an example. Suppose there are two spatio-temporal volumes, V_1 and V_2, as shown in Figure 2. We assume that they span the same solid angle in the spatial domain and have uniform intensities. Let us first consider the strata of two volumes at time $t = 0$. Since the two strata belong to one environment frame, if the stratum S_1^0 in V_1 is more important than the stratum S_2^0 in V_2, more samples should be assigned to S_1^0. Due to the uniform intensity assumption, an intermediate stratum S_1^t in V_1 is also more important than any intermediate stratum S_2^t in V_2, although they may be from two different frames. As a result, more samples should be given to each stratum in V_1. Note that this analysis always holds no matter which volume has more temporal slices. In other words, the sampling depends more on the regional importance of each stratum (i.e. the importance of a 2D region) than the temporal duration of the volume.

Following this idea, we define the volumetric importance metric as the average of the regional importances of strata in a volume, given by

$$\Gamma_v = \frac{1}{T}\sum_t \Gamma(t), \tag{2}$$

where $\Gamma(t)$ is the regional importance of the stratum at time t and T is the number of strata in the volume. Particularly, when $T = 1$, the volumetric importance metric Γ_v reduces to a regional importance metric. Our metric can cope with the existing regional importance metrics [6], [8], [5], [24]. In our work, we adopt the metric proposed by Agarwal et al. [5]. By combining the stratified sampling and illumination-based importance sampling, they described the importance metric for a 2D region Ω as

$$\Gamma = L^a \Delta\omega^b, \tag{3}$$

where $\Delta\omega$ is the solid angle of Ω, and L is the integrated illumination, given by

$$L = \int_\Omega L_{in}(\vec{\omega}) d\vec{\omega}. \tag{4}$$

A large value of parameter a favors the illumination component and a large value of b favors the area component. The hybrid metric avoids oversampling small bright regions, while for equal-area regions it is equivalent to the integrated illumination. In this paper, we adopt the setting $a = 1$ and $b = 1/4$.

By substituting (3) into (2), our volumetric importance metric becomes

$$\Gamma_v = \frac{1}{T}\sum_t [L(t)]^a \, \Delta\omega^b, \tag{5}$$

where $L(t)$ is the integrated illumination of the stratum at time t. The proposed importance metric not only considers the illumination and region area of a volume but also eliminates the impact of the temporal duration of the volume. We can further speed up its computation by using the summed area table [25].

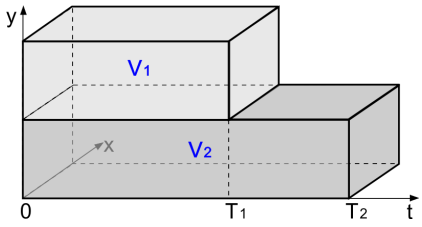


Fig. 2. Estimation of the importances of spatio-temporal volumes.

4.2 Adaptive Volume Stratification

With the proposed importance metric, we now describe our hybrid tree structure for volume stratification. It applies a binary tree in the time domain and a quadtree in the spatial domain. This hybrid tree structure has the advantages of both *simplicity* and *stability* of the quadtree in 2D splitting and the binary tree in 1D splitting, respectively. Using a quadtree in the spatial domain guarantees a low-discrepancy sampling pattern when the environment lighting is an image of constant radiances [9]. On the other hand, applying a binary tree in the time domain helps to capture apparent changes between consecutive environment frames.

Input: A spatio-temporal volume.
Output: Several non-overlapping sub-volumes.

1. Compute the splitting cost of subdividing the volume along one dimension, getting C_x, C_y and C_t;
2. If $2C_t > C_x + C_y$, split the volume using a binary tree in the time domain;
3. Otherwise, split the volume using a quadtree in the spatial domain.

Algorithm 1: Splitting a spatio-temporal volume.

Specifically, for a given volume, we first determine which domain is to be split (explained later), and then either split the volume at the center in the time domain (which results in two sub-volumes), or split it equally along the two spatial dimensions (which results in four sub-volumes). Figure 3 illustrates the two splitting schemes. Recall that each stratum in a resulted volume is collapsed to one light sample. Therefore, after the temporal splitting the total number of samples in the volume remains unchanged while the spatial splitting increases the total number of samples four times.

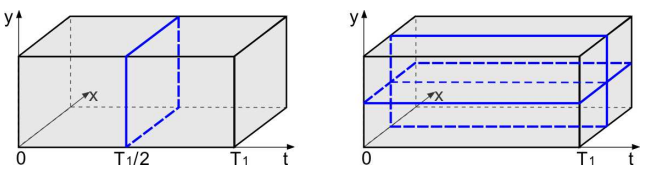


Fig. 3. A volume can be split (a) in the time domain which results in two sub-volumes, or (b) in the spatial domain which results in four sub-volumes.

Note that the splitting should capture important changes in the illumination, including temporal changes between successive frames and spatial changes between nearby regions in the spatial domain. Hence, we select the domain that yields a larger difference between the importances of the resulted sub-volumes. For this purpose, we define a splitting cost. Suppose the volume is split along one dimension to get two sub-volumes, v_1 and v_2. The splitting cost is given by,

$$C = |\Gamma_{v1} - \Gamma_{v2}|. \tag{6}$$

Let the cost of the temporal splitting to be C_t, and the costs of splitting along one spatial dimension only are C_x and C_y, respectively. Since we always split a volume in the spatial domain with two subdivisions, the temporal splitting is performed if and only if the following condition holds,

$$2C_t > C_x + C_y. \tag{7}$$

The steps of splitting a volume are summarized in Algorithm 1.

At the end, we illustrate how the spatio-temporal sampling works using a simple example. Figure 4(a) shows an input volume. It is composed of three parts, each with constant intensities (light gray colors refer to high intensities). At the first subdivision, since the cost of the temporal splitting is much larger than the costs of the spatial splittings, the volume is split with a binary tree in the time domain (Figure 4(b)). The resulted sub-volume on the right part owns a higher importance and is selected at the second subdivision. At this time, the cost of the temporal splitting is zero, so the quadtree-based stratification is applied in the spatial domain (Figure 4(c)). Figure 4(d) shows the generated sampling patterns. As demonstrated in this example, the hybrid volume stratification can capture both the abrupt illumination changes between successive frames (Figure 4(b)) and the important changes in the spatial domain (Figure 4(c)).

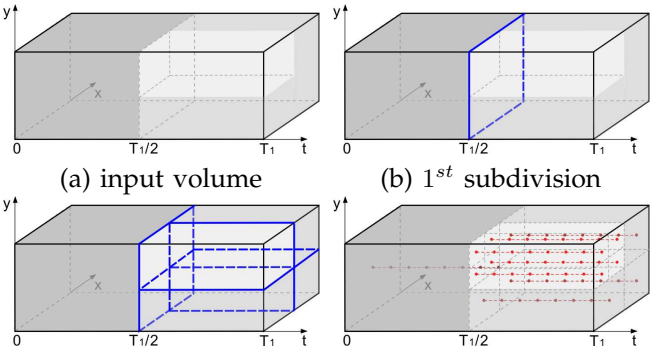


Fig. 4. Illustration of spatio-temporal sampling. (a) is an input volume. Light gray color indicates high intensity. (b) and (c) show the results of the first and second subdivisions, respectively. (d) gives the possible generated sampling patterns.

5 Experiments and Discussion

We first introduce a metric for measuring the temporal inconsistency of an importance sampling method. Next, the proposed spatio-temporal sampling is compared with several existing methods, which is followed by a discussion of our method.

5.1 Temporal Inconsistency Metric

We define a temporal inconsistency metric to quantitatively measure how similar the rendering variation of a sampling method is to that of the control results globally. The metric is computed as the mean absolute error between the temporal differences of the rendered results from a sampling method and those of the control results, which is given by,

$$E(t) = \frac{1}{N}\sum_i w_i |\triangle X_i(t) - \triangle \hat{X}_i(t)|, \tag{8}$$

where $X_i(t)$ is the intensity of pixel i in the t-th rendered frame from a sampling method; $\hat{X}_i(t)$ is the counterpart of $X_i(t)$ in the control result; and N is the total number of pixels of one rendered frame. The operator $\triangle(\cdot)$ computes the temporal difference between two consecutive rendered frames, given by $\triangle X_i(t) = X_i(t) - X_{i+1}(t)$. In addition, we can take the rendering accuracy into account and define the weighting factor $w_i = |X_i(t) - \hat{X}_i(t)| + 1$. It means that the higher the difference between the rendered result and the control image, the larger the inconsistency will be. Note that the metric is evaluated over the whole rendered image although the variations in the shadows are more visually obvious than other regions. Based on this metric, a sampling method is said to produce more temporally consistent results if its inconsistency values are closer to zero.

5.2 Experimental Results

We consider two dynamic environment sequences, including a synthetic sequence and a real captured sequence. The synthetic data "grace_flame" contains 120 frames. It was created by blending a synthetic HDR flame sequence with the "grace" environment map [3]. To obtain the real data, we built an HDR panorama video capturing system based on Ladybug2, which is a spherical digital video camera [26]. We captured the "reading_room" sequence (consisting of 160 frames) in a large reading room where two hand-held spotlights were turned on-and-off during rotation.

Our approach is compared with several state-of-the-art methods, including

- Spherical q^2-tree [9].
- Structured importance sampling [5]. Note that it relies on the thresholds that may vary frame to frame, which may cause severe flickers [9]. In our implementation, we fixed the thresholds for each environment sequence.
- Hierarchical sample warping (HSW) [16]. For this method, we set the BRDF function to be a constant and sampled environment sequences in both framewise and volumetric manners. The framewise HSW (named HSW2D) samples each environment frame independently by using the same 2D Hammersley point sequence as input. We also implemented HSW in 3-dimensions (named HSW3D) and sampled the environment volume as a whole. Considering temporal coherence, we prepared the input random point set by generating a 3D Hammersley point sequence for one frame (defined within a limited time range), and duplicating it for the remaining frames. We then constructed a 3D mipmap from the environment volume and distributed samples within the volume using HSW. Finally, the resulted samples are quantized into frames.
- Importance sampling using temporal filtering [10]. Note that the method proposed by Havran et al. [10] handles hemispherical environment data and is not directly applicable for spherical environment sequences. To make a proper comparison, we extended their method to spherical domain. We first generated uniformly distributed samples on the sphere, and then warped the samples by using the environment map as a PDF function. During the temporal filtering, a FIR filter (using the parameters provided by the authors) is applied in a symmetric manner to both sample energy and position.

In our experiments, we set the average number of samples for each environment frame to be 300. In other words, except for our method and HSW3D, all other methods will produce 300 samples for each frame. In addition, we generate control images by using $10K$ uniformly distributed light samples drawn from each environment frame.

5.2.1 Synthetic Environment Sequence

The first experiment uses the "grace_flame" sequence for illumination. After the environment sampling, an animation is rendered by illuminating a girl model using light samples of each environment frame.

Figure 5 plots the temporal inconsistency curves for the six sampling methods. We can see that our spatio-temporal sampling has minimum inconsistency values on average. In other words, it produces a more temporally consistent rendered animation. On the other hand, the structured importance sampling and HSW2D have larger inconsistency values, since they do not consider the temporal coherence between frames during the sampling. The HSW3D, although sampling the environment sequence in a volumetric manner, still suffers apparent inconsistency. By observing the sampling patterns (as shown in the supplementary video), we find that it can produce stable samples in static regions, but introduce jumping samples in regions with dynamic illumination changes. One possible reason is that by building a 3D-tree over the environment volume, the HSW3D may perform different warps for successive frames in these regions. Importance sampling with temporal filtering also has poor consistency. It is probably because the FIR filter adopted is not suitable for environment sequences with large dynamic changes (designing a suitable FIR filter for arbitrary environment sequence is out of the scope of this paper). Furthermore, FIR filtering may introduce bias in sample energy and positions, and hence the rendered results may not conform to the control images. The spherical q^2-tree has smaller inconsistency values in most frames, but it may still have severe inconsistency sometimes. This may be because the sample number is not adequate to capture the important illumination in some frames.

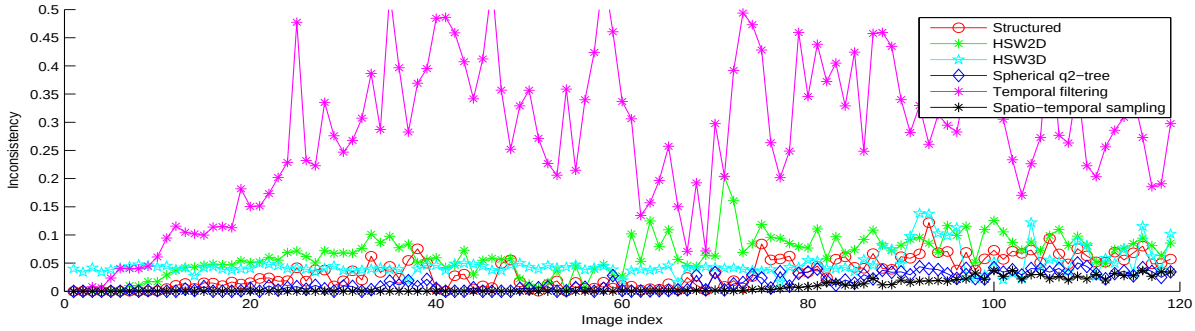


Fig. 5. For the "grace_flame" environment sequence, the temporal inconsistency is measured with respect to the image index. To clearly illustrate the differences between the testing methods, we clip the curve of importance sampling with temporal filtering at some peaks. Among the six sampling methods, our spatio-temporal sampling produces a more temporally consistent result.

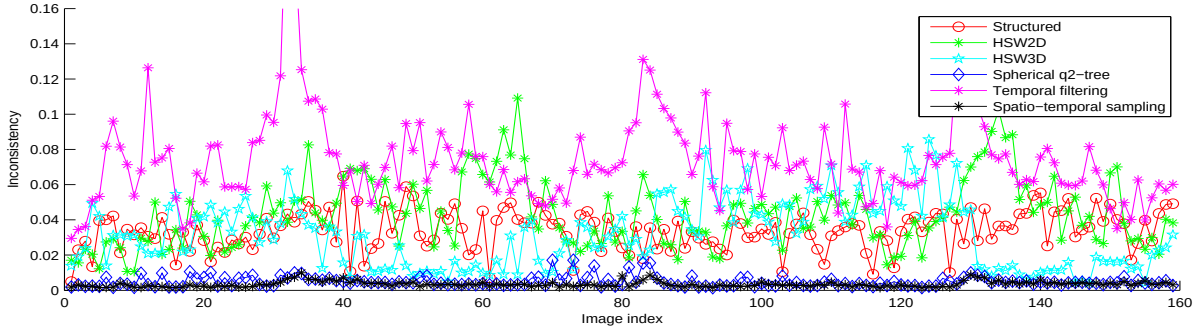


Fig. 6. The temporal inconsistency of different methods for the "reading_room" environment sequence. To clearly illustrate the differences between the testing methods, we clip the curve of importance sampling with temporal filtering at some peaks. In comparison, the spatio-temporal sampling has smaller inconsistency values on average.

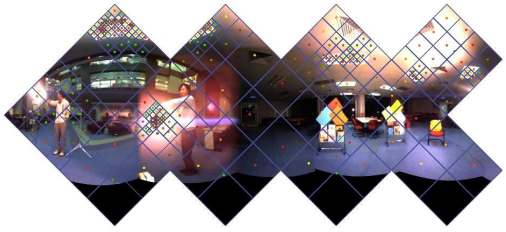
We further compare visual quality of rendered images in Figure 12. The fourth and fifth columns visualize the differences between frames 92 & 93 and frames 93 & 94, respectively. As shown, the spherical q^2-tree, structured method, HSW2D, HSW3D, and importance sampling with temporal filtering have more obvious jumps in the shadows behind the girl. Although our spatio-temporal sampling still has flickers compared to the control results, it is much better than other sampling methods. [Readers are referred to the companion video for a clearer comparison as static pictures may not be obvious to illustrate the temporal inconsistency.]

5.2.2 *Real Captured Environment Sequence*

In the second experiment, we employ the Ward BRDF model [27] and use the "reading_room" sequence to illuminate a set of table and chair models.

Figure 13 compares four consecutive rendered frames for the six methods. The rightmost three columns show the blow-ups of the differences between every two successive rendered frames. The difference images are enhanced for a clearer comparison. It is obvious that the proposed spatio-temporal sampling generates the most consistent results with respect to the control images, while other methods introduce more or less severe changes in the rendering. The corresponding PSNR value and the sample number for each frame are listed in the table shown in Figure 7. The spatio-temporal sampling and HSW3D use less than 300 (valid) samples for the four frames, while other methods are restricted to 300 samples per frame. Note that because of duplicating the input point set for all the environment frames, the HSW3D may generate samples overlapping each other at some frames, which actually leads to a waste. In this example, most methods have similar PSNR values that are higher than importance sampling with temporal filtering. Although the HSW2D method has the highest PSNR values for the four frames, it suffers apparent temporal inconsistency. This validates our claim that the rendering quality should consider both the rendering accuracy and the temporal consistency.

Next, we quantitatively measure the temporal inconsistency of the rendered animations (shown in Figure 6). The structured method, HSW2D, and importance sampling with temporal filtering have large and continuously varying inconsistency values. The



(a)

PSNR (dB) & Sample number	Frame 83	Frame 84	Frame 85	Frame 86
Structured	40.51	41.11	41.50	41.97
HSW2D	43.43	42.83	43.53	43.89
HSW3D	39.97 333/256	39.48 302/247	39.26 266/240	42.12 243/228
Temporal filtering	39.77	35.12	33.07	32.31
Spherical q2-tree	41.17	41.72	41.99	42.65
Spatio-temporal	41.14 294	41.39 291	41.91 276	42.46 267

(b)

Fig. 7. (a) Sampling pattern of frame 84 in the "reading_room" sequence. (b) PSNR values and the sample numbers for the four consecutive frames, 83-86. For the HSW3D method, the two sample numbers per frame correspond to the assigned sample number and the valid sample number, respectively. The methods with no sample number specified all use 300 samples for each frame.

HSW3D exhibits better consistency for frames with more samples assigned (refer to the supplementary video). Compared to the spherical q^2-tree, the spatio-temporal sampling has better consistency in most parts of the animation. Referring to the animations, we also observe that the structured method, HSW2D and importance sampling with temporal filtering suffer continuous flickering; the HSW3D shows coherence in two parts of animations; the spherical q^2-tree has apparent jumps at sometimes; our spatio-temporal sampling has a temporally coherent animation, much like the control result.

Besides the temporal inconsistency, we are also concerned about the rendering accuracy. The rendering accuracy is measured using the root mean square error (RMSE) between the rendered results and the control images. As shown in Figure 8, both the spherical q^2-tree and spatio-temporal sampling have RMSE curves lower than other methods, while the spatio-temporal sampling has less variance in the achieved rendering quality than the spherical q^2-tree. We also notice that there are apparent variations in RMSE values for all the six methods, which is due to the continuous slight flickering of the lamps in the reading room.

5.3 Discussion

Different sampling methods represent the spherical environment in different formats. Specifically, the structured method, HSW2D, HSW3D and importance sampling with temporal filtering use the longitude-latitude format in a resolution of 1024×512. The spherical q^2-tree and spatio-temporal sampling adopts the HEALPix format in a resolution of $256 \times 256 \times 12$. All our experiments are performed on a PC with AMD Athlon(tm) 64 Dual 2.01 GHz. For the "reading_room" environment sequence, our unoptimized code takes an average time of 0.8 seconds per frame. On the same machine, the spherical q^2-tree takes around 0.68 seconds for each frame.

Since the proposed method considers the environment sequence as a 3-dimensional volume, the generated sampling patterns may be affected by the sequence length. We conduct one experiment to test the robustness of our method. In this experiment, we extract a clip from the "reading_room" sequence, e.g. the first 120 frames, for illuminating the 3D scene used in the second experiment. We also set the average number of samples for each frame to be 300. Figure 9 compares the RMSE curves of the clipped sequence and the original sequence. Since the sample number for each frame of the clipped sequence may change compared to the counterpart of the original sequence,

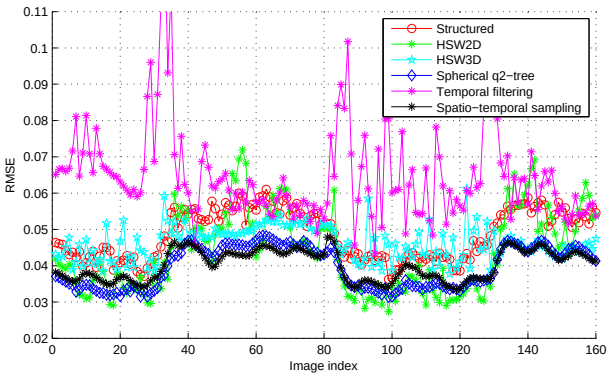


Fig. 8. The rendering accuracy of different sampling methods for the "reading_room" environment sequence. Note that the spatio-temporal sampling achieves small RMSE values on average and is more stable than other methods.

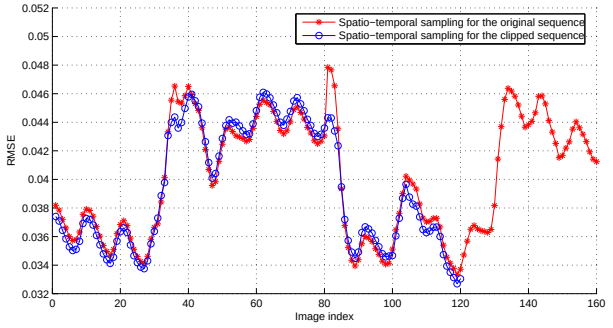


Fig. 9. Robustness of the spatio-temporal sampling method. As shown, the RMSE curve of the clipped "reading_room" sequence is similar to the corresponding partial curve of the entire sequence.

the RMSE values may not be exactly identical in the two cases. However, as shown in Figure 9, the two RMSE curves are similar in the corresponding parts. This tells us that the spatio-temporal sampling is somehow robust to the sequence length.

Note also that our spatio-temporal sampling can adjust the number of samples for different environment frames automatically (see Figure 10). By assigning more samples to more important frames, we are able to achieve higher rendering quality. To verify this point, we compare the proposed method with a modified version that is restricted to have the same number of samples for each frame. As shown in Figure 11, the spatio-temporal sampling results in much smaller consistency values, while the modified sampling method leads to large fluctuations.

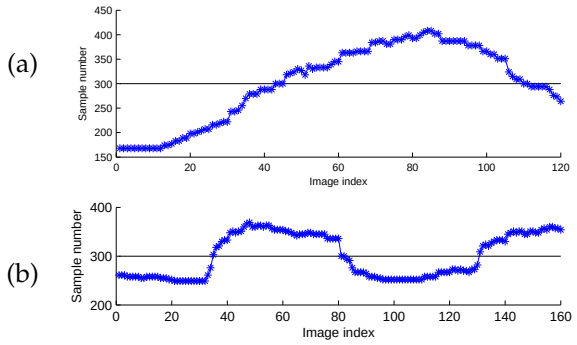


Fig. 10. Sample number for each frame from the proposed method. (a) "grace_flame" environment sequence. (b) "reading_room" environment sequence.

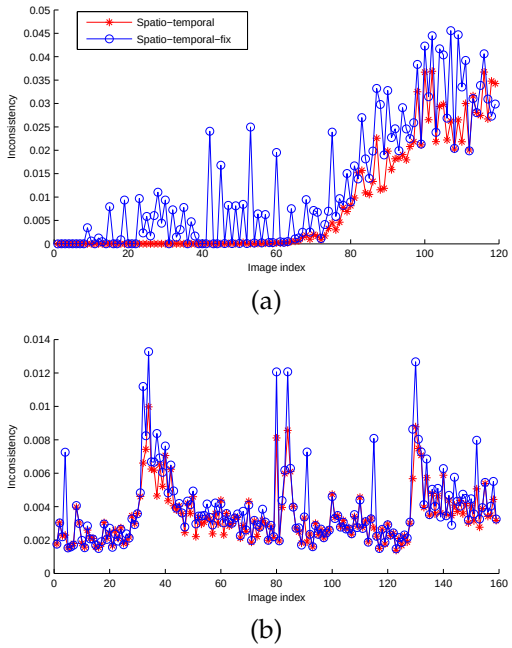


Fig. 11. Comparison between the spatio-temporal sampling method ("spatio-temporal") and a modified version ("spatio-temporal-fix") that is restricted to have the same number of samples for each environment frame. (a) shows the inconsistency curves for the "grace_flame" sequence. (b) shows the inconsistency curves for the "reading_room" sequence. Note that using a different amount of samples for different frames allows for higher consistency.

6 Conclusion

This paper addresses the temporal consistency problem in importance sampling of dynamic environment sequences. We propose a new approach, spatio-temporal sampling, which treats the environment sequence as a spatio-temporal volume and stratifies the volume adaptively. Unlike most of previous methods, our approach can simultaneously exploit the temporal and spatial coherence of the sequence during the sampling. Our main contributions are a volumetric importance metric and a hybrid tree-structured volume stratification scheme. The proposed approach can automatically adjust the sample number for each frame and generate temporally consistent sampling patterns. As evidenced by the experiments, our method produces consistent rendered animations in terms of both temporal inconsistency values and visual quality.

Currently, we stratify the spatio-temporal volume into rectangular sub-volumes, which may cause a waste of samples in some low-energy regions. Hence one future work is to explore a feasible way to subdivide the spatio-temporal volume into irregular regions. This may increase the sampling efficiency and further improve the rendering quality. For this problem, a possible solution is to study the application of state-of-the-art segmentation methods [28].

Acknowledgment

This work was supported by General Research Funds from Hong Kong Government (Project No.: CityU 116508) and (Project No. CUHK 417107).

References

[1] J. F. Blinn and M. E. Newell, "Texture and reflection in computer generated images," *Communications of the ACM*, vol. 19, no. 10, pp. 542–546, October 1976.

[2] N. Greene, "Environment mapping and other applications of world projections," *IEEE Computer Graphics and Applications*, vol. 6, no. 11, pp. 21–29, November 1986.

[3] P. Debevec, "Rendering synthetic objects into real scenes: bridging traditional and image-based graphics with global illumination and high dynamic range photography," in *SIGGRAPH '98*, 1998, pp. 189–198.

[4] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *SIGGRAPH'97*, 1997, pp. 369–378.

[5] S. Agarwal, R. Ramamoorthi, S. Belongie, and H. W. Jensen, "Structured importance sampling of environment maps," *ACM Transaction on Graphics (SIGGRAPH 2003)*, vol. 22, no. 3, pp. 605–612, 2003.

[6] T. Kollig and A. Keller, "Efficient illumination by high dynamic range images," in *Eurographics Symposium on Rendering: 14th Eurographics Workshop on Rendering*, 2003, pp. 45–51.

[7] V. Ostromoukhov, C. Donohue, and P. M. Jodoin, "Fast hierarchical importance sampling with blue noise properties," *ACM Transaction on Graphics (SIGGRAPH 2004)*, vol. 23, no. 3, pp. 488–495, 2004.

[8] P. Debevec, "A median cut algorithm for light probe sampling," in *Siggraph sketch*, 2005.

[9] L. Wan, T.-T. Wong, and C.-S. Leung, "Spherical Q^2-tree for sampling dynamic environment sequences," in *Proceedings of the Eurographics Symposium on Rendering 2005*, 2005, pp. 21–30.

[10] V. Havran, M. Smyk, G. Krawczyk, K. Myszkowski, and H.-P. Seidel, "Interactive system for dynamic scene lighting using captured video environment maps," in *Eurographics Symposium on Rendering 2005*, 2005, pp. 31–42.

[11] C. B. Madsen, M. K. D. Sørensen, and M. Vittrup, "Estimating positions and radiances of a small number of light sources for real-time image-based lighting," in *Proceedings: Annual Conference of the European Association for Computer Graphics, EUROGRAPHICS 2003*, 2003, pp. 37 – 44.

[12] P. Shirley, "Physically based lighting calculations for computer graphics," Ph.D. dissertation, University of Illinois at Urbana-Champaign, 1990.

[13] M. D. McCool and P. K. Harwood, "Probability trees," in *Proceedings of Graphics Interface '97*, 1997, pp. 37–46.

[14] J. Lawrence, S. Rusinkiewicz, and R. Ramamoorthi, "Efficient brdf importance sampling using a factored representation," *ACM Transaction on Graphics*, vol. 23, no. 3, pp. 496–505, 2004.

[15] D. Burke, A. Ghosh, and W. Heidrich, "Bidirectional importance sampling for direct illumination," in *Proceedings of the Eurographics Symposium on Rendering 2005*, 2005, pp. 147–156.

[16] P. Clarberg, W. Jarosz, T. Akenine-Möller, and H. W. Jensen, "Wavelet importance sampling: efficiently evaluating products of complex functions," in *SIGGRAPH '05*, 2005, pp. 1166–1175.

[17] A. Ghosh, A. Doucet, and W. Heidrich, "Sequential sampling for dynamic environment map illumination," in *Proc. of Eurographics Symposium on Rendering 2006*, 2006, pp. 115–126.

[18] P. Clarberg and T. Akenine-Möller, "Practical product importance sampling for direct illumination," *Computer Graphics Forum (Proceedings of Eurographics 2008)*, vol. 27, no. 2, pp. 681–690, 2008.

[19] W. Jarosz, N. A. Carr, and H. W. Jensen, "Importance sampling spherical harmonics," *Computer Graphics Forum (Proc. Eurographics EG'09)*, vol. 28, no. 2, pp. 577–586, 4 2009.

[20] S. Gibson and A. Murta, "Interactive rendering with real-world illumination," in *Proceedings of the Eurographics Workshop on Rendering Techniques 2000*. London, UK: Springer-Verlag, 2000, pp. 365–376.

[21] T. Annen, Z. Dong, T. Mertens, P. Bekaert, H.-P. Seidel, and J. Kautz, "Real-time, all-frequency shadows in dynamic scenes," *ACM Transaction on Graphics*, vol. 27, no. 3, pp. 1–8, 2008.

[22] M. Hasan, E. Velázquez-Armendáriz, F. Pellacini, and K. Bala, "Tensor clustering for rendering many-light animations," *Computer Graphics Forum*, vol. 27, no. 4, pp. 1105–1114, 2008.

[23] K. M. Gorski, B. D. Wandelt, E. Hivon, F. K. Hansen, and A. J. Banday, "The HEALPix primer," Theoretical Astrophysics Center (TAC) Kopenhagen, Tech. Rep., Feb. 2003, astro-ph/9905275.

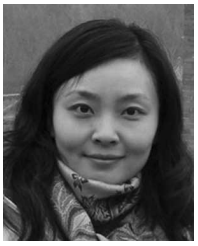
[24] D. Cline, P. K. Egbert, J. Talbot, and D. Cardon, "Two stage importance sampling for direct lighting," in *Eurographics Symposium on Rendering 2006*, 2006, pp. 103–113.

[25] F. C. Crow, "Summed-area tables for texture mapping," in *Proceedings of SIGGRAPH 1984*, 1984, pp. 207–212.

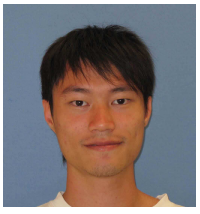
[26] "Ladybug2," Point Grey Research, 2006. [Online]. Available: http://www.ptgrey.com/products/ladybug2/

[27] G. J. Ward, "Measuring and modeling anisotropic reflection," *SIGGRAPH Comput. Graph.*, vol. 26, no. 2, pp. 265–272, 1992.

[28] W. Feng, J. Jia, and Z.-Q. Liu, "Self-validated labeling of markov random fields for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 10, pp. 1871–1887, 2010.



Liang Wan received the B.Eng. and M.Eng. degrees in computer science and engineering from Northwestern Polytechnical University in China, in 2000 and 2003 respectively, and the PhD. degree in computer science from the Chinese University of Hong Kong in 2007. She is currently an Associate Professor in the School of Computer Software, Tianjin University, P.R. China. Her main research interest is computer graphics, including precomputed lighting, non-photorealistic rendering and image processing.



Shue-Kwan Mak received the B.Eng. degree in electronics from City University of Hong Kong in 2009. He is currently a research student in the Department of Electronic Engineering, City University of Hong Kong. His research focuses on Neural Networks and GPU programming.

Tien-Tsin Wong received the B.Sci., M.Phil., and PhD. degrees in computer science from the Chinese University of Hong Kong in 1992, 1994, and 1998, respectively. Currently, he is a Professor in the Department of Computer Science & Engineering, Chinese University of Hong Kong. His main research interest is computer graphics, including computational manga, image-based rendering, GPU techniques, natural phenomena modeling, and multimedia data compression. He received IEEE Transactions on Multimedia Prize Paper Award 2005 and Young Researcher Award 2004.

Chi-Sing Leung received the B.Sci. degree in electronics, the M.Phil. degree in information engineering, and the PhD. degree in computer science from the Chinese University of Hong Kong in 1989, 1991, and 1995, respectively. He is currently an Associate Professor in the Department of Electronic Engineering, City University of Hong Kong. His research interests include neural computing, data mining, and computer graphics. In 2005, he received the 2005 IEEE Transactions on Multimedia Prize Paper Award for his paper titled, "the Plenoptic Illumination Function" published in 2002. He is the Program Chair of ICONIP2009. He is also a governing board member of the Asian Pacific Neural Network Assembly (APNNA).

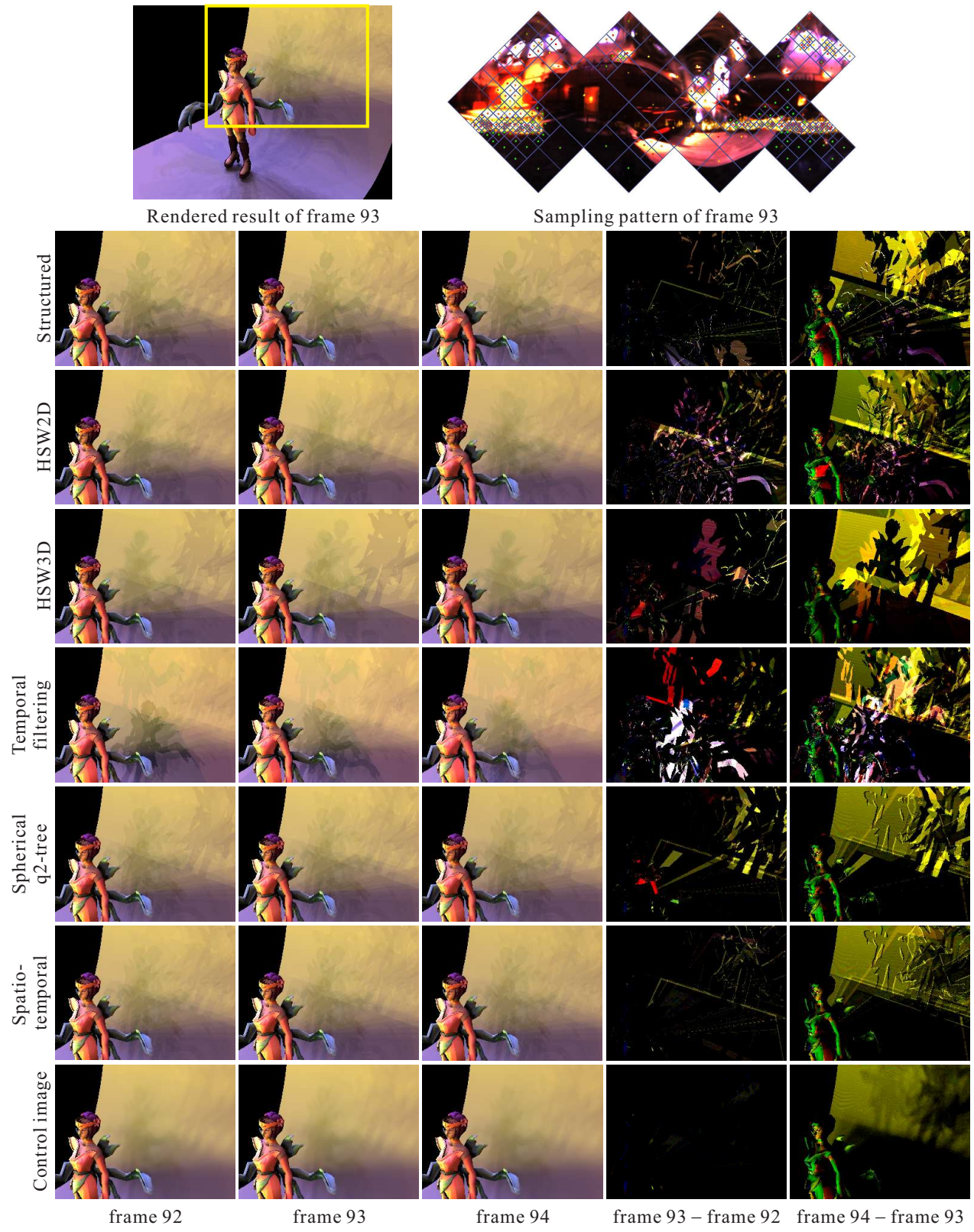


Fig. 12. Comparison of different methods for the “grace_flame” environment sequence. The first row shows results using the spatio-temporal sampling for environment frame 93: the rendered image on the left and the corresponding sampling pattern on the right. From the second row are rendered results and the difference images from different sampling methods and control images. Note the choppy jumps in shadows behind the girl for the first five methods between consecutive animation frames. The proposed spatio-temporal sampling, on the other hand, exhibits small flickering artifacts. *[Readers are referred to the companion video for a clearer comparison of the animated sequences.]*

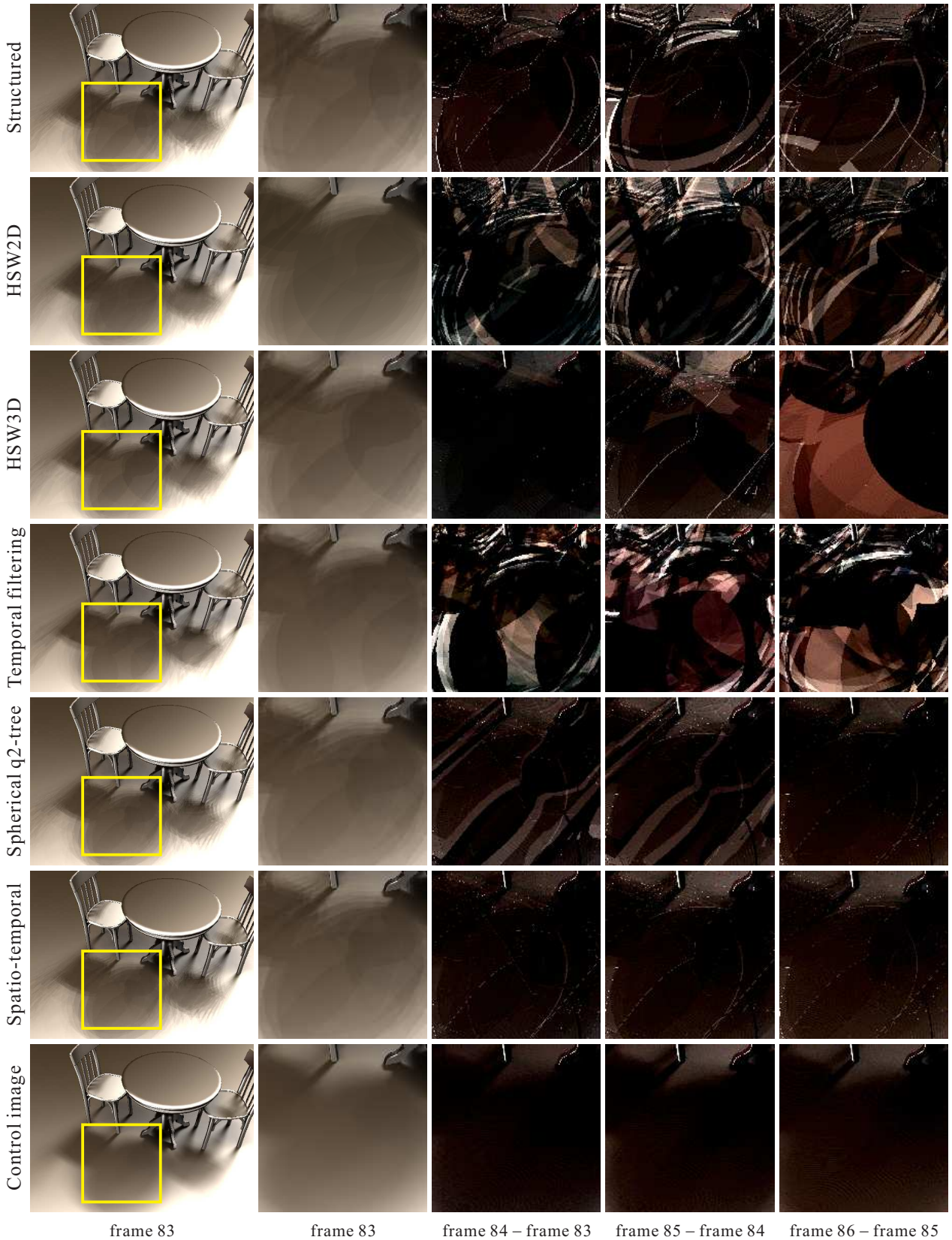


Fig. 13. Comparison of different methods for the “reading_room” environment sequence. Note that the structured method, HSW2D, HSW3D and importance sampling with temporal filtering have obvious changes in the ground shadows between consecutive frames. For the spherical q2-tree, the jumps in the shadows are also apparent. In comparison, the results from the spatio-temporal sampling are more consistent to control results in the last row.