CrossMark

ORIGINAL ARTICLE

Visual importance-based adaptive photon tracing

Quan Zheng[1,2] · Chang-Wen Zheng[1]

Abstract This paper proposes an adaptive photon tracing approach based on a novel importance function, which combines visual importance and photon path visibility. The generation of photon path is guided by sampling this function to trace more photons to visible and more contributive regions. As a first step, a hierarchy of visual importance maps is constructed. Next, photon paths are produced using a new hybrid mutation strategy, which consists of large mutation and small mutation. The mutation parameter used in small mutation is automatically adjusted using the adaptive Markov chain sampling method. Meanwhile, to find a suitable initial parameter, a mutation parameter initialization method is developed. Experiments show that, compared with previous methods, this approach yields results with better visual quality and smaller numerical error.

Keywords Visual importance · Adaptive photon tracing · Photorealistic rendering · Global illumination

1 Introduction

Efficiently rendering scenes with complex lighting configurations is a challenging problem of global illumination. Notable examples of such scenes are a dark room illuminated by indirect illumination through a door slit, or a house lit by sunlight through windows. Most existing methods fail to efficiently solve this problem. Though Monte Carlo ray tracing methods such as path tracing (PT) [1] and bi-directional path tracing (BDPT) [2] provide versatile solutions for solving the rendering equation [1], they are inefficient to sample light-carrying paths in such scenes.

Photon mapping (PM) [3] and its evolutionary versions, progressive photon mapping (PPM) [4] and stochastic progressive photon mapping (SPPM) [5], have been widely used for handling scenes with specular-diffuse-specular (SDS) paths. However, these methods become highly inefficient when rendering scenes with complex lighting settings. Due to occlusion in these scenes, it is hard to trace photons to visible regions, resulting in a slow convergence rate in these regions.

Metropolis photon sampling (MPS) [6] introduces Metropolis-Hastings sampling to photon mapping. But it traces light paths via path tracing, which cannot efficiently sample SDS paths. Chen et al. [7] devise an importance function based on initial photon density, increasing photon density in regions where initial photon density is low. Robust adaptive photon tracing method (RAPT) [8] gives an importance function based on photon path visibility, which enables the method to trace photons to visible regions. Unfortunately, RAPT cannot distinguish the contribution of different regions. It equally distributes computational efforts to all visible regions. Excessive computation in regions which are visible but less contributive to the final image costs much time but receives little benefit.

Motivated by these observations, we propose a visual importance-based adaptive photon tracing approach (VIAPT) to solve the aforementioned problems. A new importance function based on visual importance and photon path visibility is designed and it is set as the target distribution of photon paths sampling. Due to the importance sampling on

✉ Quan Zheng
zhengquan@acm.org; qzhengblif@163.com

Chang-Wen Zheng
cwzheng@ieee.org

[1] Science and Technology on Integrated Information System Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing, China

[2] University of Chinese Academy of Sciences, Beijing, China

the importance function, photons are adaptively guided to visible and more contributive regions, effectively accelerating the convergence rate of rendering. When generating photon paths, a hybrid mutation strategy combining large mutation and small mutation is implemented to take advantage of existing paths which arrive at highly contributive regions. In addition, a new mutation parameter initialization method based on tempered inter-chain adaptation is introduced to find a suitable starting parameter for small mutation. Experiments using various illumination configurations show that VIAPT can produce images with higher quality and lower error than existing adaptive photon tracing methods.

2 Related work

Scenes with complex lighting settings pose a tough challenge to existing rendering methods used for solving the rendering equations [1]. These methods can be divided into two categories: unbiased Monte Carlo ray tracing and particle tracing.

A seminal method of the first category is the metropolis light transport [9], which applies Markov chain Monte Carlo to rendering. MLT generates new paths from mutations of high radiance paths using several mutation kernels. Kelemen et al. [10] give a symmetric mutation kernel to perform mutation in the primary sample space. But the kernel parameter is set to a fixed value, which cannot adapt to different scenes. Hoberock and Hart [11] propose to dynamically update importance function based on image noise, demonstrating that the importance function can be crafted. Unfortunately, the above methods are inefficient to sample SDS paths from small light sources, since SDS paths are usually sampled with extremely low probability by path sampling. In recent years, more variants have been proposed to improve MLT, such as replica exchange Monte Carlo [12] and gradient-domain MLT [13].

To handle scenes with SDS paths, particle tracing methods are more widely used. PM [3] can robustly render caustics, but its rendering result is restricted by the physical memory. To remove the limit, PPM [4] is proposed to progressively render consistent results using limited memory. SPPM [5] extends PPM to handle distributed ray tracing effects such as motion blur and depth-of-field. However, these methods are inefficient to render scenes with complex lighting settings, in which case it is hard to trace photons to visible regions.

To solve the above problem, MPS [6] is proposed to deposit photons along light paths from the viewpoint to the light source. But the light paths are sampled via path tracing, which is inefficient to sample SDS paths. Chen et al. [7] give an importance function based on the initial photon density. But they use the mutation kernel proposed in [10], which cannot adapt to different scenes. RAPT [8] presents an importance function based on photon path visibility and

applies adaptive Markov chain Monte Carlo (AMCMC) to adjust the mutation parameter. However, RAPT treats all visible regions equally. Redundant mutations in less contributive regions do not provide additional benefits for improving the rendering results, while incurring much computation cost. In contrast, we design a novel importance function based on visual importance and photon path visibility, targeting regions which are visible and contribute highly to the final image.

The initial mutation parameter directly affects the performance of AMCMC, but it is manually set by users. RAPT [8] assigns the parameter as 1. Collin et al. [14] suggest that a smaller parameter is more appropriate. In contrast, we introduce a mutation parameter initialization method based on the tempered inter-chain adaptation (TINCA) [15], which successively performs Metropolis sampling for a series of target distributions. In our method, we generate multiple visual importance maps to construct a series of target distributions.

Visual importance, which measures the contribution of light paths to the final image, has been widely used in computer graphics. A comprehensive review of visual importance is presented in [16]. Peter et al. [17] provide a three-pass particle tracing method, which introduces an extra pass to trace importance particles (importons) and builds an importon map. Recently, Bashford et al. [18] utilize visual importance to improve rendering efficiency of image-based lighting. However, they also use an extra pass to trace importons. Vorba et al. [19] propose a method to learn the distribution of visual importance in an online training fashion. In our work, we construct the visual importance map in the first eye pass of SPPM. No extra pass is required. In this way, our method is easy to be merged with standard PPM and SPPM.

3 Overview

The core idea of VIAPT is to adaptively guide more photons to visible regions which contribute much to the final image, aimed at accelerating the convergence rate of rendering. An example is shown in Fig. 1. VIAPT can adaptively
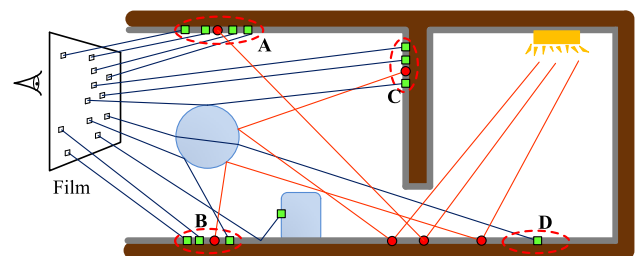

Fig. 1 Our method adaptively traces photons (*small red circles*) to visible and highly contributive regions (*A*, *B* and *C*), where more hit points (*small green squares*) are deposited. Region *D* is visible but less contributive since less hit points are deposited there
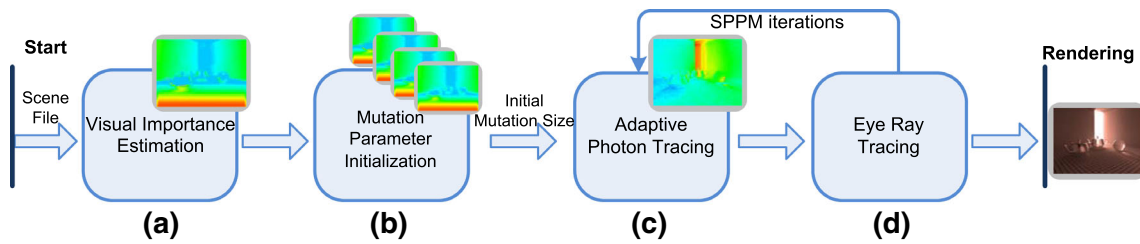
Fig. 2 Our algorithm framework: **a** visual importance estimation, **b** mutation parameter initialization and **c** adaptive photon tracing. **d** is the eye ray tracing pass of SPPM

trace more photons to regions A, B and C, where more hit points are deposited. Photons arriving at these regions contribute to more pixels in the final image. While region D is visible, it is less contributive to the final image. Photons reaching region D only contribute to one pixel. Since RAPT treats all visible regions equally, it still traces a lot of photons to region D, causing unnecessary computation cost. In contrast, VIAPT seeks to concentrate on tracing photons to visible and more contributive regions.

VIAPT is built on SPPM, which includes multiple iterations. Each iteration consists of a photon pass and an eye pass. There are three main steps in VIAPT: visual importance map construction, mutation parameter initialization and adaptive photon tracing (Fig. 2).

In the visual importance map construction step, we estimate the visual importance shot from the viewpoint to the scene in the first eye pass of SPPM. Then a visual importance map is constructed to facilitate visual importance query. In the mutation parameter initialization step, TINCA [15] is employed to select a suitable initial parameter. In the adaptive photon tracing step, photon paths are generated by performing importance sampling on the importance function. A new hybrid mutation strategy is introduced to generate new photon paths.

4 Importance function for photon tracing

The sampling space of our importance function is a hypercube space [10]. A point in the space, which is a vector of real numbers, corresponds to a photon path. Our importance function is defined as:

$$F(P) = \frac{V(P) \cdot M(P)}{s}. \qquad (1)$$

Here, $P = v_1 v_2 \ldots v_k$ is a photon path with k vertexes. A photon is deposited at each vertex. $V(P)$ is a visibility function, which is set to 1 if P splats any photons to any hit points, and $V(P) = 0$ otherwise. $M(P) = \max\{M(v_i)\}(1 \leq i \leq k)$ is the visual importance of P. To accelerate visual importance estimation for $M(v_i)$, we build a visual importance map beforehand. $s = \int_{\Omega} F(P) d\Omega$ is the normalization
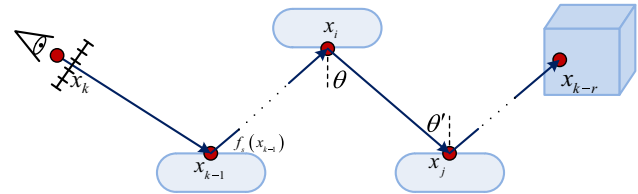
factor. As P is sampled with a probability proportional to $F(P)$, the energy α of photons deposited by P should be weighted as: $\alpha' = \frac{\alpha}{F(P)} = s \cdot \frac{\alpha}{V(P) \cdot M(P)}$.

4.1 Visual importance estimation

We estimate the visual importance in the eye pass of SPPM. Eye rays are emitted from the viewpoint and traced in a way similar to classical path tracing. Initially, all eye rays carry the same visual importance. Then the visual importance is accordingly updated if a scattering event occurs as an eye ray hits a surface in the scene.

Light transport between the light source and the camera can be formulated as a path integral, i.e. rendering equation [1],

$$L = \int_{\Omega_L} f(X) d\mu(X), \qquad (2)$$

where L is the pixel measurement, X is a light path, Ω_L is the path space of all light paths, $d\mu(X)$ is the differential product area measure, and $f(X)$ is the measurement contribution function.

Similar to Eq. (2), visual importance emitted from the viewpoint can be formulated as:

$$M = \int_{\Omega_M} g(\overline{X}) d\omega(\overline{X}). \qquad (3)$$

Here M is the visual importance measurement of a pixel. $\overline{X} = x_k, x_{k-1}, \ldots, x_{k-r}$ is an eye ray with $r + 1$ vertexes, where the first vertex x_k is on the camera lens and other vertices are scattering points on the scene surfaces, as shown in Fig. 3. Ω_M is the space comprised of all eye rays, $d\omega(\overline{X})$ is the differential product area measure. $g(\overline{X})$ is the

Fig. 3 View importance is emitted from the viewpoint to the scene

visual importance measurement function, which measures the visual importance arriving at a point. It can be expressed as:

$$g(\overline{X}) = W(x_k \to x_{k-1}) \left[\prod_{i=0}^{r-2} f_s(x_{k-i-1}) G(x_{k-i}, x_{k-i-1}) \right]$$
$$\times G(x_{k-r+1}, x_{k-r}), \tag{4}$$

where $W(x_k \to x_{k-1})$ is the visual importance emitted from x_k to x_{k-1}, $f_s(x_{k-i-1})$ is the BSDF at vertex x_{k-i-1}, $G(x_i, x_j) = V(x_i, x_j) \frac{|\cos\theta||\cos\theta'|}{\|x_i - x_j\|^2}$ is the geometry factor between vertex x_i and x_j, and $V(x_i, x_j)$ is the visibility term.

According to Eq. (3), we estimate the visual importance emitted from the viewpoint via tracing eye rays. If an eye ray hits a surface, Russian roulette technique is employed to decide whether to store a hit point on the surface. A hit point stores the position x, the incident direction ω, the BSDF and the visual importance M. After tracing eye rays from all pixels, we build a KD-tree to store all hit points in the scene.

4.2 Visual importance map construction

Visual importance is required when depositing a photon. To avoid the computation cost of estimating visual importance at runtime, we estimate a visual importance map in advance. For a hit point located at x, we search for its neighbor hit points in a disk area with radius R. The visual importance at x is:

$$M(x) = \frac{1}{\pi R^2} \sum_{i=1}^{K} m_i. \tag{5}$$

Here K is the number of hit points found in the disk. m_i is the visual importance of the ith hit point. Figure 4a shows the visual importance map of a Dark Room scene (Fig. 8). To eliminate local extreme values, we clamp each value between a new range (e.g. 10–90 % of the original range). The visual importance map effectively distinguishes less contributive regions and more contributive regions. Note that regions near the viewpoint have prominently larger visual importance than the far regions. More photon paths generated by sampling our

importance function will arrive at highly contributive regions, which is desirable.

Finally, we apply low-pass filters with different supports to smooth Fig. 4a, constructing a hierarchy of visual importance maps (Figs. 4b–e), which will be used in Sect. 6.

5 Adaptive photon tracing

Regular Markov chain Monte Carlo method can generate new paths via locally perturbing an existing path using the mutation kernel. However, it becomes highly inefficient if the target distribution is a multimodal distribution. Its Markov chain may get trapped around a local peak, preventing the exploration of the entire sampling space. Furthermore, the mutation parameter is relevant to specific scene settings. There is no universal parameter which is suitable for all scenes. Unsuitable parameter will lead to a slow convergence rate of Markov chain.

To overcome these flaws, we implement photon path sampling with a hybrid mutation strategy, which combines large mutation and small mutation. In the large mutation step, replica exchange [12] is used to obtain a visible photon path from an assistant Uniform distribution. In the small mutation step, AMCMC is employed to generate a new photon path from a slight perturbation of the current photon path using our mutation kernel. Meanwhile, the mutation kernel parameter is automatically adjusted by AMCMC.

5.1 Hybrid mutation strategy

Instead of generating new photon paths totally relying on the large mutation [8], we propose a novel hybrid mutation strategy for photon path generation using both large mutation and small mutation. Before generating a photon path, we firstly compute the probability q to implement large mutation. Then a random number δ is used to decide mutation strategy. If $\delta < \min(1, q)$, we implement path sampling using the Uniform distribution to achieve large mutation. Small mutation strategy is used otherwise.

We define the visual importance ratio σ between the current path P_i and the previous path P_{i-1} of the target dis-

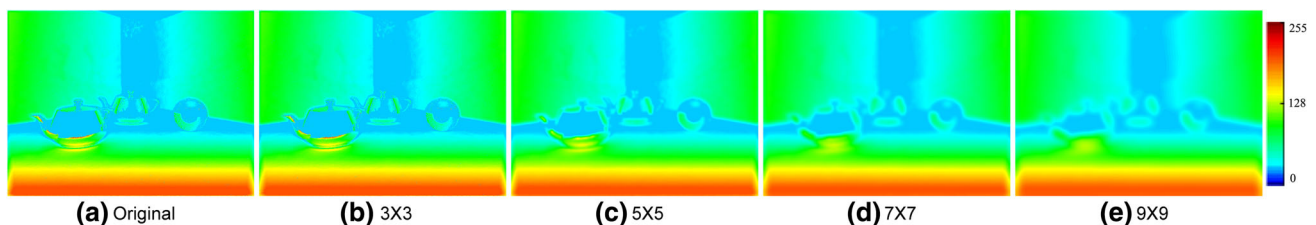**(a)** Original   **(b)** 3X3   **(c)** 5X5   **(d)** 7X7   **(e)** 9X9

Fig. 4 The visual importance map hierarchy of the Dark Room scene. **a** is the original visual importance map. **b–e** are produced by smoothing, **a** with low-pass filers. The filter support is given below each image

tribution: $\sigma = \frac{M(P_i)}{M(P_{i-1})}$. Depending on the means to generate P_i, there are two cases for computing q:

1. P_i is generated from small mutation.

   – If P_i is accepted,

   $$q = \begin{cases} 0, & \sigma \geq 1 \\ 1 - e^{-\sigma \cdot n}, & \sigma < 1 \end{cases}, \qquad (6)$$

   where n is the consecutive times for generating new paths using small mutation and it is set to zero after generating a path from large mutation;

   – Otherwise, q is set to 1. Next photon path will be generated from large mutation.

2. P_i is generated from large mutation.

   – If P_i is accepted, q can be estimated using Eq. (6);
   – Otherwise, P_i is replaced with a photon path generated by perturbing P_{i-1}, and q is estimated from case 1.

Intuitively speaking, if σ is less than 1, q keeps increasing as n increases. This is desirable, since we prefer that the chain visits more parts of the sampling space by performing large mutation.

5.2 Large mutation

Our target distribution is a multimodal distribution, because the visual importance varies a lot in different regions. To prevent the Markov chain from being trapped around local modes, we employ large mutation strategy to generate samples which are far from local modes.

An assistant Uniform distribution is used to generate uniform random samples in the sampling space. Then we utilize replica exchange Monte Carlo to implement interchain exchange. The main advantage of replica exchange Monte Carlo is to exchange samples between multiple chains without changing the distribution of each individual Markov chain.

We simultaneously run two Markov chains: C_u and C_t, where C_u corresponds to the Uniform distribution and C_t is the Markov chain of the importance function, namely the target distribution. Since C_u can easily generate new samples according to the Uniform distribution, we only perform unidirectional replica exchange, replacing the current path of C_t with a visible path of C_u, rather than mutually exchanging paths between C_u and C_t. This process is depicted in Fig. 5.

C_t accepts a path of C_u with a probability: $q(C_t \leftrightarrow C_u) = V(P_u)$. It is straightforward. Given a new photon path P_u from C_u, we replace the current path of C_t with P_u as long as
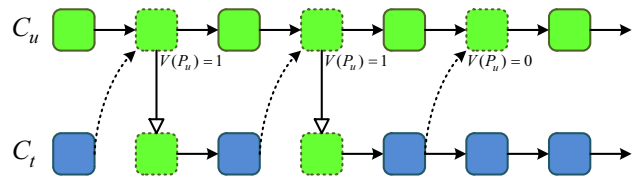

Fig. 5 The *hollow arrow* shows that the target distribution (C_t) accepts visible path P_u from an assistance Uniform distribution (C_u). The *solid arrow* shows small mutation and the dashed arrow shows large mutation

it is visible, since it is difficult to find a visible path in scenes with complex lighting settings.

5.3 Small mutation

To locally explore the target distribution, we generate new samples by slightly perturbing existing samples using a mutation kernel. Depending on the principle of symmetric random walk kernel [20], we design a novel mutation kernel: $x'_k = x_k + \hat{x}$, which adds each dimension of a point x with

$$\hat{x} = \begin{cases} \xi_1 \left(\frac{1}{e}\right)^{\frac{1}{\lambda_i}}, & \xi_2 > 0.5 \\ -\xi_1 \left(\frac{1}{e}\right)^{\frac{1}{\lambda_i}}, & \xi_2 < 0.5 \end{cases}. \qquad (7)$$

Here, ξ_1 and ξ_2 are two random numbers in the range of $(0, 1)$. λ_i is the mutation parameter of iteration i. λ_i directly controls the mutation extent. As it approaches infinity, the kernel degenerates to a uniform random variable, which generates arbitrarily large mutation. On the contrary, $\lambda_i = 0$ results in a constant x. The new sample x' is accepted with a probability $a(x \rightarrow x') = \min\left(1, \frac{F(x')}{F(x)}\right) = \min\left(1, \frac{V(x')M(x')}{V(x)M(x)}\right)$.

We use the controlled adaptive Marov chain Monte Carlo method [20] to adjust λ_i so that the acceptance rate reaches a target value. This process can be described as:

$$\lambda_{i+1} = \lambda_i + (A_i - A_*) \cdot \gamma_i \qquad (8)$$

Here A_i is the acceptance rate up to iteration i, which subsumes information of all historical samples. The acceptance rate is the ratio of accepted mutations to all mutations. A_* is the target acceptance rate. The optimum acceptance rate has been derived to be 0.44 in one dimension, while its value is close to 0.234 in high-dimensional space [21]. Since photon paths are sampled in a high-dimensional hypercube space, we use $A_* = 0.234$ in the algorithm. γ_i is a scaling factor to control adaptation extent. To ensure that the Markov chain converges to the target distribution, γ_i should satisfy two conditions [22]:

$$\begin{cases} \lim_{i \rightarrow +\infty} \gamma_i = 0 \\ \sum_{i=1}^{+\infty} \gamma_i \rightarrow +\infty \end{cases}. \qquad (9)$$

Similar to the idea of [23], we define $\gamma_i = \frac{1}{t}$, where $\{t\}$ is a non-decreasing positive integer series starting from 1. γ_i satisfies the conditions in Eq. (9) and approaches 0 at a moderate rate. Eq. (8) can be rewritten as:

$$\lambda_{i+1} = \lambda_i + (A_i - A_*)\big/ t . \tag{10}$$

Initially, an alternating sign for $A_i - A_*$ indicates that λ_i is oscillating around the stationary parameter, thus we keep increasing t to accelerate the convergence. Additionally, if the sign of $A_i - A_*$ is constant and the subtraction result is continuously approaching zero, which reveals that A_i is approaching A_*, we also increase t. On the contrary, if the sign of $A_i - A_*$ is constant and the subtraction result is far from zero, we do not increase t to keep the adaptation extent of γ_i.

5.4 Implementation details

Normalization factor estimation The normalization factor s is estimated using $s \approx \frac{1}{n} \sum_{i=1}^{n} V(P_i) \cdot M(P_i)$, where n is the count of photon paths generated from the Uniform distribution. Since the summation term can be simplified as a sum of the visual importance of visible paths, we progressively estimate s by accumulating the visual importance and the number of photon paths generated from the Uniform distribution.

Algorithm The pseudocode of our adaptive photon tracing method is shown in Algorithm 1. It is easily implemented in the framework of SPPM.

6 Mutation parameter initialization

One critical problem regrading the adaptation process in Eq. (10) is to select an appropriate initial mutation parameter λ_1. Existing methods [8,14] manually set an initial value. However, unsuitable staring parameter will result in that the mutation parameter ends up with a suboptimal value. We therefore propose a mutation parameter initialization method based on the tempered inter-chain adaptation method [15] to perform this task.

The main idea is to generate a hierarchy of target distributions $\{\Pi_T\}$, where T comes from a "temperature" parameter set $\{t_0, t_1, \cdots, t_N\}$. This theory requires that the difference between two adjacent distributions is small, while the difference between Π_{t_0} and Π_{t_N} is substantial. In our method, five temperature scales from 0 to 4 are used. In Fig. 4, the original visual importance map (Fig. 4a) corresponds to Π_{t_0}, and other visual importance maps with increasing smoothness correspond to distributions of increasingly higher temperature.

Algorithm 1: AdaptivePhotonTracing (initParam)

---
mutateSize←initParam, acceptCnt←1, mutateCnt←0, uniformCnt←1, largeProb←1, visSum←0, t←1

$P \leftarrow$ InitPhotonPath( );
$F \leftarrow$ TracePhotonPath(P);
for i $\leftarrow$ 0 to PhotonsPerPass
    doLargeMut $\leftarrow$ *Random*() < largeProb;
    if doLargeMut **then**
        $P' \leftarrow$ UniformSamp( );
        $F' \leftarrow$ TracePhotonPath(P');
        if IsVisible(P') **then**
            $P \leftarrow P'$; $F \leftarrow F'$;
            visSum $\leftarrow$ visSum $+ F$;
            unifromCnt $\leftarrow$ unifromCnt $+ 1$;
        end
        else doLargeMut $\leftarrow$ *false*;
    end
    if !doLargeMut **then**
        $P' \leftarrow$ Mutate(P,mutateSize);
        $F' \leftarrow$ TracePhotonPath(P');
        mutateCnt $\leftarrow$ mutateCnt+1;
        $a \leftarrow$ AcceptProb(F',F);
        if *Random*() < a **then**
            $P \leftarrow P'$; $F \leftarrow F'$;
            acceptCnt $\leftarrow$ acceptCnt+1;
        end
        inc $\leftarrow$ acceptCnt$\big/$mutateCnt $- 0.234$;
        if IsSignAlter(inc) or IsApproachZero(inc) **then**
            t $\leftarrow$ t $+ 1$;
        end
        mutateSize $\leftarrow$ mutateSize $+$ inc/t;
    end
    RecordContrib(P, 1/F);
    largeProb $\leftarrow$ UpdateLargeProb( )
    ScaleContrib(visSum$\big/$unifromCnt);
end

Algorithm 2 depicts this method. It starts adaptive photon tracing from distribution Π_{t_4} (Fig. 4e), which corresponds to the highest temperature. After the mutation parameter λ_{t_4} converges to a stationary value, Π_{t_3} is set as the target distribution and the stationary value is used as the initial mutation parameter of Π_{t_3}. This process continues until reaching the lowest temperature distribution Π_{t_0}. Then λ_{t_1} is used as the initial mutation parameter of Π_{t_0}. This is a reasonable initial parameter, since it has undergone a series of adaptation to learn a better value.

Algorithm 2: Mutation parameter initialization

N←4, initParam←1

for $T \leftarrow t_N$ to t_1
  UpdateTargetDistribution(T);
  repeat
    mutateParam $\leftarrow$ AdaptivePhotonTracing(initParam)
  until HasCoverged(mutateParam);
  initParam $\leftarrow$ mutateParam;
end

Table 1 Rendering parameters used in each scene

Scene	Resolution	Photons per pass (K)
Dark Room	640×480	50
Cornell Slit	512×512	100
Door	512×512	200
Cornell	512×512	100

Table 2 Initial and stationary mutation parameters

Scenes	Initial parameter		Stationary parameter	
	RAPT	VIAPT	RAPT	VIAPT
Dark Room	1.0	0.496	1.276	0.245
Cornell Slit	1.0	0.381	0.903	0.319
Door	1.0	0.213	0.418	0.117
Cornell	1.0	9.120	8.936	9.853

7 Results and discussion

Our algorithm and previous approaches are implemented on LuxRender [24]. All results are rendered on a dual quad-core 2.4 GHz Intel Xeon E5-2609 CPU, using 8 threads. For all scenes, the parameter α in SPPM is 0.7, the temperature scale N equals 5, and the optimum acceptance rate is 0.234. The R in Eq. (5) and the initial radius in SPPM are set to a 4-pixel width. Other different parameters used in each scene are presented in Table 1. Reference images in Fig. 10 are rendered by PPM using 24 h. Table 2 presents mutation parameters of test scenes. VIAPT begins with a suitable mutation parameter which is closer to the stationary value.

Experiments mainly focus on comparing VIAPT against RAPT and SPPM using uniform random photon sampling (URPS). Firstly, the effects of importance function and the hybrid mutation strategy are analysed by comparing the photon distribution. Secondly, the mutation parameter initialization method and the scaling factor are analysed. Finally, we compare VIAPT with previous rendering methods in four test scenes.

7.1 Algorithm analysis

Photon distribution analysis Depending on the hybrid mutation strategy, VIAPT efficiently samples photon paths accord-ing to the importance function. Photon paths adaptively guide photons to visible and more contributive regions. Figure 6 compares the photon distribution of VIAPT, URPS and RAPT by rendering the Dark Room scenes with 1000 photon passes. Due to the complex occlusion, the photon density of URPS is rather low in most regions. RAPT equally raises the photon density in all visible regions. In contrast, VIAPT improves the overall photon density and guides more photons to the interior floor and walls, where the visual importance is relatively higher (see Fig. 4a).

Mutation parameter analysis The effects of our mutation parameter initialization method and novel scaling factor series are analysed by rendering the Dark Room scene using VIAPT and RAPT (Fig. 7). The initial mutation parameter of RAPT is 1.0, which is used in [8]. VIAPT starts with an initial mutation parameter 0.496, which is selected by Algorithm 2.
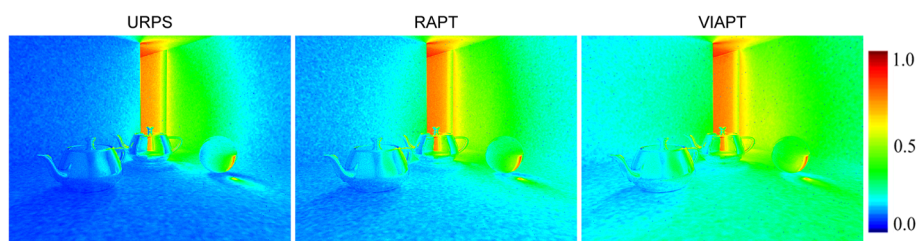
Figure 7a shows that the mutation parameter of VIAPT has a relatively larger variation than RAPT, because our scaling factor series (Sect. 5.3) drops in a moderate rate rather than decreasing monotonically. Although the mutation parameters of both methods converge to stationary values using less than 50 passes, VIAPT ends up with a better value than RAPT. This is validated in Fig. 7c, where the acceptance rate of VIAPT is consistently higher than RAPT. Note that both lines descend gradually, since the hit point radius in SPPM is decreased after each photon pass and the area where visibility function equals 1 is also decreased.

Figure 7b compares the acceptance rate of VIAPT and RAPT in the first 50 photon passes. Due to a well-selected initial mutation parameter and the scaling factor, the acceptance rate of VIAPT keeps oscillating around the optimum value 0.234. On the contrary, the acceptance rate of RAPT quickly falls after about 1 photon pass, because its scaling factor approaches 0 quickly and it cannot adjust the mutation parameter continuously.

7.2 Results

The Dark Room scene in Fig. 8 is a typical scene with difficult visibility settings. Only a small proportion of visible regions receive direct lighting from an area light in the adjacent room. We compare rendering results of URPS, RAPT and VIAPT in the same photon passes of 50 and 1000. VIAPT gener-

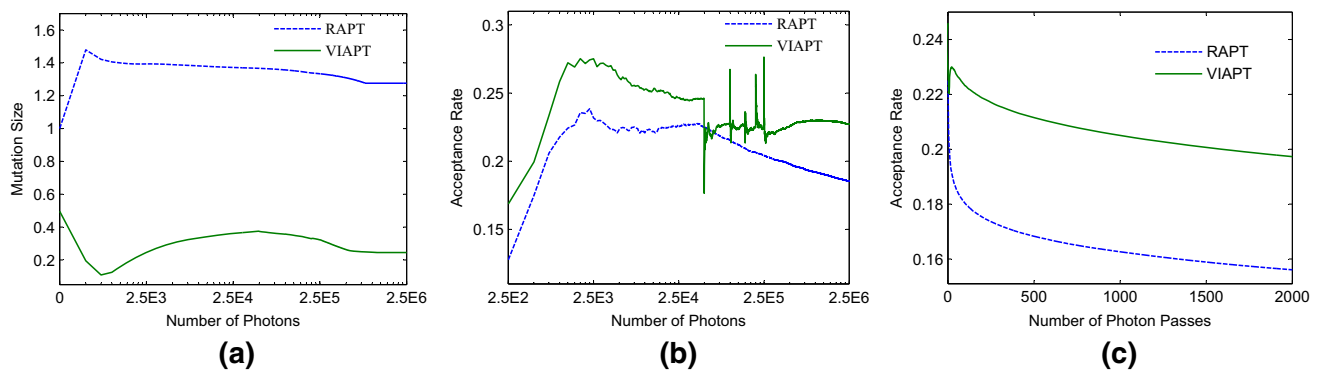**Fig. 6** Photon distributions of URPS, RAPT and VIAPT after 1000 photon passes

Fig. 7 Comparison of the mutation parameter (**a**) and acceptance rate (**b**) between RAPT and VIAPT in the first 50 photon passes with 50,000 photons per pass. **b** starts from 2.5E2 photons. (**c**) The acceptance rate variation trend of RAPT and VIAPT between 0 and 2000 photon passes
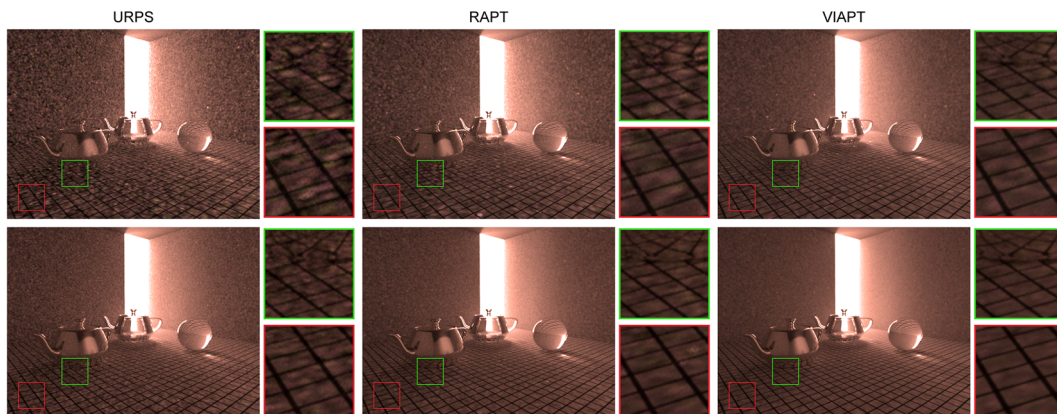

Fig. 8 Dark Room scene results of URPS (*left*), RAPT (*middle*) and VIAPT (*right*) in the same photon passes. *Top row* 50 photon passes. *Bottom row* 1000 photon passes
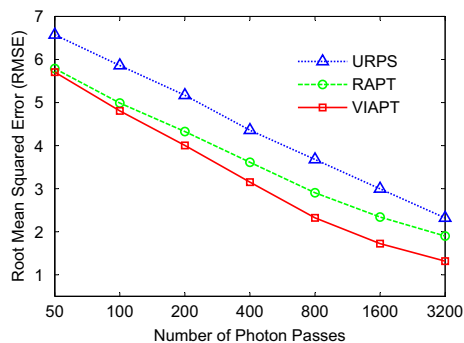

Fig. 9 RMSE of rendering the Dark Room scene using URPS, RAPT and VIAPT. It shows that VIAPT converges faster than other methods

ates visually smoother results compared with other methods. Note that, in pass 50, VIAPT does not show a large advantage over RAPT, since VIAPT involves a mutation parameter initialization step, which takes several passes. But VIAPT significantly outperforms RAPT later, as it dedicates to trace more photons to visible and more contributive regions. The root mean square error (RMSE) of the scene (Fig. 9) shows that VIAPT gives better results with lower numerical error in the same photon pass.

We also use structural similarity index (SSIM, [25]) besides RMSE to measure the similarity between the tested image and the reference image. SSIM builds on a model of the human visual system. An index of 1 indicates that the images are the same, whereas an index of 0 indicates that the two images share no similarity.

Figure 10a is the Cornell Slit scene illuminated by a light source outside the room through the door slit. We compare all the methods in the same rendering time of 30 min. Note that VIAPT usually performs less photon passes than RAPT, because the acceptance rate of VIAPT is consistently higher than RAPT and more accepted photon paths are traced in each pass. Nevertheless, VIAPT still generates a better result with lower RMSE and higher SSIM. Caustics on the floor produced by PT and MLT are rather noisy. Both methods are unable to generate the caustic on the back wall, since it is hard for them to sample the SDS paths. In contrast, VIAPT, URPS and RAPT robustly render the caustics on the floor and the back wall.

Figure 10b shows the classical Door scene, which was tested in [9]. In this difficult case, it is hard to pilot photons to the interior room. Compared with other methods, VIAPT
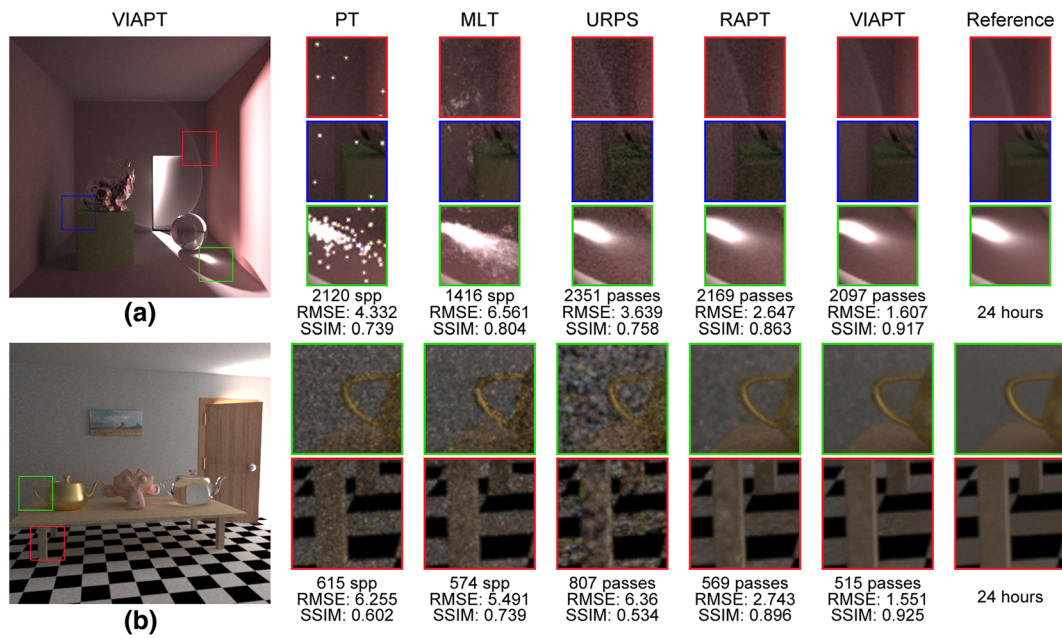
Fig. 10 The comparison of rendering the Cornell Slit scene (**a**) and the Door scene (**b**) in the same 30 min using PT, MLT, URPS, RAPT and VIAPT
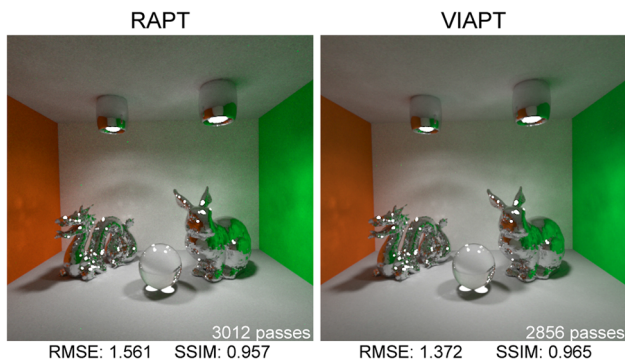

Fig. 11 The Cornell scene with a dragon and a rabbit rendered in an hour using RAPT (*left*) and VIAPT (*right*)

generates a higher quality result which is closer to the reference image in 30 min.

Figure 11 is a Cornell scene with simple lighting settings, where the light source can be seen from the viewpoint. We compare VIAPT with RAPT in the same rendering time of 1 hour. Since most photon paths can arrive at visible regions and the visual importance is uniformly distributed in the scene, VIAPT does not provide additional benefits, obtaining approximately similar results compared with RAPT, but costs an small computation effort to build visual importance maps and perform mutation parameter initialization.

7.3 Discussion

Although the visual importance map construction and the mutation parameter initialization cost extra time (less than 5

photon passes), as results show, the convergence rate benefits from improved photons distributions. The temperature scale is 5, but other values can also be used. A small value may not be sufficient to learn a suitable mutation parameter, meanwhile a large value requires more time for mutation parameter initialization. The current algorithm is implemented in a multithreaded fashion with 8 threads. More threads are possible, but the increased overhead in managing multiple threads can also degrade performance.

Since the visual importance estimation is implemented in the eye pass of SPPM, we can easily update visual importance map in each eye pass, but it is not worthwhile. Given a viewpoint, the visual importance distribution is fixed. Therefore, we generate visual importance map only once in the first eye pass, which saves the computational cost for updating visual importance map.

Our importance function results in a dynamic target distribution because the radii of hit points are gradually reduced, causing that the regions where $V(P) = 1$ are also reduced. The Markov chain convergence property with dynamic target distribution has not been fully verified. Nevertheless, experiments have proved VIAPT to be a practical method for rendering scenes with difficult visibility settings.

8 Conclusion and future work

This paper has presented a novel adaptive photon tracing approach using visual importance. A new importance function based on visual importance and photon path visibility

is proposed, and it is used as the target distribution for photon path sampling. Furthermore, a new mutation parameter initialization method based on TINCA is introduced to learn a suitable initial mutation parameter. Experiments have demonstrated that, when handling scenes with complex lighting configurations, our method efficiently distributes more photons to visible and highly contributive regions than existing adaptive photon tracing methods, while still rendering high-quality results.

Although our method is currently implemented based on SPPM, it can also be used for PPM. We also expect that our novel importance function and mutation parameter initialization method can be used for other Markov chain Monte Carlo methods and other adaptive photon tracing methods. Another possibility for future work we are interested in is to investigate the Markov chain convergence property on dynamic target distribution.

References

1. Kajiya, J.T.: The rendering equation. In: Proceedings of the 13th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH' 86, pp. 143–150. ACM, New York (1986)
2. Lafortune, E.P., Willems, Y.D.: Bi-directional path tracing. In: Proceedings of Third International Conference on Computational Graphics and Visualization Techniques (COMPUGRAPHICS' 93), pp. 145–153 (1993)
3. Jensen, H.W.: Global illumination using photon maps. In: Proceedings of the Eurographics Workshop on Rendering Techniques' 96, pp. 21–30. Springer, London (1996)
4. Hachisuka, T., Ogaki, S., Jensen, H.W.: Progressive photon mapping. ACM Trans. Graph. **27**(5), 130:1–130:8 (2008)
5. Hachisuka, T., Jensen, H.W.: Stochastic progressive photon mapping. ACM Trans. Graph. **28**(5), 141:1–141:8 (2009)
6. Fan, S., Chenney, S., Lai, Y.C.: Metropolis photon sampling with optional user guidance. In: Proceedings of the 16th Eurographics Symposium on Rendering, pp. 127–138. Eurographics Association, Aire-la-Ville (2005)
7. Chen, J., Wang, B., Yong, J.H.: Improved stochastic progressive photon mapping with metropolis sampling. Comput. Graph. Forum **30**(4), 1205–1213 (2011)
8. Hachisuka, T., Jensen, H.W.: Robust adaptive photon tracing using photon path visibility. ACM Trans. Graph. **30**(5), 114:1–114:11 (2011)
9. Veach, E., Guibas, L.J.: Metropolis light transport. Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH' 97, pp. 65–76. ACM, New York (1997)
10. Kelemen, C., Szirmay-Kalos, L., Antal, G., Csonka, F.: A simple and robust mutation strategy for the metropolis light transport algorithm. Comput. Graph. Forum **21**(3), 531–540 (2002)
11. Hoberock, J., Hart, J.C.: Arbitrary importance functions for metropolis light transport. Comput. Graph. Forum **29**(6), 1993–2003 (2010)
12. Kitaoka, S., Kitamura, Y., Kishino, F.: Replica exchange light transport. Comput. Graph. Forum **28**(8), 2330–2342 (2009)
13. Lehtinen, J., Karras, T., Laine, S., Aittala, M., Durand, F., Aila, T.: Gradient-domain metropolis light transport. ACM Trans. Graph. **32**(4), 95:1–95:12 (2013)
14. Collin, C., Ribardière, M., Gruson, A., Cozot, R., Pattanaik, S., Bouatouch, K.: Visibility-driven progressive volume photon tracing. Vis. Comput. **29**(9), 849–859 (2013)
15. Craiu, R.V., Rosenthal, J., Yang, C.: Learn from thy neighbor: parallel-chain and regional adaptive mcmc. J. Am. Stat. Assoc. **104**(488), 1454–1466 (2009)
16. Christensen, P.H.: Adjoints and importance in rendering: an overview. IEEE Trans. Vis. Comput. Graph. **9**(3), 329–340 (2003)
17. Peter, I., Pietrek, G.: Importance driven construction of photon maps. In: Proceedings of the 9th Eurographics Workshop on Rendering Techniques' 98, pp. 269–280. Springer, London (1998)
18. Bashford-Rogers, T., Debattista, K., Chalmers, A.: Importance driven environment map sampling. IEEE Trans. Vis. Comput. Graph. **20**(6), 907–918 (2014)
19. Vorba, J., Karlík, O., Šik, M., Ritschel, T., Křivánek, J.: On-line learning of parametric mixture models for light transport simulation. ACM Trans. Graph. **33**(4), 101 (2014)
20. Atchadé, Y.F., Rosenthal, J.S.: On adaptive markov chain Monte Carlo algorithms. Bernoulli **11**(5), 815–828 (2005)
21. Roberts, G.O., Rosenthal, J.S.: Optimal scaling for various metropolis-hastings algorithms. Stat. Sci. **16**(4), 351–367 (2001)
22. Andrieu, C., Robert, C.P.: Controlled mcmc for optimal sampling. Tech. rep, Centre de Recherche en Economie et Statistique (2001)
23. Kesten, H.: Accelerated stochastic approximation. The Annals of Mathematical Statistics, pp. 41–59 (1958)
24. http://www.luxrender.net (2008)
25. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to struc-tural similarity. IEEE Trans. Image. Process. **13**(4), 600–612 (2004)