Voxel classification methodology for rapid Monte Carlo simulation of light propagation in complex media

Nunu Ren (任努努)[1], Heng Zhao (赵 恒)[1], Shouping Zhu (朱守平)[1], Xiaochao Qu (屈晓超)[1], Hongliang Liu(刘宏亮)[1], Zhenhua Hu (胡振华)[1], Jimin Liang (梁继民)[1*], and Jie Tian (田 捷)[1,2]**

[1]*Life Sciences Research Center, School of Life Sciences and Technology, Xidian University, Xi'an 710071, China*

[2]*Institute of Automation, Chinese Academy of Science, Beijing 100190, China*

[*]*Corresponding author: jiminliang@gmail.com;* [**]*corresponding author: tian@ieee.org*

Received September 14, 2010; accepted December 7, 2010; posted online March 28, 2011

Monte Carlo (MC) method is a statistical method for simulating photon propagation in media in the optical molecular imaging field. However, obtaining an accurate result using the method is quite time-consuming, especially because the boundary of the media is complex. A voxel classification method is proposed to reduce the computation cost. All the voxels generated by dividing the media are classified into three types (outside, boundary, and inside) according to the position of the voxel. The classified information is used to determine the relative position of the photon and the intersection between photon path and media boundary in the MC method. The influencing factors and effectiveness of the proposed method are analyzed and validated by simulation experiments.

OCIS codes: 170.3660, 170.5280.

doi: 10.3788/COL201109.041701.

In the molecular imaging field, various optical molecular imaging techniques have been developed to achieve visualization and measurement of biological processes at cellular and molecular levels[1−3]. The study of light propagation in media is the key problem in optical molecular imaging. A highly accurate description of light propagation in turbid media is provided by the radiative transport equation (RTE). However, RTE is extremely computationally expensive due to its integro-differential nature. As a statistical method, Monte Carlo (MC) is a good choice to solve RTE with enough accuracy.

MC method is based on randomly constructing a set of trajectories of photons propagation in media, while the stepsize and direction of each trajectory depend on the absorption and scattering properties of the media. MC method was first introduced to study light propagation in tissues by Wilson *et al.*[4], and then improved by many researchers to achieve a more accurate result. Prahl *et al.* introduced anisotropy and internal light reflection into light propagation[5]. In 1995, Wang *et al.* proposed a MC modeling of light transport in multi-layered tissues (MCML)[6]. Boas *et al.* described a voxelized model (tMCimg) to improve the flexibility of the MCML model[7]. However, Binzoni *et al.* pointed out that it was difficult to incorporate precise boundary information using this model[8]. Margallo-Balbás *et al.*, therefore, proposed a triangle mesh-based model to improve the flexibility[9]. Li *et al.* developed a MC simulation platform (named molecular optical simulation environment (MOSE)) based on turbid media with regular shapes; in its current version, the triangle structure has also been introduced[10]. Recently, Shen *et al.*[11] and Fang *et al.*[12] proposed a tetrahedron-based model. By introducing the tetrahedron-based model, both speed and flexibility are solved more efficiently compared with the previous method. However, the tetrahedralization is still a big problem, especially for the inhomogeneous complex geometry.

For the MC simulation of light propagation in complex media, there are mainly two aspects influencing the time needed for MC simulation: the determination of relative position of photon and the intersection calculation between photon path and complex geometry. In this letter, we introduce a voxel classification method for MC simulation of light propagation in complex media, of which the boundary is constructed by triangles. The main idea is the classification of the voxels generated by dividing the occupied space of the media. The voxels are classified into three types according to the relative position between voxel and boundary: outside, boundary, and inside. By classifying the voxels, the time needed for determining the relative position of the photon and the intersection between photon path and geometry boundary can be reduced significantly. Calculation of the proposed method also has little increase even though the complex boundary is refined several hundred times.

The details of the method are introduced as follows. The voxelization process of complex media and the classification of the voxels are introduced first. The process for determining the relative position of the photon and the intersection calculation between photon path and complex geometry are then described. Finally,

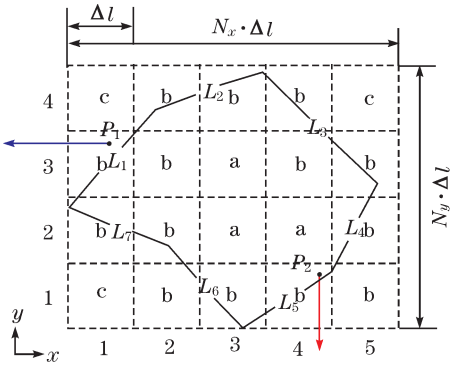

Fig. 1. Complex geometry in 2D is voxelized to illustrate the voxelization process.

some numerical experiments are conducted to analyze influencing factors and to validate the effectiveness of the proposed method.

The method proposed is concentrated on speeding up MC simulation while the detail rules for simulating light propagation in turbid media is the same as that in MCML[6]. For simplicity, the procedure used in our work is modified based on MCML. The structure of the complex media used in the method is the same as that described in Ref. [13].

Given a complex geometry corresponding to the media boundary constructed by triangles, the three-dimensional (3D) axis-aligned bounding-box of the geometry is first calculated, and the maximum $(x_{\max}, y_{\max}, z_{\max})$ and minimum $(x_{\min}, y_{\min}, z_{\min})$ of the vertices of the bounding-box are obtained. The bounding-box is then voxelized using a uniform grid, and the dimensions of the rectangular voxel $(\Delta x, \Delta y, \Delta z)$ and the number of voxels in each direction (N_x, N_y, N_z) are used to determine the grid parameters. For simplicity, the voxels are specified as cubes $(\Delta x = \Delta y = \Delta z = \Delta l)$ in our work and the coordinate of the voxel is defined as (x_i, y_j, z_k). An example of the voxelization process of the complex geometry in two dimensions (2D) is shown in Fig. 1. The complex geometry constructed by triangles in 3D is simplified to the polygon marked with $L_1 \sim L_7$ in 2D, and the bounding-box of the polygon is divided into a grid by square voxels. The number of voxels (N_x, N_y) in the x and y directions are computed by

$$N_x = \mathrm{CELL}[(x_{\max} - x_{\min})/\Delta l], \tag{1}$$

$$N_y = \mathrm{CELL}[(y_{\max} - y_{\min})/\Delta l], \tag{2}$$

where Δl is the side length of the square, and the function CELL $[\cdot]$ rounds a number upwards to the nearest integer. Thus, the side lengths of the grid are $N_x \cdot \Delta l$ and $N_y \cdot \Delta l$, respectively.

All the square voxels are classified into three types according to the relative position between voxel and geometry: outside, boundary, and inside. Boundary voxels are first determined through the overlap testing of triangle-cube in 3D or the border-square in 2D using the method provided by Möller[14]. As shown in Fig. 1, the squares marked with "b" are on the boundary of the polygon. More importantly, if a voxel falls into the "boundary" type, it will correspond to a number list of associated triangles which intersect it. For example, the boundary voxel (x_2, y_2) corresponds to the list (L_6, L_7), while the boundary voxel (x_1, y_2) corresponds to the list (L_1, L_7).

The remaining voxels are then classified into "inside" and "outside" using an improved ray method on the basis of "boundary" voxels. In the improved ray method, the direction of the ray is perpendicular to the nearest border of the bounding-box, so it can minimize the number of triangles checked in the calculation of ray-triangle intersection. For example, as shown in Fig. 1, the direction of the ray emitted from point P_1 is along the negative x axis, and that emitted from point P_2 is along the negative y axis.

Through the steps above, a voxel table, in which all the voxels are classified into three types, is constructed. The triangle lists corresponding to the "boundary" voxels are also established.

Given the voxel table and the photon coordinate (x, y), the number (x_i, y_j) of the voxel containing the photon can be computed by

$$x_i = \mathrm{FLOOR}[(x - x_{\min})/\Delta l], \tag{3}$$

$$y_i = \mathrm{FLOOR}[(y - y_{\min})/\Delta l], \tag{4}$$

where the function FLOOR rounds a number downwards to the nearest integer. By referring to the voxel table, the relative position of the photon can be determined quickly according to the voxel type. The steps are as follows:

1. Inside: The photon is inside the complex geometry.
2. Outside: The photon is outside the complex geometry.
3. Boundary: In this case, the relative position of the photon is determined by the improved ray method.

The relative position of the photon can be determined quickly by referring to the voxel table while the photon locates in the "inside" or "outside" voxels because the voxel table has been constructed before the MC simulation. Even for the case of "boundary" voxel, the computational cost is reduced by the improved ray method compared to the traditional ray method.

The photon may cross the boundary of the media during propagation, so reducing the computation of photon path-triangle intersection is another important aspect to speed up MC simulation. As shown in Fig. 2, the grid for dividing the polygon is the dashed line and the photon path is the segment PQ. The borders (or triangles) that may be intersected with the photon path are in the "path bounding-box" determined by the start and end points of the photon path, such as the rectangle marked with blue lines shown in Fig. 2. Note that each border (or triangles) in the "path bounding-box" will be checked just once. The number of borders needed to be checked is related to the size of the "path bounding-box." The size of the "path bounding-box" is small because the stepsize of the photon path is small. In addition, the "path bounding-box" can be further reduced by refining the grid.

The code of the proposed method is realized based on the modification of MOSE which has been validated in many papers[10,11,13]. The following numerical experiments then focus on analyzing the influencing factors and effectiveness of the proposed method. Through the

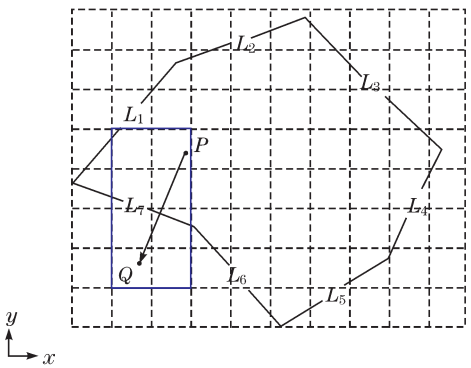

Fig. 2. (Color online) Intersection computation between photon path and mediaboundary.

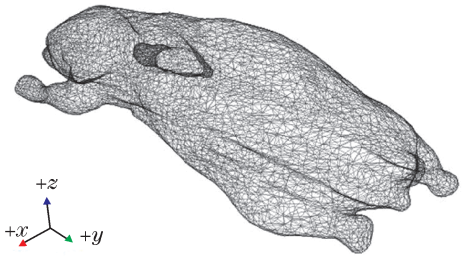

Fig. 3. Surface rendering of the digimouse whose boundary is constructed by 10000 triangles.

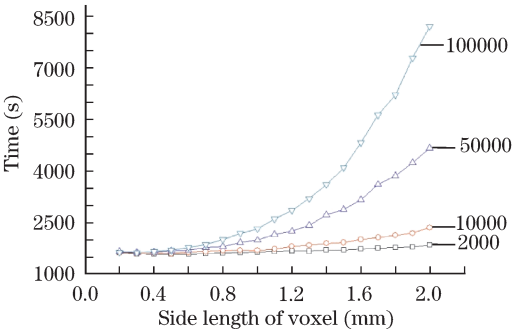

Fig. 4. Relationship between the simulation time and the side length of the voxel.

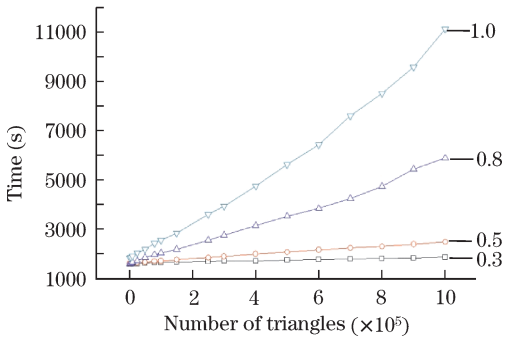

Fig. 5. Relationship between the simulation time and the number of triangles.

above analysis, the performance of the method is mainly influenced by the side length of the voxel, so the first set of experiments is conducted to analyze the relationship between simulation time and side length of the voxel. The second set is conducted to analyze the stability of the method. Finally, the effectiveness of the proposed method is validated.

Homogeneous Digimouse (38×99.2×20.8 (mm)) data provided by Dogdas are used in the following experiments[15]. The optical parameters of the Digimouse are assigned as follows: absorption coefficient μ_a =0.0088 mm^{-1}, scattering coefficient μ_s = 20.97 mm^{-1}, anisotropy coefficient $g = 0.94$, and relative index of refraction $n = 1.35$. The number of the Digimouse surface is different in the following experiments. A solid spherical

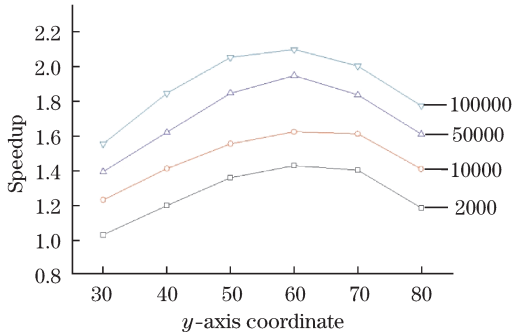

Fig. 6. Speedup of the proposed method over MOSE (v2.1).

light source with 1.0-nW total power is used as the *in vivo* light source and sampled by 10^6 photons. Figure 3 shows the 3D surface rendering of the Digimouse and the light source. The central processing unit (CPU) code of MC simulation is executed on a 2.67-GHz Intel Xeon processor.

Through the above analysis, we know that the time needed for determining the photon position is related to the voxel type in which the photon locates. The calculated amount when the photon locates in a "boundary" voxel is greater than in others, so the simulation time should mainly be related to the number of "boundary" voxel, which is directly proportional to the voxel size. The side length of the voxel varies from 0.2 to 2.0, while the numbers of triangles are 2000, 10000, 50000, and 100000, respectively. The light source is located at (15, 60, 14) mm. The simulation results are shown in Fig. 4, from which shows that the simulation time are all directly proportional to the side length, although the increasing rates are different. The simulation results validate the above analysis, so the MC simulation can be sped up by reducing the side length using the proposed method. However, the side length cannot be reduced without limits due to memory constraints.

The stability of the proposed method is studied. The number of triangles varies from 2000 to 1 billion while the side lengths are 0.3, 0.5, 0.8, and 1.0 mm, respectively. The light source is also located at (15, 60, 14) mm. The simulation results are shown in Fig. 5, in which we can observe that the simulation time are all directly proportional to the number of triangles, and the increasing rate is also proportional to the side length of the voxel. Notably, the rate of time increase is just only 17.73% while the side length is 0.3 mm. This increase rate is much smaller than that mentioned in Ref. [9], in which the increase rate is 66.67% while the number of triangles varies from 1000 to 100000. However, choosing a relatively small value of side length is necessary for maintaining the stability of the proposed method.

In the last set of experiments, the speedup of the proposed method is investigated by comparing it with the previous method used in MOSE[10,13]. The experiments are conducted while moving the light source along the y-axis—the start point is (15, 30, 14) mm and the end point is (15, 80, 14) mm. The side length of the voxel is set at 0.3 mm. From the simulation results shown in Fig. 6, the speedup of the proposed method over MOSE (v2.1) is between 1.02 and 2.1 times.

In conclusion, we propose a voxel classification method

to speed up MC simulation. Compared with previous methods, the proposed method reduces computation cost while preserving the flexibility of the media. Influencing factors and effectiveness of the method are validated by the numerical experiments. From the simulation results, we know that determining an appropriate value of side length of the voxel is an important problem in the proposed method. Note that simulation speed becomes a fixed value as the side length becomes smaller. Memory constraints should also be considered. According to the characteristic of near-infrared light propagation in turbid media, in which the average stepsize of photon is approximately between 0.02 and 0.2 mm, the side length set between 0.1 and 0.3 mm is an appropriate value according to the above experiments. However, although the proposed method has good efficiency in determining photon position while the photon locates in inside/outside voxels, the determination process while the photon locates in the boundary voxels is still time-consuming. To overcome this deficiency, further research is needed and other methods could be a good reference.

This work was supported by the National "973" Program of China (Nos. 2006CB705700 and 2011CB707702), CAS Hundred Talents Program, the National Natural Science Foundation of China (Nos. 81090272, 81000632, and 30900334), the Shaanxi Provincial Natural Science Foundation Research Project (No. 2009JQ8018), and the Fundamental Research Funds for the Central Universities.

References

1. V. Ntziachristos, J. Ripoll, L. V. Wang, and R. Weissleder, Nat. Biotechnol. **23,** 313 (2005).
2. G. Wang, W. Cong, Y. Li, W. Han, D. Kumar, X. Qian, H. Shen, M. Jiang, T. Zhou, J. Cheng, J. Tian, Y. Lv, H. Li, and J. Luo, Curr. Med. Imaging. Rev. **2,** 453 (2006).
3. L. Jin, Y. Wu, J. Tian, H. Huang, and X. Qu, Chin. Opt. Lett. **7,** 614 (2009).
4. C. Wilson and G. Adam, Med. Phys. **10,** 824 (1983).
5. S. A. Prahl, M. Keijzer, S. L. Jacques, and A. J. Welch, Proc. SPIE IS **5,** 102 (1989).
6. L. V. Wang, S. L. Jacques, and L. Q. Zheng, Comput. Meth. Prog. Bio. **47,** 131 (1995).
7. D. A. Boas, J. Culver, J. Stott, and A. Dunn, Opt. Express **10,** 159 (2002).
8. T. Binzoni, T. S. Leung, R. Giust, D. Rüfenach, and A. H. Gandjbakhche, Comput. Meth. Prog. Bio. **89,** 14 (2008).
9. E. Margallo-Balbás and P. J. French, Opt. Express **15,** 14086 (2007).
10. H. Li, J. Tian, F. Zhu, W. X. Cong, L. V. Wang, and G. Wang, Acad. Radiol. **11,** 1029 (2004).
11. H. Shen and G. Wang, Phys. Med. Biol. **55,** 947 (2010).
12. Q. Fang, Opt. Express **1,** 165 (2007).
13. N. Ren, J. Liang, X. Qu, J. Li, B. Lu, and J. Tian, Opt. Express **18,** 6811 (2010).
14. T. Möller, J. Graph. Tools **6,** 29 (2001).
15. B. Dogdas, D. Stout, A. F. Chatziioannou, and R. M. Leahy, Phys. Med. Biol. **52,** 577 (2007).