

三次元形状を考慮した半透明物体のリアルタイムレンダリング

持田 恵佑^{†1} 岡本 翠^{†1} 小澤 禎裕^{†1} 久保 尋之^{†2} 森島 繁生^{†3}

概要: 人肌, 歯, 大理石などといった半透明物体のレンダリング技術は, 高品質な CG を制作する上で欠かすことができないが, 表面下散乱の影響を考慮する必要があるため, 膨大な計算時間を必要となることが課題であった. 本研究では, メッシュデータをボクセライズすることで, 物体の表面だけでなく内部の構造を含めた三次元形状を把握することで, 物体の厚みに依存しやすい単一散乱の影響による放射輝度の近似計算が可能となり, 半透明物体のリアルタイムレンダリングを実現する. 本手法を用いることで, 既存手法では表現できなかった物体内部に空洞などの不均一な構造を含むような形状においても, 破綻のない高品質な描画が可能である. また三次元形状を Look Up Table (LUT)に格納して各シェーダステージから参照することで, GPU を用いた高速な描画を実現する.

1. はじめに

近年, 映画やゲームで用いられる CG のクオリティは, フォトリアリスティックレンダリング技術の向上により, 飛躍的な発展を遂げている. このようなフォトリアリスティックな CG の表現上欠かせない要素の一つに, 半透明物体の描画が挙げられる. 半透明物体は, 人間の肌, 歯, 大理石などに代表され, 表面下散乱と呼ばれる特有の物理現象が発生する. 入射した光の一部は, 物体内部で繰り返し散乱を起こし, 入射点とは異なる点から出射する. 表面下散乱は大きく二種類に分類され, 物体内部で一度だけ散乱を起こしたのちに物体外部に光が出射する単一散乱[1]と, 物体内部で複数回散乱を起こしたのちに光が出射する多重散乱[2]に分けられる. 表面下散乱による光の影響を物理的に正しく計算し, 半透明物体の描画を行うには, モンテカルロレイトレーシング法[3]やフォトンマッピング法[4]といった計算コストの高い手法を用いる必要がある. そのため, ゲームなど, ユーザーの入力に対してリアルタイムで動作することが求められるシーンにそのまま適用することは困難である. そこで, 物理現象を近似することで, 半透明物体のリアルタイムレンダリングを実現するための様々な手法が研究されてきた.

本研究では, 物体の表面のみではなく, 内部構造を含めた三次元形状をメッシュデータのボクセライズによって近似し, 表面ボクセル間の厚みを事前計算することで, 単一散乱が支配的な半透明物体のリアルタイムレンダリングをおこなう手法を提案する. 本手法を用いることで, 既存手法では表現できなかった, 物体内部に空洞などの不均一な構造を含むような物体においても, 破綻のない描画が可能となる.

2. 関連研究

単一散乱を近似したリアルタイムレンダリング手法の一つに, Dachsbacher らの提案した Translucent Shadow Maps (TSM)[5]が挙げられる. TSM では, 物体の厚みが大きいほどその内部を直進する光の減衰の度合いが大きくなると仮定することで, 単一散乱による入射光の減衰を近似し半透明物体のリアルタイムレンダリングをおこなう. 物体の厚みを得るためにシャドウマップ[6]を使用することで, 高速な描画を行うことが可能である. シャドウマップとは, ある視点から物体までの奥行きを各画素に格納したテクスチャのことである. 図1にシャドウマップ作成の概要図を示す. シャドウマップは GPU を用いることで高速に生成することが可能であるため, 物体に対する光源や視点の位置が変化しても, その都度新たなシャドウマップを生成することができる. 図2に TSM の概要図を示す. TSM では, 光源方向からシャドウマップを作成し物体上に投影することで, 描画する頂点と光の入射点との対応付けをおこない, 対応画素を参照することで光源から光の入射点までの距離 d_{in} を得ることができる. これを, 光源から描画する頂点までの距離 d_{out} から差し引くことで物体の厚みを得ることができる. この手法を用いることで, 逆光により物体が透けるような表現を, リアルタイムで再現することが可能である. しかし, 物体に凹形状が含まれるような場合, d_{in} と d_{out} の間に存在する空洞部も厚みとして算入されてしまうため, 正しい厚みを取得することができず, 不自然な描画となってしまうという問題がある.

Kosaka ら[7]は, この問題に対し, 物体の凹形状を把握できる視点位置から, 物体の空洞部の割合を格納した誤差マップを作成し, これを参照することで空洞形状を考慮した厚みに補正する手法を提案した. 誤差マップ作成の概要図を図3に示す. TSM で取得した厚みに対して, この誤差マップ上に格納された誤差の割合を乗算することで, 正しい厚みに補正することができる. 図4に, それぞれの手法で

^{†1} 早稲田大学

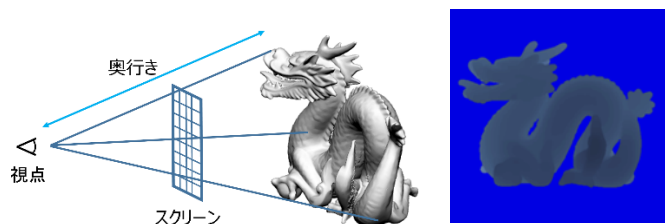
Waseda University

^{†2} 奈良先端科学技術大学院大学

Nara Institute of Science and Technology

^{†3} 早稲田大学理工学術院理工学総合研究所/JST CREST

Waseda Research Institute for Science and Engineering/JST CREST



(a) シェドウマップ作成の概要図 (b) シェドウマップ

図1 シェドウマップの作成

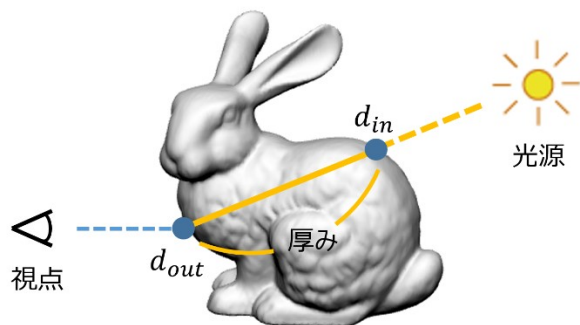


図2 TSMの概要図

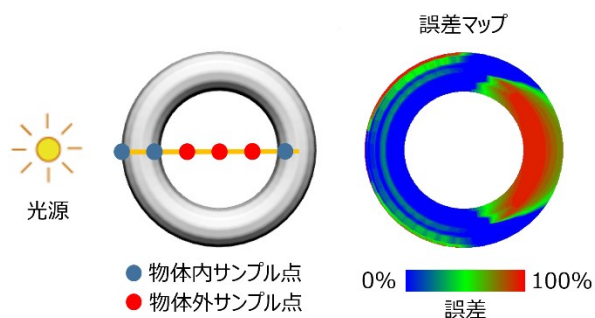
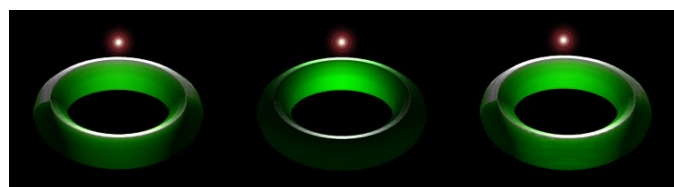


図3 誤差マップ作成の概要図



(a) 正解値 (b) Dachsbacher et al. (c) Kosaka et al.

図4 各手法の比較

トーラスを描画した結果を示す。シェドウマップ同様、誤差マップも GPU を用いることで高速に生成することが可能であるため、Kosaka らの手法においてもリアルタイムレンダリングをおこなうことが可能である。しかし、外部視

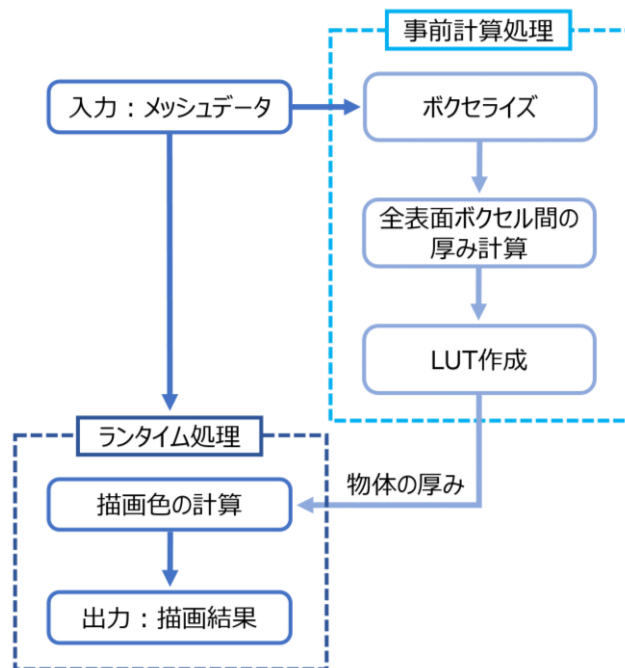


図5 提案手法のフローチャート

点から物体内部の空洞形状が把握できないような物体においては、正しく誤差マップを作成することができず、不自然な描画になってしまうという問題がある。

また、Bernabei ら[8]は、球面調和関数を用いて物体形状を近似することで、内部に不均質な構造を持つ単一散乱が支配的な半透明物体のリアルタイムレンダリングをおこなう手法を提案した。しかし、球面調和関数を用いたことにより高周波成分が再現されず、ぼやけた描画結果になってしまう。

3. 提案手法

本研究では、内部構造を含めた物体の形状を事前計算により把握することで、高速に半透明物体の描画をおこなう。図5に提案手法のフローチャートを示す。事前計算処理部分では、まず物体の内部構造を含めたメッシュデータの三次元形状をボクセライズによって近似し、物体形状の離散化をおこなう。次に、全ての表面ボクセルに対し、他のすべての表面ボクセルとの対応を取り、厚みの事前計算をおこなう。ランタイム処理部分では、事前計算した厚みを参照することで、減衰関数から描画色を決定し、半透明物体の描画をおこなう。

本手法を用いることで、物体が空洞形状を含む場合においても、破綻のない描画が可能となる。また、事前計算した厚みをテクスチャに格納して、各シェーダステージから参照することで、リアルタイムレンダリングを実現する。

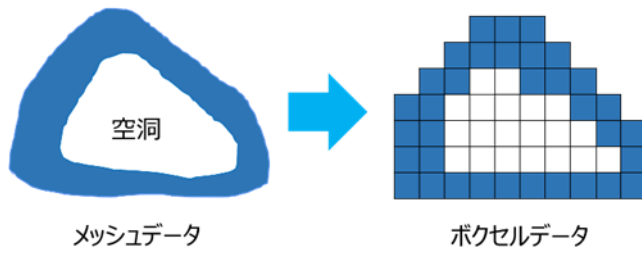


図 6 ボクセライズ

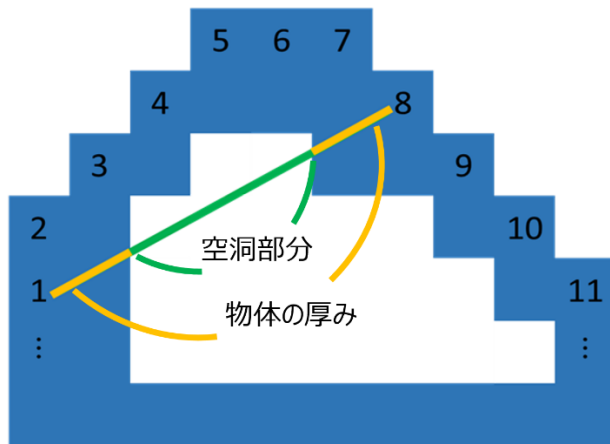


図 7 表面ボクセルのナンバリングと厚み取得

3.1 メッシュデータのボクセライズ

Look Up Table (LUT)と呼ばれるテクスチャに対象物体の厚みを事前に格納するため、図 6 に示すようにメッシュデータのボクセライズをおこなう。頂点数の多い物体では LUT に使用するメモリ容量が膨大になるため、実用的ではない。そこで、ボクセライズを用いてメッシュデータとして存在する物体の形状をボクセルに近似することで、物体の厚みを離散化し、LUT に格納する。

3.2 表面ボクセル間の厚み情報取得

物体の厚みを事前に取得するため、まず物体表面に存在するボクセルのみを抽出し、表面ボクセルに番号付けをおこなう。また、元のメッシュデータの頂点が、何番の表面ボクセルに属するかの記述もおこなう。番号付けをしておくことで、GPU による描画時に、LUT の参照すべきピクセルとの対応付けが容易になる。

次に、2 つの表面ボクセルの重心座標間を結ぶ線分を作成し、線分上に存在する物体の空洞部分を差し引いた長さを、物体の厚みとして取得する。ここまでの概要を図 7 に示す。厚みの取得は、全ての表面ボクセル間に対しておこなう。その後、取得した 2 点間の厚みをグレースケール変

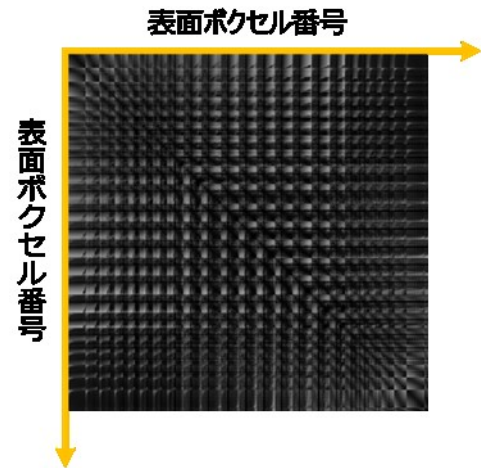


図 8 作成した LUT

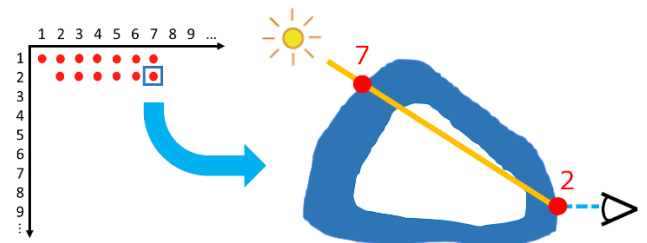


図 9 LUT を用いた厚みの取得

換し、LUT へと格納する。図 8 に実際に作成した LUT の例を示す。縦軸、横軸はともに表面ボクセルの番号を表し、輝度が高い方が、対応するボクセル番号間の物体の厚みが大きいことを表している。

3.3 描画色の計算

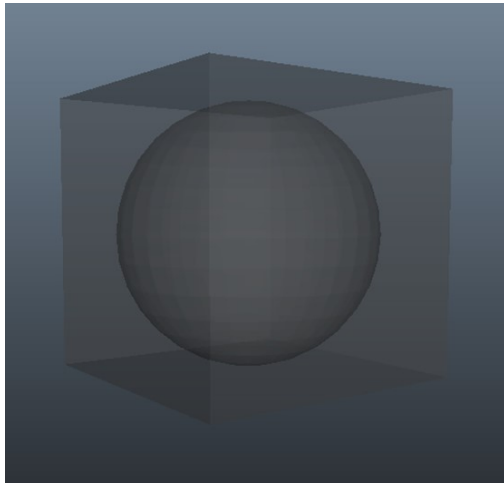
描画色を計算するために、入射光に対する減衰関数を以下のように定義する。まず、入射光の散乱には一般的な粒子の散乱モデルが適用できると仮定する。この時、物体内に進入した光の粒子が距離 x だけ進んだ地点で散乱をおこす確率を $P(x)$ 、平均自由行程を散乱係数の逆数 s^{-1} と定義すると、粒子の散乱確率は

$$P(x) = \exp(-sx) \quad (1)$$

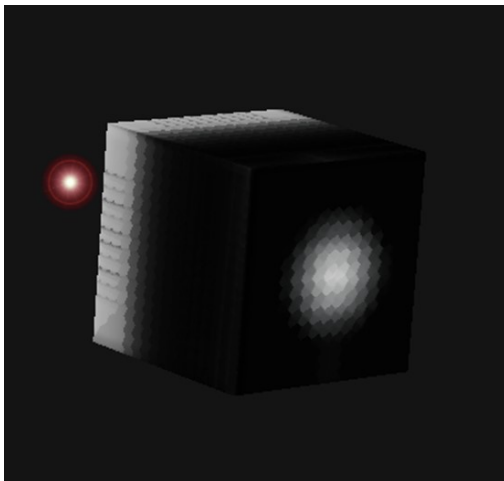
となる。したがって、強度 I_{in} の入射光が物体内部を距離 d だけ進んで物体外部に射出するとき、入射光の強度は距離に対して指数関数的に減衰する。このとき射出光強度 I_{out} は、

$$I_{out} = I_{in} \exp(-sd) \quad (2)$$

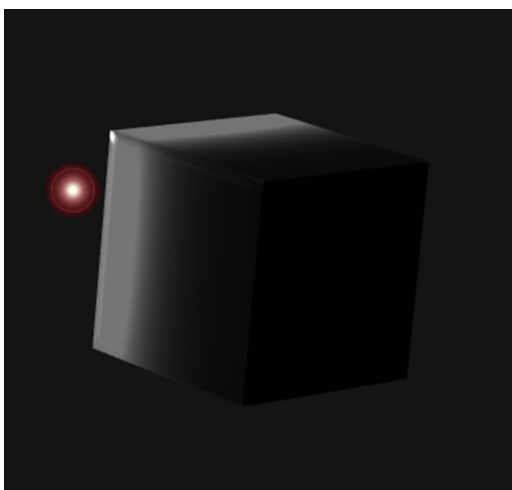
により計算することができる。ここで、距離 d は物体の厚



(a) 形状モデル



(b) 描画結果 (提案手法)



(c) 描画結果 (Dachsbacher et al.[2003])

図 10 内部に球状の空洞を持つ半透明なキューブ



図 11 トーラスの描画結果

みと同義であり, 図 9 に示すように, LUT を参照することで取得することができる. また, 本手法では空洞部での光の減衰量は微小であり, 無視できると仮定した. そのため, 空洞部が存在した場合, その厚みを差し引いた値を物体の厚みとして用いる.

出射光強度 I_{out} を RGB 値ごとに計算することで, 描画色を決定し, レンダリングをおこなう.

4. 結果

本章では, 提案手法における結果を示し, それに基づく考察をおこなう. 実行環境には, CPU: Intel Xeon 2.93GHz, GPU: NVIDIA Quadro 4000, DirectX 9.0c を用いた.

4.1 描画結果

まず, 内部に球状の空洞を持つ半透明なキューブを用いて描画をおこなった. 入力形状モデルを図 10(a)に示し, 提案手法, Dachsbacher らの手法(TSM)を適用して描画をおこなった結果を, それぞれ図 10(b),(c)に示す. 提案手法におけるボクセル分割数は $25 \times 25 \times 25$ とした. 描画結果から, 既存手法においては正しく厚みの把握が出来ないため, 物体内の空洞を考慮することができない. しかし, 提案手法を適用した際は, 物体の持つ厚みの違いにより, キューブに入射した光のうち, 厚みの大きい部分を通過する光は輝度が大きく減衰し, 厚みの小さい部分を通過する光は輝度の減衰が少なくなることが確認できる.

次に, トーラスに対して描画をおこなった. 提案手法による描画結果を図 11 に示す. ボクセル分割数は $60 \times 60 \times 60$ とした. 提案手法では, 物体に内包される空洞のみではなく, 物体外部の中空部分の厚みも考慮することが可能であり, 図のように, 透過光がトーラスの手前部分まで届く

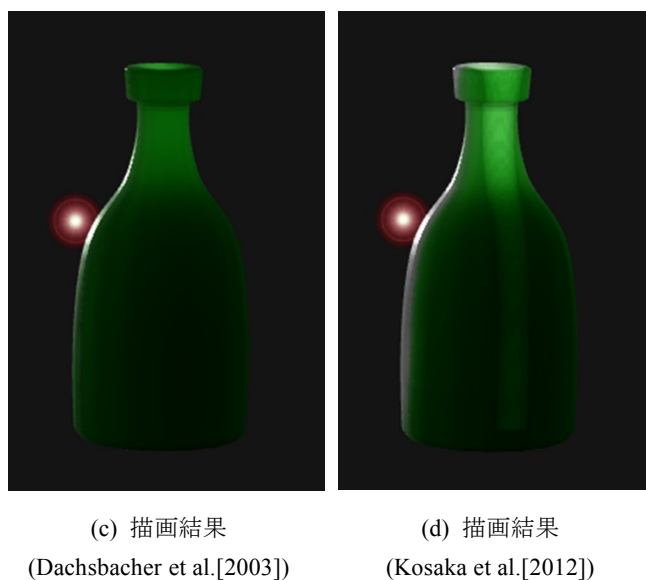
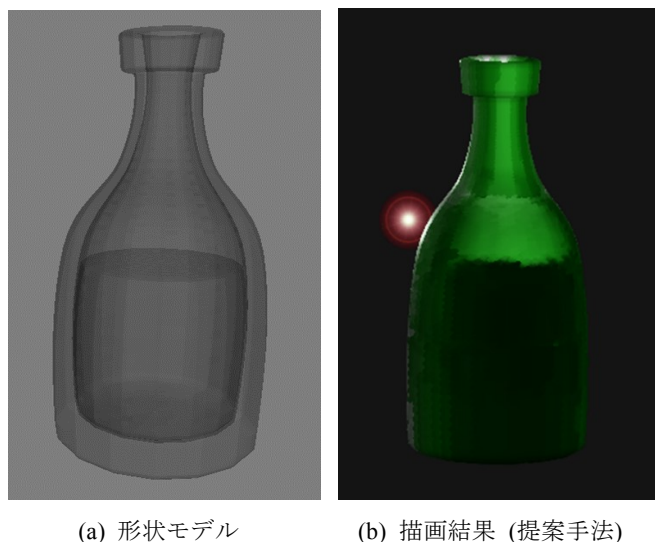


図 12 内部の一部が液体で満たされたボトル

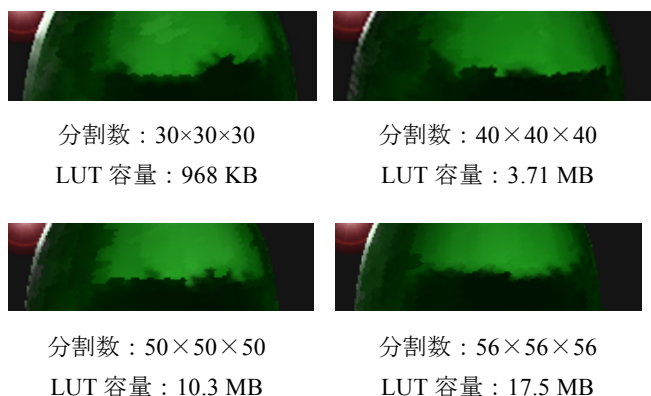


図 13 ボクセル分割数ごとの量子化誤差と LUT の容量

ようなシーンを表現することができる。

最後に、内部が空洞で下半分が液体で満たされたボトルに対して描画をおこなった。入力形状モデルを図 12(a)に示し、提案手法、Dachsbacher らの手法、Kosaka らの手法を適用して描画をおこなった結果を、それぞれ図 12(b),(c),(d)に示す。ボクセル分割数は $56 \times 56 \times 56$ とした。Dachsbacher らの手法では内部の空洞が考慮されておらず、Kosaka らの手法では、誤った空洞の把握をおこなっていることが、描画結果からわかる。それに対し、提案手法では事前に作成した LUT を参照することで、モデルの空洞部の形状を正しく把握し、表現することが可能である。

また、本研究同様、内部の空洞を表現することのできる Bernabei らの手法では、球面調和関数を用いるために全体的にぼやけた描画となるのに対し、提案手法では、頂点情報を離散化しているため、厚みが急激に変化する箇所でも明瞭な境界線を得ることが可能である。

また、表 1 に各モデルにおける本手法での描画速度を示す。15000 頂点程度のモデルに対しても 150FPS を超える描画速度で動作し、Dachsbacher らの手法や Kosaka らの手法と比較しても、ほぼ同等の速度でレンダリングをおこなうことが可能である。また、ボクセル分割数は LUT の容量のみに関係し、描画速度には影響を与えない。

表 1 各モデルにおける描画速度

モデル	計算時間[fps]	頂点数	ボクセル分割数
キューブ	160.0	2020	$25 \times 25 \times 25$
トーラス	159.4	9216	$60 \times 60 \times 60$
ボトル	156.3	15648	$56 \times 56 \times 56$

4.2 ボクセル分割数と量子化誤差の影響

本手法ではボクセルサイズによる厚みの離散化をおこなっているため、メッシュデータのボクセル分割数に依存して量子化誤差が生じる。ボクセル分割数を増加させることで量子化誤差を軽減させることができるが、作成する LUT の解像度も増加するため、ボクセル分割数と LUT に使用する容量はトレードオフの関係となる。図 13 に、図 12 で用いたボトルを、ボクセル分割数を変えて描画した際の量子化誤差の様子と、LUT の容量を示す。ボクセル分割数を増加させるほど、ボトルの水面におけるばらつきが減少し、LUT の容量が増加していくことが確認できる。

5. まとめと今後の課題

本研究では、物体の内部構造を含めた3次元形状を、メッシュデータをボクセライズすることにより把握し、表面に存在するボクセル間の厚みを事前計算しておくことで、不均一な内部構造を持つ半透明物体の描画を実現した。また、事前計算した厚みをLUTに格納し、描画時に参照することで、GPUを用いたリアルタイムレンダリングを実現した。

今後の課題としては、内部の空洞による光の散乱現象への影響をより厳密に考慮することや、物質内部に空洞だけでなく、異なる性質を持った物質が複数含まれている場合にも破綻なく描画できる減衰関数の考慮が挙げられる。また、ボクセライズによる量子化誤差を改善する必要があると考えられる。

参考文献

- 1) Hanrahan, P. and Krueger, W.: Reflection from layered surfaces due to subsurface scattering, Proceedings of the 20th annual conference on Computer graphics and interactive techniques, ACM, pp.165-174(1993).
- 2) Stam, J.: Multiple scattering as a diffusion process, Rendering Techniques'95, Springer, pp.41-50(1995).
- 3) Pharr, M. and Hanrahan, P.: Monte Carlo evaluation of non-linear scattering equations for subsurface reflection, Proceedings of the 27th annual conference on Computer graphics and interactive techniques, ACM, pp.75-84 (2000).
- 4) Jensen, H. W. and Christensen, P. H.: Efficient simulation of light transport in scenes with participating media using photon maps, Proceedings of the 25th annual conference on Computer graphics and interactive techniques, ACM, pp.311-320 (1998).
- 5) Dachsbacher, C. and Stamminger, M.: Translucent Shadow Maps, Rendering Techniques 2003, pp.197-201(2003).
- 6) Williams, L.: Casting curved shadows on curved surfaces, Proceedings of the 5th annual conference on Computer graphics and interactive techniques, ACM, pp.270-274(1978).
- 7) Kosaka, T., Hattori, T., Kubo, H., and Morishima, S.: Rapid and Authentic Rendering of Translucent Materials Using Multi-view Depth-Maps, SIGGRAPH Asia 2012 Posters, No.45, ACM, (2012).
- 8) Bernabei, D.: Real-time single scattering inside inhomogeneous materials, The Visual Computer, Vol.26, No.6, Springer, pp.583-593(2010).