

微分可能レンダリングによるコースティクス画像からの 鏡面反射物体の形状復元

岩口 堯史^{1,a)} 久保 尋之² 川崎 洋¹

概要: 本発表ではすりガラスなどの拡散体の背後にある鏡面反射物体形状の再構成に取り組む。拡散体背後の物体は環境光の下ではボケて観測されるため形状の復元は困難だが、拡散体上の微小領域に光源を照射すると、拡散体の表面には反射による集光模様であるコースティクスが現れる。本論文では、光学現象を考慮した微分可能レンダリングによりコースティクスを解析し拡散体の背後にある物体の表面法線を復元する方法を提案する。提案するレンダリング手法はメッシュ表現を利用するため、レンダリング画像だけでなく、幾何学的なパラメータを対象に最適化を行うことができる。実験では、反射光分布から形状を復元し、反射光の位置から幾何学的パラメータを最適化することが可能であることを示す。さらに、正確なコースティクスの再現が要求される Goal-based caustics へ応用しレンダリング能力を示す。

1. はじめに

直接観測できない物体の形状計測は、Non-line-of-sight[1]に代表されるように大きな注目を集めている。そこで本研究では、すりガラスのような拡散体の後ろに配置された鏡面反射物体の形状計測に取り組む。鏡面反射物体で生じる鏡面反射光は、コースティクスとよばれる集光模様を生じるため、点光源で照明されたときのコースティクスの観測画像を手掛かりに、物体形状の復元を行う。

このような問題設定におけるひとつの解法として、微分可能レンダリングを用いて、観測画像とレンダリング画像に対して誤差を逆伝播することにより、形状パラメータを最適化して物体形状を復元する方法が考えられる。微分可能レンダリングのうち、微分可能ラスタライザはメッシュをラスタライズすることで効率的にレンダリングすることが可能であるが、コースティクスのような光学現象を再現することは難しい。一方で、物理ベースの微分可能レンダリング手法は、光線の経路を考慮し様々な光学現象を再現可能であるが、計算コストが高いという問題がある。本稿では、コースティクスを効率的にレンダリングし形状パラメータを最適化するためにメッシュ表現を用い、微分可能ラスタライザを利用した微分可能レンダリングを提案する。本稿で形状を復元する対象となる鏡面物体は、頂点、エッ

ジ、面から構成されるメッシュで表現される。物理ベースレンダリングでは光線ごとに計算される鏡面物体上のメッシュの各面に対応する反射光を、本手法ではメッシュで表現することで、反射方向に基づいて拡散板上へ投影する。このとき反射光が持つエネルギーが保存されるため、反射光の面と鏡面物体上の面の面積比を考慮することで観測される放射照度を近似することが可能である。これらの計算は線形であるため、微分可能ラスタライザ上に実装することですべて微分可能な形式でコースティクスを再現することができる。さらに、提案する微分可能レンダリングを用いてコースティクスによる鏡面物体の形状推定、任意のコースティクスを生成する表面形状の設計という2つの問題に取り組む。シミュレーション実験、実環境実験により単一・複数画像のコースティクス画像のみを入力として、法線推定が可能であることを示す。

2. 関連研究

2.1 微分可能レンダリング

微分可能レンダリングはシーンの3次元構造を深層学習に取り入れるため近年注目を集める技術である。3次元シーンの表現に広くメッシュ表現が用いられているが、メッシュを2次元画像のピクセルにサンプリングするラスタライズは離散的な操作であるため逆伝播の計算ができない。遮蔽や影の境界部において勾配を近似し、微分可能にする手法は微分可能ラスタライザと呼ばれる[2], [3], [4], [5], [6]。提案手法はコースティクスのレンダリングに微分可能ラスタライザの代表的な実装であるPyTorch3D [7]を用いる。

¹ 九州大学
744 Motoooka, Nishi-ku, Fukuoka 819-0395, Japan

² 千葉大学
Chiba University

a) iwaguchi@ait.kyushu-u.ac.jp

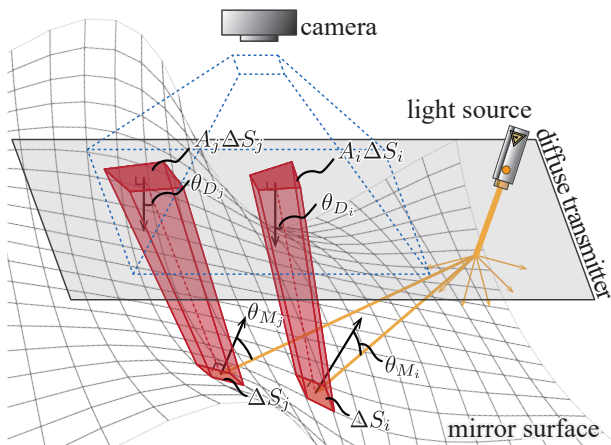


図 1 計測系の模式図。拡散板上の入射点から拡散した光は物体表面上で鏡面反射し拡散板に到達する。物体表面上と拡散板上の放射照度の関係は式 (1) で表される。

ラスタライズでは再現不可能な複数回の反射を含むような光学現象を再現できる物理ベースレンダリングを微分可能にした、物理ベース微分可能レンダリングに関する研究もおこなわれている [8], [9], [10]。一般的なフレームワークである Mitsuba2[11] やコースティクスや大域照明など特定の光学現象を対象としたアルゴリズム [10], [12] が提案されている。これらの手法は光線追跡に基づくため計算コストが高くなりやすい。また、レンダリングアルゴリズムによっては再現できない光学現象があり、例えば Mitsuba2 はパストレーシングに基づくため点光源下のコースティクスの再現は不可能である。提案手法ではラスタライズを利用しコースティクスの微分可能レンダリングを実現する。

2.2 コースティクスを利用した形状解析

Goal-based caustics (GBC) は目標画像と同じコースティクスを生成するように鏡面・屈折面を設計する研究である [13], [14], [15], [16]。また、Frisvad らの研究 [12] では光伝播モデルを用いた微分可能レンダリングによりガラスの表面形状を再構成している。提案手法は反射系について GBC や表面形状再構成を行うための微分可能レンダリングを提案する。

3. 微分可能レンダリングによる形状推定

3.1 計測系

計測系の模式図を図 1 に示す。レーザー光が鏡面物体の上にある拡散板に照射され、透過光は入射点から鏡面物体に向かって拡散する。これは拡散板上に点光源が存在するとみなすこともできる。拡散光が鏡面物体表面で反射し、拡散板上に到達したときに拡散板上に生成されるコースティクスを、上部に設置したカメラにより撮影する。拡散板は平面、また鏡面反射は 1 回のみおこると仮定する。

3.2 メッシュ投影を用いたコースティクスの再現

コースティクスのレンダリング 鏡面形状は頂点、エッジ、面からなるメッシュで表現される。各頂点は隣り合う面で共有される。

反射光メッシュの投影 反射光方向の計算のため、まず、物体表面上の各頂点から光源の入射点に対する光源方向ベクトルを計算する。次に光源方向ベクトルと法線方向のベクトルから反射方向ベクトルを計算し、拡散板との交点を計算する。

反射光強度の計算 鏡面上の微小面積 ΔS が受け取る光束 ϕ は拡散板上の面積 $A\Delta S$ にすべて到達するため、エネルギー保存則より

$$E_D = \frac{\cos \theta_D}{A \cos \theta_M} E_M \quad (1)$$

が成立する。ただし、 E_D, E_M はそれぞれ放射板上、物体表面上の放射照度、 θ_D, θ_M はそれぞれの微小面と入射光のなす角を表す。

レンダリング時には鏡面メッシュのある 1 面と対応する反射光メッシュの 1 面に注目し、エネルギー保存則を近似する。つまり、微小面積をそれぞれのメッシュの面で離散化し拡散板上に生じる反射光強度の計算を行い、反射光メッシュの各面を放射照度の明るさで描画することでコースティクスを再現する。反射光メッシュの面の重畳が発生する場合は、ラスタライズ時にそれぞれの面の放射照度の合計をピクセルの明るさとして計算することで集光現象が適切に表現される。

3.3 最急降下法による最適化

幾何計算、反射光強度の計算は線形であるため、メッシュの描画を微分可能ラスタライズにより行うことで、メッシュのエッジの不連続性にもかかわらず全体を通して勾配の計算ができ、誤差逆伝搬が可能である。この微分可能レンダリングを用いて出力に対して、光源パラメータ、物体の法線などの形状パラメータ、反射マップなどの材質パラメータの最適化を行う。パラメータの最適化は再急降下法により行う。

物体の形状推定のため、異なる位置に配置された N 個の光源に対応する N 枚の画像をレンダリングし入力画像とする。入力画像とレンダリング画像間の誤差関数を以下のように定義する：

$$\mathcal{L}_{RGB} = \frac{1}{N} \|I_i^{RGB} - \tilde{I}_i^{RGB}\|_2^2. \quad (2)$$

4. 実験結果

4.1 Goal-based caustics

この実験では提案するアルゴリズムによりコースティクスが正確に再現されることを GBC のタスクで確認する。GBC では入力される目標画像と同じ輝度分布を持つコー

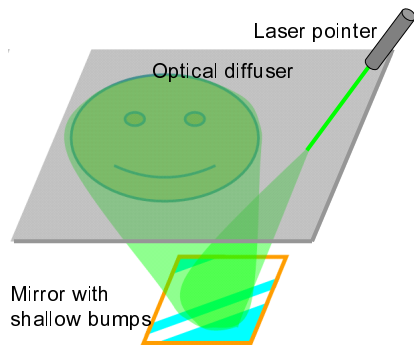


図 2 GBC のシーンにおける光源、物体、拡散板の配置. 鏡面表面には浅い凹凸のみ存在すると仮定する.

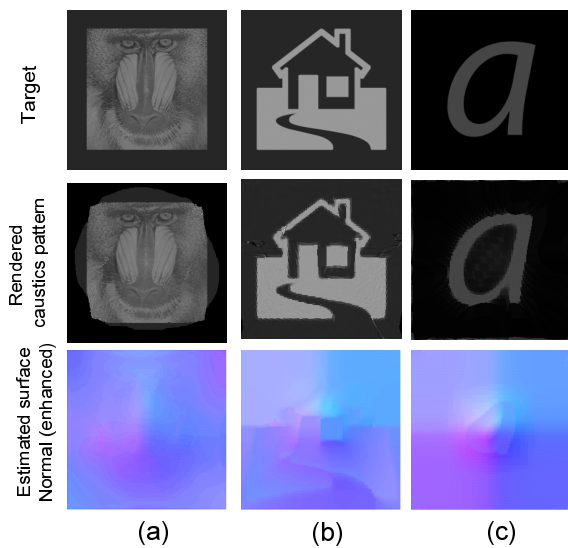


図 3 GBC の結果. 目標画像とほぼ一致するコースティクスを生じる法線マップを推定可能であることが確認された.

スティクスを生じさせるような表面形状を求める. 表面形状は微小な凹凸をもつ平面と仮定し, 法線 \mathbf{n} のみを最適化する. 誤差関数は

$$\mathcal{L} = \mathcal{L}_{RGB} + |\mathbf{n}|_{TV} \quad (3)$$

で表される. 画像の解像度は 256×256 , 法線マップの解像度は 128×128 である. 最適化の反復回数は 1000 回で打ち切る. 実験環境 (GeForce RTX 2080Ti) における計算時間はおよそ 200 秒であった. 図 3 に入力画像, 再現されたコースティクスと法線マップを示す. それぞれ (a) は自然画像, (b) 平面的な輝度分布, (c) 高コントラストな目標画像に対し適切な法線を生成できていることが確認できる.

4.2 鏡面物体の形状推定

既知の光源下のコースティクス画像から鏡面物体形状の再構成を行う. 損失関数には式 (3) を用いた. 最初のシミュレーション実験では, 提案するレンダラーにより真値の法線マップを用いて生成されたコースティクス画像を入力として法線推定を行う. 実験結果 (図 4) からは, 単純な Cylinder や Bowl の形状では 1 枚の入力画像から推定さ

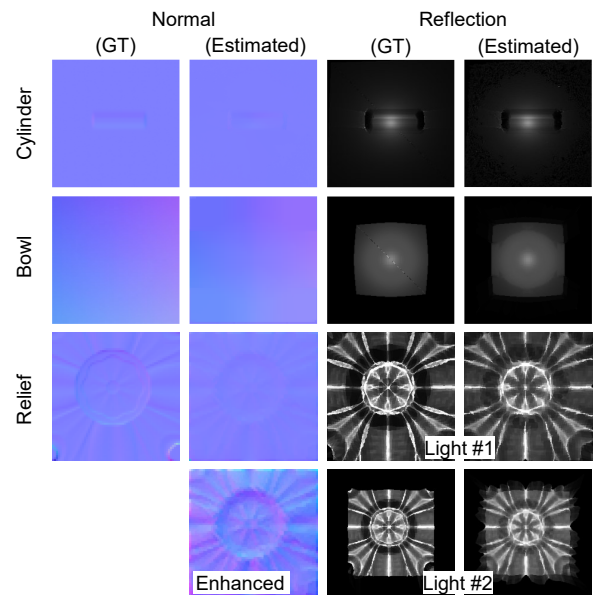


図 4 シミュレーション環境における形状再構成の結果. Cylinder, Bowl のような単純な形状を単一画像から, 高周波形状を含む Relief を 2 枚の入力画像から推定された法線は真値とほぼ一致することが確認された.

れた法線は真値とほぼ一致した. Relief については複数の反射光分布を与えることで法線の曖昧性が解消されることを期待し光源位置の異なる 2 枚の入力画像とした. 実験結果からは真値に近い法線が推定されており, 高周波な形状も復元されていることが確認できる.

次に, 実環境における計測実験を行った. 計測装置を図 5 に示すとおりであり, 光源としてポイントレーザーを照射する. ディフューザと物体間の距離は計測値, 光源の入射位置は観測画像の輝度から推定した位置を与える. 入力画像を 1 枚としたときの実験結果を図 6 に示す. 凹形状のボタン, コインいずれについても, 再構成された法線を用いてレンダリング画像は入力画像とほぼ一致する. 法線についてはボタンは凹形状が右上に, コインは平坦な形状が中央に正しく再構成されている一方で, 輪郭が実際よりソフトで曖昧になっている部分も見られる. 入力画像を 3 枚としたときの実験結果を図 7 に示す. レンダリング画像は観測画像と比較してボケが見られ, 法線にはソフトな輪郭が見られるが内部の様子はほとんど再現されていない. これは光源位置のキャリブレーション誤差が原因となり 3 つの光源環境下のコースティクスを一貫して再現するような法線の推定が困難となったことが原因と考えられる.

5. 結論

本稿ではメッシュ投影によりコースティクスを再現することで, 既存の微分可能ラスタライザ実装を用いた効率的なコースティクスの微分可能レンダリング手法を提案した. GBC のタスクでは, 異なる輝度分布の特徴を持つ目標画像に対して提案手法の有効性を示した. また, 鏡面物

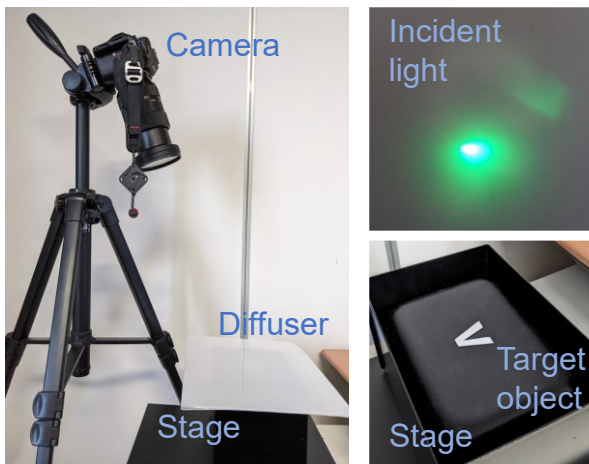


図5 実環境実験の計測装置。レーザーポインタにより拡散板を照射し、拡散板上に現れるコースティクスをカメラから撮影する。

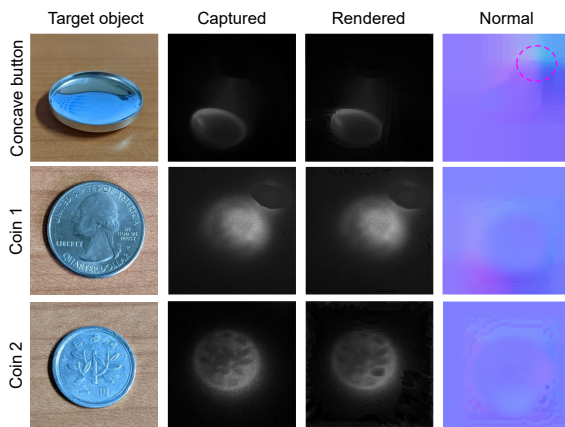


図6 単一画像からの形状再構成結果。入力画像とほぼ一致するレンダリング画像が得られている一方で、法線においては輪郭にソフトな部分が見られる。

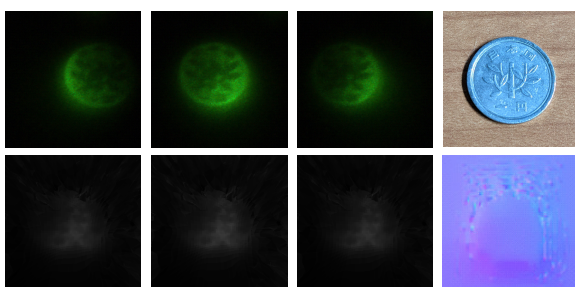


図7 3枚の入力画像を用いた形状再構成の結果。物体の輪郭は復元されているが、内部の高周波形状の再現性は低い。

体の形状再構成への応用においてはシミュレーション環境における有効性が確認された一方で、実環境においては光源のキャリブレーション誤差などが原因となり、精度の低下がみられた。今後は光源位置の同時推定、深層学習への導入に取り組みたい。

謝辞 本研究は JSPS 科研費 JP20K19825, JP19H04138, JP21K19799 および JST 創発的研究支援事業 JPMJFR206I の助成を受けたものである。

参考文献

- [1] A. Velten, T. Willwacher, O. Gupta, A. Veeraraghavan, M. G. Bawendi, and R. Raskar, "Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging," *Nature*, p. 745, 2012.
- [2] M. M. Loper and M. J. Black, "Opendr: An approximate differentiable renderer," in *ECCV*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 154–169.
- [3] H. Kato, Y. Ushiku, and T. Harada, "Neural 3d mesh renderer," in *CVPR*, 2018.
- [4] T. Nguyen-Phuoc, C. Li, S. Balaban, and Y.-L. Yang, "Rendernet: A deep convolutional network for differentiable rendering from 3d shapes," in *Advances in Neural Information Processing Systems 31*, 2018.
- [5] S. Liu, T. Li, W. Chen, and H. Li, "Soft rasterizer: A differentiable renderer for image-based 3d reasoning," *ICCV*, Oct 2019.
- [6] W. Chen, J. Gao, H. Ling, E. J. Smith, J. Lehtinen, A. Jacobson, and S. Fidler, *Learning to Predict 3D Objects with an Interpolation-Based Differentiable Renderer*. Red Hook, NY, USA: Curran Associates Inc., 2019.
- [7] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Accelerating 3d deep learning with pytorch3d," *arXiv:2007.08501*, 2020.
- [8] S. Zhao, W. Jakob, and T.-M. Li, "Physics-based differentiable rendering: From theory to implementation," in *ACM SIGGRAPH 2020 Courses*, ser. SIGGRAPH '20. New York, NY, USA: Association for Computing Machinery, 2020.
- [9] G. Loubet, N. Holzschuch, and W. Jakob, "Reparameterizing discontinuous integrands for differentiable rendering," *Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, vol. 38, no. 6, Dec. 2019.
- [10] T.-M. Li, M. Aittala, F. Durand, and J. Lehtinen, "Differentiable monte carlo ray tracing through edge sampling," *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, vol. 37, no. 6, pp. 222:1–222:11, 2018.
- [11] M. Nimier-David, D. Vicini, T. Zeltner, and W. Jakob, "Mitsuba 2: A retargetable forward and inverse renderer," *Transactions on Graphics (Proceedings of SIGGRAPH Asia)*, vol. 38, no. 6, Dec. 2019.
- [12] J. R. Frisvad, L. Schjøth, K. Erleben, and J. Sporring, "Photon differential splatting for rendering caustics," *Comput. Graph. Forum*, vol. 33, no. 6, p. 252–263, Sep. 2014. [Online]. Available: <https://doi.org/10.1111/cgf.12347>
- [13] M. Papas, W. Jarosz, W. Jakob, S. Rusinkiewicz, W. Matusik, and T. Weyrich, "Goal-based caustics," in *Comput. Graph. Forum*, 2011.
- [14] Y. Schwartzburg, R. Testuz, A. Tagliasacchi, and M. Pauly, "High-contrast computational caustic design," *ACM TOG*, vol. 33, no. 4, 2014.
- [15] Y. Yue, K. Iwasaki, B. Y. Chen, Y. Dobashi, and T. Nishita, "Poisson-based continuous surface generation for goal-based caustics," *ACM TOG*, vol. 33, no. 3, 2014.
- [16] J. Meyron, Q. Mérigot, and B. Thibert, "Light in power: A general and parameter-free algorithm for caustic design," *ACM TOG*, vol. 37, no. 6, Dec. 2018. [Online]. Available: <https://doi.org/10.1145/3272127.3275056>