

宇宙科学データのコンテンツ制作支援に係る一考察 A Study on Content Creation Support for Space Science Data

三浦 昭[†]
Akira Miura

1. はじめに

筆者らは、広く宇宙科学データに係る可視化やコンテンツ制作に関する研究を進めて来たが、その一方で、進行中のミッションや将来計画の検討等の場面においても可視化は重要な要素となっており、種々の可視化が行われている。これら多様な可視化にあたっては、様々なツールや開発環境が用いられており、一律にそれら可視化の支援を行うのは必ずしも容易ではない。

このような状況下において、本稿では、様々な側面で扱っているコンテンツ制作手法の共通要素に鑑みつつ、主として技術的な支援を主眼とした考察を行う。

2. 内製 3D CG ソフトウェアとその課題

宇宙科学分野の可視化ツールとして、筆者らが開発・維持管理している、レイトレーシングベースの 3D CG ソフトウェア[1]が存在する(以下、「内製ソフトウェア」と称する)。これは宇宙科学に係る特定の要求に特化したレンダリングエンジンである。各種プロジェクトや研究等の分野での利用を想定しており、SPICE や FITS 等、係る分野固有のライブラリを用いて、研究・開発向けのデータ形式に直接アクセス可能である。その他天体表面の反射特性の模擬や、関連する科学データに基づいた天体のマップ作製等、必要に応じて機能追加しつつ現在に至っている。

宇宙分野に特化したツールとしては、PANGU[2]が有名である。PANGU は試験用途等も含めて、各種専門的な機能が提供されており、同分野において信頼性の高いモデリングやレンダリングが可能となっている。これに対して筆者らは、関連プロジェクトの利用環境対応や機能追加の柔軟性等の観点から内製ソフトウェアの開発・管理を行ってきた。

しかしながら内製ソフトウェアを長期間維持管理するための体制は十分とは言えない。その理由の一つとして、係るソフトウェアの開発や維持管理には、可視化等の知識の他に宇宙科学分野の専門知識も必要であるため、要員の確保が難しいことも挙げられる。結果として、内製ソフトウェアに蓄積されたノウハウを、広く可視化環境の利用者と容易に共有できる状況には至っていない。

このような状況に鑑みると、広く普及している汎用の CG ソフトウェアに機能追加して内製ソフトウェアと同等の機能を提供することを検討する余地があると考えられる。この場合、レンダリングエンジン等の基本的な機能はそれぞれの CG ソフトウェアに依存し、宇宙科学分野に関連した付加機能に焦点を当てた開発を行うことで開発の効率化が期待され、また一般用途も含めて、係る機能の利用者拡大にも繋がると思われる。

3. 汎用 3D CG ソフトウェアの可能性について

検討対象となりそうなソフトウェアについて以下に概観する。本節では筆者らが見分けたソフトウェアを中心に述

べる。これら以外にも様々な CG ソフトウェアが存在するが、その詳細は、ここでは割愛する。

3.1 POV-Ray (GUI を持たない汎用ソフトウェア)

それ単体では、モデリングやレイアウト等の GUI を持たない、スクリプトベースのレイトレーシングソフトウェアである。レンダリングの詳細はスクリプトで記述でき、コマンドラインから実行できるため、バッチ処理で科学データの可視化を行うには適している。また軽量の演算等は、スクリプト中で実行可能である。

その一方、スクリプトの文法は独自のものであり、他のスクリプティング言語との互換性はない。選択できる表面材質は限られており、プラグインやシェーディング言語等を用いた機能拡張は困難である。

3.2 GUI ベースの汎用ソフトウェア

例えばフリーのソフトウェアである Blender の場合、Python や OSL (Open Shading Language) を用いた機能拡張が可能である。Blender と OSL の組み合わせは既に宇宙探査関連の研究 ([3]等) にも使われており、宇宙科学データの取扱とも親和性は高いと考えられる。

他にも Python や OSL を利用できる 3D CG ソフトウェアは複数存在しており、これらに共通に適用可能な実装部分と、各ソフトウェア固有の実装部分に分けることができれば、宇宙科学データの可視化に係るプログラミングの一部を効率的に共有できる可能性があると考えられる。

3.3 リアルタイムレンダリング

GUI 等を用いたリアルタイムレンダリング用途としては例えば WebGL に基づいた JavaScript ベースの API が幾つか公開されており、筆者らも将来計画に係る可視化に利用した[4]実績がある。またゲーム制作環境 (Unity や Unreal Engine 等) も検討候補に挙げられる。

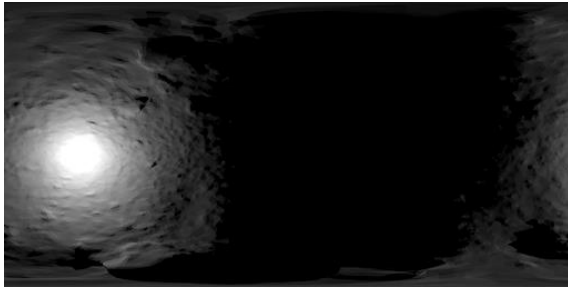
これらの環境は GUI の構築が容易であり、また GPU を用いることでリアルタイム性も高い。シェーディング言語を用いて表面材質をカスタマイズすることも可能である。

前述の OSL や HLSL、GLSL 共に類似した文法のシェーディング言語であるため、相互の移植性を考慮した開発を行うことで、さらに適用対象が広がるのが期待されるが、その結果として類似機能のバリエーションが増えることには、動作検証等のサポート関連コストが高まる懸念もある。

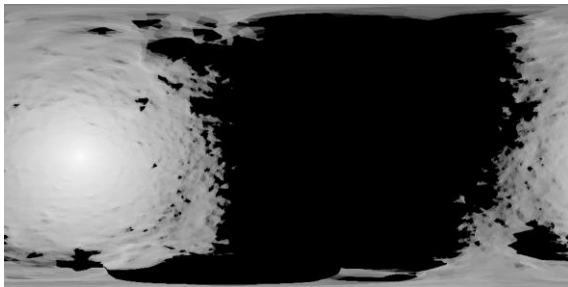
留意点として、ここでレンダリングに想定されている浮動小数点演算は単精度であり、宇宙科学データの精密な表現が必要な場合には相応の注意が必要である。また影の描写が必須となるような用途ではその精度に課題が残る。

4. 汎用 3D CG ソフトウェアとの比較事例

図 1 に、内製ソフトウェアを用いた全球マップの例を示す。これは、ある条件下での小惑星[5]上の日照・日陰の状況を緯度・経度 (Equirectangular) 表示したものである。

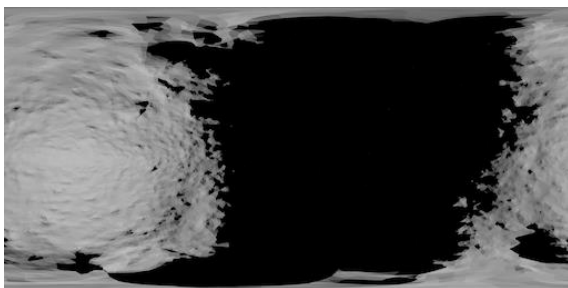


リニアスケール表示



log スケール表示

図 1 日照・日陰マップのレンダリング(内製ソフト)



リニアスケール表示

図 2 日照・日陰マップのレンダリング(POV-Ray)

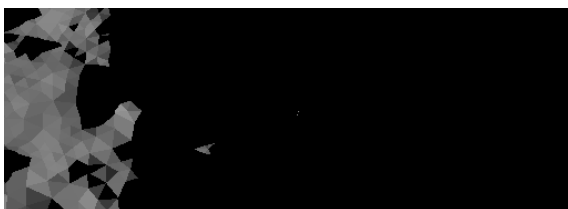


図 3 レンダリング時のエラー例 (画像中央の白点)

同等の処理を POV-Ray で実行した例を図 2 に示す。日照と日陰の判定は概ね実現できているが、実行結果には、一部でエラーが見受けられた。図 3 にその例を示す。同図中央の 2 つの白点は本来、陰の部分であり、日照判定される領域ではない。このようなエラーはランダムに発生していると見受けられ、解決すべき問題と考えられる。

また図 1 では小惑星等に特有の地質に対応した表面材質も割り当てられている。これを一般の CG ソフトに適用するためには、シェーディング言語等による表面材質のカスタマイズで対応することになると考えられる。

この例に限らず、汎用 CG ソフトウェアを用いる場合、探査機や天体等の位置・姿勢計算等のライブラリを直接利用することが困難なことも課題のひとつである。そのようなケースでは、予め各座標値や姿勢を計算する等の前処理が必要となる。

5. 課題

前節に述べた課題以外にも、汎用ソフトウェアを活用するにあたっての留意点が幾つか見受けられる。これらの中には、広報・普及の分野や一般のコンテンツ制作では許容範囲内と考えられるものも含まれており、必ずしも汎用 CG ソフトウェアへの機能追加を妨げるものではないが、各種ミッションや研究用途等、精密な結果を必要とする場合に留意すべき課題であると考えられる。以下に例を示す。

ひとつには、一定以上離れたオブジェクトは正常にレンダリングされない現象が見受けられる。日常的なレンダリングでは、遠距離のオブジェクトを無視することが合理的である場合も想定されるが、宇宙科学分野で用いるには、多くの場合、設定された全てのオブジェクトが適切にレンダリングされる必要がある。またレンダリングの精度やアルゴリズム等が必ずしも公開されているとは限らないため、前節の例に限らず、レンダリング結果の信頼性について、都度検証する必要があると考えられる。

宇宙機搭載光学機器の視野シミュレーション等では、大規模な形状モデル等が用いられる場合があるが、汎用の CG ソフトウェアで扱える形状モデル等の規模は、その環境毎に評価する必要がある。

6. おわりに

技術的にコンテンツ制作支援の枠組みを広げる手段について考察した。

内製ソフトウェアは要望に応じて柔軟な機能追加等が可能である反面、開発体制や維持管理に課題を抱えている。

汎用 CG ソフトウェアの利用はレンダリングエンジンの管理コスト低減が期待されることであり、内製ソフトウェアで培われた機能を適用させることで、それら機能のサポート範囲を拡大できる可能性がある。一方で、そのレンダリング結果の妥当性については十分吟味する必要がある。

今後は、本稿に述べた考察等に基づいて、具体的な機能拡張等を検討することになる。内製ソフトウェアが必要な場面も残るため、追加の開発負荷が発生することになるが、複数手段での実行結果を相互比較することによる妥当性検証等、信頼性向上に繋がる側面もあり、相互補完的な開発を目指すものである。

参考文献

- [1] 三浦昭, 武井悠人, 山口智宏, 高橋忠輝, 佐伯孝尚, “「はやぶさ 2」ハードウェアシミュレータに係るレイトレーシングソフトウェアの開発と評価”, 宇宙科学情報解析論文誌, Vol.9 (2020).
- [2] Martin, I & Dunstan, M, PANGU v6: Planet and Asteroid Natural Scene Generation Utility, 2021.
- [3] Issa A. D. Nesnas, et al., Autonomous Exploration of Small Bodies Toward Greater Autonomy for Deep Space Missions, Front. Robot. AI, 01 November 2021.
- [4] 三浦昭, 松崎恵一, 石田貴行, 田中雅光, 井上弘士, “SFQ 伝搬回路のインタラクティブ可視化”, 2022 年電子情報通信学会総合大会, 2022.
- [5] 国立研究開発法人宇宙航空研究開発機構 宇宙探査イノベーションハブ, “リュウグウの簡易形状モデルの公開”, 2018, <https://www.isas.jaxa.jp/topics/001948.html>(2022 年 6 月 24 日閲覧)

† 宇宙航空研究開発機構宇宙科学研究所

Institute of Space and Astronautical Science, Japan Aerospace Exploration Agency