

# 文化財の3DCG再現におけるレンダリング環境の性能評価

## Performance evaluation of rendering environment for 3DCG reproduction of cultural properties

望月宏祐\* 田中法博\*\*

Kosuke MOCHIZUKI Norihiro TANAKA

### 概要

本稿では、大型で複雑な文化財を3DCG再現するためのレンダリングに適した環境の構築を検討するため、小諸城とその城下町の3DCGモデルを対象にレンダリングの性能評価を行う。検証の対象とするレンダリングアルゴリズムはオフラインレンダリングを想定したレイトレーシング法とする。性能評価のシステムの対象は、CPUによる性能評価とGPGPUを用いた性能評価を行う。特にGPGPUによる計算はNVIDIA社のGPUに搭載される最新のレイトレーシング専用ハードウェアであるRTコアの性能評価も行う。さらにレンダリング用のシステムを構築する上では、アプリケーションソフトウェアだけでなく、低レベル層に位置するOSの性能についても検証を行う。最後にオフラインレンダリング環境におけるOSの違いとしてWindowsとLinux環境それぞれを構築して比較検証を行う。

キーワード:文化財, CPU, GPU, 3DCG, レイトレーシング法, レンダリング速度

### 1. はじめに

近年、大型の文化財のデジタルアーカイブや復元に3DCG技術が用いられるようになってきている[1]。3DCGで文化財を映像再現する場合、ユーザとの対話処理に応じてリアルタイムに3DCG生成する方法とリアルタイム性は犠牲にして高品質な映像生成を目指すオフラインレンダリングという手法が映像の使用目的に応じて使われる。

本研究で扱う映像は、オフラインレンダリングを対象とする。オフラインレンダリングで生成された映像はBlu-ray、DVDといった保存用のメディアでも扱える。

こういった高品質な3DCG映像を生成するためには光反射や光伝搬を精密に計算できるレイトレーシング法[2]をベースとしたレンダリング手法が有効とされている。しかし、この方法によるレンダリングはZバッファ法など他のレンダリング手法に比べて膨大な計算が必要となる。

これまで我々は、レイトレーシング法をGraphics Pricessing Unit(GPU)をGeneral Purpose GPU

(GPGPU)として用いることで、分光ベースのレイトレーシングに基づいて高速にレンダリングする技術を開発した[3]。しかしここでは最大3つの球型モデルと1枚の平面モデルを用いたレンダリングの検証に留っており、リアルタイムでのレンダリングが可能になったものの、実際の文化財を想定した複雑な形状のレンダリングには至っていない。

近年では、複数の2次元画像から3次元形状を推定するフォトグラメトリ技術やレーザー測量技術の発展に加え、ドローンによって自由度の高い視点からの広範囲に及ぶ3次元計測が可能になってきている。しかし、こういった計測から得られたデータの量は非常に膨大になることから、レイトレーシング法と組み合わせた3DCGレンダリングには非常に多くの時間を要するといった問題がある。

我々は大型の文化財の3DCG復元の研究として、現在の長野県小諸市にかつて存在した小諸城を対象に活動を行ってきた[4]。小諸城は現在、小諸城址・懐古園となっており、大手門や三の門といった国の重要

文化財が残されているが、本丸や二の丸、北の丸、南の丸など多数の建物が消失している。しかし、江戸期に石倉芳隣によって描かれた110点以上の古文書が残されていることから、これらの情報と現在の計測情報を基に、小諸城と城下町の3DCG復元を試みている[5][6]。図1は大手門の古文書一枚であり、建造物の立面を示し、寸法や当時の様子が記載されている。こういった古文書が建造物ごとに残されている。

本研究では、大型で複雑な文化財を3DCG再現するためのレンダリングに適した環境の構築を検討するため、小諸城とその城下町の3DCGモデルを対象にレンダリングの性能評価を行う。

性能評価のシステムの対象は、CPUによる性能評価とGPGPUを用いた性能評価を行う。特にGPGPUによる計算ではNVIDIA社製のGPUに搭載される最新のレイトレーシング専用のハードウェアであるRTコアの性能評価も行う。

さらにレンダリング用のシステムを構築する上では、アプリケーションソフトウェアだけでなく、低レベル層に位置するOSの性能が大きな影響を与えることが知られている。

そこで本研究では、オフラインレンダリング環境におけるOSの違いとしてWindowsとLinux環境それぞれを構築して比較検証を行う。

## 2. 歴史的建造物と地形の3DCGモデル

本研究のレンダリング環境の性能評価には3DCGモデルとして小諸城と城下町を用いる。このモデルは江戸期の古文書を基に復元している情報である。この地形情報は古文書に加え、現在の地形の計測情報を基にして復元している[6][7]。図2はワイヤーフレームで示す本研究の検証に用いる3次元形状の全体像である。

本研究でレンダリング環境の性能評価に用いるレンダリングのシーンは、国の重要文化財とされている大手門と、現在は取り壊されて存在していない本丸とする。本丸は小諸城で最も敷地面積が広い建造物である。図3は大手門の3次元モデルをワイヤーフレームで示したものである。図4は本丸の3次元モデルをワイヤーフレームで示したものである。

## 3. レイトレーシング法を用いたレンダリング

本研究では、文化財材質の高品質なCG再現を行うため、詳細に光反射の計算ができるレイトレーシング

法を採用している。レイトレーシング法は、視点から画素数と同数の光線(Ray)を放ち、その光線と物体の交差判定を行う。光線が物体に当たった場合には物体の材質に応じた反射や屈折、透過をする。さらに物体間の相互反射を含む光線の経路を求めることで光の光反射や透過、影をCGとして表現する。この手法は文化財を構成する材質の質感を精密に再現できるという利点があるが、計算負荷が高く計算時間が膨大にかかるという問題がある。

図5はレイトレーシング法の概略図である。本研究でレンダリング対象としている小諸城は複雑な建造物であり、今回実験で用いた大手門や本丸は室内外の壁面や扉、柱などが多数存在することから膨大な計算量になりレンダリング時間が大幅に増えることが予想される。

## 4. 3DCGレンダリング環境の構築

本研究で、レンダリングに用いるPCのハードウェアは表1の構成とした。ソフトウェアとしてOSはMicrosoft Windows 10とLinux Mint 20.3とする。Linux MintはオープンソースのOSであるLinuxのディストリビューションの一つであり、Ubuntuをベースとしている。Windowsと比較して、レンダリング時はハードウェアリソースを効率よく使用できることが知られており、レンダリングの高速化が期待できる。本研究では同じ仕様のSSDを二つ用意し、それぞれにWindows10とLinux Mint 20.3をインストールする。

次に、本研究で用いる3DCGレンダリング用ソフトウェアはBlender 3.0とする。Blender3.0はオープンソースのソフトウェアであり、レイトレーシング法をベースとしたレンダリングアルゴリズムが実装されている。また、Blender3.0はCPUを用いたレンダリングに加え、CUDAによるGPUを用いたレンダリングに対応している。CUDAは、NVIDIA社のGPUでCUDAコアと呼ばれるコアを用いて汎用の並列計算を可能にさせるプラットフォームである。CPUレンダリングと比較してレンダリングの高速化が期待できる。

さらに、Blender3.0はNVIDIA社製のRTXシリーズのGPUに搭載されているRTコアによるレンダリングに対応している(BlenderではOptiXと呼ぶ)。

本研究で用いるGPUは、CUDAコアを2304基搭載し、レイトレーシング処理向けにRTコアは36基搭載している。これらの専用ハードウェアを利用することにより、CUDAもしくはOptiXによるレンダリングがCPUレン

ダリングよりも高速化できる。

表1.レンダリング用PCのハードウェア構成

種類	各ハードウェアの仕様
CPU	intel Core i7-8700 6コア8スレッド, 最大4.60GHz
メインメモリ	DDR4 2400Mhz 64GB
GPU	NVIDIA RTX 2070 8GB
SSD	Western Digital Blue 1TB SATA接続

## 5. レンダリングの検証実験

### 5.1 実験条件

本研究では、レンダリング時間を計測することでレンダリング環境の性能を検証する。実験に用いるPCは4章で述べたシステムを使用する。レンダリング時にOSはWindows 10 とLinux Mint20.3 のそれぞれ同条件でレンダリングを行う。レンダリングに用いる3次元モデルは2章で述べた小諸城とその城下町を含む3次元モデルとする。このモデルは、建造物と地形情報、一部の木々の情報を含んでいる。このモデル全体の頂点数の合計は6,836,451個となる。さらに詳細な地形や石垣の詳細な凹凸情報をもつ3次元モデルとした場合、メインメモリの不足によってレンダリングが不安定になるため、この実験ではメモリ不足が発生しない精度の3次元モデルを用いる。3次元モデルの各材質は、部位ごとに異なる光反射モデルとテクスチャ情報を与えている。地面はテクスチャ情報に加え、微細な凹凸の質感を再現するために法線マップ情報を与えている。

また、Blenderのレンダリングエンジンはレイトレーシング法をベースとしたCyclesを用いる。照明はHDR画像(15000×7500画素)を用いた環境マッピングによって表現する。レンダリング解像度はFull HD(1920×1080画素)と4K(3840×2160画素)とする。レンダリングの計算用ハードウェアはCPUのみを用いる場合、CUDAを用いたGPUレンダリング、OptiXを用いたGPUレンダリングの3パターンで時間差を検証する。

検証するシーンは小諸城の3次元モデルの中に存在する大手門と本丸とし、これらの建造物がそれぞれ映るカメラ視点でレンダリングをする。

### 5.2 実験結果

図6はレンダリング結果の大手門のシーンである。

図7はレンダリング結果の本丸のシーンである。建造物とその周りの環境に加え影の影響も再現できていることがわかる。

表2はWindows 10でCG解像度をFull HDとしてレンダリングした結果である。表3はLinux Mint 20.3でCG解像度をFull HDとしてレンダリングした結果である。表4はWindows 10でCG解像度を4Kとしてレンダリングした結果である。表5はLinux Mint 20.3でCG解像度を4Kとしてレンダリングした結果である。

この結果から、まず、すべての条件において、大手門よりも本丸のシーンのレンダリング時間が長くなった。これは本丸の敷地面積が広く、3次元形状が複雑化していることからレイトレーシング処理に必要な計算負荷が増えたためと考えられる。次に、どちらのシーンに対してCPUを用いたレンダリングよりもCUDAを用いたレンダリングが速い結果となった。さらにOptiXによるレンダリングは今回のレンダリングで最も速い結果となった。次にOS別に検証する。今回のいずれのレンダリング結果においても、Windows 10よりもLinux Mint 20.3のレンダリングが高速であるという結果となった。この理由の一つとしては、Windowsと比較して、ハードウェアリソースを効率よく使用できたことが挙げられる。

次に、CGの解像度別に検証した結果、まずFull HDではOptiXによるレンダリングがCPUによるレンダリングに比べ最大で6倍高速化し、4Kは最大で10倍高速化した。また、2つのシーンに共通して、Linux MintとOptiXによる組み合わせが、最も速くレンダリングできることがわかった。

## 6. まとめ

本研究では、大型で複雑な文化財を対象とした3DCG再現のためのオフラインレンダリングに適した環境の構築を検討するため、小諸城と城下町の3DCGモデルを用いてレンダリング環境の性能評価を行った。今回の実験では、レンダリングに用いるOSの違いにより速度が変化する結果となった。またCPUを用いたレンダリングよりもGPUを用いたレンダリングが高速であるという結果となった。今回の実験では限られたハードウェアや3次元モデルでの検証となっているため、今後、さらに精密な3次元データなど大規模な情報をもつ文化財を対象としたレンダリング環境の構築に向けた検証をする必要がある。

謝辞

本研究を遂行するにあたって、小諸市の牧野和人氏に全面的にご協力をいただき、小諸市教育委員会からは小諸城の古文書と計測データの提供をいただきました。みすず総合コンサルタント株式会社様には3次元レーザー計測にご協力をいただきました。ここに深謝いたします。

参考文献

- [1] 池内克史、倉爪亮、西野恒、佐川立昌、大石岳史、高瀬裕、“The Great Buddha Project -大規模文化遺産のデジタルコンテンツ化-”、日本バーチャルリアリティ学会論文誌, Vol.7, No.1, pp.103-113, 2002.
- [2] Wallace, J. R., Cohen M. F., Greenberg, D. P.: A Two-Pass Solution to the Rendering Equation: A Synthesis of Ray Tracing and Radiosity Methods, Computer Graphics 21-4, pp.311-320, 1987.
- [3] 望月宏祐、田中法博、戸谷重幸、森川英明、三浦幹彦“分光レイトレーシング法に基づいた相互反射の色再現手法” 日本デザイン学会誌 デザイン学研究Vol.60, No.1, pp.11-20, 2013.
- [4] 櫻井千寛、望月宏祐、田中法博、“古文書と計測データに基づいた小諸城のCG 復元”, 日本デザイン学会誌デザイン学研究特集号Vol.24, No.3, pp.24-29,2017.
- [5] 小諸市誌編纂委員会、“小諸城城郭絵図”、小諸市教育委員会、1992.
- [6] 小井土和宏 望月宏祐 田中法博、“小諸城周の地形情報の3DCG復元”,日本色彩学会画像色彩研究会研究発表会,pp.5-9,2018.
- [7] 小諸市誌編纂委員会,”小諸市誌 歴史篇(三)近世史”,小諸市教育委員会、1991.

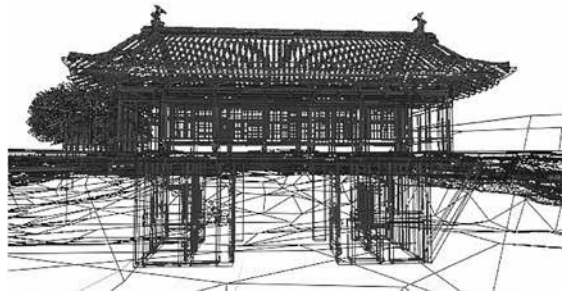


図3. 大手門のワイヤーフレーム

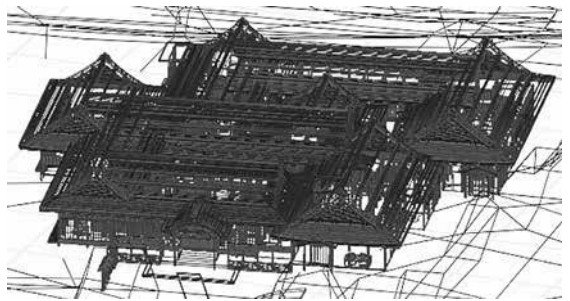


図4. 本丸のワイヤーフレーム

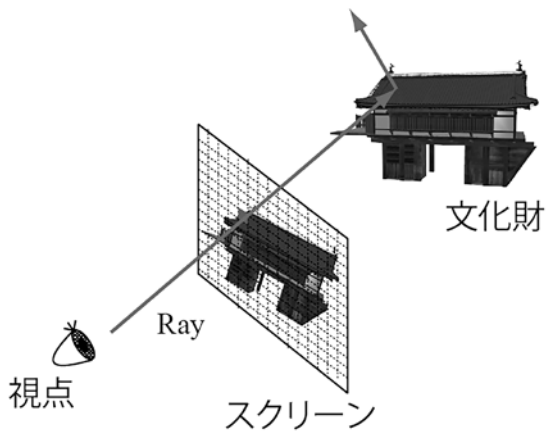


図5. レイトレーシング法の概略図

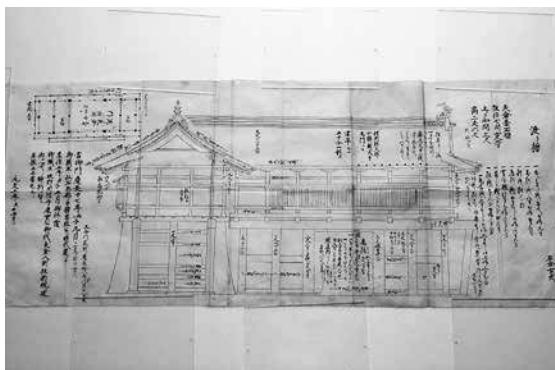


図1. 小諸城大手門の古文書  
(1740年 小諸市教育委員会提供)

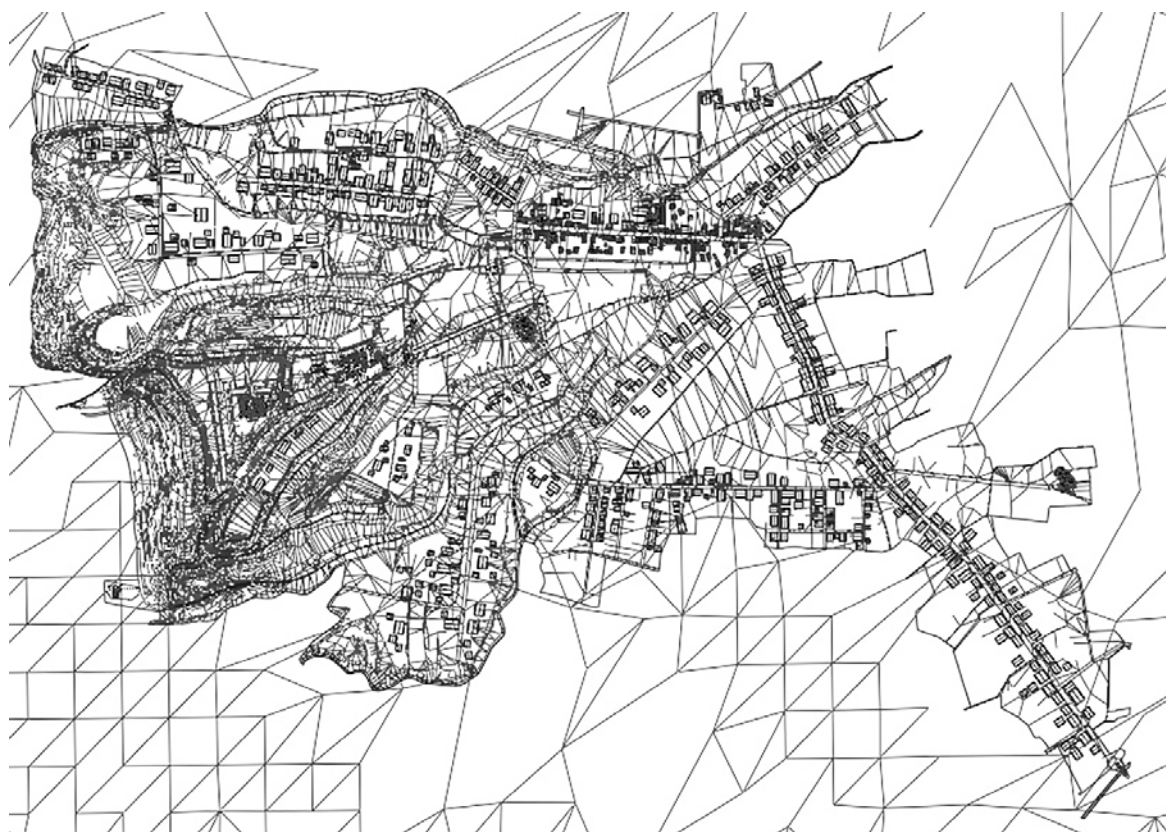


図2. 小諸城と城下町の三次元モデル全体のワイヤーフレーム



図6. 大手門のレンダリング結果(3DCG)



図7. 本丸のレンダリング結果(3DCG)

表2. レンダリング時間(Windows 10 Full HD)

シーン	CPU	CUDA	OptiX
大手門	2分43秒	48秒	33秒
本丸	3分51秒	53秒	35秒

表3. レンダリング時間(Linux Mint 20.3 Full HD)

シーン	CPU	CUDA	OptiX
大手門	2分25秒	41秒	29秒
本丸	3分21秒	47秒	31秒

表4. レンダリング時間(Windows 10 4K)

シーン	CPU	CUDA	OptiX
大手門	10分53秒	1分53秒	1分19秒
本丸	15分10秒	2分25秒	1分25秒

表5. レンダリング時間(Linux Mint 20.3 4K)

シーン	CPU	CUDA	OptiX
大手門	9分50秒	1分42秒	1分12秒
本丸	13分27秒	2分10秒	1分19秒