

高フレームレートの動的投影に向けた時間的再利用による パストレーシングのデノイズに関する検証

宮本 駿一^{*1}

野元 貴史^{*1}

渡辺 義浩^{*1}

Abstract — 近年、拡張現実の分野で動的な物体に投影を行うダイナミックプロジェクションマッピング (DPM) が注目されてきている。投影する映像を描画する方法として、写実的な表現ができるパストレーシングを用いる手法が提案されている。しかし、パストレーシングは計算時間が短いと描画される映像にノイズが発生する問題がある。DPM では映像を高速に投影することにより知覚的に各画像が積分され、ノイズが低減される効果があるが、実行時間の短さにより発生するノイズを完全に除去することは困難であった。そこで本稿では、コンピュータグラフィックス (CG) 分野のノイズ低減手法を高速化し、高速投影によって知覚されるノイズを低減することを提案する。

Keywords : パストレーシング, 高速プロジェクタ, 投影型拡張現実

1 はじめに

近年、動く物体へ投影を行うダイナミックプロジェクションマッピング (DPM) の研究が進んでいる。同事例では、観測者に対象物体と投影映像の位置のずれを知覚させない必要があるが、映像が物体に対して約 6 ms 遅れると、人間は映像の遅れを知覚するとされている [1]。同要請を満たすためには、センシング、グラフィックスレンダリング、投影などの処理を 500-1000 fps でそれぞれ実行することが必要である [2]。このように、レンダリング時間が通常のグラフィックス分野での要請に比べて短いことが、DPM の表現を高める上での制約となっている。

一方、CG の分野では、物理法則をベースにしたレンダリング手法が注目されている。この一つが大域照明の表現が可能なパストレーシングである。パストレーシングは、光の反射をシミュレーションするレイトレーシングを応用したもので、視点から出た仮想的なレイが光源にあたるまでの反射を計算する手法である。この手法はランダムにレイの反射方向のサンプリングを行う。そのため、図 1 (a) に示すように、画素毎のサンプリング数 samples per pixel (spp) が少ない状況下ではレンダリング画像にノイズが発生する。このため、ビデオゲームのようリアルタイムに描画を行う状況下では、spp を十分に増加させられず、ノイズの発生を抑えることが困難であった。さらに、DPM ではレンダリング時間がより短いため、より多くのノイズが発生する。

上記の課題に対し、Nomoto らは、上記のパストレーシングを用いたリアリスティックプロジェクションマッピングを提案した [3]。同手法は、ノイズを含むレン

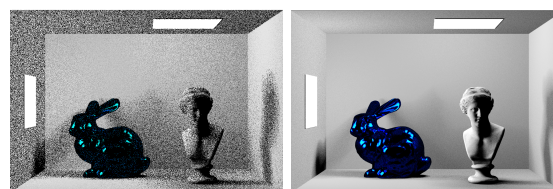


図 1 異なる spp のパストレーシングでレンダリングした画像。左から順に、(a) 1 spp, (b) 100 spp.

Fig. 1 Images rendered by path tracing with different spp. From the left, (a) 1 spp, (b) 100 spp.

ダリング画像を約 1000 fps で投影し、物体への投影遅延を最小限に抑えて DPM を行う。さらに、ノイズを含む画像を運動に応じて高速に更新することによって、人間の目に本来の画像よりもノイズが減少したように知覚させることを提案した。この効果は、人間が画像を 1 枚ずつ識別できる速度より速く投影画像を更新すると、人間の目には複数の画像が合成されて知覚されることを利用している。しかし、実行時間の制約から、画像 1 枚あたり 1-3 spp 程度のパストレーシングにとどまっていた。このため、ノイズの完全な除去は達成されておらず、依然としてノイズが知覚されることが問題となっていた。

一方、CG の分野では、リアルタイムレンダリングにパストレーシングを用いるためのノイズ低減手法が提案されている [4, 5]。しかし、従来のノイズ低減手法は、高速なものでも数 ms の処理時間を要する。DPM にこのような手法を用いた場合、この処理時間分だけシステム全体の遅延が増加し、投影の遅れが知覚される要因となる。

そこで本稿では、DPM の高フレームレートの特性

^{*1}東京工業大学

を利用し、高速に計算可能なノイズ低減手法を提案する。また、約 1000 fps の投影を介した実験により、ノイズ低減の検証を行った。

2 関連研究

2.1 リアルタイムレンダリングにおいて画像を時間的に再利用する手法

リアルタイムレンダリングでは、時間的に連続でレンダリングされる画像間に多くのコヒーレンスがあることが注目され、利用可能性が検討されてきた [6]。例として、時間的に画素内でのサンプル位置を変化させてフレームを足し合わせ、空間的なアンチエイリアシングを行う手法がある [7]。また、時間的なコヒーレンスを活用して前フレームのシェーディング結果の再利用を行い、レンダリング時間を減らすことも研究されている [9]。実際の再利用方法として、Nehab らは、現在フレームと前フレームの画素位置の対応を画像座標上におけるモーションベクターとして計算し、シェーディング結果を再利用することを提案した [8]。このモーションベクターの計算手法は、前フレームを時間的に再利用する多くの手法のベースとなっている。しかし、同手法はシェーディング計算を行う点以外の物体の影響を計算できない。そのため、視点に依存する鏡面反射や他の物体による影、遮蔽がある場合にアーティファクトが発生する問題があった。

2.2 リアルタイムパストレーシングのノイズ除去

リアルタイムパストレーシングのサンプリング数を増やさずにノイズ除去する手法が研究されている。Schied らは空間的なコヒーレンスと時間的なコヒーレンスの両方を生かしてレンダリング画像をフィルタリングする手法を提案した [4]。この手法は解像度 1280×720 の 1 spp の画像を約 5 ms でノイズ除去し、60 fps でのリアルタイムパストレーシングを行うことができる。しかし、同手法は前述した Nehab らのモーションベクターを用いているため、物体に映る影が正しい位置から離れる問題があった。

Zeng らは、Nehab らの手法におけるアーティファクトの問題を、レイトレーシングの 2 次反射以降のレイの計算や複数のモーションベクターを用いて除去する手法を提案した [5]。しかし、影の過剰なブラーや、曲面の鏡面反射成分におけるアーティファクトの発生などの課題があった。

これらの手法は、数 ms の実行時間でノイズを除去することができるが、DPM においてはシステムの遅延を増加させる要因となる。このため、本稿では、高フレームレート投影ではフレーム間でのシーンの変化が微小であることを利用する。これにより、1000 fps でパストレーシングを行いながらノイズを低減させ、

投影遅延と Nehab らの手法の問題を同時に解決する。

2.3 ダイナミックプロジェクションマッピングにおける写実的な表現

Nomoto らは、パストレーシングを DPM のレンダリングに用いることで、知覚的にノイズが低減したように提示できることを提案した [3]。

3 高フレームレートのレンダリングを利用した時間的に画像を再利用するノイズ低減手法

3.1 手法概要

前節で述べたように、フレーム間の画素の対応を求めるために Nehab らの手法 [8] をベースにしたノイズ低減の手法が多く使われている。これにより異なるフレーム間で対応する画素値を足し合わせ、ノイズを平均化している。しかし、この手法はアーティファクトが発生する問題がある。このアーティファクトは、シェーディングを行う点以外の物体や視点によるシェーディングの変化を計算できないことに起因する。しかし、DPM においては、約 1000 fps の高フレームレートでレンダリングを行うため、フレーム間のシーンの変化が微小である。そのため、フレーム間のシェーディングの変化も小さくなり、前述の問題による影響が小さくなると考えられる。このことから本稿では、DPM における投影画像のノイズ低減に Nehab らの手法をベースにした時間的にシェーディング結果を再利用する手法を提案する。

提案手法の流れを図 2 に示す。パストレーシングによって現在フレームの画素値を求めた後、モーションベクターの計算を行う。このモーションベクターを用いて前フレームへの再投影を行い、画素に表示される物体が同一かどうかを確認する。同一であれば足し合わせ、異なる場合は保存している前フレームの結果を破棄し、現在フレームの計算結果をそのまま出力する。パストレーシングから出力までの処理を各画素について行い、1 枚の画像として次フレームで利用するためのバッファに保存する。その後、ガンマ補正やトーンマッピングなどのポストプロセスを行い、レンダリング画像として出力する。



図 2 手法概要図

Fig.2 Overview of our method

3.2 時間的再利用

Nehab らの手法 [8] は、前フレームの再利用性を向上させるため、現在フレームと前フレームの画素位置

の対応を計算する。以下でその方法を説明する。

i フレームの画像座標 X_i に対応するモデル座標 $\mathbf{x}_{m,i}$ を求められる。 $\mathbf{x}_{m,i}$ を $i-1$ フレームへ投影することで、対応する画像座標 X_{i-1} を得る。この2点の画像座標からモーションベクターが得られる。一方、パストレーシングでは、視点から画像座標 X_i の画素に向けて射出されたレイの衝突点のモデル座標 $\mathbf{x}_{m,i}$ が直接取得できる。この時、後述する画素ごとの足し合わせを行うか決定するパラメータとして、レイの衝突した物体の ID をバッファに保存しておく。

得られた再投影点 X_{i-1} を用い、フレーム間の画素の対応づけを行う。対応づけられた画素同士を足し合わせる前に、 X_i と X_{i-1} でそれぞれ保存した衝突物体の ID が一致するか確認する。一致する場合は現在フレームと前フレームを以下の指数移動平均の式で足し合わせる。一致しない場合は現在フレームの画素値をそのまま利用する。

$$\bar{c}_i(X_i) = \alpha \cdot c_i(X_i) + (1 - \alpha) \cdot \bar{c}_{i-1}(X_{i-1})$$

ここで、 $\bar{c}_i(X_i)$ は i フレームにおける画素 X_i の足し合わせた後の画素値、 $c_i(X_i)$ は i フレームの画素 X_i のパストレーシングの計算によって得られた画素値である。 α が小さいほど、より多くの過去のフレームが足し合わされるので、ノイズがより多く低減する。一方で、鏡面のハイライトや影の遅延などのアーティファクトが発生しやすくなる。 α が大きくなるほど、現在フレームの割合が大きくなり、 $\alpha = 1$ の場合は足し合わせを行わないパストレーシングと同一の結果になる。

上記の方法で計算されるモーションベクターは、静的なシーンの場合にはすべての画素において 0 となる。また、シーンの物体の移動量が小さい場合、計算したモーションベクターは 0 に近づき、1 画素の幅より小さくなる。全画素についてこれが成り立つと考えると、 $X_i = X_{i-1}$ とし、前フレームにおける同一画像座標の画素値をそのまま足し合わせればよい。フレーム間のシーンの変化が小さい DPM では、上記の仮定が成り立つ可能性があると考えられる。この仮定が成立するかどうかについても検証するため、4 章では、モーションベクターの計算を行う場合と行わない場合の両方について実験を行った。

3.3 知覚積分との組み合わせ

DPM では、約 1000 fps の高フレームレートで投影を行うため、投影される各画像を観測者は区別できず、複数枚が積分されて知覚される。そのため、Nomoto らの手法 [3] では、本来の画像よりもノイズが低減されて知覚される。しかし、この効果は本稿で提案する手法と重複する可能性がある。知覚的に画像が積分される際には、フレームごとに同一の画像座標の画素あ

るいは近傍の画素が足し合わされていると考えられる。一方、本稿の提案手法はフレーム間の移動量が小さいことを利用し、近傍の画素値を多く足し合わせることで、画像のノイズを低減するため、知覚と同一の画素をノイズ低減に利用している。

しかし、本稿の提案手法は描画する物体が一致している限り画素の足し合わせを続けるため、足し合わせる前フレームの画像は視覚より多くの過去の画素情報を保持すると考えられる。そのため、知覚積分が起こる状況下においても、本手法による画像のノイズ低減効果が得られると期待される。

4 実験

4.1 投影画像におけるノイズ低減効果の評価

提案手法の投影画像に対するノイズ低減効果を調べるため、画像評価指標である SSIM と PSNR を用いてレンダリング画像の比較を行った。レンダリングするシーンは、直方体の内部に 2 つの Stanford bunny (以下、bunny とする) を設置し、上部に光源を配置した静的なシーンとした。静的なシーンの場合、提案手法にモーションベクターを用いる場合と用いない場合で同じ結果が得られる。そのため、本節と 4.2 節では、モーションベクターの計算を省略し、同一画像座標の画素を足し合わせる方法を用いた。

まず、1-20 spp のパストレーシングのみ、1 spp のパストレーシングと提案手法を用いた場合のそれぞれについて画像をレンダリングした。これらの画像について、1000 spp でレンダリングした参照画像に対する SSIM と PSNR を計算した。計算した PSNR と SSIM を比較し、パストレーシングのみでレンダリングした場合に、1 spp 画像に提案手法を適用した画像と最も評価が近くなる spp を求めた。これにより、静的なシーンにおける投影画像のノイズ低減効果を評価した。 α は 0.2 とした。

レンダリングした画像と PSNR、SSIM の計算結果をそれぞれ図 3 と表 1 に示す。その結果、レンダリング画像の時点では、提案手法は 9 spp のパストレーシングのみでレンダリングした画像と同程度の画質が得られた。

4.2 静的シーンを高速投影した際に知覚されるノイズ量

静的なシーンのレンダリング画像をスクリーンに投影し、カメラで撮影した。これにより、投影時に現れるノイズ量を確認した。レンダリングとシーンの設定は 4.1 節と同様にした。

実験システムを図 5 左に示す。実験に使用した計算機は CPU に Xeon Gold 6136 3.0 GHz dual, GPU に NVIDIA GeForce RTX 3090 を搭載したものである。

表 1 図 3 の画像のリファレンス画像に対する PSNR と SSIM
Table 1 PSNR and SSIM of the image in Figure 3 against the reference image

	(a) 1000 spp (reference)	(b) 1 spp	(c) 1 spp with ours	(d) 9 spp
PSNR	-	22.29	31.66	31.67
SSIM	1.00	0.61	0.79	0.79

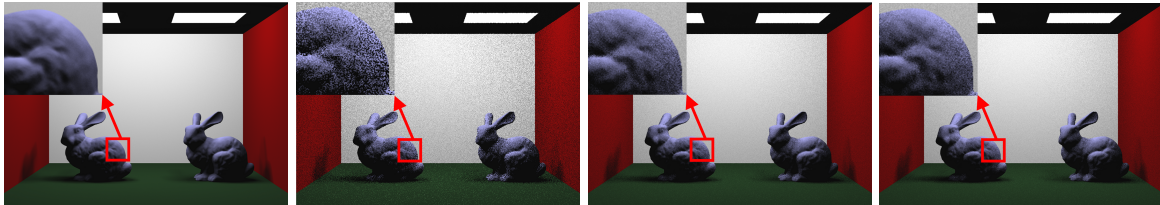


図 3 パストレーシングを用いて静的なシーンをレンダリングした画像。左から順に、(a) 1000 spp (リファレンス) (b) 1 spp, (c) 1 spp と提案手法, (d) 9 spp.

Fig.3 An image of a static scene rendered using path tracing. From the left, (a) 1000 spp (reference), (b) 1 spp, (c) 1 spp with ours, (d) 9 spp.

プロジェクタは 24 bit で最大 947 fps でのカラー投影が可能なプロジェクタ [10](解像度 1024×768) を用いた。高速プロジェクタはスクリーンと約 1 m 離して設置した。撮影に用いたカメラは SONY ILCE-7RM4 であり、焦点距離は 48 mm, シャッタースピードは 1/60 秒に設定した。このカメラを高速プロジェクタの上部に設置し、スクリーンの撮影を行った。

レンダリングした画像を 947 fps でスクリーンに投影し、投影されている映像を撮影した結果を図 4 に示す。提案手法は従来手法の 1 spp 投影に対してノイズ量が減少していることが確認できる。しかし、画像 1 枚を比較した場合に比べ、ノイズ量の差が小さくなっている。



図 5 システム構成。左から順に、(a) スクリーン投影のシステム、(b) ダイナミックプロジェクションマッピングのシステム。

Fig.5 System configuration. From the left, (a) system of screen projection, (b) system of dynamic projection mapping.

4.3 動的シーンを高速投影した際に知覚されるノイズ量

動的なシーンのレンダリング画像をスクリーンに投影し、複数の α に関してカメラで撮影した。これにより、動的シーンの投影時に現れるノイズ量および前フ

レームの再利用性の限界を確認した。また、モーションベクターの計算を行い足し合わせる手法 (以下、手法 1 とする) と、モーションベクターを計算せず、同一画像座標の点を足し合わせる手法 (以下、手法 2 とする) のノイズ低減効果についても確認した。

レンダリングしたシーンは、直方体の内部に対称な 2 つの bunny を設置し、上部に 2 つの光源を配置したものである。bunny には鏡面反射成分を含むマテリアルを設定した。シーン全体が左右対称になっており、2 つの bunny は回転しながら前後に移動する。実験システムは実験 4.2 と同様で、 α は 0.05, 0.2, 0.1 とした。カメラは実験 4.2 と同じものを用いたが、本実験ではシャッタースピード 1/60 秒で動画を撮影し、画像を切り出した。

bunny が移動するシーンを 947 fps でスクリーンに投影し、投影されている映像を撮影した。投影画像は 1 spp のパストレーシングを行った後、画像の右半分にはのみ手法 1 と手法 2 を適用してレンダリングしたものである。結果をそれぞれ図 7 と図 8 に示す。手法 1 と手法 2 のどちらにおいても、 α の値にかかわらず、左半分の 1 spp と比較して、ノイズが低減されている。一方、 $\alpha = 0.05$ に関しては、回転運動を行う水平方向に関してブラーが生じ、表面の形状が認識できなくなっている。また、どの手法と α の組み合わせに関しても、bunny の背中部分のハイライト部が 1 spp でレンダリングした左半分に対して大きくなっていることが確認できる。

今回の実験では、手法 1 と手法 2 の間ではどの α に関しても顕著な違いは生じず、類似した結果が得られた。このことから、DPM では、シーンの変化が微小であり、モーションベクターが 0 に近づく仮定が成り

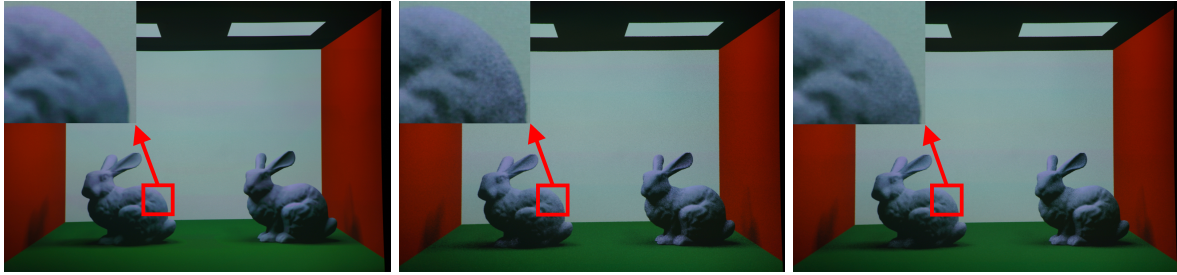


図 4 静的なシーンをレンダリングした画像をスクリーンに投影し、カメラで撮影した画像。左から順に、(a) 1000 spp (リファレンス) (b) 1 spp, (c) 1 spp と提案手法。

Fig. 4 An image of a static scene rendered and projected onto a screen and captured by a camera. From the left, (a) 1000 spp (reference), (b) 1 spp, (c) 1 spp with ours.

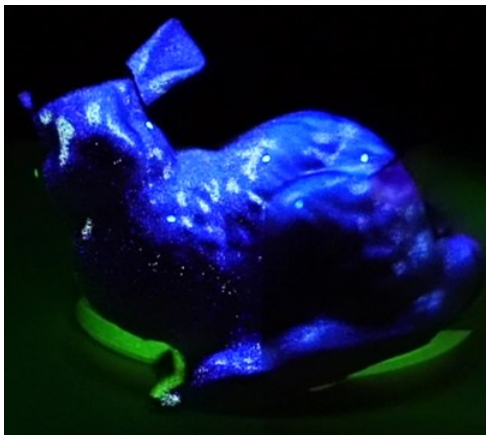


図 6 提案手法を用いた DPM. bunny に投影されている画像は、1 spp のパストレーシングを行った後に画像の右半分のみに手法 2 を適用し、レンダリングしている。 $\alpha = 0.2$ とした。

Fig. 6 DPM using our method. The images projected onto the bunny are rendered using the proposed method with only 1 spp path tracing on the left half and 1 spp path tracing on the right half. $\alpha = 0.2$ and no motion vector computation is performed.

立っていると考えられる。

4.4 提案手法を用いた DPM

また、本手法を用いてダイナミックプロジェクションマッピングを行うシステムを構築し、移動する実物体に対して追従して投影を行い、知覚される投影像を確認した。実験 4.3 の結果より、手法 1、手法 2 のどちらを用いても同様の結果が得られると考えられる。そのため、本実験ではより計算量の少ない手法 2 を使用した。

ダイナミックプロジェクションマッピングの実験システムを図 5 右に示す。計算機とプロジェクタは 4.2 節の実験と同じものを用いた。また、物体のトラッキングのためのカメラとして 10 台の OptiTrack Prime 17W を利用した。このカメラは 360 fps, 4.5 ms の遅

延で物体の位置姿勢が取得可能である。撮影に用いたカメラは実験 4.3 と同様にした。

実物体の bunny をプロジェクタに対して水平に移動させながら、947 fps で投影している様子を撮影した。その結果を図 6 に示す。立体物に投影を行った場合においても、提案手法によってノイズが低減されていることが確認できる。

5 考察

本手法を用いて、約 1000 fps でパストレーシングと投影を行うシステムを構築し、投影された映像を撮影して比較することで、ノイズが低減されていることを確認した。

一方、投影時のノイズ低減効果は、投影する前の画像で得られていたノイズ低減効果より小さい効果となった。これは、3.3 節で述べたように、知覚積分によるノイズ低減効果との重複によるものと考えられる。提案手法によって得られるノイズ低減効果の一部は既に知覚積分によって得られており、画像自体を比較した場合と比べてノイズの低減量が小さくなった。しかし、知覚積分で参照される過去フレームの画素数に対して、画像自体に足し合わされる過去フレームの画素数が多いと考えられるため、その差分が投影時のノイズ低減効果として現れている。

また、本手法を用いてノイズ低減を行った後でも、投影時にノイズを完全に排除するには至っていない。パストレーシングはシーンの位置によってノイズ量が異なり、ノイズ量が多い領域ではまだノイズが知覚される。

6 まとめ

本稿では、時間的に前フレームまでのレンダリング結果を再利用する方法を用いてダイナミックプロジェクションマッピングにおけるパストレーシングのノイズを低減する手法を提案した。シーンの移動が微小で

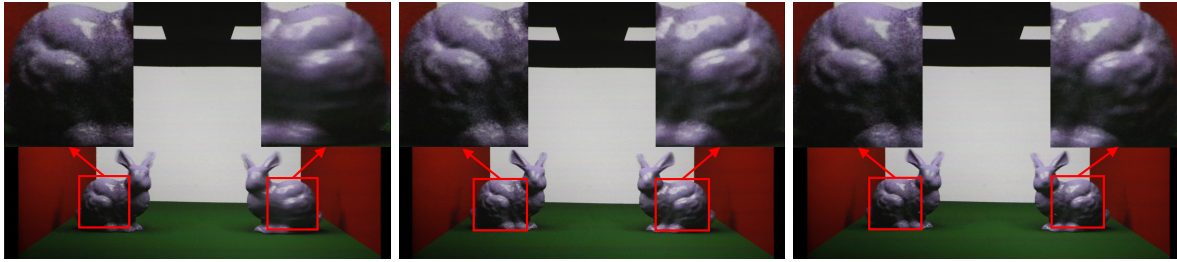


図7 動的なシーンをレンダリングした画像をスクリーンに投影し、カメラで撮影した画像。左右の bunny は回転しながら前後に移動している。画像全体で 1 spp のパストレーシングを行った後、右半分のみ手法 1 を適用しノイズを低減している。左から順に、(a) $\alpha = 0.05$, (b) $\alpha = 0.1$, (c) $\alpha = 0.2$.

Fig. 7 An image of a dynamic scene rendered and projected on a screen, and captured by a camera. The left and right bunnies are rotating and moving back and forth. The left half of the projected image is rendered using only 1 spp path tracing, while the right half is rendered using the proposed method with motion vectors after computing 1 spp path tracing. From the left, (a) $\alpha = 0.05$, (b) $\alpha = 0.1$, (c) $\alpha = 0.2$.

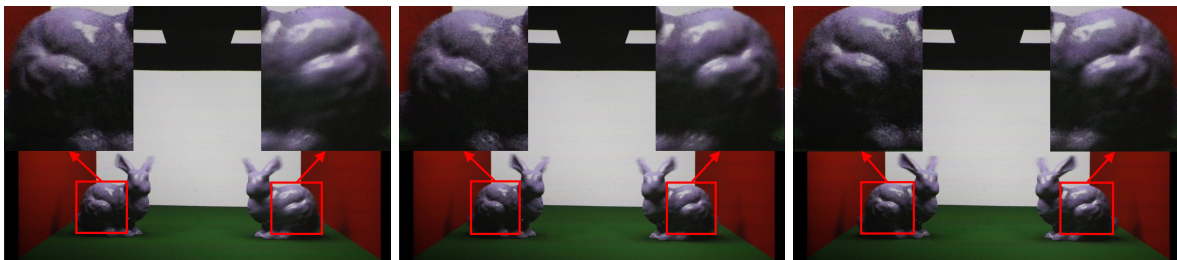


図8 投影画像の右半分に手法 2 を適用し、投影した画像をカメラで撮影したもの。モーションベクター以外の条件は図 7 と同様。左から順に、(a) $\alpha = 0.05$, (b) $\alpha = 0.1$, (c) $\alpha = 0.2$.

Fig. 8 The right half of the projected image is rendered by the proposed method, which omits the motion vector calculation, and the projected image is photographed. The conditions other than motion vectors are the same as in Figure 7. From the left, (a) $\alpha = 0.05$, (b) $\alpha = 0.1$, (c) $\alpha = 0.2$.

あることを利用して DPM の速度要請を満たしつつ効果が得られることを確認した。

参考文献

- [1] Ng, A., Lepinski, J., Wigdor, D., Sanders, S., and Dietz, P. Designing for Low-Latency Direct-Touch Input; UIST. 2012
- [2] Watanabe, Y. High-speed optical 3D sensing and its applications; Advanced Optical Technologies, 5(5-6), 367-376. 2016.
- [3] Nomoto, T., Koishihara, R., and Watanabe, Y. Realistic Dynamic Projection Mapping Using Real-Time Ray Tracing; ACM SIGGRAPH 2020 Emerging Technologies. 2020
- [4] Schied, C., Kaplanyan, A., Wyman, C., Patney, A., Chaitanya, C. R. A., Burgess, J., Liu, S., Dachsbacher, C., Lefohn, A., and Salvi, M. Spatiotemporal variance-guided filtering: Real-time reconstruction for path-traced global illumination; Proceedings of High Performance Graphics. 2017
- [5] Zeng, Z., Liu, S., Yang, J., Wang, L., and Yan, L. Q. Temporally Reliable Motion Vectors for Real-time Ray Tracing; Computer Graphics Forum, 40(2), 79-90. 2021
- [6] Scherzer, D., Yang, L., Mattausch, O., Nehab, D., Sander, P. V., Wimmer, M., and Eisemann, E. Temporal coherence methods in real-time rendering; Computer Graphics Forum, 31(8), 2378-2408. 2012
- [7] Karis, B. High-Quality Temporal Supersampling; In SIGGRAPH Courses: Advances in Real-Time Rendering in Games. 2014
- [8] Nehab, D., Sander, P. V., Lawrence, J., Tatarchuk, N., and Isidoro, J. R. Accelerating real-time shading with reverse reprojection caching; Proceedings of the SIGGRAPH/Eurographics Workshop on Graphics Hardware, 25-35. 2007
- [9] Mueller, J. H., Neff, T., Voglreiter, P., Steinberger, M., and Schmalstieg, D. Temporally Adaptive Shading Reuse for Real-Time Rendering and Virtual Reality; ACM Transactions on Graphics, 40(2). 2021
- [10] Watanabe, Y., and Ishikawa, M. High-speed and high-brightness color single-chip DLP projector using high-power LED-based light sources; Proceedings of the International Display Workshops, 3, 1350-1352. 2019