

分散協調型マルチプロジェクタ・カメラシステムによる 重複領域処理と色補償

三浦 和之^{*1} 竹内 正稀^{*1} 岩井 大輔^{*1}

Overlapping Region Blending and Color Compensation
in Distributed Cooperative Multi-Projector-Camera Systems

Kazuyuki Miura^{*1}, Masaki Takeuchi^{*1} and Daisuke Iwai^{*1}

Abstract — 本研究では、複数台のカメラを利用した分散協調型マルチプロジェクタ・カメラシステムにおける、重複領域処理と色補償手法について提案する。1台のカメラでは部屋全体などの広範囲な領域や遮蔽物に遮られた領域の投影画像生成処理を行うことが不可能であった。そこで、複数のカメラからの撮影により得られた情報を統合し、プロジェクタ間の重複領域処理と色補償を行うアルゴリズムを提案した。加えて、分散システムの利点を生かし、プロジェクタを追加する際にも、全体の再キャリブレーションを行うことなく、局所的な処理のみでスケラブルな拡張を可能にした。また、実験では提案したアルゴリズムを11台のプロジェクタと3台のカメラを用いて実装したシステムを用いて、その有効性を確認した。

Keywords : プロジェクションマッピング, 分散協調最適化, 拡張現実感

1 はじめに

プロジェクションマッピング (Projection Mapping: PM) は、実物体に映像を投影し、その色や質感といった見た目を変化させる技術である。これまで、PMは暗い環境での利用が一般的だった。これは、環境光が投影対象に重畳されることで映像のコントラストが低下し、鮮明な映像表示が困難となるからである。この問題は、製造現場における組立作業支援 [1]、医療作業支援 [2] など、明るい環境を必要とするPMの利用シーンを大きく制限してきた。

近年、この問題を解決するためプロジェクタを照明として利用する手法が活発に研究されている。Yasuiらは、1台のプロジェクタとレンズアレイを組み合わせた光線制御により、局所的な照明制御を実現した [3]。また、Takeuchiらは環境光を制御するため一般的な室内照明を複数台のプロジェクタに置き換え、投影対象以外の環境面に環境照明を再現する疑似環境照明技術を構築した [4]。さらに筆者らは、Takeuchiらの手法を発展させ、分散最適化に基づく色補償と境界処理による実環境照明再現手法 [5] を提案した。

しかし、これらの先行研究ではシステムの拡張性や複数の視点からの見えの補償といった未解決の課題が存在する。具体的には、Yasuiらの手法ではレンズアレイといった特殊なデバイスを必要とするうえ、複数台のプロジェクタ利用に伴う投影領域の重複処理や、色補償に対する手法が検討されていないことなどが挙

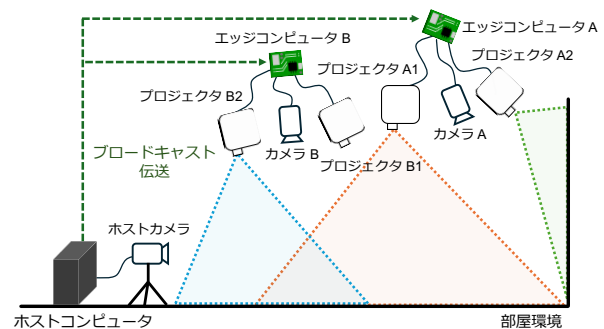


図1 提案する分散協調型マルチプロジェクタ・カメラシステムの構成概要

Fig. 1 System architecture of distributed cooperative multi-projector-camera system

げられる。また、Takeuchiらの手法では、単一カメラの視点に基づいた最適化を行うため、カメラから観測できない遮蔽領域やカメラ範囲外の領域では補償効果を得られないということが挙げられる。この場合、カメラを移動して補償範囲を拡張したり、別の視点からの補償画像を取得することも可能だが、再実行に伴う処理の複雑化や時間的コストの増大が実用上の制約となる問題が存在する。

筆者らの先行研究においても、Takeuchiらの手法と同様に単一のカメラに依存していたため、複数視点での見えの補償という課題は未解決のままであった。また、プロジェクタの重複領域処理として適用していた輝度減衰マスクでは、プロジェクタの投影範囲の端で意図しない輝度誤差が生じ、照明再現の忠実度に限

^{*1}大阪大学

^{*1}The University of Osaka

界があった。より高精度な再現には、重複領域処理手法 [6] の導入が求められるが、それをプロジェクタの増減に柔軟に対応できるスケーラブルなシステム上で効率的に計算することが新たな課題となる。

そこで本研究では、これらの課題を解決するため、広範囲な投影面に対して、複数台のカメラを用いた照明再現手法について検討する。特に、複数台のプロジェクタを利用する際に必要となる重複領域処理と色補償についての手法を提案する。本手法は、単一のコンピュータでシステム全体を管理する従来の中央制御型とは異なり、分散型のマルチプロジェクション環境を利用することで、拡張性の向上を図る。プロジェクタ・カメラをエッジコンピュータに接続し、エッジコンピュータ間の協調により重複領域処理と色補償を分散的に実現する。これにより、新規プロジェクタ追加時には影響を受ける範囲のみを局所的に再校正するだけで済み、システム全体の再キャリブレーションを回避可能とする。本手法により、任意配置されたプロジェクタ・カメラによる高いスケーラビリティを備えたマルチプロジェクションシステムを実現する。

2 提案手法

2.1 分散協調マルチプロジェクタ・カメラシステム

本研究では、小型のシングルボードコンピュータ (Single Board Computer: SBC) をそれぞれ独立したエッジコンピュータとして用いる分散協調型マルチプロジェクタ・カメラシステムを利用する。システム全体の構成を図 1 に示す。本システムは、同期制御を担うホストコンピュータと、投影処理を担当する複数台のエッジコンピュータによって構成される分散協調型のシステムである。さらに、本研究のシステム構成では、エッジコンピュータにカメラを接続することで、ホストコンピュータに依存せず各エッジコンピュータが独自に環境情報を取得可能である。カメラの設置位置には特定のプロジェクタに紐づくといった厳密な制約はなく、システムを構成する全てのカメラを合わせた観測範囲が、全プロジェクタの投影領域を網羅していることのみが設置条件となる。

2.2 重複領域処理

複数台のプロジェクタを用いて広範囲に投影する際、各プロジェクタの投影に重複領域が生じる。この重複投影された部分は、複数の光源からの光が加算されるため輝度値が増加し、つなぎ目が不自然に明るく見えしてしまうという問題が発生する。

これを解決し、プロジェクタ間の重なりをシームレスに見せるためにブレンディングと呼ばれる輝度補正技術 [6] を利用する。重複領域処理用のマスク M_j^i は、各プロジェクタ i の投影領域内における投影面上のピ

表 1 記号一覧
Table 1 List of notations

i	プロジェクタのインデックス
j	ピクセルのインデックス
$c \in \{1, 2, \dots, C\}$	カメラのインデックス
t	イテレーションのインデックス
$r_j^c \in \mathbb{R}_+$	目標画像
$Y_j^c \in \mathbb{R}_+$	カメラ撮影画像
$E_j^i \in \mathbb{R}$	誤差
$x_j^i \in \mathbb{R}_+$	プロジェクタ入力画像強度
$p_j^i \in \mathbb{R}_+$	フォームファクタ
$M_j^i \in \mathbb{R}_+$	重複領域処理用マスク
$w_j^c \in \mathbb{R}_+$	誤差画像の重みマスク
$d_j^c \in \mathbb{R}_+$	環境光
$\lambda \in \mathbb{R}_+$	学習率

クセル j の輝度への寄与度を制御するための重みであり、重複領域において、あるピクセルを照らす全てのプロジェクタの重みの合計が 1 になるように設計する。また、重みはプロジェクタの非線形性を補正するため、計測したガンマ値を用いて投影時の線形性を実現している。

具体的な計算手順としては、初めに各プロジェクタ i について、個別の重みマップ α_j^i を作成する。 α_j^i の値は、ピクセル j の、プロジェクタ投影領域の境界からの距離に基づいて決定される。本稿では滑らかで自然な輝度補正を行うため、プロジェクタ画像の各辺までの距離の積を用いる手法 [7] を利用し、以下のように α_j^i を計算する。ここで、 $D_{j,m}$ はピクセル j から m 番目の画像の端までの距離を示す。

$$\alpha_j^i = \prod_{m=1}^4 D_{j,m} \quad (1)$$

最終的なマスク M_j^i は、あるピクセル j を照らす全てのプロジェクタの重み α_j^l の合計値で正規化することで得られる。

$$M_j^i = \frac{\alpha_j^i}{\sum_l \alpha_j^l} \quad (2)$$

得られたマスク画像を投影画像に乗算することで、重複領域における輝度の過剰な加算が補正され、シームレスな投影が可能となる。

2.3 複数カメラを用いた分散最適化色補償

プロジェクタから投影する画像は、シーンの幾何形状や反射特性、および他のプロジェクタからの光の影響を考慮して決定される必要がある。本研究では、シーンを撮影する複数台のカメラを用いて、全てのカメラ視点において色再現を目標に近づける最適化問題を解くことで、各プロジェクタの投影画像を生成する。

その際、本研究ではシーン内の環境面がランバート面であると仮定する。これは、投影面のある点における輝度が観測するカメラの位置に依存せず一定であることを意味する。この仮定により、複数カメラからの

情報を統合する際のモデルを定義できる。

ここで、各記号と添え字の説明を表1に示す。カメラ c の撮影画像 Y^c は、シーンの点 j における反射光がカメラに入射したものであり、これは2.2節で述べた重複領域処理用マスク M_j^i 、プロジェクタの入力画像強度 x_j^i 、プロジェクタ i から投影された光が投影面上の点 j で反射する割合をモデル化したフォームファクタ p_j^i 、および環境光 d_j^c を用いて、以下のように表現できる。

$$Y_j^c = \sum_i M_j^i p_j^i x_j^i + d_j^c \quad (3)$$

前述のとおりカメラが複数台あるシステム環境において、全てのカメラ視点において色再現を目標に近づける最適化問題を設計する。これは、各カメラにおいて再現対象となる目標画像 r^c と、各カメラでの撮影画像 Y^c の誤差を最小化する問題として定式化する。ここで、本システムのようにカメラとプロジェクタの視点異なる場合、カメラ視点で計算された目標画像と撮影画像の差である誤差画像をプロジェクタ視点に座標変換して利用する必要がある。この座標変換を行うと、あるカメラの画角の端が、プロジェクタの投影領域の内部に現れる場合がある。本稿では、この特定のカメラ由来の誤差情報が存在する領域を有効領域と呼ぶ。有効領域の境界は誤差画像上で不自然なエッジとなり、最適化後もアーティファクトとして投影画像に残存する問題が生じる。

この問題に対処し、より滑らかな補正を実現するため、ピクセルの位置に応じた重み付けを導入する。具体的には各カメラの有効領域に対して、有効領域の境界からの距離に基づいた重みマスク w_j^c を事前に計算する。これにより、画角の端に由来する急激な誤差の変化を緩和し、最適化計算への影響を低減させる。この重み w_j^c を導入し、最適化問題を以下のように定式化する。

$$(x_j^i)^* = \arg \min_{x_j^i} \sum_{c=1}^C \sum_j w_j^c (r_j^c - Y_j^c)^2 \quad (4)$$

s.t. $x_j^i \in [0, 1]$

この最適化問題を解くために、勾配降下法を用いる。プロジェクタ i の入力画像強度 x_j^i の更新のために用いられる誤差 E_j^i は、最適化問題から以下のように表される。

$$E_j^i = \sum_{c=1}^C w_j^c (r_j^c - Y_j^c) \quad (5)$$

この E_j^i と学習率 λ を用いて、プロジェクタ i の入力

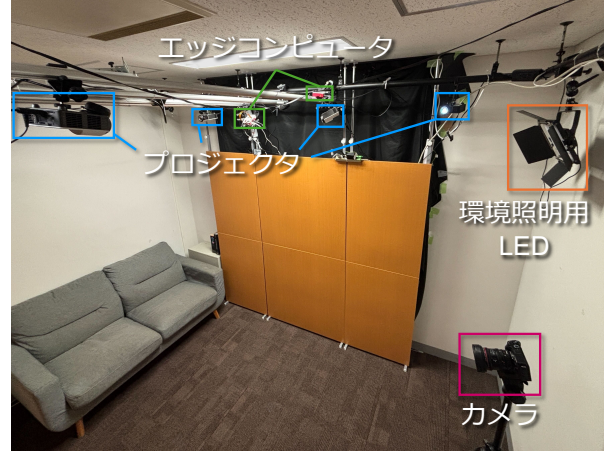


図2 実験に用いたプロジェクタ 11 台とカメラ 3 台からなる実装システム

Fig. 2 The implemented system consisting of 11 projectors and 3 cameras for the experiment

画像強度 x の更新式は、以下のように表される。

$$x_j^i[t+1] = x_j^i[t] - \lambda \cdot E_j^i \cdot p_j^i \quad (6)$$

この更新式 (6) に従って、全ての変数 x_j^i を繰り返し更新し、最終的な投影画像 $(x_j^i)^*$ を得る。また、更新毎に x_j^i の値が制約条件である $[0, 1]$ の範囲に収まるようにクリッピング処理を行う。

2.4 プロジェクタ追加時のアルゴリズム

本稿で提案する分散協調型システムの利点の1つは、新たなプロジェクタを容易にシステムに追加可能であるというスケーラビリティである。ここでスケーラビリティとは、システムにリソースの追加、あるいはリソースの削除に応じて、増減する作業量を効率的に処理できるシステムの特長として定義される [8]。本節では既存のプロジェクタから投影する画像は色補償により最適化されている状態で、新たにプロジェクタを追加するという状況を考える。

プロジェクタが追加された際の重複領域処理は以下のアルゴリズムにより再計算される。

1. 初期化フェーズ (initiation phase): 新規プロジェクタがエッジコンピュータに接続され、システムに登録されると、ホストコンピュータがその参加を全エッジコンピュータに通知し、通知を受けた各エッジコンピュータは、検出フェーズに移行する。
2. 検出フェーズ (detection phase): 追加されたプロジェクタのみが、グレイコードパターンを投影する。このパターンを、システム内に存在する全てのカメラが撮影する。この処理により、

表2 プロジェクタ 11 台構成時における計算コストの比較

評価項目	従来手法（全体再計算）	提案手法（局所的再計算）	削減率
計算時間 [ms]	41,003	8,339	79.7%
計算対象カメラ画素数 [ピクセル]	6,000,000	1,281,294	78.6%
総計算プロジェクタ画素数 [ピクセル]	15,074,382	3,077,512	79.6%

どの既存プロジェクタの投影領域と新たに追加されたプロジェクタの投影領域が重複するかを検出する。また、グレイコードパターン投影時に一様な白色画像の投影と撮影を行っているため、色補償計算で必要となるフォームファクタ p_j^i も同時に取得可能である。

3. 計算フェーズ (calculation phase): 検出フェーズで得られた重複領域の情報を基に、新たに検出された重複領域についてのみ、2.2 節で述べた手法で重複領域処理のための重みマスク (α_j^i および M_j^i) を再計算する。計算された新しいマスクは、関連するエッジコンピュータへ送信される。

以上の処理から、プロジェクタ追加に必要な計算コストと時間を最小限に抑える。

重複領域処理の完了後は、色補償を再開する。2.3 節で述べた分散最適化は、各カメラが誤差情報をブロードキャストすることで計算可能である。そのため、新規プロジェクタが追加されたことに対して、プロジェクタごとの特別な処理を行う必要はなく、継続して色補償が可能である。このように、新しいプロジェクタをシステムに登録する操作を行うことで、追加するプロジェクタによるグレイコードパターンの投影・撮影と局所的なマスクの再計算が自動的に行われ、既存の最適化計算の続きから処理を再開可能である。

3 実装システム

3.1 実装システム

実験環境と実装したシステムを図2に示す。実験環境は、2800 × 2200 mm のリビングを模した室内である。本研究では11台のプロジェクタ (Optoma ML1050ST+S1J: 6台, RICOH PJ WXC1110: 4台, Acer H6517ST: 1台) と5台のエッジコンピュータ (Raspberry Pi 5) を用いてシステムの構築を行った。ホストコンピュータには魚眼レンズ (Canon EF8-15 mm f/4L Fisheye USM) を取り付けカメラ (Canon EOS M6 Mark II) を接続した。また、エッジコンピュータ2台に対してカメラ (Raspberry Pi Camera Module 3 wide) を接続した。また、部屋全体を照明可能な位置に環境照明用 LED (NEEWER NL660 Bi-Color LED

プロジェクタ台数と計算負荷の関係

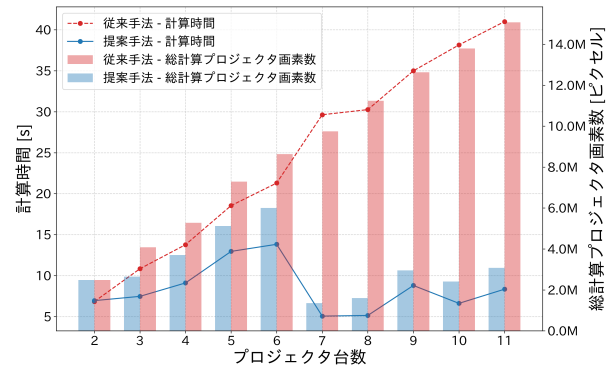


図3 プロジェクタ台数増加に伴うマスク計算コストの比較

Fig. 3 Comparison of computational cost for mask recalculation as the number of projectors increases

Panel Light) を設置した。本 LED 照明を用いて部屋を照らした際の環境を撮影することで色補償最適化の目標画像を取得している。

3.2 プロジェクタ追加時の計算コスト評価

提案手法のスケラビリティを定量的に評価するため、計算効率の観点から、プロジェクタを追加した際のマスク再計算コストを従来手法と比較する実験を行った。比較対象の従来手法は、プロジェクタが1台追加されるたびに、システムに存在する全てのプロジェクタの重複領域を再計算する手法とした。実験では、2台のプロジェクタが稼働している状態から、1台ずつプロジェクタを追加し、最終的に11台構成になるまで各ステップでのマスク計算時間とマスク計算に用いた画素数を計測した。なお、カメラ画像の解像度は3000 × 2000であり、プロジェクタ画像の解像度は1280 × 800である。

従来手法と提案手法の計算時間と計算画素数を比較したグラフを図3に、システムが最大の11台構成になった時点での計算コストを表2に示す。

図3から、従来手法（赤線）の計算時間はプロジェクタ台数の増加にほぼ比例して増大した。これは、システムの規模が拡大するにつれて、1台の追加に伴う再キャリブレーションの負荷が線形的に増大していくことを示している。一方、提案手法（青線）の計算時間は、台数が増加しても一定の水準で推移した。

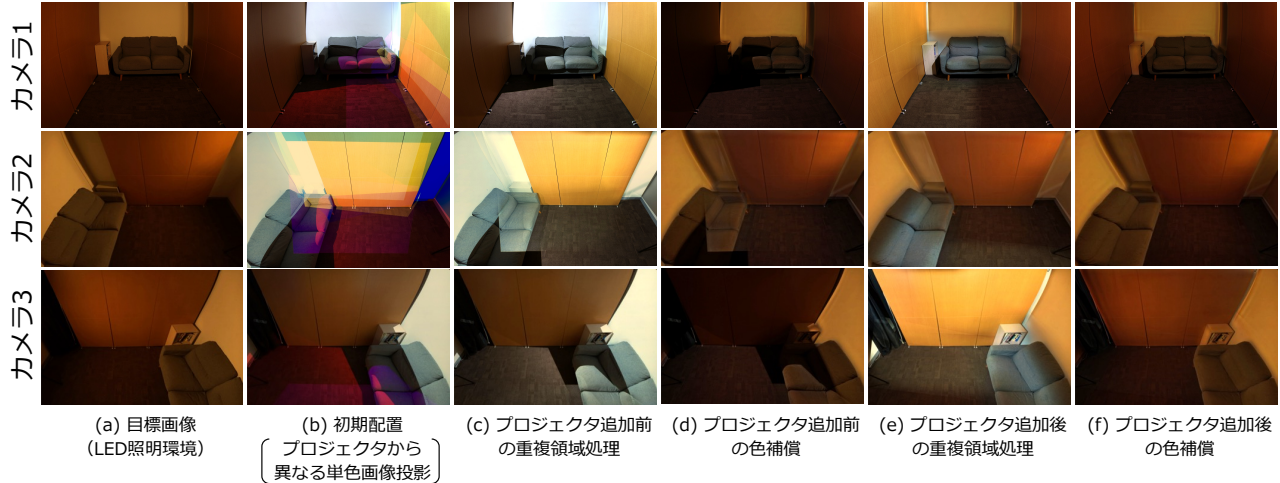


図4 各カメラで撮影された各条件の投影結果

Fig. 4 Projection results at each step of the projector addition experiment from each camera's viewpoint

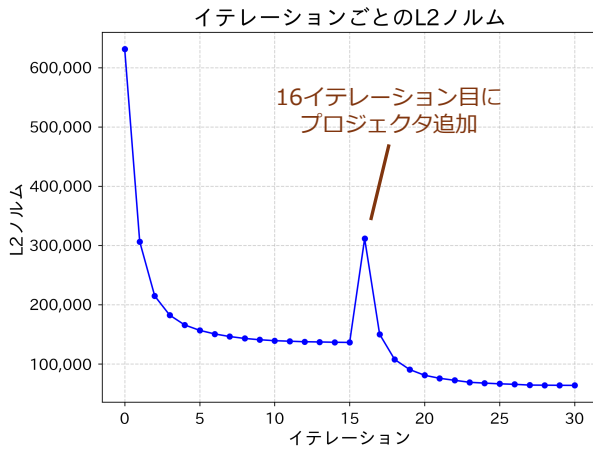


図5 プロジェクタ追加前後における色補償誤差 (L2 ノルム) の収束過程

Fig. 5 Error convergence (L2 norm) during the color compensation process before and after adding a new projector

特に、提案手法の計算時間の変動は、右軸に示す総計算プロジェクタ画素数（棒グラフ）と強い相関があることが分かる。例えば、7台目のプロジェクタ追加時は、既存プロジェクタとの重複領域が少なかったため総計算プロジェクタ画素数は少なく、それに伴い計算時間も大幅に減少した。逆に6台目追加時は、重複領域の面積が広だけでなく、重複に関与するプロジェクタの台数も多かったため、総計算プロジェクタ画素数も多くなり、計算時間もピークに達した。この結果は、提案手法の計算コストが、システム全体の規模ではなく、あくまで局所的な重複領域の広さと何台のプロジェクタが重複しているかという複雑さに起因することを示している。

また、表2に示す通り、11台構成時において、提案手法は従来手法と比較して、計算時間が約79.7%短かった。これらの結果から、提案手法によるマスク計算がプロジェクタ追加時にも計算量が低く抑えられることを確認し、システムの規模拡大に対する高い計算効率とスケーラビリティを持つことを確認した。

3.3 プロジェクタ追加時の色補償性能評価

前節での計算コスト評価に加え、本節では実際に実装システムを用いて色補償を行い、2.4節で提案したアルゴリズムに基づいて、稼働中のシステムに新たなプロジェクタを追加する実験を行った。具体的には、2.4節で述べたアルゴリズムに基づき、システム全体の再キャリブレーションを行うことなく、最適化中に投影範囲を拡張し、色補償を継続できることを示す。

実験手順として、まず実装システムの一部プロジェクタのみを稼働させた初期状態での環境照明再現を行った。図4(b)は実験の初期状態で使用するプロジェクタから異なる単色を投影したものである。この限定された領域に対し、重複領域処理を適用した結果が図4(c)である。次に、目標画像(図4(a))を再現するように色補償を行い、誤差が収束した状態を図4(d)に示す。次に、この稼働中のシステムに対し、左側を照らすプロジェクタを新たに追加した。提案手法に基づき、追加されたプロジェクタから順にグレイコードパターンを投影し、重複領域を検出後、影響を受ける領域のマスク画像のみが局所的に再計算される。図4(e)は、この局所的なマスク更新を終えた直後の様子である。既存領域の色補償は維持されたまま、新たに追加された領域には重複領域処理のみが適用された白色光が投影されている。

マスク更新後、色補償最適化をシステム全体で再開した。図 4 (f) に最終的な照明再現結果を示す。結果として、プロジェクタが追加された境界領域も既存投影領域とシームレスに繋がり、部屋全体に渡って均一な照明が再現できていることが確認できる。

図 5 に、プロジェクタ追加前後における最適化の収束過程を示す。グラフの縦軸は目標画像と撮影画像の誤差の指標である L2 ノルムであり、横軸は最適化のイテレーション数を表す。初期状態のプロジェクタ群で誤差が一度収束した後、プロジェクタを追加すると、色補償されていない領域が発生するため一時的に誤差が増大する。しかし、提案手法により最適化が継続されると、誤差が再び減少し、収束することがわかる。以上より、提案手法がプロジェクタ追加に対して、継続的に色補償を実行可能であり、高いスケーラビリティを持つことを確認した。

また、図 5 の最終的な収束値に注目すると、プロジェクタ追加後の誤差が、追加前の収束値を下回っていることがわかる。これは、プロジェクタの追加によって投影範囲が拡大し、システム全体で再現できる目標画像の領域が増加したためと考えられる。初期状態では再現できていなかった領域もプロジェクタ追加により投影が可能になったことで、目標画像との全体的な差分が減少し、結果として L2 ノルムで評価される誤差が改善した。このことは、本手法が単にシステムの拡張を可能にするだけでなく、プロジェクタ台数の増加に応じて照明再現の質の向上にも寄与することを示唆している。

4 おわりに

本研究では、複数台のプロジェクタと複数台のカメラを用いた環境において、重複領域処理と最適化色補償を分散的に実現する手法を提案した。また、カメラから得られる局所的な情報を基に、各エッジコンピュータが独立して最適化を行う枠組みにより、システム全体のスケーラビリティを確保しつつ、効率的に計算可能なアルゴリズムの提案も行った。実験では、プロジェクタ台数が増加しても処理が破綻せず、提案手法が拡張性に優れていることを示した。今後は、相互反射のようなより複雑な光学的現象を考慮した補償モデルの導入や、さらに大規模な空間での実用性の検証を目指す。

参考文献

- [1] Oliver Sand, Sebastian Büttner, Volker Paelke and Carsten Röcker. smart. assembly-projection-based augmented reality for supporting assembly workers. International conference on virtual, augmented and mixed reality. Cham: Springer Inter-

- national Publishing, 2016.
- [2] Hiroto Nishino, Etsuro Hatano, Satoru Seo, Takashi Nitta, Tomoyuki Saito, Masaaki Nakamura, Kayo Hattori, Muneo Takatani, Hiroaki Fuji, Kojiro Taura, Shinji Uemoto. Real-time navigation for liver surgery using projection mapping with indocyanine green fluorescence: Development of the novel medical imaging projection system. *Annals of surgery*, 2018, 267(6): 1134-1140.
- [3] Masahiko Yasui, Ryota Iwataki, Masatoshi Ishikawa and Yoshihiro Watanabe. Projection mapping with a brightly lit surrounding using a mixed light field approach. *IEEE Transactions on Visualization and Computer Graphics*, 2024, 30(5): 2217-2227.
- [4] Masaki Takeuchi, Hiroki Kusuyama, Daisuke Iwai and Kosuke Sato. Projection mapping under environmental lighting by replacing room lights with heterogeneous projectors. *IEEE Transactions on Visualization and Computer Graphics*, 2024, 30(5): 2151-2161.
- [5] 三浦和之, 岩井大輔, 佐藤宏介. 分散協調マルチプロジェクションシステムによる実環境照明再現. 第 29 回日本バーチャルリアリティ学会大会論文集, 2024, 3E2-09:1-3E2-09:3.
- [6] Bimber Oliver, and Ramesh Raskar. Spatial augmented reality: merging real and virtual worlds. CRC press, 2005.
- [7] Harville Michael, Culbertson Bruce, Sobel Irwin, Gelb Dan, Fitzhugh Andrew, Tanguay Donald. Practical methods for geometric and photometric correction of tiled projector. *Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, 2006.
- [8] Richer Gaëlle, Pister Alexis, Abdelaal Moataz, Fekete Jean-Daniel, Sedlmair Michael, Weiskopf Daniel. Scalability in visualization. *IEEE Transactions on Visualization and Computer Graphics*, 2022, 30(7): 3314-3330.

© 2025 by the Virtual Reality Society of Japan (VRSJ)