

# 自由視点依存の軽量化による多視点 RGB-D データの高速ネットワーク通信技術の開発

代表研究者 藤本 忠博 岩手大学 理工学部 教授

## 1 序論

自由視点映像とは、3次元空間上の自由な視点位置から対象とする物体やシーンを見ることができる映像である。最近では、カラー（RGB）データに加え、カメラから物体の表面までの距離であるデプス（Depth）データをリアルタイムで取得できる RGB-D カメラを用いることで、従来の RGB カメラを用いた場合に必要であった物体の3次元形状の推定を行わず、直接、RGB 値を持つ3次元点群を取得して高速に自由視点映像を生成することが可能となった。しかし、RGB-D データの容量は RGB データに比べて大きい。さらに、複数の RGB-D カメラで取得した多視点 RGB-D データを用いて、遠隔地で要求される自由視点に対する自由視点映像をリアルタイムで生成するためには、高速なネットワーク転送のためのデータの軽量化が必須である。そこで、本研究では、自由視点に依存した効果的な RGB-D データの軽量化の手法を提案する。この軽量化手法については、代表研究者の研究グループで先行研究を実施している[1-3]。本研究では、その問題点を改善し、より効果的な軽量化の手法を開発する。

また、上記に関連して、RGB-D カメラによりリアルタイムで取得される RGB-D データから自由視点映像を生成する際、RGB-D データから得られる3次元点群が振動し、高品質な自由視点映像が得られない場合が多い。本研究では、その振動を軽減する方法を提案する。さらに、複数のカメラを用いて自由視点映像を生成する手法として、代表研究者の研究グループでは、複数の RGB カメラの映像から生成した自由視点映像上の任意の注目物体をリアルタイムでエディットする実時間ビデオエディット手法を開発している[4-10]。将来的に、この先行研究の手法に RGB-D カメラを利用し、遠隔地での利用も可能とすることを計画している。本研究では、その手法の拡張に向けた内容も実施する。

## 2 研究内容

### 2-1 多視点 RGB-D データの高速転送のための自由視点依存の軽量化

#### (1) はじめに

自由視点映像とは、対象物体を自由な視点位置から見ることを可能とする映像である。この映像により、ユーザは実際のカメラが存在しない位置からでも対象物体を見ることができる。近年、通信技術の向上が著しく、特に、無線通信技術の発達には目覚ましいものがある。高速にデータを転送する通信技術を利用して自由視点映像をリアルタイムで生成・配信する技術は、国内外を問わず、多くの研究がなされている。自由視点映像生成の手法として、これまで、複数のカラー（RGB）カメラを用いる手法がよく利用されてきたが、カメラ映像の各画素が持つ RGB 値による2次元情報から物体の3次元形状の推定を行なう必要があり、その推定の計算量や精度の問題がある。一方、最近では、画素ごとの RGB 値に加えて、デプス（Depth）値を取得できる RGB-D カメラの開発が行われ、利用できるようになってきている。現在では、RGB-D カメラを利用した自由視点映像の生成に関する研究も行われている。RGB-D データによって、直接、物体の3次元形状を得ることができることから、その推定のための計算量を削減することができ、リアルタイムでの自由視点映像生成が容易となる。しかし、RGB データに比べて RGB-D データは容量が大きく、データの転送の際の欠点となる。本研究では、複数の RGB-D カメラで取得された多視点 RGB-D データから遠隔地でリアルタイムの自由視点映像生成を可能とすることを目指し、大量の RGB-D データを高速に転送するための自由視点に依存した軽量化手法を提案する。

#### (2) 関連研究

##### (2-1) RGB カメラを用いた方法

コンピュータグラフィックスやコンピュータビジョンの分野では、複数の RGB カメラによって取得した多視点 RGB 映像から自由視点映像を生成する手法が活発に研究されてきた。代表的なものとして、大量の参照画像を膨大なレイの集合として扱うライトフィールドレンダリング手法[11]、参照画像上の物体の輪郭を利

用する手法[12]などが挙げられる。後者でよく用いられる3次元形状の表現方法としてビジュアルハルがあり、画像ベースで自由視点映像の画素色を高速に計算する方法[13]、ポリゴンモデルにより形状を復元する方法[14]、ボクセルモデルにより形状を復元する方法などが提案されている。ボクセルモデルを利用する代表的な手法として視体積交差法[15]があり、さらに、参照画像の色情報を利用してビジュアルハルより精度の高いフォトハルを生成するボクセルカラーリング法[16]などもある。

### (2-2) RGB-D カメラを用いた方法

RGB-D カメラは、物体表面までのデプスを直接的に取得できるため、その3次元形状の推定が不要であるという利点を利用した多くの研究が行われている。複数のRGB-D カメラで取得された多視点RGB-D 映像を利用して自由視点映像を生成する手法[17, 18]、距離画像センサを用いた空間再現手法[19]などが挙げられる。また、複数のRGB-D カメラを用いて、中央に設置したカメラの映像中の遮蔽領域を半透明にすることで、遮蔽されている背景が透視可能な半隠消映像を生成する手法[20]が提案されている。また、動いている観測対象を高精度にレンダリングする手法[21]も提案されている。

### (2-3) センサネットワーク

複数の様々なセンサを広範囲に設置して、無線によるネットワークを構築するマルチメディアセンサによるネットワーク (Wireless Multimedia Sensor Network: WSN) などの研究が行われている[22]。多視点カメラ映像による自由視点映像の生成では、複数のカメラを制御するためのネットワークの構築が必要となる。近年では、無線ネットワーク技術の飛躍的な進歩により、携帯端末を用いた自由視点映像の受信に関する技術なども研究されている[23]。

### (2-4) 先行研究

本研究の先行研究では、多視点RGB-D 映像を利用した自由視点映像のリアルタイム生成を目指し、RGB-D データを自由視点依存で軽量化するための基礎技術の開発が行われていた[1-3]。その基礎技術においては、多視点RGB-D 映像を取得するローカル環境での動作を前提としたシステムの構築が行われており、遠隔地へのRGB-D データの転送は考慮されていなかった。そこで、本研究では、遠隔地へのデータ転送を考慮した技術の開発を目的とする。

## (3) 提案手法

### (3-1) 概要

本研究では、物体を囲む複数のRGB-D カメラで取得した多視点RGB-D 映像から、物体の異なる表面部分のRGB-D データを統合し、自由視点に依存した効果的な軽量化を行うことで、遠隔地に高速に送信する手法を提案する。本研究では、各RGB-D カメラを個別のカメラ用PCに接続する。遠隔地のユーザ用PCから要求された自由視点を拡張した複数の拡張自由視点に対して、各RGB-D カメラで取得したRGB-D データをカメラ用PC上でプリレンダリングすることで、複数の拡張自由視点RGB-D データを生成する。それらを統合用PCへ送信して統合した後、統合用PCからユーザ用PCへ送信する。

### (3-2) システム構成

ローカル環境において、複数のRGB-D カメラとカメラ用PCを用意し、各RGB-D カメラを個別のカメラ用PCに1対1で接続する。また、各カメラ用PCから送信されるRGB-D データを受信する1台の統合用PCを用意する。そして、これらのPCをネットワーク接続することでPCクラスタを構築する。また、遠隔地にあることを想定したユーザ用PCを用意する。各RGB-D カメラによって取得したRGB-D データは、ローカル環境内でカメラ用PCと統合用PCによって軽量化され、統合用PCから外部ネットワークを経由して遠隔地のユーザ用PCに送信される。ローカル環境内のデータ転送にはMPI (Message Passing Interface)、ローカル環境と遠隔地間のデータ転送にはソケット通信を用いる。

### (3-3) カメラ校正

複数のRGB-D カメラを利用するためには、それらのカメラ校正を行う必要がある。本研究では、RGB 画像のみを用いるZhangの校正法により、チェッカーボードを用いて各カメラの内部パラメータと外部パラメータを推定することとした。

### (3-4) ローカルRGB-D データの取得

各RGB-D カメラがリアルタイムで取得するRGB-D 映像をローカルRGB-D データと呼ぶこととする。RGB-D 映像上の任意の画素Pの画素座標( $X_p, Y_p$ )とRGB-D 値( $R_p, G_p, B_p, D_p$ )から、画素Pのカメラ座標系における3次元座標値( $x, y, z$ )とRGB 値( $r, g, b$ )を求めることができる。そのため、RGB-D 映像中のフレーム時刻ごとのフレーム画像が持つRGB-D データは、カメラから可視である物体の表面部分を構成するRGB 値付き

の3次元点群といえる。

#### (3-5) 自由視点 RGB-D データの生成

各 RGB-D カメラが取得したローカル RGB-D データには、物体の表面のうち、自由視点から不可視である多くの表面部分が含まれている。そこで、それらを除外し、可視である表面部分だけを各カメラ用 PC から統合用 PC に送信する。これを実現するため、カメラごとに、そのカメラのローカル RGB-D データによる3次元点群を自由視点に対してレンダリングし、その結果をフレームバッファとデプスバッファから取り出す。この取り出したデータは、自由視点から可視である表面部分のみから構成される。これを自由視点 RGB-D データと呼び、このレンダリングをプリレンダリングと呼ぶこととする。プリレンダリングの際、3次元点群から生成したメッシュに対してレンダリングを行うこととし、さらに、自由視点から見て裏面となる部分を除去するバックフェースカリングを行うことにより、自由視点 RGB-D データの品質向上と冗長削減がなされる。また、ローカル RGB-D データ中で対象物体の背後の背景の部分は、デプス値の閾値を与えることでプリレンダリングの対象から除外する。プリレンダリングは各カメラ用 PC で行われるため、全体の処理を分散して効率的なデータの生成を行うことができる。

#### (3-6) 拡張自由視点 RGB-D データの生成

自由視点に依存した自由視点 RGB-D データをローカル環境の統合用 PC から遠隔地のユーザ用 PC に送信する全体の流れを考えた場合、ユーザ用 PC から統合用 PC への自由視点の要求、ローカル環境内での自由視点 RGB-D データの生成、そして、統合用 PC からユーザ用 PC への自由視点 RGB-D データの送信に要する時間により、大きな遅延が発生する可能性がある。すなわち、ユーザ側では、要求した自由視点に対する自由視点 RGB-D データが遅れて到着し、その到着の時点では要求したい自由視点が移動している可能性がある。そこで、本研究では、ユーザから要求された自由視点の位置を近傍に拡張した複数の拡張自由視点に対する自由視点 RGB-D データを生成して転送することにより、ユーザ側で自由視点を移動させても遅延の少ない自由視点映像を生成できる手法を提案する。

各カメラ用 PC では、それぞれの拡張自由視点に対してローカル RGB-D データのプリレンダリングを行い、それぞれの結果として得られる自由視点 RGB-D データを統合用 PC へ転送する。これらを拡張自由視点 RGB-D データと呼ぶこととする。拡張自由視点の個数を  $n$ 、RGB-D カメラの個数（カメラ用 PC の個数）を  $m$  とした場合、合計で  $n \times m$  個の拡張自由視点 RGB-D データが統合用 PC に転送されることになる。このうち、共通の拡張自由視点に対する  $m$  個のデータはデプスバッファリングで統合できるため、結果として  $n$  個の拡張自由視点 RGB-D データとすることができる。そして、その  $n$  個のデータを統合用 PC からユーザ用 PC へ送信する。ユーザ用 PC は、受信した  $n$  個のデータを利用して、その時点でユーザが要求したい自由視点に対する自由視点画像を遅延なく生成する。

本研究の実験では、拡張自由視点の個数を  $n = 4$  とした。自由視点を含み、その視線方向を法線とする平面上で、自由視点を中心とする正方形を与え、その正方形の4頂点に拡張自由視点を置く。各拡張自由視点の視線方向は自由視点と同じものとする。

#### (3-7) 拡張自由視点 RGB-D データの圧縮による軽量化

高速な転送のため、拡張自由視点 RGB-D データに対して圧縮処理によるさらなる軽量化を行う。様々な圧縮法が適用可能であるが、本研究ではシンプルな RLE (Run Length Encoding) 法を用いることとした。拡張自由視点 RGB-D データ上の画素を走査線順に調べ、物体表面の画素値が格納されている場合には RGB-D 値を圧縮データに格納する。格納されていない場合には、それ以降に物体表面の画素値が格納されていない画素が連続する個数を圧縮データに格納する。この圧縮法は、物体表面の画素値が格納されている画素が減るほど有効である。

#### (4) 実験

4台の RGB-D カメラ (Microsoft Xbox One Kinect センサ) を利用した実験を行った。各カメラを個別に4台のカメラ用 PC に接続し、別の1台の統合用 PC を含む合計5台の PC で PC クラスタを構築し、それらの PC 間で MPI によりデータの送受信を行う。また、遠隔地にあることを想定した1台のユーザ用 PC に対して、統合用 PC との間でソケット通信を用いてデータの送受信を行う。

図 1-1 は、4個の拡張自由視点のうちの一つに対して各カメラ用 PC でプリレンダリングを行って得られた拡張自由視点 RGB-D データの画像である。図 1-2 は、そのデータを統合用 PC で受信し、デプスバッファリングにより統合したデータの画像である。各 RGB-D カメラによる可視部分が統合され、拡張自由視点から可視となる適切な RGB-D データが得られている。しかし、カメラのキャリブレーションの精度に依存した若干の

ズレが見られ、今後、改善が必要である。統合用 PC とユーザ用 PC の間のソケット通信によるデータの送受信については、現在、拡張自由視点ごとに独立した拡張自由視点 RGB-D データから冗長を軽減して効率的に送受信する方法を検討し、実装を行っている段階である。

#### (5) おわりに

本研究では、複数の RGB-D カメラによって取得した多視点 RGB-D データに対して、自由視点に依存した軽量化を行い、遠隔地へ効率的に送信する手法を提案した。本手法では、各カメラ用 PC で複数の拡張自由視点 RGB-D データを生成し、統合用 PC に送信する。統合用 PC では、それらを統合し、遠隔地のユーザ用 PC へ送信する。今後の課題として、統合用 PC からユーザ用 PC に送信する拡張自由視点ごとの拡張自由視点 RGB-D データを統合して軽量化する手法の開発が挙げられる。また、送信されたデータを利用してユーザ用 PC で効率的に自由視点映像を生成する方法の検討も必要である。

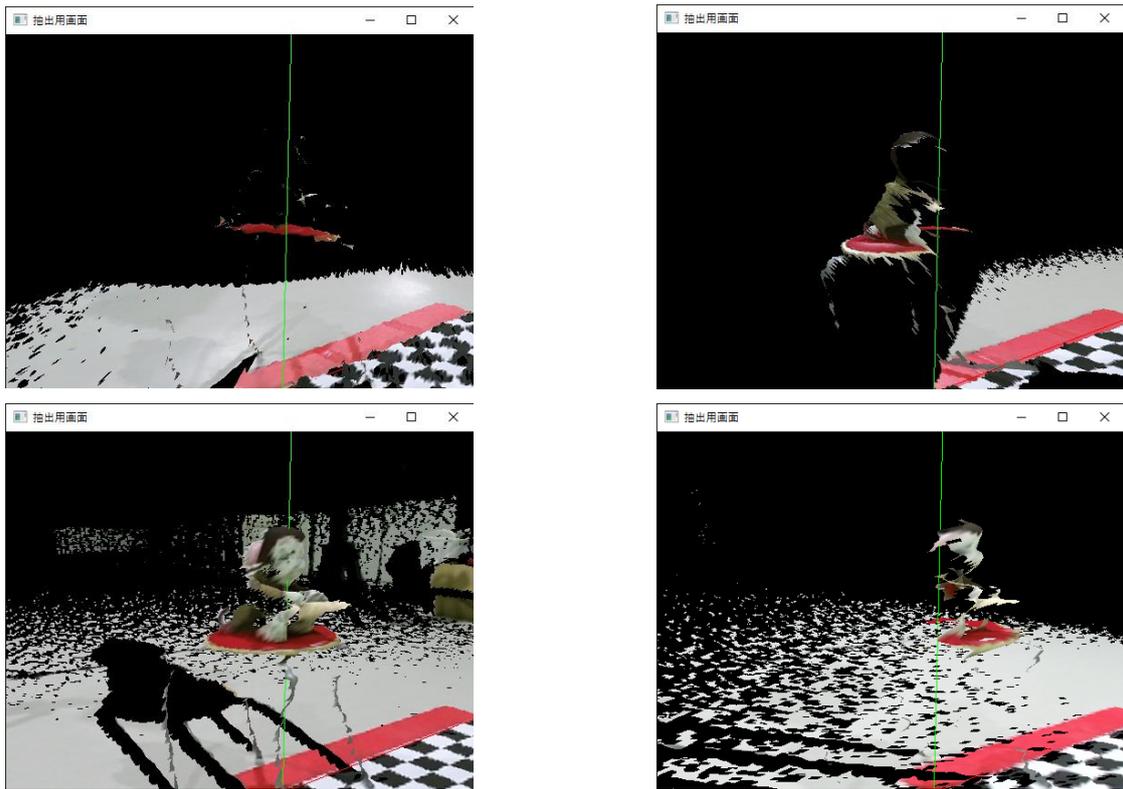


図 1-1 各カメラ用 PC による拡張自由視点 RGB-D データの画像

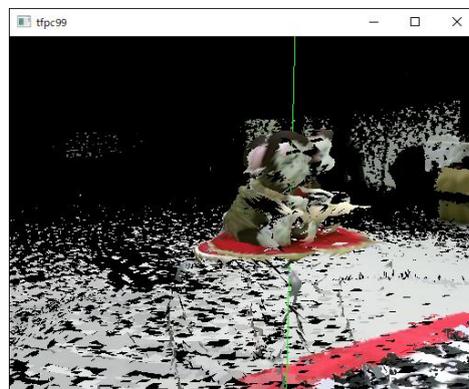


図 1-2 統合用 PC により統合した拡張自由視点 RGB-D データの画像

## 2-2 実時間デプスデータの振動の軽減化

### (1) はじめに

自由視点映像は、対象とする物体やシーンを任意の自由な視点から見る事ができる映像である。近年では、カラー (RGB) 映像に加えてカメラから物体表面までのデプス (Depth) をピクセルごとに持つ RGB-D 映像を撮影できる RGB-D カメラが利用できるようになり、RGB 映像のみの場合に必要となる物体表面の 3 次元形状の推定が不要となり、リアルタイムで自由視点映像を生成することが可能となっている。複数の RGB-D カメラで撮影された多視点 RGB-D 映像を用いることで、それぞれのカメラが撮影できない部分を互いに補いながら物体の全体形状を見ることが出来る自由視点映像が生成できる。しかし、取得した多視点 RGB-D 映像をそのまま用いた場合には、高品質な自由視点映像を得ることができないことが多い。その原因の一つとして、RGB-D 映像を 3 次元点群データにして表示する際、点群が振動してしまうことが挙げられる。これは、RGB-D カメラで取得したデプス値の振動が原因として考えられる。本研究では、その振動を軽減する方法を開発する。

### (2) 関連研究と関連技術

RGB-D 映像から高品質な自由視点映像を生成する方法[24, 25]が幾つか提案されているが、いずれも複雑な処理を要する。本研究では、容易に適用が可能な基礎技術として、RGB-D 映像が持つデプス値の振動を軽減する方法を検討する。

本研究では RGB-D カメラとして Kinect V2 を使用する。Kinect V2 は、Time of Flight (ToF) と呼ばれる方式により、カメラから投光した赤外線が物体表面で反射して戻る (受光される) までの時間からデプス値を得る。

### (3) 提案手法

本研究では、取得されるデプス値にはデプス方向の誤差とピクセル平面方向の誤差が含まれるものと考えられる。前者は赤外線の投光から受光までの時間から計算されるデプス値自体が持つ誤差、後者は投光と受光の位置がピクセル位置からずれることで生じる誤差である。そして、全てのフレーム時刻と全てのピクセル位置において、それらの誤差は互いに独立しているものとする。よって、次の 2 つの方法でデプス値を平均化するフィルタを適用することで、それらの誤差を相殺して軽減化することを目指す。

- フレーム時刻間で平均化するフィルタ
- 隣接ピクセル間で平均化するフィルタ

#### (3-1) フレーム時刻間で平均化するフィルタ

フレーム時刻ごとに取得したデプス画像を現時刻までの  $N_f$  フレーム分だけ保存し、ピクセルごとに  $N_f$  個のデプス値を平均化した値を現時刻のデプス値として扱う。フレーム時刻間の誤差は互いに独立しており、それらを平均化により相殺するという考えから、フレーム時刻間で異なる重みを与える加重平均化はせず、全てのフレーム時刻のデプス値を同等に扱う単純な平均化を行う。

物体の輪郭部分が映るピクセルにおいては、ピクセル平面方向の誤差の影響から、フレーム時刻によって赤外線が物体に当たる場合とその背後の背景に当たる場合があり、結果として、平均化する  $N_f$  個のデプス値に物体と背景の両方のデプス値が混在する可能性がある。そこで、 $N_f$  個のデプス値のうちで値が大きく異なるものがある場合、物体と背景のデプス値が混在するものとみなし、全てのデプス値の平均を閾値としてデプス値の大小から物体か背景かを区別し、個数が多いほうのデプス値だけの平均を求める。

#### (3-2) 隣接ピクセル間で平均化するフィルタ

全てのピクセルについて、注目ピクセルを中心とした  $M_p \times M_p$  個のピクセルのデプス値を平均化し、その値を注目ピクセルのデプス値とする。この場合にも、ピクセル間の誤差は互いに独立しているという考えから、加重平均化はせず、単純な平均化を行う。

### (4) 実験

実験環境として、1 台の Kinect V2 を PC に接続し、屋内のシーンを撮影して RGB-D 映像を取得し、2 種類のフィルタをデプス値に適用する実験を行った。ピクセルごとに取得された RGB 値とフィルタを適用したデプス値から RGB 値を持つ 3 次元点を求め、全てのピクセルによる点群を自由視点から見た映像を表示した。以下、異なる条件で実験を行った結果について述べ、その画像を図 2-1 から図 2-3 に示す。

#### (4-1) フレーム時刻間の平均化

(実験 1-1) フィルタを適用しない場合、点群の振動が見られた。

(実験 1-2) 平均化するフレーム数を  $N_f = 10$  とした場合、全体的に点群の振動は軽減されたが、物体の輪

郭部分で振動によるノイズが発生した。

(実験 1-3) 実験 1-2 に物体か背景かを区別する輪郭部分の処理を加えたところ、輪郭部分のノイズが軽減された。

(実験 1-4) フィルタを適用せず、点群をメッシュ化して表示した。メッシュ化しない場合に見られた点の間の隙間がほとんど無くなったが、メッシュの振動が見られた。

(実験 1-5) フレーム数を  $N_f = 10$  としてデプス値を平均化し、点群をメッシュ化した場合、振動が軽減されることが確認できた。

(実験 1-6) フレーム数を  $N_f = 100$  としてデプス値を平均化し、点群をメッシュ化した場合、実験 1-5 の場合より振動が軽減されることが確認できた。しかし、映像中の物体が動いた場合、平均化するフレーム数が多いほど物体が大きくぼやけてしまうことが確認できた。

#### (4-2) 隣接ピクセル間の平均化

(実験 2-1) 平均化する隣接ピクセル数を  $M_p \times M_p = 3 \times 3$  とした場合、点群の振動が若干は軽減されたが、フレーム時刻間の平均化に比べて効果が低いことが確認できた。

(実験 2-2) 隣接ピクセル数を  $M_p \times M_p = 5 \times 5$  とした場合、実験 2-1 の場合に比べて、わずかではあるが、より振動を軽減する効果が確認できた。

#### (4-3) フレーム時刻間と隣接ピクセル間の平均化

(実験 3-1) フレーム数を  $N_f = 10$  とし、隣接ピクセル数を  $M_p \times M_p = 3 \times 3$  として、フレーム時刻間と隣接ピクセル間のデプス値の平均化を同時に適用した。フレーム時刻間のみの平均化と比較して、わずかではあるが、より振動を軽減する効果が確認できた。

#### (5) おわりに

実験結果から、フレーム時刻間でデプス値を平均化するフィルタは点群の振動を軽減する効果があり、平均化するフレーム数を増やすほど効果が高まるが、動く物体がある場合には大きくぼやけが発生することが確認できた。フレーム時刻間の平均化に比べて、隣接ピクセル間の平均化は効果が低いことが確認でき、両者を合わせることで、より効果が高められることが確認できた。今後の課題として、フレーム時刻間と隣接ピクセル間の平均化を適切に組み合わせることで、動く物体の振動を効果的に軽減する手法を開発することが挙げられる。



(実験 1-1)



(実験 1-2)



(実験 1-3)



(実験 1-4)

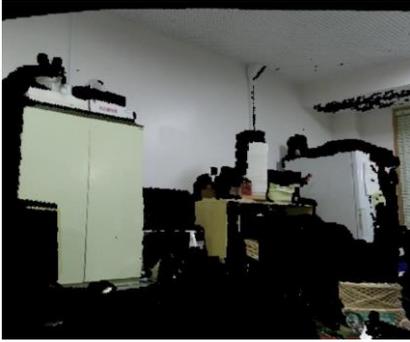


(実験 1-5)



(実験 1-6)

図 2-1 フレーム時刻間のデプス値の平均化



(実験 2-1)



(実験 2-2)



(実験 3-1)

図 2-2 隣接ピクセル間のデプス値の平均化

図 2-3 フレーム時刻間と隣接ピクセル間のデプス値の平均化

## 2-3 実時間ビデオエディットの拡張に向けたカメラアレイ分散管理と仮想空間構築

### (1) はじめに

複数の RGB カメラから構成されるカメラアレイを利用して、各カメラで撮影された多視点 RGB 映像を合成することにより、自由な視点から見た自由視点映像を生成することができる。さらに、合成開口法と呼ばれる技術により、カメラアレイから見て注目したい物体の前方にある遮蔽物体をぼかし、注目物体にフォーカスした自由視点映像を生成することもできる。この技術を利用して、本研究の先行研究において、自由視点映像から遮蔽物体を除去して注目物体のみを抽出し、注目物体に拡大・縮小、平行移動、回転などの 2 次元エディット処理をリアルタイムで行う実時間ビデオエディット手法が提案されている[4-10]。しかし、この手法では、カメラの台数と撮影の範囲に制限があり、エディット処理が自由視点映像面に平行な 2 次元平面上に限定されている。

本研究の最終目標は、カメラの台数を増やして物体を周囲の広範囲から撮影することで、先行研究のエディット処理を 2 次元平面上から 3 次元空間上に拡張することである。このとき、1 台の PC に接続できるカメラの台数には制限があることから、カメラ台数の増加のためには、それらを複数台の PC に分けて接続し、適切に分散管理する必要がある。また、多数のカメラの 3 次元空間上での配置を考慮して正確にキャリブレーションを行い、注目物体と遮蔽物体の位置に応じた試行錯誤的な実験を行う必要がある。そのため、実空間上での実施の前に、実空間を模擬する仮想空間をコンピュータ内に構築してシミュレーションを実施することで、手法の開発を効率的に進めることが可能となる。そこで、本研究では、実時間ビデオエディット手法の拡張に向けて、複数台の PC によるカメラアレイの適切な分散管理、ならびに、実空間を模擬する適切な仮想空間の構築を実現することを目的とする。

### (2) 先行研究

先行研究の手法では、合成開口法の原理を用いて、カメラアレイによって得られた多視点 RGB 映像を利用して前方の遮蔽物体と後方の背景から注目物体のみを分離して抽出する。通常、多視点 RGB 映像から自由視点映像を生成するためには、物体表面の 3 次元形状と色を推定する必要がある。一方、合成開口法では、カメラのフォーカスの原理を用いて、3 次元空間上で物体表面が存在するものと予測される位置に焦点面を設けることで、物体表面の形状を推定することなく、色の推定を行う。具体的には、物体表面が焦点面上にあるものと仮定し、各カメラの映像を焦点面に投影して平均化することで自由視点映像を得る。そのため、焦点面と物体表面が近い場合は、物体表面の一点の色が焦点面上の同一点に投影される（フォーカスする）全てのカメラの画素にほぼ正確に映るため、鮮明な自由視点映像になる。逆に、焦点面と物体表面が離れている場合は、フォーカスする各カメラの画素に物体表面の異なる位置の色が映るため、ぼやけた映像となる。注目物体の表面が焦点面上にある場合でも、注目物体の手前にある遮蔽物体が幾つかのカメラの視野を遮る場合には、焦点面上でフォーカスする各カメラの画素のうち、幾つかの画素には遮蔽物体上の異なる点、残りの画素には注目物体上の同一点が映ることになり、結果的に、全カメラの画素値（色）の単純な平均化では注目物体が映る自由視点映像上に遮蔽物体によるぼやけが生じる。

そこで、先行研究の手法では、遮蔽物体が多くカメラを遮らないという前提の下で、フォーカスする各カメラ画素のうちの多くが互いに類似する色を持つ場合は、それらに注目物体が映っていると判断し、それ

らの画素値を平均化して自由視点映像の画素に与える。逆に、注目物体の表面が焦点面上にない自由視点映像の画素では、フォーカスする各カメラ画素は遮蔽物体か背景の異なる位置の色を持ち、それらは互いに類似しない。この場合は、注目物体が映っていないと判断し、自由視点映像の画素には色を与えない。注目物体と同様の方法で、後方の背景にも焦点面を設置することで背景の抽出を行う。そして、抽出した注目物体に対して拡大・縮小、回転、平行移動の2次元エディット処理を適用し、抽出した背景の上に合成することで、自由視点映像をリアルタイムで描画する。

### (3) カメラアレイの分散管理

先行研究の手法では1台のPCのみを用いており、ハードウェアの制約から、接続できるカメラが20台までに制限されていた。本研究では、より多くのカメラを接続して管理するため、複数台のPCをネットワーク接続したPCクラスタを構築し、カメラを各PCに分けて接続することでカメラアレイを分散管理することとした。図3-1のように、カメラを数台ずつ接続した複数台の子PCと1台の親PCでPCクラスタを構築する。各カメラの映像が子PCから親PCに送信され、親PC上でエディット処理と自由視点映像の生成が行われる。

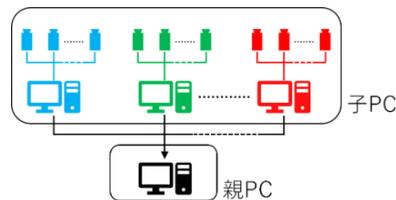


図3-1 PCクラスタを用いたカメラアレイの分散管理

### (4) 仮想空間の構築

グラフィックライブラリ OpenGL を用いて仮想空間を構築した。図3-2のように、立方体を注目物体とし、部屋を背景とした環境を構築した。この空間内に実際のカメラアレイを模擬する複数のカメラ視点を設け、オフスクリーンレンダリングによって疑似的なカメラ映像を生成する。前述のPCクラスタの実現に向けて、それぞれの子PCと親PCに個別のプロセスを割り当て、並列処理におけるデータ通信のためのMPIを用いて子プロセスで生成したカメラ映像を親プロセスへ送信する。その後、親プロセスでは、受信したカメラ映像から自由視点映像を生成してエディット処理を行う。また、様々な条件によるシミュレーションを実施するため、図3-3のシミュレータを実装し、様々な閾値等をインタラクティブに変更可能にしている。

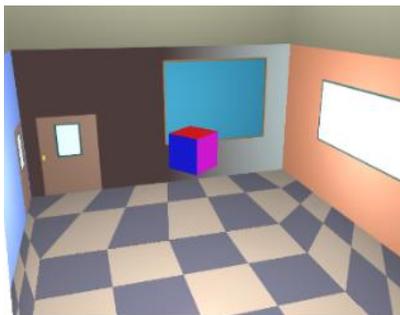


図3-2 構築した仮想空間



図3-3 仮想空間のシミュレータ画面

### (5) 実験

実空間上での実施においては、フレーム時刻ごとのカメラ画像の高速な送信が必要である。そこで、送信するデータ量の削減に向けて、画像圧縮コーデックを用いてカメラ画像を圧縮して送信する実験を行った。リアルタイム処理に適した圧縮効率、圧縮・復元時間が速い手法を検討するため、JPEG、PNG、WEBPの3種類の手法を比較した。実験結果を表3-1から表3-3に示す。この実験結果から、送信速度が大幅に遅れる場合には圧縮率の高いWEBPを使用し、送信速度が速い場合には圧縮・復元の速いJPEGを使用することが適切であることがわかった。

表 3-1 圧縮時間

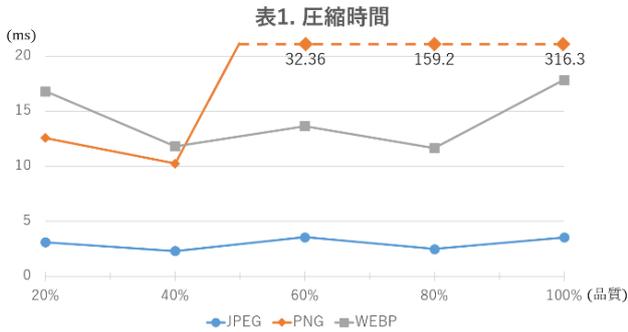


表 3-2 復元時間

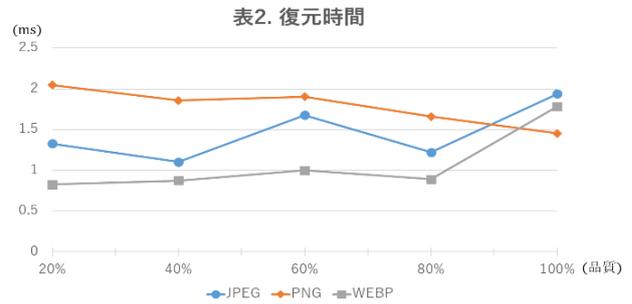
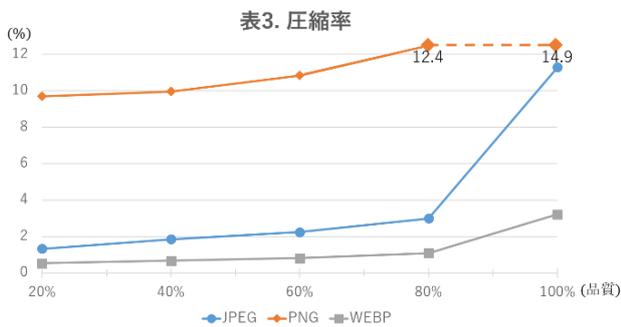


表 3-3 圧縮率



### (6) おわりに

本研究では、実時間ビデオエディットの拡張に向けて、多数のカメラを分散管理する方法を提案し、仮想空間の構築とシミュレータの実装を行った。今後は、本研究の結果をもとに、実空間上での実時間ビデオエディットの拡張を実現する予定である。

## 3 結論

本研究では、多視点 RGB-D データの高速転送のための自由視点依存の軽量化、実時間デプスデータの振動の軽減化、実時間ビデオエディットの拡張に向けたカメラレイ分散管理と仮想空間構築の3つの研究内容を実施した。いずれも有益な結果が得られたが、十分な成果に至らなかった内容も多い。それらを今後の課題として、研究を継続していく予定である。

### 【参考文献】

- [1] 佐々木悠輔, 藤本忠博, 複数の RGBD カメラを用いた自由視点映像生成のための基礎技術の開発, 平成 28 年度第 2 回芸術科学会東北支部研究会, 2016. 10. 8.
- [2] 佐々木悠輔, 藤本忠博, 複数の RGB-D カメラを用いた視点依存プリレンダリングによる効率的な自由視点映像生成法, NICOGRAPH 2017, 2017. 11. 10-12.
- [3] Yusuke Sasaki, Tadahiro Fujimoto, Efficient Virtual View Rendering by Merging Pre-rendered RGB-D Data from Multiple Cameras, IWAIT 2018, 2018. 1. 7-9.
- [4] 大門孝裕, 藤本忠博, カメラレイによる遮蔽物除去可能な注目物体抽出法, 平成 25 年度第 3 回芸術科学会東北支部研究会, 2013. 12. 7.
- [5] 大門孝裕, 藤本忠博, カメラレイによる注目物体の実時間エディット手法, 映像情報メディア学会メディア工学研究会, 2014. 2. 22.

- [6] Takahiro Daimon and Tadahiro Fujimoto, Interactive Video Editing for Occluded Object Using Synthetic Aperture Imaging, *The Journal of the Society for Art and Science*, Vol.14, No.5, pp.198-206, 2015.
- [7] 向井尚人, 藤本忠博, 焦平面スライプを用いた合成開口法による注目物体抽出の改善, 平成 28 年度芸術科学会東北支部大会, 2017.1.28.
- [8] 徐 迅, 藤本忠博, プレーンスライプ法を用いた部分遮蔽された凹凸のある物体の実時間抽出手法, 平成 30 年度第 1 回芸術科学会東北支部研究会, 2018.7.28.
- [9] 溝江史也, 藤本忠博, ビデオ映像上の遮蔽物体除去を伴う注目物体抽出法の高速化, 平成 30 年度第 3 回芸術科学会東北支部研究会, 2018.12.8.
- [10] 溝江史也, 藤本忠博, インタラクティブ・ビデオエディットにおける画素値のクラスタリング処理の高速化に関する検討, 令和元年度芸術科学会東北支部大会, 2020.1.25.
- [11] M. Levoy and P. Hanrahan, Light Field Rendering, *Proc. of SIGGRAPH 1996*, pp.31-42, 1996.
- [12] A. Laurentini, The Visual Hull Concept for Silhouette-based Image Understanding, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.16, No.2, pp.150-162, 1994.
- [13] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan, Image Based Visual Hulls, *Proc. of SIGGRAPH 2000*, pp.369-374, 2000.
- [14] W. Matusik, C. Buehler, and L. McMillan, Polyhedral Visual Hulls for Real-Time Rendering, *Proc. of Eurographics Workshop on Rendering 2001*, pp.115-125, 2001.
- [15] 松山隆司, 高井勇志, ウ小軍, 延原章平, 3 次元ビデオ映像の撮影・編集・表示, *日本バーチャルリアリティ学会論文誌*, Vol.7, No.4, 2002.
- [16] S. M. Seitz and C. R. Dyer, Photorealistic Scene Reconstruction by Voxel Coloring, *Computer Vision and Pattern Recognition Conf.*, pp.1067-1073, 1997.
- [17] A. Bogomjakov, C. Gotsman, and M. Magnor, Free-Viewpoint Video from Depth Cameras, *Proc. of Vision, Modeling and Visualization (VMV)*, pp. 89-96, 2006.
- [18] Andrew MacQuarrie and Anthony Steed, Improving Free-Viewpoint Video Content Production Using RGB-Camera-Based Skeletal Tracking, 2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), 2020.
- [19] 三攻浩嗣, 内藤 整, 距離画像センサを用いた空間再現手法の検討, 第 11 回情報科学技術フォーラム, 2012.
- [20] 橘高達也, 藤井浩光, 山下 淳, 浅間 一, 移動可能な RGB-D センサを用いた任意視点からの遮蔽物透視システム, *精密工学会誌*, Vol.83, No.3, pp.235-244, 2017 年.
- [21] Joel Carranza, Christian Theobalt, Marcus A. Magnor, and Hans-Peter Seidel, Free-Viewpoint Video of Human Actors, *ACM Transactions on Graphics*, Vol.22, No.3, pp.569-577, 2003.
- [22] A. Mammeri, B. Hadjou, and A. Khoumsi, A Survey of Image Compression Algorithms for Visual Sensor Networks, *International Scholarly Research Network (ISRN), Sensor Networks*, vol.2012, 760320, pp.1-19, 2012.
- [23] 野中敬介, 渡邊良亮, 塚本航平, 5G 網を利用した自由視点映像リアルタイム配信技術, *映像メディア学会誌*, Vol.74, No.1, pp.180-186, 2020.
- [24] M. Dou, S. Khamis, Y. Degtyarev, P. L. Davidson, S. R. Fanello, A. Kowdle, S. Orts-Escolano, C. Rhemann, D. Kim, J. Taylor, P. Kohli, V. Tankovich, and S. Izadi, Fusion4D: real-time performance capture of challenging scenes, *ACM Transactions on Graphics*, Vol.35, No.4, pp.114:1-114:13, 2016.
- [25] M. Dou, P. L. Davidson, S. R. Fanello, S. Khamis, A. Kowdle, C. Rhemann, V. Tankovich, and S. Izadi, Motion2fusion: real-time volumetric performance capture, *ACM Transactions on Graphics*, Vol.36, No.6, pp.246:1-246:16, 2017.

### 〈発表資料〉

題名	掲載誌・学会名等	発表年月
RGB-D データの高速転送のための自由視点依存の軽量化 (仲上浩豪, 藤本忠博)	令和2年度芸術科学会東北支部大会	2021年1月23日
実時間デプスデータの振動軽減法 (松浦知哉, 藤本忠博)	令和2年度第2回芸術科学会東北支部研究会	2021年3月17日
実時間ビデオエディットの拡張のための仮想空間構築とカメラレイ分散管理 (平井諒, 藤本忠博)	令和2年度第2回芸術科学会東北支部研究会	2021年3月17日