

博士論文

実世界を対象とした時空間映像データの  
高度利用システムに関する研究

(System for Advanced Use of the Spatiotemporal  
Imagery Data in the Real World)

2022年3月

立命館大学大学院情報理工学研究科  
情報理工学専攻博士課程後期課程

山崎 賢人

立命館大学審査博士論文

実世界を対象とした時空間映像データの  
高度利用システムに関する研究

(System for Advanced Use of the Spatiotemporal  
Imagery Data in the Real World)

2022年3月

March 2022

立命館大学大学院情報理工学研究科

情報理工学専攻博士課程後期課程

Doctoral Program in Advanced  
Information Science and Engineering Graduate School of  
Information Science and Engineering  
Ritsumeikan University

山崎 賢人

YAMAZAKI Kento

研究指導教員：柴田 史久 教授

Supervisor : Professor SHIBATA Fumihisa

# 実世界を対象とした時空間映像データの高度利用システムに関する研究

山崎賢人

## 内容梗概

将来、いたるところに設置された監視カメラや自動車の車載カメラなどによって、実世界を観測した映像が常時記録される社会が訪れようとしている。今後も我々の暮らす社会に存在するカメラの数は、増加の一途をたどるであろう。増加するのはカメラだけではない。近年の自動運転を見据えた自動車には、実世界を2次元的に観測するカメラだけではなく、3次元的に観測するLiDARが搭載されている。この一例からも実世界を常時観測するセンサは多種多様化することは必須である。さらに、今までカメラから得られる映像はクローズドなデータとして、カメラの所有者などの特定の組織でのみ活用されてきた。しかし、近年ではオープンデータ化の機運が高まっている。

これらの背景から、街中に多種多様なセンサが遍在する社会において、得られた観測データはオープンデータ化する時代が来ると予想できる。本研究では、先見した社会を見据え、これらの観測データをオープンデータとして収集・蓄積するアプリケーションフレームワークSIGMA(Spatiotemporal Images with Generalized Management Architecture)の研究開発に着手した。本論文では、SIGMAフレームワークの設計と実装について述べるとともに、本フレームワークを用いた観測データの高度な利用についても述べる。

本フレームワークの開発目標は、時間的かつ空間的に遍在する大量の観測データを一元管理し、ネットワークを介して様々なアプリケーションで利用可能にする仕組みを提供することである。この観測データを本研究では時空間映像データと称した。時空間映像データを共有するために、主に登録・蓄積・検索・利用の4つの機構を検討してきた。この4つの機構の中でも本フレームワークを実現する上で核となる機構のひとつである検索機構“SIGMA Retriever”を設計し、時空間映像データに対して、センサの特徴を活かした検索手法を提案を試みた。ただしセンサは多種多様であり、全てを網羅するようにシステムを構築することは不可能である。そのため、センサをカメラに限定した。また、利用方法は多視点幾何を用いたコンテンツ生成手法であるイメージベースドモデリング・レンダリング(Image-based Modeling and Rendering; IBMR)を想定しているため、一般的なIBMRを概観し、これらの処理の流れを抽象化した上で、検索手法を体系的にまとめた後に、カメラで撮影した画像に対して、カメラの特徴を活かした検索手法を提案するとともに、それを可能とするデータベースの設計した。設計した検索手法は鎌倉駅旧駅舎時計台を撮影した画像を蓄積したデータベースからIBMRの例である3Dモデルやパノラマ画像の生成に必要な画像を抽出することで、有用性を確認した。

同時に、本論文ではシステム設計からのアプローチだけではなく、利用方法からのアプローチについても検討している。利用方法の一例として、視野角によって生じる死角と遮蔽物によって生じる死角を、複数視点のカメラ映像を接合することで同時に解消する画像接合手法を提案す

る。視野角によって生じる死角は特殊な光学系を採用する方法や複数視点からのカメラ画像を接合する方法などで解消できる。一方、遮蔽物によって生じる死角については、隠背景を観測して、遮蔽物が占める領域に投影・合成する隠消現実感技術によって解消できる。しかし、これらの死角は同時に発生し得るため、各々を個別に解決すると非効率になる可能性がある。そこで本研究では、画像接合技術と隠消現実感技術に共通する考え方を整理し、両者を同時に解決する手法について検討した。その結果、画像接合のための境界線決定時に、遮蔽物を基に生成したマスク画像を用いることで、広視野画像から遮蔽物を除去した画像を生成できることを確認した。

最後に、生成したパノラマ画像中の外灯に対して、試作した SIGMA フレームワークを用いて、隠背景画像を検索し、外灯を視覚的に除去できることも確認した。

# System for Advanced Use of the Spatiotemporal Imagery Data in the Real World

YAMAZAKI Kento

## Abstract

In the future, I will live in a society in which images of the real world are constantly being recorded by surveillance cameras installed everywhere and in-vehicle cameras in automobiles. The number of cameras in our society will continue to increase in the future. However, it is not only the number of cameras that will increase. In recent years, cars that are designed for automated driving are equipped with not only cameras that observe the real world in two dimensions, but also LiDAR that observes it in three dimensions. This example shows that it is essential to have a wide variety of sensors to constantly observe the real world. Furthermore, until now, images obtained from cameras have been used as closed data only by specific organizations such as the owner of the camera. In recent years, however, there has been a growing trend toward open data.

With this background, I can expect that the era of open data will come in a society where a wide variety of sensors are ubiquitous in the city. In this research, I have started development of an application framework SIGMA (Spatiotemporal Images with Generalized Management Architecture), which collects and stores observation data as open data. In this paper, I describe the development of SIGMA. In this dissertation, I describe the design and implementation of the SIGMA framework, and the advanced use of observation data using this framework.

The goal of this framework is to provide a mechanism to centrally manage a large amount of observation data that is spatially and temporally ubiquitous, and to make it available to various applications via networks. This observation data is called "spatiotemporal imagery data" in this research. In order to share the spatiotemporal imagery data, I have studied four main mechanisms: registration, storage, retrieval, and utilization. I designed a retrieval mechanism "SIGMA Retriever", which is one of the core mechanisms to realize our framework, and attempted to propose a retrieval method for spatiotemporal imagery data using the characteristics of sensors. However, there are many kinds of sensors, and it is impossible to construct a system that covers all of them. Therefore, I limited the sensors to cameras. The proposed method is based on image-based modeling and rendering (IBMR), which is a content generation method using multi-view geometry. After systematically summarizing the methods, I proposed a retrieval method for images captured by a camera that takes advantage of the features of the camera, and designed a database that enables the method. The usefulness of the designed retrieval method was confirmed by extracting images

necessary for generating 3D models and panoramic images, which are examples of IBMR, from a database of images taken of the clock tower of the old Kamakura Station.

Simultaneously, in this dissertation, I study not only the approach from the system design perspective but also the approach from the usage perspective. As an example, I propose a photo stitching method to solve two kinds of blind spots on a camera image at the same time. Blind spots due to the viewing angle can be eliminated by employing a special optical system or by stitching multiple viewpoint images. On the other hand, blind spots due to occluding objects can be eliminated by diminished reality (DR) technology, which visualizes occluded areas by replacing occluding objects with hidden area images. However, these blind spots could occur at the same time, so that it is inefficient to solve two problems separately. Therefore, I considered a method for solving both types of blind spots simultaneously by extracting common ideas between photo stitching and DR technologies. As a result, I found that the method which defines stitching seams with a mask images of occluding objects can generate a wide-field image without occluding objects.

Finally, I used the prototype SIGMA framework to search for hidden background images for the exterior lights in the generated panoramic images, and confirmed that the exterior lights can be visually removed.

# 目次

第 1 章 序論.....	1
第 2 章 関連研究.....	5
第 3 章 時空間映像データの共有 .....	7
3.1 基本方針 .....	7
3.2 SIGMA 座標系 .....	8
3.3 時空間映像データの管理方法 .....	11
3.4 リソースデータベースの設計 .....	16
第 4 章 サーバの設計 .....	18
4.1 設計方針 .....	18
4.2 AP サーバと管理者用アプリの定義 .....	19
4.3 管理者用アプリ .....	20
4.3.1 ROI を用いた圧縮機構 .....	20
4.3.2 位置姿勢推定機構 .....	21
第 5 章 SIGMA Retriever の設計.....	24
5.1 イメージベースドモデリング・レンダリング概観.....	24
5.2 処理の流れの抽象化.....	25
5.3 設計方針.....	26
5.4 検索手法の提案 .....	27
5.4.1 内部パラメータをキーとした検索手法.....	27
5.4.2 外部パラメータをキーとした検索手法.....	28
第 6 章 SIGMA フレームワークの試作と評価 .....	37
6.1 評価環境の構築 .....	37
6.2 評価のための準備 .....	38
6.3 評価方法 .....	40
6.4 「外部パラメータをキーとした検索」の動作確認.....	40
6.5 「内部パラメータをキーとした検索」の評価 .....	41
6.5.1 評価 1 : 「注視点に基づく検索」の比較評価 .....	41
6.5.2 評価 2 : 「視錐台カリングに基づく検索」の比較評価.....	42

6.5.3	評価3:「注視点に基づく検索」と「視錐台カリングに基づく検索」の比較評価	45
6.5.4	評価4:「エピソード幾何に基づく検索」の比較評価	46
6.6	まとめ	47
<b>第7章</b>	<b>高度利用方法の検討</b>	<b>49</b>
7.1	検討方針	49
7.2	コンテンツの選定	50
7.3	隠消現実感技術の検討	51
7.4	広視野画像の生成方法	52
<b>第8章</b>	<b>広視野画像のための隠消現実感達成法の提案</b>	<b>56</b>
8.1	事前検討	56
8.2	提案手法	60
8.3	Graph Cut への応用	63
<b>第9章</b>	<b>広視野画像のための隠消現実感達成法の評価</b>	<b>65</b>
9.1	評価の目的と環境	65
9.2	画像生成結果	68
9.3	処理時間比較評価	68
9.4	まとめ	70
<b>第10章</b>	<b>SIGMA フレームワークにおける高度利用</b>	<b>71</b>
<b>第11章</b>	<b>結論</b>	<b>74</b>
	謝辞	76
	参考文献	77
	研究業績	82



## 図目次

図 1-1 遍在したセンサのイメージ	1
図 1-2 前方車両による視覚領域の除去	2
図 1-3 SIGMA フレームワークのイメージ図	3
図 2-1 PLATEAU VIEW による皇居周辺のモデル	5
図 3-1 時空間映像データを共有するための 4 フェーズ	7
図 3-2 異なるアプリ間での情報共有	8
図 3-3 登録ツールとアップロードの流れ	9
図 3-4 日本の平面直角座標系 ([17]引用)	10
図 3-5 UTM グリッド ([18]引用)	11
図 3-6 木構造における Object と Resource	12
図 3-7 Object の位置姿勢の決め方	13
図 3-8 群集とみなす例	14
図 3-9 Object の設定方法	15
図 3-10 テーブルの関係	16
図 3-11 グリッドの分割	17
図 4-1 サーバのアーキテクチャ	18
図 4-2 AP サーバでの処理イメージ	19
図 4-3 空間軸に指定した Bounding Box と ROI	20
図 4-4 SfM の処理イメージ	22
図 5-1 抽象化した IBMR の処理の流れ	25
図 5-2 全画像を検索対象にした例	26
図 5-3 提案手法の分類	28
図 5-4 時空間による検索	29
図 5-5 領域の和集合	30
図 5-6 円環による検索	30
図 5-7 視錐台カリングに基づく検索	32
図 5-8 被観測物とカメラとの距離	33
図 5-9 注視点に基づく検索	33
図 5-10 エピポーラ幾何に基づく検索	35
図 5-11 エピポーラ幾何に基づく検索のフローチャート	36
図 6-1 評価の種類	37
図 6-2 撮影した画像の一例	38
図 6-3 時計台と周辺の三次元再構成 (プロットした点はカメラの撮影位置を示す)	39
図 6-4 検索範囲	41

図 6-5 時間軸による変化.....	41
図 6-6 案内図（評価 1 の結果） .....	43
図 6-7 時計台（評価 2 の結果） .....	43
図 6-8 ROI 適用との比較.....	44
図 6-9 圧縮品質の比較 .....	45
図 6-10 時計台（評価 3 の結果） .....	46
図 6-11 「エピポーラ幾何に基づく検索」の評価結果 .....	47
図 7-1 死角の種類.....	51
図 7-2 隠消現実感の構成要素（[53]引用） .....	52
図 7-3 パノラマ画像を構成するカメラ画像群 .....	53
図 7-4 球状の投影面 .....	53
図 7-5 マスク画像.....	54
図 7-6 Brown らの手法によるパノラマ画像 .....	55
図 8-1 処理フローの改善.....	56
図 8-2 投影面を用いた投影 .....	57
図 8-3 隠背景領域の抽出.....	58
図 8-4 隠背景投影画像を重畳したパノラマ画像 .....	59
図 8-5 領域分割 .....	60
図 8-6 マスクを用いて生成したパノラマ画像 .....	61
図 8-7 2 種類のマスク画像.....	62
図 8-8 提案手法による領域分割.....	62
図 8-9 Graph Cut による例 .....	64
図 9-1 カメラ配置（環境 A） .....	66
図 9-2 カメラ配置（環境 B） .....	67
図 9-3 評価時における提案手法の処理フロー .....	68
図 9-4 実行結果 .....	69
図 9-5 処理速度の比較 .....	69
図 10-1 検索範囲と「エピポーラ幾何に基づく検索」の位置.....	71
図 10-2 試行 1 の結果.....	71
図 10-3 撮影画像を用いて生成した広視野画像 .....	72
図 10-4 抽出した隠背景画像 .....	72
図 10-5 パノラマ画像を構成する画像とそれに対応するマスク画像.....	73
図 10-6 外灯を除去した画像 .....	73

## 表目次

表 6-1 撮影時期と撮影枚数 .....	38
表 9-1 PC の仕様.....	68

## 第1章 序論

いたるところに設置された監視カメラや自動車の車載カメラなどによって、実世界を観測した映像が常時記録される社会が訪れようとしている。近年活躍が期待されている低空を安定して飛行できる UAV (Unmanned Aerial Vehicle)などの自律型移動体にもカメラは欠かせない。今後も我々の暮らす社会に存在するカメラの数は、増加の一途をたどるであろう (図 1-1)。

増加するのはカメラだけではない。近年の自動運転を見据えた自動車には、2 次元的に実世界を観測するカメラだけではなく、3 次元的に観測する LiDAR が搭載されている。この一例からも実世界を常時観測するセンサは多種多様化することは必須である。さらに、今までカメラなどのセンサから得られるデータはクローズドなデータとして、カメラの所有者などの特定の組織でのみ活用されてきた。しかし、近年ではオープンデータ化の機運が高まっている。国土交通省主導の 3D 都市モデルの整備・オープンデータ化プロジェクト「PLATEAU」もその一例である[1]。筆者は近い将来、街中に遍在するカメラなどのセンサから得られるデータをオープンデータ化する時代が来ると予想している。こうした時代を見据え、これらのデータをオープンデータ化するシステムの研究開発に取り組むことにした。

本研究では、カメラなどのセンサから得られる映像・データを議論してきた。しかし、広範なセンサを汎用的に扱えるシステムを構築することは難しい。そこで、本論文では最も身近なセンサであるカメラを中心に議論を深めていく。

街中に遍在するカメラから得られる映像は、単独で使用するよりも、多視点の映像を組み合わせるほうが、利用の幅が広がる。前述のとおり、センサはカメラに絞ったが、カメラによる多視点映像の利用方法だけでも、物体認識など、様々なものがある。こうした背景から、本研究では、利用方法も多視点幾何 (Multi View Geometry; MVG)を用いたデジタルコンテンツ生成手法で

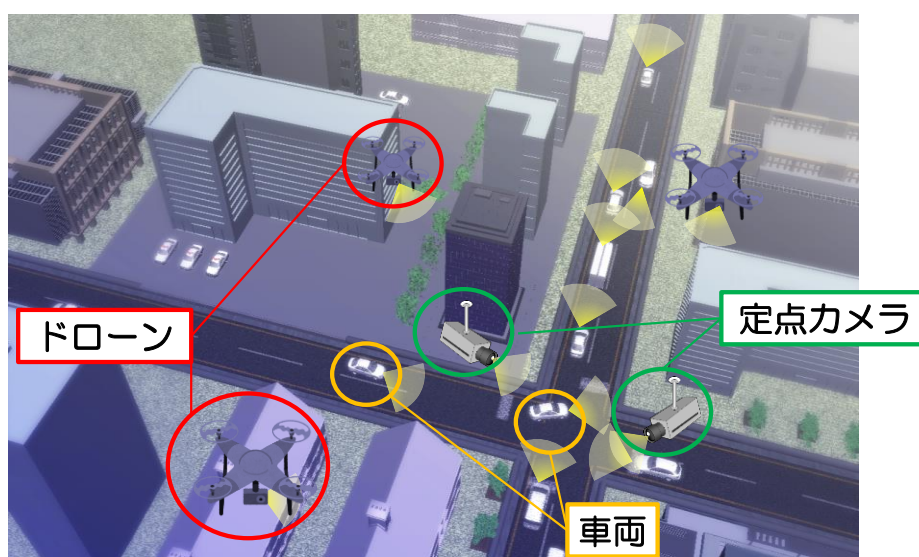


図 1-1 遍在したセンサのイメージ

あるイメージベースドモデリング・レンダリング (Image-based Modeling and Rendering; IBMR)に焦点を絞ることとした[2].

IBMR では、文献[3]のように複数視点の画像に基づいて 3D モデルを生成したり、広視野画像や仮想視点画像など撮影画像とは異なった風景を描画することが可能である。文献[3]は空間的に広がった映像を利用したものであるが、映像は空間だけではなく、時間的にも広がりを持つ。例えば、カメラで取得した映像を蓄積することで、Mapillary などのように、ほぼ同じ位置で撮影した過去の映像を見ることができる[4]。さらに現在の映像の一部に過去の映像を組み合わせる試みも始まっている[5]。東日本大震災の被災地を対象として、震災前後の様子を仮想化空間で再現するような試みもある[6]。これらはカメラの映像を蓄積しているが、蓄積するだけでは目的を達成できない。例えば、文献[7]では、自動車の運転手の眼前の景色を前方車両が遮ることによって生じる死角領域に対して、隠消現実感技術を用いて別視点の自動車で撮影した画像を重畳することによって死角領域を除去している (図 1-2)。この研究では、重畳に適した画像を撮影できていることを前提に、課題の解決を図っている。多くの研究でも同様に遍在する画像から適した画像を抽出することは容易ではない。これは、カメラなどの視覚センサの特徴として、ジャイロスコープのようにセンサ自身の状態を計測する種類のセンサではないことに起因する。すなわち、視覚センサはセンサ自身の位置を情報として得るのではなく、センサが向いている方向の光景を情報として得ているからである。例えば、任意の被写体が写っている画像を抽出したい場合、その被写体の位置を基にカメラの位置を検索しても、必ずしもそのカメラが撮影した画像に任意の被写体が写っているとは限らない。つまり、カメラが大量に存在するとしても、実世界を観測した情報を確認するまでは、どのカメラからの画像が適しているを判断することが難しい。

さらに、必ずしも使用した画像の数に比例して、高精度なコンテンツが生成されるとは限らない。本研究で対象とする画像には、画像は監視カメラのように固定されたカメラで撮影した画像

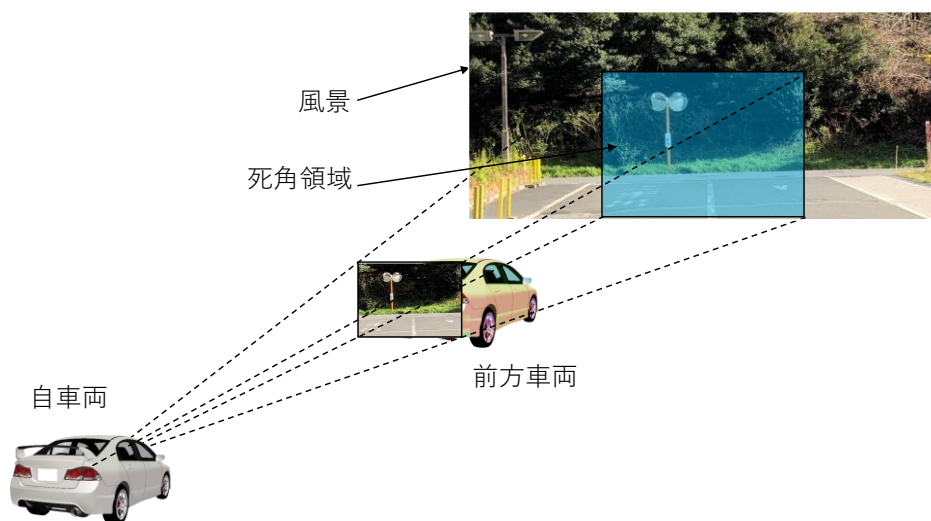


図 1-2 前方車両による視覚領域の除去

だけではなく、UAV や自動車などに搭載したカメラで撮影した画像も含むことを想定すると、たとえ、撮影位置が既知の画像を街中から大量に収集・蓄積できていたとしても、その中から利用目的に合った画像を得られなければ、収集・蓄積した大量の画像を使用することができないため、利用者のニーズを満たすことはできない。

本研究では、前述した視覚センサにより実世界を観測した映像・データを大規模に収集・蓄積するだけでなく、コンテンツ作成までを総合的に扱えるシステムの構築を目的としている。本システムでは時間的・空間的に遍在する映像・データを時空間映像データと称し、これらを総合的に扱えるシステムとして、アプリケーションフレームワーク SIGMA (Spatiotemporal Images with Generalized Management Architecture)を提案する (図 1-3)。

今までも SIGMA フレームワークを構築するために、概念設計や基本アーキテクチャの議論を重ねてきたが[8]、SIGMA フレームワークを実現するためには、時空間映像データの共有方法をより明確化する必要がある。そこで本論文では、視角センサの中でも、最も普及しているカメラを対象を限定した上で、以下の2つを要求仕様とした。

1. 蓄積した時空間映像データはシステム上で時間軸と空間軸の値を画一的に扱うことが可能である
2. 時空間映像データを共有でき、時空間映像データ特有の検索方法を有する

また、本研究では、これらの要求仕様を満たす SIGMA フレームワークを設計・構築するとともに、SIGMA フレームワークを使用して開発するアプリケーションも検討している。そこで、本論文では、多くの実課題の中から、IBMR を用いて解決できそうな課題を選定し、ソリューションを提案したことについて述べる。具体的には、2種類の死角に対して、Photo Stitching と隠消現実感 (Diminished Reality; DR)を組み合わせることで、効率よく2種類の死角を除去できる手法を提案した。

このように、様々なソリューションとしての高度利用例も検討することで、取得できるデータ

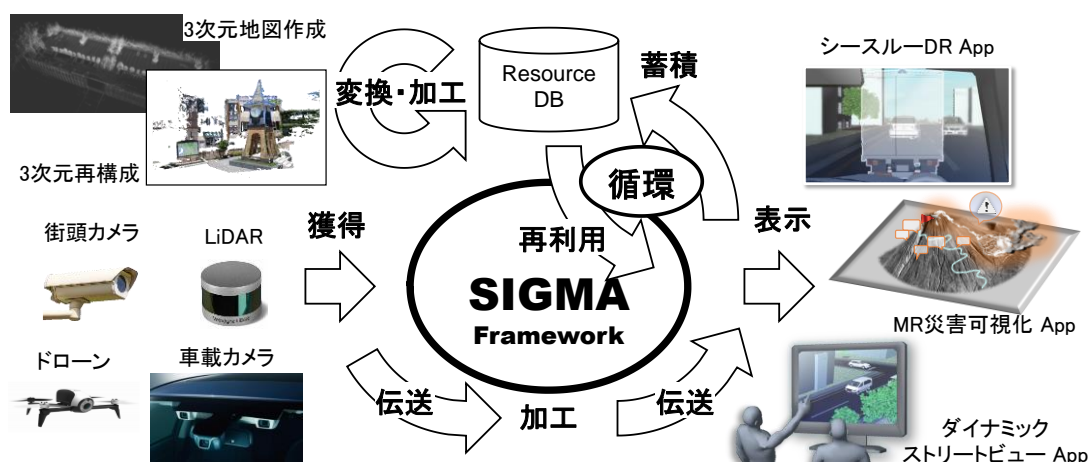


図 1-3 SIGMA フレームワークのイメージ図

からのアプローチと、開発するコンテンツからのアプローチ、双方向から SIGMA フレームワークを練ることで、より俯瞰的な視点で設計することが可能となる。

以降、2章で関連研究を概観したのち、3章において時空間映像データの共有について定義する。4章ではサーバの設計について述べる。5章で時空間映像データ特有の検索方法を提案し、データベース (Database; DB)の設計について述べる。6章では、SIGMA フレームワークの試作システムについて評価し、それに対して考察する。7章は SIGMA フレームワークの利用例について検討し、本研究における実課題を選定する。8章では、広視野画像のための隠消現実感達成法を提案し、9章では提案手法を評価する。10章では、SIGMA フレームワークを用いてコンテンツ生成を試行する。11章では、まとめについて述べる。

## 第2章 関連研究

実世界を観測した情報を蓄積し、共有するシステムは多く存在する。例えば、AWE (Augmented World Expo)の創設者のひとりである Inbar が提唱した AR Cloud もそのひとつである[9]。AR Cloud では以下の 3 つの機能を提唱しており、これを実現するために様々なプロジェクトが立ち上がっている。この 3 つのうち、2 つは情報の共有に主眼を置いている。

- A scalable shareable point cloud
- An instant ubiquitous localizer
- Realtime multiuser interaction

何らかのデータを共有する際には、そのデータを記述するフォーマットを標準化することが重要となる。3次元地理空間分野において、標準化団体 Open Geospatial Consortium が CityGML を策定しており、世界中で CityGML を用いた 3D 都市モデルの整備が進められている[10]。例えばフィンランドでは、ヘルシンキ市全域の 3D 都市モデルの整備を進めた「Kira-digi」を 2016 年から 2018 年まで実施した。シンガポールでも「Virtual Singapore」が推進されている。日本でも、国土交通省が主導する「Project PLATEAU」によって全国 56 都市の 3D 都市モデルが整備されており、2021 年 3 月 26 日に“PLATEAU ver1.0”をリリースするとともに、全国に先駆けて東京 23 区の 3D 都市モデルのオープンデータが公開された(図 2-1)。時空間映像データを多くの人と共有するためには、目的に合わせたデータフォーマットの設計が必要となる。

ITS (高度道路交通システム) 分野では点群を自動運転などに活用している。このときの点群は 3 次元地図と称することが多い。3 次元地図とは、道路や建造物といった静的な物体を点群

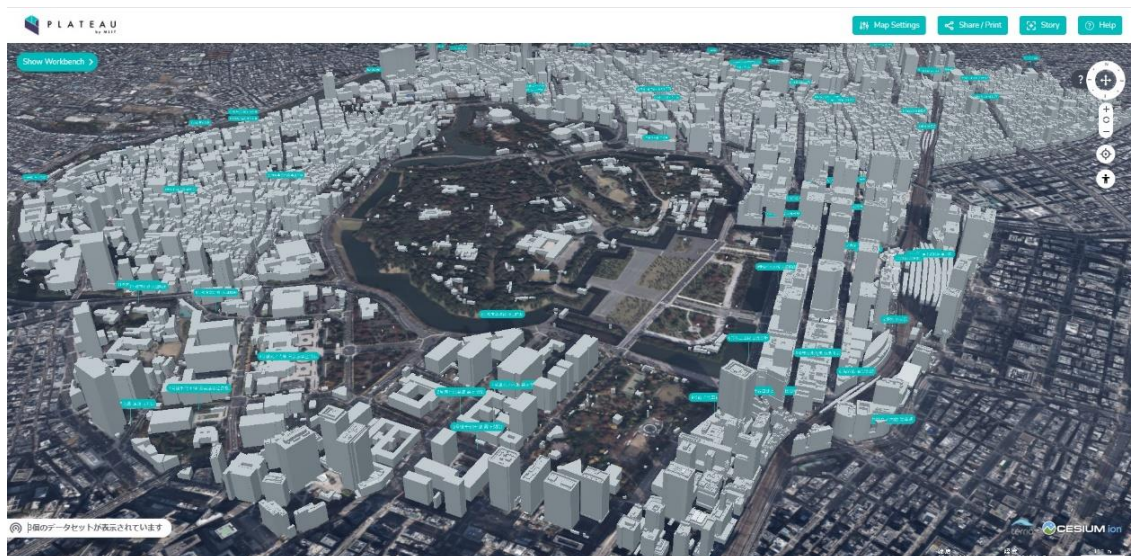


図 2-1 PLATEAU VIEW による皇居周辺のモデル

出典：国土交通省ホームページ (<https://plateauview.jp/>)



で表現する 3 次元の地図であり、モービルマッピングシステム (MMS) などによって作成されるが[11], 3 次元地図を利用するメーカーごとに作成・保守できるスケールではない。そこで 3 次元地図をインフラとして扱うため、各社の垣根を越えた整備が進められている。しかし、3 次元地図は一度作成すればよいわけではない。静的な物体であっても時間経過によって形状が変化することがあるため、定期的な地図の更新は必要である。そこで Sakurada らは異なる時刻に撮影した画像から 3 次元形状を更新する方法を提案した[12]。他にも物体の 3D モデルを作成するにあたって、人物などの動体を、時間情報を用いて除外するなどの活用事例もある[13]。

このように 3 次元情報のみを着目した場合では解決できなかった課題に対し、時間軸を含めた 4 次元で考えることによって様々な課題を解決する糸口が見出されてきた。

4 次元の情報を扱うだけではなく、多種多様なセンサから得られた情報を共有するためには、情報の管理方法が重要となる。川住らはスマートフォンなどの移動型端末から得られたセンサデータを管理するため、センシング時刻の分布を考慮した時空間インデックス構築手法を提案している[14]。文献[14]におけるセンサデータは位置や温度などを指すが、本研究が対象するとセンサデータは主に映像であり、センサ位置と被写体の位置が一致しない。そこで本研究が提案するフレームワークでは実世界を観測したデータを高度利用できるようにするため、映像である特徴を活かし、汎用的に 4 次元情報を共有/活用できるシステムを構築する。

そのためには、システム全体を俯瞰して設計する必要がある。筆者はスマートフォンに代表されるモバイル機器を念頭に、複数端末でコンテンツを共有する複合現実感 (Mixed Reality; MR) システムを開発した知見がある[15]。このシステムでは、モバイル機器特有の課題として性能に応じた情報提示のスタイルを変更できる枠組みを実現している。多くのシステムがそうであるように、本研究でもデータを収集するデバイスなどの対象を決める必要があるため、今までの知見を活かしてシステムを設計していく。次章では、そのための設計方針について述べる。

## 第3章 時空間映像データの共有

### 3.1 基本方針

本論文ではカメラで得られた映像を中心に議論するが、視覚センサ全般を対象とした時空間映像データを扱うことは視野に入れて議論している。ここでは、時空間映像データが撮影から利用までの流れを整理した。本研究では時空間映像データを共有することが重要であり、共有するためには、主に登録・蓄積・検索・利用の4フェーズが考えられる(図3-1)。これらのフェーズの内、データ共有に特に重要となるのは、蓄積および検索の仕組みである。そこで、本章では主に蓄積についての検討結果について述べ、検索については5章で詳細を説明する。時空間映像データはカメラの周辺を観測したデータであるため、観測時の情報が重要となる。例えば、カメラ画像を幾何学的に利用する場合、カメラの位置姿勢や内部パラメータは必要な要素となる。これらを共有するためには、カメラ画像だけではなく観測時の情報なども含めてDBに蓄積する必要がある。本フレームワークでは、時空間に遍在したカメラ画像を扱うため、まずは観測時の空間と時間を指標に、4次元情報を基に時空間映像データを整理する。また、内部パラメータなどの観測時のカメラの状態もメタデータとして蓄積する。

4次元情報を基に蓄積するためには、全時空間映像データが同じ座標系で扱われることが好ましい。しかし、多くのアプリケーション(以下、アプリ)は、そのアプリ固有の座標系を有しており、そのアプリで使用するデータは、そのアプリ内で閉じていることが多い。しかし、本論文で扱う時空間映像データを共有するため、オープン性が求められる。そのため、SIGMAフレームワークを用いて開発したアプリでは、同一アプリ内に限定した共有ではなく、異なるアプリからの時空間映像データの共有も想定している(図3-2)。これらのことから筆者は、実世界を観測して得られた時空間映像データを多種多様なアプリで利用するために、センサから取得したデータをそのまま保存し、利用方法はアプリ開発者に一任するが、4次元の座標系は統一する方針を立てた[16]。本フレームワークが扱うデータは一意の場所・時間で得られたデータではなく、

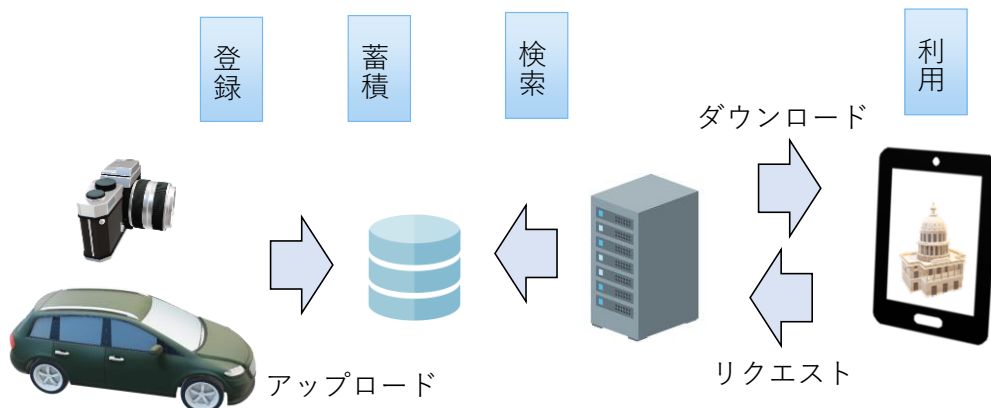


図3-1 時空間映像データを共有するための4フェーズ

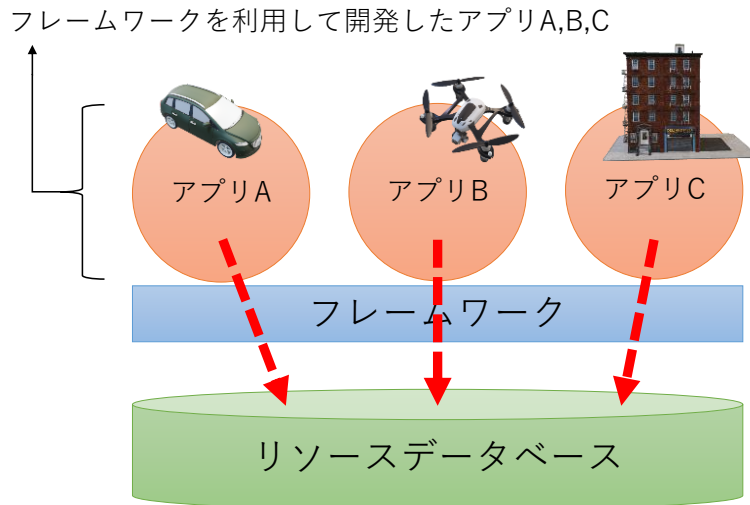


図 3-2 異なるアプリ間での情報共有

様々な場所・時間に遍在する。これらを共有するためには、いかにサーバで一元管理し、サーバ上のリソース DB から任意のデータにアクセスできるかが重要となる。本フレームワークでは、データに自由にアクセスする交換条件として、本フレームワークを用いて開発したアプリは時空間映像データを自動的にアップロードし、データを共有することを前提とした。アップロード機能自体をアプリ開発者に委ねた場合、誤った座標系に基づいて位置姿勢を送信する可能性がある。そこで、アップロード機能はアプリ開発者が実装せず、本フレームワークに内包された機能としてバックグラウンドで自動的に実行されるものとした。

次に、カメラ画像の関係について検討する。近年、自動車やスマートフォンなどの多くの機器は複数台のカメラを搭載しているものが普及し始めている。特にスマートフォンでは、画角や焦点距離が異なるレンズを搭載したカメラが複数搭載されているものが一般的になっている。これらのカメラは各々が独立しているのではなく、他のカメラとの関係が存在する。例えば、ステレオカメラは、2つのカメラで奥行きを計測するために、互いが同じステレオカメラを構成するカメラとして、両者が関係を把握していなければならない。しかし、遍在するカメラ画像は統一された座標系上の情報を保持したままリソース DB に蓄積した場合、それは独立した画像であって、実世界における関係も含めて保持することはできない。そこで、このような関係を維持するために本研究では、個々のセンサ間関係を、データ構造を用いて明示し、蓄積することを検討した。

ここまで述べた方針を基に 3.2 節では座標系を定義し、3.3 節では時空間データの管理方法について述べる。3.4 節ではこれらを満たすリソース DB の設計について記述する。

## 3.2 SIGMA 座標系

様々なアプリからアップロードされた時空間映像データを管理するためには、座標系を統一する必要がある。各アプリではアプリ開発者が定義する座標系 (アプリ座標系) の使用を許容す

るが、アップロードする時空間映像データでは、SIGMA 座標系に変換してからとする。アプリ開発者に SIGMA 座標系を意識させないため、バックグラウンド処理内で座標系を変換する(図 3-3)。同様に、ダウンロード時もバックグラウンド処理内で SIGMA 座標系からアプリ座標系に変換する。

SIGMA 座標系は、1 章で述べた要求仕様(1)にもあるとおり、システム上は空間軸 3 つと時間軸 1 つからなる 4 次元座標系である。空間軸を衛星測位システム (Global Navigation Satellite System; GNSS) などのように地球規模で使用するシステムにおいては経緯度を用いて定義した場合、3 次元のうち 2 次元は角度で表記されるのに対して、もう 1 次元は長さとなり、単位が異なる。さらに緯度は北緯/南緯、経度は東経/西経のように、回転の向きが赤道や本初子午線を境に変化するなど、管理が複雑となる。本フレームワークでは、センサで取得した広範囲な情報を収集する前提から、地球規模かつ、直観的な直交座標系で定義したいが、地球規模の場合、単一の直交座標系を定義することは難しい。そこで、本フレームワークでは、地球規模であっても直交座標系で表現する方法を検討した。

世界中で様々な地理座標系が使われているが、単一の直交座標系で表現することはできない。

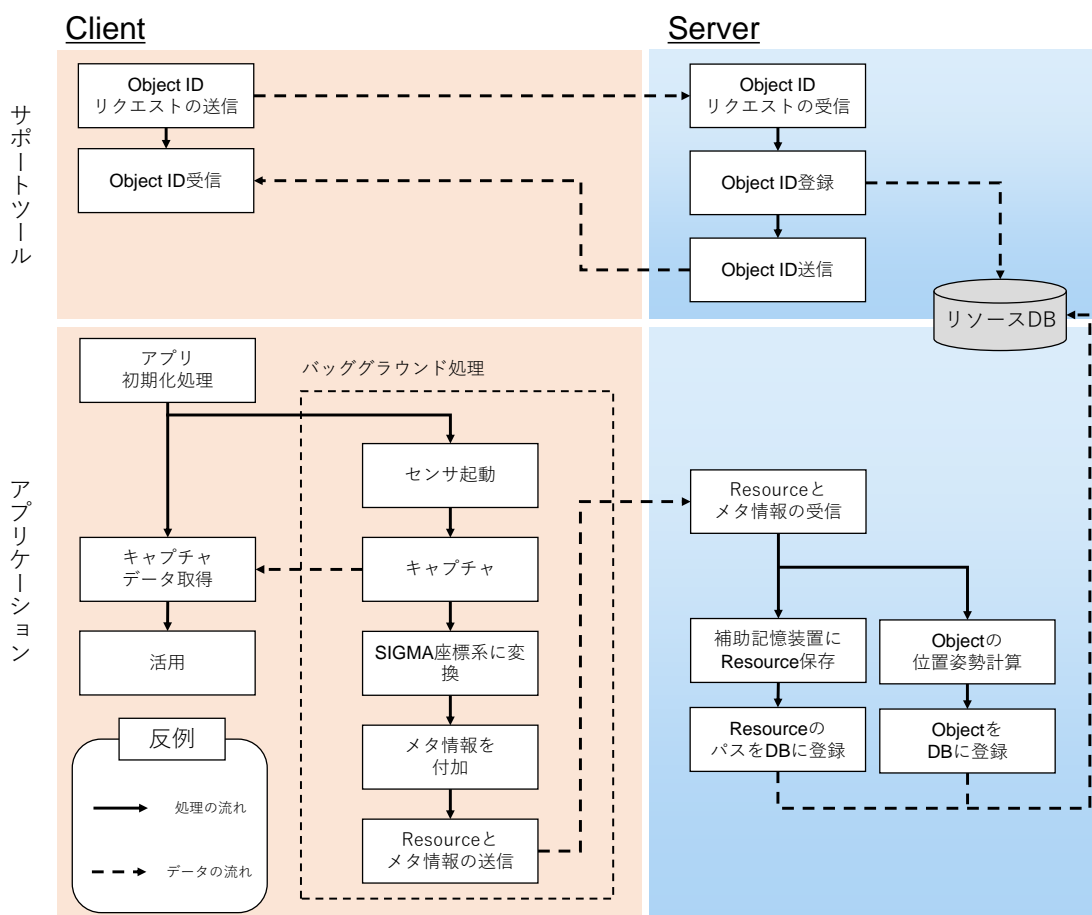


図 3-3 登録ツールとアップロードの流れ

日本における公共測量にはガウス・クリューゲル図法を採用した平面直角座標系が用いられており、国土地理院刊行の大縮尺地図は平面直角座標系を利用している[17]。この座標系では、日本を 19 のゾーンに分割した座標系で表す (図 3-4)。

このように空間軸を直交座標系で表すにはゾーンの分割が必須である。しかし、日本国内で使用されている平面直角座標系が表す範囲は日本国内に限定されている。そこで、本フレームワークでは、全世界で統一的に使用できる UTM (Universal Transverse Mercator)座標系を SIGMA 座標系に採用した。UTM は全世界を経度 6 度ごとのゾーンに分けて表現する座標系であり、日本では国土地理院の地形図や地勢図で採用されている。UTM 座標系も平面直角座標系と同じで直交座標系にするため、平面直角座標系と同様にゾーンを分割する必要がある。図 3-5 に示すとおり、MGRS (Military Grid Reference System)によって分割されている[18]。SIGMA 座標系でも MGRS を基に分割して扱う。

次に高さ軸について検討する。センサで高さを取得するには GNSS を用いることが一般的であるが、GNSS で取得できる高度は楕円体高である。しかし、地図に記載されている等高線や水準点は標高であるため、楕円体高では扱いにくい。そこで楕円体高からジオイド高を引いた標高を SIGMA 座標系の高さ軸と定義した。これらから、位置はゾーンと  $(x, y, z)$ 、姿勢は四元数

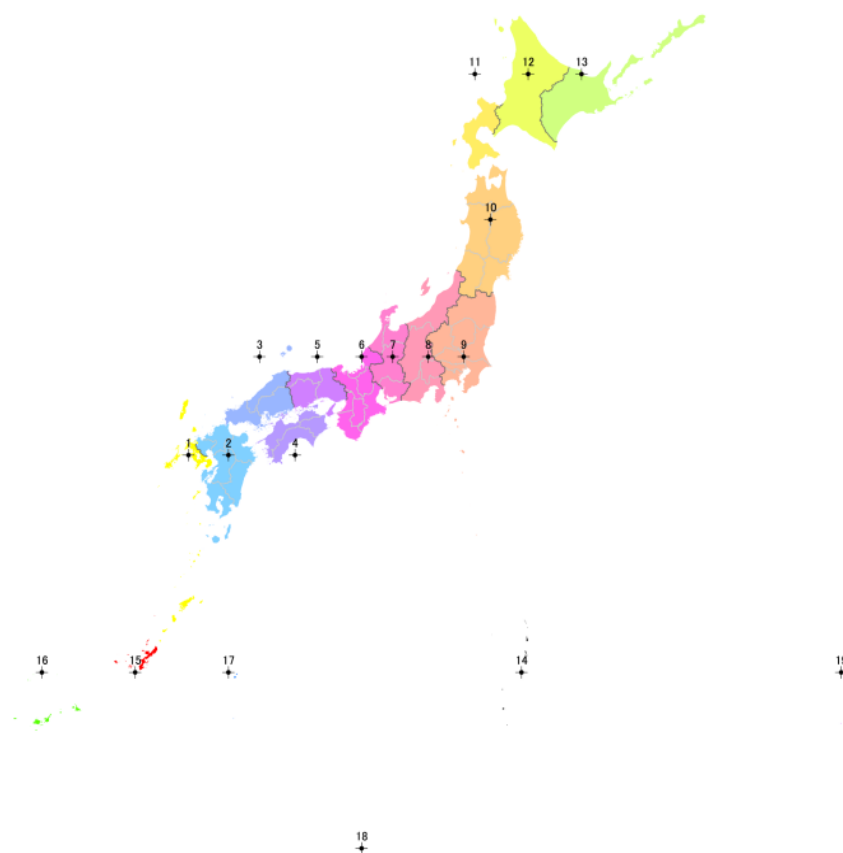


図 3-4 日本の平面直角座標系 ([17]引用)

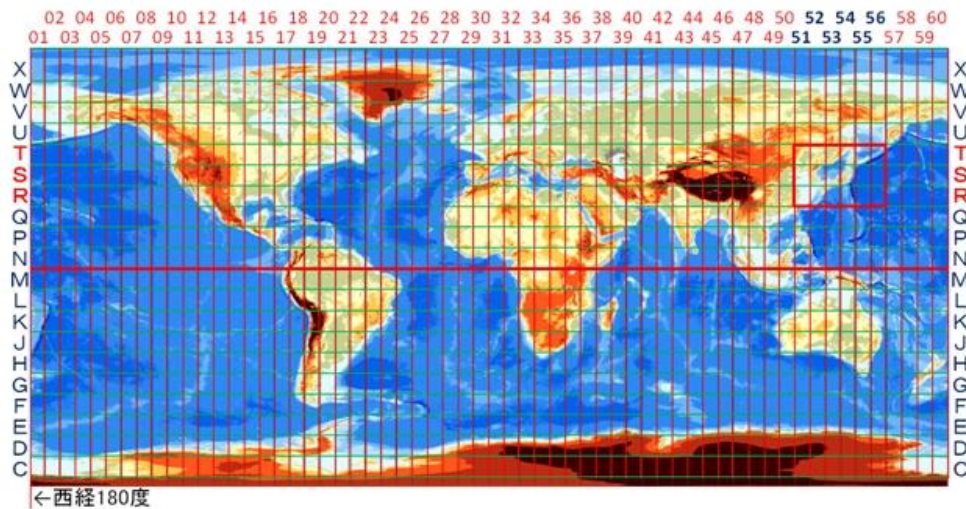


図 3-5 UTM グリッド ([18]引用)

$(x, y, z, w)$ で表すことができる。

最後に時間軸について検討する。時間は、24 時間や 60 分など、様々な進数で周期している。空間軸が 10 進数の座標軸であるため、時間軸も画一的に扱うためには、同様の座標軸で表記する必要がある。そこで、UNIX 時間と同じように任意のタイミングからの経過時間で表すものを使用する。ただし、UNIX 時間では粒度が秒であることから、一般的なセンサのフレームレートより粗い。SIGMA 座標系の時間軸では ROS Time のようなミリ秒まで扱えるシステム時刻を想定する。

### 3.3 時空間映像データの管理方法

データ構造において、カメラをノードに見立てた場合、同一カメラであってもパラメータが異なると、同一カメラとして扱うのに不都合が生じる可能性がある。カメラ画像をノードにした場合、各ノードをエッジで結ぶと、ノード間の関係を定義することが難しい。時間的な関係や空間的な関係で定義した場合、全ノード間の関係の保持が求められる。そこで、カメラなどのデバイスではなく、独自に定義した抽象的な「Object」をノードとみなし、親ノードは Object の上位概念とする木構造を構築した。また各ノードにはカメラ画像などの時空間映像データとメタ情報からなる「Resource」を紐づけする構造を設計した。Object を木構造にすることで、任意のノードを根とする部分木の全ノードは、任意のノードの属性を有しているため、部分的な抽出がしやすくなる。具体例を図 3-6 に示す。この図は自動車に搭載された複数のセンサから木構造を構成する例であり、車両の前方にステレオカメラを搭載していたとする。各カメラ画像は「右」Object、「左」Object に紐づき、「右」Object、「左」Object は上位概念として「フロント」Object を親に持つ。さらに「リア」Object と「フロント」Object は上位概念である「自動車」Object を親に持つ。「フロント」Object に紐づく Resource を指定した場合、時刻  $t$  では 2 枚の画像を取得できるが、「自動車」Object に紐づく Resource を指定した場合、「フ

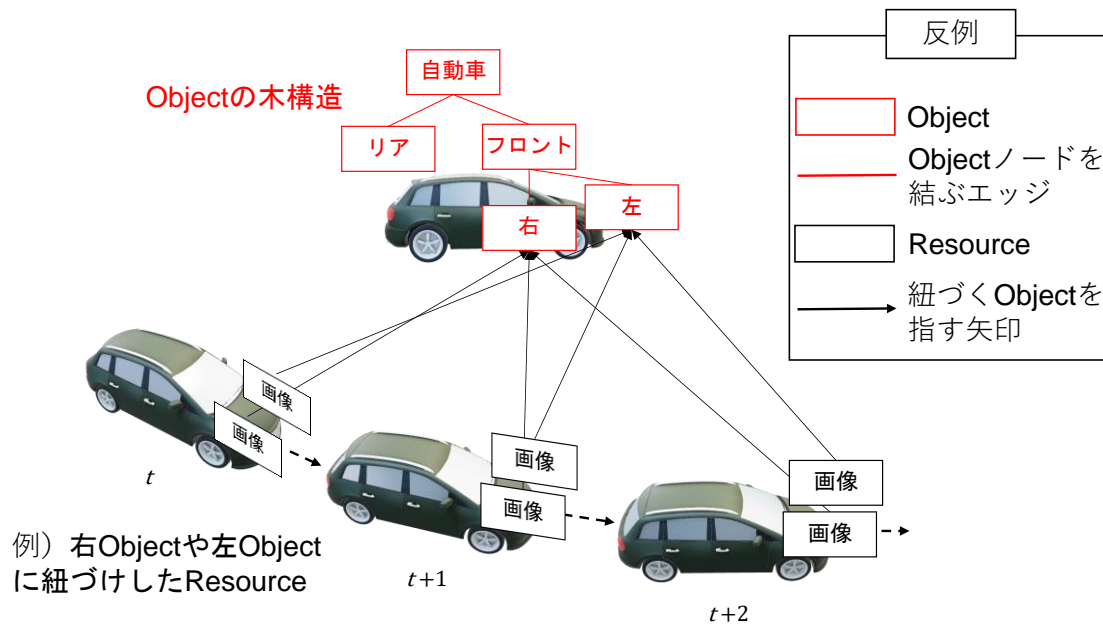


図 3-6 木構造における Object と Resource

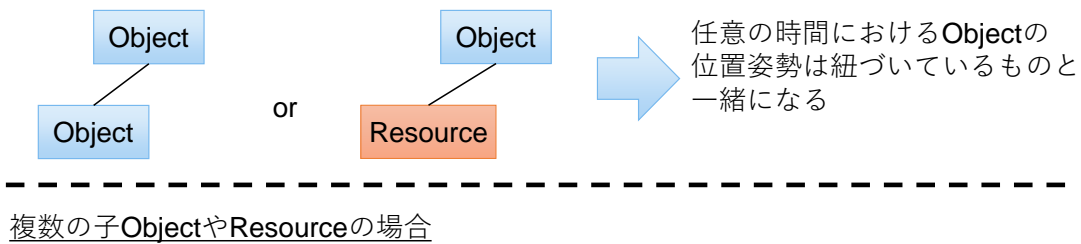
フロント」Object に紐づく Resource だけではなく、「リア」Object に紐づく Resource も取得できる。

Object は Resource を抽象化したものであるため、Resource は紐づいている Object の ID をメタ情報に有する。一方、Resource は複数のカメラが実世界を観測するごとに任意の Object に紐づくことから、Object と Resource は 1 対 n の関係である。そのため、Object は Resource の ID を持たない設計にした。

アプリが観測した Resource を紐づける Object はアプリ開発者が定義する。ただし、各アプリ開発者が定義した Object はユニークである必要がある。そこで、アプリ開発者は事前登録のリクエストを Object を管理しているサーバに送り、Object の ID を受け取る (図 3-3)。このとき、Object の木構造もリクエストするが、本フレームワークのコンセプトでは異なるアプリ間であってもデータを共有可能なため、木構造はアプリ内で閉じるわけではなく、他アプリで構成した木構造と連結することも可能である。つまり、登録済み Object の木構造が保持できれば、アプリ開発者は自由に木構造を拡張できる。

デジタルカメラで撮影した画像には GNSS で取得した位置、撮影日時や焦点距離などのメタデータが Exif (Exchangeable image file format) で埋め込んである。本フレームワークの Resource も同様に実世界を観測したときの情報であるカメラの内部パラメータや歪み係数などをメタデータとして含む。しかし、カメラの歪み係数は通常のレンズと魚眼レンズとでは、補正方法が異なる [19][20]。多くのアプリで共有する値は一意にする必要があるため、レンズなどの種類に対応した歪み補正の種類をメタデータに含める必要がある。アプリ開発時点で既知のものはアプリ開発者が、未知のものはアプリ利用者がセンサの種類の登録と同時にパラメー

### 1つの場合



### 複数の子ObjectやResourceの場合

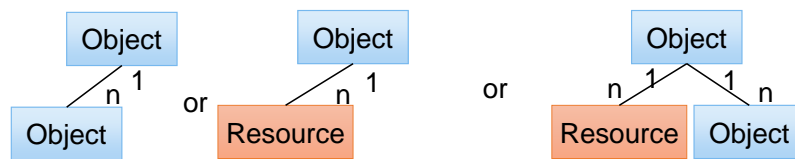


図 3-7 Object の位置姿勢の決め方

タのキャリブレーションを行う。

Object と Resource はともに ID による検索が可能であるが、さらに、本フレームワークでは 4 次元の情報を用いて検索ができるようにする。

Resource には画像そのものだけではなく、取得時のカメラの位置姿勢・時刻などのメタデータが必要である。一方、Object は抽象的な存在であることから位置姿勢・時間を定義しなければならない。任意の Object の位置姿勢・時間を定義するには、子 Object もしくは紐づいている Resource の位置姿勢・時間を使用する。図 3-7 に示すとおり、任意の Object の子 Object もしくは紐づいている Resource が 1 つの場合、それらの位置姿勢・時間を使用することが想定できる。一方で、本フレームワークにおいて Object には、複数の子 Object もしくは紐づいている Resource が存在する可能性が高い。

そこで複数の子 Object もしくは紐づいている Resource が存在する Object の位置姿勢・時間の決め方は 2 パターン想定した。1 つ目はアプリ開発者が任意の値を与える方法である。例えば、あるビル内に複数の固定された監視カメラがあったとする。ビルを Object とした場合、位置姿勢・時間を、ビルの所在地や竣工日から与えることができる。ただし、アプリ開発者が与えた値は、他アプリでも共有可能な SIGMA 座標系に変換後、リソース DB に登録される。2 つ目は「子 Object」や「紐づいている Resource」から計算する方法である。「子 Object」や「紐づいている Resource」が 1 つの場合の計算は容易である。一方、複数の場合については、適した値を計算可能かどうかについて先に検討する必要がある。例えば、自動車に搭載した複数のセンサは、ドアとともに動く電子ミラー用カメラであっても自動車から離れていくことはない(図 3-8(a))。また、ビル内を自由に動き回る監視ロボット群はビル内から出ることはない(図 3-8(b))。これらのことから、複数の場合の「子 Object や紐づいている Resource」は法則性を持った群集とみなせるため、計算は複雑化するが、この法則性を基に Object に適した値を計算することができる。ただし、Object 自身に複数の位置姿勢・時間をメタ情報として有すること





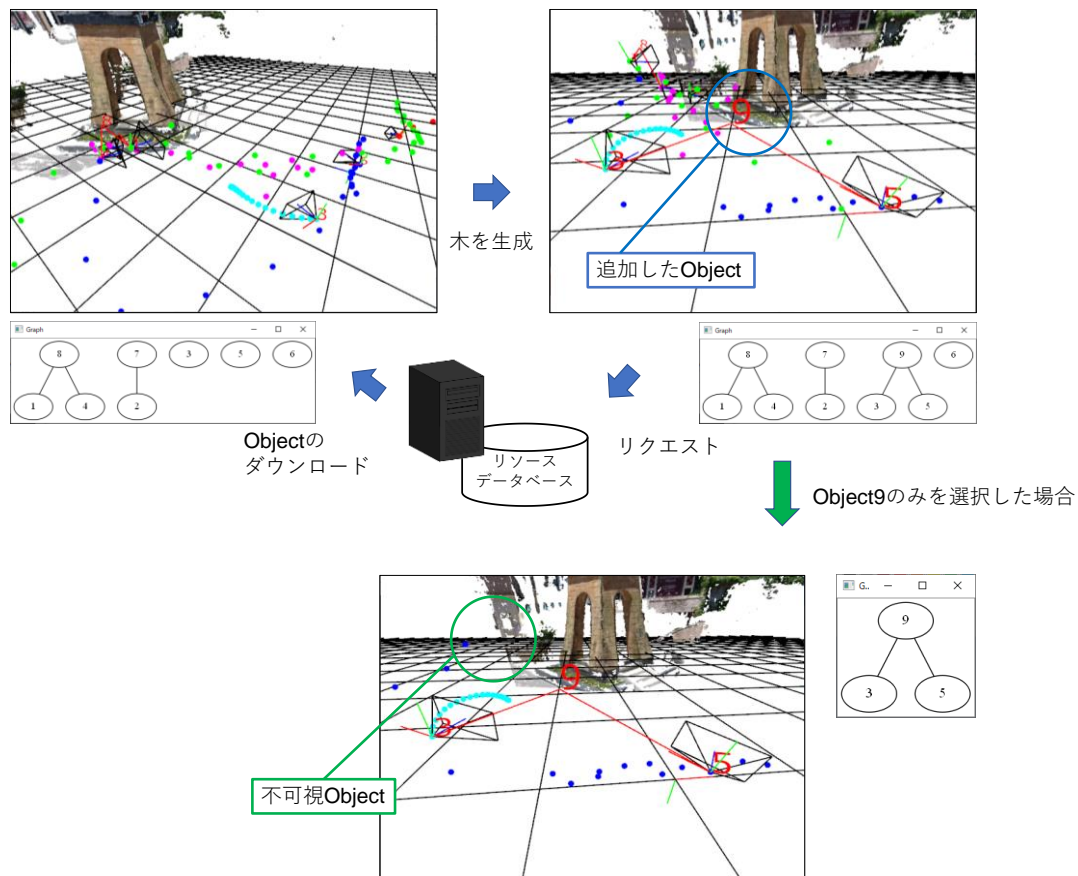


図 3-9 Object の設定方法

りが設定した Object を公開するのと同時に、他アプリの Object の設定も把握したい。そこで、**図 3-9** に示すとおり、本フレームワークに登録された Object や Resource を、サポートツールを用いて可視化できるようにした。このサポートツールは飛行機の位置をリアルタイムで表示する Flightrader24 のように、Resource や Object がどのように動いているのかも確認できるものを目指している。そのためには、Resource や Object だけではなく、観測した実世界との関係も把握できる必要がある。そこで、Resource と、位置姿勢の情報から生成した点群を表示することで、実世界との関係も把握可能になる。

さらに Object の構造が把握しやすいように木構造の可視化も行う。また、任意の Object を選択することで、その Object をルートとする部分木のみを可視化することも可能である。Object の構造が把握しやすいようにグラフの可視化も行う。

これらのことから、既存の Object を確認しながら Object の新規登録および、既存の Object に親 Object や子 Object の追加登録も可能である。さらに複数の木構造のルート Object を子 Object にする新規 Object も追加であることから、複数の木構造を 1 つにまとめることもできる。つまり、アプリ開発者全員で Object の木を育てていくイメージである。

### 3.4 リソースデータベースの設計

ここまで述べた設計を実現するために、リソース DB は関係型データベースを用いて 5 つのテーブルを用意した (図 3-10) .

ここまでの方針において必要なデータは、カメラの位置姿勢、撮影時刻、焦点距離、主点、解像度、歪み係数である。本研究では多機種のカメラを想定しているため、歪み係数を使用するアルゴリズムを明確にする。本研究では、時空間軸を表すカメラの位置姿勢、撮影時刻を外部パラメータ、焦点距離、主点、解像度、歪み係数を内部パラメータと定義し、これらのデータをメタデータとして、DB に画像とともに蓄積する。外部パラメータは **Geometry** テーブルに、内部パラメータや、画像が保存されている補助記憶装置へのファイルパス (**Resource** のパス) は **Resource** テーブルとした。このとき保存される画像は利用方法が一意でないことから、png 形式などの可逆圧縮が望ましい。

実世界を観測した **Resource** は即時アップロードされるため、計算時間の都合上、アプリが位置姿勢の情報を付加してアップロードすることは難しい。しかし、アプリからのリクエストに応じて検索する場合、画像の位置姿勢の情報をを用いることが多い。これに対処するため、サーバで位置姿勢を付加する方針を立てた。アップロードした **Resource** を何の情報もなく位置姿勢を推定することは難しい。しかし、事前に **Resource** は紐づく **Object** が既知であるため、連続で実世界を観測したものであれば、前フレームの情報から位置姿勢を推定しやすくなる。しかし、1 ショットのみを観測のみの場合、位置姿勢を推定するのが難しくなる。そこで、GNSS の測位情報を使用できるようにリソース DB を拡張し、**Geometry** テーブルには 2 種類の位置姿勢を格納できるようにした。1 つ目は推定値であり、2 つ目は実測値である。前者は SfM (Structure from Motion) などの各 **Resource** の位置姿勢を推定できる手法で計算した値を想定している。後者は GNSS の測位情報のように物理センサの値を想定した。推定値を付加する前にアップロー

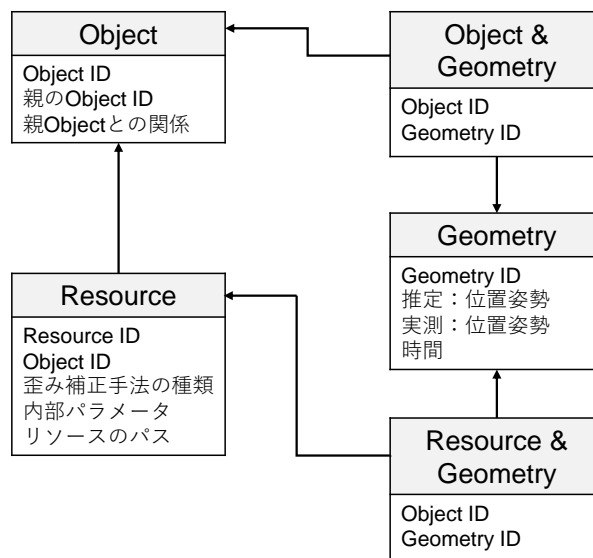


図 3-10 テーブルの関係

ドされた場合は、リソース DB の推定値が空欄となる。Resource を位置姿勢で検索する場合、推定値を基に実施するが、推定値が空欄時は実測値を検索対象にする必要がある。少しでも検索にかかる処理を減らすためにも、初期値として実測値を入れておくことで推定値が空欄かどうかの判断が不要となる。その結果、デフォルトの検索対象が推定値のみに絞ることができる。ただし、推定値はあくまでも推定値のため、誤った計算結果が入っている可能性がある。そこで、手の加えられていない実測値が必要になる可能性を考慮して、2種類の位置姿勢を保存する。また、より高精度な位置姿勢の結果を得られた場合は、サーバでの処理に限り推定値の更新を許可する。クライアントアプリ（以後、クライアントサーバシステムにおけるアプリをクライアントアプリと称する）での処理結果による更新まで許可すると、クライアントアプリ独自で計算した結果による更新リクエストになるため、精度の保証や、更新の制御などが難しくなるためである。つまり、サーバではリソース DB の情報を加工する機構が必要となる。次章では、サーバの設計について述べる。

また、Geometry の ID は、ユニークなものにするとともに、検索を高速化するため、近傍探索可能な ID を生成する。具体的には、同じ位置・時間に異なるカメラが存在することはあり得ないため、ID は位置・時間を基に生成したバイナリーコードを Base32 を用いて文字にした。図 3-11 に示すとおり、位置は GeoHash を参考に MGRS の各グリッドを南北/東西を交互に分割した 2 分木を基にバイナリーコードを生成することで検索スピードを向上させた[21][22]。このバイナリーコードを空間インデックスと称し、空間インデックスを用いた検索の高速化処理もサーバで実施する機構とする方針である。

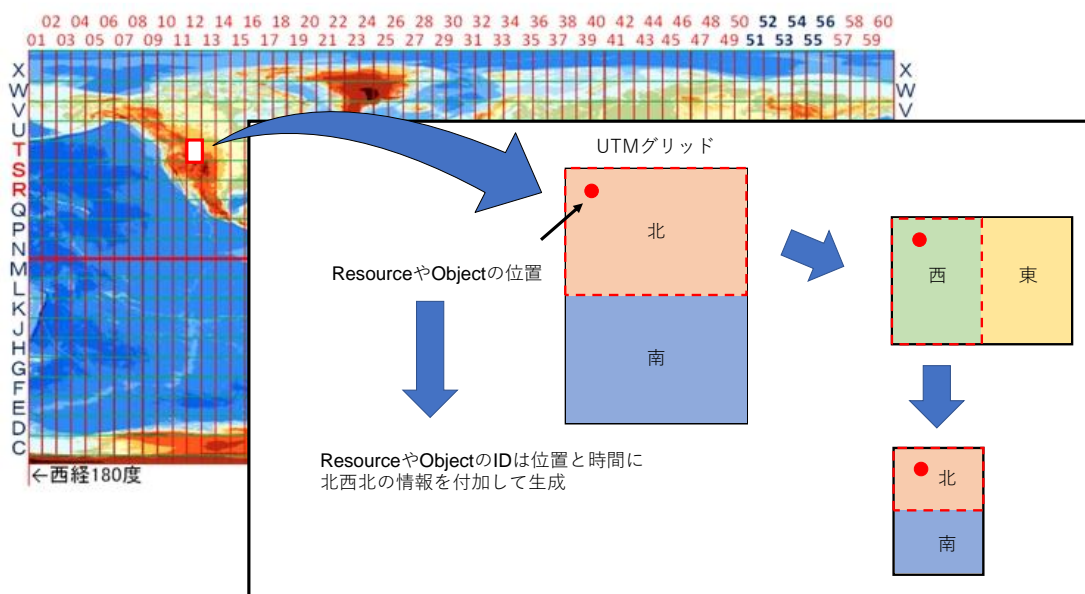


図 3-11 グリッドの分割

## 第4章 サーバの設計

### 4.1 設計方針

本章では、3章で述べた時空間映像データを共有するためのサーバを設計する。本フレームワークでは多くのアプリがアクセスすることから、次の経路が考えられる。

1. クライアントアプリからサーバへのアクセス
  - 1.1 Resource のリクエスト
  - 1.2 Resource のアップロード
2. サーバからクライアントへのリクエストに応じたレスポンス

経路 1.1 はリクエストであるため欠損することなくサーバに届く必要がある。そのため、プロトコルは TCP などのコネクション型通信を想定する。経路 1.2 は、大量のデータが連続でアップロードされることからサーバに蓄積するまでの時間を考慮し、UDP などのコネクションレス型通信を想定する。経路 2 はリクエストに応じて変化する。例えば、任意の時間の画像をダウンロードして、3D モデルを作成したい場合は、ダウンロード時間の制約が少ない。一方、リアルタイムに監視カメラ映像を接合する場合は、ストリーミングする必要がある。そこで、Web3 層構成のアーキテクチャをベースに、ストリーミングサーバを追加したアーキテクチャとする (図 4-1)。

Web3 層構成とは、Web アプリケーションサーバの構成要素のひとつであり、大量のリクエス

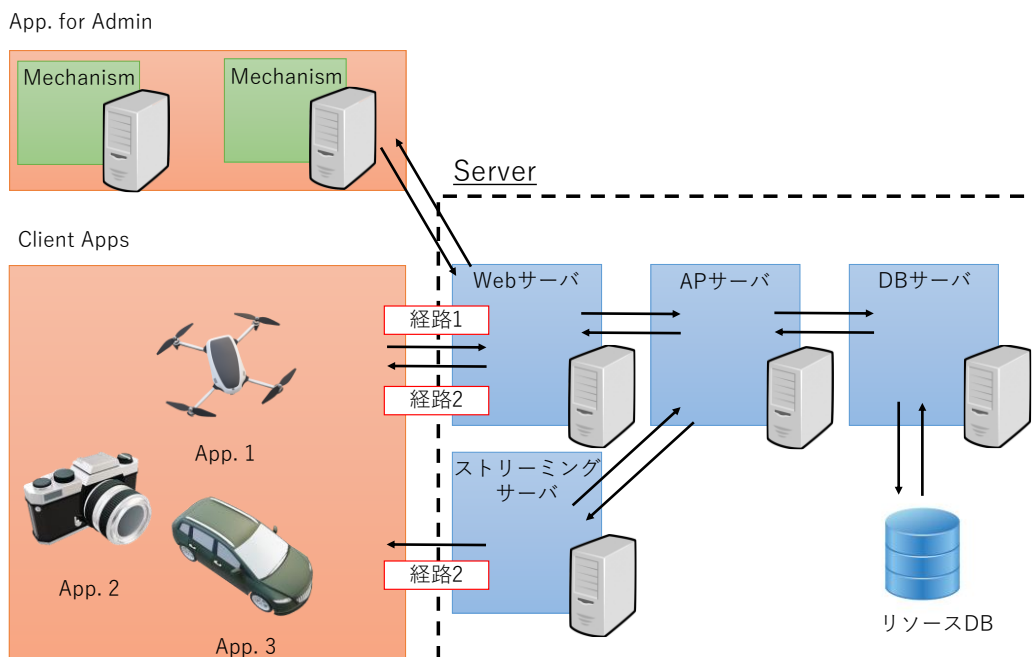


図 4-1 サーバのアーキテクチャ

トを高速で処理するために構成する。図 4-1 に示すとおり、Web サーバ、アプリケーションサーバ (AP サーバ)、データベースサーバ (DB サーバ) の 3 つからなる。

Web サーバはフロントエンドにあり、ユーザとのインタフェースの役割となるサーバである。つまり、このサーバがアプリからのリクエストを受け取り、そのリクエストを AP サーバに送る。また処理された結果を AP サーバから受け取り、クライアントに返す。AP サーバはクライアントからのリクエストに応じて処理するサーバであり、必要なデータは DB サーバにアクセスすることで取得する。DB サーバは、AP サーバの要求に応じて、DB サーバを制御するサーバである。この Web3 層構成に、経路 1.2 や経路 2 を満たすため、ストリーミングサーバを拡張した。

## 4.2 AP サーバと管理者用アプリの定義

本フレームワークにおける AP サーバには様々な機能を想定している。一般的に AP サーバは、クライアントからのリクエストに応じて処理する。例えば、アプリからアップロードする画像などの Resource は一般的に圧縮してから送信する。ダウンロードも同様である。そこで、クライアントアプリからのリクエストに応じて圧縮/展開する機構を AP サーバに持たせる。さらに本フレームワークでは、3.4 節で述べたとおり、リソース DB の Geometry テーブルに保存する位置姿勢の推定値を更新する機構も必要となるが、これはクライアントアプリからのリクエストではなく、フレームワークの機能である。そこで、管理者用アプリからリクエストすることで位置姿勢の推定値を更新する仕様にした。こうすることで更新タイミングなどのパラメータ調整をサーバ内のソフトウェアを書き換えることなく、変更することも可能となる。

AP サーバでは、クライアントアプリと管理者用アプリ、両方からのリクエストに基づいた処理を実行するため、図 4-2 に示すとおり、バッチファイルなどを用いてコマンドライン引数を入力した実行ファイルを実行する。したがって、バッチファイルと実行ファイルを AP サーバ内に増やすだけで様々な機構を追加できる仕様である。

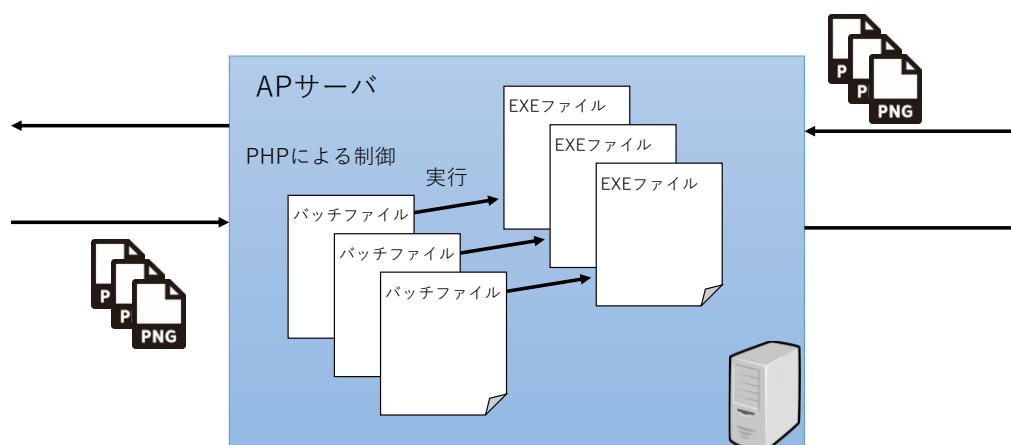


図 4-2 AP サーバでの処理イメージ

## 4.3 管理者用アプリ

本節では開発した管理者用アプリについて述べる。

### 4.3.1 ROI を用いた圧縮機構

本項では、サーバでの圧縮機構について述べる。アプリからのリクエストに応じて検索した画像を送信前に圧縮することは一般的な処理であるが、本フレームワークでは空間軸に遍在する画像であるため、一般的な画像圧縮より効果的な圧縮機構を検討した。検討した結果、ROI (Region of Interest)に着目した[23]。例えば 3D モデルを生成するために複数枚の画像をダウンロードする場合、画像において 3D モデルを生成する対象が写っている領域以外は重要でない場合がある。そのため、図 4-3 に示すとおり、空間軸上で Bounding Box を指定し、この Bounding Box を画像に投影することで ROI を指定した。こうすることで、ROI と ROI 以外の領域において圧縮方式を変更することができるため、効率よく画像群のダウンロードが可能となる[24]。本フレームワークでは WebP を用いて圧縮率を可変にしてある[25]。

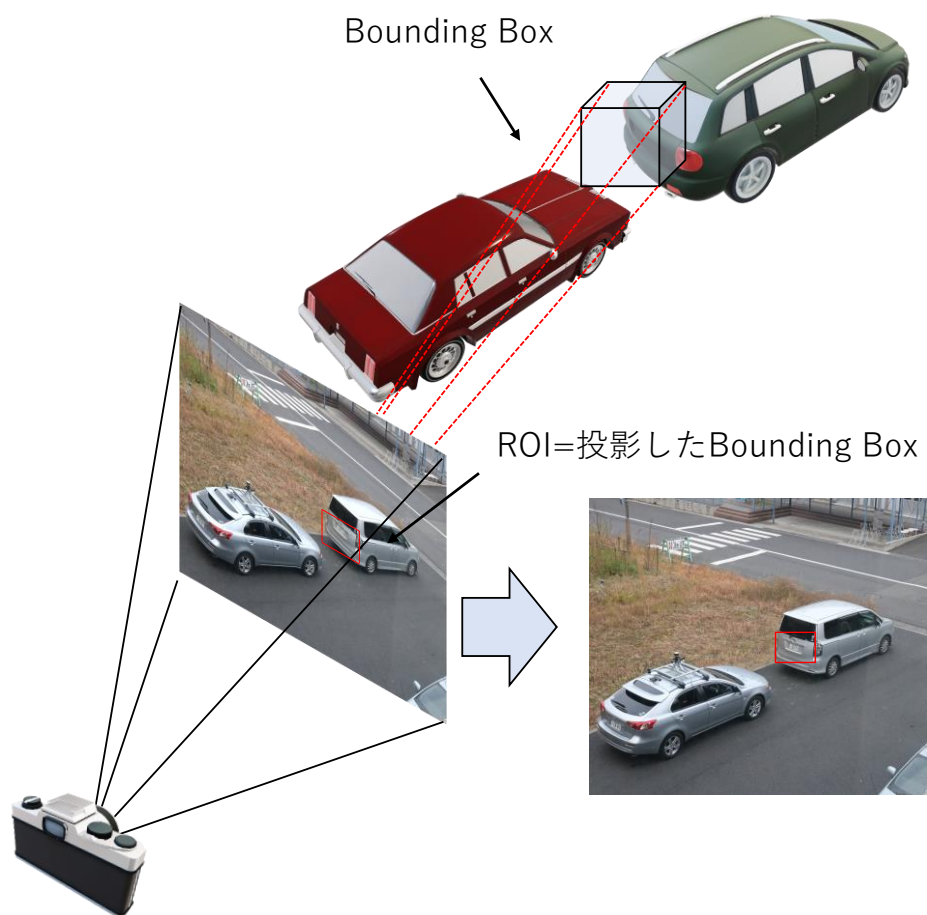


図 4-3 空間軸に指定した Bounding Box と ROI

#### 4.3.2 位置姿勢推定機構

本項では位置姿勢推定機構について述べる。本フレームワークではリソース DB の Geometry テーブルに保存する位置姿勢の推定値を更新するために位置姿勢推定機構を設計した。

本機構では、SfM を用いて各 Resource の位置姿勢を推定する[26]。SfM とは、カメラモーションから撮影対象を再構成する技術である。一般的に、多視点画像を用いて、カメラの内部パラメータ、外部パラメータと撮影対象を表す点群の座標値を推定することで、撮影対象を再構成する。代表的なパイプラインを列挙する。

1. 任意の 2 画像間の特徴点を検出，対応付け
2. 任意の 2 画像間の対応付けされた特徴点から，8 点アルゴリズムを用いて基礎行列を計算
3. 計算した基礎行列と各カメラの内部パラメータから 2 画像間のカメラの外部パラメータを推定
4. 2 画像の対応付けされた特徴点を各カメラの内部パラメータと外部パラメータを用いて三角測量により 3 次元座標値を計算（この 3 次元座標値の集合が点群）
5. 推定した点群を内部パラメータと外部パラメータにより画像に再投影し，検出した特徴点とのユークリッド距離（再投影誤差）が 0 に近づくようにパラメータを最適化（バンドル調整[27]）
6. バンドル調整後，3 枚目の画像から特徴点を検出し，既存画像の特徴点と対応付け
7. 既存の画像の特徴点に対応した点群と，3 枚目の画像の特徴点を対応付け（図 4-4）
8. 点群と特徴点から PnP 問題を解いて外部パラメータを推定
9. 4, 5 を実施
10. バンドル調整後，4 枚目を追加し，6 から 9 を実施

このように，はじめに 2 枚の画像を選び，パラメータを推定しながら追加していくことから，Incremental SfM とも言われている。処理 3 において外部パラメータを推定する前に，基本行列を求めるが，基本行列を求めるためには内部パラメータが必要である，しかし，SfM に入力する画像の内部パラメータは校正済みでない可能性が高いため，Exif から焦点距離などの情報を取得し，一時的な内部パラメータを設定する。一時的な内部パラメータであるため，処理 5 においてバンドル調整の対象となるが，校正済みの場合，内部パラメータを固定してバンドル調整する。バンドル調整では，パラメータ数が少ないほうが収束しやすい。文献[26]では，上記における 1 枚追加するごとに実施するバンドル調整をローカルバンドル調整，ある程度の画像が追加後に実施するバンドル調整をグローバルバンドル調整を称している。また，SfM で求めた点群は対応付けできた特徴点であるため，カメラ画像の解像度と比較して，数が少ない。そのため，疎な点群と言われており，三次元再構成するには，後段の処理として，多視点ステレオ (Multi-view stereo; MVS)を用いる[28]。Agisoft の Metashape や Pix4D の Pix4Dmapper などのソフトウェアでは多視点画像を入力すると，SfM から MVS までを利用者が意識することなく処理するなど，SfM と MVS の境界は曖昧である。しかし本機構の目的はカメラの位置姿勢を推定す



ることであるため、本機構では MVS を含めない方針である。

SfM において処理時間のボトルネックとなる処理はバンドル調整である。そのため、Incremental SfM はバンドル調整を複数回実施することから、処理に時間がかかる。そこで、Incremental SfM に対してカメラの位置姿勢・点群を全て求めてからバンドル調整するため、Incremental SfM よりも高速な Global SfM もあるが[29]、大量に Resource がアップロードしてくる本フレームワークには適さないため、本フレームワークでは Incremental SfM を実装する。本フレームワークでは Resource とともにメタ情報として内部パラメータを保持しているため、内部パラメータは既知であるため、バンドル調整のパラメータを減らすことが可能である。さらに、SfM では処理 2 や処理 3 で述べたとおり、初期抽出の 2 枚の画像や、画像追加時に特徴点を対応付けする画像を選択することが難しい。1 章で述べた SfM が対象にしたのは web にアップロードされた内部パラメータや外部パラメータが未知の画像であったため、td-idf 法によるスコアリングを用いて対応付けする画像を選択した。他にも画像類似度から Vocabulary Tree を構築する方法を提案するなど、様々な工夫がなれている。しかし、本フレームワークにおける Resource は GNSS の測位情報などの測定値を所持しているため、近くに最適なペアを探すため Resource の範囲を絞ることができる。もしくは、紐づいている Object から撮影時刻が近い Resource は撮影風景が類似していると想定できるため、最適なペアの探索が容易である。さらに Object から任意の画像の関係が、ステレオカメラの左右のカメラ画像の関係であるなどもわかる。このように複数のカメラの相対的な位置姿勢が既知である場合、バンドル調整する外部パラメータの数を削減することが可能である。

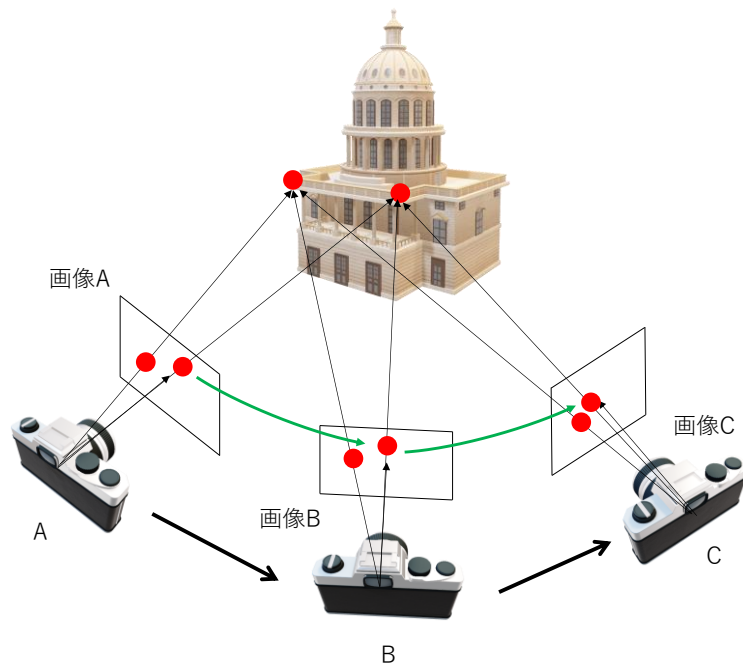


図 4-4 SfM の処理イメージ

一方、本フレームワークでは大量の **Resource** がアップロードされ続けることから、上記で示したように 1 枚の画像を追加するごとにバンドル調整をすることは処理に時間がかかるため、キューが増え続ける。そこで、ローカルバンドル調整は **Resource** が複数枚追加されてから実施し、ローカルバンドル調整後にリソース DB の推定値を更新する。さらに、グローバルバンドル調整後に、リソース DB の推定値を再更新する。

## 第5章 SIGMA Retriever の設計

本章では、4章で述べた管理者用アプリでもあり、本フレームワークで時空間映像データを共有するための4つのフェーズのうちのひとつでもある検索について述べる。

### 5.1 イメージベースドモデリング・レンダリング概観

1章にて本フレームワークが想定する利用方法は IBMR に焦点を絞ることを述べたが、一般に、IBMR を利用する場面は千差万別であり、ひとつの検索手法のみですべての利用目的に合致した画像を検索することは難しい。例えば、文献[3]の Agarwal らは写真共有サイト Flickr で “Roma” を検索キーとして抽出した画像を用いてローマの街並みの 3D モデルを作成した。Flickr は 3D モデルを生成するために集めた画像がアップロードされているわけではないため、抽出した画像には 3D モデルの生成には不適切な画像も多く含む。そこで、Agarwal らは SfM の処理を改善し、3D モデルの生成に適した画像のみを抽出している。このように、利用目的が一意であれば、それに対応する検索手法も一意に決めることができる。しかし、本研究のように 2 つの抽象的な条件「多種多様なカメラが遍在」「IBMR で利用」のみを制約とした場合、様々な利用目的に対して網羅的に検索できるような機能を準備することは困難である。したがって、少ない制約条件下で、広範な用途で有用な手法を提案するためには、利用目的や条件を抽象化する必要がある。そこで、本研究では、利用目的である IBMR について概観し、処理の流れを抽象化した上で、検索手法を体系的にまとめるとともに、検索キーの保存方法について検討することとした。こうして開発した検索機構を “SIGMA Retriever” と称する。

IBMR について概観する前に、IBMR にはどのようなものがあるのかを列挙する。SfM や v-SLAM (Visual Simultaneous Localization and Mapping)[30]では現実空間にある物体から点群で表す 3D モデルを生成するだけでなく、各画像の位置姿勢を推定できる。3D モデルの生成例は机上の模型から、文献[3]で例示されているようなローマの街並み、さらには航空機から地上に向けて撮影した画像を用いた数値表層モデルに至るまで[31]、様々な生成対象の大きさや、対象物までの距離が想定される。

映画『マトリックス』で有名になった Bullet time では、被写体を囲むようにカメラを配置し、仮想的に動かしたいカメラの位置にある映像を切り替えていくことで、被写体はスローモーションになるが、カメラは高速に動いているような映像を生成している。日産自動車のアラウンドビュー[32]では俯瞰映像を、Google Street View では広視野な画像を多視点カメラの映像から生成している。これらは仮想投影面上で、複数視点の画像を縫い合わせるように接合するため、Photo Stitching と呼ばれている。この手法では各カメラの位置姿勢から任意の面に映像を投影し、映像の重複領域をブレンド処理することで、現実には存在しない映像を生成している[33]。Light Field Rendering では、3D モデルを生成せずに、多視点画像の光線を DB に蓄積することで、仮想カメラの光線を補間し、自由視点映像を生成している[34]。

IBMR では、カメラ位置だけではなく、撮影時刻について考慮しているような場合もある。例

例えば、映像の中から任意の物体を視覚的に消去する DR では、他視点の映像を用いて、消去対象の背景映像を生成し、この背景映像を物体に重畳するが[35]、隠背景に写っているものが非静止物の場合、撮影時刻が異なる画像から生成した隠背景では不都合が生じる場合がある。また、撮影時刻の異なる画像群から街並みの時間的な変化を推定して 3D モデル化する 4D Cities Project もある[36]。

ここで述べたように、一口に IBMR といっても、その目的や用途は大きく異なっており、そこで必要とされる画像についても多様であることがわかる。

## 5.2 処理の流れの抽象化

5.1 節で概観した様々な手法をもとに IBMR の処理の流れを抽象化すると、図 5-1 に示す 4 つのステップ「撮影」「メタデータの推定」「実世界の計測」「描画」に分けられる。「撮影」とは、カメラで実世界を撮影することである。このとき、撮影画像には GNSS で取得したカメラの位置、撮影日時や焦点距離など、撮影時のカメラの状態を表すメタデータである Exif タグが付与される場合がある。

「メタデータの推定」とは、カメラの位置姿勢や内部パラメータなどを推定することである。本研究では、チェスボードなどによって「撮影」前にキャリブレーションしたものや、Exif タグのようにセンサが付加したものを計測値と定義し、このステップでは、画像や計測値を初期値として使用することで、高精度なメタデータを推定する。撮影時のメタデータが高精度であれば、推定値を計算しないこともある。例えば、SfM において、カメラキャリブレーション済みのカメラを用いる場合は、内部パラメータは計算対象にならない。

「実世界の計測」とは、カメラ位置と画像を基に実世界を 3 次元的に計測することである。SfM や v-SLAM では、計測結果を点群で表すが、この点群が SIFT などの特徴点に基づく場合、疎な点群と呼ばれる[37]。より正確な実世界の情報である密な点群やメッシュを生成するためには、

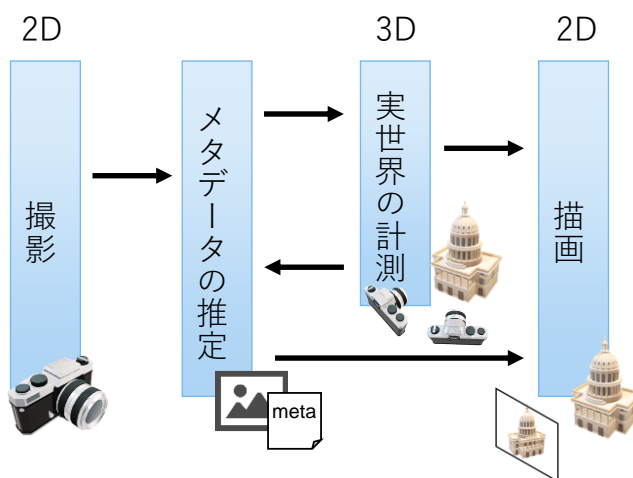


図 5-1 抽象化した IBMR の処理の流れ

Patch Match Stereo[38]など、目的に合わせた MVS が必要となる。さらに、SfM 中の一処理であるバンドル調整では、カメラの焦点距離や位置姿勢とともに点群の座標値を最適化するため、「メタデータの推定」と「実世界の計測」は両方向の矢印としている。

「描画」では、利用者に提示するため、計測した実世界を描画面に投影することである。ただし、Photo Stitching などは、実世界の計測自体はしていないため、「メタデータの推定」から直接「描画」に進む矢印も追加した。

### 5.3 設計方針

5.2 節で整理した処理の流れから、IBMR には、「メタデータの推定」「実世界の計測」「描画」の 3 つのタイミングで画像の入力が必要となる。これらの画像を多くの画像の中から抽出するためには検索条件が必要となるが、画像そのものを検索キーとして DB から検索することはできないため、画像を撮影したカメラのメタデータはすでに存在することが前提となる。

「実世界の計測」では計測対象を撮影している画像が抽出対象であり、「描画」では描画対象を撮影している画像が抽出対象である。「実世界の計測」は対象の実世界（被写体）が写っているかを画像中から認識する必要がある。認識する被写体が事前に決まっていれば、アップロードされた画像に検出した物体をラベル付けし、メタデータに含むことができるかもしれない。しかし、広範な用途が想定される本研究の検索において、検索に合わせたラベル付けを設計することは難しい。そこで、実世界において、対象となる領域を指定し、その対象が撮影されている可能性があるかを、カメラの位置姿勢や画角などを基に計算することで判断する。

大量に蓄積した全画像に対して検索すると、たとえば、神奈川県鎌倉駅前を対象領域に指定した場合、どこで撮影した画像であったとしても、鎌倉駅前の方向を向いていれば、抽出される可能性があるため、全画像を対象にすることは現実的ではない（図 5-2）。したがって、検索対象となる画像は空間上で制限を設ける必要がある。

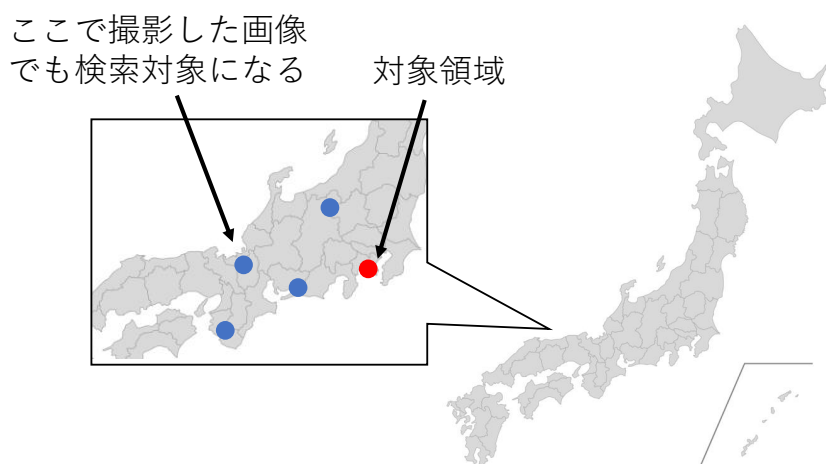


図 5-2 全画像を検索対象にした例

本研究が想定しているカメラは街中に遍在しているだけでなく、時間とともに移動する場合がある。したがって、画像を検索する際には、検索対象の画像が空間軸上だけではなく、時間軸上にも撮影時のカメラが遍在することを考慮する必要がある。手順としては、時空間軸に基づく絞り込みを行った上で、カメラ画角などの内部パラメータを用いて検索する。「描画」も同様に、描画対象が写っているかを内部パラメータを利用して検索する。「メタデータの推定」は「実世界の計測」「描画」で使用する画像のメタデータを抽出することが必要条件であるが、十分な条件は一意に決められない。そのため、時空間軸を利用して検索するしかない。

## 5.4 検索手法の提案

DBに蓄積した画像を抽出するためには、画像そのものを **Object** や **Resource** の ID などで指定か、画像のメタデータを基に検索する手法がある。街中に遍在するカメラの画像を検索するには、外部パラメータをキーとして検索するのが一般的な手法である。しかし、外部パラメータを利用して検索できるのは、カメラの位置や撮影時刻であり、この手法で検索した画像中に任意の被写体が撮影しているとは限らない。たとえば、画像中の被写体は、カメラの位置から数  $m$  離れた位置に存在することが一般的である。5.3 節で述べたとおり、外部パラメータで検索対象を絞った上で、任意の領域を撮影しているかを、画角や焦点距離などの内部パラメータに基づいて検索する。

任意の領域とは「実世界の計測」「描画」から、「空間で指定できる領域」「任意の視点から見た領域」の 2 つから定義できる。これらの領域を検索するために、一般的な検索手法に限定せず、目的に合致する手法を選択することが重要である。「空間で指定できる領域」は一般的に、空間軸で多面体を指定できるため、その多面体が各画像を撮影したカメラで撮影できるかを判定するには、視錐台カリングを用いる手法を提案する（「視錐台カリングに基づく検索」）。しかし、各画像に対して視錐台カリングを用いると計算量が大きい。また、指定した多面体が遠くにある場合は、計算量の削減するため、多面体を点で代替する「注視点に基づく検索」を提案する。「任意の視点から見た領域」は仮想カメラと検索対象のカメラとのエビポラ幾何に基づいて検索する。

本節では、図 5-3 に示すとおり、「外部パラメータをキーとした検索手法」と「内部パラメータをキーとした検索手法（視錐台カリングに基づく検索、注視点に基づく検索、エビポラ幾何に基づく検索）」について述べる。

### 5.4.1 内部パラメータをキーとした検索手法

この検索手法でキーとする外部パラメータは 3.2 節で定義した時空間軸である。具体的な手法としては、各座標軸における任意の領域を指定し検索する。領域は、各座標軸の中心座標値  $(x, y, z, t)$  と範囲  $(a, b, c, d)$  を指定する。その検索方法の空間軸は立方体となるため、範囲は立方体の 1 辺の長さとなる。また、中心座標値からの距離に応じた領域も指定可能とし、検索値をそのままに次の式 1 が成り立つ検索方法も提案する。式(1)の場合、空間軸は楕円球となる（図 5-4）。

$$\frac{x^2}{\left(\frac{a}{2}\right)^2} + \frac{y^2}{\left(\frac{b}{2}\right)^2} + \frac{z^2}{\left(\frac{c}{2}\right)^2} \leq 1 \quad (1)$$

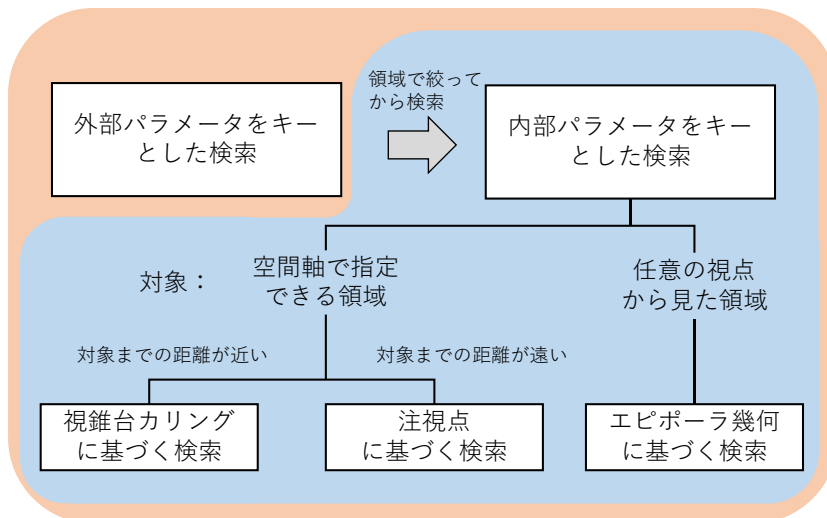
「距離」や「範囲」だけでは単純な形状でしか検索できなかつたが，実世界において単純な形状での検索で十分なケースは少ない．例えば，自動車が通過した経路を検索の対象にしたいとき，道路上で右折や左折を繰り返す自動車を 1 つの形状で表すことは難しい．このようにしてできた経路を検索の対象にするには「外部パラメータをキーとした検索」の「距離」や「範囲」を変更しながら，何回も検索する必要があるが，**図 5-5** に示すとおり，重複する領域もあり，検索処理が冗長的になる．こうした背景から「外部パラメータをキーとした検索」で指定できる領域は 1 つだけではなく，複数指定でき，それらは積集合や和集合として扱うことも可能である．

時間軸は空間軸とは異なり周期性があることから，直線的な側面と円環的な側面を合わせもつ．例えばシステム時刻のように，任意の時点からの経過時間で表すシステム時刻は直線的である．一方，「正午からの前後 1 時間のみ」や「毎月 15 日からの前後 3 日間のみ」は円環的な表現である．検索においても，円環の一部を指定したい場合もあるため，周期性を満たす検索値も使用可能とした (**図 5-6**)．

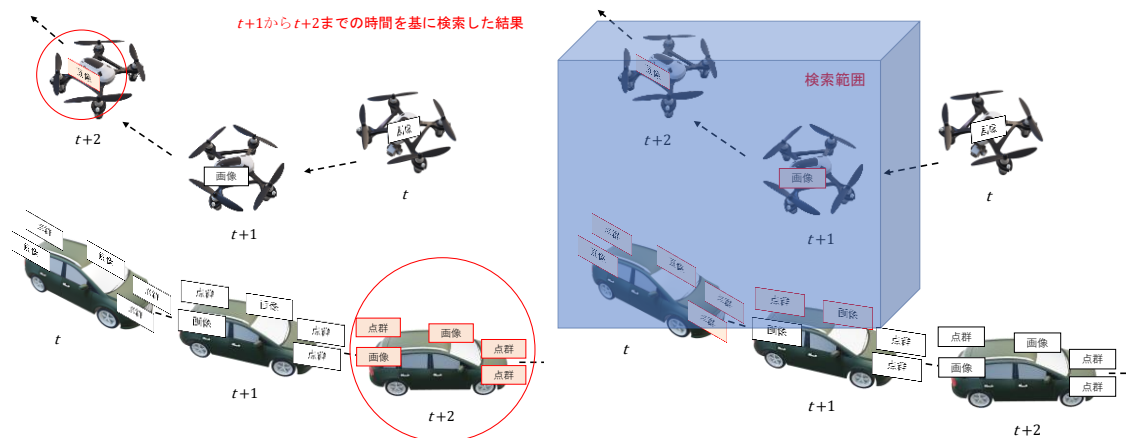
#### 5.4.2 外部パラメータをキーとした検索手法

##### 【視錐台カリングに基づく検索】

5.3 節で述べたとおり，本手法は視錐台カリングを用いる．視錐台カリングは，CGI (Computer Generated Imagery) 生成時に視錐台内に 3D モデルが含まれているかを判定することで，描画処理を減らす手法である．しかし，視錐台カリングは計算量が多いことから，多面体に軸平行境界ボックス (Axis-Aligned Bounding Box; AABB) を用いて計算量を削減することが多い．3D モデルを囲む AABB を指定し，視錐台の各面と AABB との関係を計算する[39]．**図 5-7 (a)** に示す

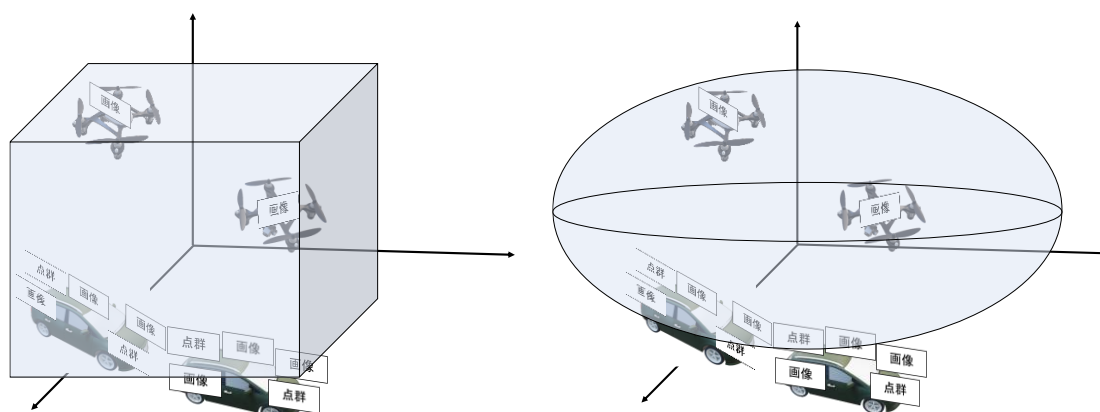


**図 5-3** 提案手法の分類



(a) 時間軸による検索

(b) 空間軸による検索



(c) 範囲による検索 (空間軸のイメージ)

(d) 距離による検索 (空間軸のイメージ)

図 5-4 時空間による検索

とおりに、AABB の各頂点から、視錐台の任意の面の法線ベクトルの向きに最も近い対角線を作成する 2 頂点を選ぶ (視錐台カリングにおいて、面の法線ベクトルは視錐台の内側を向いている方が正)。任意の面から選択した 2 頂点への法線ベクトルの大きさを計算し、大きい方を **p-vertex**、小さい方を **n-vertex** と称する。各面に対する **p-vertex** が全て正の場合は視錐台内に AABB が含まれていると判定できる。ただし、本検索の対象はカメラ画像であるため、CGI 描画におけるファークリップ面を設定することができない。同様にニアクリップ面はカメラの視点に近い。これらから、AABB の頂点と、視錐台のニアクリップ面やファークリップ面との関係は計算しないことで、計算量も減らすことができる。したがってカメラ 1 台に対する計算量  $s$  は、式(2)となる。式(2)における  $\alpha$  は対角線と面の法線とのなす角の計算、 $\beta$  は **p-vertex** から面までの法線の大きさの計算であり、面上の任意の点を  $Q$ 、**p-vertex** を  $P$ 、 $P$  から面への垂線を  $n$  としたとき、 $\beta$  は式(3)で求めることができる (図 5-7(b))。つまり、カメラ  $n$  台におけるオーダはカメラ台数に比例することから  $O(n)$  となる。



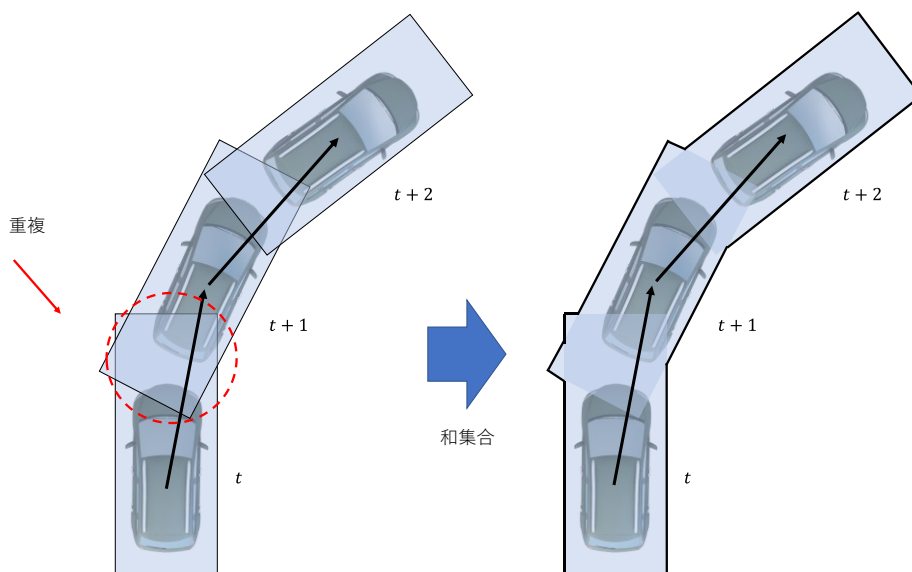


図 5-5 領域の和集合

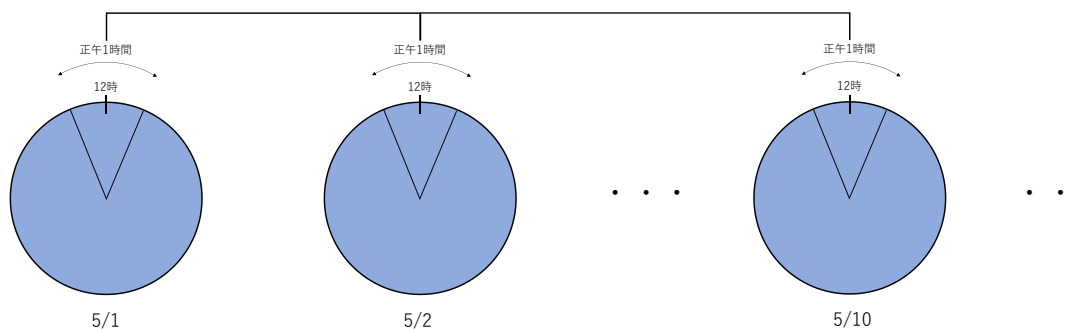


図 5-6 円環による検索

$$s = 4 \times (4\alpha + \beta) \quad (2)$$

$$\beta = \left| \overrightarrow{QP} \cdot \frac{\vec{m}}{\|\vec{m}\|} \right| \quad (3)$$

図 5-7(c)に示すとおり、カメラが 5 台 (カメラ A~E) あるときは、各カメラの視錐台と AABB との関係进行計算することで、カメラ C は抽出されないことがわかる。また、少しでも撮影画像内に AABB が含まれると検索条件に当てはまるため、図 5-7(c)に示すとおり、任意の物体を向いていないカメラ E の画像であっても抽出する。

「視錐台カリングに基づく検索」は、少しでも撮影画像内に AABB が含まれると検索条件に当てはまるため、建築物を向いていないカメラ E の画像であってもアプリに送信するカメラ画像に含まれるためデータ総量が増大する (図 5-7 (b))。そこで、SIGMA フレームワークでは、3次元空間上に立方体を指定することで、必要な情報はそのままに画像を圧縮する手法を 4.3.1 項で

提案した。この手法では、3次元空間に立方体などの3次元形状を指定し、それを各 Resource に投影することで ROI を指定する。本論文では、この手法の3次元形状に AABB を応用することで、各カメラ画像に ROI を別途指定する必要がなくなる。具体的には、カメラ画像に投影した AABB を内包し、カメラ画像座標系の軸に平行な外接矩形を ROI としている。このとき、カメラ画像座標系の軸に平行にするのは AABB 同様、計算量を減らす狙いがある。この ROI 以外の画質を低下させることで、効率よく画像を圧縮している (図 5-7(c))。

この ROI を指定する手法は、本提案手法との親和性が高く、立方体を AABB にすることで、DB サーバで検索してから AP サーバで画像を圧縮するまで効率よく処理することが可能となる。

さらに、遠方の物体を検索する場合、AABB が相対的に小さくなる (図 5-8)。また遠方であるために対象となるカメラの総数も多いことが想定される。そこで、AABB よりも簡易的な手法として注視点を利用した手法も提案する。

#### 【注視点に基づく検索】

空間軸上に注視点を指定し、各カメラが注視点を撮影可能範囲かを判断する。これは次の2条件で判断可能である。

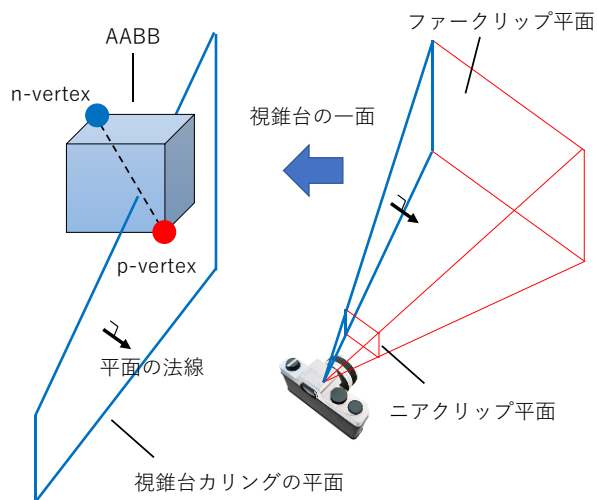
1. 注視点はカメラの光軸方向において正の値
2. 注視点をカメラ画像に投影可能

検索対象のカメラが位置  $t$ 、姿勢  $R$ 、焦点距離  $f$ 、主点  $(c_x, c_y)$  の場合、注視点の世界座標系の座標値  $(X_w, Y_w, Z_w)$  は式(4)を用いてカメラ座標系の座標値  $(X_c, Y_c, Z_c)$  に変換する。このとき  $Z_c$  が正の値であれば条件(1)を満たす。また、カメラ座標系の座標値  $(X_c, Y_c, Z_c)$  は式(5)を用いて、カメラ画像座標系の座標値  $(u, v)$  に透視投影する ( $Z_c = 1$  の面に投影されるように  $h$  を用いて正規化する)。この  $(u, v)$  が 0 以上かつカメラの解像度未満の値であれば条件(2)を満たす。図 5-9 に示すとおり、両条件を満たすのはカメラ C のみである。

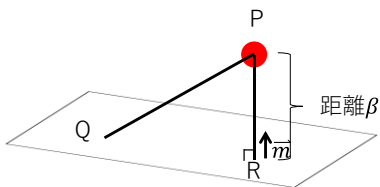
$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = (R_{3 \times 3} \quad t_{3 \times 1}) \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \quad (4)$$

$$h \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} \quad (5)$$

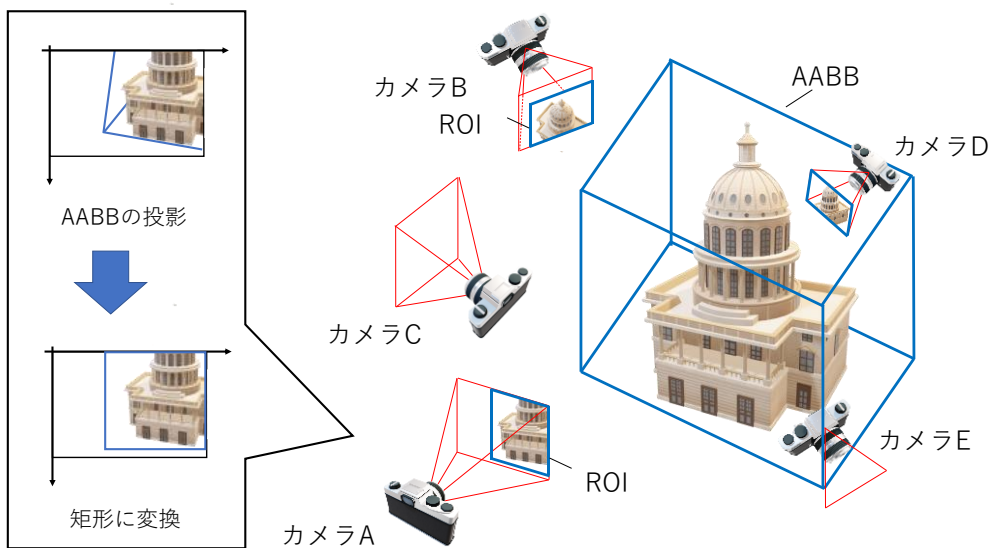
「注視点に基づく検索」も「視錐台カリングに基づく手法」同様、カメラ  $n$  台におけるオーダは  $O(n)$  であるが、式(4)は乗算が 12 回であり、式(5)は 0 を考慮すると乗算が 5 回、除算が 2 回であるため、カメラ 1 台に対して、「注視点に基づく手法」は乗算 17 回と除算 2 回で計算が可能である。式(2)では  $\alpha$  を 16 回、 $\beta$  を 4 回計算するため、「注視点に基づく検索」は「視錐台カリングに基づく手法」より計算量が少ないことがわかる。本論文では、「注視点に基づく検索」は「視錐台カリングに基づく検索」よりも指定領域が遠方のときに使用することを提案している



(a) 視錐台カリング



(b) 点と面までの距離



(c) 抽出されるカメラ

図 5-7 視錐台カリングに基づく検索

が、遠方に具体的な数値的基準はなく、指定した領域やカメラの位置、総数などの要因によって最適な検索手法が変化するため、利用目的に合わせて決める必要がある。

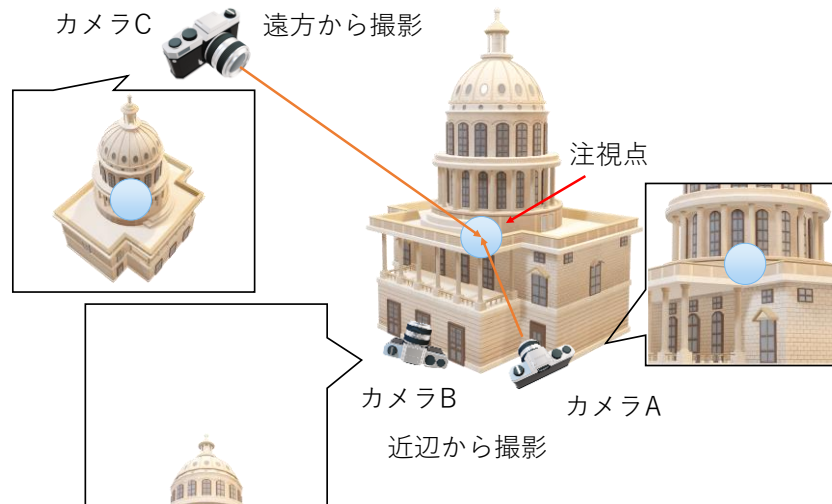


図 5-8 被観測物とカメラとの距離

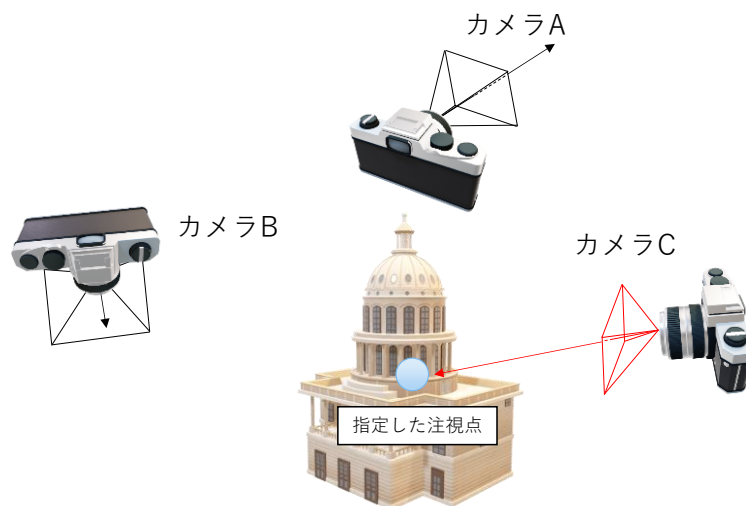


図 5-9 注視点に基づく検索

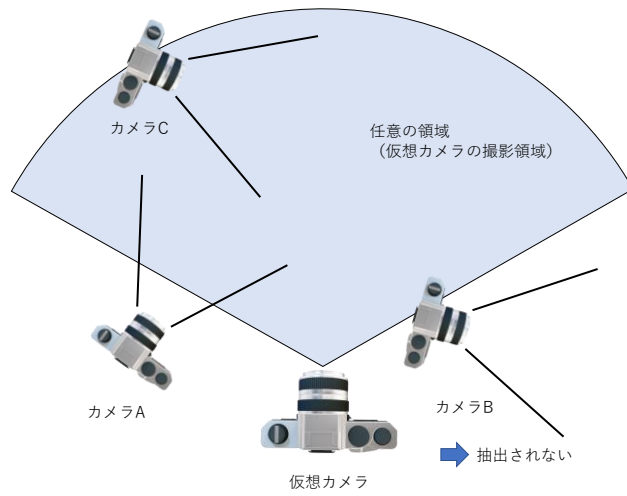
【エピポーラ幾何に基づく検索】

設定した仮想カメラの撮影領域を検索対象の各カメラが撮影しているかを調べる。図 5-10(a)に示すカメラ A, C のように、仮想カメラの撮影領域と検索対象の任意のカメラの撮影領域に共通部分がある場合は検索条件を満たす。この検索条件は 2 カメラの撮影領域における共通部分の有無と言い換えることができる。

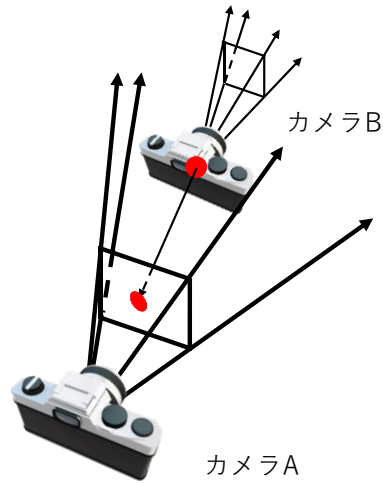
図 5-10(b)に示すとおり、お互いのカメラが撮影領域内にある場合は、2 カメラの撮影領域に共有部分があると言える。具体的にはどちらかのカメラの視点をもう一方のカメラの画像に投影できるか(エピポールの有無)を調べる。両方のカメラの視点が投影できるかではなく、どちらか一方でも条件を満たせば共有部分があると判断できる。図 5-10(a)ではカメラ C がこの条件を満たす。

しかし任意のカメラの撮影領域内に、もう一方のカメラがない場合もある。この場合は、共通部分の有無についてはエピポーラ線を用いて調べる。もし共通部分がある場合は、どちらか一方のカメラの画像の各画素から **Ray-casting** のように光線を飛ばし、もう一方の画像に、この光線を投影し、できた線（エピポーラ線）が 1 本でもあれば、共通部分があるといえる。しかし、全画像に対して処理するには、計算量が大きい。そこでカメラ画像の四隅の光線のみを飛ばし、エピポーラ線の有無によって判断する。しかし、四隅の光線だけでは図 5-10 (c) に示すとおり、エピポーラ線が生成できない場合がある。これは、画角の大きい方のカメラから小さい方のカメラに光線を飛ばしたことが原因であるため、画角の小さい方から光線を飛ばすことで判定が可能となる。図 5-10 (a) ではカメラ A がこの条件を満たす。

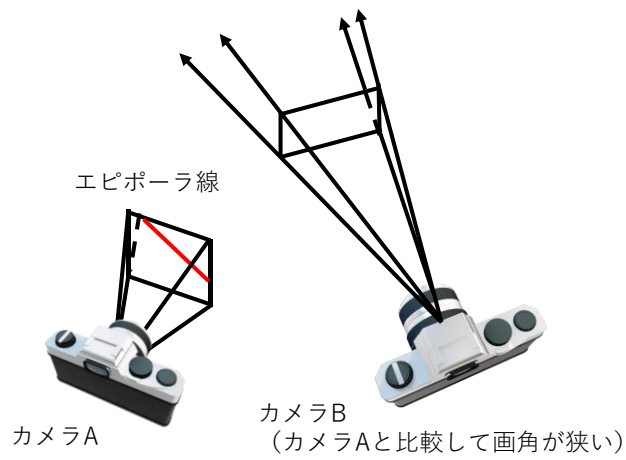
前述したとおり、「視点の投影」は計算量が少ないため、「視点の投影」を優先して用いる。「エピポーラ幾何に基づく検索手法」のフローチャートは図 5-11 に示す。



(a) 仮想カメラの撮影領域



(b) 視点の投影



(c) 光線の投影

図 5-10 エピソード幾何に基づく検索

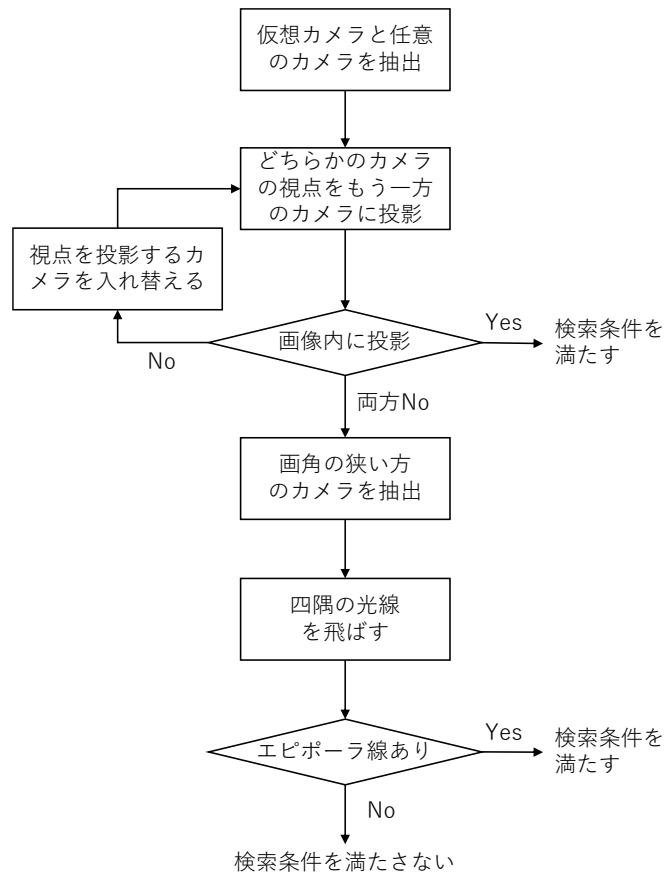


図 5-11 エピポーラ幾何に基づく検索のフローチャート

## 第6章 SIGMA フレームワークの試作と評価

### 6.1 評価環境の構築

本章では、提案した検索手法の動作確認および評価を実施した。6.4節では「外部パラメータをキーとした検索」の動作確認および、動作確認に対して考察した。6.5節では「内部パラメータをキーとした検索」を評価した。5.3節で述べたとおり、「内部パラメータをキーとした検索」は「外部パラメータをキーとした検索」に条件を追加して使用する。そのため、「外部パラメータをキーとした検索」を評価基準とし、これと比較することで、「内部パラメータをキーとした検索」によって生じた変化を考察した。評価1では「注視点に基づく検索」の比較評価し、評価2では「視錐台カリングに基づく検索」の比較評価、評価4では「エピポーラ幾何に基づく検索」

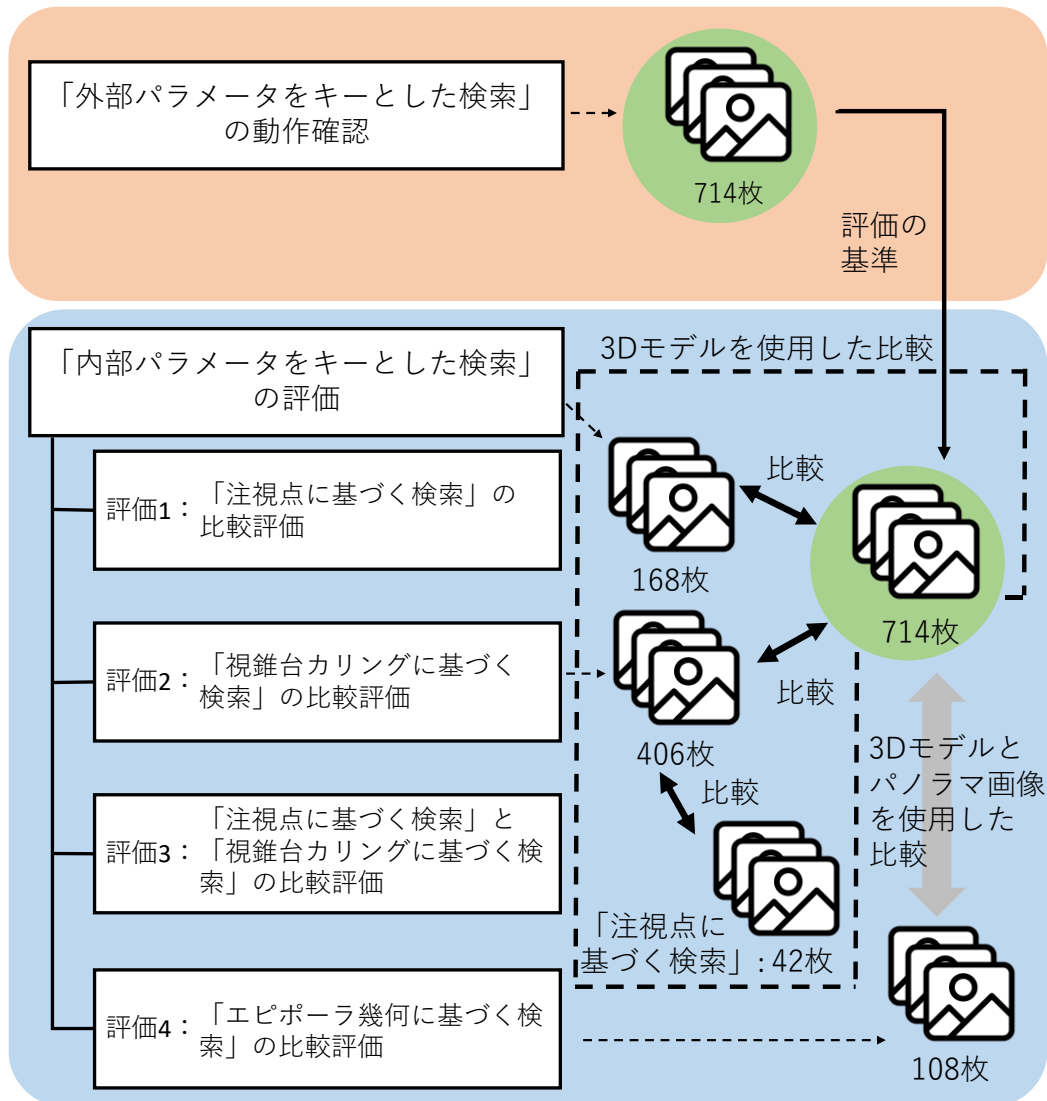


図 6-1 評価の種類



の比較評価した。また、評価3では、本評価で準備した環境において、「視錐台カリングに基づく検索」「注視点に基づく検索」を比較し、比較に対して考察した(図6-1)。

## 6.2 評価のための準備

3.4節で構築したリソースDBに画像を登録するため、鎌倉駅周辺を撮影した。使用したカメラは4台(カメラ1: Canon Kiss M (EF-M15-45 IS STM), カメラ2: SONY XPERIA X Compact メインカメラ, カメラ3: SONY Cyber-shot DSC-WX350, カメラ4: GoPro HERO3)であり、カメラ4は魚眼レンズである。カメラ1,2,3は1,920×1,080[px]で撮影し、カメラ4は2,704×1,520[px]で撮影した。また、撮影時期と撮影枚数は

表6-1に、各カメラで撮影した画像の一部を図6-2に示す。

リソースDBに計1,160枚の画像を登録した後、4.3.2項で述べたSfMを用いて画像の位置姿勢を推定し、リソースDBの位置姿勢の推定値を更新した。その結果、モーションブレンダーなどが

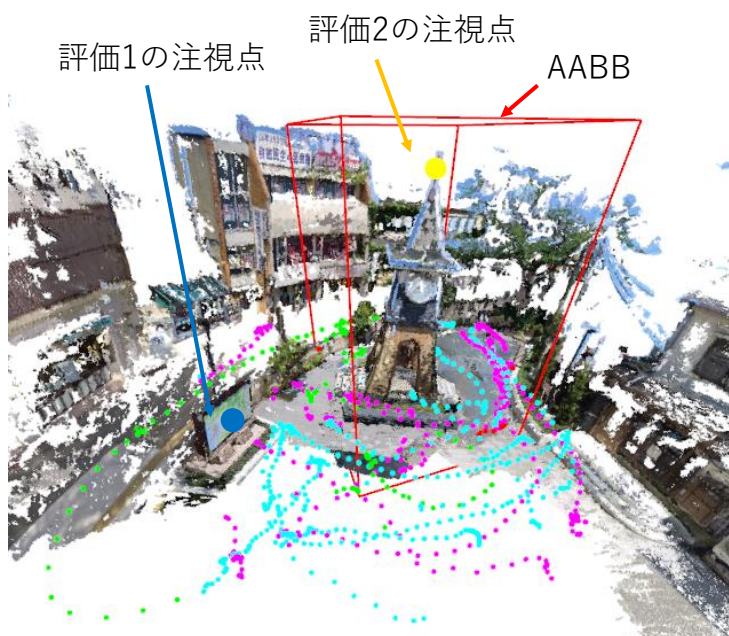
表 6-1 撮影時期と撮影枚数

	5月昼	12月昼	12月夕方
カメラ1(青)	227	30	114
カメラ2(緑)	324	163	55
カメラ3(紫)	-	117	92
カメラ4(赤)	-	12	26
計	551	322	287

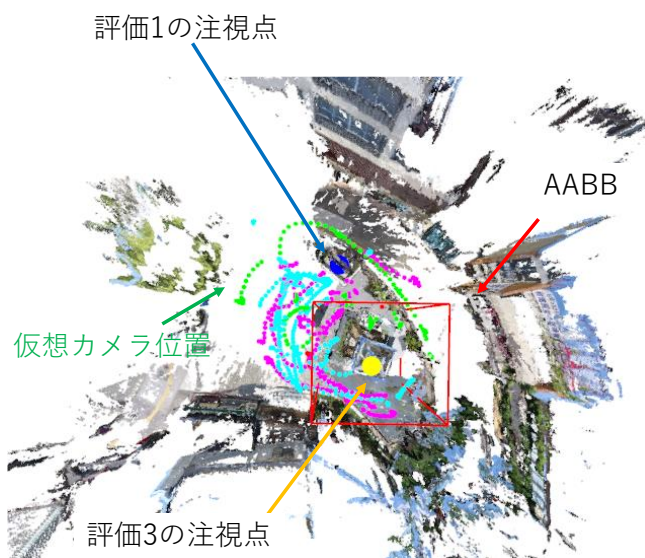


図 6-2 撮影した画像の一例

原因で推定できなかったものを除く，1,105枚の位置姿勢をリソースDBの推定値に蓄積した。これらの画像と推定した位置姿勢を基に鎌倉駅旧駅舎時計台および、その周辺を三次元再構成した結果を図6-3に示す。図6-3中にプロットした色は撮影位置を表しており、色は図6-2に示したカメラの色に合わせてある。評価1の注視点は時計台近くの案内図上に青丸で表し、評価3の視点は時計台頂上に黄色い丸で表し、評価4で指定した仮想カメラの位置を緑丸で表した。さらに、評価2,3で使用するAABBも併せて描画した。



(a) 近景



(b) 俯瞰

図6-3 時計台と周辺の三次元再構成（プロットした点はカメラの撮影位置を示す）

### 6.3 評価方法

比較方法は、5.1 節の例で挙げた 3D モデルの生成や、パノラマ画像の生成を用いる。

3D モデルは、提案する検索手法によって抽出した画像を SfM によりカメラの位置姿勢を計算し、疎な点群を構成した後、Patch Match Stereo をベースに密な点群を構成することで生成する。このとき、リソース DB からは、画像に対応するメタデータを抽出している。SfM のバンドル調整時は、このメタデータを使うが、内部パラメータは固定し、位置姿勢の推定値は初期値として最適化した。そのため、SfM に用いる画像群が異なると計算結果である位置姿勢の値と点群に誤差が生じる。

3D モデルを構成する点群自体の品質を評価するには、Ground truth との比較が必要になる。しかし、本研究では、3D モデルの品質向上するための手法を提案しているわけではなく、検索手法を提案している。つまり、本評価では、生成した 3D モデルは比較評価するためのものであって、それ自体が評価の対象ではないため、評価基準を設定し、評価対象との比較して評価した。目標は、提案手法を基に抽出した画像から生成した点群が評価基準と同じ点群になっていることである。さらに、「内部パラメータをキーとした検索」の結果どうしても結果も比較する。比較評価する項目は点群の数とするが、単純に数だけ比較した場合、点群の一部が密になったり、一部が欠損したりするなど、3D モデルの見た目が異なっても、数値上の変化は見られない。そこで、見た目に変化が現れる数値を基に比較するために、点の密度を一定にした。点の密度は、三次元グリッドを 1 辺が 10cm の正立方体とし、点群処理ライブラリ Point Cloud Library の Voxel Grid フィルタを用いたダウンサンプリングによって一定にしている[40]。以後、比較する点群は、このダウンサンプリングしたものとする。

評価 4 では、抽出した画像からパノラマ画像を生成し、仮想カメラが想定した視野のパノラマ画像になっているかを、点群とパノラマ画像とを比較し評価した。

### 6.4 「外部パラメータをキーとした検索」の動作確認

本節では、時空間軸として図 6-4 に示すアプリ座標系上の赤丸（時計台の中央）と正午を中心に、範囲は、 $x$ 軸と $y$ 軸を 10m、 $z$ 軸 6m、時間 1 時間を指定して検索した。検索した結果、714 枚が抽出されたことを確認した（図 6-4 は本論文のために地理院地図と OpenStreetMap[41]を加工して作成した図であるが、本フレームワークではこれらの地図は使用していない）。

図 6-5 には、714 枚を 5 月と 12 月に分けて抽出した画像を基に三次元再構成した結果を示す。12 月の再構成結果と比較して、5 月の再構成結果のほうが木々が青々しいことがわかる。このように、空間軸だけではなく、時間軸も検索値に使用できることから、よりニーズに応じた画像を検索することも可能である。CV 分野の処理においては、太陽高度などによる陰影や植生の条件をそろえることが可能なため、特徴点の検出や、対応付けの精度を向上させることもできる。

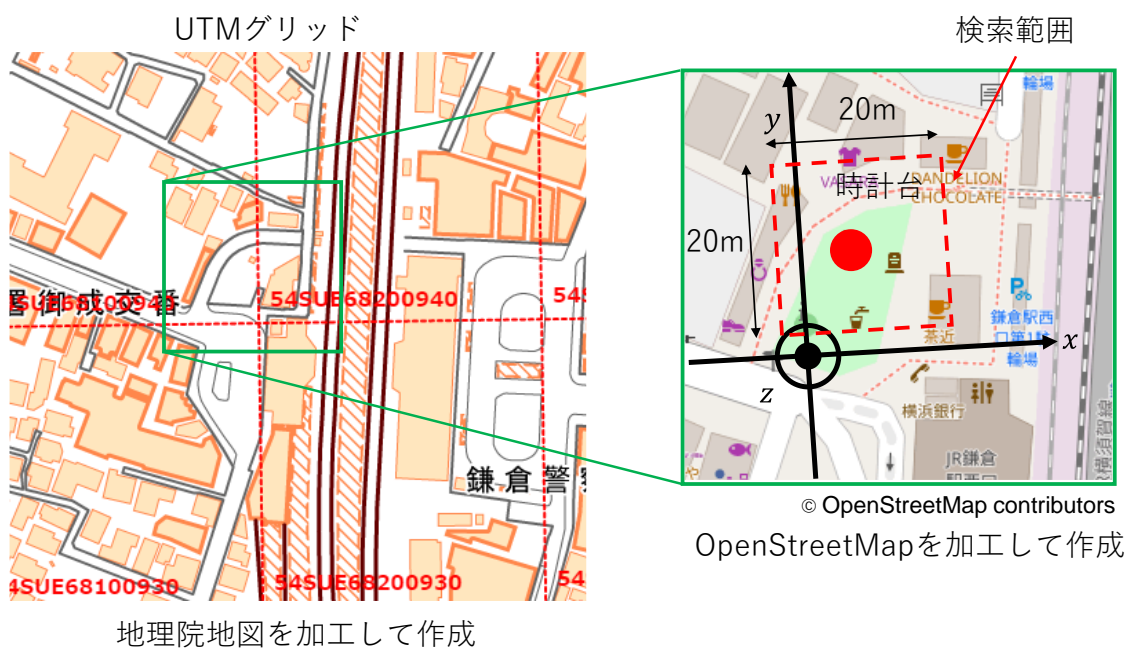


図 6-4 検索範囲

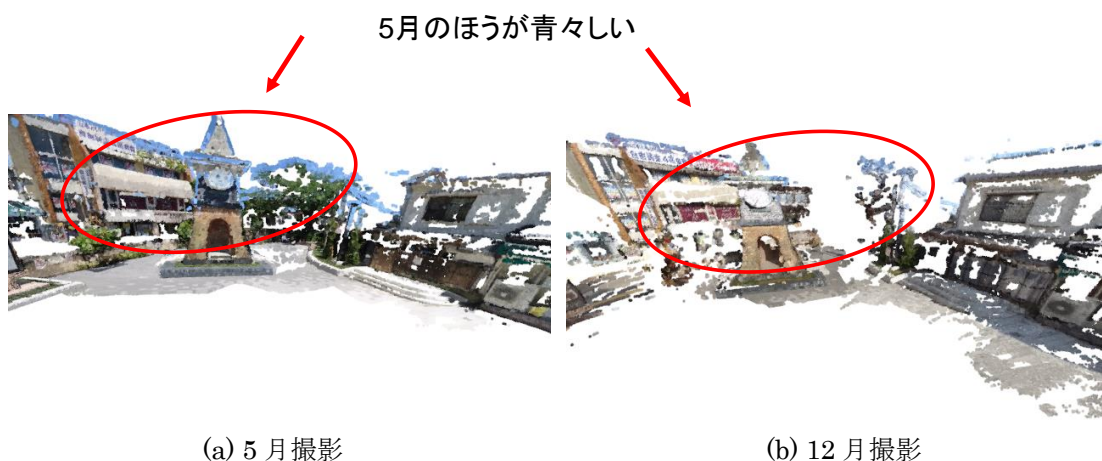


図 6-5 時間軸による変化

## 6.5 「内部パラメータをキーとした検索」の評価

本評価では、4.4 節で使用した「外部パラメータをキーとした検索」に条件を追加して検索している。つまり 714 枚から「内部パラメータをキーとした検索」により抽出した画像を基に評価した。

### 6.5.1 評価 1: 「注視点に基づく検索」の比較評価

本評価では、時計台近くの案内図を三次元再構成するために、図 6-2、図 6-3 に示した青丸の座標値を注視点とし、「注視点に基づく検索」を行った。検索した結果、168 枚を抽出した。抽出した画像を用いて三次元再構成した結果、案内図が再構成されていることがわかる(図 6-6)。

714 枚の画像により再構成した案内図の点群（評価基準の点群）は 1826 点であり、評価 1 の検索結果における画像により再構成した案内図の点群は 1737 点であった。三次元再構成に使用した画像は 23.5%であったが、点群は 95.1%であったため 3D モデルの品質はほぼ同等と考えられる。

一方で、図 6-6 から時計台で遮蔽された画像も抽出していることもわかる。筆者が目視で数えたところ 16 枚が時計台の遮蔽により案内図を視認できなかった。遮蔽も考慮するためには、画像中の物体を認識する必要がある。

## 6.5.2 評価 2 : 「視錐台カリングに基づく検索」の比較評価

本評価では、時計台を三次元再構成するために、時計台を囲むように AABB の頂点 8 つを指定し（図 6-3 に示した赤枠）、「視錐台カリングに基づく検索」を行った結果、615 枚を抽出した。抽出した画像を用いて三次元再構成した結果を図 6-7 に示す。この再構成した点群は 16,035 点であり、714 枚の画像により再構成した時計台の点群（評価基準の点群）16,102 点と比較しても 99.6%の点群を維持できている。99.6%になった理由は、同じ AABB 内であってもバンドル調整により点群の座標値が評価基準時と異なるため、ダウンサンプリング時に異なる点が削減された可能性があるため、3D モデルの品質は同等と考えられる。

次に、AABB を基に自動的に ROI を計算可能なことを確認し、ROI の有無によりデータ総量が減少しても品質を保てるかを評価した。

4.3.1 項で提案したとおり、615 枚の各画像に対し、ROI を別途指定することなく ROI 圧縮できることを確認した。本論文では、ROI と非 ROI とで異なる圧縮率とし、WebP 形式に変換した。リソース DB 登録形式である png 画像の全容量が 769[MB]であり、WebP 形式に変換した場合、163[MB]となる。この画像に対して、ROI 画像圧縮した場合は 133[MB]となり、ROI 画像圧縮による圧縮率は 18.5%であった。圧縮前の画像を用いて三次元再構成した結果を図 6-8(a)、圧縮後の画像を用いた結果を図 6-8(b)に示す。また図 6-8(c)(d)は使用した画像の一例である。

図 6-8(d)の左側が非 ROI であり、画質が低下していることがわかる（図 6-9）。図 6-8(e)は図 6-8(a)の点群から ROI を適用したことにより欠損した点群の色を赤で表現したものである。

図 6-8(e)より、欠損した点群の大半が、AABB の範囲外であることがわかる。三次元再構成の品質を評価したところ、ROI 圧縮した画像を使用した場合、点群が 158,044 点となり、98.6%の点群を維持できたため品質は同等と考えられる。

ROI 画像圧縮により、三次元再構成の品質を低下することなく、さらに各画像に対して自動的に ROI を指定でき、データ総量も減少可能であることが確認できた。

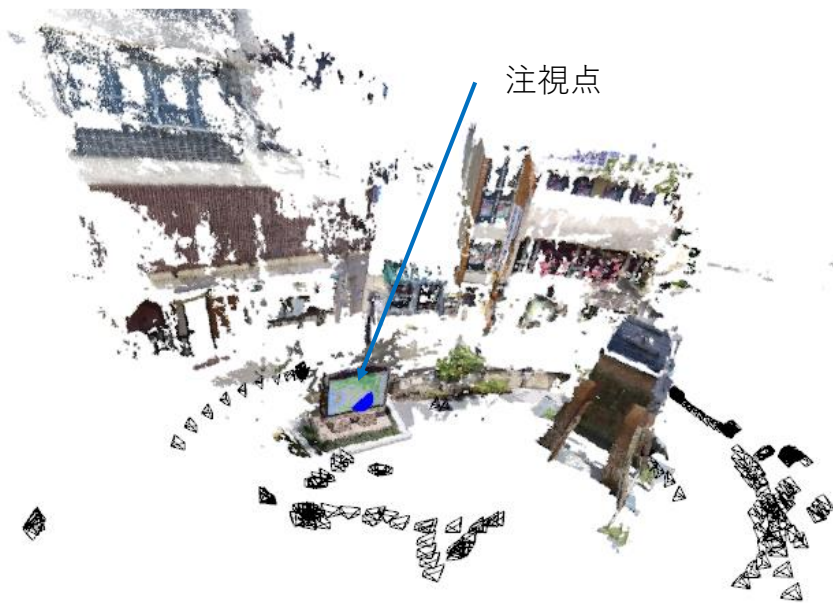


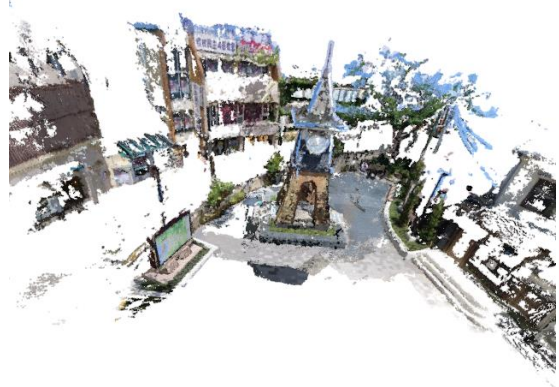
図 6-6 案内図 (評価 1 の結果)



図 6-7 時計台 (評価 2 の結果)



(a) 再構成 (ROI 非適用)



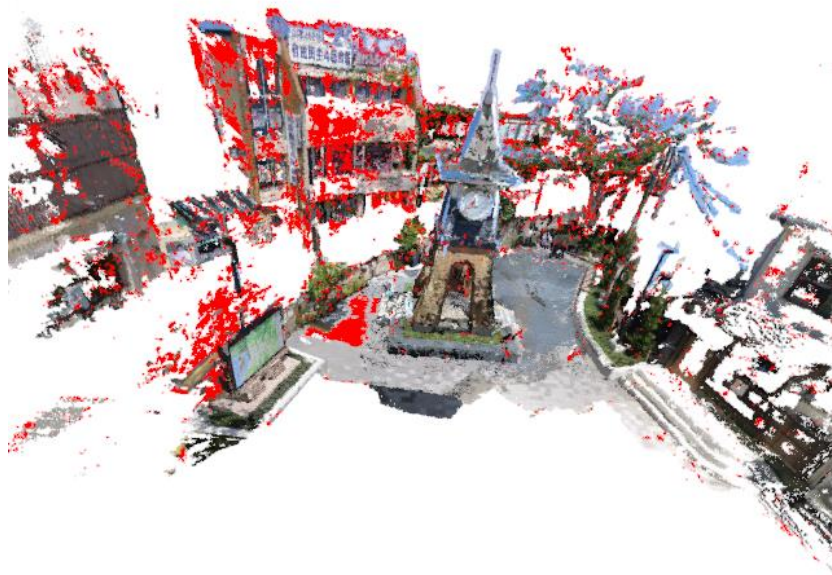
(b) 再構成 (ROI 適用)



(c) 画像 (ROI 非適用)



(d) 画像 (ROI 適用)



(e) 再構成 (差分)

図 6-8 ROI 適用との比較



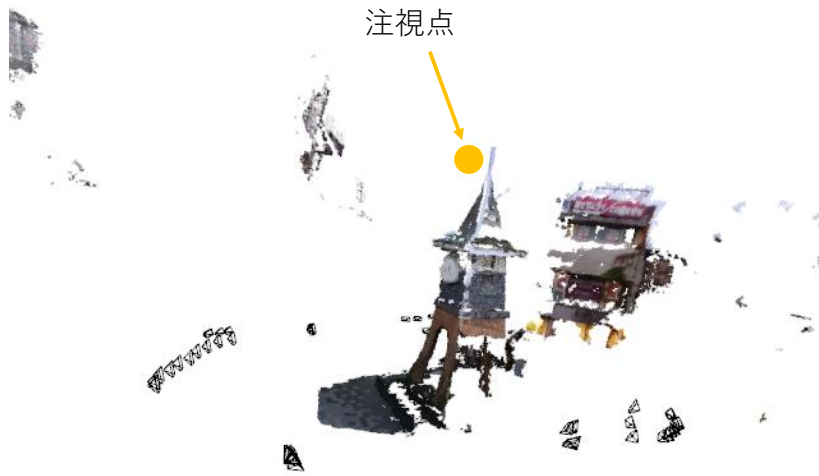
図 6-9 圧縮品質の比較

### 6.5.3 評価 3 : 「注視点に基づく検索」と「視錐台カリングに基づく検索」の比較評価

本評価では、「注視点に基づく検索」と「視錐台カリングに基づく検索」を比較するために、あえて三次元再構成結果が悪くなりそうな点を注視点の座標値に指定することで、三次元再構成の結果に影響が出ることを示す。例えば、図 6-3 に示した黄色い丸の座標値を注視点に設定し「注視点に基づく検索」し、42 枚を抽出した。これらの画像を用いて三次元再構成した結果を図 6-10 に示す。本評価における点群は 5376 点であり、評価 2 で、「視錐台カリングに基づく検索」した結果と比較すると、33.4% になっており、品質が低下していることがわかる。評価 1 では品質よく再構成できた「注視点に基づく検索」であるが、時計台は案内図より大きく、撮影位置からも多いため悪い結果となった。注視点として選ぶ場所や DB に蓄積された画像、再構成対象の大きさや距離によって三次元再構成の結果は大きく影響を受けていることがわかる。

一方、三次元再構成の品質だけを比較した場合、「視錐台カリングに基づく検索」が適しているように見える。しかし、検索により抽出した画像は DB からダウンロードすることを考えると、画像のデータ量は少ない方が好ましいが、視錐台カリングに基づく AABB に入っているカメラは、時計台を向いていなくても抽出される。時計台を三次元再構成する目的を満たすには、画像中に時計台が少しでも写っている必要があるが、筆者が目視で数えたところ、この目的に合っていない画像が 103 枚あった。これはカメラの位置が AABB 内であるが、被写体を向いていない画像である。これを改善するためには、「注視点に基づく検索」などの他の検索手法と組み合わせが必要になる。





(a) 注視点に基づく検索



(b) 同じ点から見たの両検索の比較

図 6-10 時計台 (評価 3 の結果)

#### 6.5.4 評価 4 : 「エピソード幾何に基づく検索」の比較評価

図 6-3(b)の位置から水平画角  $90^\circ$  , 解像度 2K の仮想カメラを設定し「エピソード幾何に基づく検索」をした結果, 108 枚を抽出した (図 6-11(a)). この 108 枚から, 文献[7]の手法を基に, この 108 枚中の 10 枚を使用し, 生成したパノラマ画像を図 6-11(b) に示す.

仮想カメラから見た景色がパノラマ画像において生成されていることがわかる. 「エピソード幾何に基づく検索」では仮想カメラの視錐台内を撮影しているカメラ画像は全て検索されるため, 本来ならば, 仮想カメラから見た景色よりも広視野な画像が生成される. 図 6-11 に示すとおり, 案内図よりも左側までパノラマ画像が生成された. パノラマ画像を生成するときは, 必要に応じてトリミングするなどの処理が求められるが検索としては, 十分な画像を抽出できたといえる.



(a) 仮想カメラから見た点群



(b) パノラマ画像

図 6-11 「エピポーラ幾何に基づく検索」の評価結果

## 6.6 まとめ

本章では、様々な視点かつ様々な機種のカメラから収集した大量の画像を IBMR に用いるために設計した検索機構を評価したことについて述べた。一言で IBMR と言っても様々な手法があり、利用目的・利用手法に応じて必要な画像も変わってくる。このような状況下で、利用者のニーズを満たすには、IBMR の処理の流れを整理し、抽出を所望する画像を明確化する必要があった。

本提案では、SfM などの IBMR の手法を概観し、抽象化した処理の流れを基にカメラの外部パラメータを検索キーにする手法や、視錐台カリング、エピポーラ幾何など、一般的には検索に使用しない既存技術を応用した検索手法を提案した。提案した検索手法は検索機構“SIGMA Retriever”の機能として SIGMA フレームワークに実装し、利用目的・利用手法に合った画像

を抽出できることを確認した。しかし、評価において、視錐台カリングやエピソード幾何に基づいた検索手法で抽出した画像は必要な画像を含んでいたものの、提案した手法では不必要な画像も含んでいた。本研究は利用目的・利用手法を狭めることなく、IBMR の広範なものに対応することを目標にしていたため、提案手法によって必要条件是満たした。一方で、提案手法のままでは不必要な画像も含む可能性がある。4.5.1 項でも述べたとおり、画像だけでは、より高度な検索手法を提案することが難しい。そこで、リソース DB に蓄積した画像に紐づくメタデータだけではなく、画像から生成した 3D モデルも活用できるようにリソース DB を拡張するとともに、活用方法を検討する。

今後は、さらなる利用目的・利用手法を視野に検索手法を検討するとともに、IBMR だけではなく、物体検出などの異なるアプローチから検索手法を検討する。

## 第7章 高度利用方法の検討

### 7.1 検討方針

3章から6章まで、SIGMA フレームワークを構築するための設計について述べた。ここからは SIGMA フレームワークを用いたコンテンツ生成について述べる。

本研究では、カメラなどの視覚センサにより観測した実世界の情報を利用するため、コンテンツも検討している。世の中にあるフレームワークやライブラリは、分野の大小はあれ、対象となる分野は決まっている。例えば、GLFW はグラフィック描画 API である OpenGL を使用しやすくするためのライブラリであり、ARToolKit は拡張現実感 (Augmented Reality; AR) を実現するためのライブラリである[42]。本フレームワークでは実世界を観測した時空間映像データを用いたコンテンツ生成するためのアプリケーションを効率よく開発することを目的としている。

アプリケーションフレームワークの利点として、次のものがある。

- 開発速度が向上
- 再利用可能な機能が実装済み
- コードの記述方法を統一可能

これらを満たすためには、アプリ開発者が開発するアプリを想定し、API などを整理する必要がある。そこで、当該分野において、必要となる機構を検討した。

- Pose Estimator : Resource の位置姿勢を推定
- 2D-3D Mapper : 2次元座標値と3次元座標値を対応付け
- Muti-view Stereo : 異なる多視点の情報から3次元情報を再構成
- Communicator : クライアントアプリとサーバとの通信
- Database Manager : データベース制御
- Resource Selector ; Resource や Object の検索
- Data Encoder : データの符号化, 圧縮, 展開
- Rendering Engine : 描画

これらの機構には、様々な API を定義しており、本フレームワークを構築時に使用するだけでなく、アプリ開発者にも公開する。

一般的に三次元再構成というと撮影画像を入力し、3D モデルを生成するまでのものが多い。同様に複合現実感 (Mixed Reality; MR) や AR という撮影画像の位置姿勢に合わせて仮想物体を重畳描画するまでを含む[43]。このように三次元再構成や複合現実感の名称は、パイプライン処理の総称として使われることが多い。前述したとおり、三次元再構成では、COLMAP[44] や Theia[45], OpenMVG[46] など、入力画像から点群を再構成までの処理をする SfM ライブラリが多く存在する。特に COLMAP は疎な点群から密な点群の生成まで可能である。Theia や OpenMVG では密な点群を生成するまでのサポートがないため、OpenMVS[47] などの別のライブラリに依存する必要がある。

このように既存ライブラリとして多く存在するものは、まずそれらの構成は参考した。4.3 節でも述べたとおり、入力画像から再構成される点群を副産物とみなし、主題をカメラの位置姿勢推定とした場合、Theia, OpenMVG に含まれてる API の多くは「Pose Estimator」とみませる。一方、COLMAP は密な点群の生成まで可能なことから、「Pose Estimator」「Muti-view Stereo」を含む。さらに ARToolkit は撮影画像の位置姿勢を推定する API が多いため、「Pose Estimator」に分類できる。これらのライブラリやフレームワークの API を参考にすることで、本フレームワークで実装する API の設計が可能となる。

このように既存のライブラリやフレームワークの調査は必要不可欠であるが、机上検討だけでは、アプリ開発者のニーズに応えられない。そこで、調査と並行して、本フレームワークのコンセプトの下、収集した画像を用いて生成できるコンテンツを検討すると、ともに実課題に対するソリューションを念頭にコンテンツを検討した。

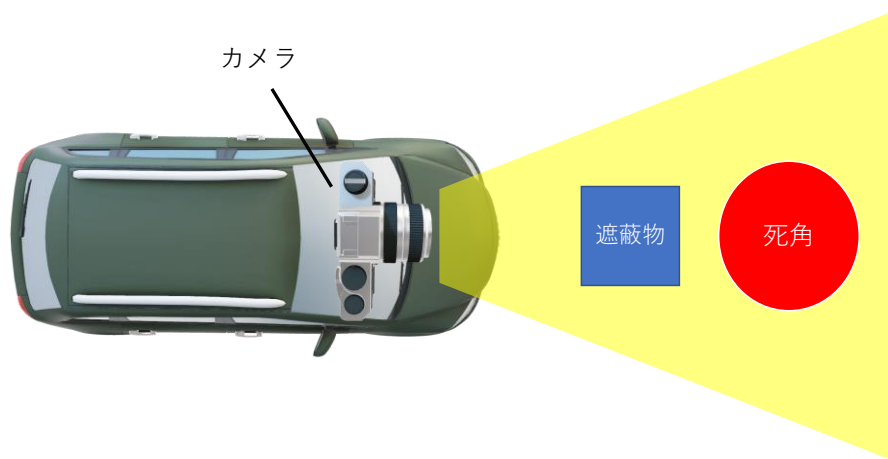
## 7.2 コンテンツの選定

開発するコンテンツは、時空間映像データのみを使用することを前提に検討した。MR/AR では仮想物体を重量描画することから、時空間映像データからは生成できない。一方、近年では MR/AR を含め XR を総称することが多い。筆者の研究グループでも、DR の研究に取り組んでいることから、これをひとつのテーマとして着目した。

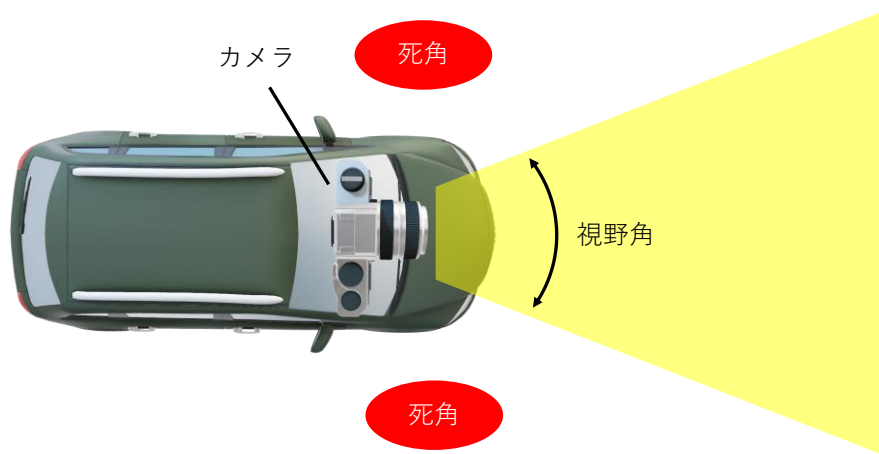
DR はカメラ映像中から何らかの方法で視覚的に遮蔽物を隠蔽・消去する技術であるため、遮蔽物に起因する死角を減らすことができる (図 7-1 (a))。しかし、カメラ映像における死角は遮蔽物に起因するものだけではなく、視野角に起因するものもある (図 7-1 (b))。

視野角に起因する死角を減らすために、より広視野なカメラ映像を生成できる製品が登場している。たとえば、魚眼レンズや全天周カメラなどが、その 1 例である。しかし、これらのハードウェアによる解決策はレンズの歪みや解像度の低下などの原因となる。ソフトウェアによる解決策も多く提案されている。たとえば、日産自動車のアラウンドビューや三菱電機の Fairyview<sup>®</sup>、沖電気工業のフライングビュー<sup>®</sup>などのように、複数視点のカメラ映像を縫い合わせるように接合する Photo Stitching が様々な場面で運用されている[48]。

視野角に起因する死角や、遮蔽物に起因する死角は、各シチュエーションに応じて解決策が講じられてきた。しかし、これら 2 つの死角の発生条件は排反事象ではない。たとえば、周辺監視のために複数台のカメラを設置して広視野画像を生成したとする。この場合、カメラ設置位置によっては遮蔽物が原因で遠方まで見渡せない状況もありうるし、設置時は問題がなくとも、後に遮蔽する物体が出現する可能性もある。遮蔽された死角領域を撮影するカメラを別途用意した場合、この画像を見るためには広視野画像とは別のウィンドウもしくはモニターが必要となるが、監視者にとって複数の画像を同時に監視することは身体的な負担が大きい。死角領域の数が増加するにともない負担も増加することから、本研究は広視野画像に対して、身体的な負担が小さく、かつ直観的に空間を把握しやすくするために、DR 技術を用いて遮蔽物を視覚的に除去する手法を提案する。



(a) 遮蔽物による死角



(b) 画角による死角

図 7-1 死角の種類

### 7.3 隠消現実感技術の検討

BOSCH は前方車両によるドライバーの死角をなくすために、前方車両を別視点のカメラ映像を用いて消去した[49]。Kido らは、古くなったビルなどの建築物を解体・撤去した後の景観シミュレーションをするために、建築物など非移動物体と、人や車などの移動物体を同時に除去するシステムを開発した[50]。このほかにも DR 技術は、様々な分野で展開が始まっている。

DR は主に下記の 3 つに分類できる[51]。

- 事前観測型 DR (Pre-Observation Based DR : POB-DR)  
除去対象がない状態で、除去対象の背後の光景を事前に観測し、DR 時に利用する手法であり、除去対象の背後の光景が静的な場合に有効
- 実時間観測型 DR (Real-time Observation Based DR : ROB-DR)  
DR 時に除去対象の背後の光景を実時間で観測し、利用する手法であり、除去対象の背後の光景が動的な場合に有効

- 画像修復型 DR (Image Inpainting Based DR : IB-DR)

除去対象周辺の画素などを用いて除去対象の背後の光景を推定し, Inpainting する[52].

本研究では, Photo Stitching と DR とを併用することで, 2 種類の死角への解決策を講じるが, 前述したとおり, 除去対象の背後の光景は動的であり, さらに死角の除去を目標にしていることから, 現実とは異なる IB-DR は不適切である. したがって, 提案する手法は ROB-DR を中心に検討する. ROB-DR を用いた遮蔽物を視覚的に除去するまでの一般的な流れは文献[53]に詳しい.

具体的には, 遮蔽物 (除去対象) の背後の光景である「隠背景 (空間)」を何らかの手段で観測することによって「隠背景投影画像」を生成し, 体験者視点に合わせてこれを除去対象物が占める領域に投影・合成することによって除去対象物を視覚的に消去する (図 7-2) .

#### 7.4 広視野画像の生成方法

次に Photo Stitching を用いた広視野画像の生成方法について述べる. 本研究では, 建物などの周辺を広範囲にわたって監視するアプリケーションでの利用を想定していることから, カメラは周辺を見渡せるように放射線状に設置することが望ましい. このとき, 各カメラが 1 つの視点を中心にパンするような配置が理想的だが, 実世界において視点を一致させることは難しく, 視差が発生してしまう. さらにビル壁面などのように設置位置が限定されている環境では, さらに視差は大きくなる.

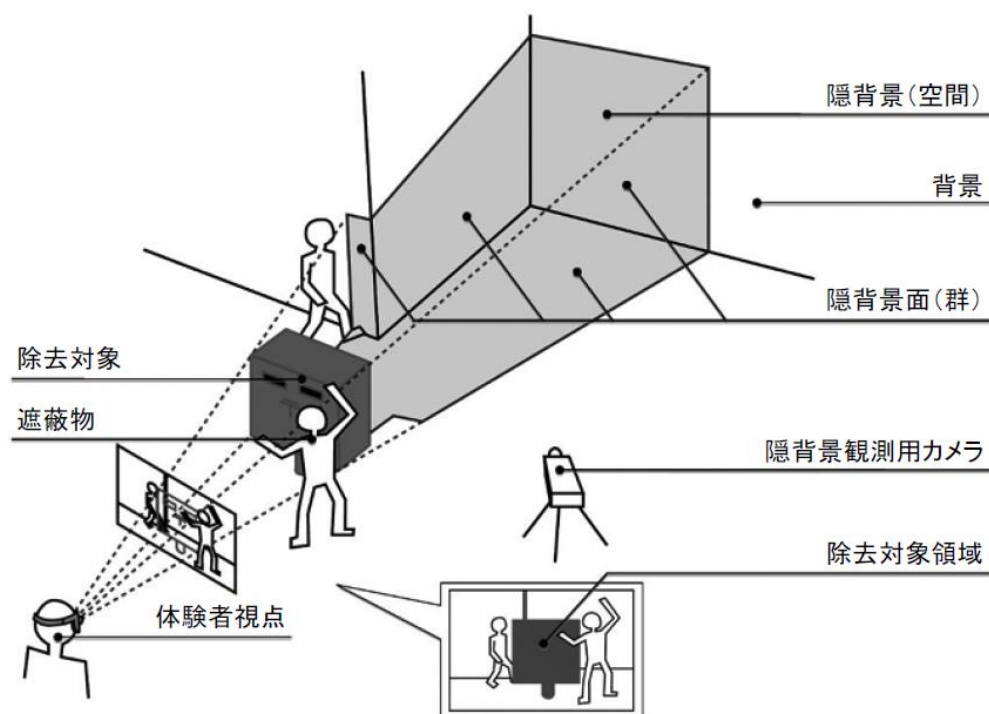


図 7-2 隠消現実感の構成要素 ([53]引用)

Photo Stitching の手法は数多く提案されているが、仮想面を設定し、そこに各カメラ画像を投影することで画像を接合している手法が多い。本研究では、これらの手法の中から、投影面の位置を無限遠と仮定しているため、視差を無視できる Brown らの手法を選んだ[33]。

本研究では広視野画像の中でも、Brown らの手法により生成した画像をパノラマ画像と称する。以降では、図 7-3 に示すカメラ画像群を元にパノラマ画像を生成する手順について述べる。図 7-3 の画像群は視差があってもパノラマ画像を生成できることを示すために撮影した画像群である。

画像接合の事前準備として、各カメラの位置関係を推定する。各カメラ画像の特徴点を検出後、任意の画像間で特徴点を対応付けし、被写体が遠方にあることを前提にホモグラフィ行列を計算した。ホモグラフィ行列から任意の座標系における各カメラの回転成分を求めた。しかし、2 画像間の回転成分は求めることはできても、複数枚で整合性を保つのは難しいため、SfM と同様に全画像の特徴点を基にバンドル調整する。しかし、被写体が無限遠にあることから、三角測量を用いて、特徴点の 3 次元座標値を計算できないため、3 次元座標値から 2 次元座標値へ投影する再投影誤差ではなく、任意の画像の特徴点を求めたホモグラフィ行列を用いて対応する画像に投影したときの距離を 0 に近づくように最適化する。



図 7-3 パノラマ画像を構成するカメラ画像群

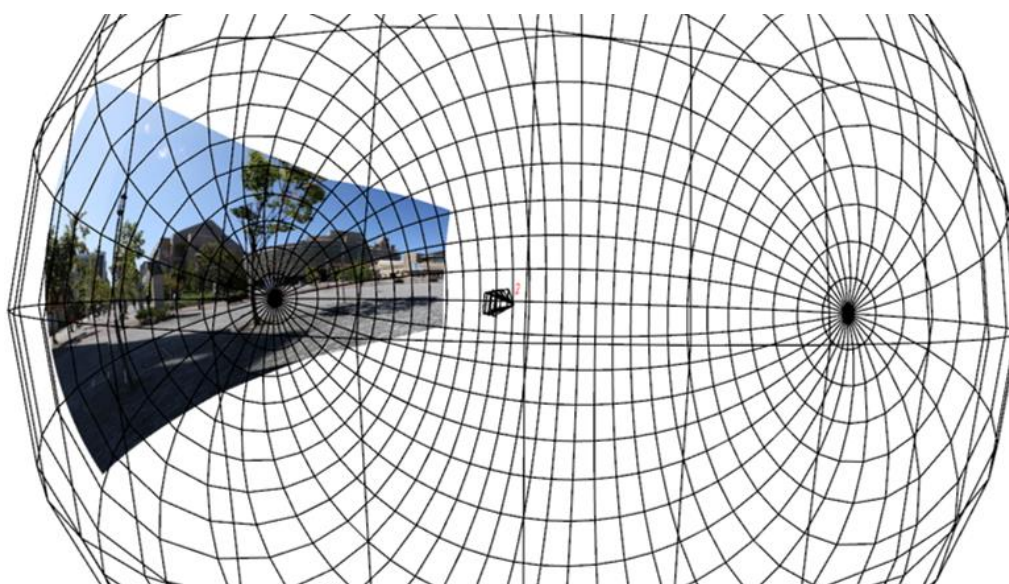


図 7-4 球状の投影面



次に仮想面を定義する。仮想面は平面や円筒，球など様々な形状のものがあるが，ここでは球にした場合について述べる。仮想面に各カメラ画像に投影後（図 7-4），正距円筒図法により 2 次元平面に展開した。本論文では，この展開した画像を Warp 画像と呼ぶ。この Warp 画像をシームレスに接合するためには画像が重複する領域をブレンディングする必要がある。このとき，重複する領域の画素を一様に混ぜるわけではなく，重複する領域内で境界線を決め，その境界線を基に画像をブレンディングする。Warp 画像間の重複する領域を処理するために，Warp 画像の有効画素からマスク画像を生成した（図 7-5(a)）。この Warp 画像とマスク画像を基に重複領域および境界線を決定することでブレンディングが可能となる。図 7-5(b)に示すのは Graph Cut アルゴリズムを用いて境界線を決めた後のマスク画像である[54]。このマスク画像を基にマルチバンドブレンディングした結果を図 7-6 に示す[55]。

マルチバンドブレンディングとは，画像を複数の周波数ごとにブレンディングする手法である。マスク画像に対してガウシアンフィルタをかけながら画像サイズを小さくしていくことでガウシアンピラミッドを作成する。ピラミッドの階層が上がれば上がるほど，画像には低周波成分しか残らない。次にカメラ画像では，画像を小さくした画像と，元画像との差分からエッジをとる（ラプラシアンフィルタ）。これを繰り返すことでラプラシアンピラミッドを生成する。生成したピラミッドの任意の階層においてマスク画像を基にラプラシアンフィルタのかかったカメラ画像を  $\alpha$  ブレンディングする。これを全ての階層で行ったのち，最上層の画像から一つ下の階層の画像サイズにリサイズ後，ブレンディングを繰り返す。

ここまで DR と Photo Stitching の手法について検討したことで，視野角に起因する場合であっても，遮蔽物に起因する場合であっても，死角を減らすためには，複数視点のカメラ映像を



(a) Warp 後のマスク画像



(b) 領域分割後のマスク画像

図 7-5 マスク画像



図 7-6 Brown らの手法によるパノラマ画像

違和感なく接合することには変わりはないことがわかる。

こうした観点から、本研究では、両課題を同時に解決するため、複数視点のカメラ映像を活用した画像接合手法を提案し、開発する。

## 第8章 広視野画像のための隠消現実感達成法の提案

### 8.1 事前検討

本研究では文献[33]に基づいてパノラマ画像を生成することを前提に、遮蔽物を除去するための手法を検討した。

複数カメラではないが、Hasegawaらはパンするカメラが撮影した歩行者を除去するために、複数フレームの画像 (=別視点の画像) を用いて補完し合う手法を提案した[56]。

しかし、パノラマ画像を構成するカメラ画像のみで遮蔽物を除去できるように設置できるに限らない。まずは隠背景を確実に撮影するために隠背景観測用カメラを設置したうえで、2つの手法について検討した。1つ目はパノラマ画像を構成する各カメラ画像に対して、遮蔽物を除去してから接合する手法である。2つ目は生成したパノラマ画像から遮蔽物を除去する手法である。1つ目の手法では、複数のカメラ画像から遮蔽物を除去するために、隠背景を各カメラ画像に対して生成し重畳する必要があるが、2つ目の手法よりも計算コストが高いことが予想される。しかし、2つ目の手法も処理フローを整理すると画像接合処理後にDR処理を実行する形となっている。7.4節でも述べたとおり、画像接合処理とDR処理は一見異なる処理に見えるが、両処理ともに「境界線を決定」し、「ブレンディング」することは変わらない(図8-1(a))。そこで、

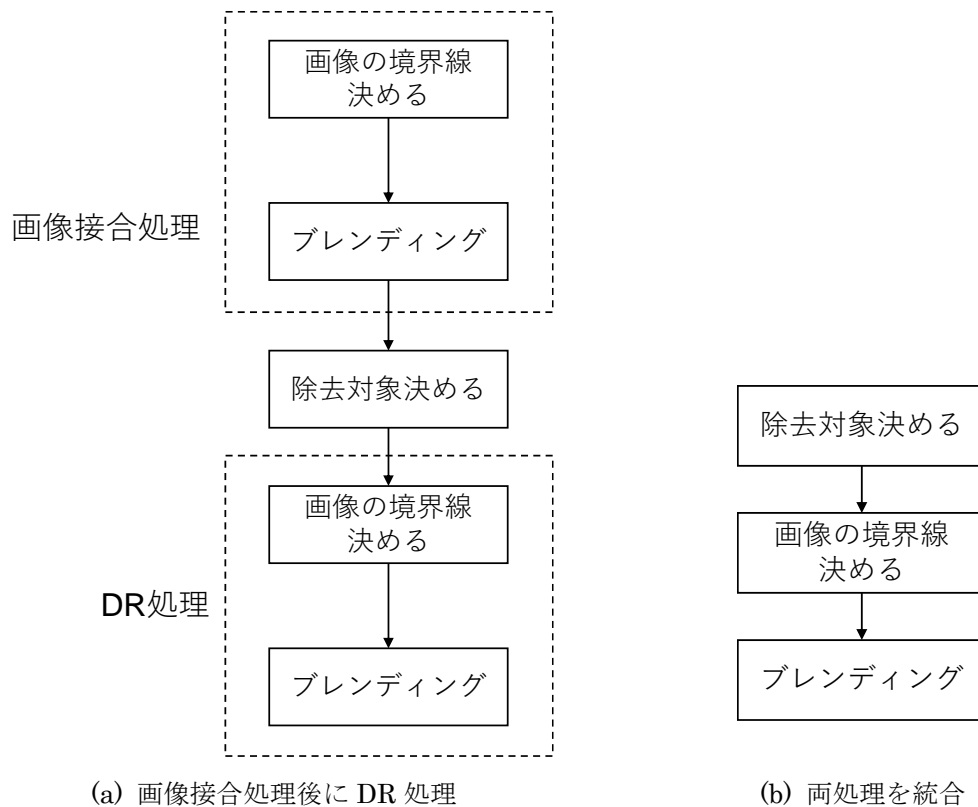
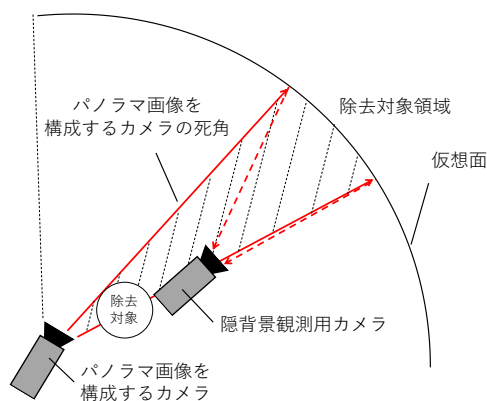


図8-1 処理フローの改善

図 8-1(b)に示すように「除去対象を決定」してから、まとめて「境界線を決定」および、その後、「ブレンド」する処理が実現可能な処理フローについて検討した。

まず、図 8-1(a)における「除去対象の決定」処理を画像接合処理における「境界線の決定」処理前に移動させることを検討した。「除去対象の決定」のために、各カメラから除去対象を抽出し、抽出した領域から DR 処理で必要となる隠背景を推定する。このとき、各カメラ画像から抽出した除去対象の領域を、仮想投影面で和集合を取ったのちに、隠背景観測用カメラ画像に投影する。図 7-3 における手前の生えている木を除去対象としたとき、隠背景観測用カメラに投影し(図 8-2(a))、各カメラの除去対象領域の和集合を図 8-2(b)に示す。このまま図 8-2(b)が示す領域を隠背景観測用カメラ画像から抽出し、パノラマ画像に重畳した場合、除去対象である木の幹が 2 本あるなど、近くの物体は視差の関係上、二重像の領域のまま隠背景観測用カメラ画像に投影され、隠背景を抽出される可能性がある。そこで図 8-3 に示すとおり、各撮影画像の除去対象と領域分割後のマスク画像の積集合を生成し、これらの和集合をとることで隠背景領域のマスク画像を生成した。このマスク画像を用いて隠背景観測用カメラ画像から隠背景を抽出した。抽出した隠背景を重畳描画した結果を図 8-4(a)に、ブレンドした結果を図 8-4(b)に示す[57]。



(a) 再投影による隠背景投影画像の生成



(b) 各カメラの除去対象を統合したマスク画像

図 8-2 投影面を用いた投影

このときのブレンドリングはポアソンブレンドリングを使用した[58].

生成した画像を評価した結果,いくつかの課題があることが分かった. **Brown** らの手法では仮想面に画像を投影していることから,カメラ画像中の3次元形状は考慮していない(図8-4(c)). そのため, 隠背景観測用カメラの設置位置は除去対象より前方に設置するなどの制限がある. パノラマ画像を構成するカメラ群とは異なり, 前方にカメラを設置するため, 地面などはパノラマ画像を構成するカメラ群と隠背景観測用カメラとで視差が生じることから, 滑らかに接合できない(図8-4(d)). この問題を避けるためには可能な限り視差が生じないカメラ位置の画像を用いて遮蔽物を除去したい.

また文献[57]の手法では DR 処理における「境界線の決定」には, 画像接合処理の境界線の情報を基に生成した隠背景が必要となるため, 画像接合処理および DR 処理における「境界線の決定」処理を統合することができない. ただし, パノラマ画像を構成するカメラ画像を用いて補間した場合, 画像接合処理のみで実現できる.

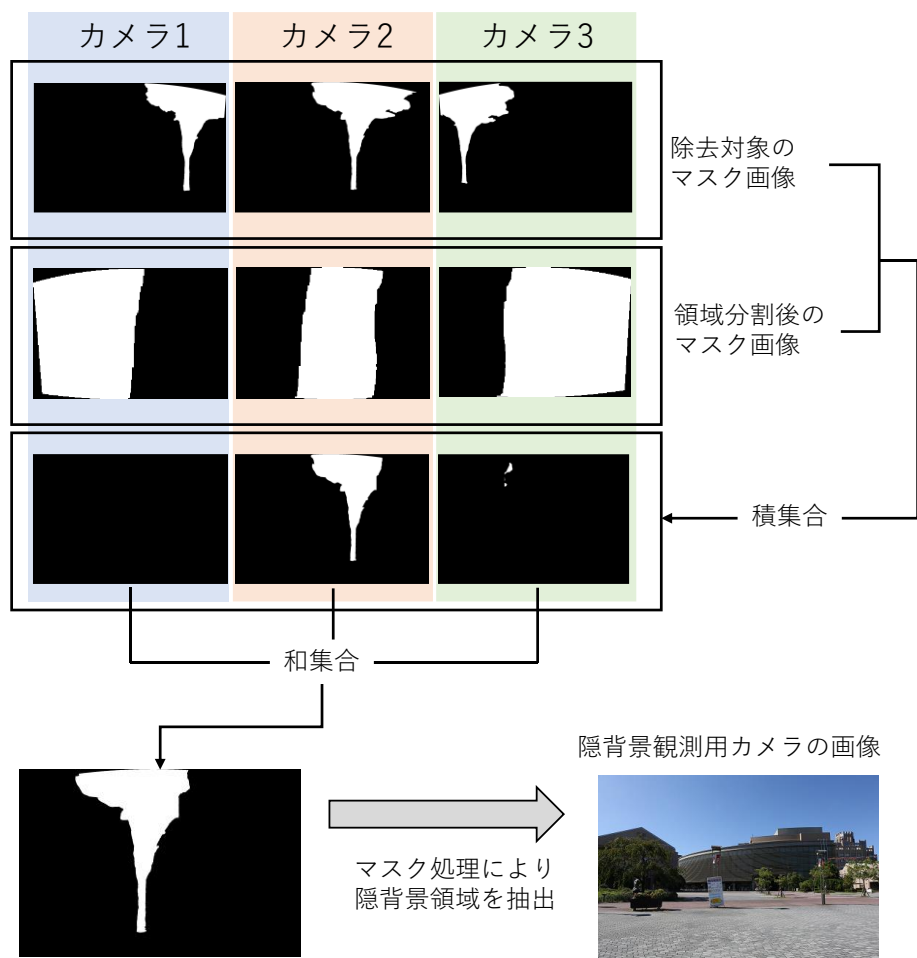


図 8-3 隠背景領域の抽出



(a) 重畳描画した結果



(b) ポアソンブレンディングした結果



(c) ブレンディングの課題



(d) 投影面の課題

図 8-4 隠背景投影画像を重畳したパノラマ画像

これらから次の 2 つを方針とするアルゴリズムを検討する。

- 隠背景に使用する画像はパノラマ画像を構成するカメラ画像を優先し、それでも不足となる領域は隠背景観測用カメラ画像を用いて補完する
- 隠背景観測用カメラ画像の使用有無にかかわらず、ほぼ同一の処理フローとする
- 隠背景観測用カメラ画像が必要な場合は、可能な限り小さく重畳する

これらの方針を満たすためには各画像の境界線の決め方が重要となる。一般に複数の画像が重複する領域における境界線は、任意の 2 枚の領域分割を全組合せで行うことで決定できる (図 8-5)。

除去対象を使用しないように、各カメラ画像の除去対象にマスク処理をして決めた境界線に基づきパノラマ画像を生成した結果を図 8-6 に示す。図 8-6(a) に示すとおり、パノラマを構成する画像だけで生成した場合、地面のズレは改善されているが、樹葉部分の隠背景が生成されていない。一方、図 8-6(b) に示すとおり、隠背景観測用カメラを追加して生成した場合、隠背景観測用カメラ画像が大きく使用され、地面のずれなどが目立つ画像となった。

つまり、隠背景観測用カメラの画像は、図 8-6(a) で示すとおり、隠背景が足りない領域に限定して重畳するが、別視点であることから違和感の原因となるため、可能な限り小さく重畳したい。ただし、図 8-6(a) を生成してから、足りない領域を推定して重畳するのではなく、図 8-1(b) の処理になるような手法を提案する。

## 8.2 提案手法

所望の領域で分割できる境界線を決定するアルゴリズムを設計することは難しい。Graph Cut のように画素情報から違和感なく領域分割する手法は提案されているが、画素値を用いた機械

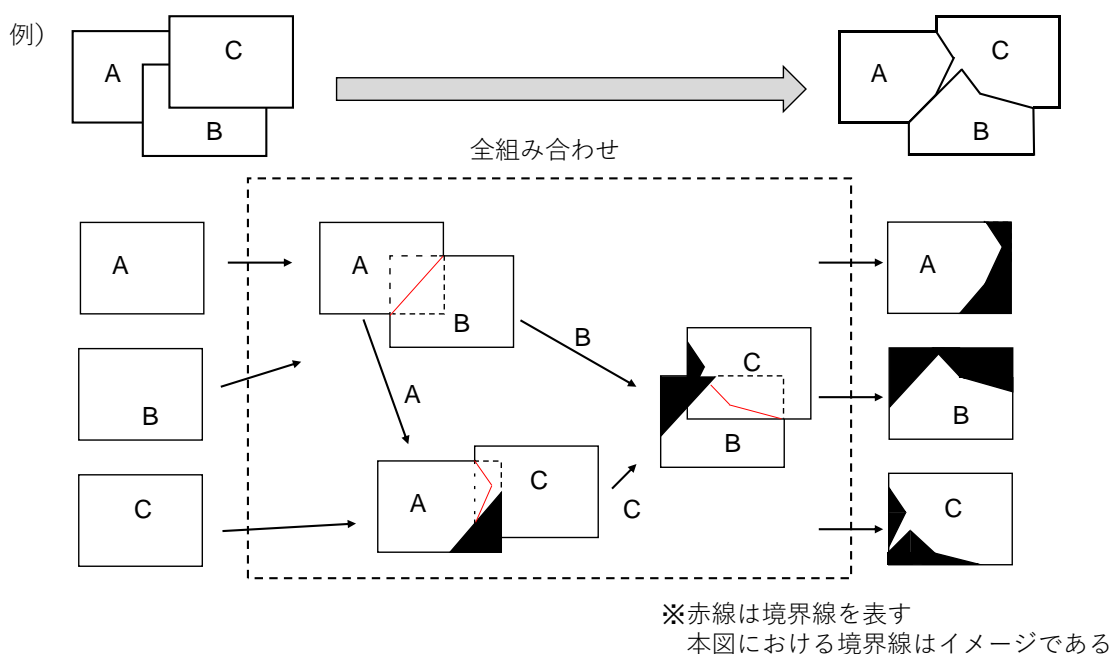
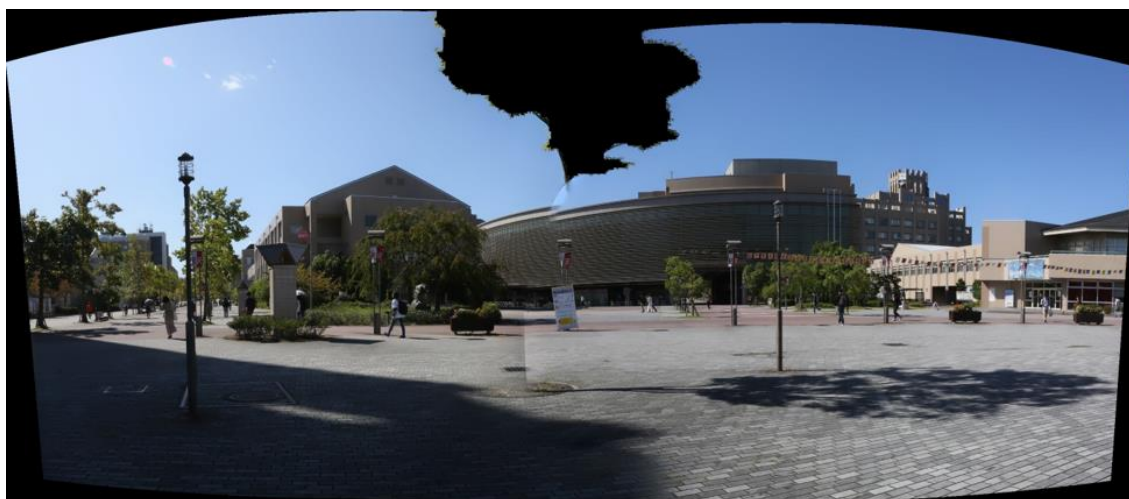


図 8-5 領域分割

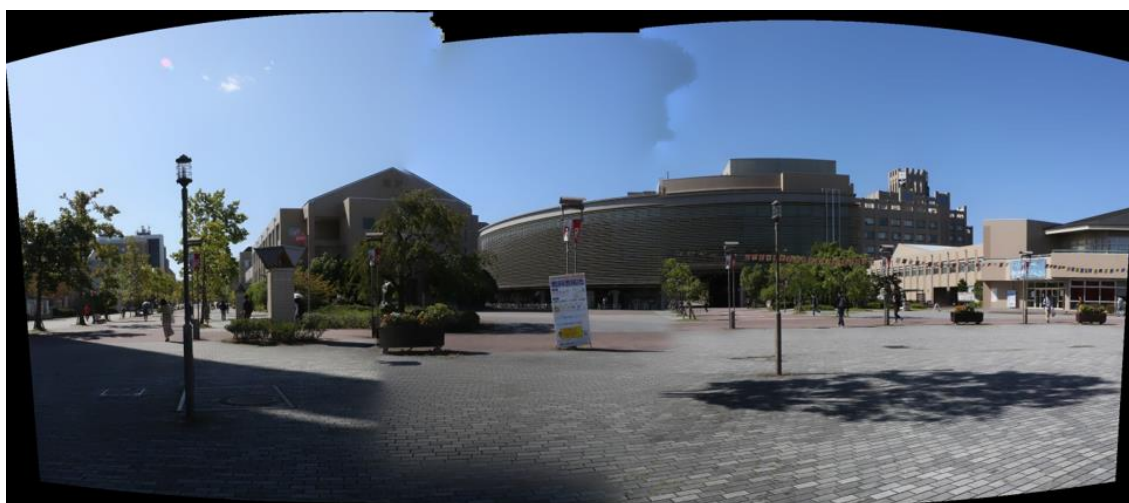
的な分割であって、人の感性の指針とは異なる場合がある。そこで Summa らは手動で境界線を動かすことで、違和感の少ない領域分割手法を提案した[59]。しかし、DR などの xR は、実世界情報を実時間で反映する必要があるため、手動により境界線を決める手法は適切ではない。

そこで本研究では境界線決定のために 2 種類のマスク画像を使用する手法を提案する。この手法自体は、広視野画像の生成において Brown らの手法に制限されるものではなく、他の手法にも有効である。

使用するマスク画像は図 8-7 に示す 2 パターンである。図 8-8 に示すとおり、パノラマ画像を構成するカメラ画像どうしで領域を分割後に、隠背景観測用カメラ画像と領域を分割する。まずパノラマ画像を構成するカメラ画像どうしの領域分割処理では図 8-7(a)に示す除去対象をマスクした画像（除去対象領域マスク画像）を使用する。次にこれらのマスク画像を用いて、隠背景画像観測用カメラの画像のマスク画像との領域を分割するが、そのまま使用した場合、除去対



(a) パノラマを構成する画像群のみ



(b) パノラマを構成する画像群と隠背景観測用カメラ画像

図 8-6 マスクを用いて生成したパノラマ画像



象の近くで分割される保証はない。そこで、除去対象の近くで分割するために、図 8-7(b)に示すマスク画像と、領域分割後のパノラマ画像を構成するカメラ画像のマスク画像（除去対象周辺領域マスク画像）の積集合を使用する。両画像は使用領域内（マスク画像における白）でのみ分割されるため、除去対象の近くでの分割が可能である。

隠背景観測用カメラ画像との組合せは、パノラマ画像を構成するカメラ画像間で領域分割した後になるが、組合せ総数自体は変化しないため、同枚数を用いた画像接合処理よりも次の 2 処理が増えるのみである。

- 除去対象領域マスク画像から除去対象周辺領域マスク画像を生成
- 領域分割後のマスク画像と除去対象周辺領域マスク画像との積集合演算

ブレンディングは各画像の境界線を基に行うため、境界線が設定した全画像に対しては隠背景観測用のカメラ画像があっても同じように処理が可能である。

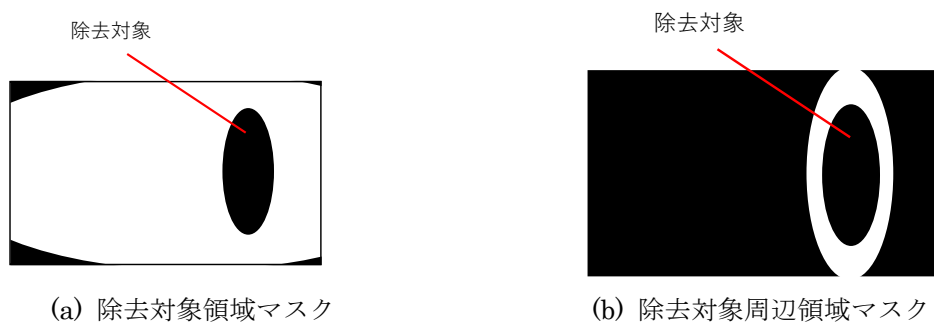


図 8-7 2 種類のマスク画像

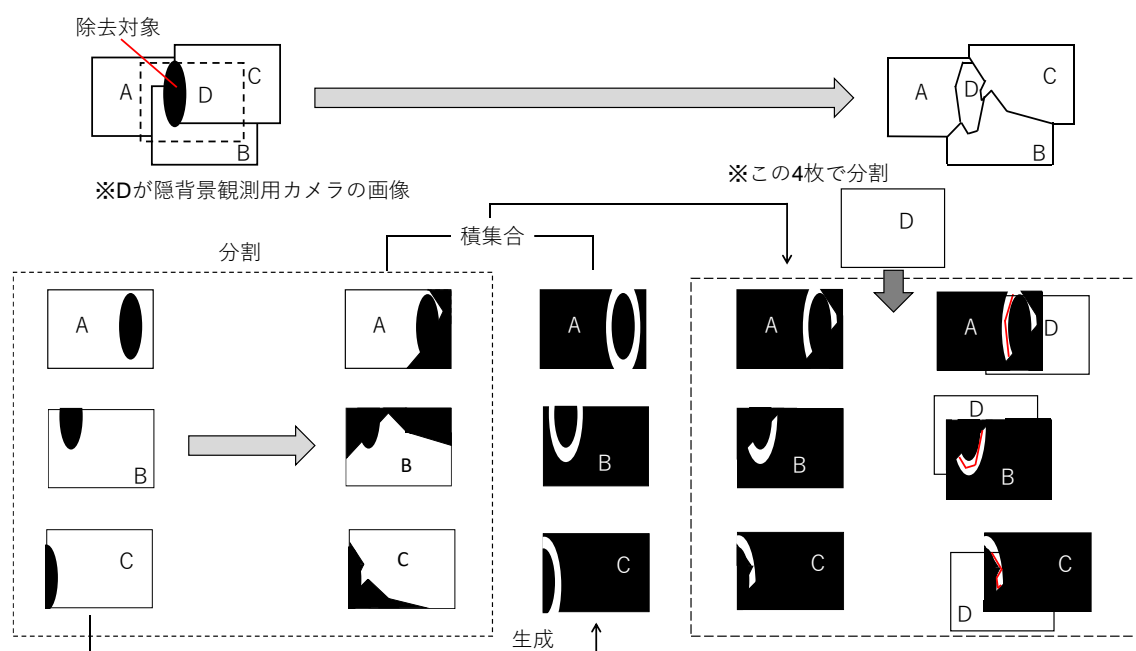


図 8-8 提案手法による領域分割

### 8.3 Graph Cut への応用

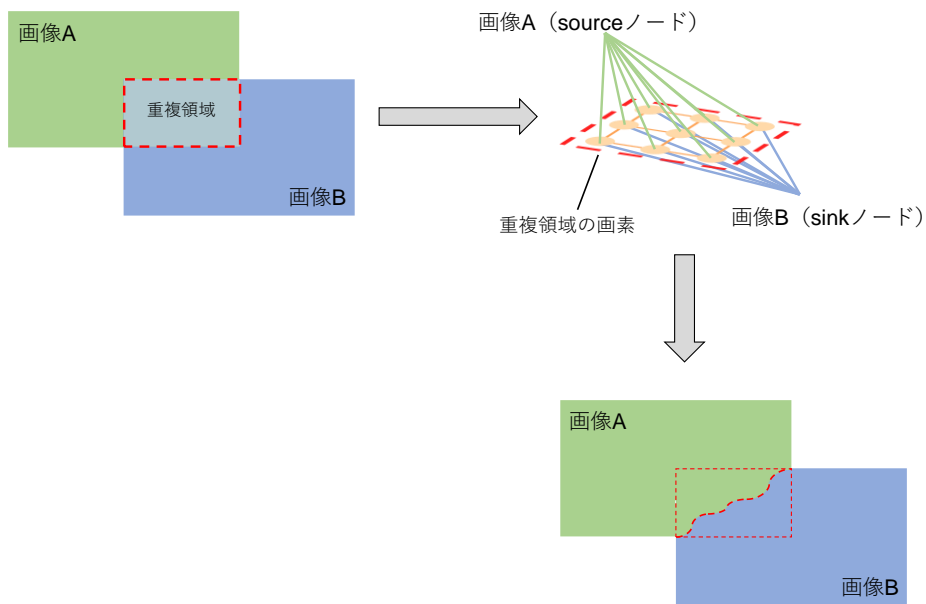
Graph Cut はパノラマ画像の生成において多く用いられており，本研究でも境界線を決めるのに使用している．提案手法を Graph Cut に応用する手法を検討する．

領域分割において Graph Cut を用いて 2 枚の画像の重複領域を分割するためには図 8-9(a) に示すとおり，重複領域の画素がどちらの画像の画素になるかを判定する．このとき，重複領域のカットエネルギーが最小となるように分割するため，これを最小カット問題と呼ばれている．この問題は画像 A (source ノード) から画像 B (sink ノード) に水を流したときの最大流量となる経路を探索することと同義のため，最大フロー問題とも呼ばれている．

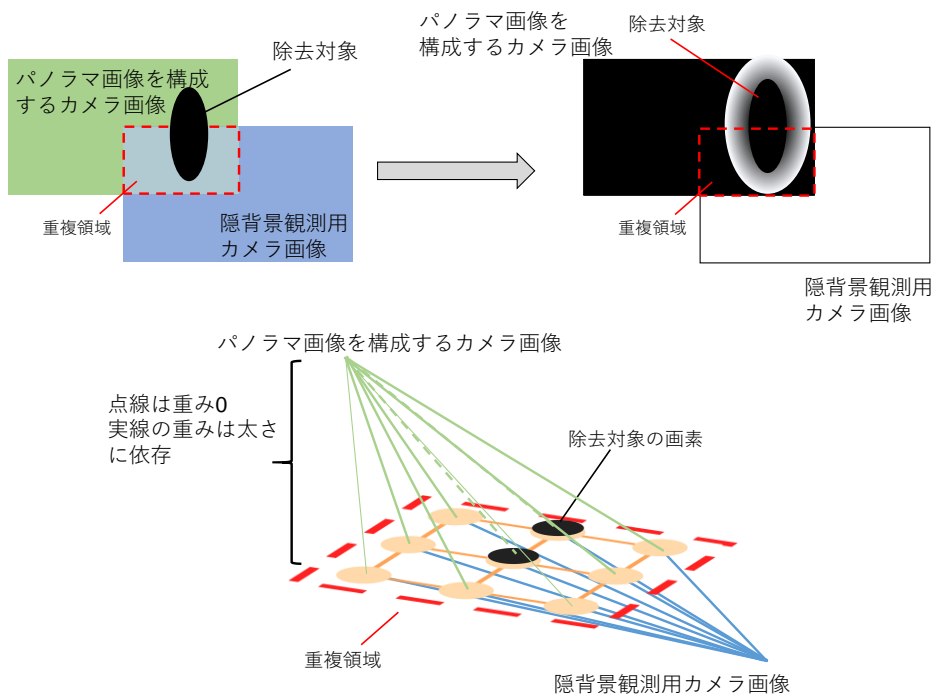
重複領域の各画素のノードは，マスク画像を基に source ノードや sink ノードとのエッジの重みを決める．エッジの重みは許容水量と見なすことができる．たとえば，画像 A における不使用領域 (黒) とのエッジの重みを 0，使用領域 (白) とのエッジの重みを 10,000 とした場合，重みが 0 の経路は水が流れなくても許容水量に到達したと見なせることから画像 A と判定されることはない．

本提案手法における隠背景観測用カメラ画像の使用領域は可能な限り小さくしたい．そこで，除去対象周辺領域マスク画像に工夫を加える．除去対象周辺領域マスク画像の除去対象領域からのユークリッド距離に応じて黒から白へとグラデーションする距離変換画像を生成しマスク画像の代替とする [60]．そうすることで，図 8-9(b) に示すとおり，重複領域における各画素からパノラマ画像を構成するカメラ画像へのエッジの重みは除去対象領域に近づくに連れて小さくなる．つまり，許容水量が小さくなるため，パノラマ画像を構成するカメラ画像と判定されにくくなる．

これは source ノードや sink ノードへのエッジの重みを混合正規分布に基づいて定義し，前景と後景に領域分割する GrabCut の思想に近い [61]．



(a) Graph Cut による領域分割



(b) 提案手法を Graph Cut に応用した場合の領域分割

図 8-9 Graph Cut による例

## 第9章 広視野画像のための隠消現実感達成法の評価

### 9.1 評価の目的と環境

本評価では除去対象をマスクした画像と撮影画像を入力とし、遮蔽物が除去された広視野画像が生成されることを確認する。評価に用いた環境は次の2つである。どちらも立命館大学びわこ・くさつキャンパス内で撮影した。

#### 【環境 A】

パノラマ画像を構成するカメラ群から除去対象である木までは約10mである(図9-1)。一方、隠背景観測用カメラは除去対象である木の近くに配置し、風景として映っている建築物までの距離は約70mの位置に配置した。

#### 【環境 B】

パノラマ画像を構成するカメラ群から除去対象である倉庫までは約12mである(図9-2)。一方、隠背景観測用カメラは除去対象である倉庫の近くに配置し、風景として約250m以内には建築物がない環境である。

また、提案手法の処理時間を比較することで効率的に画像が生成できることを示す。処理時間の計測に使用した画像は環境Aで撮影した4枚の画像であり、比較対象として、次の2つの手法も実装し、処理時間を計測した。

**手法 $\alpha$**  パノラマ画像を構成するカメラ群の画像3枚をBrownらの手法によりパノラマ画像を生成する。その後、パノラマ画像上の除去対象から隠背景を推定し、Graph Cutとマルチバンドブレンディングを用いて隠背景観測用カメラの画像を重畳する手法。

**手法 $\beta$**  除去対象を考慮せずに画像4枚(パノラマ画像を構成するカメラ群3枚+隠背景観測用カメラの画像1枚)を用いて画像接合処理のみする手法(図8-6(b))

隠背景の推定方法は文献[57]で述べた手法を用いる。ただしブレンディングは全手法ともにマルチバンドブレンディングを使用した。提案手法の処理フローと計測時間は図9-2に示す。手法 $\alpha$ と手法 $\beta$ も類似の処理フローであり、「投影面に画像とマスクを投影」「マルチバンドブレンディング」はOpenCLを用いた。

パノラマ画像生成時の領域分割はピクセルオーダーの精度は不要である。そこで、本評価で実装したGraph Cutは、入力画像サイズのまま領域分割するのではなく、処理速度を向上させるために、入力画像のアスペクト比は保ちつつ1辺を0.2倍にしたサイズで領域を分割した。また除去対象の近くで分割するために距離変換画像はユークリッド距離を5倍にした値を用いて生成した。

両環境で用いたカメラはCanon EOS 5D MarkIII、レンズはEF24-70mm F4L IS USMであり、解像度はフルHDとした。撮影画像を処理したPCの仕様は表9-1に示す。

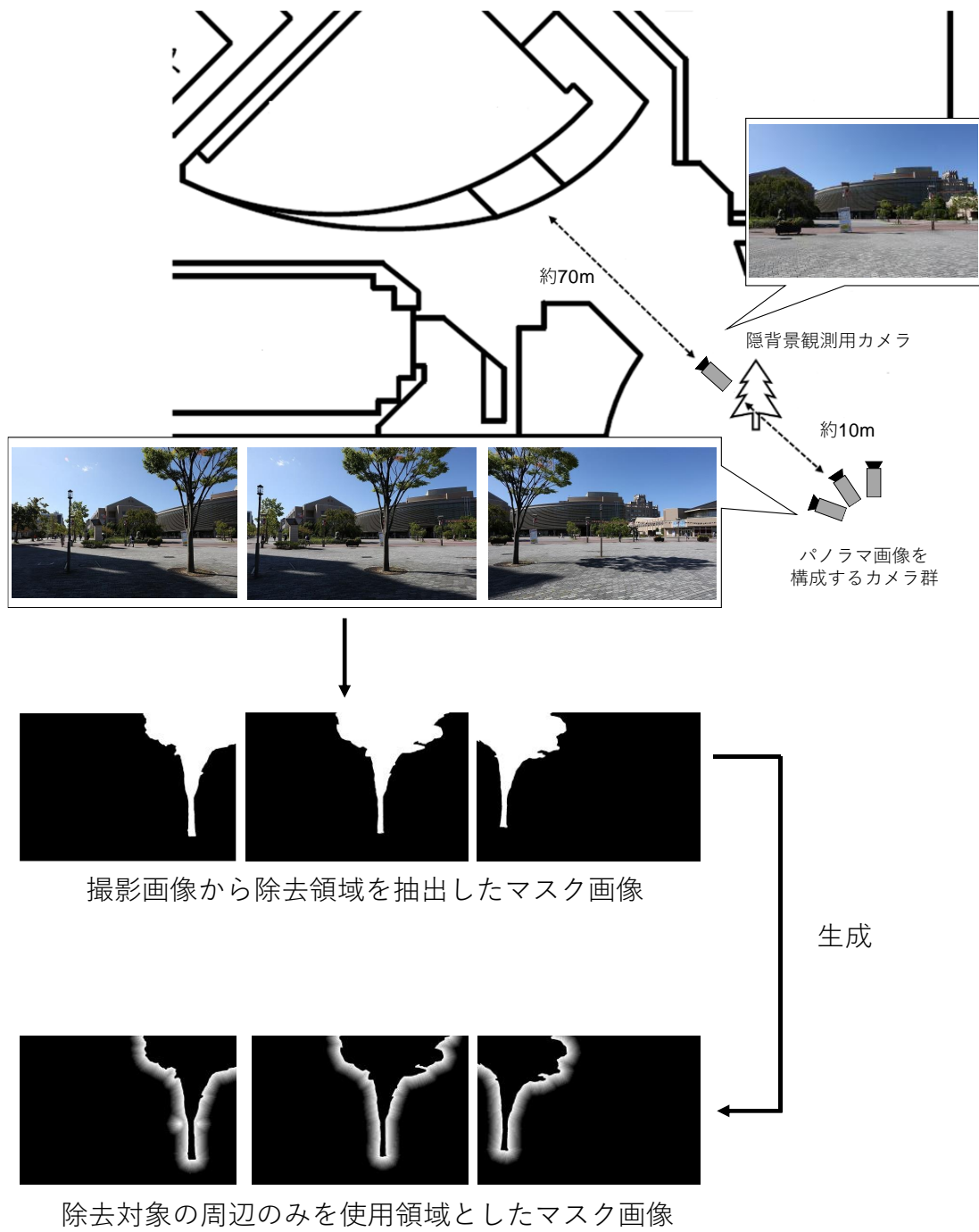


図 9-1 カメラ配置 (環境 A)

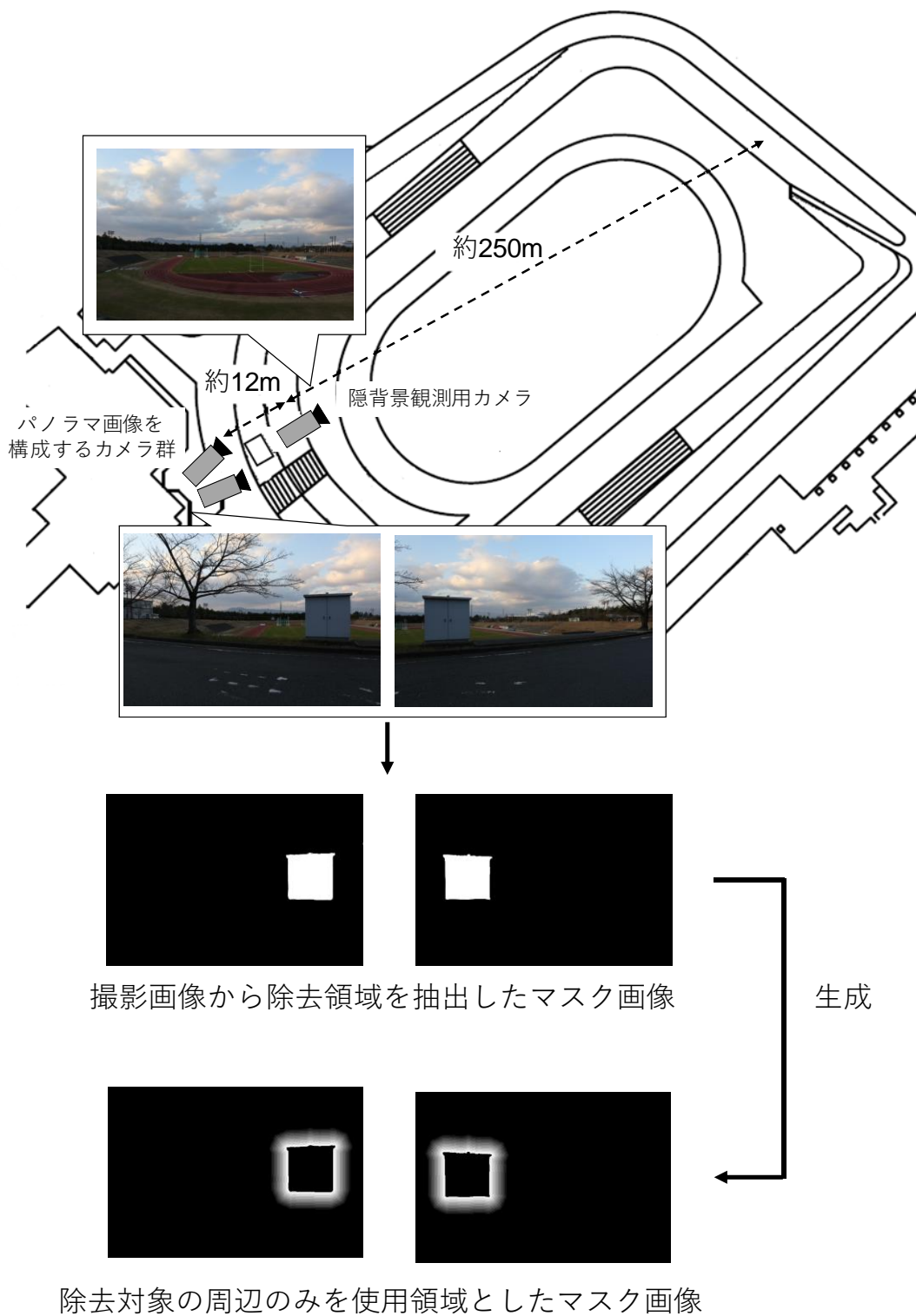


図 9-2 カメラ配置 (環境 B)

## 9.2 画像生成結果

提案手法を用いた実行結果は 9-4 に示す。左図は生成した画像であり、右下図は、各カメラの使用領域を示したものである。両環境ともに違和感が少ないことが分かった。とくに図 9-4(a)は図 8-4 よりも違和感がなく、物体を除去できていることが分かった。図 8-4 ではポアソンブレンディングを用いたが、提案手法では、隠背景を意識的に用意することなく、画像接合処理と同様にブレンディング可能なため、より違和感の少ない生成結果になったと考えられる。また、各撮影画像において除去対象領域は使用していないことから、除去対象の画素がブレンディングに影響を及ぼすこともない。

## 9.3 処理時間比較評価

次に、提案手法と手法  $\alpha$ 、手法  $\beta$  の処理時間を比較する。カメラの位置姿勢は既知かつ不動と仮定しているため、全手法ともにカメラの位置姿勢推定は処理時間から除外し、「境界線の決定」から「ブレンディング」までの時間を計測した。

計測した結果、提案手法による 100 フレームの平均処理時間は 777.84[msec] であり、手法  $\alpha$  は 1,259.5[msec]、手法  $\beta$  は 843.24[msec] であった。提案手法は手法  $\alpha$  より短く、手法  $\beta$  と同等の処理時間であることが分かった (図 9-5)。手法  $\beta$  より提案手法のほうが処理は多いが、Graph Cut の分割領域における探索量がマスク画像によって制限されているため、全体の処理時間は少なくなっている。

境界線を決めるための領域分割は、提案手法と手法  $\beta$  は  $4C^2 = 6$  通りであるが、手法  $\alpha$  は  $3C^2 = 3$  通り+1 通りの計 4 通りのため、手法  $\alpha$  が最も少ない。しかし、ブレンディング処理や、隠背景の推定方法などに時間がかかったため全体的には処理時間が最も大きくなった。

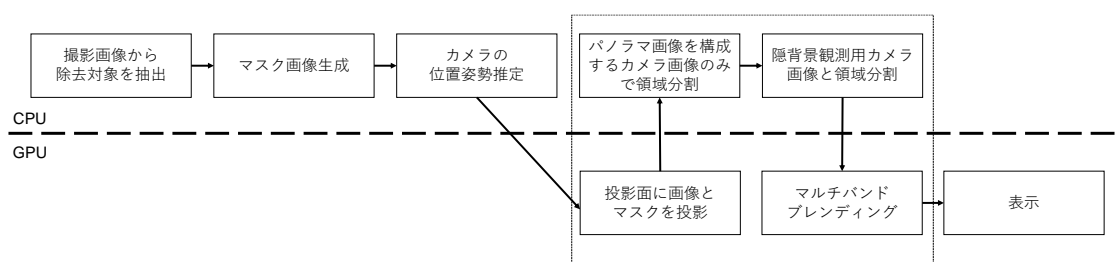


図 9-3 評価時における提案手法の処理フロー

表 9-1 PC の仕様

OS	Windows 10
メモリ	16.0GB
CPU	Intel Core i7-8700 CPU
GPU	NVIDIA GTX 1660 Ti



(a) 環境 A で生成した画像



(b) 環境 B で生成した画像

図 9-4 実行結果

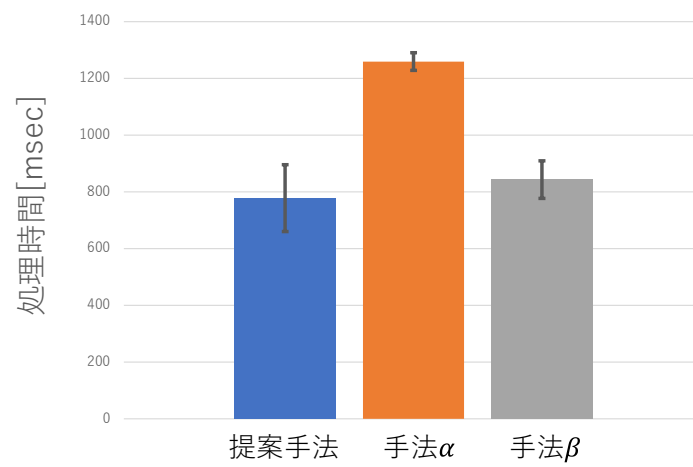


図 9-5 処理速度の比較



## 9.4 まとめ

本章では、視野角によって生じる死角と遮蔽物によって生じる死角を、複数視点のカメラ映像を接合することで同時に解消する画像接合手法を提案し、評価したことについて述べた。死角は 2 種類に大別できるが、多くの場合、それぞれの課題への対応が議論されており、同時に解決する手法は少ない。しかし、各死角への対応策である画像接合技術と DR 技術は、ともに手段として複数視点の画像を接合するため、どちらも本質は変わらない。

本研究では様々な処理フローを検討しながら、両技術の共通する考え方を整理した。その結果、境界線決定時に使用するマスク画像に、遮蔽物を基に生成したマスク画像を生成することで、効率よく広視野画像から遮蔽物を除去できることが分かった。

評価実験において提案手法の有用性は示せたが、実時間性があるとは言えない処理時間であった。本実装では CPU によるマルチスレッド処理に加え、一部、OpenCL を用いた GPU 処理も実装しているが、まだまだ最適な処理になっているとは言えない。したがって、今後は高速化処理についても検討する。

本論文では静止画像に対して遮蔽物を除去した広視野画像の生成について述べた。近年、静止画だけではなく、動画に対する画像接合技術も多く提案されている[62][63]。今後はストーリーミング配信されたカメラ映像への対応も検討する予定である。さらに、より多くの環境下を想定しながら、より最適な処理方法も追求する。

## 第10章 SIGMA フレームワークにおける高度利用

本章では、SIGMA フレームワークを用いて検索した画像を用いて IBMR 処理を行うことで、SIGMA フレームワークの高度利用の例をいくつか示す。

1 つ目はエピポーラ幾何に基づいて検索した結果を用いて広視野画像を生成する。2 つ目は撮影した画像を用いて生成する広視野画像に対して、「エピポーラ幾何に基づく検索」により検索した画像を隠背景に使用することで、任意の物体を画像上から除去する。

両試行とも、検索には、6.2 節で収集した画像を蓄積してあるリソース DB を使用する（図 10-1）。

試行 1 では、水平画角 60 度、解像度 2K で設定した仮想カメラによって抽出された 28 枚のうち、2 枚を用いて広視野画像を生成した（図 10-2）。

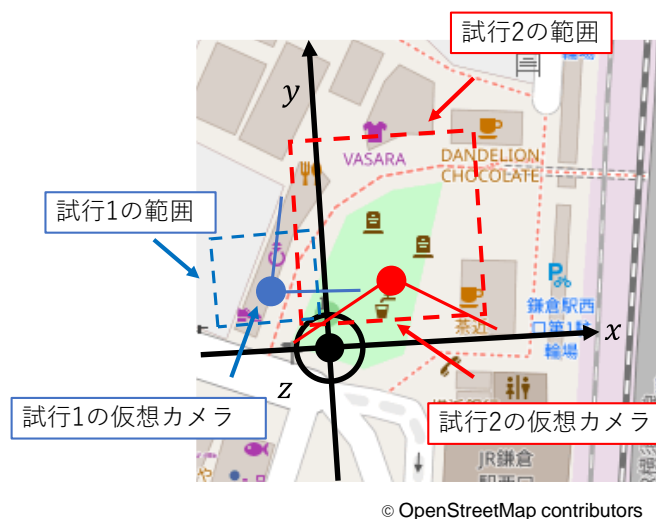


図 10-1 検索範囲と「エピポーラ幾何に基づく検索」の位置



図 10-2 試行 1 の結果

試行 2 では鎌倉駅前で撮影した画像を用いてパノラマ画像（図 10-3）を生成し，8 章で提案した手法を用いて，画像中から物体を除去する。

図 10-3 に示すのは，鎌倉駅前で撮影した 2 枚の画像と，その 2 枚から生成したパノラマ画像である．このパノラマ画像の中央に外灯があり，この外灯を除去するために，リソース DB から隠背景画像を検索する。

検索手法は図 10-1 に示すとおり，6.4 節と同様の検索範囲において，青丸で示す位置から水平画角 120 度，解像度 2K の仮想カメラを設定し，「エピポーラ幾何に基づく検索」を実施した．その結果，42 枚を抽出した．さらに，この 42 枚の中から目視で隠背景画像となる画像を選択した．図 10-4 に示すのは，選択した隠背景画像である。

この隠背景画像を用いて外灯を除去するために，パノラマ画像を構成する 2 枚の画像から除去対象である外灯をマスクした画像を生成した（図 10-5）．これらの画像を用いて外灯を除去した画像を図 10-6 に示す．この結果，SIGMA フレームワークを用いて検索した画像によりパノラマ画像中から外灯を除去することができた．

これらの試行から SIGMA フレームワークにおける高度利用の例を示すことができた．



図 10-3 撮影画像を用いて生成した広視野画像



図 10-4 抽出した隠背景画像

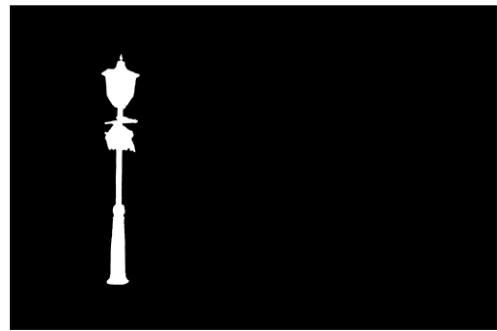
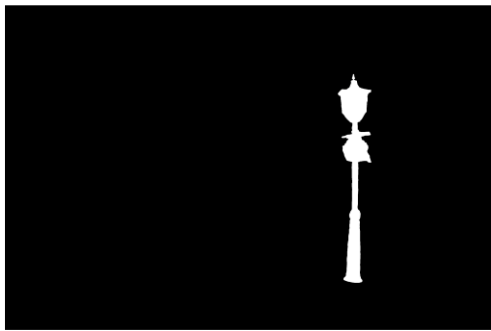
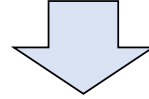
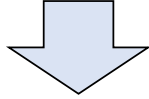


図 10-5 パノラマ画像を構成する画像とそれに対応するマスク画像



図 10-6 外灯を除去した画像

## 第11章 結論

本論文では、時空間映像データの将来的な高度利用を念頭に、実世界を観測した大量のデータを収集・蓄積だけではなく、共有・検索するためのシステム設計について述べた。近い将来、いたるところに設置されたカメラによって、実世界を観測した映像が常時記録される社会が訪れる。自動車の後方に搭載したカメラはバックモニターで見ることができ、衝突防止のために前方にはステレオカメラがついた。そして近年では、サイドミラーがカメラになりつつある。自動車1台を例にとっても日々、搭載するカメラが増加していることがわかる。他にも、街中の監視カメラの数は増加の一途をたどり、駅改札は顔認証による切符レスの実証実験が行われている。こういった動きを後押しするようにスーパーシティ構想が掲げられており、この流れが止まることはないだろう。現在は、取得した画像はクローズなことが多いが、プライバシーの問題をクリアすればオープン化の機運はさらに高まると予想できる。そうってから、システムを設計しては手遅れになるため、本研究グループでは、未来を見据え、本システムの設計に取り組んだ。

大きな構想を持って取り組んだ本研究テーマではあるが、未来を見据えてシステムを構築することは難しく、何度も議論を重ねながら、具体化と抽象化を繰り返し、実現できるものに落とし込む作業を続けた。こうした中で、SIGMA フレームワークと命名した本システムにおいて、最も核となる機構のひとつとしてリソース DB に蓄積されたデータから所望のデータを検索する仕組みについて議論を深めた。

ただ、一概にデータの検索と言っても、データを取得するセンサは多種多様であり、各センサの特徴は千差万別なため、同じように扱うことはできない。そこで本研究では、カメラで取得した画像を扱うと、具体的な目標を設定した。ただし、SIGMA フレームワークが画像に特化したものというわけではなく、あくまで足掛かりとして、画像を選択した。同様に、検索するためには、画像の利用が念頭になる。つまり、利用方法が明確にしなければ検索手法を検討することもできない。そこで手始めに IBMR を選択した。

このように、初めに描いたグラウンドデザインをどうやって具体化するのか、を試行錯誤しながら進めた。ひとつ案が浮かぶと、その案を達成するためには様々な課題があることがわかる。このような状況下においてひとつずつ丁寧に議論したものが本論文である。同時に、システム設計だけでは最終到達地点を明示することが難しい。そこで、SIGMA フレームワークの高度利用について検討した。高度利用と一言で言っても、高度とは何を指すのか、という疑問にぶつかる。そこで本研究室が培った DR 技術を、今までにないアプローチで利用する術を熟考した。その結果、パノラマ画像との親和性が高いことがわかり、これを組み合わせることによって新たな手法を提案するに至った。

このように、様々な課題を解決してきたが、議論すればするほど開発項目が増加する本テーマは、次のフェーズに移行する必要がある。今後は、他の高度利用方法を念頭に、時空間映像データの有効な共有・検索方法を検討するとともに、センサの種類や、観測データの活用をより広

汎にしていくとともに、アプリケーションフレームワークとしての時空間映像データの高度利用のための機能を拡充する。また、アプリ開発を向上させるアプリケーションフレームワークの完成度を高めるため、開発した **SIGMA** フレームワークを利用したアプリの事例を増やし、知見を貯める必要がある。

## 謝辞

本研究の全過程を通じて、御指導を頂いた立命館大学大学院情報理工学研究科 柴田史久教授に深く感謝申し上げます。

博士学位審議にあたり副査をお受けいただきました立命館大学大学院情報理工学研究科 木村朝子教授，李周浩教授には大変有意義なご意見を頂きました。深く感謝いたします。本研究に関し，貴重なご意見・ご指導を戴いた立命館大学総合科学技術研究機構 田村秀行教授には厚くご御礼申し上げます。

本研究の設計・実装に関して協力を頂いた研究グループの林光希氏，有富友紀氏，关斯琨氏，松木輝氏，三菱電機（株）情報技術総合研究所 岡原浩平氏に感謝する。

筆者の所属する三菱電機（株）松岡文啓氏，宮原浩二氏，根岸博康氏，杉本和夫氏，井須芳美氏には格別のご配慮を賜り，良好な研究環境を与えていただいたこと感謝申し上げます。

最後に，モバイルコンピューティング研究室，並びにリアリタイムメディア研究室の諸氏には，本研究を進める上で多岐にわたりお世話になった。ここに感謝の意を表する。

## 参考文献

- [1] 国土交通省： Project “PLATEAU”， <https://www.mlit.go.jp/plateau/>.
- [2] 佐藤洋一： イメージベースドモデリング，計測と制御，Vol. 41, No. 9, pp. 633–638 (2002).
- [3] Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M. and Szeliski, R.: Building rome in a day, Communications of the ACM, Vol. 54, No. 10, pp. 105–112 (2011).
- [4] Juhász, L. and Hochmair, H. H.: Cross-linkage between Mapillary street level photos and OSM edits, Geospatial Data in a Changing World, Springer, pp. 141–156 (2016).
- [5] Suzuki, K., Wakisaka, S. and Fujii, N.: Substitutional reality system: a novel experimental platform for experiencing alternative reality, Scientific reports, Vol. 2, p. 459 (2012).
- [6] 池内克史, 大石岳史, 小野晋太郎, 岡本泰英, 鎌倉真音：まちと震災のいま・過去を「仮想化空間」で伝える，生産研究， Vol. 68, No. 2, pp. 123-126 (2016).
- [7] 平松黎, 有富友紀, 若林優, 木村朝子, 柴田史久：安全運転支援のための周辺車両の半隠消表示法(1) - 車車間通信モジュールの設計と実装 -, 第 25 回日本バーチャルリアリティ学会大会論文集, 3B1-3 (2020).
- [8] 山崎賢人, 有富友紀, 关斯琨, 木村朝子, 柴田史久：実世界観測による時空間映像データの高度利用(1) - 基本アーキテクチャの概念設計と第 1 次システム試作 -, 日本バーチャルリアリティ学会複合現実感研究会, MR2019-15, 日本バーチャルリアリティ学会研究報告, Vol. 22, No. 2, pp. 49–53 (2019).
- [9] Ori, I.: Mirrorworld v. AR Cloud or: How I Learned to Stop Worrying and Love the Spatial Future, Super Ventures Blog (online), available from (<https://medium.com/super-venturesblog/mirrorworld-v-ar-cloud-orhow-i-learned-to-stopworrying-and-love-the-spatial-future-59de8fe8538f>)(accessed 2020-01-05).
- [10] 石丸伸裕：3次元地理空間データ CityGML/IndoorGML に関する国際標準化活動，地図，Vol. 52, No. 3, pp. 29-36 (2014).
- [11] Manandhar, D. and Shibasaki, R.: Vehicle-borne laser mapping system (VLMS) for 3-D GIS, IGARSS 2001. Scanning the Present and Resolving the Future. Proceedings. IEEE 2001 International Geoscience and Remote Sensing Symposium (Cat. No. 01CH37217), Vol. 5, IEEE, pp. 2073–2075 (2001).
- [12] Sakurada, K., Okatani, T. and Deguchi, K.: Detecting changes in 3D structure of a scene from multi-view images captured by a vehicle-mounted camera, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 137–144 (2013).
- [13] Furukawa, Y. and Ponce, J.: Accurate, dense, and robust multi-view stereopsis, IEEE transactions on



- pattern analysis and machine intelligence, Vol. 32, No. 8, pp. 1362–1376 (2009).
- [14] 川住涼, 義久智樹, 原隆浩, 西尾章治郎: 複数拠点統合型センサネットワークにおけるセンシング情報を考慮した時空間インデックス構築手法, 情報処理学会研究報告データベースシステム(DBS), Vol. 2014, No. 6, pp. 1–6 (2014).
- [15] 柴田史久, 松田祐樹, 川端大輔, 山崎賢人, 木村朝子: モバイル MR システム構築のための機能分散型フレームワークの設計と実装, 日本バーチャルリアリティ学会論文誌, Vol. 19, No. 2, pp. 215–225 (2014).
- [16] 山崎賢人, 琨关斯, 松木輝, 木村朝子, 柴田史久: SIGMA フレームワークにおける時空間映像データの管理手法, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), Vol. 2020-DCC-26, No. 9, pp. 1–6 (2020).
- [17] 国土交通省 国土地理院: わかりやすい平面直角座標系,  
<https://www.gsi.go.jp/sokuchikijun/jpc.html>
- [18] 国土交通省 国土地理院: 測量に関するミニ知識 第10回 UTM グリッド地図 その2,  
<https://www.gsi.go.jp/chubu/minichishiki10.html>
- [19] Zhang, Z.: A flexible new technique for camera calibration, IEEE Transactions on pattern analysis and machine intelligence, Vol. 22, No. 11, pp. 1330–1334 (2000).
- [20] Scaramuzza, D., Martinelli, A. and Siegwart, R.: A toolbox for easily calibrating omnidirectional cameras, 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, pp. 5695–5701 (2006).
- [21] Morton, G. M.: A computer oriented geodetic data base and a new technique in file sequencing (1966).
- [22] 小川輝樹, 伊藤嘉博, 中村嘉隆, 高橋修, 白石陽: 空間統計処理のための Geohash を用いたデータベースシステムの提案と評価(2012).
- [23] Y. Xia, M. Huang, X. Chen, and W. Li: A region of interest extraction approach of on-Road object, in Proceedings of 2009 WRI World Congress on Computer Science and Information Engineering, vol. 6, pp. 519–522, 2009.
- [24] Matsuki, A., Aritomi, T., Yamazaki, K., Kimura, A. and Shibata, F.: Design and Implementation of Data Communication and Compression Methods in SIGMA Framework, 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE), pp. 599–603 (2020).
- [25] Bankoski, J., Wilkins, P. and Xu, Y.: Technical overview of VP8, an open source video codec for the web, 2011 IEEE International Conference on Multimedia and Expo, IEEE, pp. 1–6 (2011).
- [26] Schonberger, J. L. and Frahm, J.-M.: Structure-from-motion revisited, Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 4104–4113 (2016).
- [27] Agarwal, S., Snavely, N., Seitz, S. M. and Szeliski, R.: Bundle adjustment in the large, European conference on computer vision, Springer, pp. 29–42 (2010).
- [28] Seitz, S. M., Curless, B., Diebel, J., Scharstein, D. and Szeliski, R.: A comparison and evaluation of multiview stereo reconstruction algorithms, 2006 IEEE computer society conference on computer

- vision and pattern recognition (CVPR'06), Vol. 1, IEEE, pp. 519–528 (2006).
- [29] Crandall, D., Owens, A., Snavely, N. and Huttenlocher, D.: Discrete-continuous optimization for largescale structure from motion, CVPR 2011, IEEE, pp. 3001–3008 (2011).
- [30] Taketomi, T., Uchiyama, H. and Ikeda, S.: Visual SLAM algorithms: a survey from 2010 to 2016, IPSJ Transactions on Computer Vision and Applications, Vol. 9, No. 1, pp. 1–11 (2017).
- [31] 中野一也 : 解説: Structure from Motion (SfM) 第三回 SfM による三次元形状復元ソフトウェアについて, 写真測量とリモートセンシング, Vol. 55, No. 5, pp. 321-324 (2016).
- [32] 酒井和彦 : 世界初アラウンドビューモニター, 自動車技術 = Journal of Society of Automotive Engineers of Japan, Vol. 62, No. 3, pp. 100-101 (2008).
- [33] Brown, M. and Lowe, D. G.: Automatic panoramic image stitching using invariant features, International Journal of Computer Vision (IJCV), Vol. 74, No. 1, pp. 59–73 (2007).
- [34] Gortler, S. J., Grzeszczuk, R., Szeliski, R. and Cohen, M. F.: The lumigraph, Proceedings of the 23rd annual conference on Computer graphics and interactive techniques, pp. 43–54 (1996).
- [35] Mori, S., Ikeda, S. and Saito, H.: A survey of diminished reality: Techniques for visually concealing, eliminating, and seeing through real objects, IPSJ Transactions on Computer Vision and Applications, Vol. 9, No. 1, pp. 1–14 (2017).
- [36] Schindler, G., Dellaert, F. and Kang, S. B.: Inferring temporal order of images from 3D structure, 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp. 1–7 (2007).
- [37] Lowe, D. G.: Distinctive image features from scaleinvariant keypoints, International journal of computer vision, Vol. 60, No. 2, pp. 91–110 (2004).
- [38] Bleyer, M., Rhemann, C. and Rother, C.: PatchMatch Stereo - Stereo Matching with Slanted Support Windows, BMVC (2011).
- [39] Assarsson, U. and Moller, T.: Optimized view frustum culling algorithms for bounding boxes, Journal of graphics tools, Vol. 5, No. 1, pp. 9–22 (2000).
- [40] Rusu, R. B. and Cousins, S.: 3d is here: Point cloud library (pcl), 2011 IEEE international conference on robotics and automation, IEEE, pp. 1–4 (2011).
- [41] OpenStreetMap Foundation: OpenStreetMap, <https://www.openstreetmap.org>.
- [42] Kato, H. and Billinghurst, M.: Marker tracking and hmd calibration for a video-based augmented reality conferencing system, Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99), IEEE, pp. 85–94 (1999).
- [43] Tamura, H., Yamamoto, H. and Katayama, A.: Mixed reality: Future dreams seen at the border between real and virtual worlds, IEEE Computer Graphics and Applications, Vol.

- 21, No. 6, pp. 64–70 (2001).
- [44] Schönberger, J. L. and Frahm, J.-M.: Structure-from-Motion Revisited, Conference on Computer Vision and Pattern Recognition (CVPR) (2016).
- [45] Sweeney, C.: Theia Multiview Geometry Library: Tutorial & Reference, <http://theia-sfm.org>.
- [46] Moulon, P., Monasse, P., Marlet, R. and Others: OpenMVG, <https://github.com/openMVG/openMVG>.
- [47] Cernea, D. OpenMVS: Open Multiple View Stereovision. <https://github.com/cdcseacave/openMVS/>
- [48] Okahara, K., Fukasawa, T., Furuki, I. and Abe, H.: Efficient implementation of top-view surveillance system using multiple cameras, IEEE Image Electronics and Visual Computing Workshop 2017 (IEVC 2017) (2017).
- [49] Rameau, F., Ha, H., Joo, K., Choi, J., Park, K. and Kweon, I.S.: A real-time augmented reality system to see-through cars, IEEE Trans. Visualization and Computer Graphics (TVCG), Vol.22, No.11, pp.2395–2404 (2016).
- [50] Kido, D., Fukuda, T. and Yabuki, N.: Diminished reality system with real-time object detection using deep learning for onsite landscape simulation during redevelopment, Environmental Modelling & Software, Vol. 131, p. 104759 (2020).
- [51] Mori, S., Eguchi, Y., Ikeda, S., Shibata, F., Kimura, A., and Tamura, H.: Design and construction of data acquisition facilities for diminished reality research, ITE Trans. on Media Technology and Applications (MTA), Vol. 4, No. 3, pp. 259 - 268 (2016).
- [52] Telea, A.: An image inpainting technique based on the fast marching method, Journal of graphics tools, Vol. 9, No. 1, pp. 23–34 (2004).
- [53] 森尚平, 一刈良介, 柴田史久, 木村朝子, 田村秀行: 隠消現実感の技術的枠組と諸問題: 現実世界に実在する物体を視覚的に隠蔽・消去・透視する技術について, 日本バーチャルリアリティ学会論文誌, Vol.16, No.2, pp.239-250 (2011).
- [54] Kwatra, V., Schödl, A., Essa, I., Turk, G. and Bobick, A.: Graphcut textures: Image and video synthesis using graph cuts, ACM Trans. Graphics (TOG), Vol.22, No.3, pp.277–286 (2003).
- [55] Burt, P.J. and Adelson, E.H.: A multiresolution spline with application to image mosaics, ACM Trans. Graphics (TOG), Vol.2, No.4, pp.217–236 (1983).
- [56] Hasegawa, K. and Saito, H.: Diminished reality for hiding a pedestrian using hand-held camera, 2015 IEEE International Symposium on Mixed and Augmented Reality Workshops, pp.47–52, IEEE (2015).
- [57] 山崎賢人, 岡原浩平, 木村朝子, 柴田史久: 画像接合型広視野映像に対する隠消現実達成法の検討, 情報処理学会研究報告デジタルコンテンツクリエーション (DCC), Vol.2020-DCC-

25, No.3, pp.1-5 (2020).

- [58] P´erez, P., Gangnet, M. and Blake, A.: Poisson image editing, ACM SIGGRAPH 2003 Papers, pp.313–318 (2003).
- [59] Summa, B., Tierny, J. and Pascucci, V.: Panorama weaving: Fast and flexible seam processing, ACM Trans. Graphics (TOG), Vol.31, No.4, pp.1–11 (2012).
- [60] Borgefors, G.: Distance transformations in digital images, Computer Vision, Graphics, and Image Processing, Vol.34, No.3, pp.344–371 (1986).
- [61] Rother, C., Kolmogorov, V. and Blake, A.: Interactive foreground extraction using iterated graph cuts, ACM Trans. Graphics (TOG), Vol.23, p.3 (2012).
- [62] Perazzi, F., Sorkine-Hornung, A., Zimmer, H., Kaufmann, P., Wang, O., Watson, S. and Gross, M.: Panoramic video from unstructured camera arrays, Computer Graphics Forum, Vol.34, No.2, pp.57–68, Wiley Online Library (2015).
- [63] Lee, J., Kim, B., Kim, K., Kim, Y. and Noh, J.:Rich360: optimized spherical representation from structured panoramic camera arrays, ACM Trans. Graphics (TOG), Vol.35, No.4, pp.1–11 (2016).

## 研究業績

### 本研究に関する主要論文

1. 山崎賢人, 岡原浩平, 木村朝子, 柴田史久: 画像接合技術に基づく広視野画像のための隠消現実感達成法, 情報処理学会論文誌デジタルコンテンツ, Vol. 9, No. 1, pp. 34 - 42 (2021).
2. 山崎賢人, 关斯琨, 松木輝, 木村朝子, 柴田史久 SIGMA Retriever: イメージベースドモデリング・レンダリングのための検索機構の設計と実装, 情報処理学会論文誌デジタルコンテンツ, Vol. 10, No. 1 (in press)

### 本研究に関する口頭発表等

1. 山崎賢人, 有富友紀, 关斯琨, 木村朝子, 柴田史久: 実世界観測による時空間映像データの高度利用 (1) —基本アーキテクチャの概念設計と第1次システム試作—, 日本バーチャルリアリティ学会複合現実感研究会, MR2019-15, 日本バーチャルリアリティ学会研究報告, Vol. 22, No. 2, pp. 49 - 53 (2019).
2. 有富友紀, 山崎賢人, 关斯琨, 木村朝子, 柴田史久: 実世界観測による時空間映像データの高度利用 (2) —位置姿勢推定機構と3次元再構成機構の機能設計—, 2020年電子情報通信学会総合大会講演論文集, H-3-6, p. 215 (2020).
3. 山崎賢人, 岡原浩平, 木村朝子, 柴田史久: 画像接合型広視野映像に対する隠消現実感達成法の検討, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), Vol. 2020-DCC-25, No. 3, pp. 1-5 (2020).
4. Kento Yamazaki, Tomonori Aritomi, Sikun Guan, Asako Kimura, and Fumihisa Shibata: The SIGMA framework for managing spatiotemporal data acquired by various cameras and sensors, Proc. 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE 2020), pp. 879 - 883 (2020).
5. Akira Matsuki, Tomonori Aritomi, Kento Yamazaki, Asako Kimura, and Fumihisa Shibata: Design and implementation of data communication and compression methods in SIGMA framework, Proc. 2020 IEEE 9th Global Conference on Consumer Electronics (GCCE 2020), pp. 884 - 888 (2020).
6. 山崎賢人, 关斯琨, 松木輝, 木村朝子, 柴田史久: SIGMA フレームワークにおける時空間映像データの管理手法, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), Vol. 2020-DCC-26, No. 9, pp. 1-6 (2020).
7. 西川卓, 松木輝, 山崎賢人, 木村朝子, 柴田史久: 実世界観測による時空間映像データの高度利用 (3) —蓄積と配信を並列させた低遅延データ配信機構—, 日本バーチャルリアリティ学会複合現実感研究会, MR2022-8, 日本バーチャルリアリティ学会研究報告, Vol. 25, No. 1 (2022).

## その他

### 学術論文

1. 柴田史久, 松田祐樹, レヴァンギア, 川端大輔, 山崎賢人, 木村朝子: モバイル MR システム構築のための機能分散型フレームワークの設計と実装, 日本バーチャルリアリティ学会論文誌, Vol. 19, No. 2, pp. 215 - 225 (2014).
2. 山崎賢人, 柴田史久, 木村朝子, 田村秀行: 複合現実感技術を用いた商品物流における仕分け作業支援, 日本バーチャルリアリティ学会論文誌, Vol. 19, No. 3, pp. 413 - 422 (2014).
3. 山崎賢人, 阿倍博信: プロジェクション型 AR における遮蔽を考慮した注釈ビューマネジメントの設計と評価, 情報処理学会論文誌デジタルコンテンツ, Vol. 7, No. 1, pp. 11 - 21 (2019).
4. 鈴木遼雅, 岡原浩平, 山崎賢人, 都丸義広, 平謙二: 測位と撮像の組み合わせに基づく道路舗装面の 3 次元計測方式の評価, 写真測量とリモートセンシング, Vol. 60, No. 3, pp. 121 - 128 (2021).

### 口頭発表等

1. Kento Yamazaki, Fumihisa Shibata, Asako Kimura, and Hideyuki Tamura: Prototype development of a mixed reality order picking system for warehouse storage, Proc. Innovations in Information and Communication Science and Technology (IICST 2013), pp.15 - 24 (2013).
2. 山崎賢人, 柴田史久, 木村朝子, 田村秀行: 商品物流における仕分け作業支援への複合現実感技術の応用, 第 18 回日本バーチャルリアリティ学会大会論文集, 13B-5, pp. 196 - 199 (2013).
3. 橋本明解, 山崎賢人, 柴田史久, 木村朝子, 田村秀行: 商品物流における仕分け作業支援への複合現実感技術の応用(2) -HMD 利用からプロジェクタ利用への移行可能性の検討-, 第 19 回日本バーチャルリアリティ学会大会論文集, 33B-6, pp. 553 - 556 (2014).
4. Tsubasa Ogino, Yuki Matsuda, Le Van Nghia, Daisuke Kawabata, Kento Yamazaki, Asako Kimura, and Fumihisa Shibata: A distributed framework for creating mobile mixed reality systems, Proc. IEEE 3rd Global Conf. on Consumer Electronics (GCCE 2014), pp. 327 - 331 (2014).
5. 山崎賢人, 柴田史久, 木村朝子, 田村秀行: 商品物流における仕分け作業への複合現実感技術の応用 (3) -光学シースルー方式を前提とした注釈情報提示法の検討-, 日本バーチャルリアリティ学会複合現実感研究会, MR2015-6, 信学技報, Vol. 114, No. 409, pp. 193 - 198 (2015).
6. 山崎賢人, 阿倍博信: プロジェクション型 AR における遮蔽を考慮した注釈ビューマネジメントの提案, 第 22 回日本バーチャルリアリティ学会大会論文集, 3E1-01, pp. 1 - 4 (2017).

7. 山崎賢人, 阿倍博信: 力学モデルに基づく注釈ビューマネジメントを用いたプロジェクト型 AR システムの開発 : 2018-DCC-18, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), No. 18, pp. 1-7 (2018).
8. 坂田真太郎, 水野慎士, 山崎賢人, 岡原浩平, 皆川 純: データの効率的生成・表示のための低・高解像度 3D スキャンシステムの提案 : 2019-DCC-21, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), No. 51, pp. 1-6 (2019).
9. 山崎賢人, 阿倍博信: 身体的特徴を活かした複数視点モーショントラッキング・システムの検討 : 2019-DCC-21, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), No. 52, pp. 1-5 (2019).
10. 皆川純, 岡原浩平, 山崎賢人, 深澤司: カメラ位置姿勢変化に伴う俯瞰合成映像の位置ずれ補正手法の提案, 情報処理学会第 81 回全国大会講演論文集, pp. 27-28 (2019).
11. Jun Minagawa, Kohei Okahara, Kento Yamazaki, Tsukasa Fukasawa: A camera recalibration method for a top-view surveillance system based on relative camera pose and structural similarity, Proc. 2019 16th IEEE Int'l Conf. on Advanced Video and Signal Based Surveillance (AVSS) (2019).
12. 岡原浩平, 深澤司, 皆川純, 山崎賢人: 複数カメラ俯瞰映像合成システムと PTZ カメラの連携方式に関する一検討: FIT 情報科学技術フォーラム, H-001 (2020).
13. 岡原浩平, 皆川純, 山崎賢人, 深澤司: 複数カメラ俯瞰映像合成システムにおけるカメラ再校正手法の評価: FIT 情報科学技術フォーラム, H-002 (2020).
14. 久米孝, 皆川純, 山崎賢人, 阿倍博信: 監視カメラ映像における 3 次元顔再構成のための顔画像スコアリング方式の提案, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), Vol. 2020-DCC-27, No. 17, pp. 1-5 (2021).
15. 久米孝, 皆川純, 山崎賢人, 阿倍博信: 映像監視システムにおけるベストショット予測のための顔画像のスコアリング方式, 情報処理学会第 83 回全国大会講演論文集, 7T-03 (2021).
16. 久米孝, 皆川純, 山崎賢人, 阿倍博信: 映像監視システムにおける 3 次元姿勢推定に基づくベストショット抽出のための顔画像のスコアリング方式, 情報処理学会研究報告デジタルコンテンツクリエーション(DCC), Vol. 2020-DCC-30, No. 22, pp. 1-8 (2022).