東京大学
情報理工学系研究科 創造情報学専攻
修士論文

工場・ビル管理における BIM に基づくデジタルツイン基盤の研究

BIM-based digital twin platform for factory and building management

川村 地平
Chihei Kawamura

指導教員　塚田 学 准教授

2022 年 1 月

概要

昨今のコロナ禍では、IoT(Internet of Things) 化の進行による影響も相まって、工場やビルなど、現実世界での施設を遠隔で管理することが多く望まれている。このような IoT デバイスなどの先進技術を用いて、工場やビルの稼働効率向上や管理コスト削減を実現することをスマートファクトリーやスマートビルディングと呼ぶ。これら施設の管理には、工作機械やサーバなどと言った施設内の主要設備に限らず、空調、照明などを始めとした環境操作設備なども必要とされている。そして、施設の管理には当然施設の状態の可視化も含まれており、これには工作機械やサーバ、ロボットなどの状態などに加え、流動的な要素である人の情報なども取れることが望ましく、現実世界での各種要素を管理するため、オブジェクトとして仮想空間に反映できることが求められている。これらのニーズを満たす基盤はデジタルツインと呼ばれる。我々は、このデジタルツインを活用し、特定のデジタルツインアーキテクチャに基づいたデジタルツインアプリケーションを構築した。本論文では、工場とビルという二つの異なる基盤に基づいて、それぞれ課題点を定めた上で、その課題点を解決するため、このデジタルツインアプリケーションを用い、実証実験と検証を行った。それぞれの基盤について、具体的な課題点として以下の二つを挙げた。まず工場では、施設の状態監視を主目的とし、工作機械の三次元ジオメトリ、電力消費や稼働状況、周辺の環境情報などを含めた各種情報を、三次元ジオメトリを用い仮想空間上に再現した工場内にて、立体的に再現することで直感的な工場施設の可視化を行うという課題を設定した。そして、ビルでは、運搬ロボットや照明、空調設備を操作するためのシステムを構築し、BIM(Building Information Modeling) や LiDAR(Light Detection And Ranging) といった高精度な環境情報を扱うことでロボットの行動計画をデジタルツインアプリケーション上から生成することを課題とし、これを解決した。

Abstract

In recent COVID-19 pandemic, combined with the impact of the ongoing shift to the IoT (Internet of Things), there is much desire to remotely manage facilities in the real world, such as factories and buildings. The use of such advanced technologies such as IoT devices to improve the operating efficiency and reduce the management costs of factories and buildings is called smart factories and smart buildings. The management of these facilities requires not only the main equipment in the facility, such as machine tools and servers, but also the environmental operation equipment, such as air conditioning and lighting. In addition to the status of machine tools, servers, robots, etc., it is desirable to be able to obtain information on people, which is a fluid element, and to be able to reflect this information as an object in the virtual space in order to manage various elements in the real world. The infrastructure that satisfies these needs is called the Digital Twin. We have utilized this digital twin to build a digital twin application based on the digital twin architecture. In this paper, based on two different infrastructures, a factory and a building, we have defined the issues to be addressed, and then used this digital twin application to solve these issues through demonstration and verification. For each infrastructure, the following two specific issues were identified. For the factory, the main objective was to monitor the condition of the facility. The task was to visualize the factory facility intuitively by reproducing various information including the 3D geometry of the machine tools, power consumption and operation status, and environmental information of the surrounding area in 3D in a factory reproduced in a virtual space using 3D geometry. And in the building, we built a system to operate the transport robots, lighting and air conditioning systems. Here, we tried to solve the problem of generating action plans for robots from digital twin applications by handling highly accurate environmental information such as BIM (Building Information Modeling) and LiDAR (Light Detection And Ranging).

目次

第 1 章　序論 1
1.1　はじめに 1
1.2　本研究について 2
1.3　本論文の構成 3

第 2 章　関連技術・研究, 使用技術 4
2.1　デジタルツイン 4
2.2　デジタルツインアーキテクチャ 5
2.3　スマートファクトリー 5
2.4　スマートビルディング 6
2.5　ジオメトリデータ生成 7

第 3 章　デジタルツインアーキテクチャ 8

第 4 章　スマートファクトリー 9
4.1　背景 9
4.2　提案手法 9
4.3　設計と実装 10
4.4　評価 16
4.5　課題 21

第 5 章　スマートビルディング 23
5.1　背景 23
5.2　提案手法 23
5.3　設計と実装 25
5.4　本アプリケーションを用いた実証実験 - ケーススタディ 32
5.5　実験 32
5.6　評価 33
5.7　今後の課題 36

第 6 章　評価、結論 38

発表文献と研究活動 39

参考文献 40

付録 A 工場可視化ツールの使用手順を説明したプレゼンテーション資料 45

付録 B スマートビルディングにおけるデジタルツインアプリケーションの各種機能説明 49

第 1 章

序論

1.1 はじめに

昨今, 急速に広まった COVID-19 パンデミックやデジタル技術の促進により,DX(Digital transformation) が注目され始めている．DX とは, 情報技術の浸透が, 人々の生活をより良い方向へ変革するといった仮説である [1]．この仮説では,DX により, 情報技術を現実が徐々に融合し結びつくと言った変化や人間やロボットなどのオブジェクトが環境や行動の変化についてネットワークを経由してシステムに知らせることが出来るようになるということを示唆している．日本でも, この DX について, 経済産業省が,DX 推進ガイドライン [2] というガイドラインを発行し, 各企業に対し DX 化を推進している．

この DX 化の一つとして, いくつかの企業が現状の課題解決策として行っていることがデジタルツインの構築である．このデジタルツインは, 昨今の IoT(Internet of Things) 化の進行と合わせ, 注目が高まってきた． デジタルツインとは, 図 1.1 に示したように, 現実空間の設備と同じものを仮想空間に構築し, 現実空間の設備からデータを取得しながら仮想空間でシミュレーションを行うことでリアルタイムに現実空間と仮想空間を連携させるといったことを指す言葉である． また, これに付随して, 図 1.1 の中央に位置するコモングラウンドといった言葉がある．これは, 現実空間と仮想空間を結ぶ基盤のことで, ハードウェアとその背後にあるソフトウェアを含めたインフラストラクチャーを指す．デジタルツインは, 製造工程の最適化のために考案された [3][4] が, 最近では IoT やビッグデータなど複数の要素技術の発展により,Industrial Internet of Things(IIoT) の形成技術として [5], ヘルスケアやスマートシティなど他の分野でも注目され始めており [6][7][8], 施設管理や都市計画, 自律走行車試験など様々な用途に用いられている．また, ドイツ政府が提唱している次世代の産業革命の基盤という Industry4．0[9] の CPS(Cyber Physical System) のコンポーネントとしても注目され始めている． なお, ここで,CPS とは, アメリカ国立科学財団によると, 計算機資源と物理資源の密接な結合, 協調を行っているシステムであり [10], その基本要素は現実空間のセンシングとコンピューティング, それに基づいたアクチュエーションである．これは, 現実世界の機器の状況をセンサなどでデータ化し, 仮想空間上にデジタルツインとして再現, 分析し活用できるようにするシステムとして考えられる．

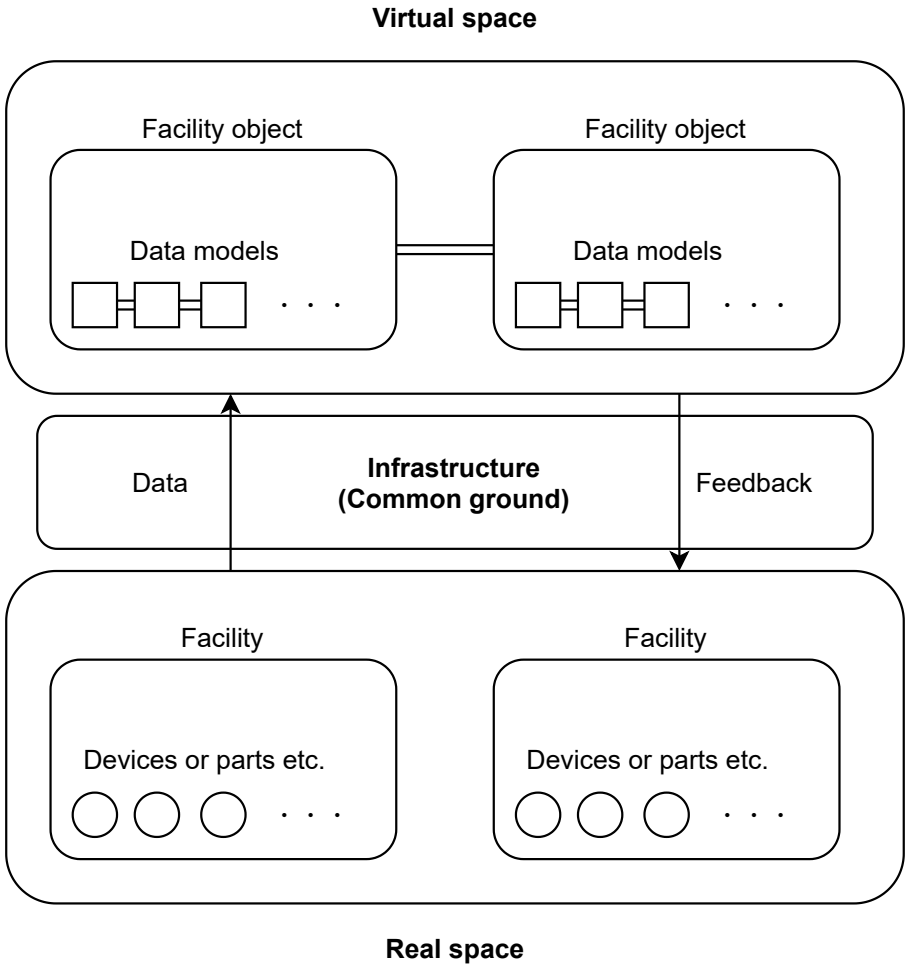


図 1.1. デジタルツイン概要図

1.2　本研究について

デジタルツインは, 現実空間の物体の情報を仮想空間でオブジェクトとして表現できれば, 対象の施設の設備の統括的管理, 照明や換気システムの最適化, ロボットの行動計画のシミュレーションなど, 様々な応用可能性がある．なお, ここでのオブジェクトとは,IoT デバイスなどと言った施設の物のことを指し, 場合によっては人などの仮想空間に取り込みにくい物も対象になる．しかし, デジタルツインは応用可能な範囲が広く, 領域や目的に応じて, 個々のデジタルツインの統合やオブジェクトの決定, 抽象化が難しいと言った課題がある．本研究では, スマートファクトリーとスマートビルディングの領域について実証実験を行い, それぞれが保有する課題を解決していくとともに, これらのデジタルツインがデジタルツイン基盤へ体系的に対応付けできるものなのかを考えていく． なお, この実証実験にて用いるデジタルツインアーキテクチャは,Shohin Aheleroff らが, 提案した製造工程に限らない汎用的なデジタルツインアーキ

テクチャである五層モデル [11] をベースに、BIM(Building Information Modeling) と呼ばれる、建物の形状情報や属性情報を保持した建築情報モデルを絡めたアーキテクチャに基づいて作成することで, 高精度な情報を保持した BIM の利点を活用しつつ目的を解決するためのデジタルツインを構築する。本研究では、これを二つの領域に対して実施し、各領域についての活動を報告しつつ、BIM をデジタルツインに組み込むことへの効果を記述する。また, このようにアーキテクチャを定め、要点を簡略化することで, 経済産業省が発行している DX レポート [12] にて, 既存システムのブラックボックス化として語られているように, システムの煩雑化によるメンテナンスの難しさを引き起こさないことを達成する.

1.3 本論文の構成

本論文では, 工場 (スマートファクトリー) とビル (スマートビルディング) という二つの異なる領域に基づいて, それぞれ課題点を定めた上で, その課題点を解決するため, このデジタルツインアプリケーションを作成, 使用し, 実証実験と検証を行った. それぞれの領域について, 具体的な課題点として以下の二つを挙げた.

- スマートファクトリー : 施設の状態監視を主目的とし, 工作機械のジオメトリデータ, 電力消費や稼働状況, 周辺の環境情報などを含めた各種情報を, ジオメトリデータを用い仮想空間上に再現した工場内にて, 立体的に再現することで直感的な工場施設の可視化を行うこと
- スマートビルディング : 運搬ロボットや照明, 空調設備を操作するためのシステムを構築し,BIM(Building Information Modeling) や LiDAR(LightDetection And Ranging) といった高精度な環境情報を扱うことでロボットの行動計画をデジタルツインアプリケーション上から生成すること

我々は, これらの領域についてこの二つの課題をそれぞれ割り当て, それぞれの要求を満たしたデジタルツインアプリケーションを構築した. 本論文では, 工場をテーマとした工場監視ツールとロボット, 照明, 空調操作を可能としたインタラクティブなデジタルツインアプリケーションを説明し, 要求事項を達成していることを示しながら, そのアーキテクチャについても述べていく. 具体的な論文構成としては,2 章にて関連研究・技術としてデジタルツインとデジタルツインアーキテクチャ、そして今回使用した技術について述べていく. 4 章や 5 章にてスマートファクトリー、スマートビルディングについて作成したデジタルツインアプリケーションの説明を行いながら、それを用いて課題に対処するといった事を記述していく。

第 2 章

関連技術・研究, 使用技術

本章では, デジタルツインについて, またそのアーキテクチャについて考察を行っている研究、及び 1.3 にて設定した各領域の課題に関する研究を挙げる。また、今回の研究で用いたジオメトリ生成サービスについても述べる.

2.1 デジタルツイン

デジタルツインとは, 狭義ではミクロの原子レベルからマクロの幾何学的なレベルまで, 現実上の物体を完全に記述する一連の仮想情報と定義される [13]. デジタルツインは製造業の製品管理の分野において設計, 生産, 運用, 保守をシミュレートすることに用いられた. 一方で広義としては, システムの状態やプロセスのシミュレート, 監視, 計算, 調整, 制御などが出来る統合的システムであると定義され, 物流, 情報の流れなどを体系的, リアルタイムに制御し, システム内の全ての事業活動を効果的に調整, 最適化することを目的としている. これは, 現実空間における対象のツインを仮想空間上に用意し, そのライフサイクルを仮想空間上に反映させ, ツインの現在や過去のデータから対象をリアルタイムに監視することが出来るシステムである. そして, これら取得データから現実空間の対象にフィードバックを行うことで, ライフサイクルや作業工程の最適化を図ることも目的とされる.

デジタルツインの元になる概念は元々,2002 年にミシガン大学の Michael Grieves が産業界に向けて行った PLM(Product Lifecycle Management) センタ設立時のプレゼンにて提唱されたものであるとされ [14], 製造者技術者協会 (Society of Manufacturing Engineering Management Forum) の会議で公に発表した [15] ものである. ここでは, 現実空間に存在するシステムと仮想空間に存在するシステムの間には,「創造」,「生産」,「運用」,「廃棄」の四つのフェーズがあり, それらが二つのシステムを繋いでいると定義した [16]. この時点で現在のデジタルツインの主要素である現実空間, 仮想空間とそれらを繋ぐデータフローのリンクを全て備えていた. 後に 2010 年にて, 最初の実用的な定義として米国航空宇宙局 (NASA) が宇宙船の物理モデルシミュレーションを改善するためにデジタルツインと呼称し, そのコンセプトを明記した [17][18]. 後の 2012 年に同じく NASA は, デジタルツインの概念を再検討し, デジタルツインを物理モデルやセンシング, それらの過去のデータを利用し, 対応した現実世界の対

象のライフサイクルを反映するマルチフィジックスかつマルチスケールな統合システムと定義した [4].

2.2 デジタルツインアーキテクチャ

上記のようにデジタルツインは元々製造工程にて用いるシステムとしてアーキテクチャが構築されていた.Grieves らは, 製造工程でのデジタルツイン基盤として, 基本的な三層モデルを提唱した [19]. この研究では, デジタルツインは物理空間, 仮想空間とこれら二つの接合部分からなる三つの主要部分によって構成できるとした.

Tao らの研究 [20] では、過去の産業界におけるデジタルツインアプリケーションに関する総合的なレビューを行い、デジタルツインの主要な理論構成やデジタルツイン研究の最先端を体系的に提示している. この研究では、デジタルツインの主要な理論構成として、

- DT Modeling, Simulation, and VV&A
- Data Fusion
- Interaction and Collaboration
- Service

の 4 点を挙げており、これらが当時 (2019 年時点) の主要なデジタルツインを利用したアプローチであるとしている. また、この研究では、デジタルツインの構成モデルとして、三次元モデルと五次元モデルを提示している. 三次元モデルでは、構成要素として物理世界、仮想世界、それら二つの接合部分の三つの要素を提示しており、五次元モデルでは、これらに加えてデータ（新しい知識体系を構築するための前提情報）やサービス (提供する課題解決能力) を付加している.

Aheleroff らは, 製造業に留まらない,Industry4.0 における汎用的なデジタルツインプラットフォーム/リファレンスアーキテクチャを示した [11]. この研究では, 従来の三層モデル（物理層-コネクション層-デジタル層）ではなく, 新たに五層モデル (物理層-コミュニケーション層-デジタル層-サイバー層-アプリケーション層) を示した. 従来と異なり現実空間のレプリカとしてのデジタル層とシミュレーションを利活用するためのデータ群としてサイバー層を分離し、一般的なユーザが用いることを想定したアプリケーション層を追加した. この研究では、Industry4.0 にて用いられている技術を考慮し、様々な分野で利用可能な汎用的なデジタルツインアーキテクチャとして図 2.1 に示したリファレンスモデルを提唱した。これは、単純なデジタルツインから複雑なデジタルツインまで幅広く対応しているとしている。

2.3 スマートファクトリー

ここでは, デジタルツインアプリケーションとして明記はしていないが, 工場の高効率化に向け, 各社が行っている取り組みについて紹介する.

棚橋らは鋳造工場に特化したエネルギー利用の効率化に取り組んだ [21]. ここでは, 鋳造工場

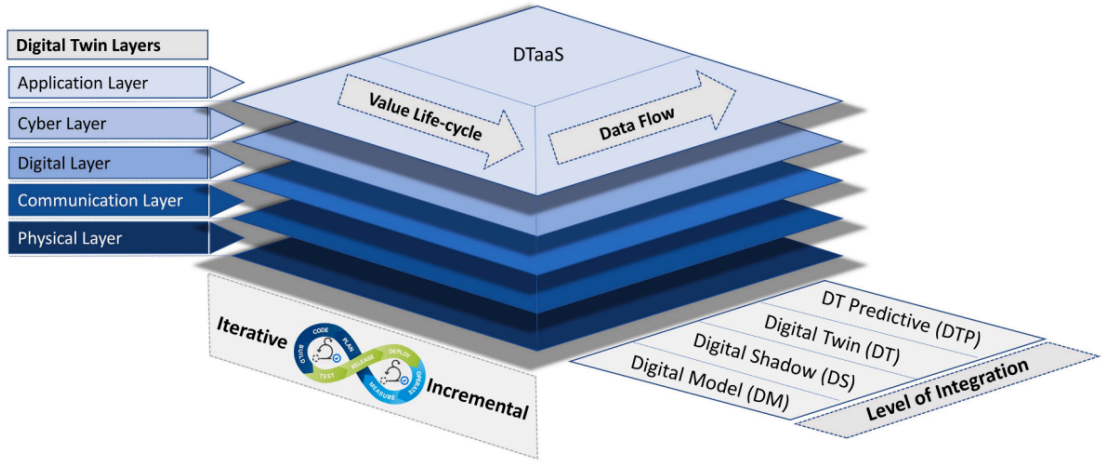


図 2.1. 提案されたデジタルツインアーキテクチャ [11] Fig. 5 から引用

での製造プロセスに着目し, ライン全体のエネルギー消費をリアルタイムに監視する機能やグラフとして可視化する機能を作成することで, エネルギー消費の可視化を行った. ライン全体の工作機械の内, 待機状態となる工作機械は溶解エネルギーのロスや生産性の低下が生じるとし, これらを可視化することも目的としている.

富士通グループでは,IoT を活用した工場の生産活動の最適化についての活動を行っている [22]. ここでは,IoT デバイスからリアルタイムにデータを取得し, それをクラウドサーバに蓄積し, 利用するまでのシステムを提案している. そして, 実際に会津若松の工場などで本システムを活用している.

このように, 工場の高効率化というテーマにて, いくつもの産業用アプリケーションが構築されているが, ジオメトリデータを活用し, 現実の工場との対応付けを明確にすることやデジタルツインとして, 仮想空間上でシミュレーションを行うといった試みはまだ少ない.

2.4　スマートビルディング

ここでは、5 章にて行う環境にセンサを設置し、情報を取得する研究やリアルタイムマッピングに関する研究を挙げる。

環境側にセンサを置き、特定の部屋の中で過ごしている人の行動や特性を把握する研究として森 武俊らの研究が挙げられる [23]。これは、生活支援の研究であり、ここでは、様々な量のセンサを多数、室内に設置し、部屋全体を可視化できるようにしている。環境側にセンサを置くことで、情報収集対象の情報を多く得ることが出来るといった利点がある。

Egemen Ertugrul らは、LiDAR を搭載した UAV を利用し、自律的にビル内の空中マッピングを行った [24]。この研究では、大学の建物をシミュレーション環境とし、Unity3D と ROS を用いて、仮想空間上でシミュレートをしている。空中から LiDAR センサを用いてマッ

ピングを行うことで、ある程度の地上に設置された障害物を無視しながらマッピングを行うことができる。

Viktor Tihanyi らは,Intelligent Transport Systems(ITS) の領域にて, 協調知覚とデジタルツインを活用した研究を行った [25]. この研究では, インフラや車両に搭載された複数のセンサを融合することで, 交通環境のデジタルツインをリアルタイムに生成し, 高い信頼性と範囲で物体を検出した協調型知覚システムを構築している.

5 章で行った実験では, 環境側に LiDAR センサを設置し、協調型知覚システムを組み込む. その際、LiDAR センサはある程度上方に設置することで障害物による対象物の遮蔽の影響を緩和する。

2.5 ジオメトリデータ生成

ジオメトリデータの生成には 3D スキャンやフォトグラメトリなどと言った技術が存在する。本研究において, 使用したジオメトリデータは, ジオメトリデータ/点群生成サービスである Matterport[*1] を用いた.Matterport は 3D スキャニングが行え, 専用カメラを用い, 撮影した場所をサービス上でジオメトリデータや点群などを作成できるクラウドサービスである. 本研究では, このサービスが指定しているいくつかの専用のカメラの内,Matterport Pro2 を用いた. 本カメラは赤外線による形状情報の取得や 4K カメラによる色情報の取得などが行える. 最大撮影距離は一回の撮影につき,4.5m で, 一回の撮影につき範囲内の正確性は 99% と記述されている [26]. が, 施設全体のジオメトリデータを生成する際は, それぞれの撮影箇所の連結操作が撮影位置, 周辺環境の特徴点の多さなどに依存するため, 場合によってはある程度の誤差が発生する.

実際に Matterport を利用した研究例として R. Shults らの, ミシガン州アッパー半島のクインシー鉱山という銅の立坑の 3D スキャニングに Matterport を用いた研究 [27] が挙げられる. 彼らは専用カメラ Matterport Pro2 を用い, 撮影を行った. 彼らの研究によると Matterport センサーの系統的な誤差は, 事前のキャリブレーションと補正無しに BIM アプリケーションに必要な数センチレベルの精度を達成するとしている. そして,Matterport の装置のコストは最も安価な地上波レーザースキャナーに匹敵すると評した.

[*1] https://matterport.com/

第 3 章

デジタルツインアーキテクチャ

この章では、提案手法として後の 4 章と 5 章にて使用するデジタルツインアーキテクチャについて述べる。

この研究では、Aheleroff らが示した五層モデルを参考に BIM を用いたデジタルツインアーキテクチャを作成し、4 章と 5 章にて各デジタルツインを作成し, 各デジタルツインにおけるオブジェクトの管理方法を模索し、各領域における課題を解決することが目的となる.

五層モデルの各層の要点を下記に示した。

- Physical Layer：センサや工作機械などと言った現実世界の情報を扱う物理層
- Communication Layer：物理層とデジタル層を接続する層
- Digital Layer：物理世界のレプリカ
- Cyber Layer：Digital Layer の中で特に ML やシミュレーションを利活用するためのデータ群
- Application Layer： これらの情報をユーザが閲覧、利用するためのアプリケーション層

この研究では、BIM データのようにジオメトリデータとその属性情報を活用することが多く、Cyber Layer には特に、ジオメトリ情報などを持ち込むことが多い。なお、Digital Layer は Cyber Layer と違い、例えば今回の研究では、BIM データの内、積極的に利活用しないデータもここに含む。

第 4 章

スマートファクトリー

4.1 背景

デジタルツインのユースケースの一つとして, 工場での使用が考えられる. 工場では工作機械や人の流れ, 作業効率など様々な要素が可視化され, 管理されることが望まれるが, 現在, 様々な企業がスマートファクトリを推進しているように, 実際にデジタルツインを構築し, 活用している工場はまだ多くなく, 普及段階である. ここで, スマートファクトリーのためのデジタルツインを考えた際, 工場の効率化のために何を重点的に可視化, フィードバックするかによって効率であるか否かが変わってくる.

我々は, 上記の課題に対して, 電力消費量の管理や物理空間の工場の稼働状況の可視化という観点から, 生産管理のためのデジタルツインアプリケーション (図 4.2), 及び工場に設置された工作機械や IoT デバイスなどの情報を集積しアプリケーションに提供するための API を作成した [28]. 本章では, 本デジタルツインの概要, 評価として実施した利用評価アンケートについて, 報告する.

4.2 提案手法

本研究では工場の効率化のために, 電力状況について着目した. 工場内の工作機械の電力消費量, 稼働状況, パトランプに関する情報などを集積し, 可視化するシステムを提案する. また本システムでは,3D ビューワを搭載し, 静的なジオメトリデータ上に工作機械の電力消費量から推定した稼働状況やパトランプによる異常状態,IoT デバイスに搭載されたセンサによって取得された温度情報や騒音情報などの環境状態の投影を行うことでリアルタイムに工場の状況を直感的に認識できるようにしている. ここで、静的なジオメトリデータは、ジオメトリデータ上の工作機械の位置などのメタ情報を付加することで、BIM のサブセットとして利用する。工作機械から蓄積したデータは, 電力消費量を棒グラフを, 稼働状況をガントチャートを作成することで, 過去のデータから工作機械の利用状況を可視化する. 過去のデータについては, 年, 月, 日といった時間に関する基準やエリアやグループなどと言った工作機械の所属するドメインによってまとめられており, データはこれらの項目によってカスタマイズされたものを表示

する. そして, 利用者の作業効率への影響という観点から, 工場に設置した工作機械や IoT デバイスの情報を反映する速度を速めることで, 工場内の情報共有や緊急性の高い事象を逃さないようにリアルタイム性を高めることを目指した.

今回実装するデジタルツインの各層の構成要素を下記に示した。

- Physical Layer：工作機械 (パトランプなど)、IoT デバイスなど
- Communication Layer：IoT ゲートウェイ、及びその通信路
- Digital Layer：照明情報など
- Cyber Layer：ジオメトリデータ、センサ情報、工作機械の電力消費量、位置情報など
- Application Layer：後に示す WebGL、Android アプリケーション

今回実装したアプリケーションの構成を図 4.1 にアプリケーションのスクリーンショットを 4.2 示した. 本実装では, バックエンドとフロントエンドに分けている. バックエンドは, ジオメトリデータを格納し,IoT ゲートウェイから取得した工作機械や IoT デバイスからの情報をリアルタイムに取得, 集積, カスタマイズを行うクラウドサーバである. フロントエンドでは, ゲームエンジンである Unity3D[*1]を用いて, ジオメトリデータを用いたサイバー空間内でのシミュレーションや 3D での可視化, 各種グラフの表示を行う.

なお, 工場をテーマにした上記デジタルツインアプリケーションでは, 株式会社シムックスイニシアティブ[*2]との共同研究という形で研究を行った. 具体的には, ジオメトリデータを作成する工場の選定, バックエンドの作成、アプリケーションデザインや一部機能の実装補助などを株式会社シムックスイニシアティブが行った.

4.3　設計と実装

4.3.1　ジオメトリデータの作成

2.5 節で述べた通り、本研究で使用した工場のジオメトリデータは、Matterport というクラウドサービスを用いた。図 4.3 は撮影を行った工場のジオメトリデータの上面図である.

4.3.2　アプリケーション

本アプリケーションの構築にはゲームエンジンである Unity3D を利用した. 利用するデータは全て、後述するクラウドサーバから取得し、ここでは、3D ビューワ内での可視化、グラフ作成のためのデータとして利用している. なお、取得しているデータは以下のとおりである.

- ジオメトリデータ
- IoT デバイスからのセンシングデータ (温度、湿度、照度、騒音)
- LiDAR センサからの点群データ

[*1] https://unity.com/
[*2] https://www.cimx-initiative.com/

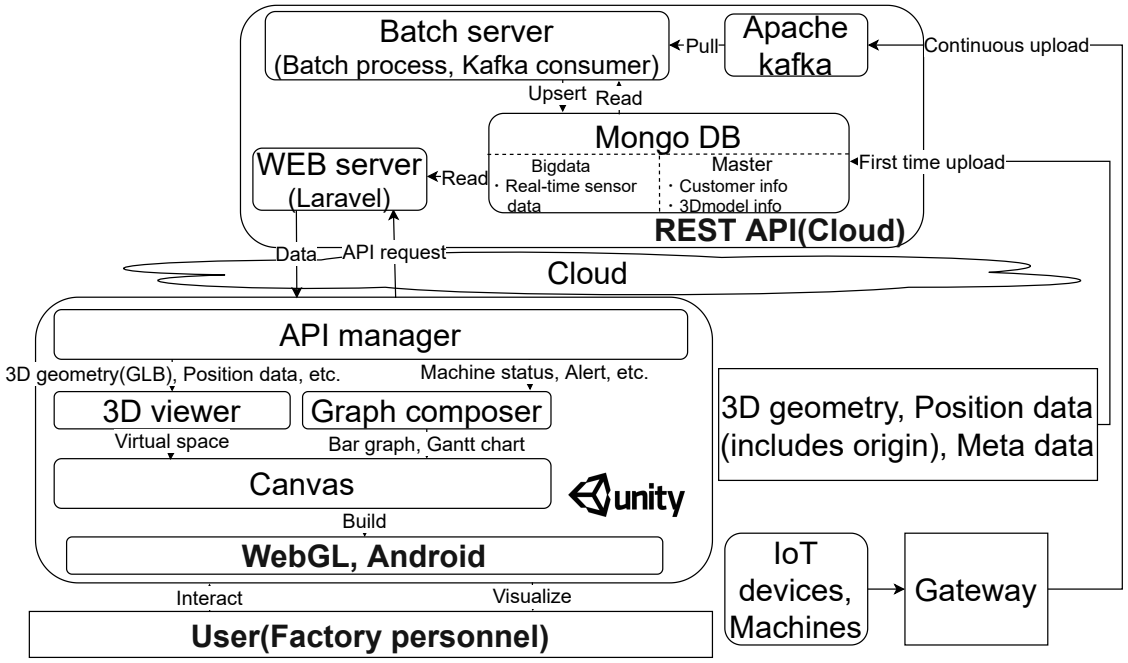


図 4.1. アプリケーション構成

- 工作機械のパトランプ
- 推定された工作機械の状態
- 工作機械の消費電力
- 顧客情報

また、IoT デバイスや工作機械では NTP との時刻同期機能を利用しており、データを取得するタイミングにて正確なタイムスタンプを保存しており、アプリケーション側にてデータを利用する際にはこれらも利用可能である。なお、基本的にデータ改変を行うことで CPU リソースを使わないために、集積されたデータのカスタマイズについては後述するクラウドサーバ上で行っている. 図 4.2 では、3D ビューワにて工作機械の稼働状況、電力消費量などを閲覧している. 工作機械の稼働状況は、対応した色を割り当てており、利用者が一目で工場全体の工作機械の稼働状況を把握できるようにしている. また、パトランプに関する情報も取得しており、図 4.2 の画面上部にあるタブを稼働から警告に変更することで、工場内全ての工作機械のパトランプの情報を取得し、画面に反映する. また、図 4.2 に表示されているように、工作機械にマウスカーソルを合わせることで、工作機械に関する詳細情報 (工作機械の名称、機械種別、稼働状況、消費電力、パトランプの状態) をポップアップにて表示する. 工場内はいくつかのエリアに分けることが可能で、広い工場の中で目的のエリアだけを中止したい場合は、図 4.4 に示されているようにエリア別に 3D ビューワーを利用することができる. また、エリアでのグループ分け以外にも機械種別ごとにグループ分けを行うことも出来る. これらのグループ分けは利用者が事前に決めることになる. 工場の 3D ジオメトリには、テキスチャが付与されてい

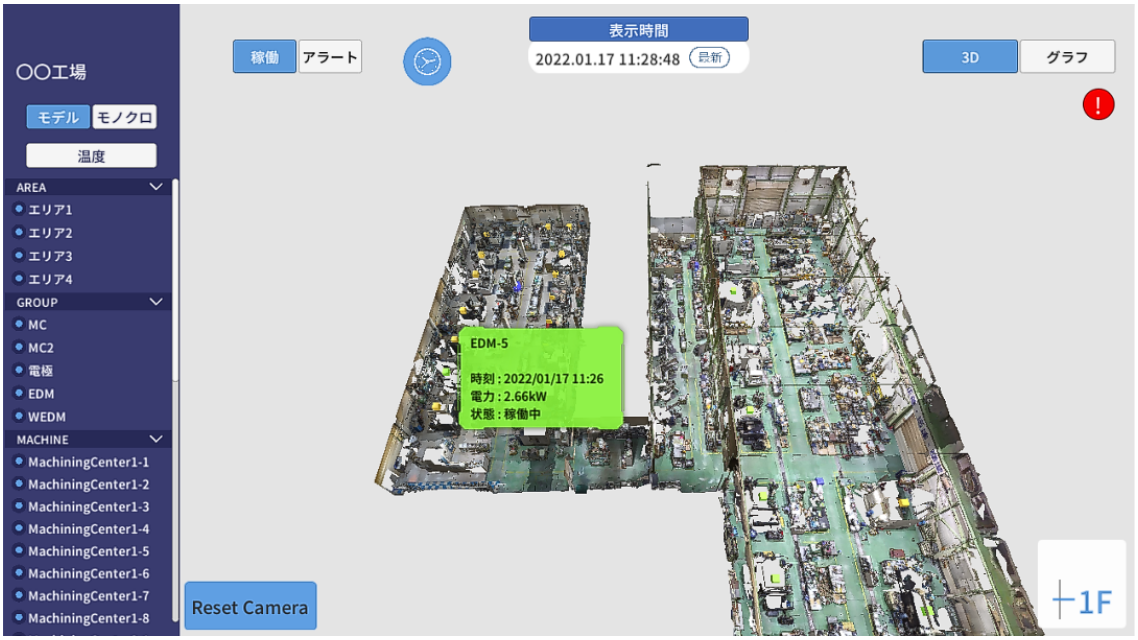


図 4.2. アプリケーションのスクリーンショット

る場合、そのテキスチャも表示されるが、その場合、工作機械を示すオブジェクトが見にくい場合がある. そのため、図 4.5 に示しているように、3D ジオメトリに割り当てられているテキスチャを単色かつ透過したものに変更する機能がある. これにより、視覚的に工作機械が捉えやすくなっている. そして、図 4.5 では、IoT デバイスから取得した環境情報をヒートマップとして 3D ジオメトリの上にオーバーレイする機能も搭載しており、環境情報を数値だけでなく、可視化することを行っている. 図 4.6 では、工作機械の稼働状況をガントチャートを用いて時間別に表示している. 図 4.7 では、工作機械の電力消費量を稼働状況による積み上げ棒グラフにて表示している. ここでは、3D ビューワーをリアルタイムに情報を把握するための機能として、グラフは過去や現在の履歴データから工場の状態変化を示すための機能である.

4.3.3　3D 空間 - 稼働/警告

稼働状況は使用電力の推移から停止 (白色)、待機 (黄色)、準備 (オレンジ色)、稼働 (緑色) の 4 項目から推測され、それぞれの工作機械を示す立方体に各種状態の色が割り当てられる。警告タブを押した場合、現在稼働している全工作機械のうち、パトランプが点灯しており、警告状態となっているものが赤く表示され、正常なものは緑として表示される。

4.3.4　グラフ機能

本アプリケーションはグラフ機能として 4.6 や 4.7 に示したグラフを表示する機能を持ち、また、これらのグラフの情報を更なるデータ分析にかけられるように CSV 形式で出力する機能を持つ。

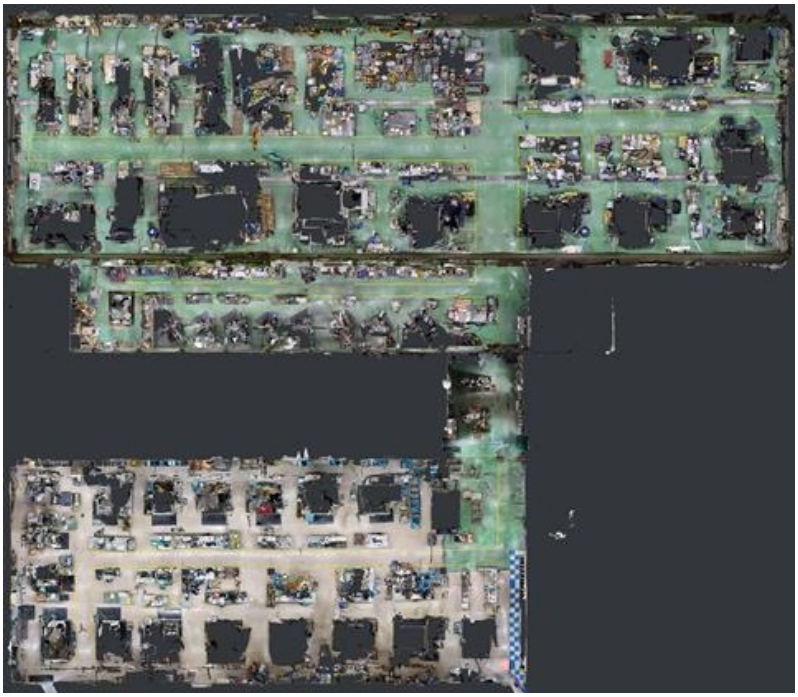


図 4.3. Matterport を用いて作成した工場のジオメトリデータの上面図

4.3.5 クラウドサーバ/API

ここでは、バックエンド側であるクラウドサーバやその API について説明する。バックエンドであるクラウドサーバは図 4.1 に記載されているようにいくつかのサーバにより構成される. まず、IoT ゲートウェイからアップロードされたデータはクラウド上の Apache Kafka[*3] に一時的に保存される.Apache Kafka からは任意のタイミングでデータの取り出しが可能なので、バッチサーバ上では Kafka Consumer が常時動作しており、アップロードされたデータを即時データベース（Mongo DB[*4]）へと投入している. データベースに保存されるデータは大きく 2 種類に分類されており、1 つは顧客情報、設備情報、ジオメトリデータ内の座標情報といったデータである. ここでは、便宜上これをマスターデータと呼ぶ。マスターデータは原則的に運用前に事前に登録しておくもので、更新頻度は低い傾向にある. もう一方は、工作機械の電力値、周囲環境の気温、湿度などの毎分のセンサ測定値を保存するデータである. これらは便宜上ビッグデータと呼ぶ。こちらには前述のように Kafka Consumer から常時データが投入されるので、更新頻度は非常に高い. また、ビッグデータにはセンサの測定値そのものも保存されている一方で、これらの測定値から生成されるデータも同様に保存されている. 例えば、バッチサーバーでは工作機械の稼働判定処理を行なうプログラムが毎分動作しており、工作機械の主電源及び加工電源の電力値の大小に基づいて稼働状態（稼働、準備、待機、停

[*3] https://kafka.apache.org/
[*4] https://www.mongodb.com/

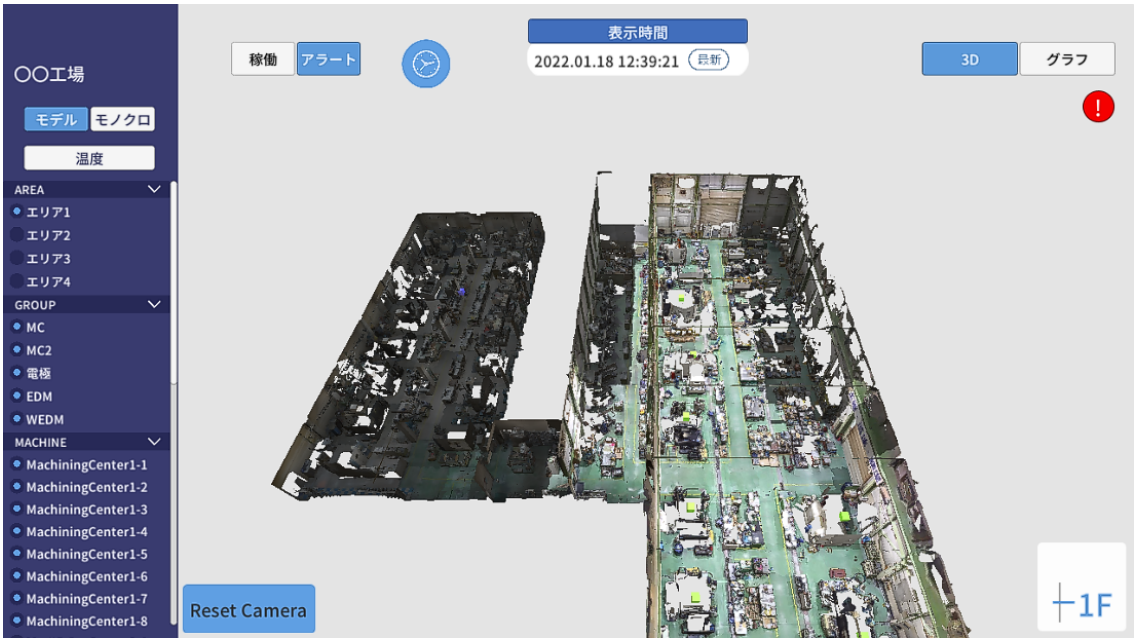


図 4.4. エリア別の強調表示機能

止）を判定し、その判定結果をデータベースへと保存している.API は PHP の WEB フレームワークである Laravel[*5]を用いて設計されており、ルーティングファイルに URL と実行処理を記述しておくことで、HTTP リクエストに対するレスポンスを規定している.URL はパスパラメータとして主に測定項目、対象（設備の ID、設備のグループの ID など）、期間、粒度（1 分単位、60 分単位、1 日単位など）を指定可能で、汎用的にデータの取得ができる設計となっている.API はこれらのパラメータに基づいてデータベースから対応するデータを抽出するとともに、1 分単位のデータから 60 分単位のデータを作るといったような集計処理、フロントエンドに常に同じ形式でデータを渡すといった整形処理の役割も果たしている.

4.3.6　負荷軽減のためのキャッシング

本アプリケーション内で最も大きいデータ容量を占めているのが, 工場のジオメトリデータである. このジオメトリデータは Khronos Group によって開発された glTF(GL Transmission Format)[*6]のバイナリ形式である GLB 形式を用いている.

2.5 節にて紹介した Matterport を用いて作成したジオメトリデータは OBJ 形式 (Wavefront OBJ) で出力されるが、glTF 形式は OpenGL 系 API と親和性が高いため、データ変換を行っている。なお、この際、テキスチャの解像度を落としたり、圧縮操作を行うといった軽量化を図っている。しかし、今回使用した工場のジオメトリデータのデータ容量は約 20MB あり, 工作機械や環境センサからのセンサ情報と比較するとアプリケーションの大半のデータ

[*5] https://laravel.com/
[*6] https://www.khronos.org/gltf/

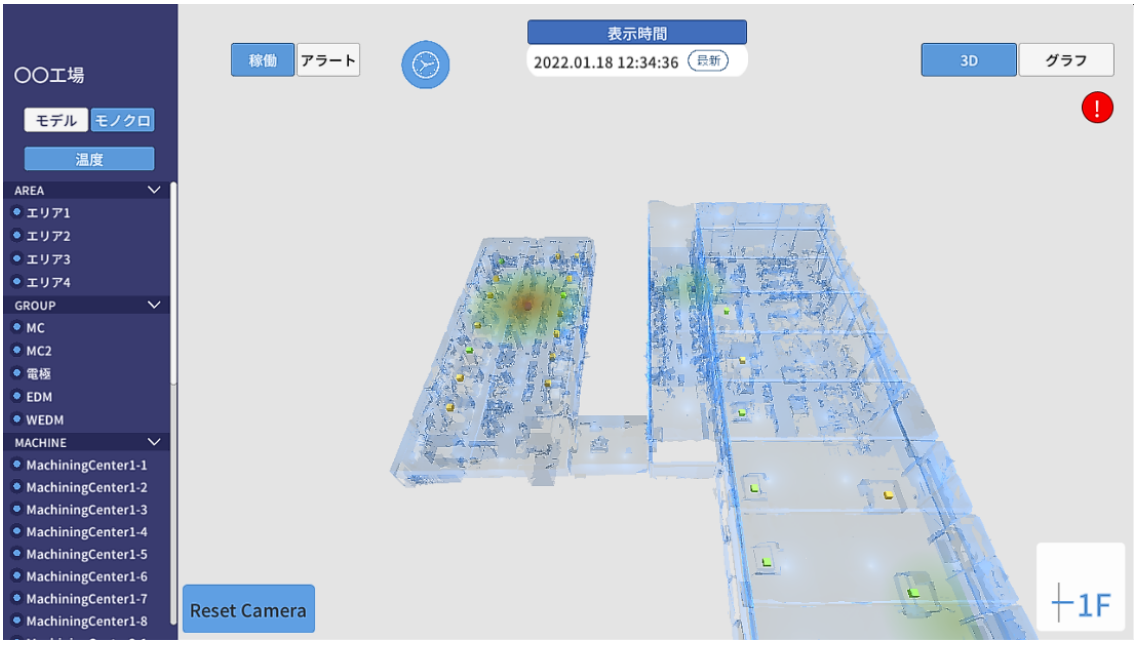


図 4.5. テキスチャ透過、ヒートマップ表示機能

転送量を占めると言える。そのため、本アプリケーションでは、通信の負荷を軽減するために、初回起動時にユーザが所属する工場のジオメトリデータのデータをダウンロードした際、ブラウザや Android デバイスにキャッシュとして保存する。次回起動時以降には、サーバに保存されているジオメトリデータの MD5 ハッシュとユーザがキャッシュとして保持したジオメトリデータの MD5 ハッシュを比較し、サーバ側のジオメトリデータの更新がなく、一致していればそのままユーザが保持するジオメトリデータをロードするといった処理を行っている。なお、ここでブラウザと Android デバイスでは、それぞれ IndexedDB、端末内のストレージにデータを保存する。

4.3.7　モバイル端末向けアプリケーション

また, ここではブラウザを用いて用意したサイトに配置された WebGL のアプリケーションについての説明を行ってきたが, 我々は別途 AndroidOS 向けにアプリケーションを作成しており, 利用可能なユーザを広げている. 図 4.8 に Android 版アプリケーションのスクリーンショットを添付した. なお,AndroidOS 向けのアプリケーションも WebGL 版と基本的には, 同様の構成となっているが, 利便性を高めるため,WebGL 版と異なり以下の 2 点が追加機能として搭載されている.

- ピンチ操作を用いて 3D モデルのズームを行う
- 左側のサイドメニュー UI が端末の解像度に応じて変化し, 拡大される

二点目に関しては, 元々 WebGL 版では PC 向けに作成していたため, 特にモバイル端末では

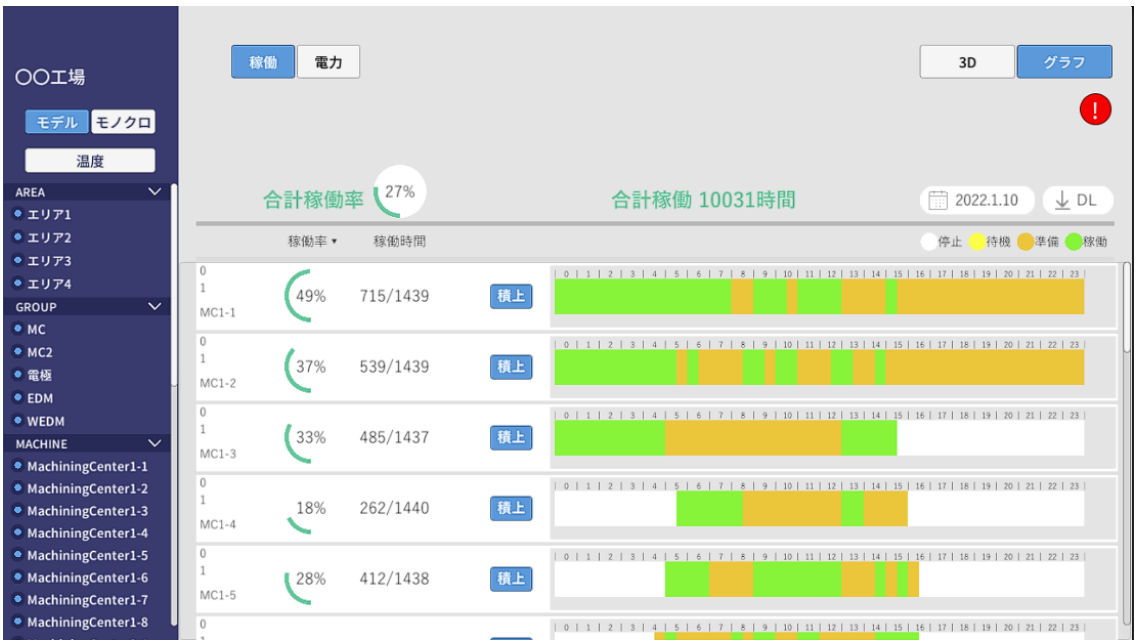


図 4.6. 工作機械の稼働状況を示したガントチャート

表示が小さく, 操作しづらいと思われた UI(ボタンサイズやフォントサイズなど) について, 端末に応じて拡大することとした.

4.4　評価

今回, このアプリケーションやクラウドサーバを含めた全体のシステムを評価するにあたって, 主観評価としてユーザアンケートを実施した. この節では、本アンケートを用いた定性的な評価を行う。なお、今回のユーザアンケートで使用したアプリケーションはブラウザで利用する WebGL 版とした。このアンケートはリッカート尺度に基づき, アプリケーションの内容について,8 つの指標を用意し, それぞれ**悪い**から**良い**までの五段階評価で回答してもらった.

4.4.1　アンケート

質問項目

アンケートでは, リッカート尺度に基づき, 以下の 8 つの質問に回答してもらった.

Q1 3D モデルの再現度に満足がいきましたか？
Q2 実際の工場 (3D モデル) に配置された工作機械と同じ場所にオブジェクトは存在したように感じましたか？
Q3 現在の工場の稼働状態を把握することができましたか？
Q4 過去の工場の稼働状態を把握することが出来ましたか？
Q5 操作は簡単にできましたか？

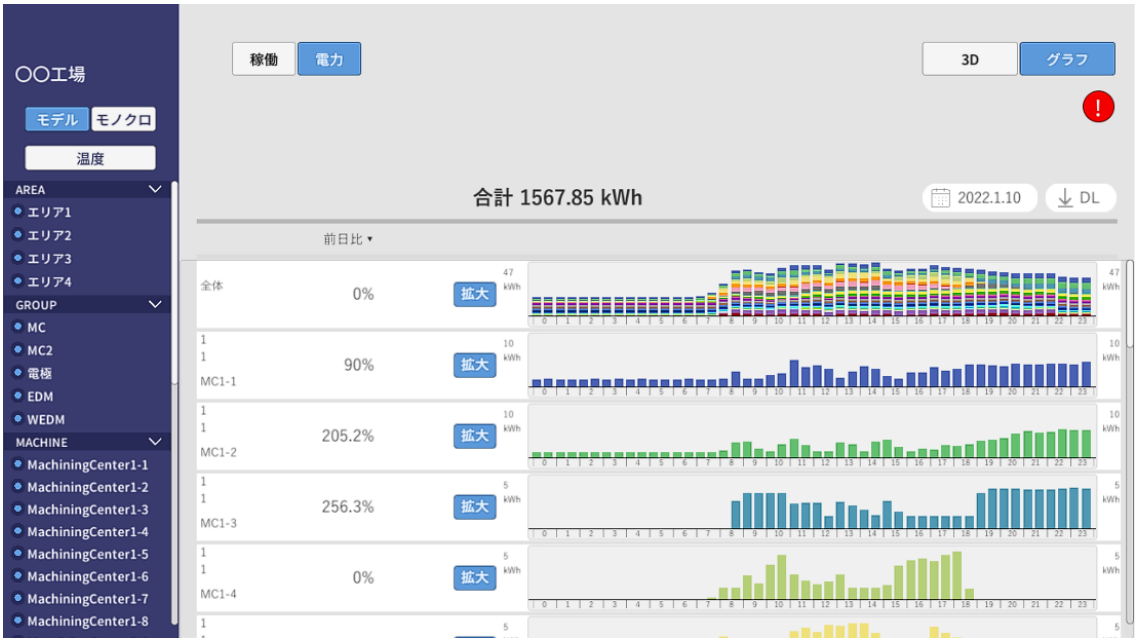


図 4.7. 工作機械の電力消費量を示した棒グラフ

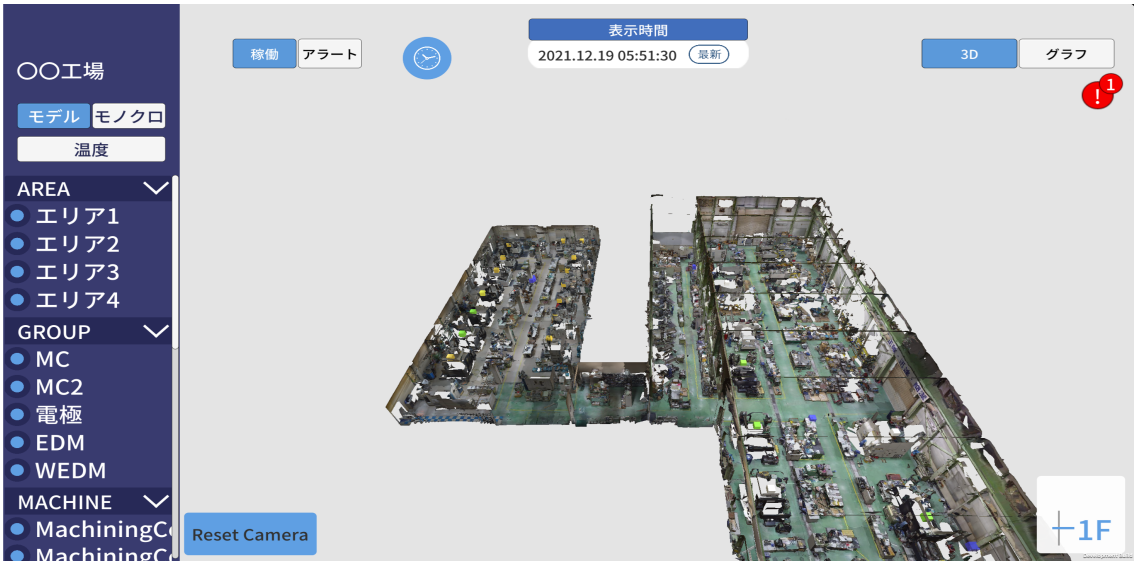


図 4.8. Android 版アプリケーションのスクリーンショット

Q6 3D による可視化は直感的でしたか？
Q7 グラフ機能による, 稼働状況の可視化はわかりやすいですか？
Q8 グラフ機能は工場業務に役に立ちそうですか？

各設問の意図として、まず設問 Q1,Q2 では、映像の再現性について問いた。現実の工場のツインとして、仮想空間上に工場のジオメトリデータと共に各種工作機械、デバイスのオ

ブジェクトを反映させているため、そこに違和感を感じたかを問いている。設問 Q1 では、Matterport を用い作成したジオメトリデータに違和感を感じるか、設問 Q2 では、工作機械やデバイスを示す立方体上のオブジェクトの位置がジオメトリデータ上の工作機械の位置からずれていないか、違和感を感じないかを問いている。設問 Q3,Q4 では、指定した操作を正しく行い、本アプリケーションの目的である現在、過去の工場の稼働状況の把握を行うことが出来たかを問いている。設問 Q5 では、稼働状況把握などの目的を達成するための操作において、その容易さを問いている。設問 Q6 では、本アンケートの主目的である、三次元空間上に工場のツインを再現し、可視化したことについて、その効果を問いている。設問 Q7,Q8 では、グラフ機能について問いている。設問 Q7 では、グラフによる稼働状況の把握のしやすさ、設問 Q8 では、その利用可能性について問いている。

また, アンケートでは, リッカート尺度に基づいた 8 つの質問の他に, 以下の補足情報についての回答項目を用意した.

- 回答者の性別
- 回答者の年齢
- 工場業務に関する専門性
- 本アプリケーションを使用するにあたって使用したデバイスの種類（名称）
- 本アプリケーションを使用するにあたって使用したデバイスの OS 名
- 本アプリケーションを使用するにあたって使用したブラウザ名
- アプリケーションを使用した際のネットワーク環境
- 本アプリケーション内にてサンプルとして使用している工場へ行ったことがあるかどうか
- 感想・要望
- 不具合報告

アンケートの回答方法、操作手順の指定

回答者には, 本アプリケーションがブラウザ上で操作できる URL とアンケートを回答するためのウェブフォーム (Google Forms を利用) の URL を提供した. なお, アンケートの説明書きの中にアプリケーションの操作方法を説明したプレゼンテーション資料や同じく操作方法を説明した動画ファイルへを参照するための URL を記載したため, 回答者は必要に応じてこれらの事前知識を得ることが出来た. 付録 A にこの時使用したプレゼンテーション資料を添付した. また, 動画については, このプレゼンテーション資料に書かれている実行してもらいたい操作内容をそのまま動画にしたものと, 簡単な他の機能の紹介に留めている. ここでは、回答者に実行してもらう操作として、3D 空間の閲覧操作、各工作機械の稼働状態の表示、過去の工作機械の状況を知れるタイムラプス機能の使用、各種グラフの表示を指定した。

アンケート実施時の状況

アンケートでは, 共同研究先である株式会社シムックスイニシアティブの中島が登壇した寺小屋というセミナーの参加者に対して実施した. ここで, 参加者は約 50 名であった。アンケートの実施までの具体的な内容としては, 中島が 2021 年 11 月 10 日に開催されたセミナーにて, アンケートの説明を行い, 先述した本アプリケーションがブラウザ上で操作できる URL とアンケートを回答するためのウェブフォームの URL を提供した. アンケートの回答期限は 2021 年 11 月 10 日から 2021 年 12 月 1 日までとした. 回答者は期間中, 自由に上記 URL を参照でき, 自由にアプリケーションを触れるものとした. また期間中, アプリケーションを用意しているサーバも同時に複数人がアクセスして良いように増強を行ってからアンケートを実施した.

アンケート結果

約 50 名の参加者の内、回答者は男性 22 名、女性 1 名であった. 年齢構成としては、20 代 2 名、30 代 3 名、40 代 8 名、50 代 8 名、60 以上 2 名であった。また、工場業務の専門性に対する解答として専門家 6 名、非専門家 15 名、無回答 2 名であった。参加者は工場関係者のみならず、ビジネス的な観点を持ち参加する人が多かったと思われる。

アンケートの調査結果を図 4.9 に示す。グラフの横軸は 1 から 5 までの 5 段階の回答比率を表し、合計値は 100% である。中間評価である 3 の回答比率の中心を横軸の原点 0% に置く。横軸の正の方向には高評価である 4 と 5 を、負の方向には低評価である 1 と 2 を置いている。これは、高評価の回答数が低評価の回答数より多い場合、グラフが正の方向に偏り、その逆の場合、負の方向に偏ることを示す。

図 4.9 を参照すると、アンケート結果は殆どの設問で概ね高評価を得られた。また、アンケートの感想や要望の項にて受け取った意見を確認すると、稼働状況の可視化がリアルであったといった意見や温度マップといった環境に関する情報を三次元空間上に投影することは分かりやすいといった意見があった。これは、本手法にて作成した三次元空間上に工場の稼働状況を再現するといった手法が一定の効能を持つことを示していると言える。

しかし、設問 Q5 については、他の設問と違い、低評価の方が顕著である。これは、操作性の項目であるが、これはアンケートに用意した感想・要望の項を確認すると、操作が難しい、望んだ工作機械に対して思うようにズームすることが出来なかったといった意見があった。これは、本アプリケーションでは、工場の中心点 (全体のメッシュデータの端を計算し、その中点を求める) を中心として回転する設計になっており、上下左右回転といった自由移動を限定しているためにあると想定される。

被験者の他の意見として、生産業務において 3D である必要性は感じなかったと言った意見があった。実際、各企業が展開している工場監視ツールや IoT デバイス管理ツールは二次元上にマッピングされた監視ツールが多い。ただ、本意見の続きとして、新入社員などが工場管理を行う場合や設備数が多く、設備番号で直感的にどこの設備だと分からない場合には有効かも知れないといった意見や温度マップなど環境に関することについては 3D マップのメリットを感じたといった意見に続けており、本アプリケーションの監視だけではない目的というもの

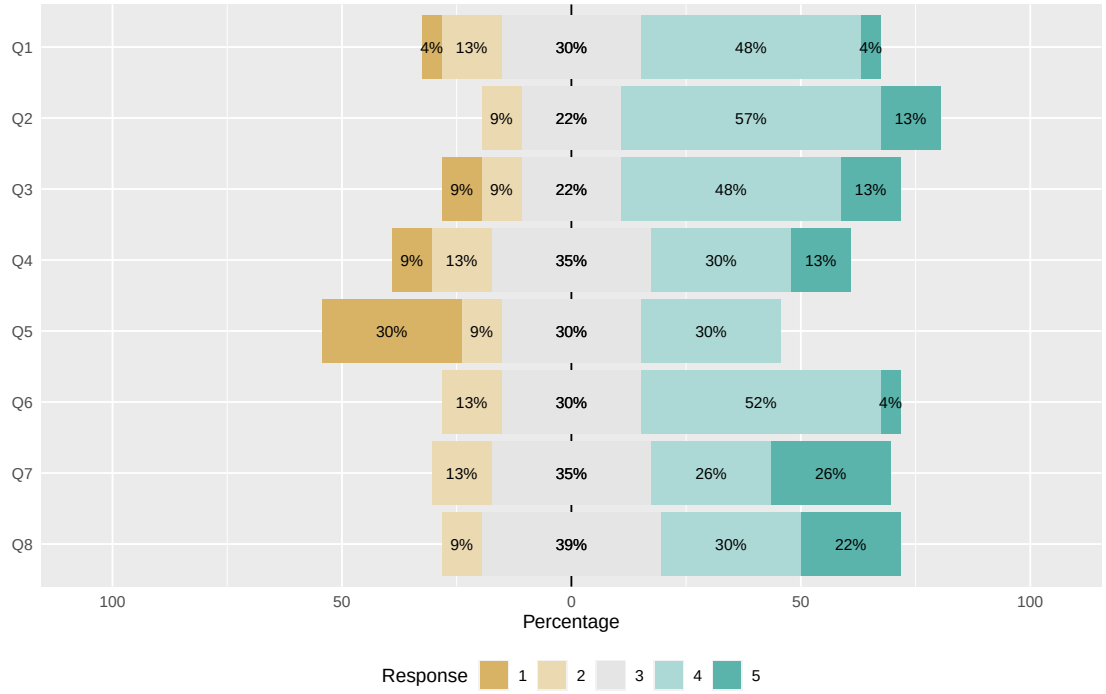


図 4.9. アンケート結果のリッカート尺度

が提示された。

本アプリケーションは、現在、温度マップの表示のみ行っており、湿度マップや騒音マップなどは作成していない。そのため、今後はそういった 3D 空間をフルに活用することが出来るヒートマップ機能の拡張や 3D 空間を活用した新たな機能の開発が求められる。また、対象とした工場内にて、立体的に監視対象のオブジェクトが配置されている場合、例えば、スペースを有効活用するべく、立体的に荷物を置いていることが多い、倉庫や物置などについては 3D としての恩恵が強いのではないかと思われる。

4.4.2　要求性能評価

ネットワーク性能

本アプリケーションを実行するにあたって本アプリケーションを継続的に利用するために必要なネットワーク性能を評価する。

本アプリケーション本体のデータは約 $9[MB]$ であり、これは初回起動時にダウンロードされたのち、Unity3D の機能によりその大部分がブラウザ上にキャッシングさせることが出来る。また、4.3.6 にて触れたが、本アプリケーションでは、ジオメトリデータはキャッシングを利用し、初回起動時以外には通信を圧迫しないため、ジオメトリデータも本アプリケーションを継続的に利用するためには、重視されない。そのため、ここでは、アプリケーション内でのユーザの操作によってリアルタイムに発生するサーバとの通信データを見ていく。

まず、工作機械や IoT デバイスなどからのセンサ値を取得する操作を行った場合にサーバへ通信を行うが、これは一つ当たり約 $800[B]$ であり、極めて軽量である。

次に、本アプリケーションが定期的に行う通信として、工作機械全体の状態把握 (三次元空間上にて工作機械を表す立方体の色分けや警告通知に使用) がある。これは、全ての工作機械の情報を一度のリクエストで取得するが、データ容量は約 $5.5[KB]$ であり、これもまた軽量であると言える。他にも様々な通信 (ログイン情報や利用者情報など) があるが、これらは全て 1 リクエスト当たり $10[KB]$ 未満である。

次にグラフ機能を使用した際に発生する、全ての工作機械の詳細データを取得した際のデータ容量を見ていく。グラフ機能は、日、月、年ごとに各工作機械の情報を閲覧できるが、これらは事前にサーバ側で計算されたデータを表示しているため、そのデータ容量は約 $33[KB]$ と小さい。

最後に、タイムラプス機能を使用した際に発生する、全ての工作機械の詳細データを取得した際のデータ容量を見ていく。まず、タイムラプス機能では、連続した日付と時間を指定し、過去の状態を閲覧することが出来る機能である。この機能では、指定した期間内の 1 分置きのデータを取得するため、通信時に発生するデータ通信量が非常に大きい。そのデータ容量は期間を 1 日に設定した際に、約 $3.6MB$ である。この機能では、何日間でも指定できるため、その期間を伸ばしていくたびに約 $3.6MB$ の通信量が 1 日ごとに発生する。

以上から、本アプリケーションを継続的に利用する場合、タイムラプス機能が特にボトルネックとなり、例えば 1 か月間のデータを一気にタイムラプス機能で見ようとした際には、約 $108[MB]$ ほどの通信を行うため、データダウンロードに 10 秒ほど待ったとしても約 $86.4[Mbps]$ の高速通信を要求されてしまうという結果となった。

なお、過去の発表 [28] では、LiDAR センサからの点群の仮想空間上へのマッピングも行っていたが、点群データは容量が大きく、帯域を圧迫してしまうため、今回のアプリケーションでは機能を削除した。

4.5 課題

4.5.1 解決した課題

ここでは、上記評価アンケートの実施時に発生したいくつかの課題について、対処した内容を記述する。

上記評価アンケートでは、操作の難しさが多く挙げられ、その中で思った工作機械に対してズームすることが難しいといった意見があった。しかし、上下左右回転全てを可能にしてしまうと、ユーザが可能な操作が多すぎてしまい、操作に簡単になれることが難しいといった意見が研究チームの中で出たため、このアンケート実施後、我々は、図 4.2 において、左側に搭載されているエリアやグループ、工作機械のフィルタリング機能 (非表示機能) に着目し、トグルボタン (画像中の青い丸で表示されている部分) と工作機械などの名称の部分の二つにボタンを設置し、それぞれフィルタリング機能、そのデバイスに対するズーム（回転操作の中心を

そのデバイスに変更する）を行う機能とした。これにより、各デバイスはズーム操作を行うことが出来るようになり、操作性はある程度向上されたと思われる。

4.5.2　今後の課題

まず、評価アンケートの項にて数多く意見が出た操作性について、考慮する必要がある。これは、現在の工場の中心点を中心に回転する操作では、工場の全体が見通せず、ユーザが強い制限をかけられているように感じることが大きいと思われる。そのため、三次元俯瞰を用いたアプリケーションを参照し、少ない制限をかけた自由視点移動を実装するなどと言った対策が考えられる。

次に、三次元であることのメリットを協調するため、温度だけではなく、湿度や騒音と言った他の情報を利用したシミュレーションを提供することが考えられる。

また、今後行うべき評価として, 設備総合効率 (Overall Equipment Effectiveness:OEE)[29] や設備機器総合有効生産力 (Total Effective Equipment Performance:TEEP) を計測するといったことが考えられる。ここで,OEE とは, 設備の稼働スケジュール内にて, 設備の設計上の稼働効率に対し, どの程度稼働しているかを定量化したものである. また,TEEP は OEE を暦上の時間に換算したものである. なお,OEE は時間稼働率, 性能稼働率, 良品率の乗算にて求められる. ただし、設備総合効率や設備機器総合有効生産力は、それらのデータの信頼性を求めるには、長期間の工場の状態変化の監視が必要であり、また比較検討する場合、本アプリケーションを使用していない状態の同データを集める必要がある。

第 5 章

スマートビルディング

5.1 背景

スマートビルディングとは空間と IoT を融合させることで, 建物内の設備を一元管理し, エネルギー効率の最適化や快適性の向上などを目指すことである. 現在, 我々はビルを対象に BIM や IoT を駆使したスマートビルディングの実証実験を行っており, この実証実験では, 事前に用意した BIM データと静的センサなどによる各種データをもとに運搬ロボットの移動経路を動的に算出し, 目的地への指令を出すなどの課題を解決する. この実証実験を通して, 現在の課題として, 人や什器などがオフィス内で常に変化しているため, 事前に用意したマッピングデータのみでは, ロボットは新しく設置された障害物や人にぶつかってしまい, 目的地にたどり着けないことが上げられる. これは配置が決まっていることが多い工場などと違い、インテリアのために什器が頻繁に置き換えることもあるオフィスならではの問題である。また、ロボットは、ロボット本体が SLAM を搭載している場合などには, 移動しながら自身に搭載されたセンサを元に自分の中にローカルマップを生成するため, 解決可能だが, データとしてはロボットからの情報に限られてしまい, 実際には最短経路でない経路を選択してしまうことも考えられる.

そこで, 本研究では, 物理空間に事前に設置した複数の LiDAR センサの協調知覚により, 本課題を解決することを目指した. これは、環境側にセンサを搭載することで、自律移動型ロボットの再マッピングの手間を省き、更に LiDAR センサを用いることで、高精度な測定を目指す。また、運搬ロボットは前面にしか LiDAR センサがついていないことが多く、他方面からの追突は予期していない場合があるため、環境側から認知することで、警告音を鳴らすといった対処が行えるといった利点が考えられる。

5.2 提案手法

図 5.1 にロボットの行動計画を行うシステムの概要図を, 図 5.3 に本研究のデジタルツインの構成図を記載した. 本研究では, 提案手法として, 対象のエリアに複数のセンサを設置し, それぞれのセンサのマッピング情報を集積してエリア全体のマッピングを行い, 統括管理を行う

方法を提唱する. これにより, ロボット単体でのマッピングと比べ, 不明瞭な空間を排斥し, 正しい最短経路を導きやすくなると考えられる. また、その際、LiDAR センサはある程度上方に設置することで障害物による対象物の遮蔽の影響を緩和する。

図 5.1 において, 本手法を説明する. まず左の LiDAR Simulator にて, 複数の LiDAR センサをシミュレートし, 複数の点群データの集合体を得る. 次にその点群データ中間処理層にてクラスタリングする. ここで得られたクラスタの中からロボットと思われる物を判別し, その位置情報を中間処理層の一つの入力として扱う. 右側の Twin Simulator では, 現実世界のロボットのツインとなるオブジェクトを配置する. 今回, 実証実験にて使用したロボット (FRUTERA A-ROBOT-102)[*1]はある程度のノイズを含んだ自己位置推定ができており, その位置情報を活用できるものとする. この位置情報を中間処理層の二つ目の入力として使用する. 最後に Euclid Clustering と後述するロボットの推定方法により推定されたロボットの位置情報と, 最初の点群の集合体からロボットと思われる点群クラスタを引いた障害物と思われる点群とを組み合わせ, ツイン上でロボットの移動経路を算出する. なお、この際、BIM データの属性情報である IfcSpatialZone を使用し、ロボットの行動可能領域を事前に定めることが出来るといった機能も用い、BIM データを活用する。

なお, ビルをテーマにした上記デジタルツインアプリケーションは, 東京大学グリーン ICT プロジェクト BIM 基盤 WG[*2]にて, 研究を行った. 我々は, 主にデジタルツインアプリケーションの作成に従事し, 利用したジオメトリデータはエヌ・ティ・ティ・コミュニケーションズ株式会社[*3]のスマートラボ,BIM データの構築やバックエンドの作成, ロボットの調達などは本ワーキンググループに参加している各企業が行った. また、後の 5.3.1 項にて作成した API サーバは研究グループの一人と協力して作成した。

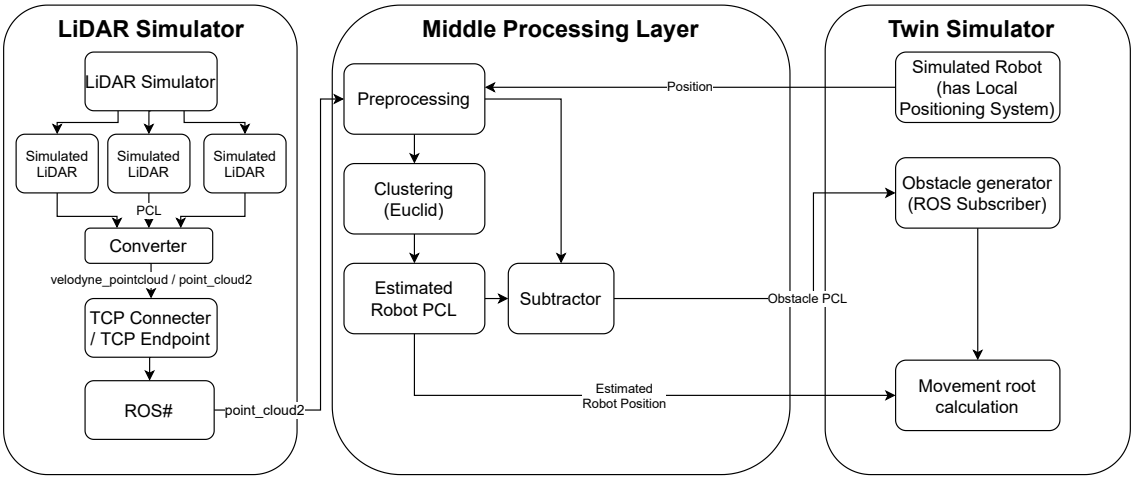


図 5.1. 行動計画システム概要図

[*1] http://androbo.jp/business/r/frutera/
[*2] https://www.gutp.jp/activity/?id=204
[*3] https://www.ntt.com/index.html

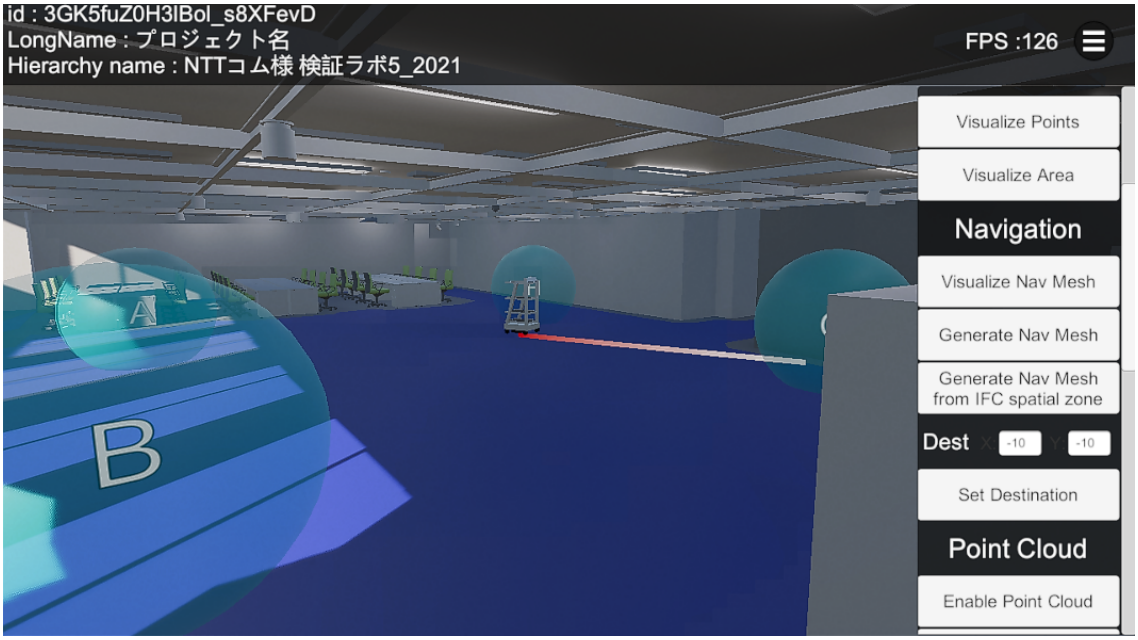


図 5.2. デジタルツインアプリケーションのスクリーンショット

5.3 設計と実装

本アプリケーションは東大グリーン ICT プロジェクトの BIM 基盤 WG の一つの実証実験として NTT コミュニケーションズ田町ラボでの実験をベースとした開発を行っている. そのため, デジタルツインアプリケーションとしてロボットの制御以外にも照明, 空調の制御, ビーコンの情報取得など様々な機構が作成されているが, 本実験と関係性が薄い部分の説明は省略, 又は省いている. シミュレーターを用いた本実験は、この様々なセンサの可視化及び制御機構を有したデジタルツインアプリケーション上で行うが、具体的にどのような機能を有しているかについては、付録 B に記載したスマートビルディングにおけるデジタルツインアプリケーションの各種機能説明を参考にされたい。

今回実装するデジタルツインの各層の構成要素を下記に示した。

- Physical Layer： ロボット、什器、オフィスなど
- Communication Layer：IoT ゲートウェイ、及びその通信路、ROS など
- Digital Layer：使用した BIM データの内、製品名や型番など
- Cyber Layer： ジオメトリデータ、センサ情報、ロボットの位置情報、IfcSpatialZone など
- Application Layer： ロボットの行動計画を算出することも行う、付録 B に示したアプリケーション

5.3.1　アプリケーション実装

今回,NTT コミュニケーションズ内での実証実験を行う際に, ロボット制御を行う API へのアクセスは, 特定 IP からのアクセスしか受容しないホワイトリスト制であった. そのため, この実証実験に向け, 我々はゲームエンジンが行う全ての対外通信を総括する API サーバを Azure を用いて新しく作成した. 具体的なアプリケーション構成を図 5.3 に示した. このサーバはゲームエンジンからは REST API として振る舞い, 内部ではロボット制御の API に対して MQTT を用いて通信を行い, 空調, 照明状態の確認を行う API に対しては REST を用いて通信を行う. また、本アプリケーションは 4 章と同様に Unity3D を用いて作成した。

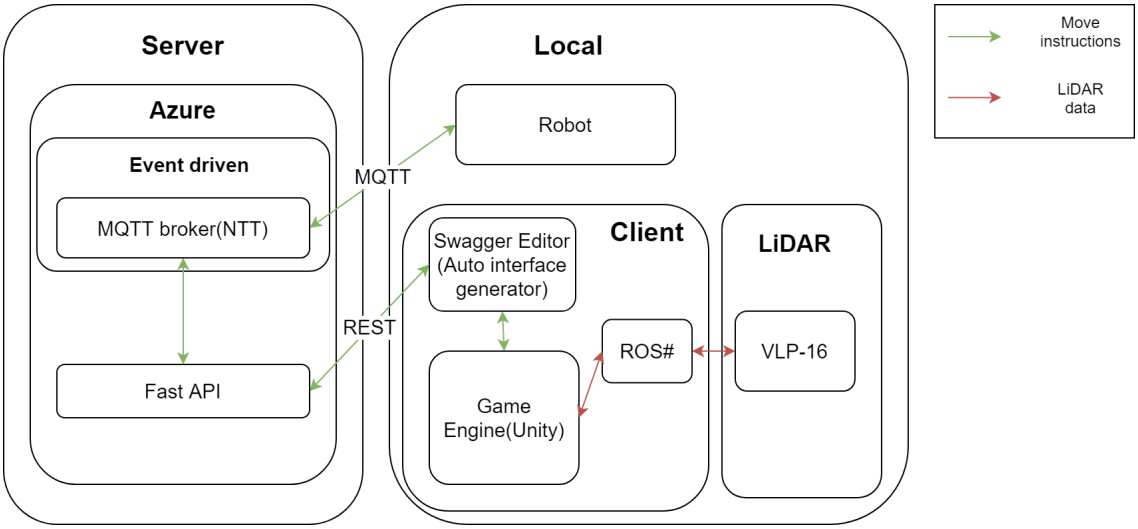


図 5.3. アプリケーション構成

5.3.2　ジオメトリデータの作成

2.5 節で述べた通り、本研究で使用したスマートラボ (ビルの一室) のジオメトリデータは、Matterport というクラウドサービスを用いた。図 5.4 は撮影を行ったスマートラボのジオメトリデータの上面図である.

5.3.3　使用した LiDAR センサ

今回使用する LiDAR として Velodyne 社の VLP-16(Puck)[*4]を用いる. 本体は図 5.5 のように, 直径 103.3mm, 高さ 72.7mm の円柱状の物とネットワークに接続するデバイスと電源からなる. このセンサは主に自動車につけ, 自動運転の補助データとして利用することを想定されている. 製品仕様としては, 水平 360°, 垂直 30°, 距離最小 0.5m から最大 100.0m の領域に

[*4] https://velodynelidar.com/products/puck/

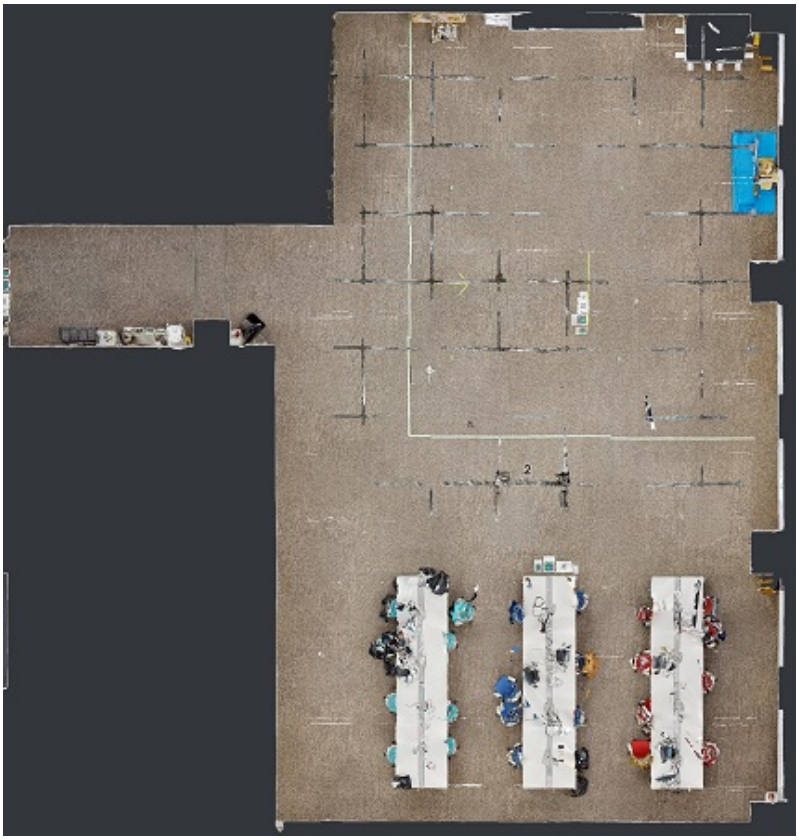


図 5.4. Matterport を用いて作成したスマートラボのジオメトリデータの上面図

て, 秒間約 300,000 点の点群を取得可能となっている. また, 取得した点群データは UDP を通して事前に設定した IP に送信することができる.Velodyne 社の公式 ROS パッケージとして, この UDP で送信されたパケットを ROS のトピックにパブリッシュすることができるパッケージが用意されている[*5]. 今回のシミュレータではこの VLP-16 と同じ挙動を行うオブジェクトを仮想空間内に作成し, 取得した点群データを ROS のトピックとして/velodyne_points にパブリッシュする.

5.3.4 中間処理層

シミュレータは LiDAR の挙動をシミュレートし、取得した点群データは ROS トピック /velodyne_points に Publish される。本項では、LiDAR センサから受け取った点群データを処理するための ROS パッケージを作成するために用いた手法を説明する。中間処理層では、ROS(Robot Operating System)[*6]と点群データを処理するために用いたライブラリとしてオープンソースにて開発されている PCL(Point Cloud Library)[*7]を用いた。PCL は点群

[*5] http://wiki.ros.org/velodyne
[*6] https://www.ros.org/
[*7] https://pointclouds.org/



図 5.5. VLP-16

処理に長けており、様々なフィルタやセグメンテーション処理を行える [30]。なお、本ライブラリのライセンスは BSD ライセンスである[*8]。

ここでは、図 5.6 のようにセンサを環境に設置し、そこにロボットを配置した際に、中間処理層にて行う各処理を実行した結果を記載していく。なお、センサの位置は固定し、点群の組み合わせを実行する際に、正しく組み合わせを行えるように全てのセンサの位置情報は事前に分かっているものとする。

点群の組み合わせ処理

各 LiDAR センサは ROS トピック/velodyne_points に点群データをアップロードするが、LiDAR センサごとに情報を分離するために、ヘッダー情報に含まれる frame_id を velodyne_x(x はナンバリング) ように変更する。次に各センサの座標を元に、それぞれの点群に対し、座標変換を行う。最後に一つの点群データとして組み合わせを行う。図 5.7 は、以上の操作を行った際の点群データの可視化である。

点群の軽量化

5.3.4 項の点群データは一つのセンサが取得する点数とセンサの数の掛け合わせを行った点数であるため、点群数の削減を行うことで処理の高速化を図る。

ここでは、Voxel Grid Downsampling を行う。これは、入力した点群データに対し、空間

[*8] https://github.com/PointCloudLibrary/pcl/blob/master/LICENSE.txt

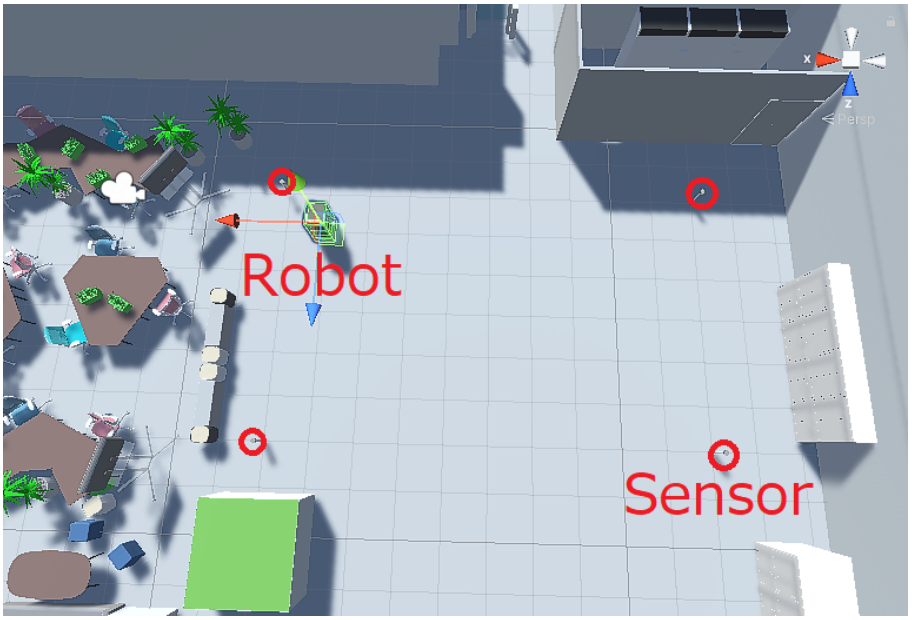


図 5.6. シミュレータ上のロボット、センサの配置

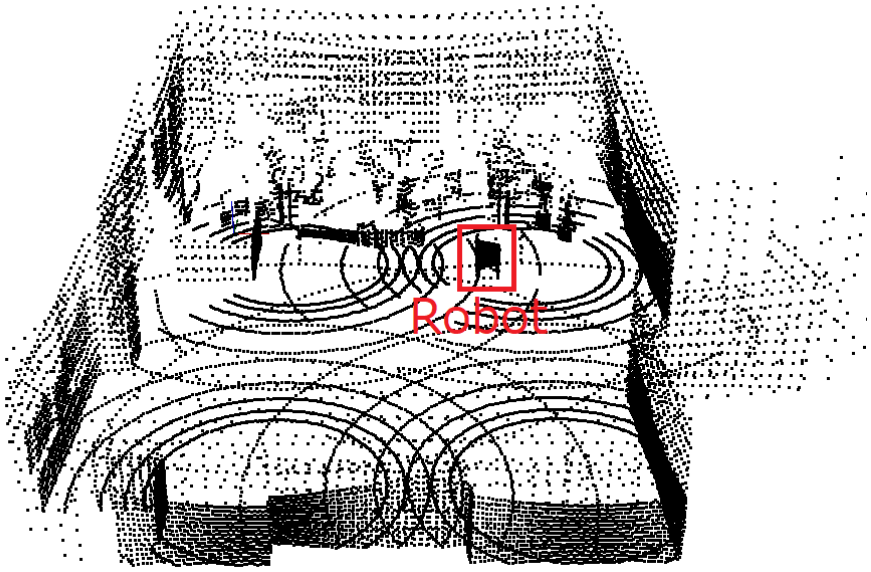


図 5.7. VLP-16 シミュレータによって得られた点群

上に三次元の立方体（ボクセル）を配置し、各ボクセルに含まれる点群をそれらの重心で近似するといった処理である。これにより、入力点群の特徴を抑えたままダウンサンプリングを行っている。5.3.4 項の点群データに対し、ボクセルサイズを 5$[cm]$ に設定し、実行したところ、約 228,000 点数の点群が約 18,000 点数まで削減された。

床面の点群の削除

与えられた点群の中から一定以上の大きさの平面を検出する手法として、Plane Model Segmentation[30] という手法が存在する。この手法では、入力モデル (ここでは平面) と距離の閾値を指定し、RANSAC[31] を用いて、入力点群の中から入力モデルと近しい平面を抽出する。

しかし、今回はロボットは段差など高低が発生する物体を登らず、移動する位置座標の高さは常に一定であるため、床を含む一定以下の高さの点群を削除するといった処理を行っている。図 5.8 は点群の軽量化を行った入力点群に対し、この処理を行った結果を可視化したものである。

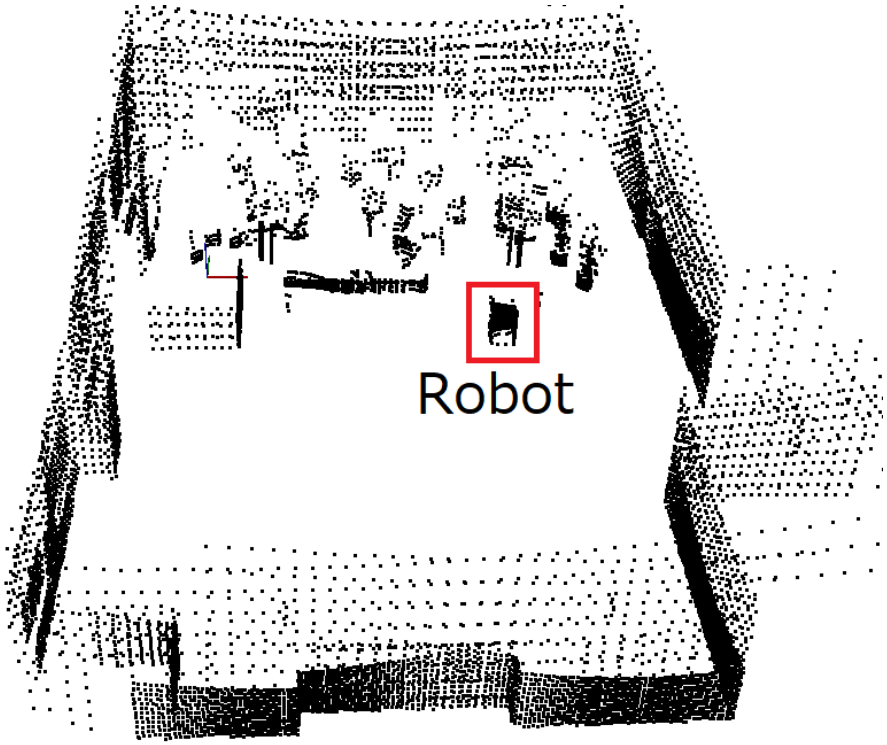


図 5.8. 床面の点群の削除を実行後に得られた点群

点群のクラスタリング

ここでは、点群の特徴を踏まえ、クラスタリングを行い、点群を各物体に分離する。ここでは Euclid Clustering Extraction[32] を用いる。Euclid Clustering Extraction とは、オクツリーデータ構造を用い、各三次元ボックス内のデータを kd-tree 構造を用いて最近傍探索を行い、各クラスタを抽出する。

図 5.9 は,5.3.4 節の操作を実行した後の点群に Euclid Clustering を実行し, 各クラスタごとに色を割り当てた際の様子である. 図の真ん中には, 対象のロボットが茶色のバウンディング

ボックスと共に黒く塗りつぶされている (赤色の四角で囲まれている物体).

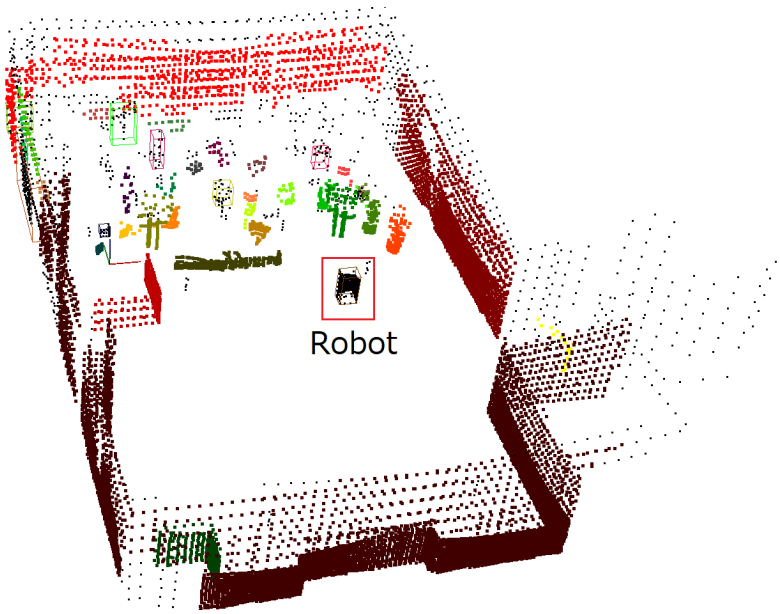


図 5.9. Euclid Clustering Extraction 実行後

ロボットの座標の推定

ここでは、抽出した各クラスタからどのクラスタがロボットであるかを特定する。ここでは、前提条件として、ロボットの位置情報が利用できるため、経路探索を行う際のロボットの初期座標を用いる。初期座標近辺に存在し、ロボットの特徴 (今回は大きさを使用した) と合致するクラスタを抽出し、そのクラスタがロボットであると想定する。

以上の操作から推定したロボットの位置は ROS トピック/processed_pose として、PoseStamped 型で Publish している。また、障害物情報は推定したロボットの点群を削除したうえで、ROS トピック/processed_pc2 として、PointCloud2 型で Publish している。

なお、ここまでの処理にかかる実行時間は、平均約 192.43$[ms]$ となった。

5.3.5 経路探索

経路探索には、Unity3D の機能の一つである Navigation System[*9]を用いる。この機能では、仮想空間上のジオメトリデータを元に、移動可能領域を NavMesh として算出し、その領域内で A*アルゴリズムを用い、経路探索を行うことができる。本研究では、VLP-16(及びそのシミュレータ) から取得し、中間処理を施した点群情報をそのままジオメトリ上に反映させ、障害物として Navigation System に認識させている。

[*9] https://docs.unity3d.com/2022.1/Documentation/Manual/Navigation.html

5.3.6　運搬ロボットについて

本実証実験で用いた運搬ロボットはアンドロボティクス株式会社の FRUTERA A-ROBOT-102 である. 以降, 後の 5.5 節にて行うシミュレーションでは本ロボットの特徴を踏まえ、独自に条件を設定した仮想ロボットを用いる.

5.4　本アプリケーションを用いた実証実験 - ケーススタディ

ここではケーススタディとして,NTT コミュニケーションズと執り行った本実験に用いたアプリケーションのベースとして開発したアプリケーションの実証実験について記載する. 我々は, 本デジタルツインの実証実験として, 検証する機能をいくつか段階に分けてテストを行った. その内, 本アプリケーションが関わっている試験シナリオのタイトルは以下の通りである.

- ゲームエンジンによるロボット制御
- VIP 判断及びロボット制御
- ゲームエンジンによる照明制御
- ロボット移動による照明制御

これらの詳細な内容と本アプリケーションを用いた具体的な手順は付録 B として添付した本デジタルツインアプリケーションの取り扱い説明書に記載してあるため, ここでは, 詳細を省く.

5.5　実験

条件

今回使用する仮想ロボットのベースとなる A-ROBOT-102 はロボットは自身の先端下部に装着している LiDAR センサを用いて, 事前にマッピングを行う (SLAM を行っていると思われる). そしてそのマッピングしたエリアの中からいくつかの目的地とホームポジションを指定する. 移動の際にはそれらの目的地を指定し, ロボットがそこまでの最短経路を内部で求め, 移動を開始する. その際に, 障害物 (人など) に接近した場合, 正面の LiDAR センサが物体を捉えることで, 障害物に接近したと判断し, 一時動作を停止する. また, 本ロボットはオドメトリを利用した自己位置推定を行っていると思われ, 車輪が空転した場合などを除き, 少量のノイズを含みつつ現在地をユーザに伝達可能である.

以上をまとめ、仮想ロボットの条件として、以下の項目を設定した。

- 一定のノイズを含む自己位置 (平面位置, 角度) をアプリケーションに伝達可能
- 障害物に対して距離 $0.5[m]$ 以内には接近不可
- 最大速度を $1.2[m/s]$, 加速度を $0.3[m/s^2]$ と仮定
- 座標を指定し、その場所に直線的に移動することが出来ると仮定

上記の条件の内、最大速度については、実証実験で用いている運搬ロボットの仕様として記載されていたものを使用した。実証実験で用いている運搬ロボットの移動機能は、事前に設定した座標に向かい移動するものであり、上記条件に設定したような自在に座標を指定する柔軟性はない。しかし現在、数多くのロボットは方向と移動量や移動時間を指定し、移動するといった操作が可能である。そのため、本実験では、仮想ロボットは自由に移動できるという仮定の下シミュレーションを実施する。ただし、実証実験で用いている運搬ロボットと同様に座標指定を行い、そこに対し直線的に移動するものとする。

シミュレータからロボットや障害物を正しく認知し、ロボットの行動計画を行えるかをテストするため、以下の実験を行った。図 5.10 には本実験にて仮想ロボットが移動を開始する移動開始地点と目的地を示している。移動開始地点は図 5.10 上にて、緑色の立方体で表されており、目的地は赤色の立方体で表されている。

複数の LiDAR は中間処理層に LiDAR が取得した全ての点群を送信し、中間処理層にてロボットや周辺の障害物などを分離、認知する。それをデジタルツインアプリケーションに送信し、デジタルツインアプリケーションがロボットの行動計画を行い、経路をロボットに伝送する。その時の様子を図 5.11 に示した。この図ではロボットのツインが与えられた情報を元に赤い経路を生成している様子が示されている。この図でロボットのモデルは推定された位置情報に位置するように配置されている。また、この図にて黒い点の集合体が障害物として認知された点群であり、床に生成された青いエリアはロボットが移動可能 (障害物から $0.5m$] 離れている場所) なエリアである。

図 5.12 には、デジタルツインアプリケーションによって生成された行動計画を遂行したシミュレータ上でのロボットが映っている。図の右上には本ロボットがデジタルツインアプリケーションから送信された移動命令 (座標情報) が表示されている。

5.6 評価

シミュレータ上の実際の仮想ロボットの位置と中間処理を終え, 提案手法により予測された仮想ロボットの位置とのずれや誤差の平均を計測する. ここで計測は仮想ロボットがスタートしてからゴールするまで行い、毎フレーム (今回の実験では 60fps) 情報を取得するものとする。ここで、図 5.10 に表示されている仮想空間上にて、左方向が X 軸正、下方向が Z 軸正である。

なお、ここで各軸毎のずれ平均 z_1 はロボットの座標の真の値を x_k(ここで、k はシミュレーション開始時から終了時までに取得したデータ (n 個) を順番に並べた際の添え字) とし、予測されたロボットの座標を y とすると

$$z_1 = \frac{\sum_{k=1}^{n}(x_k - y_k)}{n}$$

で表される。これは、今回の実験条件では、ロボットは常に X 軸負の方向、及び Z 軸正の方向に移動するため、真の値と予測値の差分を取った際に、符号が常に一致していることがあることから計測する（常に遅れて真の値の座標に到達し、符号が変わらないことがあるため)。

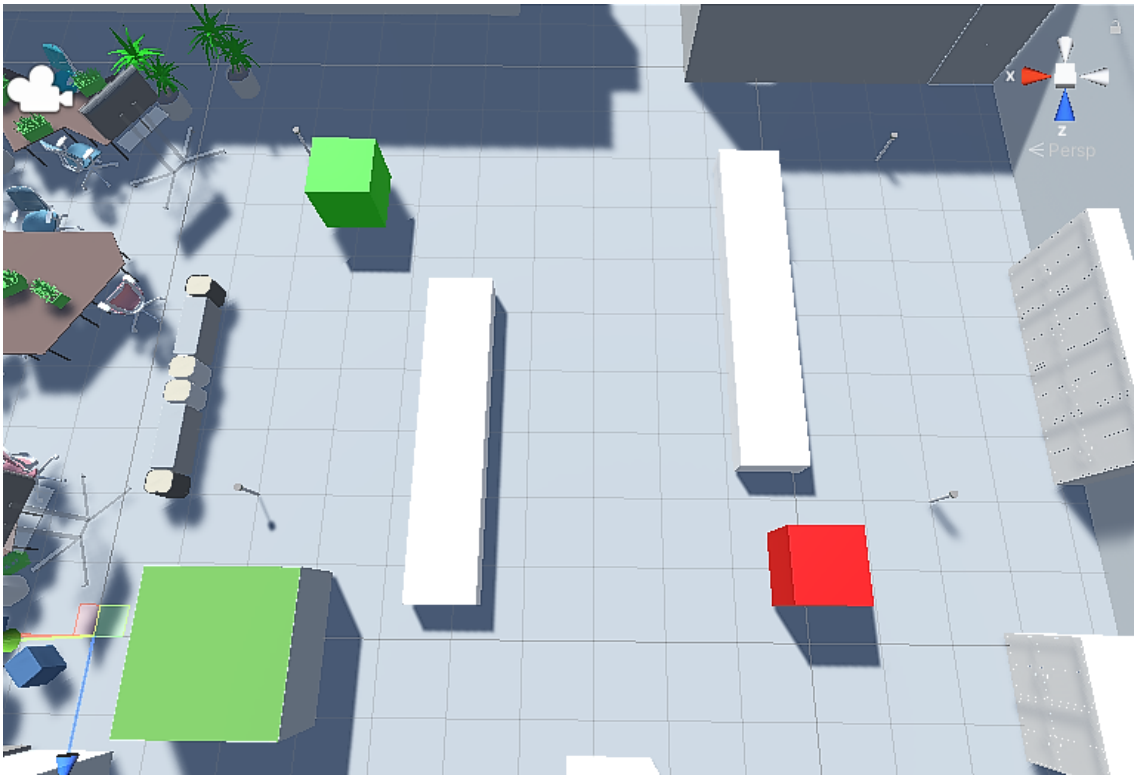


図 5.10. シミュレータ上でのスタート (緑) とゴール (赤) と障害物の配置

また、各軸毎の平均誤差 z_2 は

$$z_2 = \frac{\sum_{k=1}^{n} |x_k - y_k|}{n}$$

としてあらわされ、標準偏差は上記平均誤差 z_2 の標準偏差を取得するものとする。

最後に X 軸、Z 軸双方を考慮した誤差は X 軸と Z 軸の差分のユークリッド距離を利用する。

表 5.1 に計測されたデータを示した。表 5.1 を見ると、全体的に 200$[mm]$ ほどのずれが生じており、誤差が大きいと考えられる。ロボットの横幅は約 460$[mm]$ であるため、ロボットの半分ほどのずれが生じている。また、X 軸方向の標準偏差は 527$[mm]$、全体の標準偏差は 273.43$[mm]$ とかなり大きい。これは、今回の走行経路において、障害物がセンサからロボットを遮蔽している場所を考慮すると、ロボットを追跡できるセンサが限られているからであると思われる。ここで、ずれ平均に着目すると X 軸、Y 軸共に平均誤差とほぼ一致している。これは、上記に記載したずれの取得理由に示したように、常に予測値が真値より同じ方向にずれているからと考えられる。

この結果を踏まえると、今回の手法を用いたプログラムの実行速度が遅く、その影響がこの結果に表されていると考えられる。

そのため、新たにシミュレーション中に各フレームごとの実行時間を測定し、その時間だけ予測値の遷移をずらし、それらと真値の差分を計測した。表 5.2 に、新たに取得したデータを示した。

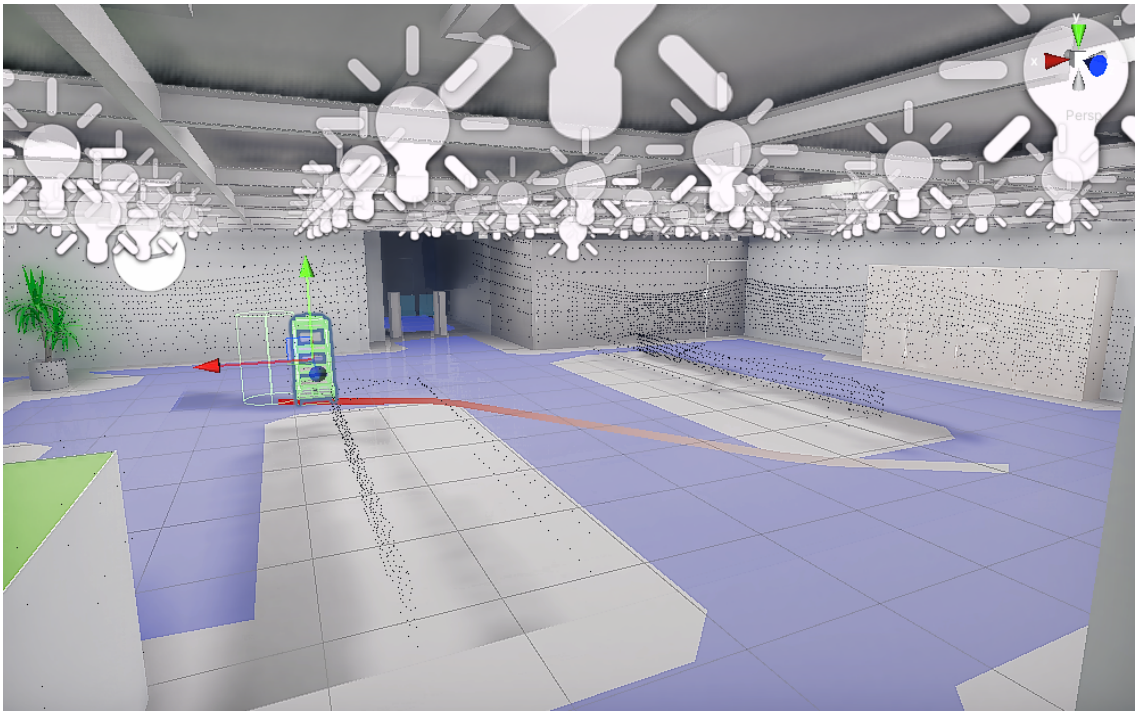


図 5.11. 生成された行動計画

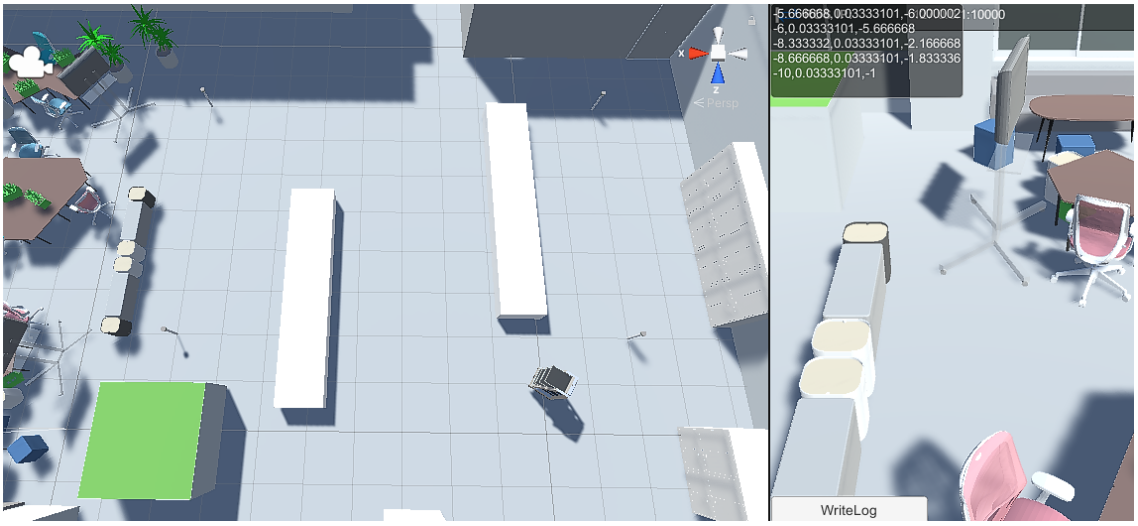


図 5.12. 生成された行動計画

これを参照すると表 5.1 と比較し、かなり誤差が減っていることがわかる。全体誤差を参照すると約 85$[mm]$ のずれが存在し、全体の標準偏差を参照すると 93$[mm]$ ほどの値が確認できるが、これは流動体などの接近時にある程度のマージンを保持し、接近時には停止する設定を持つ運搬ロボットにおいて、500$[mm]$ ほどのマージンを担保することが多いため、それと比較すると少ないものではないかと思われる。

この結果から、本手法では高精度の LiDAR を活用し、運搬ロボットの動的な行動計画に対応することができる可能性を示した。しかし、上記結果の通り、この手法を現実的な物とするには、特に大きい影響を誇る実行時間の削減を目指さなければいけないということがわかった。

表 5.1. 平均誤差

	ずれ平均 [mm]	平均誤差 [mm]	標準偏差 [mm]
X	-236.90	240.65	527.00
Z	201.15	201.33	190.34
全体		328.21	273.43

表 5.2. 平均誤差 (時間調整後)

	ずれ平均 [mm]	平均誤差 [mm]	標準偏差 [mm]
X	6.17	67.02	125.56
Z	-1.67	44.10	76.91
全体		85.01	93.26

5.7　今後の課題

本研究では、Euclid Clustering を用いて、ロボットの位置座標を追跡したが、この手法では人間などの流動体が近距離に接近した際に、Euclid Clustering のパラメータや LiDAR センサによる知覚範囲によっては、同一物体として認識されてしまう可能性がある。その場合、ロボットの推定に用いている特徴パラメータによっては、人物をロボットとして認識してしまい、ロボットの行動計画が破綻してしまう。そのため、本研究では初回のロボットの位置座標の特定でしか用いなかったロボット本体のオドメトリによるノイズを含む自己位置を利用し、接近した別の物体をロボットとして認識しないように配慮する必要がある。

また、5.3.4 項にてボクセルサイズを 5$[cm]$ に設定し、実行しているため、精度が少し荒くなっている可能性がある。しかしボクセルサイズが小さいとその分後の処理にて多くの点数を扱うこととなり、実行速度が遅くなってしまう。実行速度が遅くなってしまうと、その時間だけその時のロボットの真の位置と中間処理によって得られる推定値がずれてしまうため、精度を追求するにはこのパラメータの最適化が必要となってくる。

今回は実験に用いた LiDAR センサである VLP-16 の誤差が少なく、点群のクラスタリングによるロボットの位置推定の誤差が少なかったが、使用するセンサによっては更なる誤差が考えられる。そういった場合、ロボットのオドメトリなど、それぞれのパラメータを利用し、無香料カルマンフィルタやパーティクルフィルタといった非線形システムの推定手法を用いることが有効と思われる。

最後に、今回の実験では障害物は設置物と仮定しており、人間のような流動的な物体を用いた実験は行っていない。流動的な物体を用いた実験を行う場合、今回の手法のみでは位置推定は行えるが、ロボットの経路探索については流動体の移動方向を予測しつつ探索するといったことは行っていないため、ロボットの回避動作が間に合わず、流動体がロボットにぶつかってしまうといったことが考えられる。

第 6 章

評価、結論

本論文では、スマートファクトリー/スマートビルディングにおいて各課題を解決するための五層のデジタルツインアーキテクチャを用いて、デジタルツインを構築し、課題解決を試みた。

まず、スマートファクトリーでは、課題として「施設の状態監視を主目的とし, 工作機械のジオメトリデータ, 電力消費や稼働状況, 周辺の環境情報などを含めた各種情報を, ジオメトリデータを用い仮想空間上に再現した工場内にて, 立体的に再現することで直感的な工場施設の可視化を行うこと」を設定した。これは 4 章にて実施した主観評価アンケートにて、三次元の可視化はある程度直感的であり、効果があることが分かった。また、例えば、新人教育の際、工場の三次元モデルを活用して工場の配置を覚えさせるといった新たな使い方も発見された。そして、今後の課題として本アプリケーションの操作性に難があることが分かった。

次にスマートビルディングでは、課題として「運搬ロボットや照明, 空調設備を操作するためのシステムを構築し,BIM(Building Information Modeling) や LiDAR(LightDetection And Ranging) といった高精度な環境情報を扱うことでロボットの行動計画をデジタルツインアプリケーション上から生成すること」を設定した。これは 5 章にて、Euclid Clustering を活用したロボットの推定手法を実装し、実験を行った。ここでは、本手法に対し、ある程度の効果はあるが、実行速度を高速化したり、また流動体に対する対策といった改善点が見つかった。

今回、二つの領域の課題達成に使用した五層モデルは、その汎用性を高めるために、レイヤーを五つに細分化し、今回のように複数領域にわたってもそれぞれ柔軟に対応することができるアーキテクチャであるとわかった。今までのアーキテクチャでは、Digital Layer に全てのデジタル情報が押し込まれ、煩雑であったところ、Cyber Layer の登場により、情報を取捨選択し、今回の課題達成の際に作成したデジタルツインアプリケーションの設計の煩雑化を避けられたと思われる。

今後は、各領域の章にて記述した課題を達成することを目指していく必要がある。

発表文献と研究活動

(1) **川村地平**, 青谷和真, 藤橋貞光, 稗田啓介, 中島高英, 塚田学, 江崎浩, ”生産管理のためのデジタルツインアプリケーション”, マルチメディア，分散，協調とモバイル（DICOMO2021）シンポジウム, オンライン, 2021.

(2) 青谷和真, **川村地平**, 塚田学, 落合秀也, 江崎浩, ”デジタルツイン基盤とその 3D データ活用”, マルチメディア，分散，協調とモバイル（DICOMO2021）シンポジウム, オンライン, 2021.

(3) 加藤慎, 曽根卓, 新麗, **川村地平**, 井口和真, 塚田学, ”超臨場 SDM 方式収録データセット”, Linked Open Data チャレンジ Japan 2020, 2020.

参考文献

[1] Erik Stolterman and Anna Croon Fors. Information technology and the good life. In Bonnie Kaplan, Duane P Truex, David Wastell, A Trevor Wood-Harper, and Janice I DeGross, editors, *Information Systems Research: Relevant Theory and Informed Practice*, pp. 687–692. Springer US, Boston, MA, 2004.

[2] 経済産業省 - デジタルトランスフォーメーションを推進するためのガイドライン（DX推進ガイドライン）を取りまとめました. `https://www.meti.go.jp/press/2018/12/20181212004/20181212004.html`. 2022 年 1 月 5 日閲覧.

[3] Michael Grieves. Origins of the digital twin concept, 08 2016.

[4] Edward Glaessgen and David Stargel. The digital twin paradigm for future NASA and U.S. air force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, Structures, Structural Dynamics, and Materials and Co-located Conferences. American Institute of Aeronautics and Astronautics, April 2012.

[5] Somayeh Malakuti and Sten Grüner. Architectural aspects of digital twins in IIoT systems. In *Proceedings of the 12th European Conference on Software Architecture: Companion Proceedings*, No. Article 12 in ECSA '18, pp. 1–2, New York, NY, USA, September 2018. Association for Computing Machinery.

[6] Koen Bruynseels, Filippo Santoni de Sio, and Jeroen van den Hoven. Digital twins in health care: Ethical implications of an emerging engineering paradigm. *Front. Genet.*, Vol. 9, p. 31, February 2018.

[7] Timo Ruohomäki, Enni Airaksinen, Petteri Huuska, Outi Kesäniemi, Mikko Martikka, and Jarmo Suomisto. Smart city platform enabling digital twin. In *2018 International Conference on Intelligent Systems (IS)*, pp. 155–161, September 2018.

[8] Aidan Fuller, Zhong Fan, Charles Day, and Chris Barlow. Digital twin: Enabling technologies, challenges and open research. *IEEE Access*, Vol. 8, pp. 108952–108971, 2020.

[9] Mario Hermann, Tobias Pentek, and Boris Otto. Design principles for industrie 4.0 scenarios. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 3928–3937, January 2016.

[10] National Science Foundation - Cyber-Physical Systems (CPS). https://www.nsf.gov/pubs/2011/nsf11516/nsf11516.htm. 2022 年 1 月 5 日閲覧.

[11] Shohin Aheleroff, Xun Xu, Ray Y Zhong, and Yuqian Lu. Digital twin as a service (DTaaS) in industry 4.0: An architecture reference model. *Advanced Engineering Informatics*, Vol. 47, p. 101225, January 2021.

[12] 経済産業省 - DX レポート ～IT システム「2025 年の崖」克服と DX の本格的な展開～. https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/20180907_report.html. 2022 年 1 月 5 日閲覧.

[13] Yu Zheng, Sen Yang, and Huanchong Cheng. An application framework of digital twin and its case study. *J. Ambient Intell. Humaniz. Comput.*, Vol. 10, No. 3, pp. 1141–1153, March 2019.

[14] Michael Grieves. Origins of the digital twin concept. August 2016.

[15] Michael W Grieves. Virtually intelligent product systems: Digital and physical twins. In *Complex Systems Engineering: Theory and Practice*, Vol. 256 of *Progress in Astronautics and Aeronautics*, pp. 175–200. American Institute of Aeronautics and Astronautics, Inc., January 2019.

[16] Michael Grieves and John Vickers. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In Franz-Josef Kahlen, Shannon Flumerfelt, and Anabela Alves, editors, *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, pp. 85–113. Springer International Publishing, Cham, 2017.

[17] NASA - Technology Area 12: Materials, Structures, Mechanical Systems, and Manufacturing Road Map. https://www.nasa.gov/pdf/501625main_TA12-MSMSM-DRAFT-Nov2010-A.pdf. 2021 年 1 月 7 日閲覧.

[18] Elisa Negri, Luca Fumagalli, and Marco Macchi. A review of the roles of digital twin in CPS-based production systems. *Procedia Manufacturing*, Vol. 11, pp. 939–948, January 2017.

[19] Michael Grieves. Digital twin: Manufacturing excellence through virtual factory replication. 03 2015.

[20] F Tao, H Zhang, A Liu, and A Y C Nee. Digital twin in industry: State-of-the-Art. *IEEE Trans. Ind. Inf.*, Vol. 15, No. 4, pp. 2405–2415, April 2019.

[21] 棚橋尚貴. 鋳造工場向けエネルギー利用効率化システム 『MiELCAST (みえるキャスト)』 の開発. エレクトロヒート, Vol. 32, No. 5, pp. 24–27, 2012.

[22] 地主岳史, 知崎一鉱, 川上裕介. IoT 活用による工場の生産活動最適化. 情報処理学会 デジタルプラクティス, 2017.

[23] 森武俊, 野口博史, 佐藤知正. センシングルーム. 日本ロボット学会誌, Vol. 23, No. 6, pp. 665–669, 2005.

[24] Egemen Ertugrul, Umurcan Kocaman, and Ozgur Koray Sahingoz. Autonomous

aerial navigation and mapping for security of smart buildings. In *2018 6th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, pp. 168–172, April 2018.

[25] Viktor Tihanyi, András Rövid, Viktor Remeli, Zsolt Vincze, Mihály Csonthó, Zsombor Pethő, Mátyás Szalai, Balázs Varga, Aws Khalil, and Zsolt Szalay. Towards cooperative perception services for ITS: Digital twin in the automotive edge cloud. *Energies*, Vol. 14, No. 18, p. 5930, September 2021.

[26] Matterport - Pro2 3D Camera. `https://matterport.com/cameras/pro2-3D-camera`. 2022 年 1 月 9 日閲覧.

[27] R Shults, E Levin, R Habibi, S Shenoy, O Honcheruk, T Hart, and Z An. CAPABILITY OF MATTERPORT 3D CAMERA FOR INDUSTRIAL ARCHAEOLOGY SITES INVENTORY. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 2019.

[28] 川村地平, 青谷和真, 藤橋貞光, 稗田啓介, 中島高英, 塚田学, 江崎浩. 生産管理のためのデジタルツインアプリケーション. マルチメディア, 分散, 協調とモバイル（DICOMO2021）シンポジウム, 2021.

[29] Õrjan Ljungberg. Measurement of overall equipment effectiveness as a basis for TPM activities. *&ctx_ver=Z*, Vol. 39, p. 88, May 1998.

[30] Radu Bogdan Rusu and Steve Cousins. 3D is here: Point cloud library (PCL). In *2011 IEEE International Conference on Robotics and Automation*, pp. 1–4, May 2011.

[31] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, Vol. 24, No. 6, pp. 381–395, June 1981.

[32] Radu Bogdan Rusu. Semantic 3D object maps for everyday manipulation in human living environments. *KI - Künstliche Intelligenz*, Vol. 24, No. 4, pp. 345–348, November 2010.

謝辞

本論文の執筆に当たり, 非常に多くの方々からのご指導, ご協力を頂きました. 心より感謝申し上げます.

指導教官である塚田学准教授には, お忙しい中, 温かいご指導, ご鞭撻を頂きました. 深く感謝申し上げます.

合宿などの場において，様々なアドバイスや指摘をくださった江崎浩教授，落合秀也准教授に深く感謝申し上げます.

工場に関するデジタルツイン構築の研究を行った際に, 様々なデバイス, サーバーの手配やデジタルツイン構築のサポートをしてくださった株式会社シムックスイニシアティブの皆様には大変お世話になりました. 深く感謝申し上げます.

そして, 東京大学グリーン ICT プロジェクト BIM 基盤 WG の皆様, 粕谷貴司氏, 実証実験を行う際に手順などを取りまとめてくださったエヌ・ティ・ティ・コミュニケーションズ株式会社の皆様には, ビルに関するデジタルツイン構築の研究を行った際に大変お世話になりました. 深く感謝申し上げます。

先輩として研究生活に慣れることへのサポート, 研究や技術面の助言などを行ってくださった, 加藤慎氏, 辻尾康平氏には, 大変お世話になりました. 感謝いたします.

また, 同分野で研究を行っている青谷和真氏, 井口和馬氏には, 多くの協力, 助言をしていただきました. 感謝いたします.

研究生活を様々な面から支えてくださった岩井愛映子秘書, 高橋富美秘書に感謝いたします.

最後に, これまで私を育て, 支えてくださった両親と兄に深く感謝申し上げます.

付録 A

工場可視化ツールの使用手順を説明したプレゼンテーション資料



1

工場可視化ツール – サンプルお試し手順

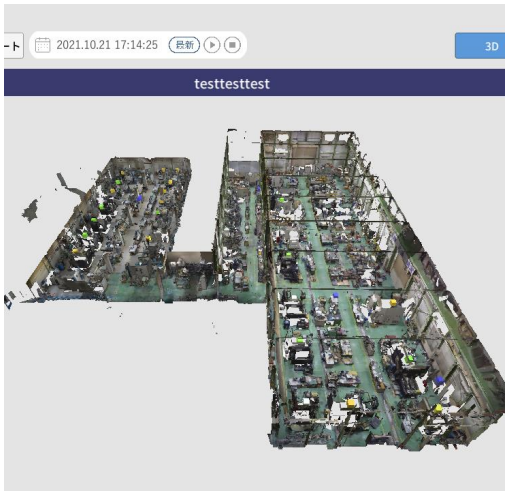
1. アプリケーションURLにアクセスしてください
   - https://ut.energy-management.jp:12443/FAMonitor/
2. 様々な機能をお試しください（試してほしい機能は以下の通りです）
   - 現在の工作機械の稼働状況の確認
   - タイムラプス機能
   - 稼働状況/電力消費のグラフ化
   - etc.
3. アンケートフォームにご回答ください
   - https://forms.gle/38h9BqxosRCe4j9y5

※ アンケートのご回答は匿名データとして扱い、大学の修士論文の中で利用する予定です

2

使用方法 – 3D画面移動

- 右の画像は起動時の画面です
- マウスクリック、又はタッチ+スライドにて工場を見回すことが出来ます
- マウスホイールにてズームすることが出来ます



3

使用方法 – 稼働状態表示

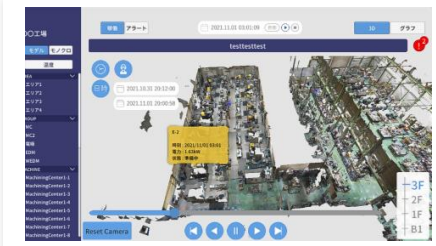
- 工場内に点在する立方体（右画像内のオレンジ枠参照）にマウスカーソルを合わせるかタッチすると選択対象の現在の電力消費量や稼働状況などがわかります
- 画面上部の稼働/アラートタブにて表示される情報を変更することが出来ます(アラートタブでは工作機械に設置されたパトランプの情報を表示)

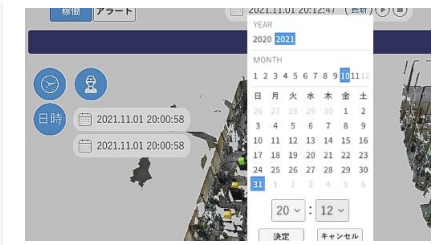


4

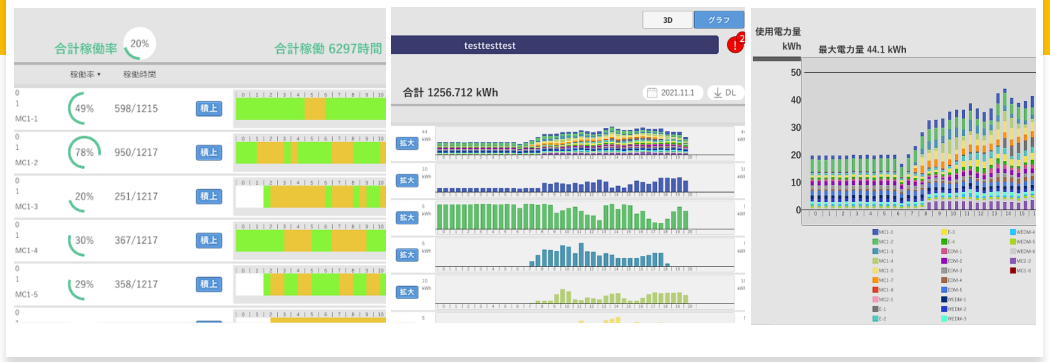
使用方法 – タイムラプス

1. 画面上部の時計マークをクリックすると日時というボタンが表示されます
2. 更にそのボタンを押すと二つの時間設定ボタンが表示されます
3. 上部の時間と下部の時間をそれぞれボタンをクリックすることで、右上の画像のようにカレンダーが表示され、時間を設定できます(ダウンロードに時間がかかるため、今回は1週間程度の範囲指定でお願いします)
4. 時間設定を行った後、画面下部の再生ボタンを押すことで画面上部の時間表示が現在の時刻からタイムラプス中の時刻に変化します
5. 画面下部のスライドバーにて、再生中の時刻を変更できます
6. 再生中に工場内の立方体にカーソルを合わせると、工作機械の過去の稼働状態を表示します





5



使用方法 – グラフ

- 画面右上のグラフタブをクリックすることで、その日の工作機械の稼働状況や工作機械の種類ごとの電力消費割合などが見れます
- 画面右上のカレンダーボタンをクリックすることで日、週、月の区切りでこれらの情報をまとめてみることが出来ます
- またそれぞれの項目の積上/拡大ボタンを押すと詳細情報を見ることが出来ます

6

機能一覧

- エリア、グループ分け / 工作機械、IoTデバイスリスト
- 3D機能
  - 現在の稼働状況 / 電力消費量の確認
  - 工作機械のパトランプの警告情報
  - 温度のヒートマップ表示
  - タイムラプスによる過去の情報の確認
- グラフ
  - 稼働率を示したガントチャート
  - 電力消費を示した積み上げ棒グラフ
  - 過去のデータの確認
  - CSVダウンロード

7

図 A.1. 工場可視化ツールの使用手順を説明したプレゼンテーション資料

付録 B

スマートビルディングにおけるデジタルツインアプリケーションの各種機能説明

GUTP
デジタルツインアプリケーション
使用説明書

v0.2

責任者：川村 地平
kawamura@hongo.wide.ad.jp

1

概要

- 試験シナリオ1~7までに対応
- IFCは”NTTコム様 検証ラボ6_20211104_IFC4Design.ifc”を使用

2

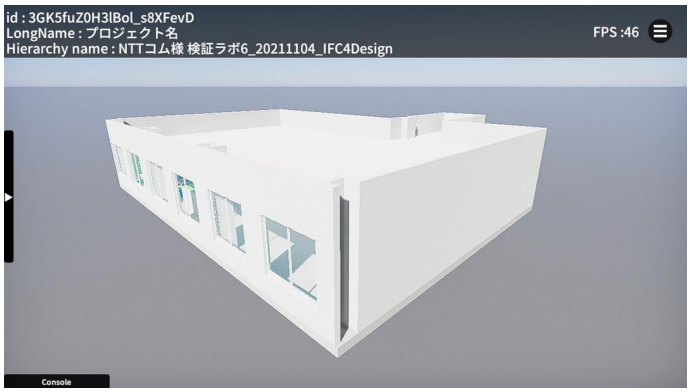
目次

- カメラ移動について
- 各種表示項目について
- メニュー項目について
- 試験シナリオ1~7までの実行方法について
- 実装済み機能について(概略)

3

カメラ移動について(基本操作)

- マウスホイール ＋ マウス移動 ： 上下左右に移動
- マウス右クリック ＋ マウス移動 ： 回転
- マウスホイール ： ズーム/ズームアウト



起動時の画面

4

カメラ移動について(簡易操作)

- 画面左の黒い線にカーソルを合わせると、カメラを事前に用意された位置に移動させることができる簡易移動ボタンが利用可能



“南東”を押した場合

5

各種表示項目について

- ①：上から順に IFCID, IFC_LongName, ファイル名
- ②：フレームレート
- ③：メニューを開くボタン
- ④：APIとの通信結果などの各種ログ



6

メニュー項目について

- ここでは、右図の赤線で囲ったボタンをクリックした際に表示されるメニュー項目について、内容と動作の説明を行う

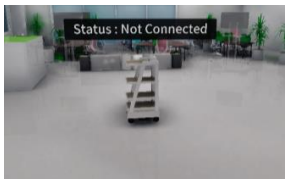


ハンバーガーメニュー

7

メニュー項目について

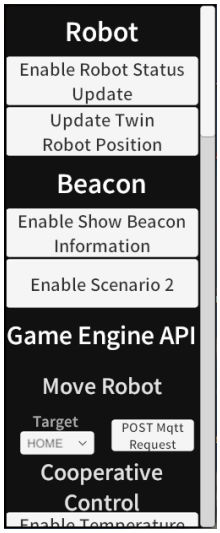
- Robot
  - Enable Robot Status Update
    - フルテラの運搬ロボットの状態をMQTTから取得し、デジタルツイン内に反映させる
  - Update Twin Robot Position
    - 後述するLiDARを接続した際に使用するシミュレーション用ロボットの位置をMQTTで取得したロボットの位置に合わせる
- Beacon
  - Enable Show Beacon Information
    - MQTTに接続し、ビーコンの情報を受け取った場合、該当するエリアにポップアップとして表示する
  - Enable Scenario2
    - “Enable Show Beacon Information”がONになっている状態でこちらもONにすると、”Area:4004”かつ”Target:Point_C”となっているビーコン情報を受け取った際に運搬ロボットに”C”へ移動するように伝える



MQTTと未接続な状態



ビーコン情報のポップアップ



メニュー項目

8

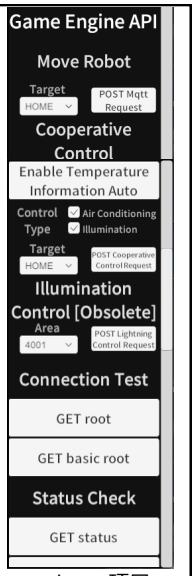
メニュー項目について

- Game Engine API (Move Robot)
  - POST MQTT Request
    - 左にあるTargetのドロップダウンメニューにて指定した位置に運搬ロボットを移動させる指令をMQTTにて送る
- Game Engine API(Cooperative Control)
  - Enable Temperature Information Auto
    - 各エリアごとの温度情報を定期的に取得、表示する
  - POST Cooperative Control Request
    - 空調と照明をTargetを指定し、操作する
    - チェックボックスがついている物を操作する
    - Targetはドロップダウンメニューとなっている
    - 照明を制御した場合、事前に設定されたシーン情報に基づいて、昼光色、昼白色、電球色など照明の色合いを変更する



温度情報の表示


照明の反映



メニュー項目

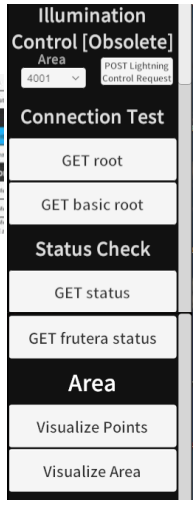
メニュー項目について

- Game Engine API(Illumination Control)
  - POST Lightning Control Request
    - エリアを指定して行う照明制御
    - 左のドロップダウンメニューにてエリアを指定する
- Game Engine API(Connection Test)
  - GET root / GET basic root
    - Game Engineとの接続確認を行う(Consoleにて確認)
- Game Engine API(Status Check)
  - GET status / GET frutera status
    - ビーコン情報やロボットの情報を得るMQTTとの接続確認を行う(Consoleにて確認)
- Area
  - Visualize Points / Visualize Area
    - 事前に割り当てられたロボットの移動指示に用いる目標点やエリアを可視化する


目標点の可視化



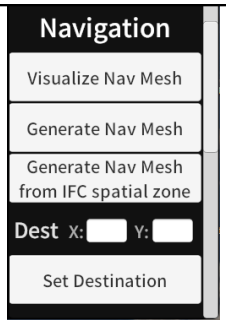
エリアの可視化



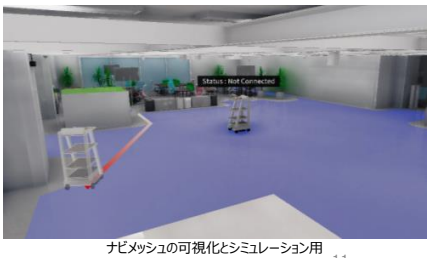
メニュー項目

メニュー項目について

- Navigation
  - Visualize Nav Mesh
    - アプリケーション内でロボットのモデルが仮想空間内での移動経路の算出に用いる移動可能な領域を描画する
  - Generate Nav Mesh
    - ナビメッシュを生成する(現在、これとは別に自動で生成している)
  - Generate Nav Mesh from IFC spatial zone
    - IFC内に定義されたIFC Spatial Zoneに基づいて、その空間内に限定してナビメッシュを生成する
  - Set Destination
    - 後述するLiDARを接続した際に使用するシミュレーション用ロボットの位置を上部の”Dest”の入力可能フィールド”X”,”Y”に入力された数値に基づいて移動する



メニュー項目

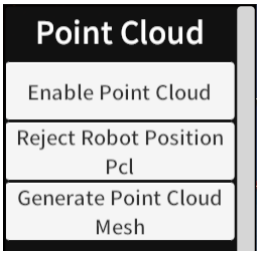

ナビメッシュの可視化とシミュレーション用ロボットの移動経路の可視化

11

メニュー項目について

- Point Cloud
  - Enable Point Cloud
    - ROSbridge(ws://localhost:9090)に接続を行い、”/velodyne_points”に点群データがパブリッシュされていた場合、それをアプリケーション内に可視化する
    - 先述していたシミュレーション向けロボットのモデルを表示する(通常のモデルと違い、半透明で表示される)
  - Reject Robot Position Pcl
    - MQTTで取得したロボットの位置近辺の点群を障害物ではないと仮定（ロボット本体の点群である可能性が高いため）し、経路探索の際に利用しない設定項目
  - Generate Point Cloud Mesh
    - 与えられた点群データを仮想空間内で障害物として利用する


シミュレーション用ロボットモデル



メニュー項目

12

メニュー項目について

- IFC
  - Change IFC Spatial Zone Highlight
    - IFC内に定義されたIFC Spatial Zoneを可視化する
  - Activate Ceiling
    - IFC内に定義された階層構造の定義を含むオブジェクトの内、一定以上の高さの物を表示/非表示を変更する(ここでは、天井の表示/非表示を変更する)
- IFC API
  - GET IFC Geometry
    - IFC幾何形状データ管理サービスにアクセスし、形状情報を指定した”Model ID”や”IFC Class”を引数として検索し、取得する(Consoleにて確認)



IFC Spatial Zoneの可視化



天井の非表示

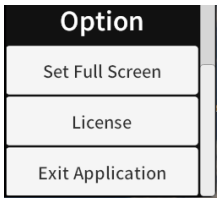


メニュー項目

13

メニュー項目について

- Option
  - Set Full Screen
    - フルスクリーンとウィンドウ表示を切り替える
  - License
    - 使用している外部スクリプトのライセンスを確認する
  - Exit Application
    - アプリケーションを終了する



メニュー項目

14

試験シナリオ1～7までの実行方法について

- ここでは以下の試験シナリオにおいて、ゲームエンジン側でどのような操作を行えばシナリオで期待されている動作を達成できるか記述する
  - 試験シナリオ1 ： ゲームエンジンによるロボット制御
  - 試験シナリオ5 ： VIP判断及びロボット制御
  - 試験シナリオ6 ： ゲームエンジンによる照明制御
  - 試験シナリオ7 ： ロボット移動による照明制御

シナリオ1 ： ゲームエンジンによるロボット制御

- 概要
  - ゲームエンジンから”HOME”,”Point_A”, ”Point_B”,”Point_C”何れかの移動指示を出す
- 操作方法
  - 右のサイドメニューを開き、Robotの項目の内、”Enable Robot Status Update”を押す
    - これを行うことで、定期的にMQTTを用いた通信を行い、運搬ロボットの現在の状態を取得、アプリケーション内に反映する
  - Game Engine API / Move Robotの項目の内、”Target”を押し、ドロップダウンメニューから移動させたいポイントを指定する
  - その後、右側の”POST Mqtt Request”を押すとその場所への移動指示が送信される



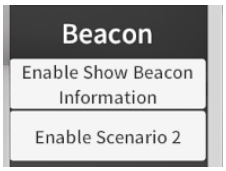
MQTTからの情報をアプリケーション内にて確認(ロボットが移動中)



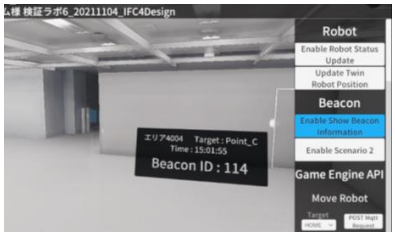
メニュー：Game Engine API / Move Robot

シナリオ5：VIP判断及びロボット制御

- 概要
  - ビーコンがエリア4004(Point_C)に到達した際に、ゲームエンジンが”Point_C”へのロボットの移動指示を出す
- 操作方法
  - 右のサイドメニューを開き、Robotの項目の内、”Enable Robot Status Update”を押す
  - Beaconの項目の内、”Enable Show Beacon Information”を押す
    - これを押すとMQTTを用いて、ビーコン情報を常に受信するようになり、受信した場合、右の図のように情報を仮想空間内に表示する
  - Beaconの項目の内、”Enable Scenario 2”を押す
    - 上記の” Enable Show Beacon Information”が押されている時に、これを押すと”Area：4004”, “Target：Point_C”の情報を含んだビーコンの到達情報が取得された際に、運搬ロボットに対し、”Point_C”への移動指示を送る



メニュー：Beacon

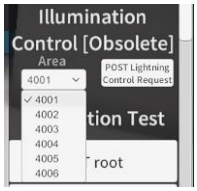


ビーコン情報のポップアップ

17

シナリオ6：ゲームエンジンによる照明制御

- 概要
  - ゲームエンジンからエリア4001~4006何れかの照明制御指示を送信する
- 操作方法
  - 右のサイドメニューを開き、Game Engine API / Illumination Controlの項目の内、”Area”を押し、ドロップダウンメニューからエリア番号を指定する
  - その後、右側の”POST Lightning Control Request”を押すとそのエリアの照明制御指示が送信される
  - この時、シーン情報に基づいて、右図のように照明の変化が仮想空間内で再現される
    - なお、シーン情報は”20211028_[GUTP]ユースケース説明資料_v1.1.pdf”の”3.各シナリオ概要とテスト方法”/”3.6 – シナリオ⑥(ゲームエンジンによる照明制御)のテスト方法(2/4)”に記載されている”エリア情報とシーンの相関表”に基づいている



メニュー：Game Engine API / Illumination Control



照明の反映

18

シナリオ7 ： ロボット移動による照明/空調制御

- 概要
  - ゲームエンジンからロボットへ移動指示を送信し、照明/空調制御を行う
- 操作方法
  - 右のサイドメニューを開き、Robotの項目の内、”Enable Robot Status Update”を押す
    - これは後述する照明の変化を仮想空間内に反映するために必要
  - Game Engine API / Cooperative Controlの項目の内、”Enable Temperature Information Auto”を押す
    - これを押すことにより右図のように温度情報を仮想空間内にてエリア番号と共に見ることが出来る(1分置きに情報更新)
  - Game Engine API / Cooperative Controlの項目の内、”Control Type”から制御したい項目を指定する(チェックボックスを押し、制御するか否か選択する)
  - その後、下側の”Target”を押し、ドロップダウンメニューからロボットの移動先を指定する
  - 最後に、右側の”POST Cooperative Control Request”を押すことで、ロボットの移動による照明/空調制御を行う
    - この方法による照明制御を行っても、シナリオ6と同じように照明の変化が仮想空間内に再現される
      - この時、照明の変化はMQTTから取得したロボットの位置情報に基づいて、ロボットが存在するエリアに応じた照明シーンを再現する



メニュー：Game Engine API / Cooperative Control

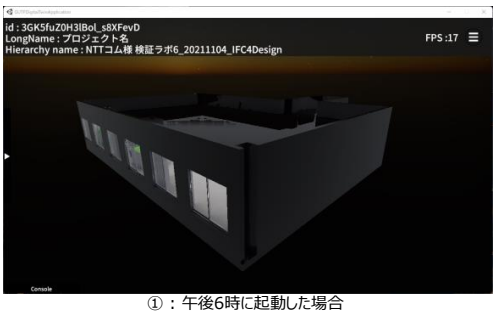


温度情報の表示

19

実装済み機能について

- ①：アプリケーションを実行する時間に応じて太陽光の向きや色合いが変わる
- ②：フルスクリーンへの変更
- ③：使用外部データのライセンス表記
- ④：~~観葉植物の葉が揺れる機能~~ (動作に負荷がかかるので削除)



①：午後6時に起動した場合

20

図 B.1. スマートビルディングにおけるデジタルツインアプリケーションの機能説明資料