

三次元データを用いた土砂災害対策の推進 技術検証レポート

Technical Report for Landslide Management System with 3D data



PLATEAU
by MLIT



目次

I. 実証概要	
1. 全体概要	3
2. 実施体制	5
3. 実証エリア	6
4. スケジュール	7
II. 実証技術の概要	
1. 活用技術	9
2. Symmetry Digital Twin Cloud	10
3. ARCore	11
4. LAStools	13
5. FME Desktop	15
6. CloudCompare	16
III. 実証システム	
1. 実証フロー	18
2. 想定事業機会	19
3. アーキテクチャ全体図	22
4. システム機能	24
5. アルゴリズム	38
6. データ	
① 活用データ	44
② データ処理	50
③ 出力データ	62
7. ユーザインタフェース	65
8. システムテスト結果	73
IV. 実証技術の検証	
1. 被害住居検出の精度検証	
① 検証内容	76
② 検証結果	83
2. ARアプリにおける自己位置推定の精度検証	
① 検証内容	84
② 検証結果	87
3. 開発システムの価値検証	
① 検証内容	90
② 検証結果	92
V. 成果と課題	
1. 今年度の実証で得られた成果	
① 3D都市モデルによる技術面での優位性	97
② 3D都市モデルによるビジネス面での優位性	98
2. 今後の取り組みに向けた課題	99
用語集	101

I. 実証概要

II. 実証技術の概要

III. 実証システム

IV. 実証技術の検証

V. 成果と課題

I. 実証概要 > 1. 全体概要

全体概要 (1/2)



ユースケース名	三次元データを用いた土砂災害対策の推進
実施場所	静岡県掛川市
目標・課題 ・創出価値	- 地方公共団体が保有する住民情報と3D都市モデルの建築物を紐づけることで、迅速かつ的確な災害対応への支援を行うことが目標である。 - 多くの自治体では土砂災害等が発生した際の被害状況を現地調査で確認しており、流出土砂の下にどの程度の家屋が存在し、行方不明者がどの程度発生し得るのか、といった情報の迅速な把握に課題がある。 - 住民情報、被害家屋情報、土砂点群などのデータが連携されておらず、災害は発生した際にも被害想定を迅速に分析できなかった事象が熱海災害などで発生している。 - 今回の実証実験では、迅速かつ的確な災害対応を行うことで、人命の迅速な救助や負傷の軽減などの実現を狙う。
ユースケース の概要	- 自治体の持つ住民情報を3D都市モデルに統合し、ドローン等を用いて取得した流出土砂の3D測量データを用いた解析を行うシステムと、AR上で可視化するモバイル向けのアプリケーションを開発する。 - ドローンによって測定可能な屋根に注目し、建物の被災有無の判定を行うアルゴリズムを開発する。 - 被害住居検出システムで検出された被災住居をスマートフォンのAR機能を用いて現実空間に投影を行うことで、災害救助活動の迅速化を支援するアプリケーションを開発する。 - 災害被害を受けた家屋等にどの程度の世帯が存在し得るのかを即座に解析でき、現地での救助オペレーションに活用できる。

I . 実証概要 > 1. 全体概要

全体概要（2/2）



実証仮説	• 建築物等の位置と三次元的な形状をデータとして保持している3D都市モデルと、流出土砂の3D測量データをシステム上で連携させることで、家屋の被害状況を迅速に把握できる。 • 3D都市モデルと点群データを用いた被害住居検出システム・ARによる可視化システムを実現することで、災害時被害状況の迅速な把握ができ、救助・調査活動の迅速化に繋がる。 • 3D都市モデルとドローンにより取得した点群データを比較することで、被災した建物と正常な建物を判別でき、地方公共団体の応急対策や救助等における有用性がある。
検証ポイント	• 被害住居検出システムの精度検証 • ARアプリにおける自己位置推定精度の検証 - VPSの精度検証 - 手動位置合わせ後のSLAM利用時の精度検証 • 開発システムの有用性検証 - 被害住居検出システムの有用性ヒアリング - ARアプリの有用性ヒアリング

I. 実証概要 > 2. 実施体制

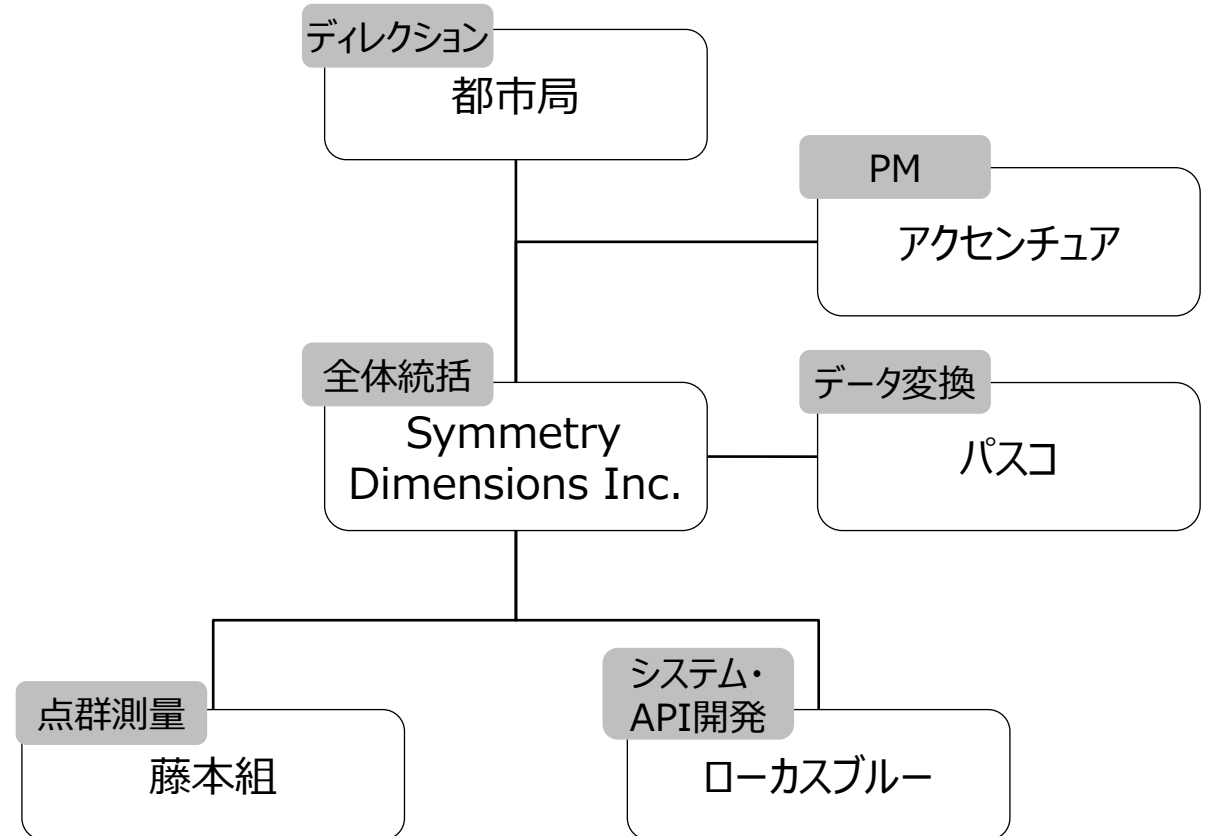
実施体制

災害対応を行う自治体、消防と連携し、本ユースケースの有用性を検証する

各主体の役割

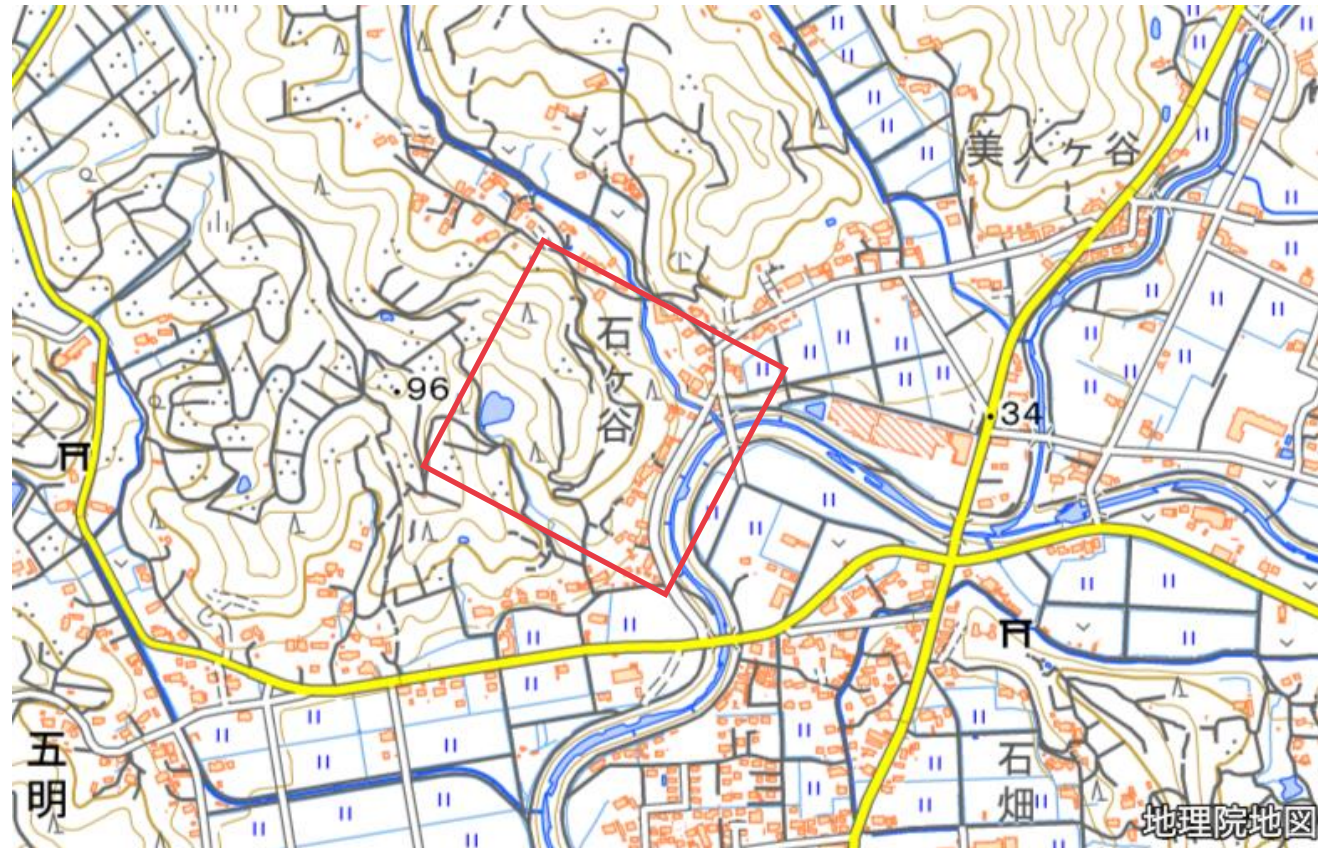
主体	役割
Symmetry Dimensions Inc.	- プロジェクトの進行管理 - Web・ARシステム開発
パスコ	掛川市住民基本台帳のデータ変換業務
ローカスブルー	- 被害住居検出システム開発業務 - API開発業務
藤本組	UAV LiDARによる点群測量業務
アクセンチュア	プロジェクトマネジメント

実施体制図



I. 実証概要 > 3. 実証エリア 実証エリア

静岡県掛川市 上西郷石ヶ谷地域（点群撮影エリア面積：0.3km²）



I . 実証概要 > 4. スケジュール スケジュール

実施事項	令和4年										令和5年		
	3月	4月	5月	6月	7月	8月	9月	10月	11月	12月	1月	2月	3月
1. 災害時の被害状況把握方法の整理		←→											
2. 住民基本台帳データ整備				←→									
3. 被災再現データの作成				←→									
4. 被災判定機能の開発								←→					
5. Webシステム開発						←→							
6. Web API開発								←→					
7. ARアプリ開発								←→					
8. 実証実験											←→		
9. 報告書作成												←→	

I. 実証概要

II. 実証技術の概要

III. 実証システム

IV. 実証技術の検証

V. 成果と課題

Ⅱ. 実証技術の概要 > 1. 活用技術

活用技術



項目	内容
Symmetry Digital Twin Cloud	• 様々なGISデータを組み合わせてWeb上で閲覧できるデジタルツインプラットフォーム
ARCore	• Google社が提供するARソフトウェア開発フレームワーク - モーションセンサーによるSLAM（自己位置推定）機能により、ARコンテンツを現実世界にフィット - ARCoreに含まれるGeospatial APIによるカメラ画像の分析により、さらに高い精度でコンテンツ位置を調整可能
LAStools	• rapidlasso社が提供している様々な点群処理を行うことができるソフトウェア
FME Desktop	• Safe Software社が提供しているGISやCAD等の様々な空間情報に関するデータを変換できるソフトウェア
CloudCompare	• 3D点群データを編集するためのオープンソースソフトウェア

Ⅱ. 実証技術の概要 > 2. Symmetry Digital Twin Cloud

Symmetry Digital Twin Cloud (SDTC) について

様々なGISデータを組み合わせてWeb上で閲覧できるデジタルツインプラットフォーム

概要	画面例
項目	詳細
名称	Symmetry Digital Twin Cloud
概要	• Symmetry社にて開発・提供しているデジタルツインクラウドサービス • 様々なデータの変換および閲覧が可能
主な機能	• プロジェクトによるメンバー／データ管理 • 点群や3Dモデル等の3D Tiles変換 • 3Dプレゼンテーションの作成および共有
本ユースケースで利用する機能	• 点群データの3D Tilesへの変換 • 3D都市モデルと点群データを組み合わせた3Dプレゼンテーションの作成 • 被災住宅検出APIの実行・結果取得・結果表示（カスタマイズ機能） • 住民情報表示（カスタマイズ機能）



Ⅱ. 実証技術の概要 > 3. ARCore

ARCore

Google社が提供するARソフトウェア開発フレームワーク

概要

項目	詳細
名称	ARCore
概要	- Google社が提供するAR（拡張現実）ソフトウェア開発用フレームワーク - Android OS 7.0以降対応 - iOS 11.0以降対応
主な機能	自己位置推定（SLAM）／水平面検出／マーカ－検出／深度マップ作成及びARオブジェクトへの遮蔽効果／環境光推定及びARオブジェクトへの反映／現実世界へのアンカー作成／顔検出及びマッピング
本ユースケースで利用する機能	自己位置推定（SLAM）：モーションセンサー及びカメラ画像によるスマートフォンデバイスによる自己位置推定を行う機能の利用

自己位置推定機能（SLAM機能）



画像： <https://developers.google.com/ar/develop/getting-started?hl=ja>

自己位置推定機能(SLAM機能)：
スマートフォン内蔵のモーションセンサーによりスマートフォンのカメラが向いている位置や方向を検出することで、スマートフォンの移動に合わせてARオブジェクトを描画することができる機能

Ⅱ. 実証技術の概要 > 3. ARCore

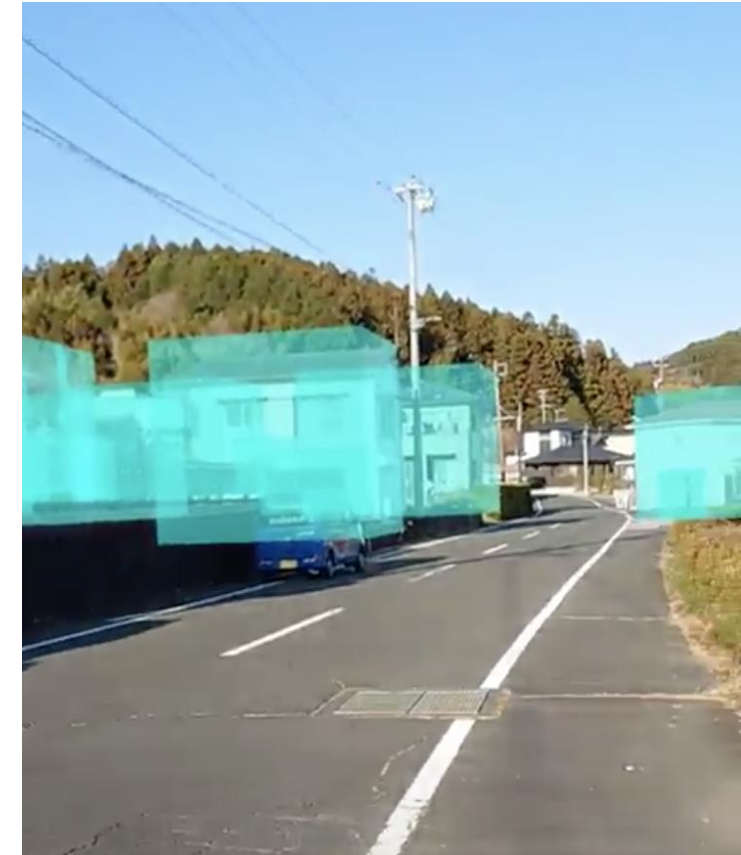
ARCore Geospatial APIについて

Google社が提供しているARCoreに含まれるカメラ画像を使った自己位置推定機能（VPS = Visual Positioning System）

概要

項目	詳細
名称	ARCore Geospatial API
概要	- Google社が提供するARCore用の画像による自己位置推定機能（VPS=Visual Positioning System） - 対応OSはARCoreに準ずる
主な機能	- カメラ画像から位置を推定するVPS機能 - 位置推定はGoogle Street View 画像提供範囲に限られる - 座標系はWGS84（緯度・経度・高度（楕円体高））
本ユースケースで利用する機能	カメラ画像からGoogleクラウドサーバ側で自己位置を推定し、緯度経度高さ情報を返すVPS機能を使用

カメラ画像による自己位置推定結果を利用したARイメージ



Ⅱ. 実証技術の概要 > 4. LAStools

LAStoolsについて

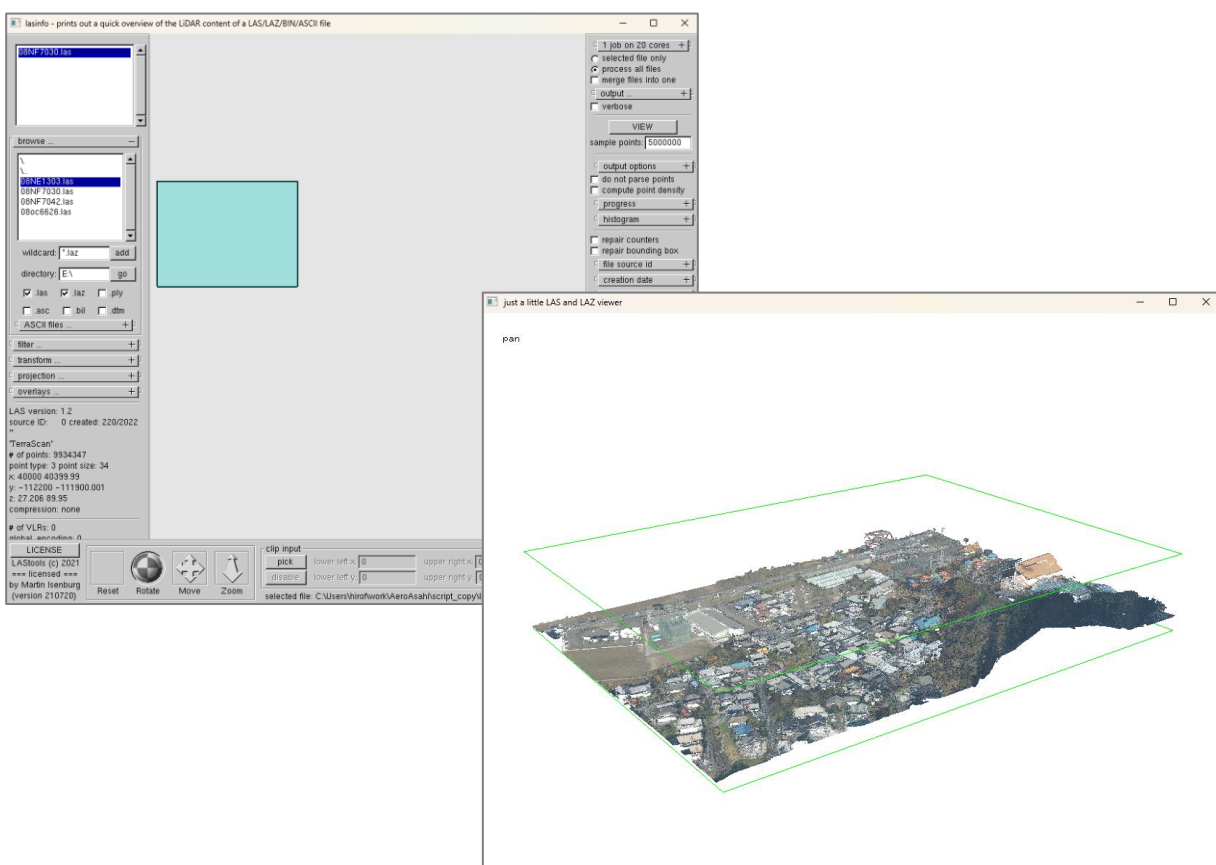
rapidlasso社が提供している様々な点群処理を行うことができるソフトウェア



概要

項目	詳細
名称	LAStools
概要	rapidlasso社が提供している点群処理ソフトウェア
主な機能	- 点群データの統計解析 - 点群データの変換 - 点群データのフィルタリング - 点群データのラスターデータ（DEM）化
本ユースケースで利用する機能	- ラスターデータ化、フィルタリング（地表面、最外形） - blast2dem：点群をDEM形式（ラスターデータ）に変換 - lasground：点群データから地表面を抽出 - lasboundary：点群データの差異外形をShapefile形式で出力

画面例



Ⅱ. 実証技術の概要 > 4. LAStools

LAStoolsによる点群の分類

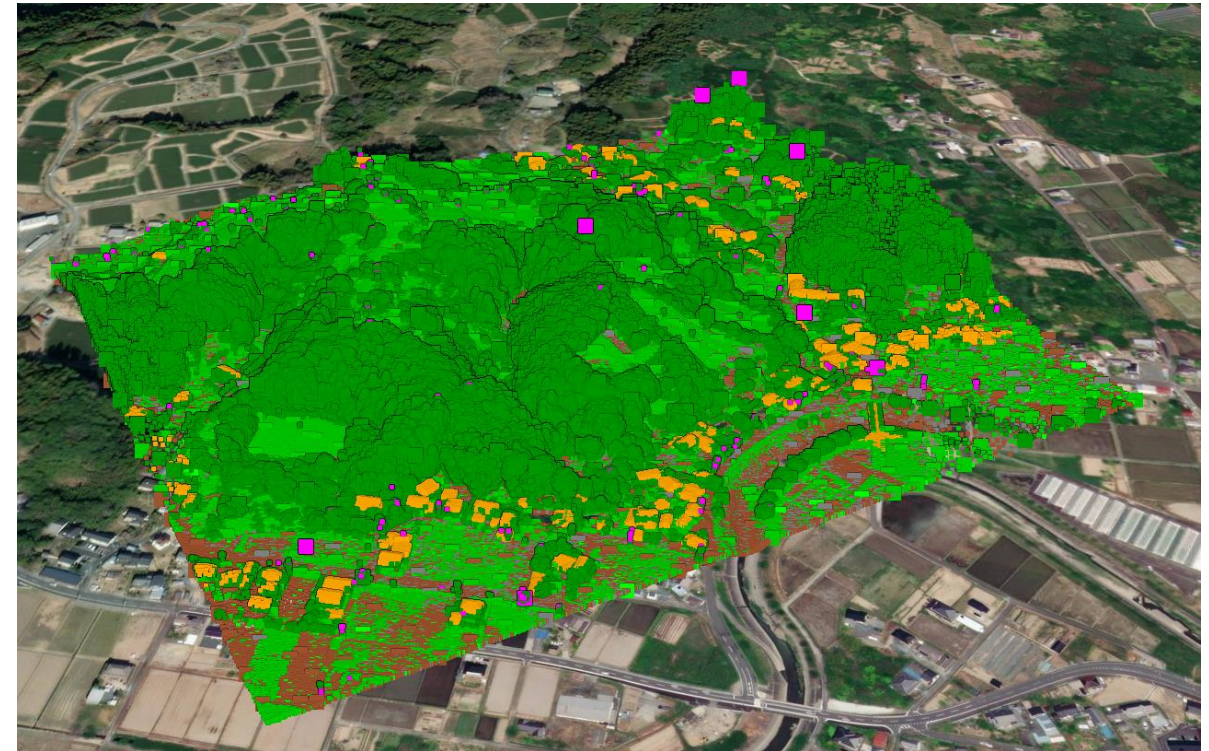
点群データを地表面・植生・建物などの7種に分類することが可能

LAStoolsによる点群の分類

- LAStoolsに含まれる複数のツールを組み合わせることで点群データを以下の7分類に判別することが可能

分類項目	判定ツール	右図における着色例
地表面	lasgoround	
植生（低）	lasheight	
植生（中）		
植生（高）		
建物	lasclassify	
ノイズ	lasnoise	
未分類	lastools	

分類イメージ

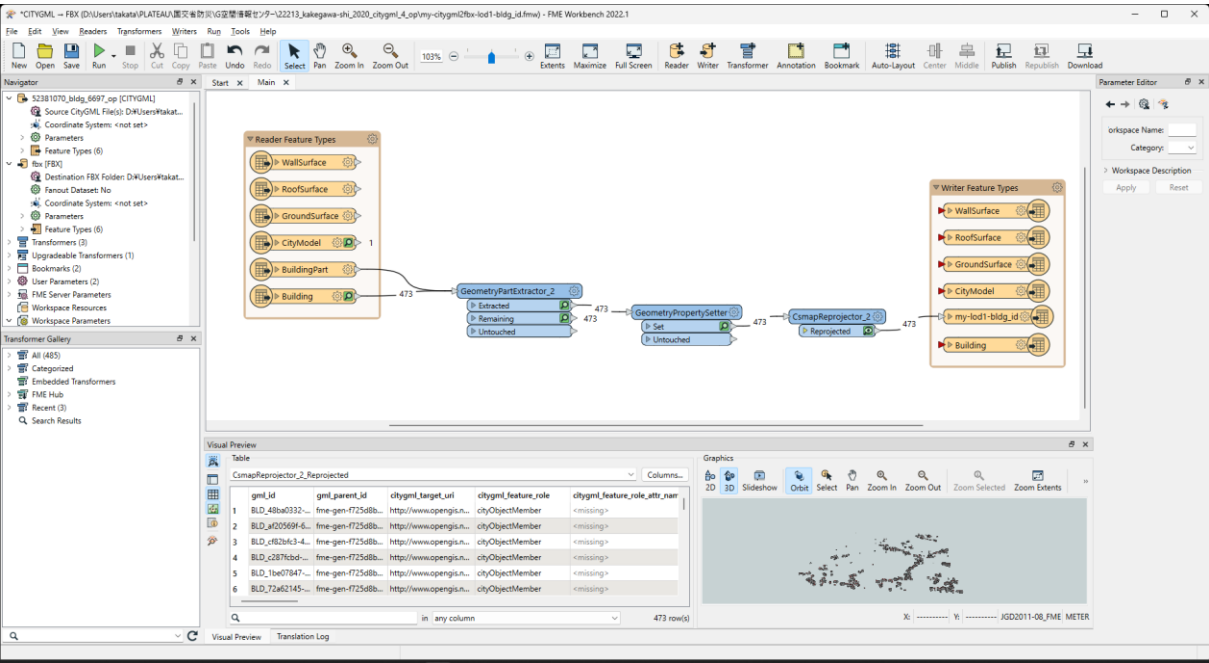


Safe Software社が提供しているGISやCAD等の様々な空間情報に関するデータを変換できるソフトウェア

概要

画面例

項目	詳細
名称	FME Desktop
概要	- Safe Softwareが提供するデータ変換ソフトウェア - GISやCAD等の空間情報、XMLやCSV等の非空間情報を統合して処理することが可能
主な機能	- 多様なモジュールによるワークスペースの作成、実行 - Reader：データの読込 - Transformer：データの変換 - Writer：データの出力 - データの確認、プレビュー
本ユースケースで利用する機能	- CityGML形式のFBX形式への変換 - CityGMLの建物IDプロパティをFBX内部のノード名として保存 - 座標参照系の変換



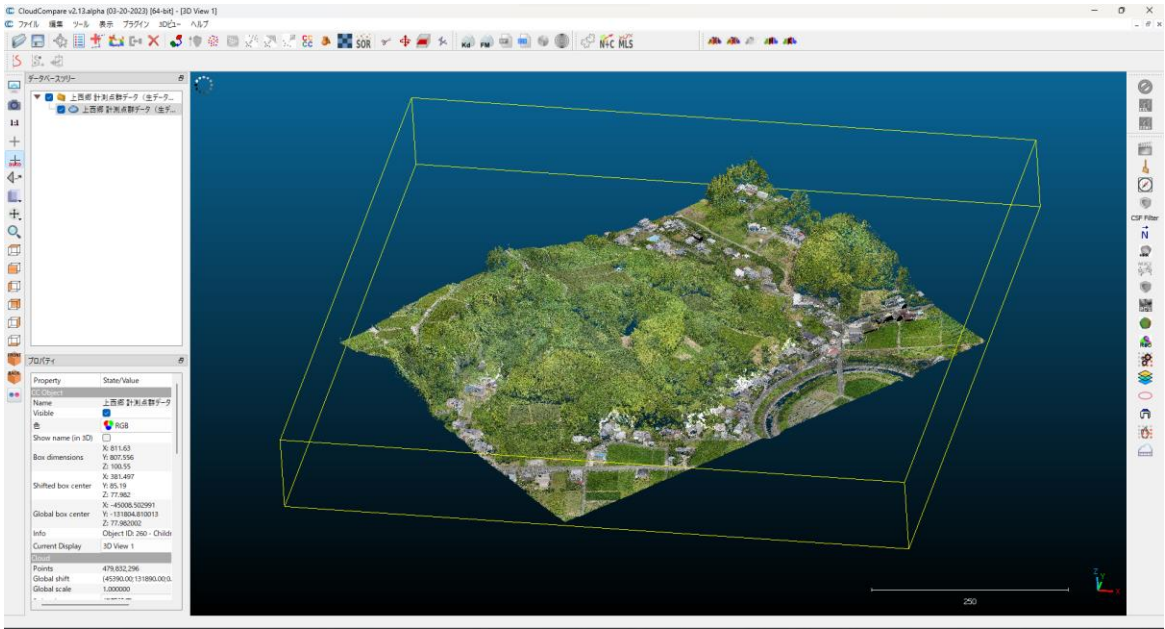
Ⅱ. 実証技術の概要 > 6. CloudCompareについて

3D点群データを編集するためのオープンソースソフトウェア

概要

項目	詳細
名称	CloudCompare
概要	3D点群データとメッシュデータの処理のためのオープンソースソフトウェア
主な機能	- 点群データの読み込みと表示 - 点群データの切り取り - 点群データの移動 - 点群データのフィルタリング
本ユースケースで利用する機能	- 点群データの切り取り - 点群データの移動

画面例



I. 実証概要

II. 実証技術の概要

III. 実証システム

IV. 実証技術の検証

V. 成果と課題

Ⅲ. 実証システム > 1. 実証フロー

実証フロー

本実証実験は下記のフローに沿って実施し、システム精度や有用性の検証を行う

調査・ヒアリング

- 土砂災害による住居移動パターンの調査
- 従来の災害時被害状況調査方法に関するヒアリング
 - ヒアリング先：静岡県防災課、静岡県掛川市危機管理室、掛川市消防局

点群データの取得 及び編集

- ドローンを用いた対象エリアの点群データ撮影
- 土砂災害による被害状況を再現のため、住宅点群データの編集

システム開発

- 被害住居検出システムの開発
- ARアプリケーションの開発
- システム間連携のためのAPI構築

被害住居検出検証 ARアプリ動作検証

- 点群データを編集し再現した4つの被害住居パターン検出精度の検証
- VPSを活用したARアプリケーションの動作検証
- 2次元地図データを用いた位置情報補正検証

システムの有用性検証

- 自治体関係者および消防局員にシステムの実演を行った後、ヒアリングを行い、本システムの実務上の有用性を評価

Ⅲ. 実証システム > 2. 想定事業機会

想定事業機会



自治体向けの災害時の状況把握支援ツールとしての活用を想定

項目	内容
利用事業者	自治体
提供価値	- 災害発生時における被害状況把握を早期化 - 通常1-2か月かかる被害状況の把握作業を、遠隔で安全に且つ効率的に実行可能 - 現地での実地確認が難しい状況でも、過去データとの突合などにより差分分析が可能 - 災害現場での被災住居特定の迅速化 - 分析結果に基づくARでの可視化により現場での被災住居の特定が可能
サービス仮説	- 被害状況の遠隔調査サービス - 誰もが3Dデータを活用できるよう対策マニュアルを整備し、各自治体の防災基盤化の実現 - 被災前後での差分分析サービス

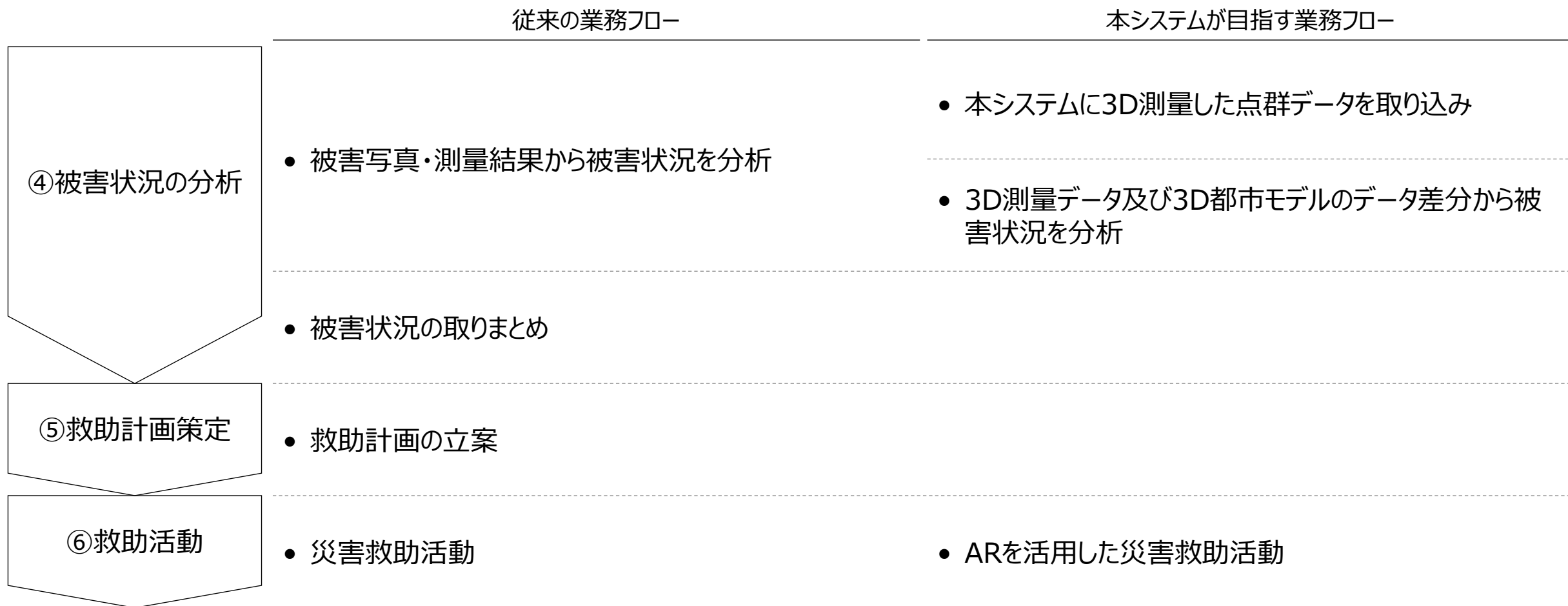
Ⅲ. 実証システム > 2. 想定事業機会 業務フロー（1/2）

本システムによって被災から被害状況の分析までのフローを大幅に短縮するとともにARを活用した救助活動の支援も行う

	従来の業務フロー	本システムが目指す業務フロー
①事前準備	(なし)	住民基本台帳のデータをあらかじめ本システムに取り込み
②対策本部設置	降雨量等をもとに、災害発生に備えて災害対策本部の設立準備	
③被害状況の情報収集	- 消防等への入電により被害の発生を把握 情報収集から分析まで、数日～数週間必要 - 自治体職員が現地に赴き写真等で被害状況についての情報収集 - 大規模な土砂災害やインフラの破損等がある場合、測量業務を実施	情報収集から分析まで、24時間以内に可能 - 自治体職員が現地に赴き被害範囲を把握（ドローン撮影範囲把握のため） - ドローンを用いて被害状況を撮影

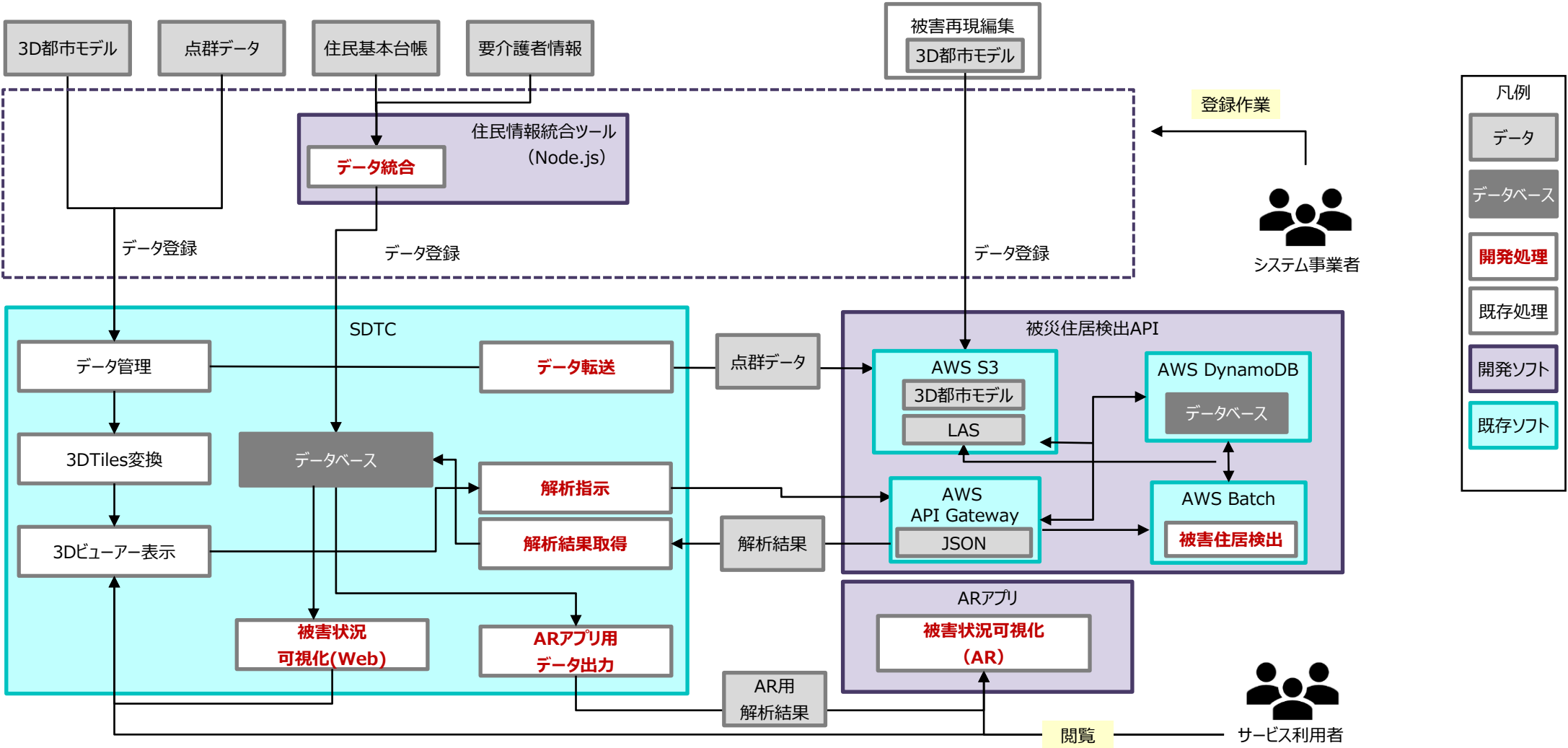
Ⅲ. 実証システム > 2. 想定事業機会 業務フロー（2/2）

本システムによって被災から被害状況の分析までのフローを大幅に短縮するとともにARを活用した救助活動の支援も行う



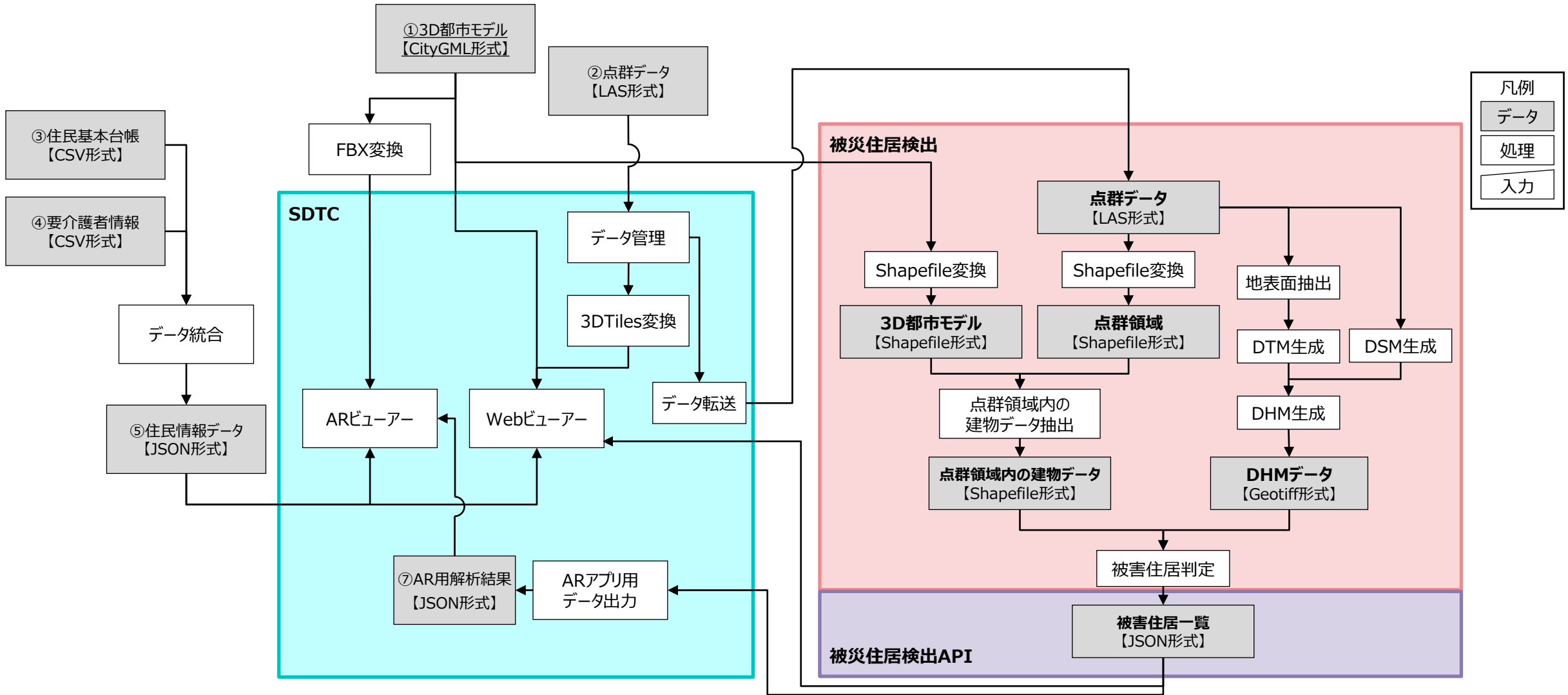
Ⅲ. 実証システム > 3. アーキテクチャ全体図

システムアーキテクチャ全体図



Ⅲ. 実証システム > 3. アーキテクチャ全体図

データアーキテクチャ全体図



Ⅲ. 実証システム > 4. システム機能

システム機能一覧

本実証実験で使用したシステムは下記のとおり

赤太字：開発機能

#	機能名	説明
1	データ管理	SDTC (Symmetry Digital Twin Cloud) の標準機能。SDTC内で使用するデータのアップロード、参照、情報更新、削除をする
2	3D Tiles変換	SDTC (Symmetry Digital Twin Cloud) の標準機能。アップロードしたデータを3Dビューアー内で表示可能な形式にするため、3D Tilesに変換する
3	3Dビューアー表示	SDTC (Symmetry Digital Twin Cloud) の標準機能。3D地図、オープンデータ、アップロード済みデータを表示する
4	A. データ統合	CSV形式の住民情報を、世帯情報－住民情報リストに構造化し、世帯番号をキーとしてプログラムからアクセスしやすいJSON形式に変換
5	B. データ転送	SDTCから点群データをアップロードされた後、解析指示の前に被災住居検出システムのストレージにコピーを行うとともにSDTCで表示可能な3D Tiles形式への変換も行う機能

Ⅲ. 実証システム > 4. システム機能

システム機能一覧

本実証実験で使用したシステムは下記のとおり

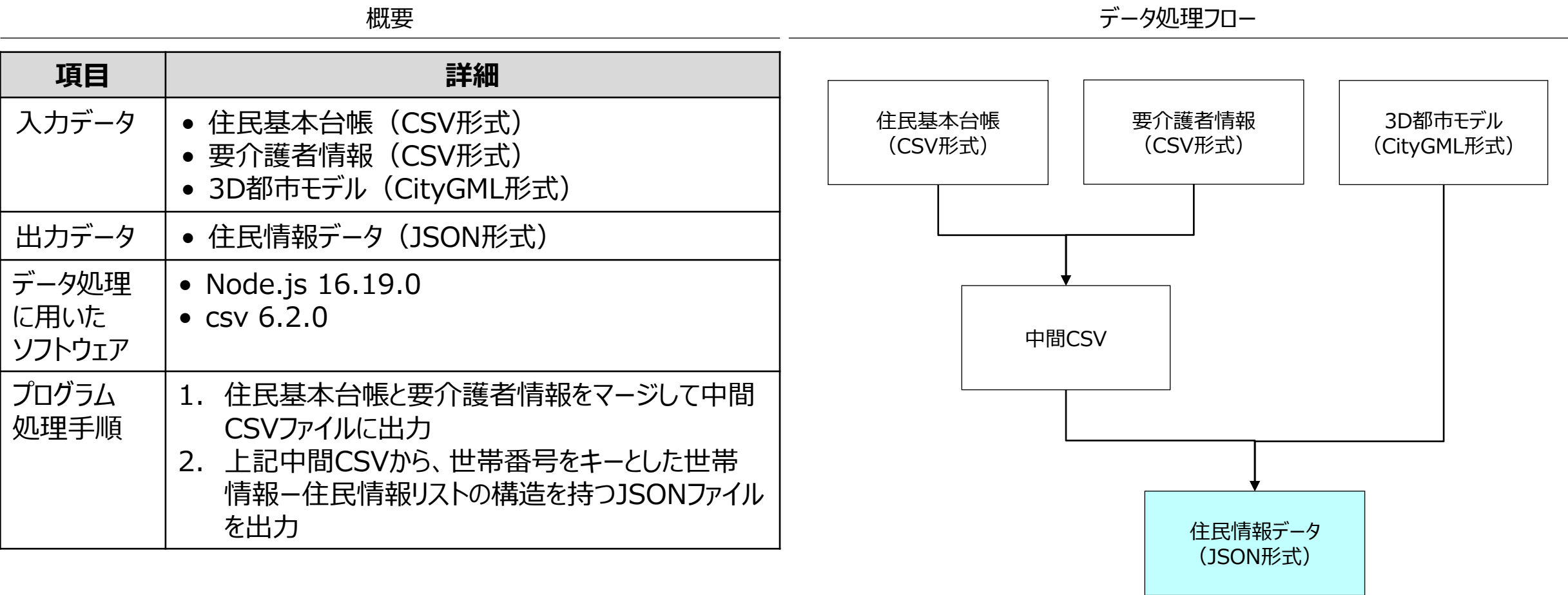
赤太字：開発機能

#	機能名	説明
6	C. 解析指示	SDTCのWeb画面のUIから、被災住居検出システムの解析APIを呼び出して解析を開始する
7	D. ARアプリ用データ出力	解析結果をARアプリで読み込めるようにJSON形式のデータファイルに出力する
8	E. 被害状況可視化 (Web)	SDTCのWeb画面で、解析結果と建物IDに基づいて3D都市モデルに被害情報を反映し、住民基本台帳データと合わせて表示する
9	F. 被害状況可視化 (AR)	ARアプリの画面で、解析結果と建物IDに基づいて3D都市モデルに被害情報を反映し、住民基本台帳データと合わせて表示する
10	G. 被害住居検出	APIの解析指示に応じて、クラウドストレージ上の点群データの被害住居検出処理を実行する
11	H. 被害住居検出API	登録された点群データを基に被災住宅検出処理を開始してジョブIDを発行し、解析処理の進捗状況を取得し、解析終了後に解析結果をレスポンスとして返信する機能

Ⅲ. 実証システム > 4. システム機能

A. データ統合

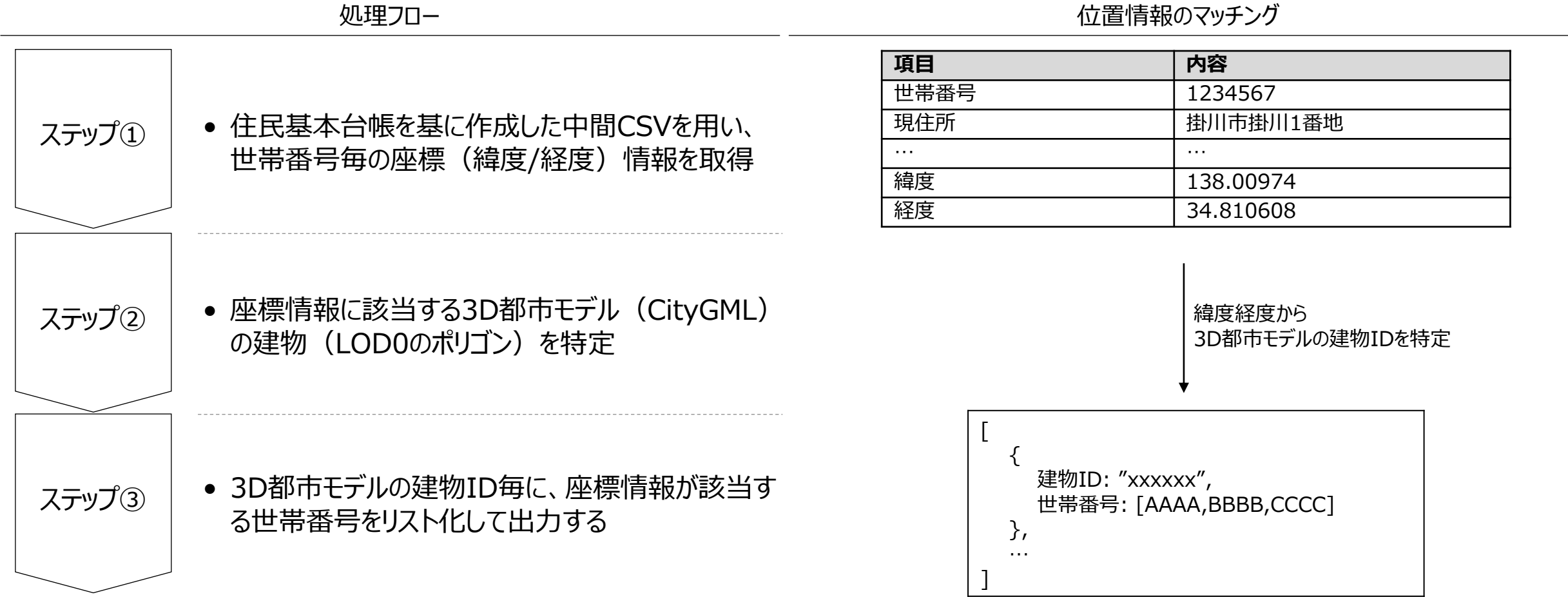
CSV形式の住民情報を、世帯情報－住民情報リストに構造化し、世帯番号をキーとしてプログラムからアクセスしやすいJSON形式に変換



Ⅲ. 実証システム > 4. システム機能

A. データ統合 | 3D都市モデルとの紐づけ

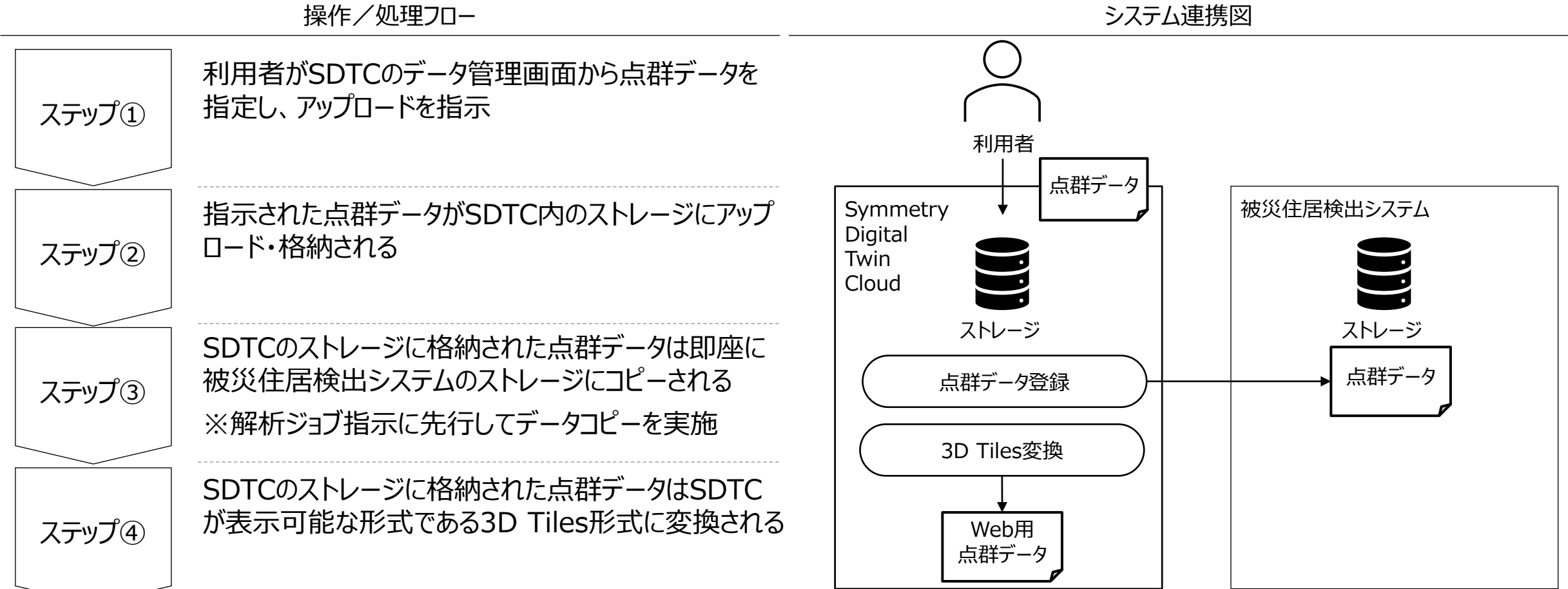
CSV形式の住民情報と3D都市モデルの建物IDとを紐づける



Ⅲ. 実証システム > 4. システム機能

B. データ転送

SDTCから点群データをアップロードされた後、解析指示の前に被災住居検出システムのストレージにコピーを行うとともにSDTCで表示可能な3D Tiles形式への変換も行う機能



Ⅲ. 実証システム > 4. システム機能

C. 解析指示

SDTCのWeb画面のUIから、被災住居検出システムの解析APIを呼び出して解析を開始する機能

操作／処理フロー

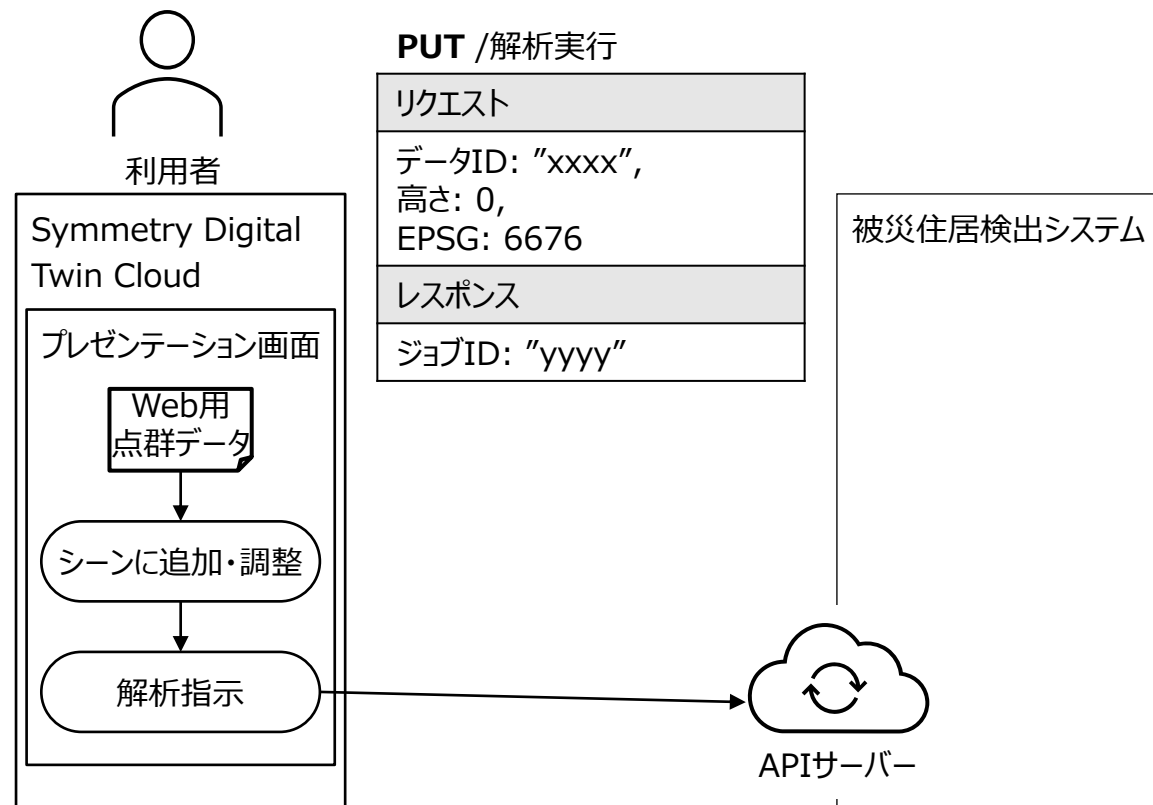
ステップ①

利用者はSDTCのプレゼンテーション画面を開き、3D Tilesに変換された点群データをシーン（SDTC上でプロジェクトを管理する単位）に追加する

ステップ②

解析指示UIのボタンをクリックし、被災住居検出システムの解析実行APIを呼び出して解析をスタートする

システム連携図



Ⅲ. 実証システム > 4. システム機能

D. ARアプリ用データ出力

解析結果をARアプリで読み込めるようにJSON形式のデータファイルに出力する機能

操作／処理フロー

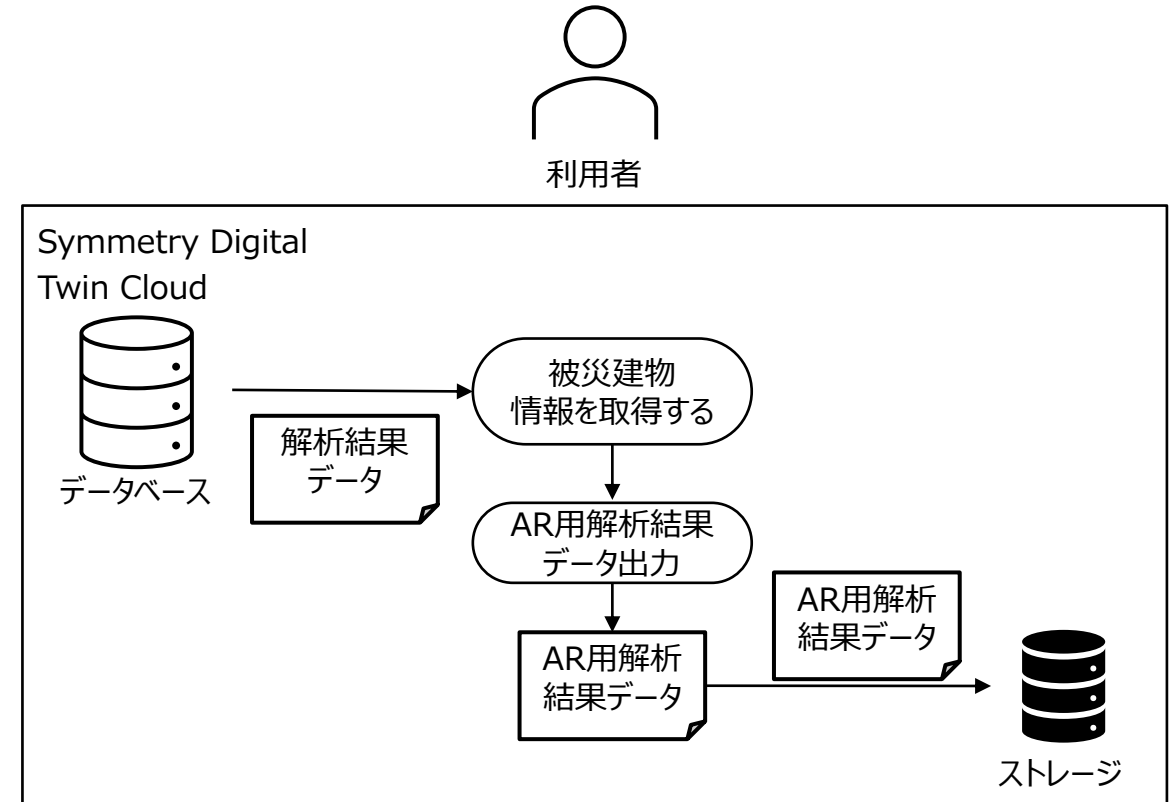
ステップ①

被災判定された住居の建物は、被災住居検出システムの結果取得APIから解析完了のレスポンスを受け取った際、SDTCのデータベース内に保存されている

ステップ②

ARアプリで表示ができるように、解析結果として保存されている解析結果をARアプリ用のJSON形式ファイルとしてストレージに出力する

システム連携図

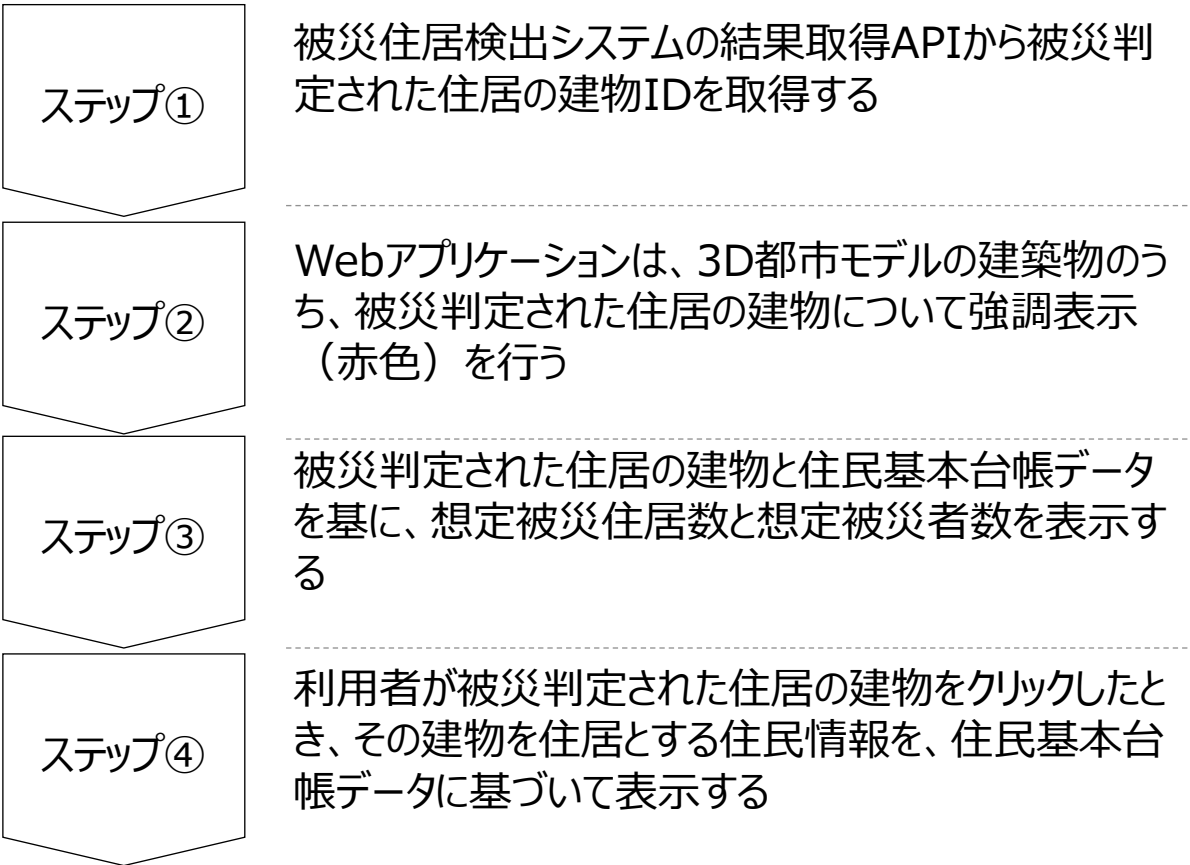


Ⅲ. 実証システム > 4. システム機能

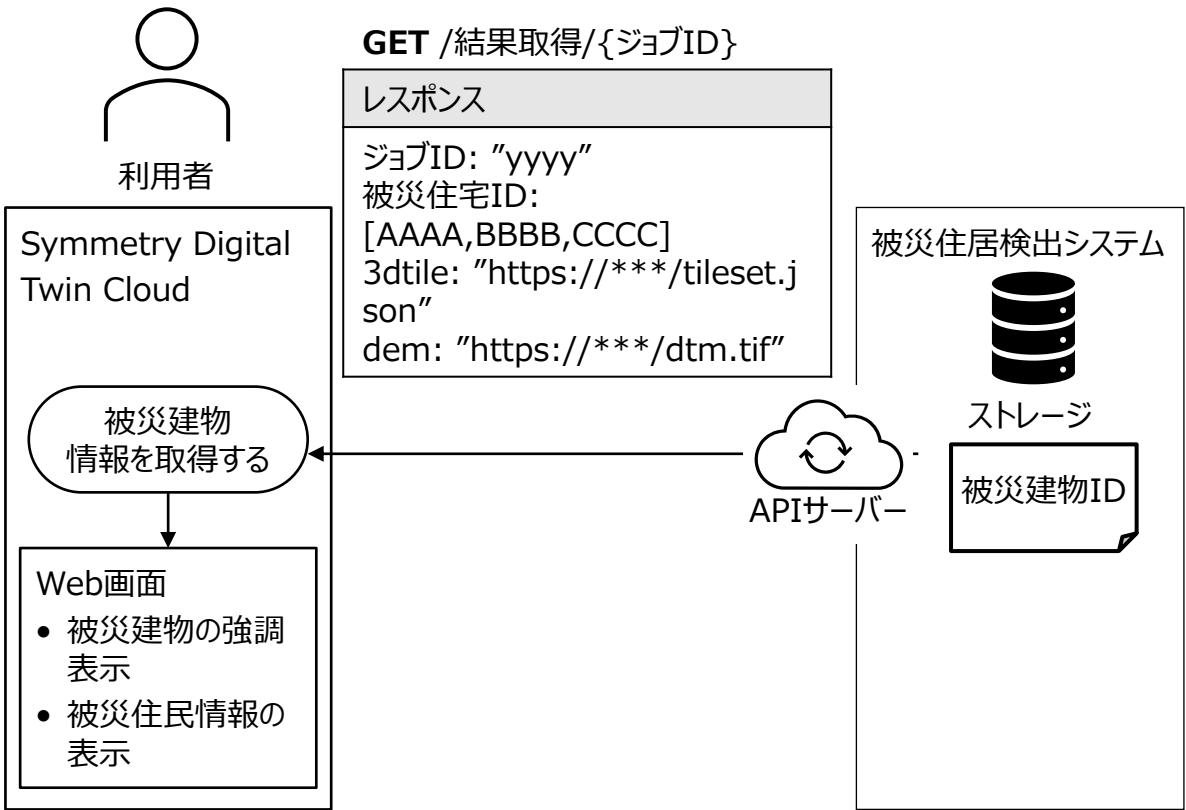
E. 被害状況可視化（Web）

被災状況検出システムで解析した結果をWeb画面で表示した3D都市モデル上に重ね合わせ、住民基本台帳データと合わせて表示する機能

操作／処理フロー



システム連携図



Ⅲ. 実証システム > 4. システム機能

F. 被害状況可視化（AR）

ARアプリの画面で、解析結果を3D都市モデルに反映し、住民基本台帳データと合わせて表示する

操作／処理フロー

ステップ①

ARアプリケーションは、『ARアプリ用データ出力機能』によりSDTCから得られるAR用解析結果データを取得する

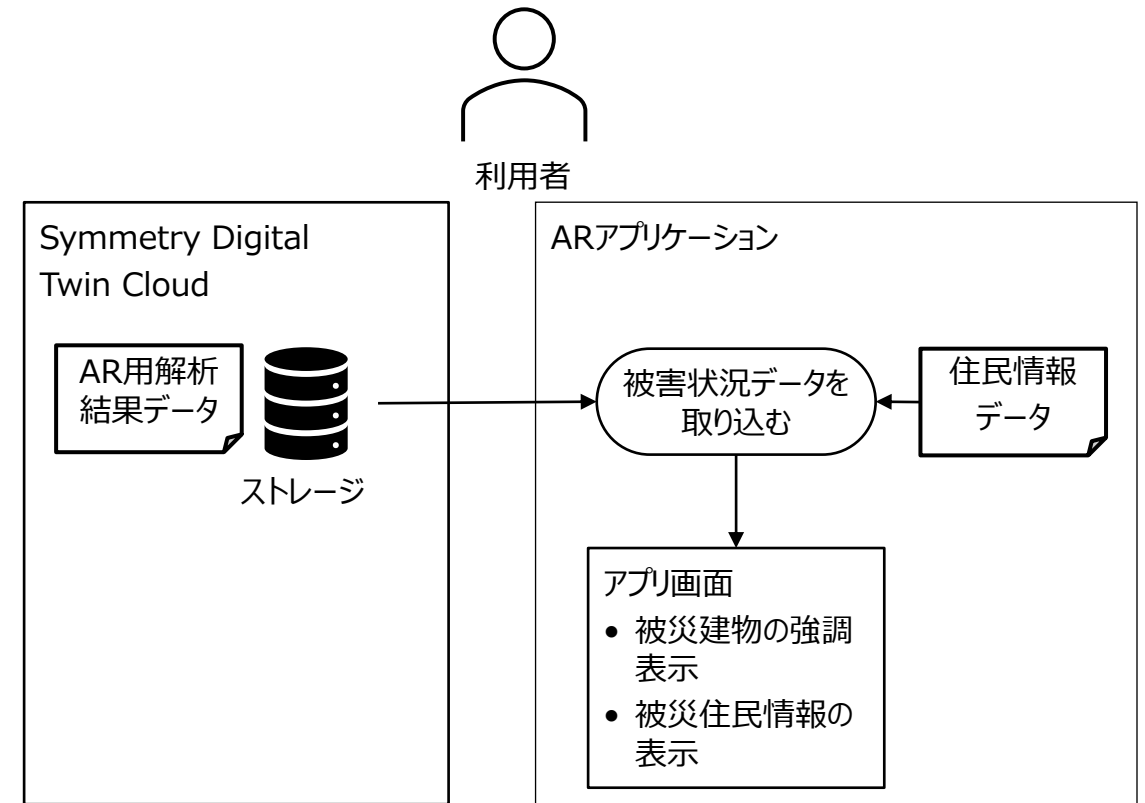
ステップ②

ARアプリケーションは、3D都市モデルの建築物のうち、被災判定された住居の建物について強調表示（赤色）を行う

ステップ③

利用者が被災した建物をタップしたとき、その建物を住居とする住民情報を、予めアプリ内に取り込んだ住民情報データに基づいて表示する

システム連携図



Ⅲ. 実証システム > 4. システム機能

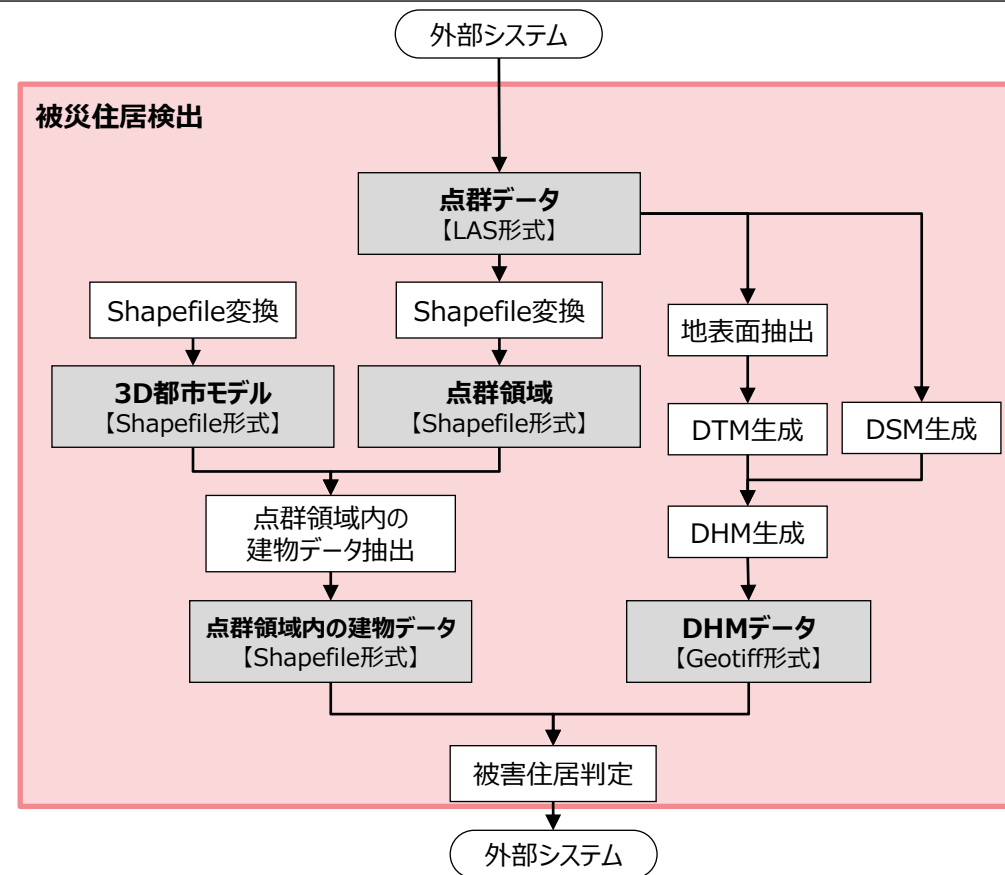
G. 被災住宅検出

被災住居検出アルゴリズムを使用して、点群データと3D都市モデルデータから被災住居の判定を行う。
 処理の手続きにおける各ステップは**5. アルゴリズム**で解説する。

処理の手続き

- ステップ1 : 点群データ領域に含まれる3D都市モデルの抽出
- ステップ2 : データから建物高さを特定
- ステップ3 : 住居の判定
- ステップ4 : 3D都市モデルのShapefile化
- ステップ5 : データ領域のShapefile化
- ステップ6 : データ領域に含まれる建物外形情報と建物IDの抽出
- ステップ7 : 建物表面高さマップ（DSM）の算出
- ステップ8 : 地表面（地形）の高さマップ（DTM）の
- ステップ9 : 建物高さマップ（DHM）の算出
- ステップ10 : 点群データに基づいた建物ごとの高さを計算
- ステップ11 : 住居の判定
- ステップ12 : 判定結果の出力

処理のフロー図



Ⅲ. 実証システム > 4. システム機能

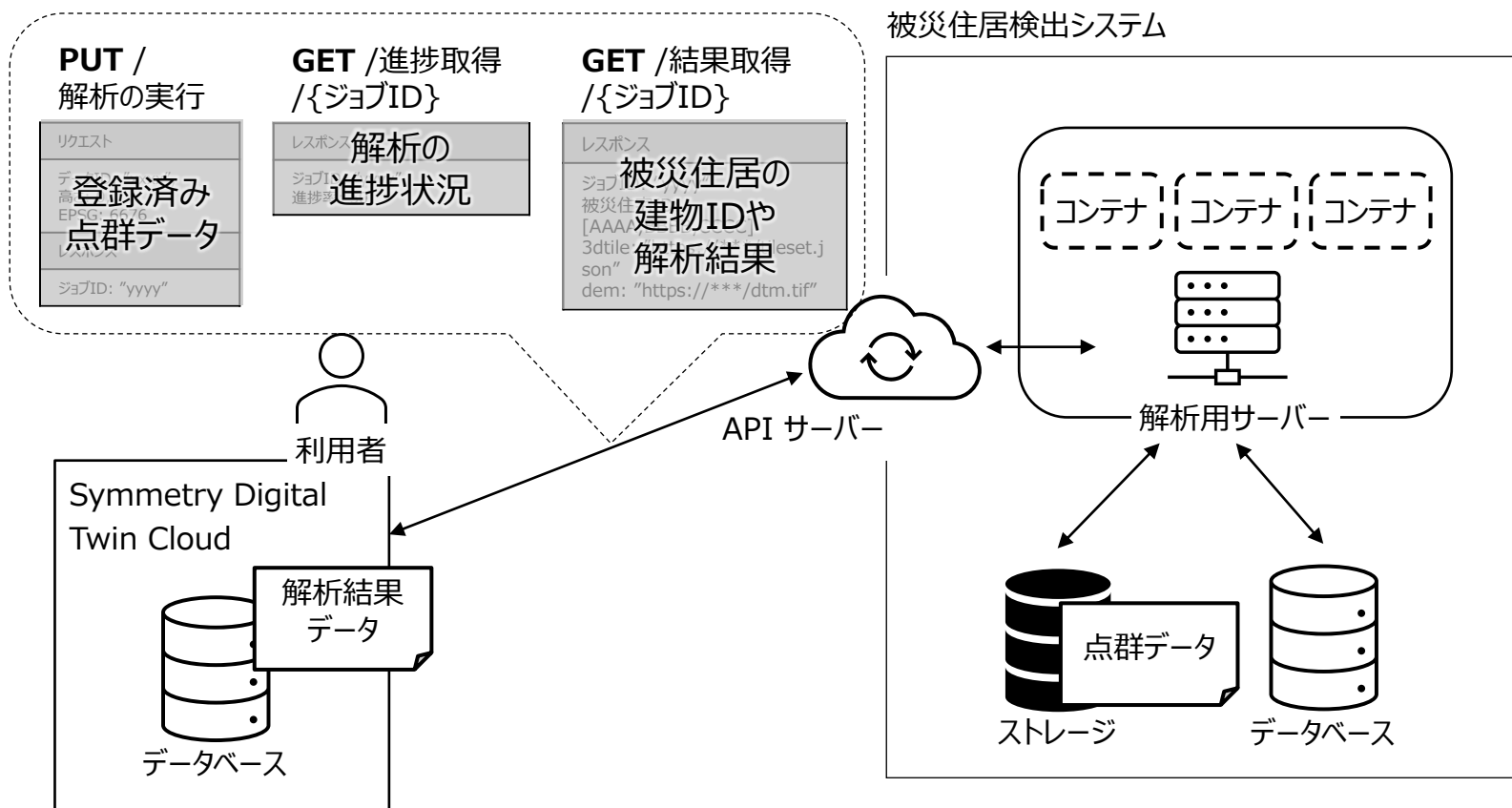
H. 被災住宅検出API

登録された点群データを基に被災住宅検出処理を開始してジョブIDを発行し、解析処理の進捗状況を取得する。
 解析終了後に解析結果をレスポンスとして返信する機能

APIを介して実行・管理される処理一覧

解析 実行	利用者のSDTC管理画面から『解析指示』を受け、『点群データ登録』で登録済みの点群データの解析を開始する
結果 取得	利用者はSDTC管理画面から被災住居の建物IDや3D Tilesデータ等といった解析結果を取得する
結果 登録	被災住居の建物IDなどの解析結果は、SDTCのデータベースに登録データの詳細情報として保存される

システム概要図



Ⅲ. 実証システム > 4. システム機能

H. 被災住宅検出API | 解析実行

LASデータの解析を実行

API仕様

プロトコル

HTTPS

メソッド

PUT

リクエストパラメータ

パラメータ名	名称	説明	値	必須
id	UUID	LASデータのUUIDを指定	UUIDの形式で指定	○
height	高さ	LASデータの高さを指定	実数で指定	○

レスポンス

フィールド	名称	説明
jobid	ジョブID	解析処理を識別するコード 進捗状況や結果の取得のためのキーとして利用

Ⅲ. 実証システム > 4. システム機能

H. 被災住宅検出API | 進捗取得



API仕様

プロトコル				
HTTPS				
メソッド				
GET				
リクエストパラメータ				
パラメータ名	名称	説明	値	必須
jobid	ジョブID	実行中の解析処理と紐づいているジョブIDを指定	解析実行のレスポンスに含まれるジョブIDを指定	○
レスポンス				
フィールド	名称	説明		
jobid	ジョブID	解析処理のジョブID		
status	ステータス	解析処理のステータスコードと説明は次の通り "SUBMITTED": 解析処理に必要な情報がサーバーへ正常に送信された。 "PENDING": 解析処理の開始準備中。 "RUNNABLE": 解析処理の開始中。 "STARTING": 解析処理の開始。 "RUNNNING": 解析処理実行中。 "SUCEEDED": 解析処理が正常に完了した。 "FAILED": 解析処理が異常終了した。		
progress	進捗率	0-100の値で解析の進捗状況を表す。異常終了した場合は-1の値を返す。		

Ⅲ. 実証システム > 4. システム機能

H. 被災住宅検出API | 結果取得



API仕様

プロトコル

HTTPS

メソッド

GET

リクエストパラメータ

パラメータ名	名称	説明	値	必須
jobid	ジョブID	実行中の解析処理と紐づいているジョブIDを指定	解析実行のレスポンスに含まれるジョブIDを指定	○

レスポンス

フィールド	名称	説明
jobid	ジョブID	解析処理のジョブID
uri	3D-TilesURI	3D Tilesデータに含まれるtileset.jsonのURL
bldgids	建物ID	被害住居として判定された建物IDのリスト

Ⅲ. 実証システム > 5. アルゴリズム 利用するアルゴリズムの一覧



名称	説明
被災住宅検出アルゴリズム	3D都市モデルと現況の点群データから住居の被災判定を行う

Ⅲ. 実証システム > 5. アルゴリズム

被災住宅検出アルゴリズムの概要

被災した建物と正常な建物を判別するために、点群データ中に含まれる建物の手掛かりとして特にUAV LiDARによって計測可能な「屋根」に着目し、地表面からの屋根の高さを基に被災を識別する

アルゴリズムの全体像

1. 点群データ領域に含まれる3D都市モデルの抽出

- 点群データとの比較分析に先立ち、点群データ領域（点群データの取得範囲）に合わせて3D都市モデルの必要範囲を抽出する

2. 点群データから建物高さを特定

- 点群データを分析・加工し、地形による標高差を排除した建物高さマップ（DHM）を作成する

3. 被災住居の判定

- 建物高さマップと3D都市モデルの建物外形情報から、「屋根」の面積に着目した判定ロジックにより被災住居判定を行う

被災判定の考え方

- 点群データにおける被災した建物は次のいずれかの特徴を持つ
 - 屋根が地表に対して水平・垂直方向に移動している
 - 屋根が消失している



屋根が水平移動したイメージ



屋根が消失したイメージ

Ⅲ. 実証システム > 5. アルゴリズム

1. 点群データ領域に含まれる3D都市モデルの抽出

点群データとの比較分析に先立ち、点群データ領域（点群データの取得範囲）に合わせて3D都市モデルの必要範囲を抽出する

3D都市モデルの Shapefile化

1. 対象エリアの3D都市モデルデータから建築物属性のFootprint（建物外形）と建物IDを抽出（Pythonライブラリ：ElementTree）
2. 建物外形の座標値を平面直角座標に変換（Pythonライブラリ：pyproj）
 - 本実証では静岡県掛川市を対象としているため、平面直角座標系 VIII系に変換した
3. 建物外形と建物IDをShapefile形式で出力（Pythonライブラリ：geopandas）

データ領域の Shapefile化

1. 点群データから点群データ領域のポリゴン（外形線）を生成する。点群データ領域のポリゴンとは点群が存在する領域の境界線を表す。lasboundaryというコマンドラインツールでは次のコマンドで生成することができる。>lasboundary -i（点群データのパス）-o（出力先のパス）（LAStoolsのlasboundaryツール）
2. 生成した点群データ領域の外形線をShapefileで出力する

データ領域に含まれる 建物外形情報と 建物IDの抽出

1. 3D都市モデルのShapfileと点群データ領域のShapefileを読み込む
2. 点群データの領域に含まれる3D都市モデルの建物外形を抽出する（Pythonライブラリ：geopandas）
3. 抽出した建物外形情報を建物IDと紐づけShapefileで出力する

Ⅲ. 実証システム > 5. アルゴリズム

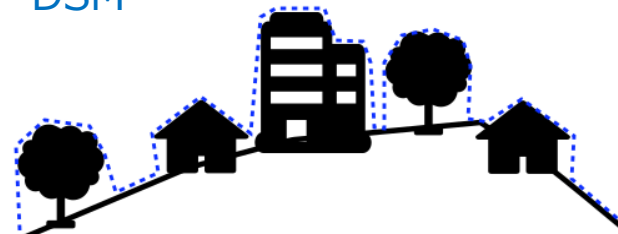
2. 点群データから建物高さを特定

点群データを分析・加工し、地形による標高差を排除した建物高さマップ（DHM）を作成する

建物表面高さマップ
（DSM）の算出

1. 点群データから建物表面の高さマップを作成
（LAStoolsのblast2demツール）
 - マップはGeoTIFF形式（0.5mメッシュ）で出力

DSM



地表面（地形）の
高さマップ（DTM）
の算出

1. 点群データの分類（地表面）を行う
（LAStoolsのlasgroundツール）
2. 地表面分類された点群データから地表面（地形）の高さマップを作成
（LAStoolsのblast2demツール）
 - マップはGeoTIFF形式（0.5mメッシュ）で出力

DTM



建物高さマップ
（DHM）の算出

1. 建物表面の高さマップと地表面（地形）の高さマップから高さの差分を計算し、
地形の高さの影響を取り除いた建物高さマップを算出
 - マップはGeoTIFF形式（0.5mメッシュ）で出力

DHM



Ⅲ. 実証システム > 5. アルゴリズム

3. 被災住居の判定

建物高さマップと3D都市モデルの建物外形情報から、「屋根」の面積に着目した判定ロジックにより被災住居判定を行う

点群データに基づいた
建物ごとの高さを計算

1. 3D都市モデルから抽出された点群データ領域の建物外形情報を使い、建物高さマップ（DHM）を建物単位に分割（Pythonライブラリ：fiona）

被災住居の判定

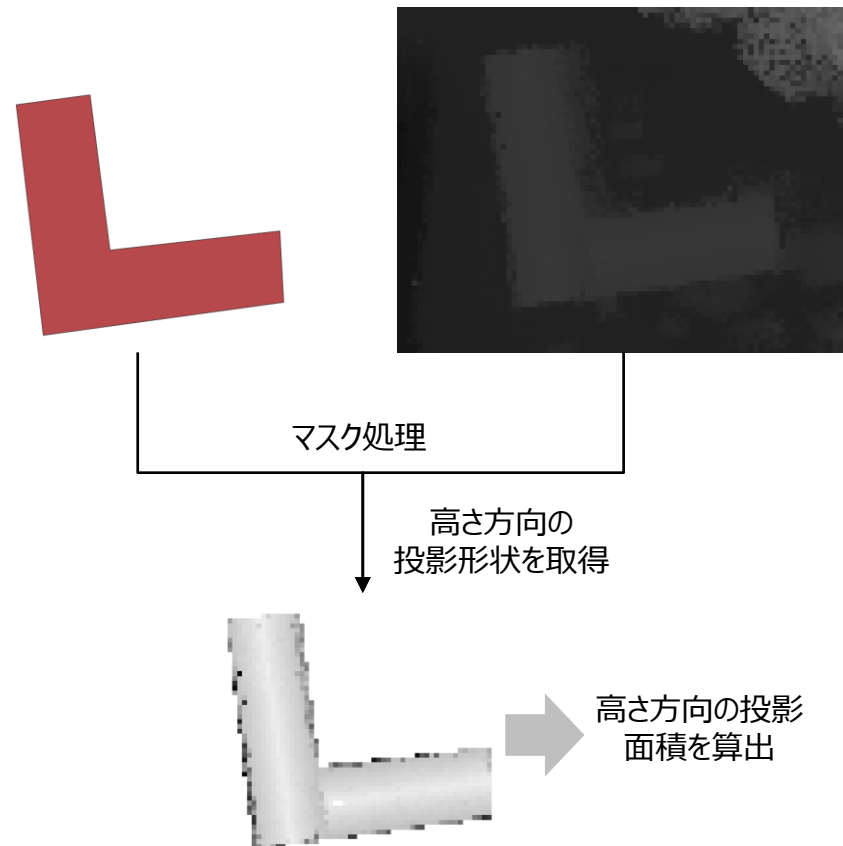
1. 建物ごとの建物高さマップから、高さ1.5m以上の部分を抽出（Pythonライブラリ：rasterio）
2. 抽出された部分から高さ方向の投影面積を算出（Pythonライブラリ：numpy）
3. 3D都市モデルのFootprint（建物外形）情報から算出した面積と投影面積の割合（比率）を算出
4. 割合を設定した閾値と比較して被災状態を判定する

判定結果の出力

1. 判定結果を建物IDに紐づけて出力
 - JSONファイルで出力

建物外形情報(Shapefile)

DHM(GeoTIFF)



Ⅲ. 実証システム > 5. アルゴリズム

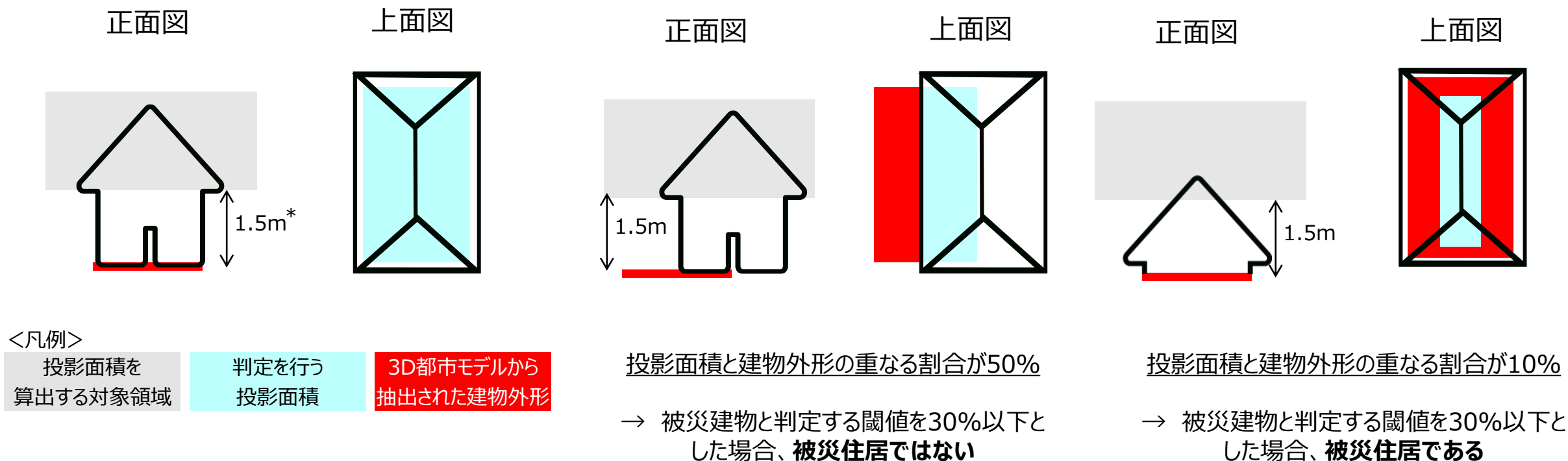
3. 被災住居の判定 | 被災住居の判定イメージ

点群データからDHMを生成し、3D都市モデルの建物外形線を利用してDHMを切り抜くことで建物領域内に含まれる屋根の面積値を推定することができる。本実証では地面から1.5m以上の高さ情報を持つDHMの値を屋根と仮定し、その面積値に対して閾値処理を行うことで被害住居判定を行った

正常な住宅の場合

被災により水平方向に移動した場合

被災により垂直方向に埋没した場合



* 本アルゴリズムは、建物底面からの高さ1.5mの位置を「屋根」と仮定して「屋根」の縦横高さずれから判定するアルゴリズムとしている

Ⅲ. 実証システム > 6. データ > ①活用データ

活用データ一覧：3D都市モデル

本実証実験で活用した3D都市モデルのデータは以下のとおり

地物	地物型	属性区分	属性名	内容
建築物LOD1	bldg:Building	空間属性	lod1Solid	• 建築物LOD1モデル
			lod0FootPrint	• 建築物の外形線で囲まれた接地面
		主題属性	stringAttribute	• 建物ID

Ⅲ. 実証システム > 6. データ > ①活用データ

活用データ一覧：その他

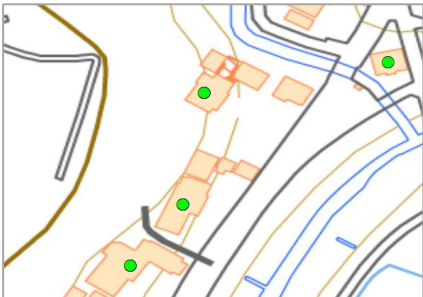
3D都市モデルデータ以外には下記のデータを活用する

活用データ	内容	データ形式	出典
住民基本台帳	掛川市土地情報システムに搭載している位置座標付きの住民基本台帳データを基に作成した実証実験用のサンプルデータ	CSV	掛川市
要介護情報	掛川市土地情報システムに搭載している位置座標付きの要介護度、認知症度データを基に作成した実証実験用のサンプルデータ	CSV	掛川市
点群データ	ドローン（DJI Zenmuse L1）によって計測した上西郷石ケ谷地区の点群データ	LAS	株式会社藤本組
AR用の建物3Dデータ	ARアプリでの利用のため、建物IDをメッシュのノード名とした3D都市モデル変換データ	FBX	PLATEAU

Ⅲ. 実証システム > 6. データ > ①活用データ 住民基本台帳

3D都市モデルと住民基本台帳を紐付け、被災した建物の住民情報を表示するために活用する

- 住民基本台帳の内容
- 住民基本台帳とは、住民基本台帳法に基づき、氏名、生年月日、性別、住所などが記載された個人を単位とする「住民票」を世帯ごとに編成したもの
 - 実証用のサンプルデータとして、掛川市の全庁型のGISツール（以下、庁内GIS）用に出力された住民基本台帳と座標情報（各世帯の居住位置の座標）からCSV形式でデータを作成
 - 住民基本台帳を3D都市モデルの建築物と紐付け表示を行うためには、「世帯が居住する位置情報」が必要
 - 掛川市の場合には庁内GISで使用するために座標情報も付与されている
 - 掛川市では空間情報の集約、共有及び業務支援のため、庁内GISを導入している。庁内GIS上に住民基本台帳を地図のレイヤとして追加することで、住民の居住位置や情報を可視化し、市の業務に活用している



住民基本台帳（●）と紐づけたマップのイメージ

住民基本台帳の記入例

項目	内容
現住所	掛川市掛川1番地
方書	
住民数	4
世帯番号	1234567
個人番号（宛名番号）	1234567
郵便番号	436-0342
行政区コード	285
小学校区	15
中学校区	56
住所変更日	1995/1/18
個人氏名	掛川 一郎
個人氏名（カナ）	カケガワ イチロウ
続柄	1（世帯主）
生年月日	昭和31年1月1日
性別	1（男）
年齢	66
緯度	138.00974
経度	34.810608

*出典：地理院地図

Ⅲ. 実証システム > 6. データ > ①活用データ

住民基本台帳 | データ仕様

元の住民基本台帳データ（CSV）をアプリケーション実行時に扱いやすいJSON形式に変換して利用している

データ仕様

- 元のCSV形式の住民基本台帳データを、アプリケーションで利用しやすいようにJSON形式に変換したもの
 - まず世帯ごとに住民を集約し、世帯番号をキーとして呼び出し、同一世帯の住民は、続柄から個人番号に紐づけて表示する
 - 住民の続柄および性別のフィールドは住民基本台帳では数字で記載されているが、コードを日本語文字列に変換している

データの例

```
{  
  "1234567": {  
    "世帯番号": "1234567",  
    "郵便番号": "436-0342",  
    "現住所": "掛川市掛川1番地",  
    "方書": "",  
    "表示世帯主": "掛川 一郎",  
    "行政区CD": "285",  
    "小学校区": "15",  
    "中学校区": "56",  
    "住民数": 4,  
    "住民": [  
      {  
        "個人番号": "1234567",  
        "個人氏名(N)": "掛川 一郎",  
        "個人氏名(K)": "カガワ イロウ",  
        "続柄": "世帯主",  
        "性別": "男",  
        "生年月日": "昭和31年1月1日",  
        "年齢": 66,  
        "住所変更日": "1995/1/18",  
        "要介護度": "",  
        "認知症度": ""  
      },  
      ...  
    ]  
  },  
  ...  
}
```

世帯番号がキーとなって呼び出している

Ⅲ. 実証システム > 6. データ > ①活用データ 要介護情報

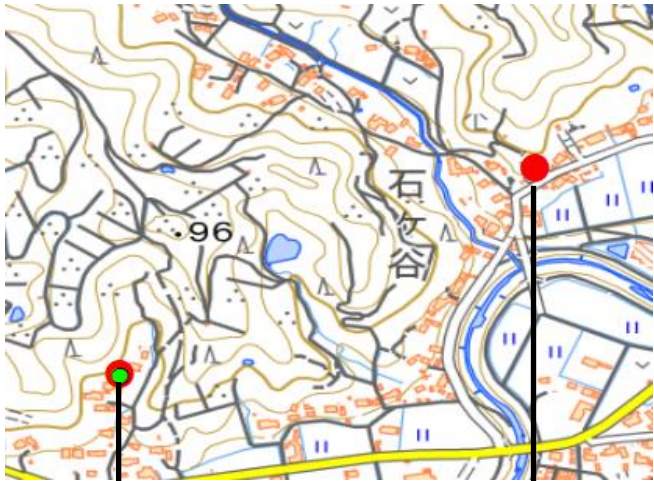
3D都市モデルと要介護情報を紐付け、被災した建物の住民の要介護情報を表示するために活用する

要介護情報の内容

- 要介護度および認知症度の情報が救助の優先度判定に有効かどうか検証するために利用
 - 要介護度
 - どれくらい介護のサービスを行う必要があるかの度合い
 - 内閣府が発行した災害時要援護者の避難支援ガイドラインによると要介護度3以上が緊急時の避難の際に支援が必要な方とされている
 - 認知症度
 - 認知症高齢者の日常生活自立度を指し、意思疎通、症状、行動の度合いなどを7段階でランク分けしたもの
- 実証用ダミーデータの作成方法
 - 掛川市の要介護度、認知症度の個人情報がマスクされたサンプルデータからダミーデータを生成
 - 住民基本台帳と要介護度、認知症度は別のデータとして存在するため、手動でデータの紐付けを実施

要介護度情報のサンプル

- 要介護度
- 認知症度



認知症度の情報

活用データ	内容
番号	101
認知症度	Ⅱ b
住所	掛川市掛川2番地
緯度	138.00564
経度	34.809524

要介護度の情報

活用データ	内容
番号	103
要介護度	要介護3
住所	掛川市掛川1番地
緯度	138.005639
経度	34.80954

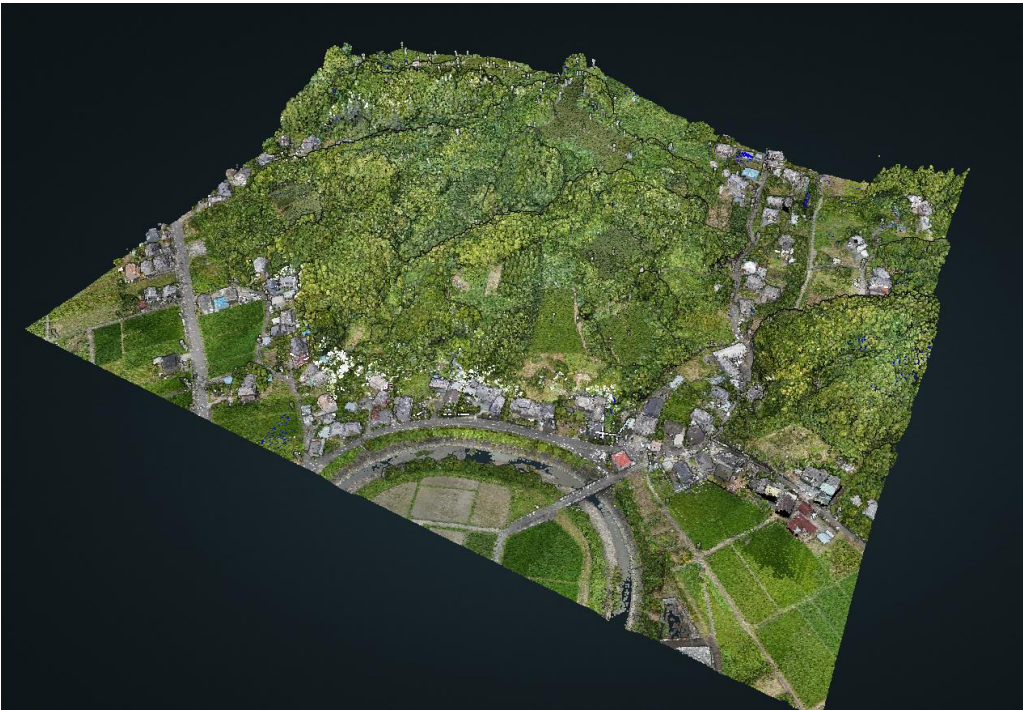
Ⅲ. 実証システム > 6. データ > ①活用データ

点群データ

DJI Zenmuse L1によって2022年7月22日に計測された上西郷石ヶ谷地区の点群データを使用した

上西郷石ヶ谷地区点群データ

データ詳細



データ形式	LAS 1.2
計測日	2022年7月22日
容量	10.9GB
面積	約325,076㎡
総点数	451,666,347点
点密度	1㎡あたり約1,389点
投影座標系	日本測地系2011 8系

Ⅲ. 実証システム > 6. データ > ②データ処理 データ処理一覧（1/2）

No.	システムに入力する データ (データ形式)	用途	処理内容	データ処理 ソフトウェア	活用データ (データ形式)
1	3D都市モデル (FBX形式)	ARアプリ上での表示のため	- PLATEAUのFMEワークスペースサンプルを利用して構築 - 変換処理実行	FME Desktop (PLATEAU 3D都市モデルのデータ変換マニュアルを利用)	3D都市モデル (CityGML形式)
2	点群データ (LAS形式)	解析処理での被害住居検出のため	被害状況の再現	CloudCompare	点群データ (LAS形式)
3	点群データ (3DTiles形式)	Webビューアー上での表示のため	- SDTCにアップロード - 変換処理を実行（自動）	Symmetry Digital Twin Cloud	点群データ (LAS形式)
4	住宅の外形線 (Shapefile)	被害住宅検出	lod0FootPrint情報をCityGML形式からShapefile形式への変換	Python	3D都市モデル (CityGML)
5	点群データ領域のShapefile化	被害住宅検出	点群データ領域の外形線をShapefileで出力する	LAStool (lasboundary)	Shapefile

Ⅲ. 実証システム > 6. データ > ②データ処理 データ処理一覧（2/2）

No.	システムに入力するデータ (データ形式)	用途	処理内容	データ処理ソフトウェア	活用データ (データ形式)
6	LASデータの領域内の建物外形 (Shapefile)	被害住宅検出	LASデータのポリゴン領域に含まれる建物外形の抽出する	LAStool (lasboundary)	LASデータから生成した領域ポリゴン (Shapefile) 点群データ(LAS形式)
7	DSM	被害住宅検出	点群データから数値表層モデルを作成する	LAStools、las2dem	点群データ(LAS形式)
8	点群データ地表面	被害住宅検出	点群データから地表面点群を抽出する	LAStools, lasground	点群データ(LAS形式)
9	DTM	被害住宅検出	点群データから数値地表モデルを作成する	LAStools、las2dem	点群データ(LAS形式)
10	DHM	被害住宅検出	DSMとDTMの差分から数値高さモデルを作成する	Python	数値標高データ (GeoTIFF形式)
11	住民情報データ (JSON形式)	1.Webビューアー上での表示のため 2.ARアプリ上での表示のため	- 住民基本台帳データと要介護者情報の紐づけ - CSV形式からJSON形式への変換	Node.js (「データ統合」機能)	住民基本台帳データ (CSV形式) 要介護者情報 (CSV形式)

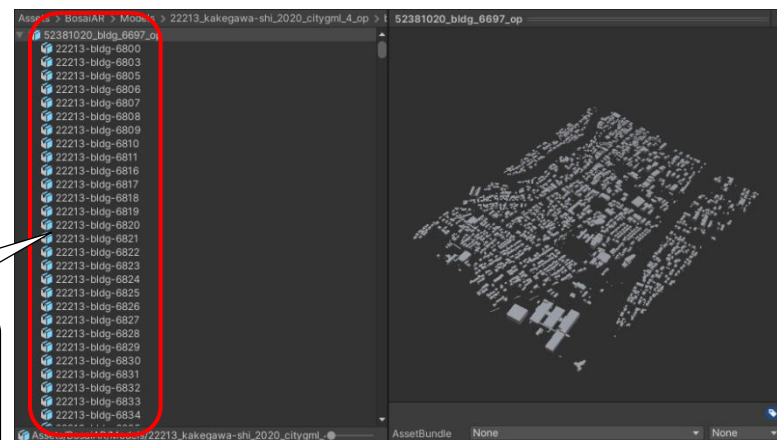
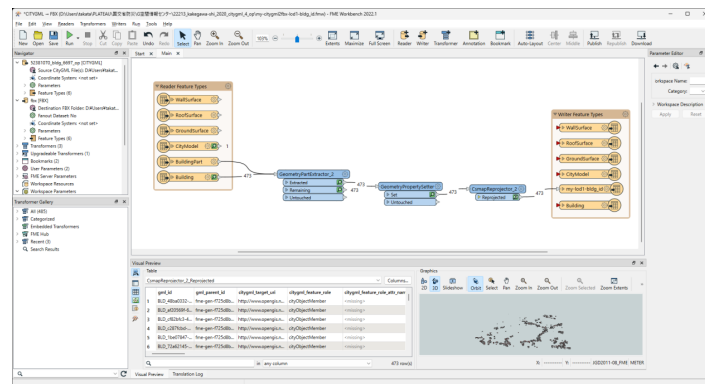
Ⅲ. 実証システム > 6. データ > ②データ処理 1. 3D都市モデルの変換

ARアプリ内で表示する3Dモデルの中に3D都市モデルの建物IDを埋め込むデータ変換を行う

概要

項目	詳細
入力データ	3D都市モデル（CityGML形式）
出力データ	3D都市モデル（FBX形式）
データ処理に 用いたソフト ウェア	FME Desktop 2022.1
データ処理 フロー	1. CityGMLファイルの読み込み 2. LOD1ジオメトリの抽出 3. ジオメトリ名を建物IDプロパティに変更 4. 座標参照系(平面直角座標8系)の変換 5. FBXファイルの書き込み

FME Desktopでの変換実行例とFBX出力例



建物IDが各建物のメッシュの
ノード名となっている

Ⅲ. 実証システム > 6. データ > ②データ処理

2. 点群データ（LAS形式）の編集

解析処理での被害住居検出のためのダミーデータを作成する



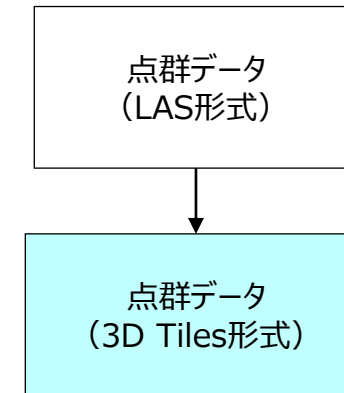
Ⅲ. 実証システム > 6. データ > ②データ処理 3. SDTCによる点群データの変換

SDTCのデータ管理画面から点群データをアップロードし、データを登録する

概要

項目	詳細
入力データ	点群データ（LAS形式）
出力データ	点群データ（3D Tiles形式）
データ処理に用いたソフトウェア	Symmetry Digital Twin Cloud（SDTC）
プログラム処理手順	1. SDTCに点群データをアップロード 2. 変換処理（自動実行）

3D Tilesへの変換



Ⅲ. 実証システム > 6. データ > ②データ処理

4. 3D都市モデルのShapefile化

概要

項目	詳細
入力データ	3D都市モデ（CityGML形式）
出力データ	2Dフットプリント（Shapefile形式）
データ処理に用いたソフトウェア	Python
プログラム処理手順	1. CityGMLのデータのパーズを行う（ElementTree） 2. 建物IDとlod0FootPrintの座標情報を取得する 3. lod0FootPrintの座標情報をPolygonに変換する(Shapely) 4. 建物IDとPolygonをshpファイルとして出力する(Geopandas)

処理フロー

入力

3D都市モデル
(CityGML形式)



処理

XMLのパーズ



建物IDと建物FootPrint情報の抽出



建物IDと建物FootPrint情報の抽出



出力

2Dフットプリント
(Shapefile形式)

Ⅲ. 実証システム > 6. データ > ②データ処理

5. LASデータからの領域ポリゴン(Shapefile)の生成

概要

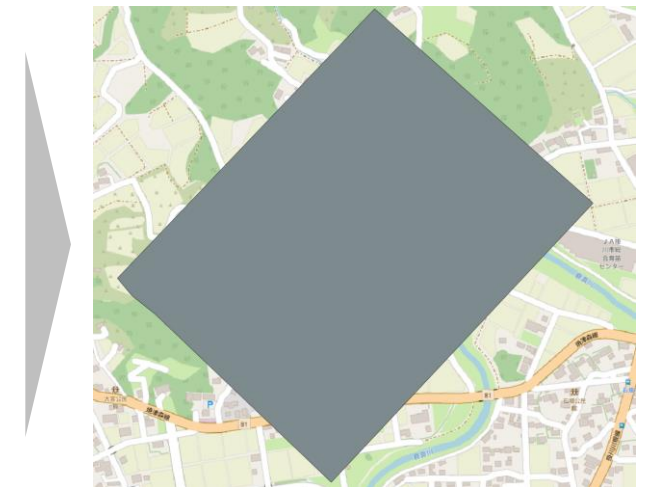
項目	詳細
入力データ	点群データ (LAS形式)
出力データ	Shapefile
データ処理に用いたソフトウェア	LAStools (Lasboundary)
プログラム処理手順	LAStoolsに含まれるlasboundaryを利用することで点群の領域の外形のShapefileを生成することができる。 LastoolsはWindowsで動作し、コマンドプロンプトを利用して実行することができる。 コマンドは以下の通り。 <pre>>lasboundary -i (LASデータのパス) -oshp</pre>

LAStools(Lasboundary)によるLASからShapefileへの変換

LASデータ



Shapefile



Ⅲ. 実証システム > 6. データ > ②データ処理

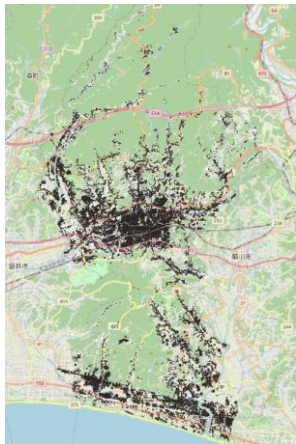
6. LASデータのポリゴン領域に含まれる建物外形の抽出

概要

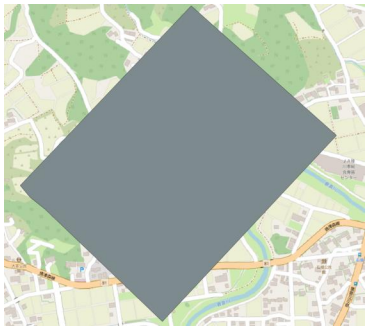
項目	詳細
入力データ	点群データ（LAS形式） LASデータから生成した領域ポリゴン（Shapefile）
出力データ	Shapefile
データ処理に用いたソフトウェア	LAStools（Lasboundary）
プログラム処理手順	- 元の3D都市データには掛川市全域の建物ポリゴンが含まれているが、被害判定の処理に必要なデータはLASデータの領域内の建物ポリゴンである - このためgeopandasやshapely等のPython用ライブラリを利用してポリゴンの包含判定を行い、LASデータの領域内の建物を抽出し、Shapefileとして出力する処理を開発した - また、本処理は被害住宅判定処理を行うサーバー上で動作し、解析処理のたびに自動的に実行される

LASデータ領域内に含まれる建物ポリゴンの抽出

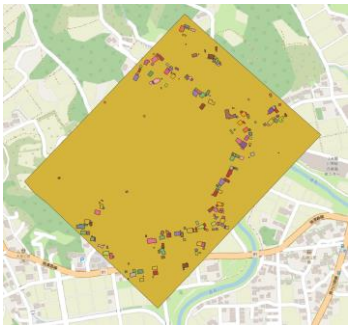
掛川市全域の建物外形



LASデータの領域ポリゴン



LASデータの領域内の建物外形



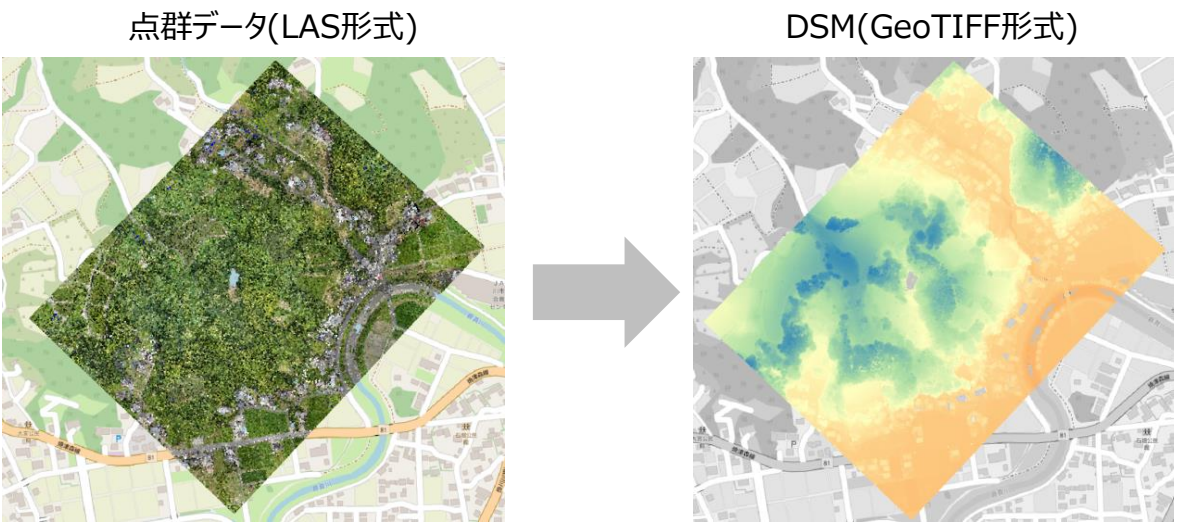
Ⅲ. 実証システム > 6. データ > ②データ処理

7. 建物表面高さマップ°（DSM）の算出

概要

LAStools(las2dem)によるLASからDSMへの変換

項目	詳細
入力データ	点群データ(LAS形式)
出力データ	DSM（GeoTIFF形式）
データ処理に用いたソフトウェア	las2dem
プログラム処理手順	LAStoolsに含まれるlas2demを利用することでDSMを生成することができる。LastoolsはWindowsのコマンドプロンプトから呼び出すことができる。 コマンドの例は以下の通り。 >las2dem -i（LASデータのパス） -otif



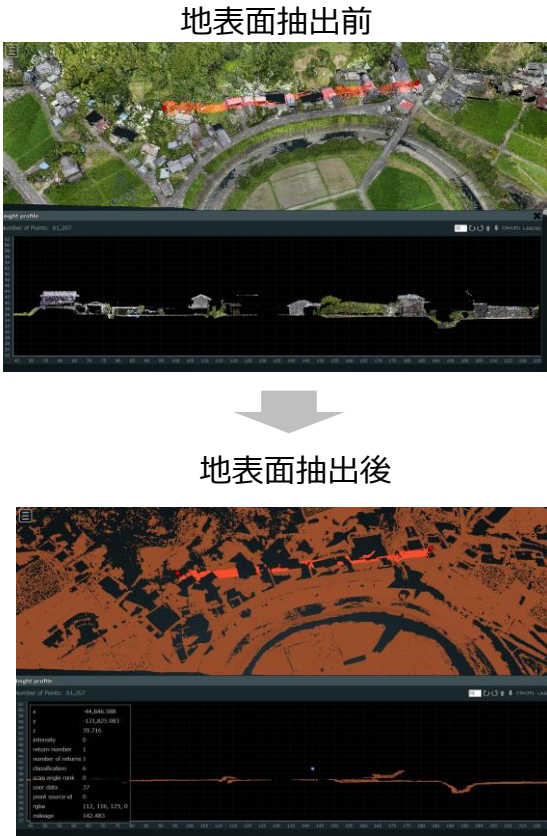
Ⅲ. 実証システム > 6. データ > ②データ処理

8. 点群の地表面抽出処理

概要

項目	詳細
入力データ	点群データ（LAS形式）
出力データ	GeoTIFF
データ処理に用いたソフトウェア	LAStools（Lasground）
プログラム処理手順	DTMはDHMを作成するために必要となる。 DTMの生成のため、点群データから地表面点群を抽出した。 LAStoolsに含まれるlasgroundというツールを利用することでLASデータから地表面点群を抽出することができる。 コマンドの例を以下に示す。 >lasgrouround -i（LASデータのパス） -olas

LAStools(lasground)による地表面点群抽出



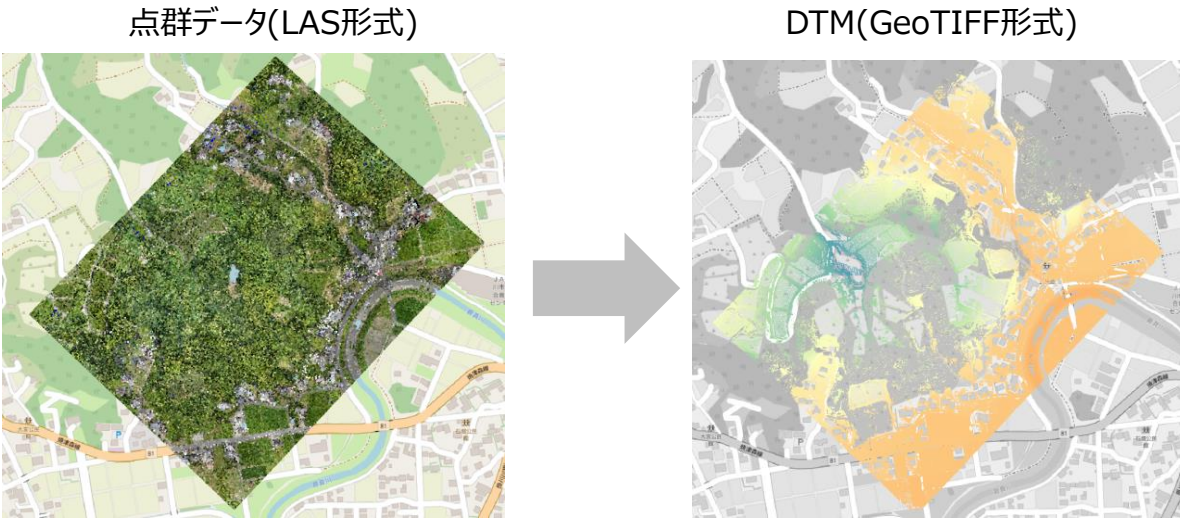
Ⅲ. 実証システム > 6. データ > ②データ処理

9. 建物表面高さマップ°（DTM）の算出

概要

LAStools(las2dem)によるLASからDTMへの変換

項目	詳細
入力データ	点群データ（LAS形式）
出力データ	DTM（GeoTIFF形式）
データ処理に用いたソフトウェア	las2dem
プログラム処理手順	地表面抽出後の点群データを使用して、DSMと同じ手順でDTMを生成することができる。 DTMの生成にはLAStoolsに含まれるlas2demを使用する。 コマンドの例は以下の通り。 >las2dem -i（LASデータのパス） -otif



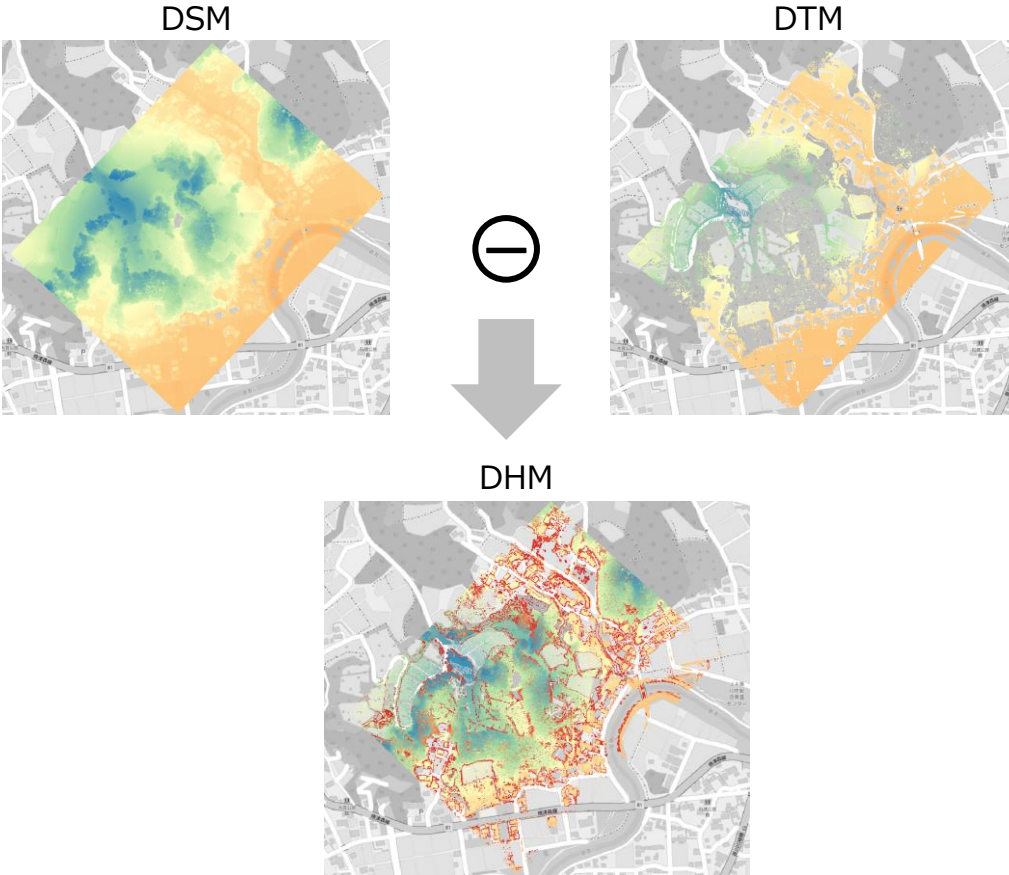
Ⅲ. 実証システム > 6. データ > ②データ処理

10. DHMの生成

概要

項目	詳細
入力データ	点群データ（LAS形式）
出力データ	GeoTIFF
データ処理に用いたソフトウェア	Rasterio、Python
プログラム処理手順	• DHMは点群から生成したDSMとDTMの差分から生成できる • このためrasterio等のPython用ライブラリを利用してDSMとDTMの差分を計算し、GeoTIFF形式で出力する処理の開発を行った • また、本処理は被害住宅判定処理を行うサーバー上で動作し、解析処理のたびに自動的に実行される

LAStools(Lasboundary)によるLASからShapefileへの変換



Ⅲ. 実証システム > 6. データ > ③出力データ 出力データ一覧

本システムから出力されるデータは下記のとおり

出力データ	内容	データ形式
被災住宅解析結果	点群データに含まれる住宅のうち、被災判定された住宅の建物ID等の解析結果データ	JSON
AR用解析結果	被災住宅解析結果をARアプリケーションで表示できる形式で出力したもの	JSON

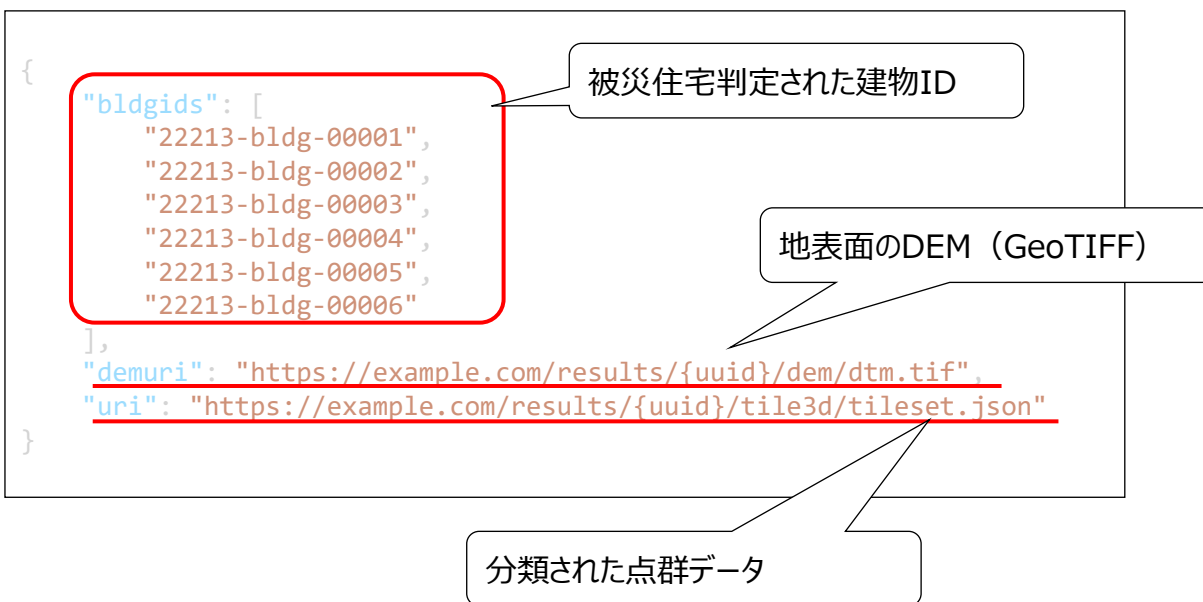
Ⅲ. 実証システム > 6. データ > ③出力データ 被災住宅解析結果

点群データに含まれる住宅のうち、被災判定された住宅の建物ID等の解析結果データ

データ仕様

- 解析処理の完了後、被災住居検出システムの結果取得APIから解析結果取得を取得することができる。レスポンスデータには被災住宅の建物ID一覧、分類された点群の3D TilesデータのURL、地表面のDEMデータ（数値標高モデル、GeoTIFF形式）のURLが含まれる

データの例



Ⅲ. 実証システム > 6. データ > ③出力データ AR用解析結果

被災住宅解析結果をARアプリケーションで表示可能な形式で出力したデータ

データ仕様

活用データ	内容
ジョブID	該当するジョブIDを表示
ジョブのステータス	SUCCEEDED, SUBMITTED, RUNNING, SUCCEEDED, FAILED
ジョブの進捗状況	0~100
SDTC内でのデータ情報	- データのID - データの名前 - 座標参照系
解析結果が得られたタイムスタンプ	Unixtime 1970年1月1日0時0分0秒からの経過秒数
解析結果の統計情報	- 被災した建物の数 - 被災した世帯の数 - 被災した住民の数

データの例

```

{
  "jobid": "91b02450-092a-4b4c-a8d3-ee173dea4bf8",
  "status": "SUCCEEDED",
  "progress": 100,
  "asset": {
    "id": "c04d0298-b0e1-4bac-8a19-8325a9c0656e",
    "name": "掛川市上西郷",
    "convert_options": {
      "--input-crs": "epsg:6676"
    }
  },
  "timestamp": 1672995544,
  "results": {
    "jobid": "91b02450-092a-4b4c-a8d3-ee173dea4bf8",
    "uri": "https://example.com/results/{uuid}/tile3d/tileset.json",
    "demuri": "https://example.com/results/{uuid}/dem/dtm.tif",
    "bldgids": [
      "22213-bldg-00001",
      "22213-bldg-00002",
      "22213-bldg-00003",
      "22213-bldg-00004",
      "22213-bldg-00005",
      "22213-bldg-00006"
    ]
  },
  "summary": {
    "higai_bldg_num": 6,
    "higai_setai_num": 3,
    "higai_jumin_num": 13
  }
}

```

SDTC内での
データ情報

解析結果APIの
レスポンスを格納

解析結果の統計情報

Ⅲ. 実証システム > 7. ユーザインタフェース

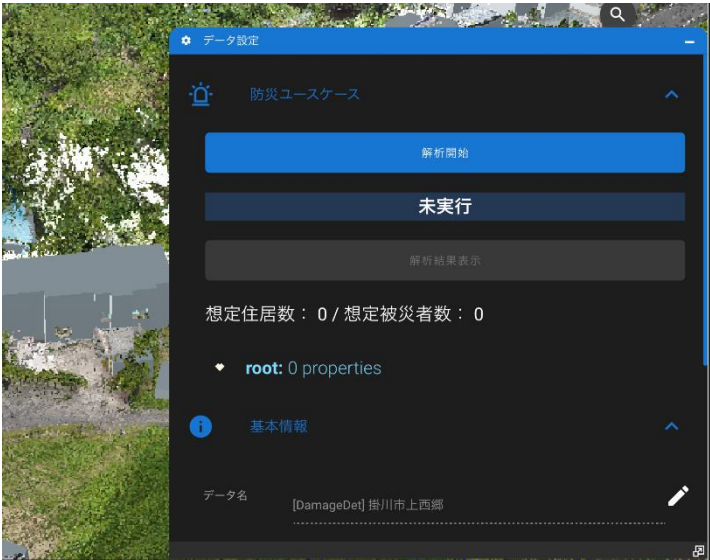
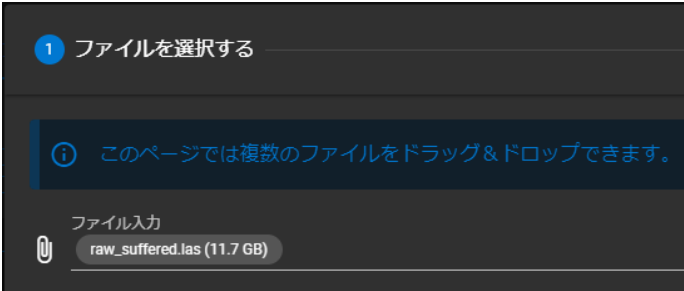
UI/UXの全体フロー（1/2）

①点群データの登録

②被災住宅解析の実行

③解析結果の確認

イメージ



概要

SDTCのデータ管理画面から点群データをアップロードし、データを登録する

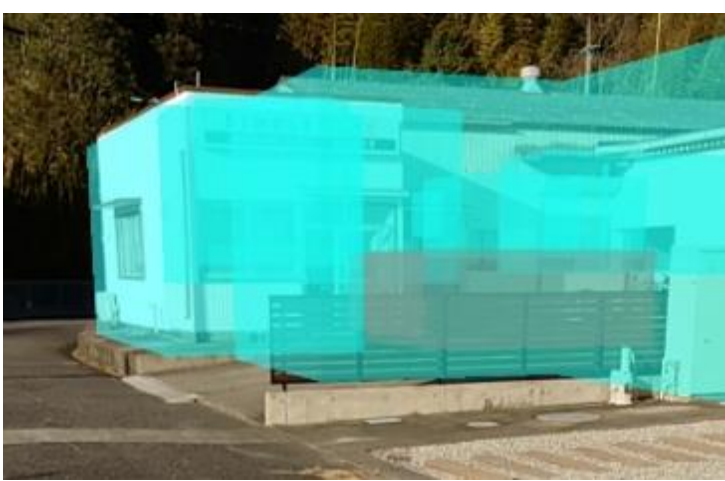
SDTCのプレゼンテーション画面に当該点群データを追加し、被災住宅解析を実行する

SDTCのプレゼンテーション画面内の3D都市モデルに被災情報を反映する。合わせて住民情報も表示する

Ⅲ. 実証システム > 7. ユーザインタフェース UI/UXの全体フロー（2/2）



イメージ



概要

ARアプリ内の地図上で被災情報を反映する

AR画面内の3D都市モデルに被災情報を反映する。合わせて住民情報も表示する

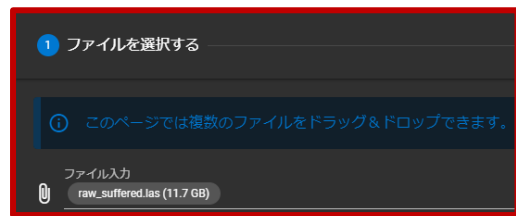
ARアプリ内の地図およびAR画面内の3D都市モデルにずれがある場合は向きや高さを調整し、ARが正しく表示されるように調整する

Ⅲ. 実証システム > 7. ユーザインタフェース

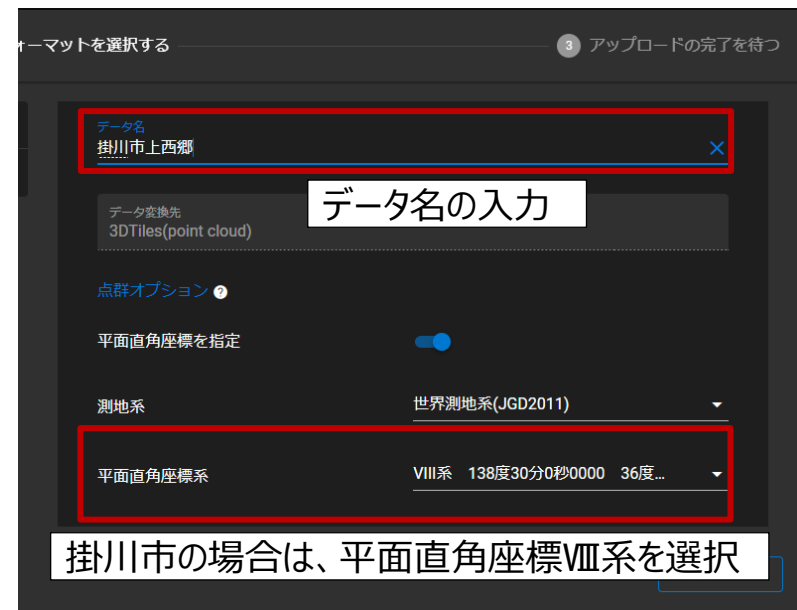
SDTC | 点群データの登録

SDTCのデータ管理画面から点群データをアップロードし、データを登録する

点群データのアップロード



- 点群データをアップロードダイアログにドラッグアンドドロップする



- データ名を入力し、適切な平面直角座標を選択する

3D Tilesへの変換

- アップロード後、自動的に3D Tilesへの変換が行なわれる



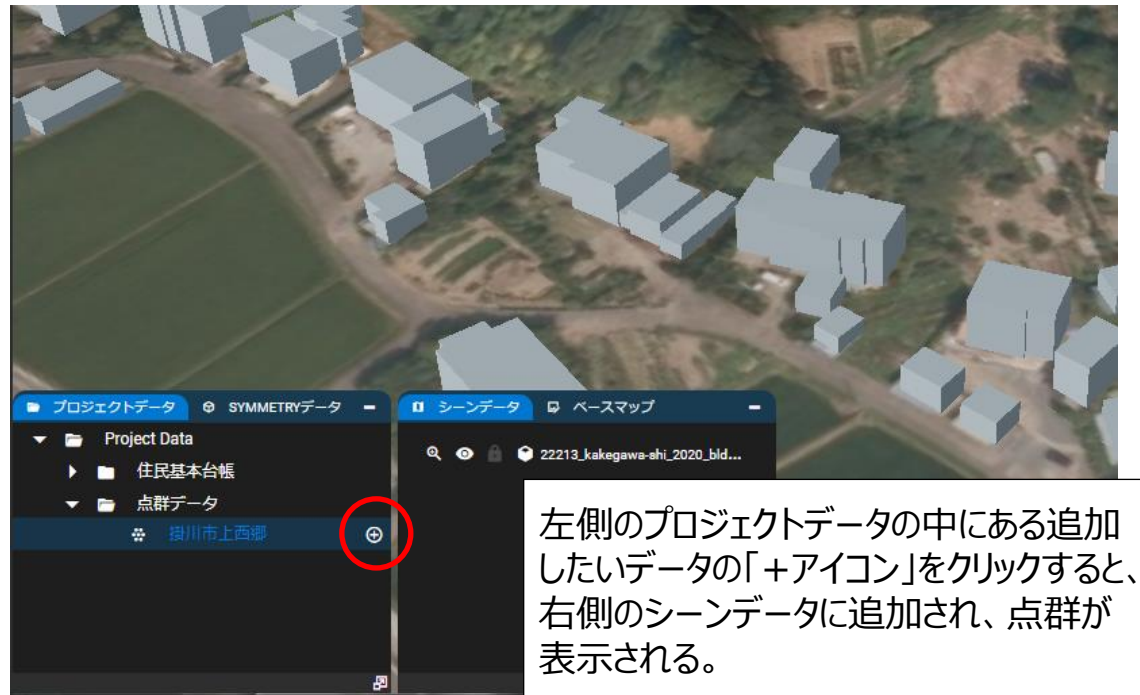
Ⅲ. 実証システム > 7. ユーザインタフェース

SDTC | 被災住宅解析の実行

SDTCのプレゼンテーション画面に当該点群データを追加し、被災住宅解析を実行する

点群データのシーンへの追加

- アップロードした点群データをシーンに追加する



解析の開始

- 点群データを選択し、「解析開始」ボタンをクリックする



Ⅲ. 実証システム > 7. ユーザインタフェース SDTC | 解析結果の表示

SDTCのプレゼンテーション画面内の3D都市モデルに被災情報を反映し、合わせて住民情報も表示する

被災建物の強調表示

- 「解析結果表示」をクリックすると、被災した建物を赤く強調表示



住民情報の表示

- 建物をクリックすると、世帯および住民の情報が表示される



Ⅲ. 実証システム > 7. ユーザインタフェース

ARアプリ | 地図上での被災住宅確認

ARアプリ内の地図およびAR画面内の3D都市モデルに被災情報を反映し、合わせて住民情報も表示する

解析結果データの選択

被災住宅の場所の確認

住民情報の確認

ARモードへの切り替え

イメージ






概要	ARアプリを起動し、解析結果一覧のなかから参照したいデータを選択する	赤いマーカが被災建物を示し、自分の位置を示す青い点からの距離や方向がわかる	マーカをタップすると、その建物に住んでいる世帯や住民の情報が閲覧できる	カメラアイコンをタップすると、自動ARモード（VPSモード）への切り替えを行う
----	---	--	--	--

Ⅲ. 実証システム > 7. ユーザインタフェース

ARアプリ | AR表示での被災住宅確認

ARアプリ内の地図およびAR画面内の3D都市モデルに被災情報を反映し、合わせて住民情報も表示する

VPSによる自己位置推定

3D都市モデルの読み込み

被災住民情報の確認

手動ARモードへの切り替え

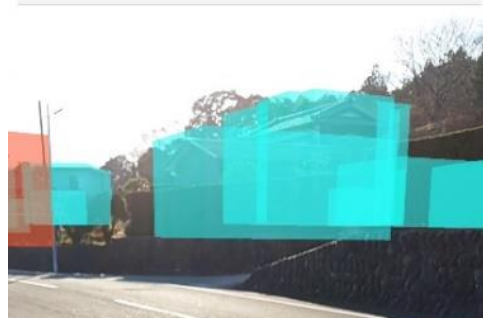
イメージ



地球アイコンが青くなるまで周りの景色にカメラを向ける

 自己位置推定が出来ていない状態

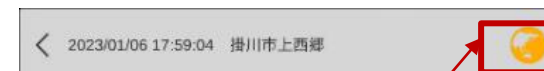
 自己位置推定が出来た状態



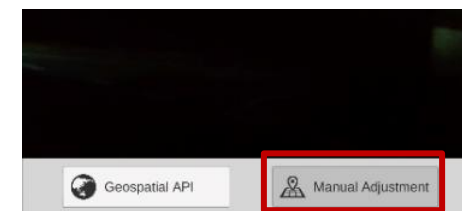
VPSによって推定された自己位置を基にAR表示される



赤い被災建物をタップ



地球アイコンが青くならない場合にここをタップ



「Manual Adjustment」ボタンをタップ

概要

端末のカメラを付近の建物や看板等に向け、VPSの自己位置推定を有効化する

VPSの有効化後、しばらく待つと自動的に3D都市モデルの読み込みが行なわれる

3D都市モデルをタップすると、その建物に住んでいる世帯や住民の情報が閲覧できる

VPSが有効でない時、右上の地球アイコンをタップして、手動ARモードに移行する

Ⅲ. 実証システム > 7. ユーザインタフェース ARアプリ | 地図とARによる位置合わせ

ARアプリ内の地図およびAR画面内の3D都市モデルにずれがある場合は向きや高さを調整し、ARが正しく表示されるように調整する

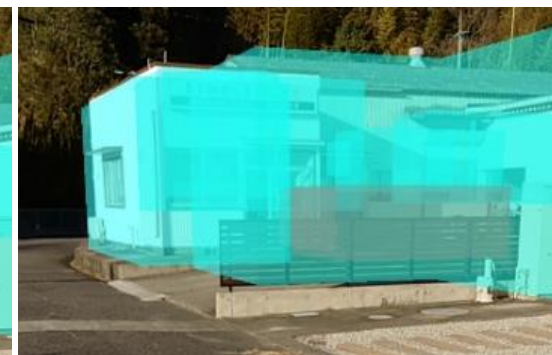
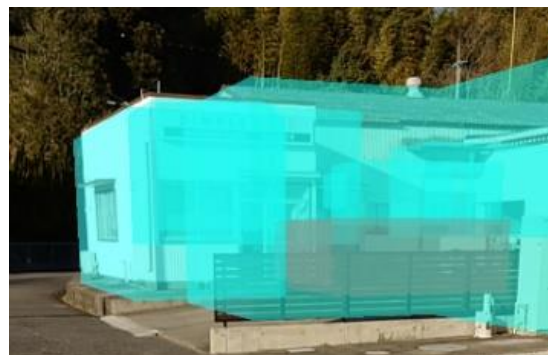
緯度／経度の指定

3D都市モデルの位置合わせ

3D都市モデルの向き調整

3D都市モデルの高さ調整

イメージ



概要

地図を表示し、自分が立っている場所をタップして、緯度・経度を指定する

「Load Model」ボタンをタップすると指定した緯度・経度情報を基に3D都市モデルが配置される

UIの左右ボタンをタップして、3D都市モデルの向きを現実の建物に合わせる

UIの上下ボタンをタップして、3D都市モデルの高さを現実の建物に合わせる

Ⅲ. 実証システム > 8. システムテスト結果

システムテスト結果一覧（1/2）

システムテストの結果は以下のとおり

対象	試験項目	確認内容	結果
SDTC	点群データの登録	点群データが解析サーバーのストレージに正常にコピーされるか	合格
	Webの解析指示UIの操作	Web画面上のUI操作が正しく行えるか	合格
	Web→解析APIの連携	Webブラウザから、解析APIが正常に呼び出せるか	合格
	解析サーバー→解析APIの連携	解析サーバーから、解析APIが正常に呼び出せるか	合格
	解析結果のWeb画面での表示	解析結果が正しくWeb画面に表示されるか	合格
	住民情報のWeb画面での表示	住民情報が正しくWeb画面に表示されるか	合格
ARアプリ	ARアプリでの解析結果データの取得	ARアプリで解析結果データが正しく取得できるか	合格
	ARアプリの画面UI操作	ARアプリ上のUI操作が正しく行えるか	合格
	3D都市モデルのARアプリでの表示	3D都市モデルのAR表示が正常に行えるか	合格
	解析結果のARアプリでの表示	解析結果が正しくARアプリに表示されるか	合格
	住民情報のARアプリでの表示	住民情報が正しくARアプリに表示されるか	合格

Ⅲ. 実証システム > 8. システムテスト結果 システムテスト結果一覧（2/2）

システムテストの結果は以下のとおり

対象	試験項目	確認内容	結果
API	点群データ登録	サーバーに点群データがアップロードされたときにデータベース上に正しく登録されるか	合格
	解析実行	HTTPリクエスト受信時に解析処理が正しく開始されるか	合格
	進捗取得	HTTPリクエスト受信時に解析処理の進捗状況が正しく出力できるか	合格
	結果取得	HTTPリクエスト受信時に解析結果が正しく出力できるか	合格
	被災住宅検出	サーバー上で解析が正しく実行されるか	合格

I. 実証概要

II. 実証技術の概要

III. 実証システム

IV. 実証技術の検証

V. 成果と課題

IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容 検証概要

被害状況検出の検証では土砂災害の被害パターンとして3つのパターンを設定し、それぞれのケースで被害を検出できるかを確認する

検証方法の概要

目的	開発したシステムの被害住居検出の精度を評価する
内容	ドローンにより取得した点群データと3D都市モデルを空間解析することで流出土砂の被害を受けた建築物を検出する機能の精度検証を行う
方法	- 被災状況を再現した点群データを用い、被災判定の閾値を変化させ、設定した被災住居が正しくシステムで検出されるかを検出率（一致率）で評価する - 被災状況の再現方法 - 点群データを編集し、土砂災害により被害を受けた住居を被災タイプ3パターンの計12戸再現 - 被災判定の閾値（以下の2パターン） ①面積比50%以下 ②面積比90%以下

対象エリア詳細



IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容

被害住居検出検証の全体像

実証地域の点群データを被災パターンに合わせて編集し、編集後のデータを基に被害をシステムが検出できるか分析精度を検証する

検証の手順

点群データの取得

- 実証地域の3次元計測を実施し、点群データを取得する

点群データの編集

- 点群データを編集し、想定する3パターンの被災状況を再現した点群データを作成する

被災住居の
分析の実行

- 点群データのアップロードを行い、Web アプリケーションから被災住居の分析を実行する

分析精度の確認

- 編集した住居と分析結果を突合し、検出率（一致率）から精度を確認する

被災住居の位置関係のイメージ



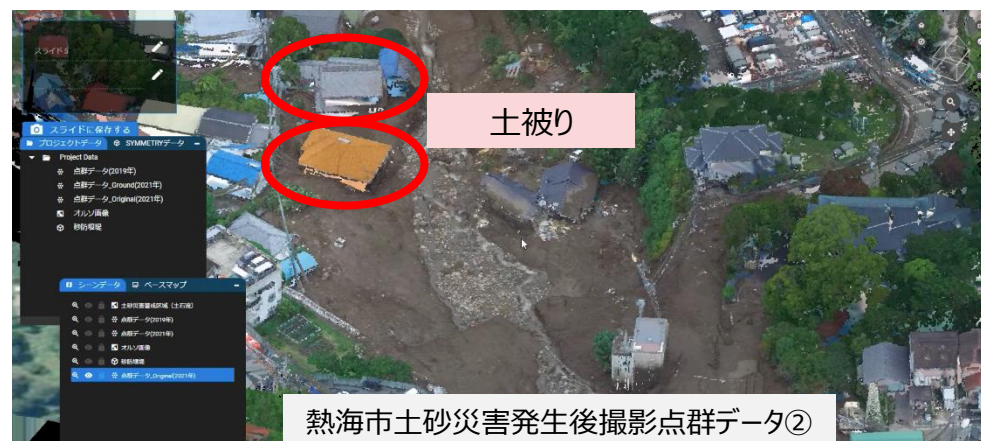
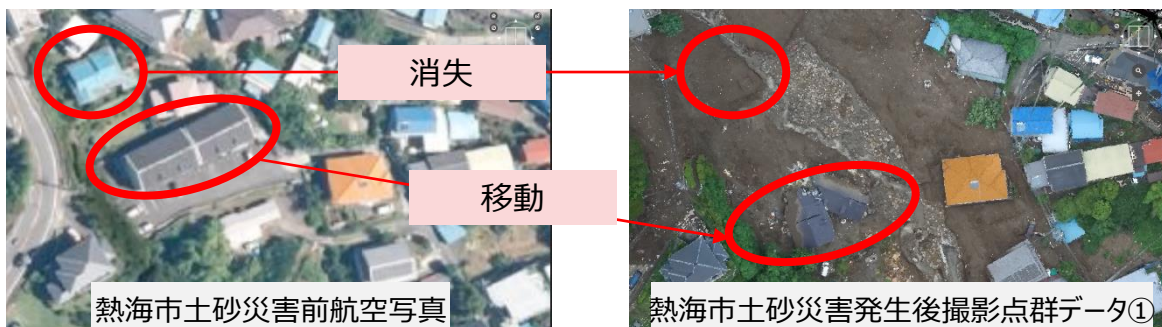
IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容 想定する被害パターン

実際の土砂災害を踏まえて想定する被害パターンを3つに設定する

土砂災害想定パターン考え方

- 2021年7月に発生した熱海市大規模土砂災害における土砂災害情報をもとに、ドローンによる上空観察から特定可能な被災状況の分類を以下の3分類と設定して、検証を行う
 - 住居消失**
 - 建物の消失（上空から観察できない状態）
 - 住居移動**
 - 土砂によって住居が移動
 - 住居土被り**
 - 住居の周りに土砂が堆積

全体画面イメージ



IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容 被害パターンの再現方法

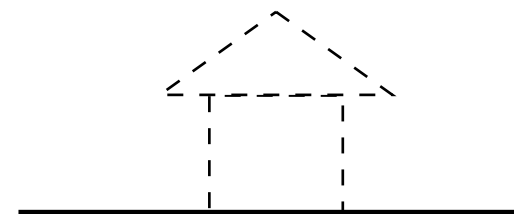
点群データを編集し3D都市モデルとのミスマッチを人為的に起こし、被災住居の特定が行えることを検証する

被害パターンの再現方法

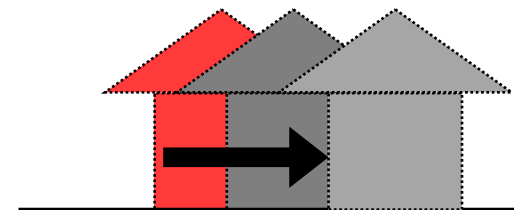
- 土砂災害により被害を受けた住居を3パターン計12戸作成
 - 点群編集ソフトウェアを使い、それぞれの被災パターンに合わせて取得した点群データの編集を行った
 - 内訳は以下の通り
 - 住居消滅：4戸
 - 住居移動：4戸
 - 住居土被り：4戸
- 各パターン4戸以上を用意することで、特定の家屋の形状や被災レベルに依らない評価を行う

点群データ加工による被害の再現イメージ

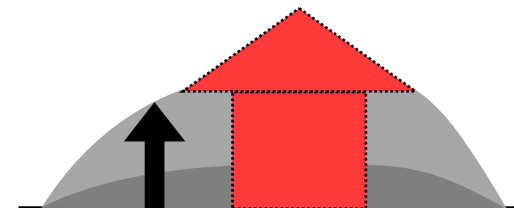
住居消失
建物を削除



住居移動
建物を横に移動



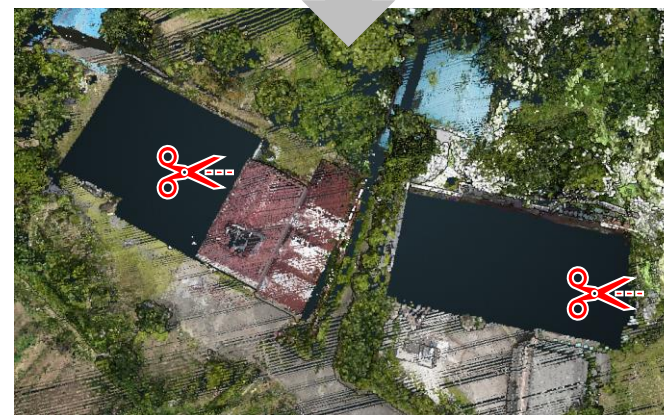
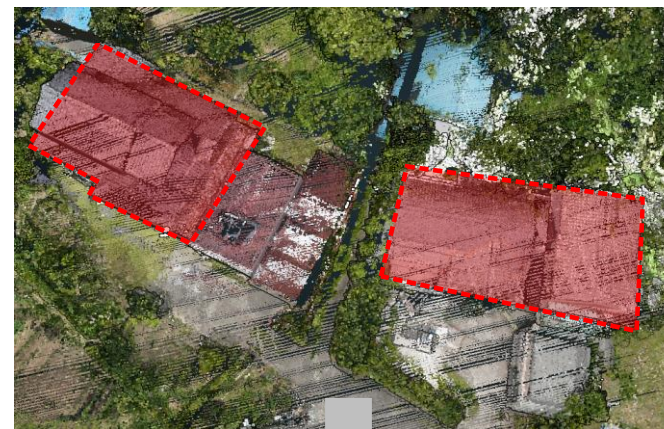
住居土被り
建物周囲の
地表面を縦に変形



IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容 住居消失

赤枠で囲った住居が消失したと仮定してシミュレーションを行うために、点群データを削除し、住居の消失が検知できるか検証する

点群データの編集（赤枠内が削除対象住居）



IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容

住居移動

赤枠で囲った住居が被害を受けたと仮定してシミュレーションを行うために、赤枠の住居が平行移動するように点群データを編集し、住居の被害が検知できるかを検証する

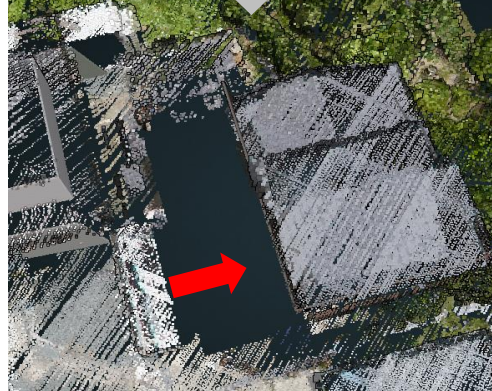
赤枠の住居を50~75%平行移動

赤枠の住居を25~50%平行移動

編集前



編集後



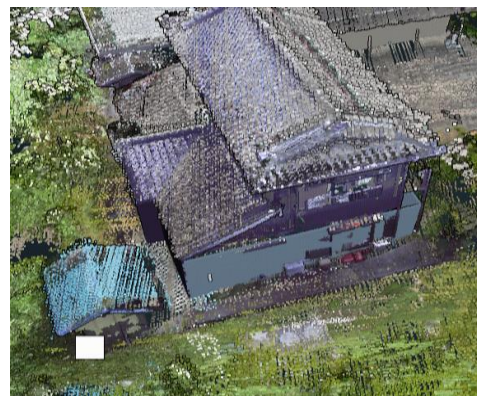
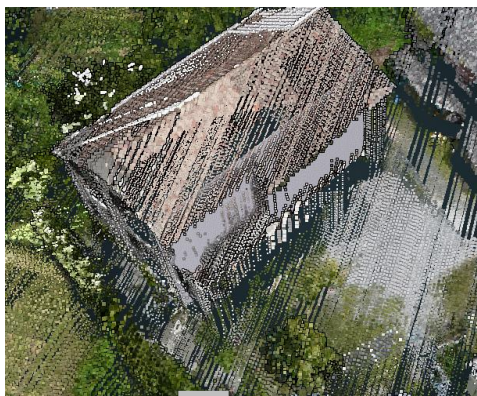
IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ① 検証内容 住居土被り

赤枠で囲った住居が土被りを受けたと仮定してシミュレーションを行うために、住居の地面を隆起したように点群データを編集し、住居の土被りが検知できるかを検証する

地面を建物高さの50~75%まで隆起

地面を建物高さの25~50%まで隆起

編集前



編集後



IV. 実証技術の検証 > 1. 被害住居検出の精度検証 > ②検証結果 検証結果

検証結果から、開発したアルゴリズムには被災パターンに対して得手不得手が明確に存在しており、実用化に向けてはすべてのパターンを網羅できるアルゴリズムの開発が肝要である

検証結果

被災パターン	検証数	検出率	
		面積比50%以下	面積比90%以下
住居消失	4戸	100% (4戸)	100% (4戸)
住居移動	4戸	50% (2戸)	100% (4戸)
住居土被り	4戸	25% (1戸)	50% (2戸)

考察・示唆

- **住居消失**
 - 閾値に依らずすべて検出できていることから、今回開発したアルゴリズムとは相性がいいと考えられる
- **住居移動**
 - 閾値と点群編集時の移動量がそのまま結果となっていることから、アルゴリズムは機能していると考えられる
 - 実際の被害状況进行分析し、実態に合った閾値を設定する・選択できるようにすることが実用化に向けた課題となる
- **住居土被り**
 - 検出率は低く面積比を90%とした場合においても検出率は50%と実用には改善が必要である
 - 地表面と建物の分離に課題があると考えられるため、アルゴリズムの改善だけでなく地表面データを別のデータソースから得られれば精度を改善できる可能性がある

想定シナリオ	検証項目	検証内容	検証条件
被災建物が少ない災害 (建物を使った位置合わせが可能)	VPS の精度検証	• 明るい時間帯に、VPSを利用して自己位置推定を行い、3D都市モデルと現実空間の建物の一致度を検証する • 日没後、十分な明るさがない時間帯において、VPSがどの程度利用できるかを確認する	• 場所 - 掛川市上西郷石ヶ谷の屋外 • 日時 - 2023年1月11日（水） • 日中（13:00-17:00） • 日没後（日没16:56~17:40） • 利用端末 - SONY Xperia 1Ⅲ - SO-G03を利用する
被災建物が多い災害 (建物を使った位置合わせが困難)	SLAM の精度検証	• VPSを使わず、地図を利用して手動で位置合わせを行った後、SLAMを利用して自己位置推定を行った場合の、3D都市モデルと現実空間の建物の一致度を検証する	

IV. 実証技術の検証 > 2. ARアプリにおける自己位置推定の精度検証 > ① 検証内容

VPSの精度検証

3D都市モデルを現実世界の建物に重畳するARアプリを実装するにあたり、VPS（ARCore Geospatial API）は、明るい時間帯において十分な位置精度を持っているかどうかを検証する

検証の手順

右図の①地点でARモード（VPS）を有効にし、3D都市モデルと現実の建物の位置の一致度合いを確認する

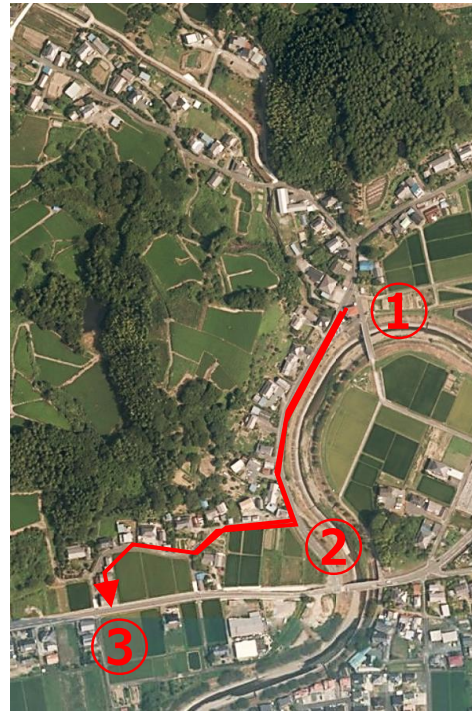
上記で正しくVPSが動作していることを確認したら、赤い歩行ルートに従って歩く

途中、②地点において、3D都市モデルと現実の建物の位置の一致度合いを確認する

②地点の後、右側の細い道に曲がる

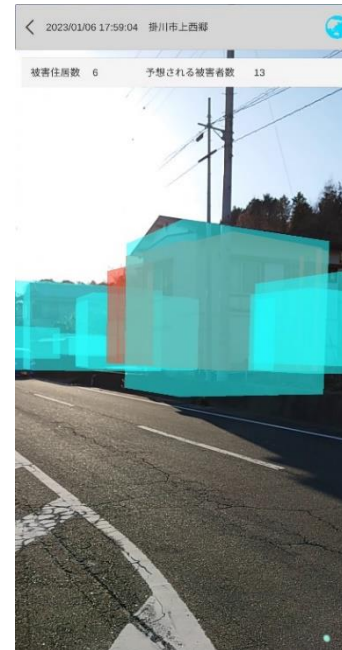
最終的に、③地点においても、3D都市モデルと現実の建物の位置の一致度合いを目測で確かめる

検証地点（概要）



検証地点の様子

地点①



地点②



地点③



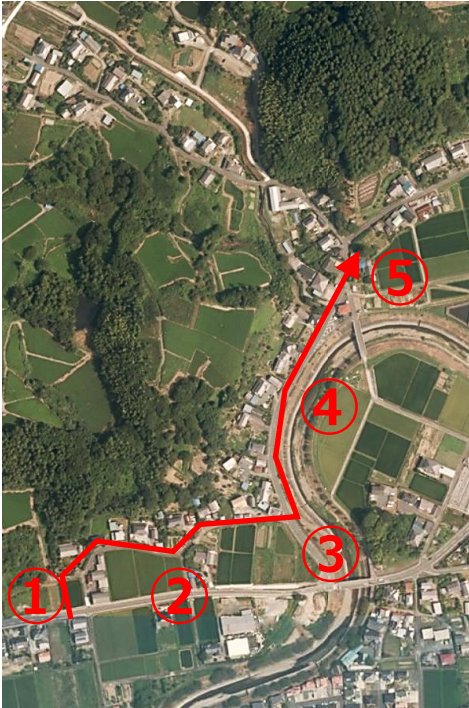
自己位置推定を自動でできなかったケースと仮定して手動で位置を合わせた後の精度がどのように変化していくかを以下4つのポイントで検証を行った

検証の手順

- ①地点で手動位置合わせを行い、その後SLAMによる自己位置推定を行いながら、②地点を経由して③地点まで歩いたときのそれぞれの位置ズレの度合いを確認する
- その後、③地点で再度手動位置合わせを行い、SLAMによる自己位置推定を行いながら、④地点を経由して⑤地点まで歩いたときの位置ズレの度合いを確認する

区間	距離
①地点～②地点	約110メートル
②地点～③地点	約115メートル
③地点～④地点	約110メートル
④地点～⑤地点	約150メートル

歩行ルート



IV. 実証技術の検証 > 2. ARアプリにおける自己位置推定の精度検証 > ②検証結果

検証結果サマリ



検証項目	検証結果	考察
VPSの精度検証	• VPS開始時、実際の建物と3D都市モデルの位置には1～2メートル程度のズレが見られた • 経路上も誤差を一定以内の範囲に保ち、ズレが拡大することはなかった • 日没後、40分程度で認識しなくなった（日没後、17時35分頃にVPSの認識NG）	• 3D都市モデルを現実空間の建物に重畳するARアプリを実装するにあたり、VPS（ARCore Geospatial API）は、明るい時間帯において十分な位置精度を持っている（隣の家と間違えるレベルではない） - 日没後は30分程度は問題ないと考えられるが、端末の暗所性能次第のため、原則として日中の利用が望ましい
SLAMの精度検証	• 位置合わせ後約110メートル地点では目測で3～5メートル程度のズレに拡大し、辛うじて本来の建物と重なっている状況 • さらに100メートル進んだ地点では、目測で10～15メートル程度のズレになっており、本来の建物とは完全に重ならない位置に3D都市モデルが表示される	• SLAMを使って自己位置推定をする場合、位置合わせをしてから100メートル以内で利用するのが望ましい

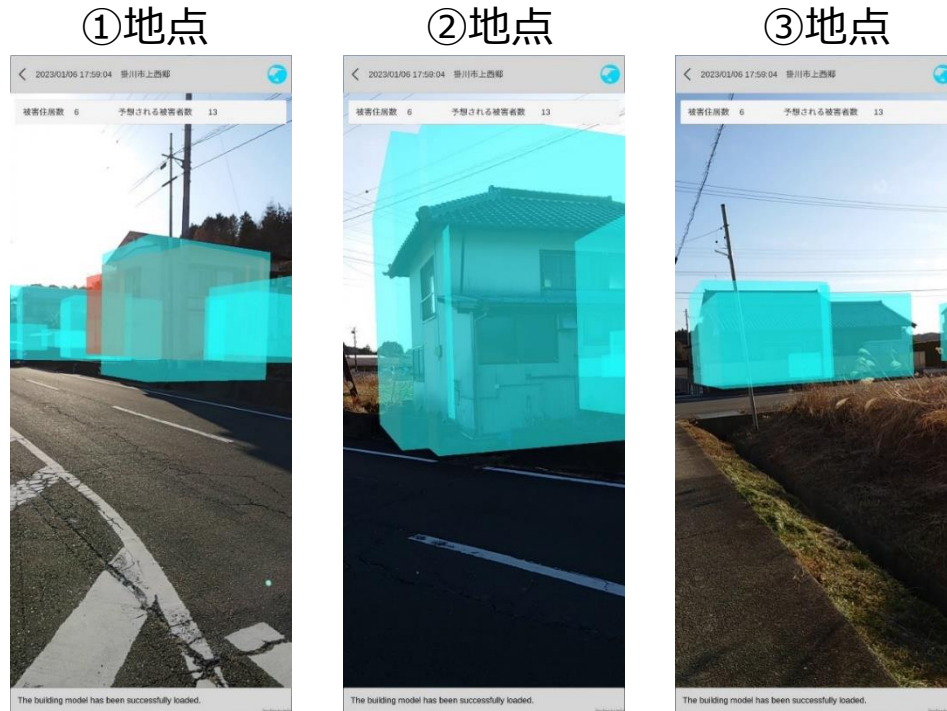
IV. 実証技術の検証 > 2. ARアプリにおける自己位置推定の精度検証 > ②検証結果

VPSの精度検証結果

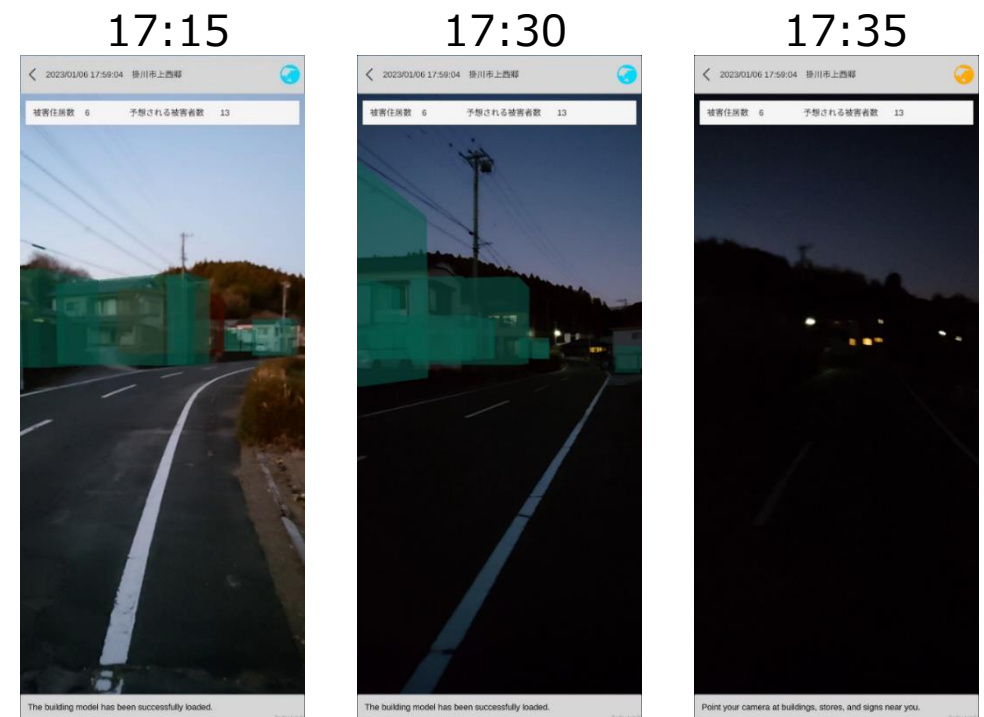
VPSが動作している日中（日没後30分程度まで）であれば1～2メートルのズレで済むため、被災住居の特定に十分に利用可能な精度と考えられる

検証結果（日中）

検証結果（日没後）



VPS開始時、実際の建物と3D都市モデルの位置には1～2メートル程度のズレが見られた。ただし、この程度の位置ズレであれば隣の家と間違えることはない



16:56の日没後約30分経過した17:30頃でも問題なくVPSが動作した。また、暗い場合でも一度VPSが認識できさえすれば、その後しばらく歩いた場合でもズレが発生することはなかった

IV. 実証技術の検証 > 2. ARアプリにおける自己位置推定の精度検証 > ②検証結果

PLATEAU
by MLIT

SLAMの精度検証結果

SLAMを使って自己位置推定をする場合、他の家と見間違えないようにするためには位置合わせをしてから100メートル以内で利用するのが望ましい

	地点①	地点②	地点③	地点④	地点⑤	
イメージ						
結果	（VPS利用時と同等レベル（1～2メートル程度のズレ）で位置合わせ）	- 地点①から110m地点 - 目測で3～5メートル程度のズレに拡大 - 辛うじて、本来の建物と重なっている状況	- 地点①から225m地点 - 目測で10～15メートル程度のズレ - 本来の建物とは完全に重ならない位置に3D都市モデルが表示	（再度位置合わせを実施）	- 地点③から110m地点 - 目測で3～5メートル程度のズレに拡大 - 辛うじて、本来の建物と重なっている状況	- 地点③から260m地点 - 目測で15～20メートル程度のズレ - 本来の建物とは完全に重ならない位置に3D都市モデルが表示

IV. 実証技術の検証 > 3. 開発システムの有用性検証 > ① 検証内容 検証概要

有用性ヒアリングの実施概要は下記のとおり

有用性検証の概要		参加者詳細		
項目	内容	団体名	部署	人数
目的	開発した災害時被害住居検出システム、ARアプリケーションの実演及びデモ体験を通じて、システムの実用性・有用性に関するヒアリングを目的とする	掛川市役所	副市長	1名
実施期間	2023/1/12（火） 10:00-12:00		DX推進課	5名
実施場所	掛川市中央消防署3F会議室		消防総務課	3名
実施内容	1. 実証の流れ、アンケート概要説明 2. Webシステム動作確認 3. ARアプリの実演及びデモ体験 4. フィードバック、質疑応答、アンケート		都市建設部	1名
			都市政策課	2名
			土木防災課	1名
			基盤整備課	3名
			維持管理課	1名
		掛川市中央消防局	掛川市消防署職員	2名
		合計	19名	

IV. 実証技術の検証 > 3. 開発システムの有用性検証 > ① 検証内容 アンケート項目と回答方法

被害住居検出システムに関するアンケート項目は下記のとおり設定した

対象	項目	観点	質問項目	回答方法
被害状況検出システム	有用性検証	情報可視化	ドローンによる被害状況を撮影し、分析することで、人間の目視ではなく、データとして被災していることが分かるようになるか	- 回答の選択肢を - はい - いいえ - どちらでもない の3段階に設定 - 選択理由を自由記述で回答
			被災状況は正しい位置に表示されているか	
		行政実務での活用	住民情報と被災建物を同時に表示することで、救助活動の判断をサポートできるか	
			被災状況のデータから救助計画を検討するデータとして利用可能か	
	操作性	使いやすさ	被災状況解析機能は使いやすいか	
			表示データ調整機能は分かりやすいか	
		見やすさ・分かりやすさ	住民情報表示機能は分かりやすいか	
ARアプリ	有用性検証	情報可視化	被災状況は正しい位置に表示されているか	
	操作性	使いやすさ	位置特定機能は使いやすいか	
			位置調整機能は使いやすいか	
		見やすさ・分かりやすさ	マップ表示画面は見やすいか	
			AR表示機能は分かりやすいか	

IV. 実証技術の検証 > 3. 開発システムの有用性検証 > ②検証結果 検証結果サマリ

被害住居検出システムとARアプリは両者とも有用性が確認されたが、操作性や精度向上を求める声が寄せられ、実用化に向けてシステムの改善が必要である

検証対象	結果
被害住居検出システム	- 土砂災害のみならず火事や洪水等の災害における活用の可能性が確認できた - 救助活動時に住民情報を可視化することで時間短縮の有用性が検証できた一方で、住民情報は個人情報であるため、自治体との災害協定等による取り扱い方法の事前協議の必要がある 3D都市モデルと合わせることで被害状況を俯瞰して確認ができ、救助計画への活用の可能性がある - 災害時、ユーザーが直感的に使えるよう、UI改善や、チュートリアル作成が必要 - 被害パターンによっては、「屋根」による被災状況検知は識別されにくい場合もあった
ARアプリ	- VPS及び位置情報補正機能を用いた被害住居のARでの可視化で調査・救助活動への有用性を確認することができた - 被害住居のARでの可視化機能を拡張することで救助者がより安全かつ簡易に使用できる

IV. 実証技術の検証 > 3. 開発システムの有用性検証 > ②検証結果 被害住居検出システム（1/2）

被害状況を可視化する機能に対して好意的な印象が得られたが、操作性や精度の向上を求める声も寄せられた

質問項目	回答の分布（N=6） はい どちらとも言えない いいえ			コメント（抜粋）
ドローンによる被害状況を撮影し、分析することで、人間の目視ではなく、データとして被災していることが分かるようになるか	100%			- 被災後の差分が可視化できるため - 点群データとの比較で、迅速な確認が出来ると思う - 目視によらないことが点群データのメリットであるため
被災住居は正しい位置に表示されているか	68%	16%	16%	- 調整が難しい印象を持った - 地形等によって、手動補正が必要
住民情報と被災建物を同時に表示することで、救助活動ための判断をサポートできるか	100%			- 住民基本台帳のデータと連携することで、優先度の判断などの手助けになる - 救助活動には住民情報が不可欠だから
被災状況のデータから復旧計画を検討するデータとして利用可能か	84%	16%		- 全体の被害状況を確認しやすいと思う - 被災状況を俯瞰して見ることが出来るため - 被災範囲や状況がわかることで、初動が早くなる

IV. 実証技術の検証 > 3. 開発システムの有用性検証 > ②検証結果 被害住居検出システム（2/2）

被災状況可視化システムの表示機能や操作性には改善を求める意見が多く寄せられており、改善の必要性がうかがえる

質問項目	回答の分布（N=6）			コメント（抜粋）
	はい	どちらとも言えない	いいえ	
被災状況解析機能は使いやすいか				- 実際の仕組みを使っていないために判断できないが、説明された感じでは使いやすいそうだった - 建物だけでなく、斜面などの変化も含め、複合情報が必要
	100%			
表示データ高さ調整機能は分かりやすいか				- 説明を受けて理解できた - マニュアルでの調整は手間になるが、VPSが使えない時には必要だと思った - 実際の仕組みを使っていないため、わからない
	32%	68%		
解析後のマップ表示機能は分かりやすいか				- 全体像をイメージしやすい - 説明を聞いただけなのでわからないことが多いが、使い勝手は良さそうな印象を受けた - 改良の余地あり
	68%		32%	
住民情報表示機能は分かりやすいか				- どのような属性の人が住んでいるかが一覧できたため - 救助の優先順位等の決定がしやすくなると思う
	32%	68%		

ARアプリは操作が難しかったという意見があった一方で、色分け機能等によって表示画面はわかりやすいと好意的な意見が寄せられた

質問項目	回答の分布（N=4） はい どちらとも言えない いいえ	コメント（抜粋）
被災状況は正しい位置に表示されているか	75% 25%	- 道路より離れた場所での土砂で埋まっている家屋がわかりにくい - 自動修正機能により位置情報が修正されるのが良い
位置特定機能は使いやすいか	50% 50%	- アイコンなどを見やすくしてもらいたい - 微調整機能を活用するとさらに高い精度になった - 自動機能は扱いやすい。ただ、手動機能は慣れが必要
位置調整機能は使いやすいか	50% 50%	- 夜間の使用が操作しにくい - 少し操作が難しいと感じた - 直感的にわかりやすかった
AR表示機能は分かりやすいか	75% 25%	- 色分けされ分かりやすい - 広範囲に確認できるように拡張機能が欲しい - 現場での位置確認がすごく分かりやすかった
被害建物表示機能は見やすいか	75% 25%	- 精度がもう少し高くなると使いやすい - 色の変化により直感的にわかりやすい

I. 実証概要

II. 実証技術の概要

III. 実証システム

IV. 実証技術の検証

V. 成果と課題

V. 成果と課題 > 1. 今年度の実証で得られた成果

① 3D都市モデルによる技術面での優位性

項目	想定される技術面での優位性
点群データからの建物領域の抽出	3D都市モデル（建築物LOD1）に含まれているFootprintの情報を利用することで、点群データから正確に建物の領域を抽出することが可能と分かった - 建物ごとに被災状況の判定を行うために広範囲の点群データから自動的に1軒ずつ建物領域を抽出することができた
ARアプリケーションでの活用	3D都市モデル（建築物LOD1）の建物の位置やFootprint形状が十分な精度を持っており、ARアプリケーションで活用できることが分かった - AR表示において、現実世界の建物と重ね合わせても位置と形状がおおむね一致しており、3D都市モデルと実際の建物に関する様々なデータ（本実証実験では住民基本台帳データ）を結び付けて管理することができる - VPSが利用できない場合であっても、AR表示された3D都市モデルと現実世界の建物が一致するように、手動で視覚的に位置を調整することにより、現実世界の座標とコンピューター世界の座標を合わせることができる

V. 成果と課題 > 1. 今年度の実証で得られた成果

②3D都市モデルによるビジネス面での優位性

項目	想定されるビジネス面での優位性
救援活動支援への適用性	被害住居を3D都市モデルと合わせて可視化することで、周辺環境との関係性を見ながら視覚的に被害規模を把握し、救助計画や復興計画を検討できる

V. 成果と課題 > 2. 今後の取り組みに向けた課題 活用にあたっての課題（1/2）

項目	活用にあたっての課題
他地域への適用	● 該当地域の3D都市モデルやShapefileが不可欠であり、かつ最新の情報に更新されている必要がある - 被災住宅判定技術はオープンデータの3D都市モデルを正常な住宅の位置の真値として扱っている ● 住民基本台帳の位置情報がない自治体は、住所から住所辞書を使用したアドレスマッチング（ジオコーディング）を行う必要がある - アドレスマッチングの結果、アンマッチや3D都市モデルとの位置の整合がとれない世帯は位置補正を行う必要がある - 住所辞書は号レベル（建物レベル）の情報が記載された市販品か庁内の地番図や住居表示台帳から作成する
詳細な損傷の分析	● 被災住宅判定を様々な被災状況に対応できるようにより形状的な特徴に注目して解析を行い損傷の程度を判定できるように改善する必要がある - 現在の判定は被災している住宅を大部分が土砂に埋まっている、あるいは大きく移動していると仮定しているが、実際の災害においてはより様々な被災状況が存在する - 手法の一つとして、過去の計測点群データとの比較を行うことで、その差分から変異を解析することができるが、そのためには一定期間ごとにレーザー計測を行い、そのデータを公開し入手しやすくする取り組みが必要である
要介護情報の表示	● 救助の際の優先度判定のために必要な要介護情報をシステム上で表示するためには、住民基本台帳と要介護情報との紐づけが必要である。そのため、キーとなる番号を住民基本台帳と要介護情報の各データに保有しておく必要がある

V. 成果と課題 > 2. 今後の取り組みに向けた課題 活用にあたっての課題（2/2）

項目	活用にあたっての課題
住民基本台帳や要介護情報の使用における事前対応	- 災害時に住民基本台帳をインターネット回線上のシステムへ搭載できる環境を整えるために、個人情報保護審議会等での審議、関係部署との調整を行い、庁内合意、搭載項目を決定することが必要 - 地方公共団体の情報セキュリティポリシーで、住民基本台帳をはじめとする個人情報の活用には制限があるため、事前の討議・合意形成が必須である - 被害を受けた範囲の住民基本台帳データをGISにて抽出する操作方法や操作する人、事業者へのデータの授受方法などを記載した運用マニュアルの策定が必要となる
住民基本台帳の使用上の注意点	- 住民基本台帳を使用する上では以下の点で最新情報が反映されていない可能性があり、最新情報をいかに入手し続けるかが課題となる - 長期出張や施設への入居した場合など住民基本台帳の住所と実際の居住場所や世帯構成が異なる - 住民基本台帳（座標情報付）の更新頻度によっては、時点が古い場合がある
アプリケーションのユーザインタフェース	- 実際に災害が発生し、緊急対応が必要な場面を想定し、できるだけ少ない操作手順でスピーディに使い、誤操作をさせないように磨きこみが重要 - 特にARアプリケーションは災害現場で迅速に利用する必要があり、中でもVPSが利用できない場合の位置合わせ機能は、位置精度と操作スピードを両立させる検討が必要である

用語集（1/6）

用語		内容
ア行	アドレスマッチング	住所や郵便番号、場所の名前などのテーブル情報を地図上に展開し、緯度経度の座標値を与える処理
	Amazon S3	AWSで提供されているクラウド上のファイルストレージサービス
	Web API	- Web Application Programming Interfaceの略 - コンピュータプログラムの提供する機能を外部の別のプログラムから呼び出して利用するための手順・規約（API : Application Programming Interface）の種類の一つで、HTTPなどWebの技術を用いて構築されたもの - Web APIを使用することで異なるプログラムやサイト間でデータをやり取りすることが可能となる
	HTTP	- Hypertext Transfer Protocolの略 - Web情報をやりとりするためのプロトコル（通信規則）
	AR Core	- Google社が提供するAndroid端末向けのARアプリ開発用プラットフォーム - スマホに搭載されたカメラや加速度センサーを利用してARコンテンツを構築するための自己位置を推定する技術
	API	- Application Interfaceの略 - ソフトウェア間で情報をやり取りするための仕組み
	FBX	- Autodesk社が開発した3Dモデリング、アニメーション、レンダリングなどのソフトウェア間でのデータ交換を可能にするファイルフォーマット 3Dモデル、アニメーション、テクスチャ、ライティング、カメラなどのデータを保存可能 - デファクトスタンダードとなっており、様々なソフトウェアで本形式を扱うことができる

用語集（2/6）

用語		内容
サ行	サーバー	- システムを構成する機器の一種 - 提供する側のコンピュータ、PCを指す
	座標系	地球上の位置を座標で表すための原点や座標の単位などのルールのこと
	GIS	- Geographic Information Systemの略 - 地理情報システムとも呼ばれる - 位置に関する情報を持ったデータ（空間データ）を総合的に管理・加工し、視覚的に表示し、高度な分析や迅速な判断を可能にする技術
	JSON	- JavaScript Object Notationの略 - 構造化されたデータの交換を軽量に行うことができるテキストフォーマット
	SHPファイル	- Shapefileの略 - 地理情報システム（GIS）間でのデータの相互運用におけるオープン標準として用いられるファイル形式 - ESRI社が開発した地理空間データを保存するためのフォーマットで、地理空間データを点や線、面で表現することができる
	GeoTIFF	- TIFF（Tagged Image File Format）に地理空間情報を追加した画像データのフォーマット - 地図上に画像を表示することができるため、GIS（地理情報システム）やCAD（Computer-Aided Design）で使われることが多い
	CityGML	- 都市空間データを表現するためのデータモデル - 都市計画や土木建設、災害対応などでデータ共有や分析を目的として使用されるほか、Web上で都市空間データを表示するために使われることもある



用語集 (3/6)

用語		内容
サ行	住民基本台帳	氏名、生年月日、性別、住所などが記載された住民票を編成したもの
	数値標高モデル	- Digital Elevation Modelと同義 - 国土地理院によって作成された、写真測量やLIDAR、干渉合成開口レーダー、測量などの技術により取得された標高のメッシュデータ - 基盤地図情報として、5mメッシュ及び10mメッシュの2種類のデータがある
	SLAM	- Simultaneous Localization and Mappingの略 - 同時位置推定と地図作成を行う技術 - ロボットなどの自律移動体が、自らがいる環境を探索しながら、自身の位置を推定しながら地図を作成する
	3D Tiles	- 大規模な3D地理空間コンテンツをストリーミングおよびレンダリングするために設計されたファイル形式 - 参考: https://github.com/CesiumGS/3d-tiles/blob/main/specification/README.md
	3Dプレゼンテーション	- 建物や構造物などを立体的に表現すること - またはその画面表示
	測量データ	- 航空機やドローンなどでレーザースキャナー等で計測したデータ - または専用の計測器を使用し、得ることができるデータ

用語集 (4/6)

用語		内容
タ行	ダイアログ	パソコンなどの画面上で表示されるウインドウの種類の一つ
	タイムスタンプ	事象が発生した日時の情報を保存し、証明できるもの
	タップ	スマートフォン等触れることで操作できる画面で、軽くたたくように触れる操作
	ダミーデータ	- 実際のデータではなく、架空、疑似的なデータのこと - テストデータとも言う
	DHM	- Digital Height Modelの略 - 地表と地形を数値化したデータ - DSMとDTMの差分から生成される
	DSM	- Digital Surface Modelの略 - 地表の高度を数値化したデータ
	DJI Zenmuse L1	- DJI社が開発したLiDARセンサー - 航空機やUAVに搭載することができる
	DTM	- Digital Terrain Modelの略 - 地形の高度を数値化したデータ
	DEM	- 数値標高モデル - 地形をサーフェスで表現したデータ
ナ行	ノード	- 結び目、節や接点 - 特定の機器や場所を指す場合もある

用語集 (5/6)

用語		内容
八行	Python	1990年代初頭ごろから公開されているプログラミング言語 2020年度からは、国家資格である基本情報技術者試験で出題されるプログラミング言語に追加された
	VPS	- Visual Positioning Systemの略 - カメラ等から得られた周囲の映像から、特徴点を抽出し、データベース上のデータと照合することで、現在位置を測定するための技術
	フィルタリング	必要なデータのみとして必要ないものを取り除くことや、測量データから建物や樹木などの高さを取り除くこと
	フットプリント	- 敷地に対する建築の占有領域 - 家形ともいう - 国土交通省が公開している「3D都市モデル標準製品仕様書」では「地表と壁面の交線」と定義されている
	フレームワーク	システムやアプリケーションを開発する際の土台として使用、機能させるソフトウェア・仕組みのこと
	ポリゴン	3Dのコンピュータグラフィックスで立体を表現する際に用いられる、多角形の平面データ
マ行	マスク	データを加工してデータを匿名化すること
	メッシュ	地図を決められた大きさに分割、格子状に区切ったもの
	モーションセンサー	物の加速度・傾き・方向等を検知して信号として出力する装置

用語集 (6/6)

用語		内容
ヤ行	UAV	- Unmanned Aerial Vehicleの略 - ドローン等、人間が操縦しない無人航空機
	UAV LiDAR	Unmanned Aerial Vehicleと呼ばれるドローン等、人間が操縦しない無人航空機レーザーを使用して周囲の環境や物体を計測する技術
	Unixtime	コンピュータやシステム上での時刻表現規則の一つ
	Unity	- Unity Technology社が開発・提供するゲームエンジンおよび開発環境 - スマートフォン向けのアプリや家庭用ゲーム機・ウェブなどの様々なプラットフォームで実行可能なアプリケーションを開発することができる
ラ行	LiDAR	- Light Detection And Rangingの略 - レーザー光を照射して、その反射光の情報をもとに、対象物までの距離や対象物の形などを計測する技術
	ライブラリ	汎用性の高いプログラムを定形化し、他のシステムやプログラム言語から利用しやすいようにまとめたもの
	LAS	- LiDARによって取得された点群の標準フォーマット - 米国写真測量学会が制定している
	ラスターデータ	- グリッド上に構成されたデータのこと - 通常の写真などの画像データは、ラスターデータになる
	LAStools	Rapidlasso社が開発したLASデータの処理を行うためのプログラム群
	レーザー計測	投影したレーダーと反射して受光するまでの時間を計測して、距離を測定する方法

三次元データを用いた土砂災害対策の推進 技術検証レポート

令和5年3月 発行

委託者：国土交通省 都市局 都市政策課

受託者：Symmetry Dimensions Inc.・株式会社パスコ

本報告書は、Symmetry Dimensions Inc.・株式会社パスコが国土交通省との間で締結した業務委託契約書に基づき作成したものです。受託者の作業は、本報告書に記載された特定の
の
手
続
や
分
析
に
限
定
さ
れ
て
お
り
、
令
和
5
年
3
月
ま
で
に
入
手
し
た
情
報
に
の
み
基
づ
い
て
実
施
し
て
お
り
ま
す
。
従
っ
て
、
令
和
5
年
4
月
以
降
に
環
境
や
状
況
の
変
化
が
あ
っ
た
と
し
て
も
、
本
報
告
書
に
記
載
さ
れ
て
い
る
内
容
に
は
反
映
さ
れ
て
お
り
ま
せ
ん
。