

# 3D都市モデルの更新優先度マップ<sup>o</sup> 技術検証レポート

Technical Report on Map Prioritization for Updating 3D City Models



PLATEAU  
by MLIT



# 目次

## I. 実証概要

1. 全体概要	4
2. 実施体制	6
3. 実証エリア	7
4. スケジュール	15

## II. 実証技術の概要

1. 活用技術	17
2. ALOS-3衛星	18
3. ASNARO-1衛星	20
4. AI (深層学習)	21
5. PyTorch	22
6. OpenCV	23
7. GDAL	24
8. Httpx	25
9. Rasterio	26
10. GeoPandas	27
11. PySAL	28
12. ABCI	29
13. QGIS	30
14. ERDAS IMAGINE	31
15. Tellus	32

## III. 実証システム

1. 実証フロー	34
2. 想定事業機会	35
3. アーキテクチャ全体図	36
4. システム機能	38
5. アルゴリズム	60
6. データ	
① 活用データ	66
② データ処理	72
③ 出力データ	89
7. ユーザインタフェース	96
8. システムテスト結果	107

## IV. 実証技術の検証

5. 更新優先度マップにおけるAIモデルの精度検証	
① 検証内容	109
② 検証結果	114
実証システムの価値検証	
6. ① 検証内容	127
② 検証結果	135

# 目次

## 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 140
  - ② 3D都市モデルによるビジネス面での優位性 141
- 2. 今後の取り組みに向けた課題 142

用語集 144

# **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)

<b>ユースケース名</b>	3D都市モデルの更新優先度マップ
<b>実施場所</b>	全国8都市（北海道札幌市/福島県郡山市/東京都23区/神奈川県横浜市/静岡県沼津市/静岡県掛川市/大阪府大阪市/大阪府豊中市）
<b>目標・課題 ・創出価値</b>	<ul style="list-style-type: none"><li>• 3D都市モデルの活用を進めるには、日々変化する現実空間をリアルタイムで仮想空間上に再現することが求められる一方で、3D都市モデルは既存測量成果を利用して整備するため、更新を頻繁に行うことが難しい。</li><li>• 今回の実証実験では、3D都市モデル整備に利用した航空写真と衛星画像を比較し、差分を検出するAIモデルを開発することで、低コストでの差分検出・更新優先度マップの構築・現実空間との差異の可視化を実現し、自治体へ3D都市モデルの更新を促す情報の提供が可能か検証する。</li><li>• 2023年から運用予定であったALOS-3衛星の衛星画像を活用することで、3D都市モデルの変化情報の取得を低コストで実現し、自治体による能動的・継続的な運用と民間領域での利活用による3D都市モデルのエコシステム活性化を目指す。</li></ul>
<b>ユースケース の概要</b>	<ul style="list-style-type: none"><li>• 3D都市モデル作成の元データとなる航空写真と、より撮影頻度が高い衛星画像とを比較し、建築物の変化を判定してメッシュ単位で評価することで、3D都市モデルの更新優先度を判別できるシステムを開発する。</li><li>• 出力データを更新優先度マップとしてPLATEAU VIEWに掲載し、新旧2時期画像と各メッシュの更新優先度を確認しながら3D都市モデルの整備・更新に関わる自治体・民間事業者の実務における有用性をヒアリングを行う。</li></ul>

# I. 実証概要 > 1. 全体概要

## 全体概要 (2/2)

<b>実証仮説</b>	<ul style="list-style-type: none"><li>● 3D都市モデルの整備に利用した航空写真と撮影頻度が高い衛星画像との差分から更新すべき箇所を検出・可視化するAIモデルを開発することで、3D都市モデルの要更新情報を低コストに取得できる。</li><li>● 上記を更新優先度マップとして活用することによって、以下の業務において低コスト化・効率化の観点で寄与する。<ul style="list-style-type: none"><li>- 自治体の都市計画基本図（基盤地図、DM等）等の部分更新</li><li>- 民間企業で管理・販売等をしている地図情報の部分更新や管理</li></ul></li></ul>
<b>検証ポイント</b>	<ul style="list-style-type: none"><li>● 更新優先度マップ作成におけるAIモデルの精度検証<ul style="list-style-type: none"><li>- 対象都市の3D都市モデル・航空写真・ALOS-3衛星シミュレーション画像を元に、更新優先度マップを試作</li><li>- AI教師データの一部エリアを用いて、更新優先度マップ作成におけるAIモデルの精度を検証</li></ul></li><li>● 自治体・民間企業への更新優先度マップの有用性検証<ul style="list-style-type: none"><li>- 更新優先度マップのサンプルを用いて、部分更新の実現性・コスト削減見込み・運用性・精度要求等の観点で、実現可能性を確認</li></ul></li></ul>

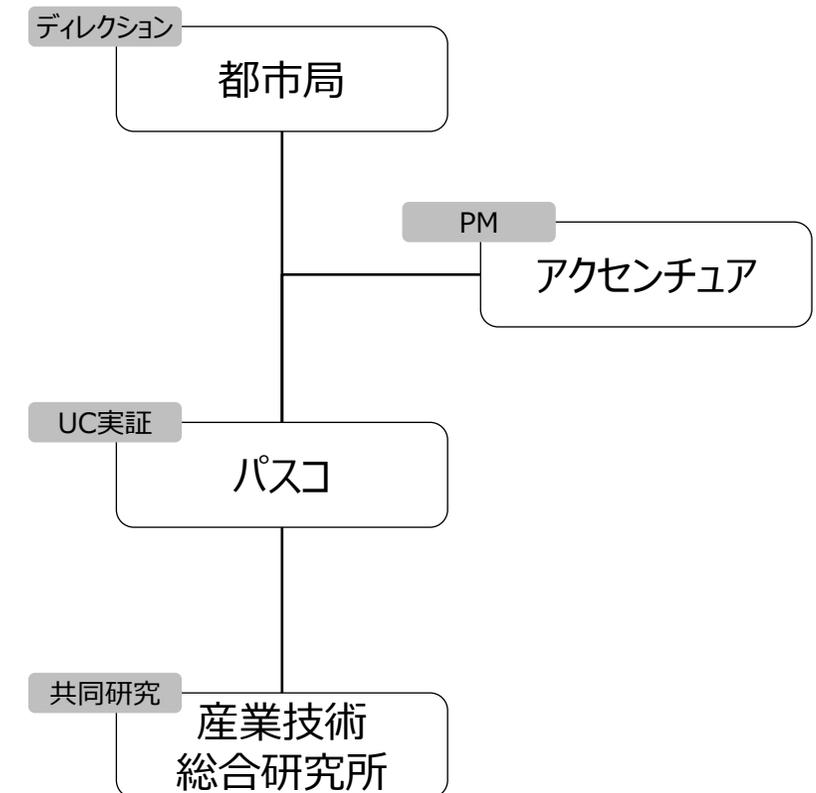
# I. 実証概要 > 2. 実施体制

## 実施体制

各主体の役割

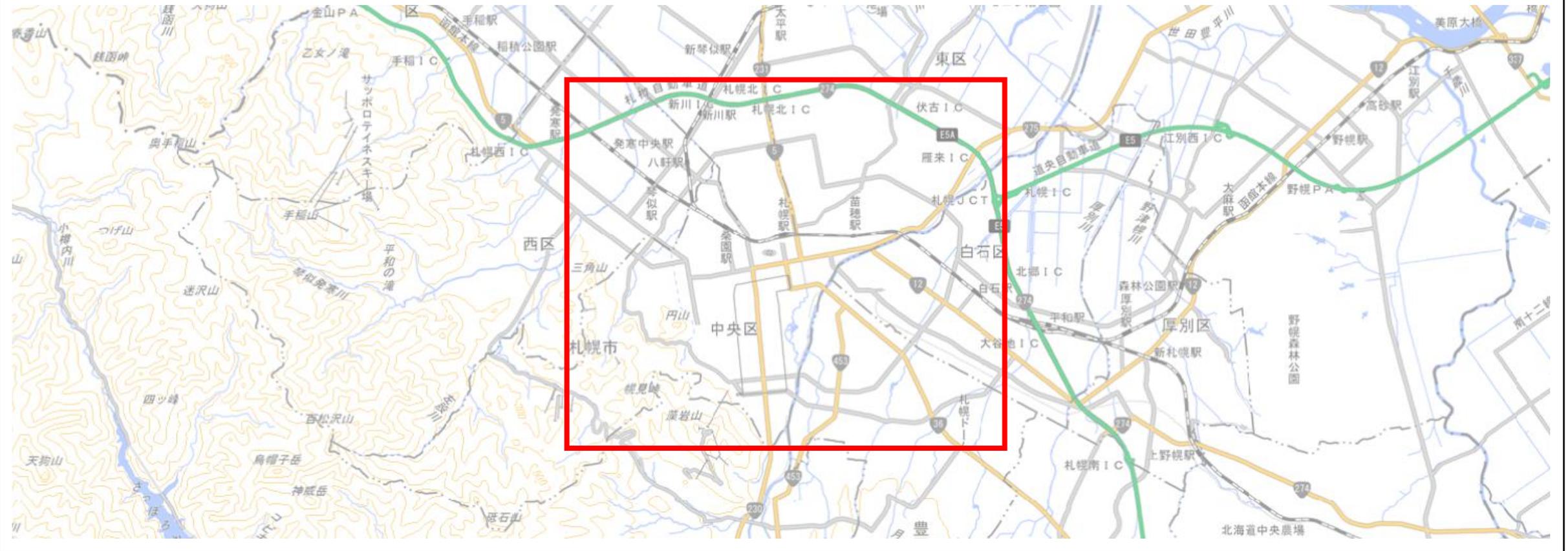
主体	役割
パスコ	<ul style="list-style-type: none"> <li>データ整備、仕様の決定</li> <li>教師データ作成</li> <li>AIモデル開発</li> <li>実証成果の可視化</li> <li>UC実証の評価、成果のとりまとめ</li> </ul>
産業技術総合研究所	<ul style="list-style-type: none"> <li>AIモデル開発の技術アドバイス</li> </ul>
アクセンチュア	<ul style="list-style-type: none"> <li>プロジェクトマネジメント</li> </ul>

実施体制図



# I. 実証概要 > 3. 実証エリア 実証エリア (1/8)

北海道 札幌市 札幌駅周辺 (100km<sup>2</sup>)

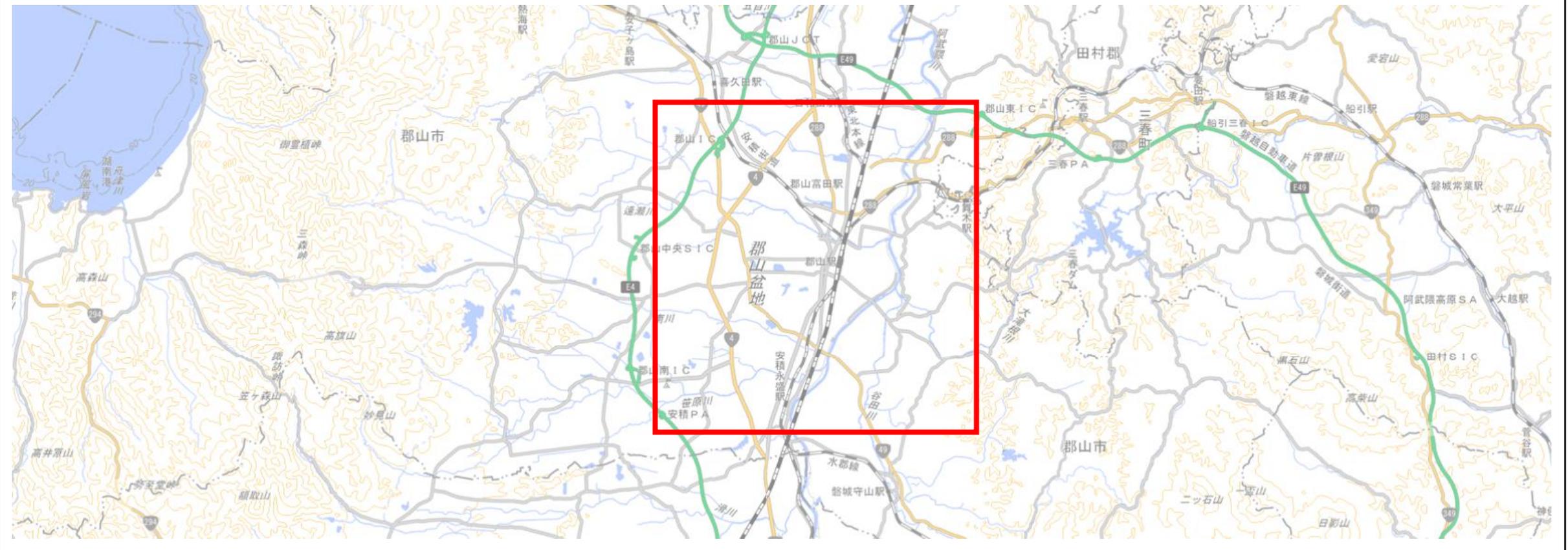


# I. 実証概要 > 3. 実証エリア 実証エリア (2/8)



PLATEAU  
by MLIT

福島県 郡山市 郡山駅周辺 (100km<sup>2</sup>)





# I. 実証概要 > 3. 実証エリア 実証エリア (4/8)

神奈川県 横浜市 (160km<sup>2</sup>)





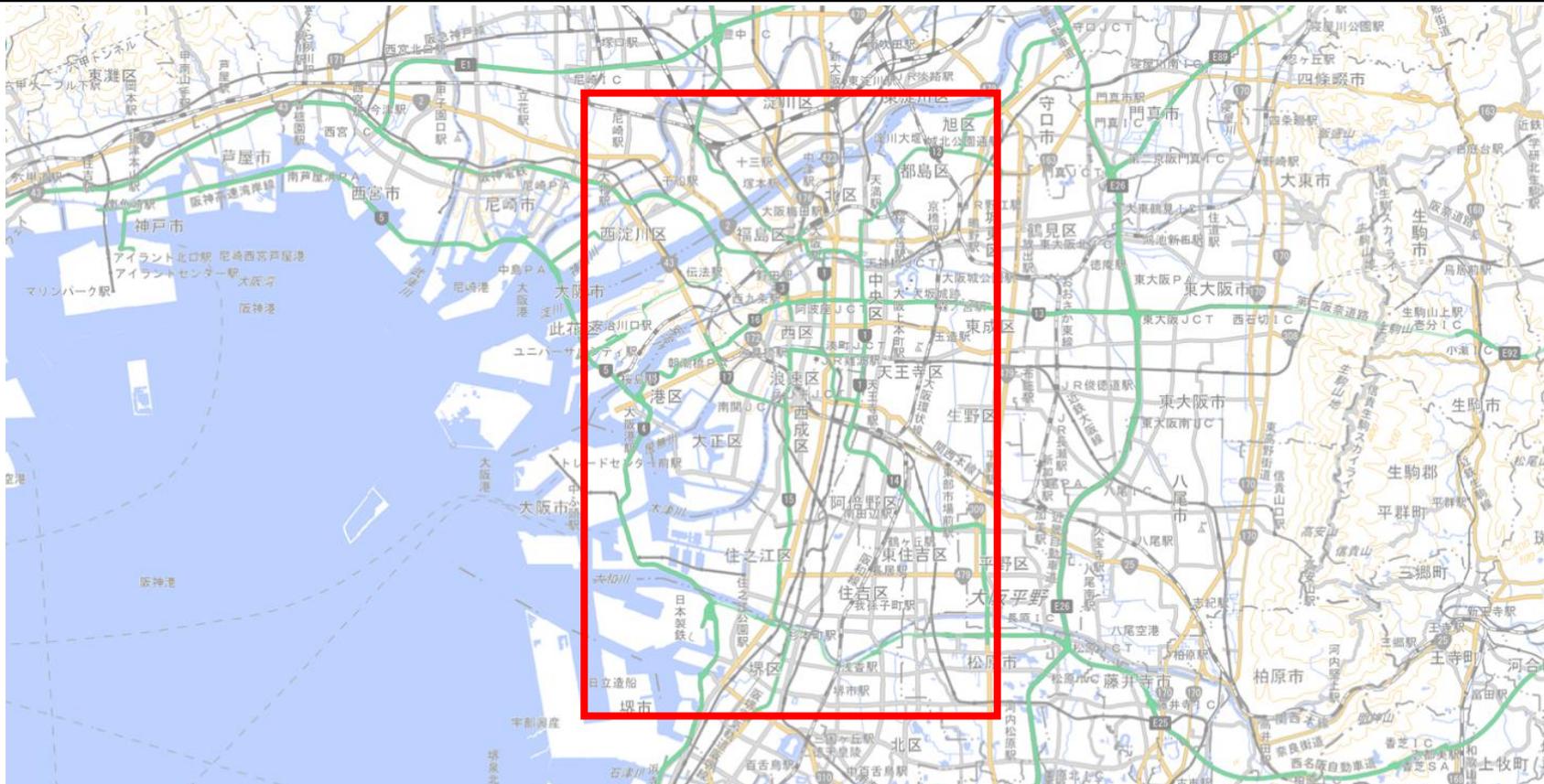
# I. 実証概要 > 3. 実証エリア 実証エリア (6/8)

静岡県 掛川市 (265km<sup>2</sup>)



# I. 実証概要 > 3. 実証エリア 実証エリア (7/8)

大阪府 大阪市 (176km<sup>2</sup>)

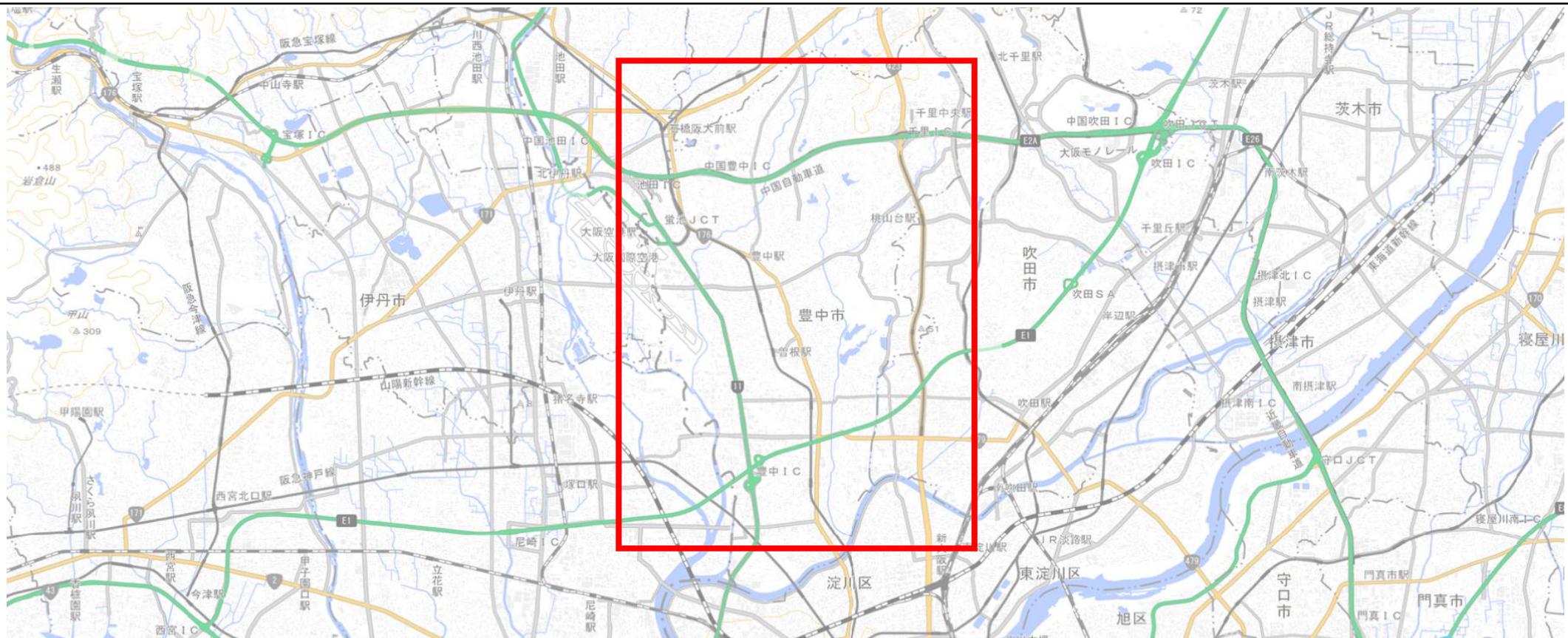


# I. 実証概要 > 3. 実証エリア 実証エリア (8/8)



PLATEAU  
by MLIT

大阪府 豊中市 (36km<sup>2</sup>)



# 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. ユースケース開発の検討	←→												
2. 実証計画の策定		←→											
3. データの収集			←→										
4. システム開発			←→										
5. AIのチューニング									←→				
6. 実証実験									←→				
7. 報告書作成・成果の公開										←→			

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

# 活用技術一覧

本実証に関連する技術は以下の通り

項目	内容
ALOS-3衛星	<ul style="list-style-type: none"> <li>2023年に運用予定であった国産光学衛星であり、直下視で観測幅70kmの広域を解像度80cmで連続撮影可能な衛星</li> </ul>
ASNARO-1衛星	<ul style="list-style-type: none"> <li>2014年に打上げた国産の小型光学衛星であり、直下視で10km×10kmシーンを解像度50cmで撮影可能な衛星</li> </ul>
AI（深層学習）	<ul style="list-style-type: none"> <li>現在の第3次人工知能ブームの中心的技術が深層学習であり、画像認識・テキスト解析・音声認識等の複数の領域で、研究開発・実務利用が進んでいる</li> </ul>
PyTorch	<ul style="list-style-type: none"> <li>深層学習モデル開発に利用される主要フレームワークの一つ。Facebookの人工知能研究開発グループによって開発された</li> </ul>
OpenCV	<ul style="list-style-type: none"> <li>画像処理や機械学習関連の主要なライブラリで、インテルが開発・公開している</li> </ul>
GDAL	<ul style="list-style-type: none"> <li>地理空間情報データ操作のライブラリで、ラスター・ベクター形式の変換等を実施</li> </ul>
HTTPX	<ul style="list-style-type: none"> <li>Python3用のHTTPクライアントライブラリで、非同期のHTTPリクエスト等を実施</li> </ul>
Rasterio	<ul style="list-style-type: none"> <li>地理空間情報画像操作のライブラリで、ラスターデータの読み書きを実施</li> </ul>
GeoPandas	<ul style="list-style-type: none"> <li>地理空間情報データ操作のライブラリで、テーブルデータ処理等を実施</li> </ul>
PySAL	<ul style="list-style-type: none"> <li>Python用の空間解析ライブラリで、階級区分図における分類等を実施</li> </ul>
ABCI	<ul style="list-style-type: none"> <li>国立研究開発法人 産業技術総合研究所が構築・運営するAI向け計算プラットフォーム</li> </ul>
QGIS	<ul style="list-style-type: none"> <li>豊富なファイル形式の空間情報の参照・加工・分析等が可能なGISフリーソフトウェア</li> </ul>
ERDAS IMAGINE	<ul style="list-style-type: none"> <li>衛星画像等のリモートセンシング画像の加工・分析が可能な主要ソフトウェア</li> </ul>
Tellus	<ul style="list-style-type: none"> <li>日本発のオープン&amp;フリーの衛星データプラットフォーム</li> </ul>

## II. 実証技術の概要 > 2. ALOS-3衛星

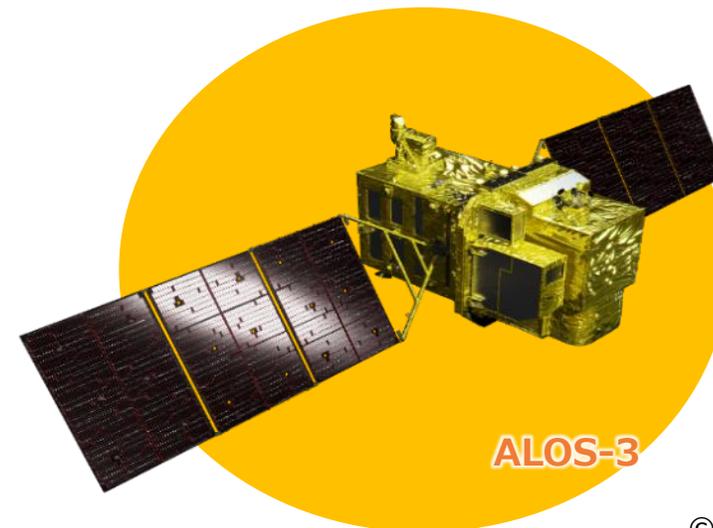
# ALOS-3衛星について

2023年に運用予定であった国産光学衛星であり、直下視で観測幅70kmの広域を解像度80cmで連続撮影可能な衛星

### 概要

項目	詳細
名称	ALOS-3衛星
概要	<ul style="list-style-type: none"> <li>2023年に運用予定だった国産の光学衛星</li> <li>観測幅70km、地上分解能80cm</li> <li>地上システムはパスコで運営</li> </ul>
主なプロダクト	<ul style="list-style-type: none"> <li>パンクロマチック 地上分解能：80cm</li> <li>マルチスペクトル 観測波長帯：6バンド 地上分解能：3.2m</li> <li>パンシャープ 地上分解能：80cm</li> </ul>
本ユースケースでの利用方法	<ul style="list-style-type: none"> <li>ASNARO-1衛星の画像を元に、ALOS-3衛星の撮影スペックに合わせた疑似画像を加工</li> </ul>

### ALOS-3衛星



© JAXA

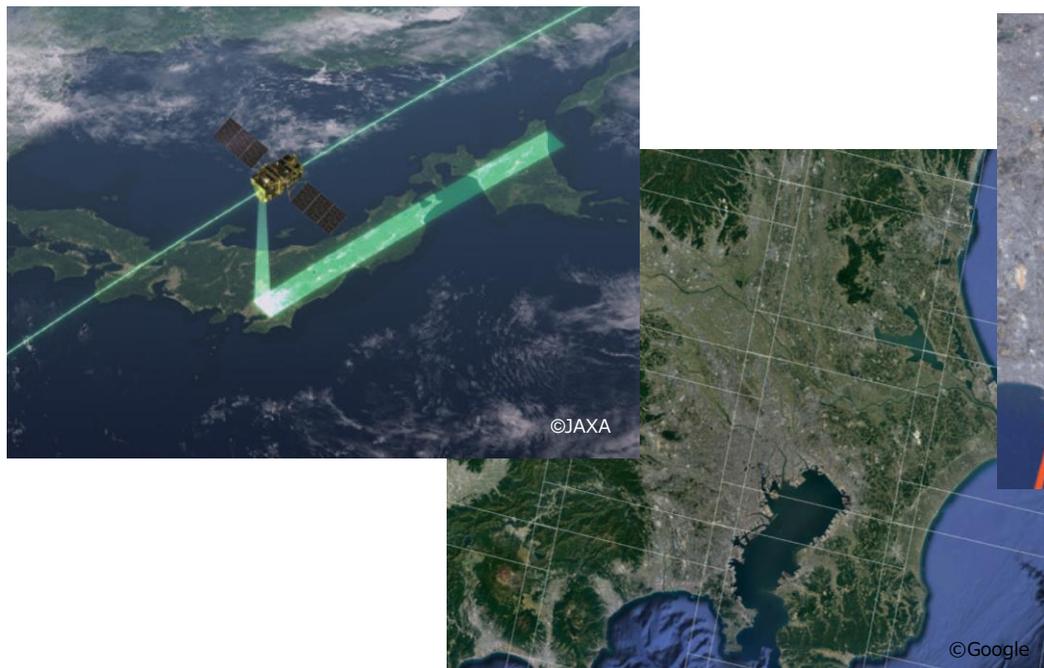
- 日本国内や途上国の高精度な地理空間情報の整備・更新に貢献するほか、多様な観測バンドによる沿岸域や植生域の環境保全への利用・研究など、様々な分野での活用が期待されている

## II. 実証技術の概要 > 2. ALOS-3衛星 ALOS-3衛星画像の特徴

ALOS-3は、歪みの少ない直下視で観測幅70kmの広域を解像度80cmで、全世界の観測と日本全国の年に複数回に渡る観測・衛星画像の提供を実現する

### ALOS-3の特長①：ベースマップとして全世界を観測

全球の高分解能画像をアーカイブしながら、日本全国を年間で複数回カバーすることで、モニタリングや変化抽出などの解析をより詳細化することが可能になる



### ALOS-3の特長②：広い視野と直下型の高い分解能

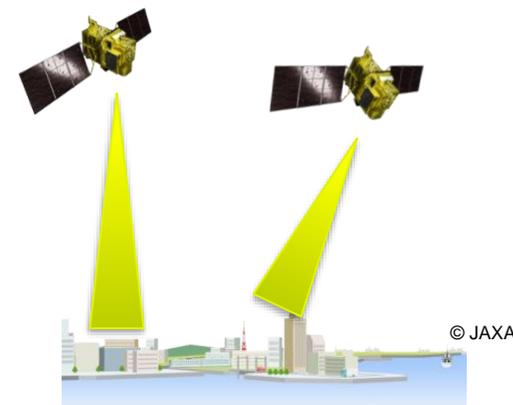
1ピクセルが80cmという衛星としては高い解像度でありながら、1シーンあたり70km×70kmという広範囲の撮影がされ、継ぎ接ぎの無い画像の活用が期待される。また、直下撮影であるため建物の傾きや地形の歪みが少ない画像となる



<歪みが少ない直下撮影>

直下視の場合

直下視でない場合



## II. 実証技術の概要 > 3. ASNARO-1衛星

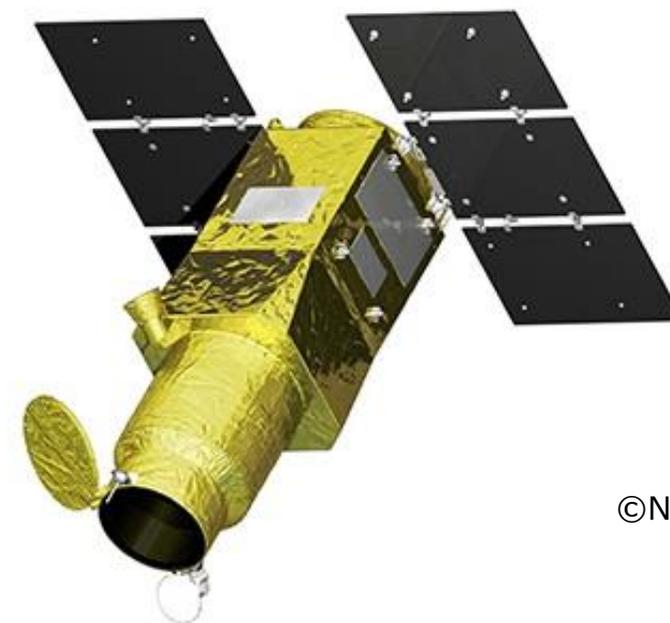
# ASNARO-1衛星について

2014年に打上げた国産の小型光学衛星であり、直下視で10km×10kmシーンを解像度50cmで撮影可能な衛星

### 概要

### ASNARO-1衛星

項目	詳細
名称	ASNARO-1衛星
概要	<ul style="list-style-type: none"> <li>• 2014年に打上げた国産の小型光学衛星</li> <li>• 観測幅：10km</li> <li>• 分解能：50cm</li> <li>• 地上システムはパスコで運営中</li> </ul>
主なプロダクト	<ul style="list-style-type: none"> <li>• パンクロマチック 地上分解能：50cm</li> <li>• マルチスペクトル 観測波長帯：4バンド 地上分解能：2.0m</li> <li>• パンシャープ 地上分解能：50cm</li> </ul>
本ユースケースでの利用方法	<ul style="list-style-type: none"> <li>• ASNARO-1衛星を元に、ALOS-3衛星シミュレーション画像を加工</li> </ul>



©NEC

## II. 実証技術の概要 > 4. AI (深層学習)

# AI (深層学習) について

現在の第3次人工知能ブームの中心的技術が深層学習であり、画像認識・テキスト解析・音声認識等の複数の領域で、研究開発・実務利用が進んでいる

### 概要

項目	詳細
名称	AI (深層学習)
概要	<ul style="list-style-type: none"> <li>人間の脳神経細胞のニューラルネットワークを再現した機械学習手法の一つ               <ul style="list-style-type: none"> <li>画像認識・テキスト解析・音声認識等の複数の領域で、研究開発・実用化が急速に進んでいる</li> </ul> </li> </ul>
主なプロダクト	<ul style="list-style-type: none"> <li>AIモデルを開発する上で、一般に利用される主要フレームワークは、以下のものがある               <ul style="list-style-type: none"> <li>TensorFlow</li> <li>PyTorch</li> <li>scikit-learn</li> </ul> </li> </ul>
本ユースケースでの利用方法	<ul style="list-style-type: none"> <li>AIモデルの開発 (要素技術)</li> </ul>

### AI (深層学習) を利用した画像認識の特徴



- 画像認識の分野において、深層学習は従来の画像処理方法に比べて、精度が高くなることが知られている
- AIモデル (深層学習) では、画素単位の判読ではなく、判読対象画素の周辺情報の質感・テクスチャ情報も利用することで、局所的な画素の異常値にも強く、従来手法よりも高い精度が期待できるためである

# II. 実証技術の概要 > 5. PyTorch

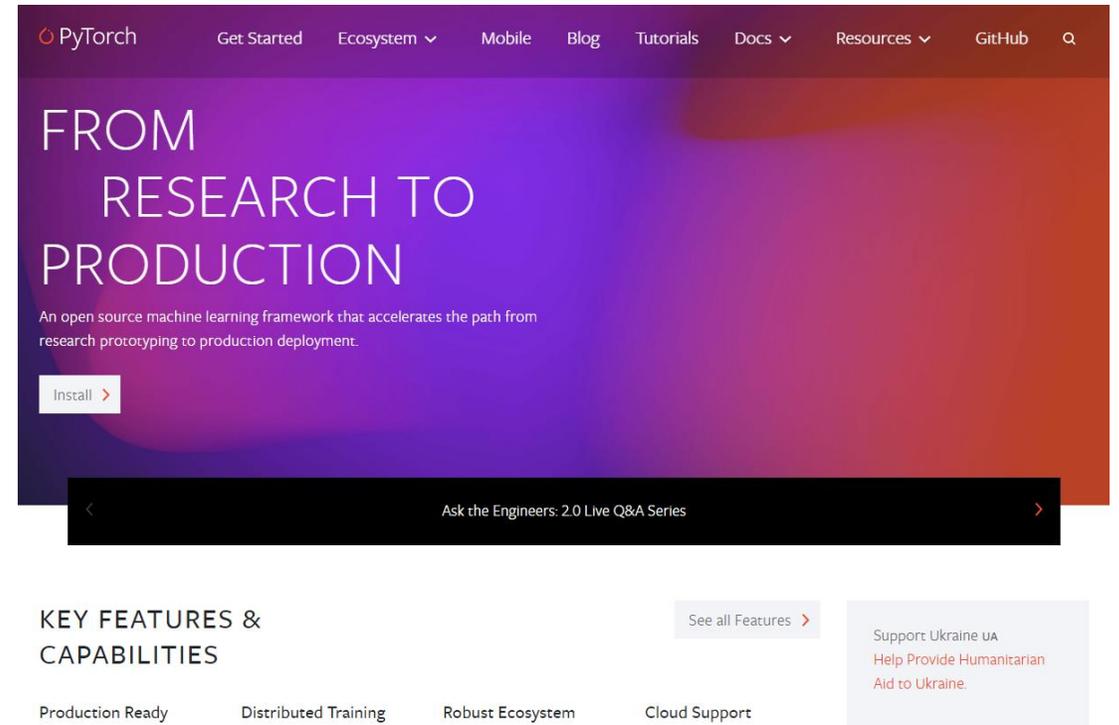
## PyTorchについて

深層学習モデル開発に利用される主要フレームワークの一つ。Facebookの人工知能研究開発グループによって開発された

### 概要

項目	詳細
名称	PyTorch (パイトーチ)
概要	<ul style="list-style-type: none"><li>深層学習モデル開発に利用される主要フレームワークの一つであり、Facebookの人工知能研究開発グループによって開発された</li><li>人工知能の研究者の利用も多く、論文等がGitHubで公開されることもあるので、最新の手法を入手しやすい</li></ul>
主な機能	<ul style="list-style-type: none"><li>AIモデルの開発フレームワーク</li></ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"><li>AIモデルの開発フレームワーク</li></ul>

### PyTorch公式HP



## II. 実証技術の概要 > 6. OpenCV

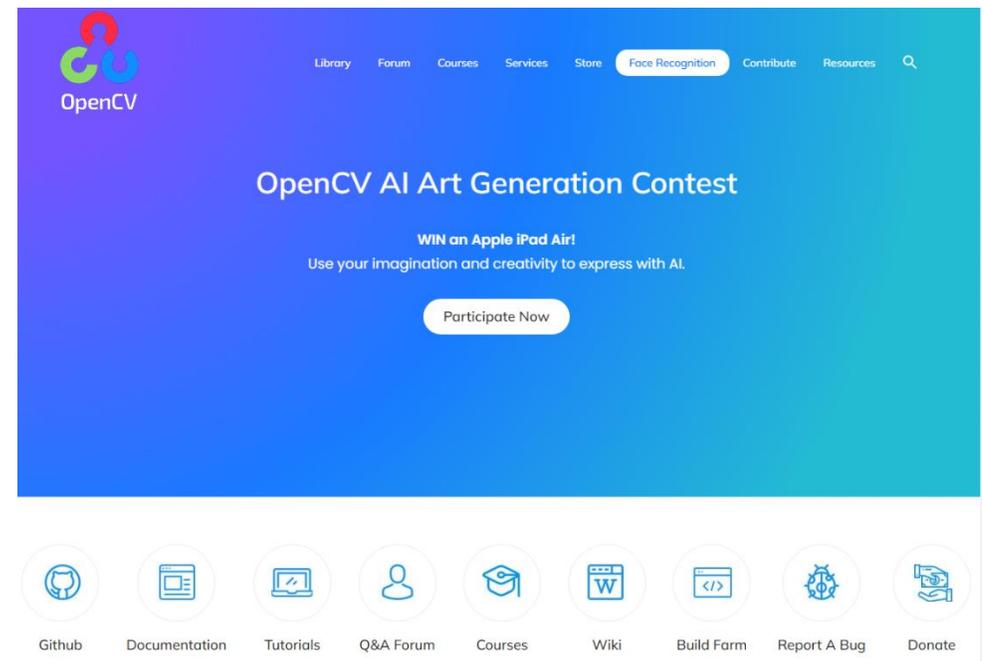
# OpenCVについて

画像処理や機械学習関連の主要なライブラリで、インテルが開発・公開している

### 概要

項目	詳細
名称	OpenCV (オープンシーヴィ)
概要	<ul style="list-style-type: none"><li>画像処理や深層学習関連の主要なライブラリの一つであり、インテルによって開発・公開された</li><li>画像処理、機械学習のライブラリ以外にも構造解析、モーション解析、物体追跡等の多くのライブラリがあり、通常は必要なものだけを個別のモジュールとして利用する</li></ul>
主な機能	<ul style="list-style-type: none"><li>画像処理や機械学習のライブラリ</li></ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"><li>AIモデルに入力するための画像加工</li></ul>

### OpenCV公式HP



# II. 実証技術の概要 > 7. GDAL

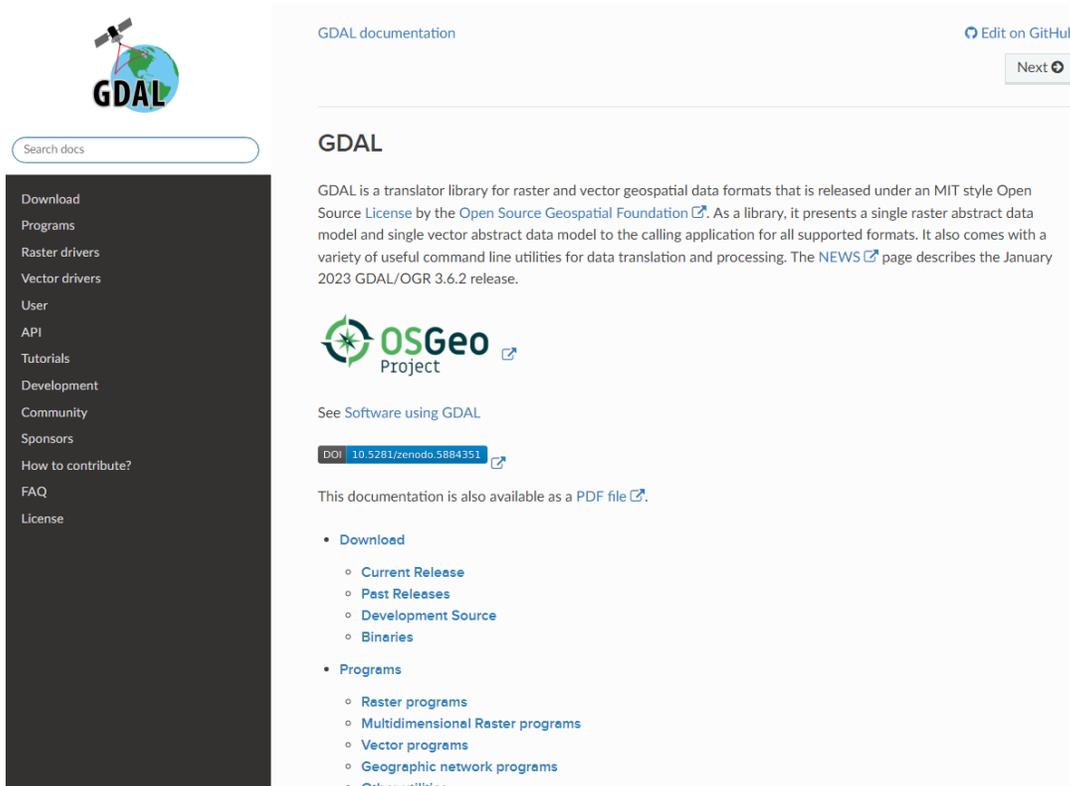
## GDALについて

地理空間情報データ操作のライブラリで、ラスター・ベクター形式の変換等を実施

### 概要

項目	詳細
名称	GDAL (ジーダル)
概要	<ul style="list-style-type: none"><li>OSGeo財団により提供されている、地理空間情報データ操作のライブラリ</li><li>ラスター・ベクター形式のフォーマット変換や地理情報付与等が可能</li></ul>
主な機能	<ul style="list-style-type: none"><li>地理空間情報データ操作のライブラリ</li></ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"><li>変化検出結果に地理情報を付与</li></ul>

### GDAL公式HP



The screenshot shows the GDAL documentation website. On the left is a dark sidebar with navigation links: Download, Programs, Raster drivers, Vector drivers, User, API, Tutorials, Development, Community, Sponsors, How to contribute?, FAQ, License. The main content area has a search bar, a 'GDAL' heading, and a paragraph describing GDAL as a translator library for raster and vector geospatial data formats. Below this is the OSGeo Project logo and a section 'See Software using GDAL' with a DOI link. At the bottom, there are lists for 'Download' (Current Release, Past Releases, Development Source, Binaries) and 'Programs' (Raster programs, Multidimensional Raster programs, Vector programs, Geographic network programs, Other utilities).

# II. 実証技術の概要 > 8. HTTPX

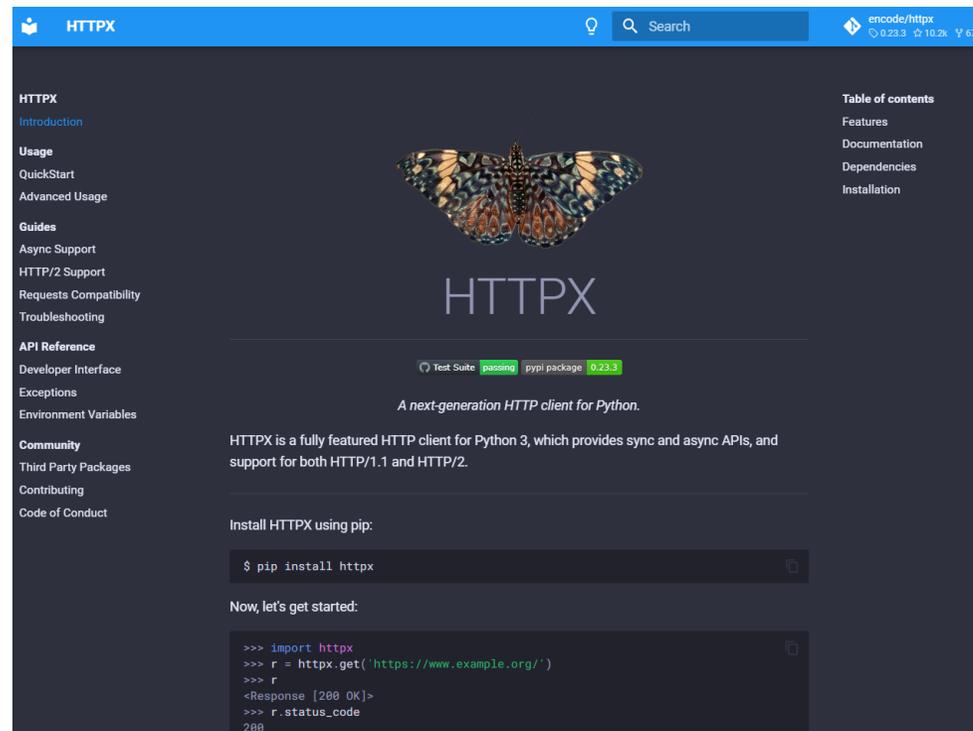
## HTTPXについて

Python3用のHTTPクライアントライブラリで、非同期のHTTPリクエスト等を実施

### 概要

項目	詳細
名称	HTTPX (エイチティーティーピーエックス)
概要	<ul style="list-style-type: none"> <li>Python3向けのHTTPクライアントライブラリ</li> </ul>
主な機能	<ul style="list-style-type: none"> <li>HTTP/1.1およびHTTP/2に対応した通信</li> <li>同期および非同期のHTTPリクエスト</li> </ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"> <li>新旧時期の画像のダウンロード</li> </ul>

### HTTPX公式HP



The screenshot shows the official HTTPX website. The page features a dark theme with a blue header. The main content area includes a large butterfly illustration and the text "HTTPX". Below this, it states "A next-generation HTTP client for Python." and provides information about the package version (0.23.3) and its status (passing tests). The page also includes a table of contents on the right and a list of navigation links on the left.

# II. 実証技術の概要 > 9. Rasterio

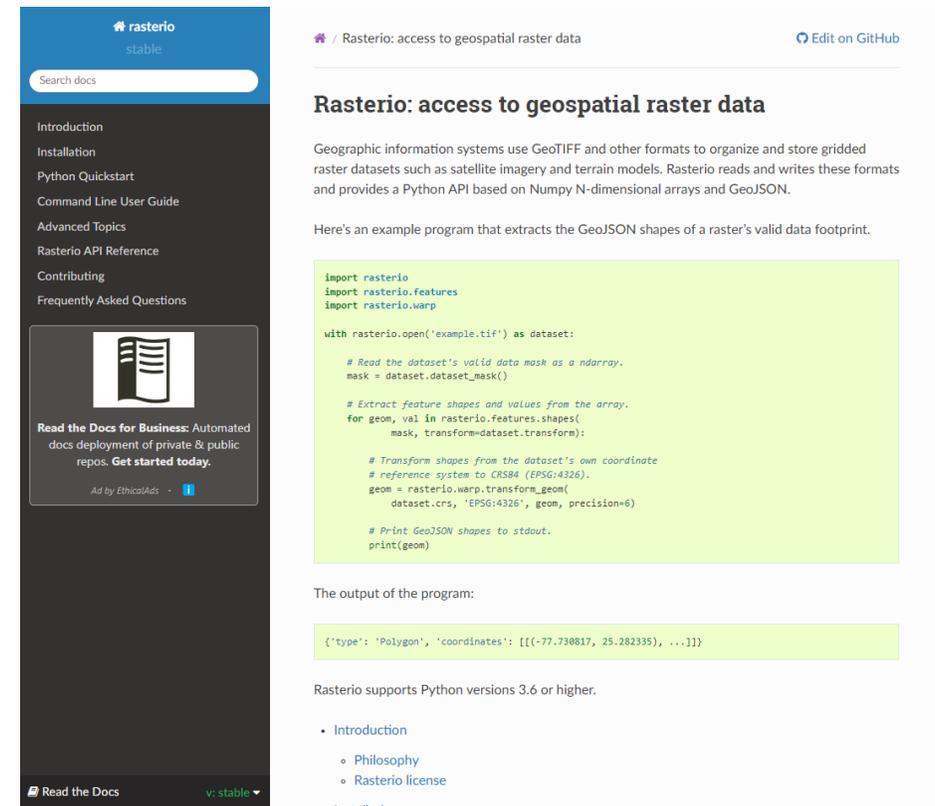
## Rasterioについて

地理空間情報画像操作のライブラリで、ラスターデータの読み書き等を実施

### 概要

項目	詳細
名称	Rasterio (ラステリオ)
概要	<ul style="list-style-type: none"><li>地理空間情報の画像データの読み書きをサポートするライブラリ</li></ul>
主な機能	<ul style="list-style-type: none"><li>地理空間情報の画像データの読み書き</li></ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"><li>変化検出結果をGISデータに変換</li></ul>

### Rasterio公式HP



The screenshot shows the Rasterio official website. The top navigation bar includes 'rasterio stable' and a search box. The main content area lists various sections: Introduction, Installation, Python Quickstart, Command Line User Guide, Advanced Topics, Rasterio API Reference, Contributing, and Frequently Asked Questions. A prominent banner reads 'Read the Docs for Business: Automated docs deployment of private & public repos. Get started today.' Below this, there is a code example for reading a raster dataset and extracting its footprint. The code is as follows:

```
import rasterio
import rasterio.features
import rasterio.warp

with rasterio.open('example.tif') as dataset:
    # Read the dataset's valid data mask as a ndarray.
    mask = dataset.dataset_mask()

    # Extract feature shapes and values from the array.
    for geom, val in rasterio.features.shapes(
        mask, transform=dataset.transform):

        # Transform shapes from the dataset's own coordinate
        # reference system to CRS84 (EPSG:4326).
        geom = rasterio.warp.transform_geom(
            dataset.crs, 'EPSG:4326', geom, precision=6)

        # Print GeoJSON shapes to stdout.
        print(geom)
```

The output of the program is shown as:

```
{'type': 'Polygon', 'coordinates': [[(-77.730817, 25.282335), ...]]}
```

Below the code, it states 'Rasterio supports Python versions 3.6 or higher.' and provides a table of contents with links to 'Introduction', 'Philosophy', and 'Rasterio license'.

# II. 実証技術の概要 > 10. GeoPandas

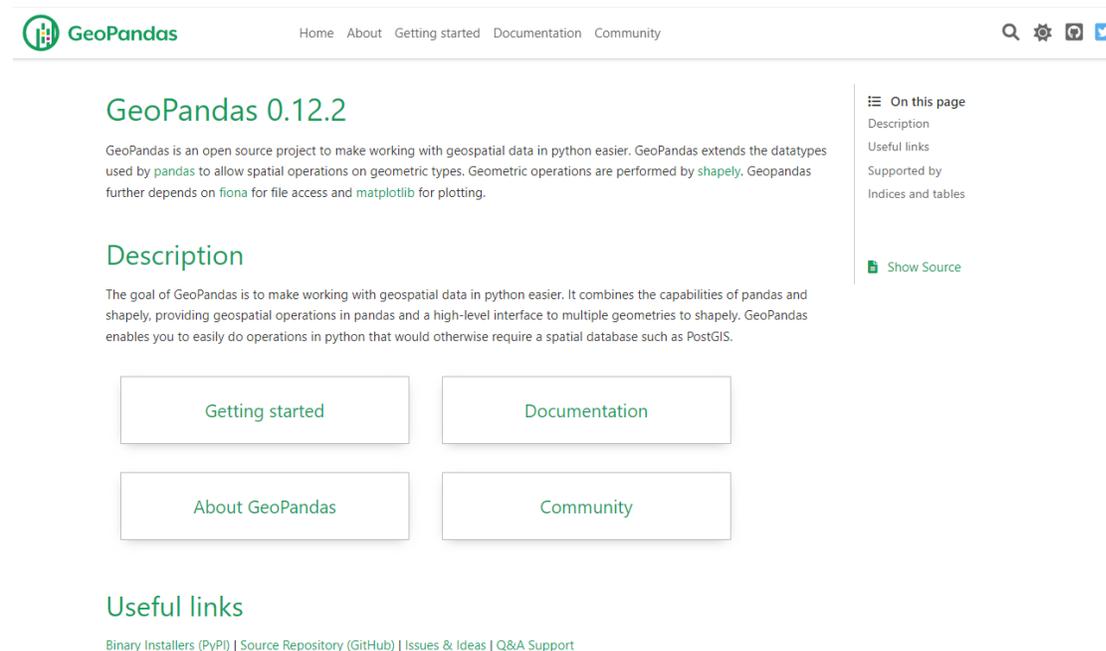
## GeoPandasについて

地理空間情報データ操作のライブラリで、テーブルデータ処理等を実施

### 概要

項目	詳細
名称	GeoPandas (ジオパンドス)
概要	<ul style="list-style-type: none"> <li>ベクター形式の地理空間情報データ操作ライブラリの一つ</li> <li>テーブルデータの処理ライブラリである pandas と親和性が高い</li> </ul>
主な機能	<ul style="list-style-type: none"> <li>地理空間情報データ操作</li> </ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"> <li>GISデータを用いてメッシュデータを生成</li> </ul>

### GeoPandas公式HP



The screenshot shows the official website for GeoPandas. The page title is "GeoPandas 0.12.2". The main content area includes a "Description" section, which states: "The goal of GeoPandas is to make working with geospatial data in python easier. It combines the capabilities of pandas and shapely, providing geospatial operations in pandas and a high-level interface to multiple geometries to shapely. GeoPandas enables you to easily do operations in python that would otherwise require a spatial database such as PostGIS." Below the description are four buttons: "Getting started", "Documentation", "About GeoPandas", and "Community". On the right side, there is a sidebar with a table of contents for the page, including "On this page", "Description", "Useful links", "Supported by", and "Indices and tables". At the bottom of the page, there are "Useful links" for "Binary Installers (PyPI)", "Source Repository (GitHub)", "Issues & Ideas", and "Q&A Support".

# II. 実証技術の概要 > 11. PySAL

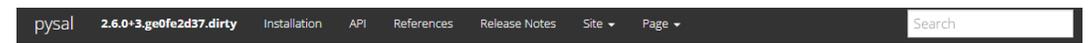
## PySALについて

Python用の空間解析ライブラリで、階級区分図における分類等を実施

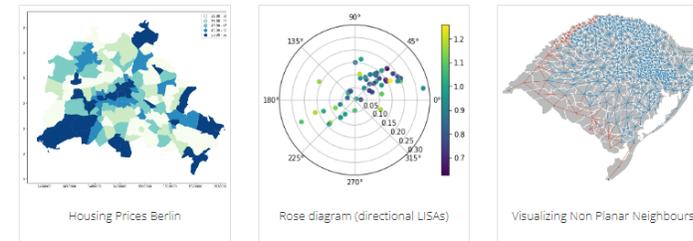
### 概要

項目	詳細
名称	PySAL (パイサル)
概要	<ul style="list-style-type: none"> <li>Python用の空間解析ライブラリ</li> </ul>
主な機能	<ul style="list-style-type: none"> <li>地理空間情報データを用いた統計的な空間解析</li> <li>階級区分図の分類</li> <li>ネットワークの分析</li> </ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"> <li>変化率を用いた更新優先度の決定 (自然分類)</li> </ul>

### PySAL公式HP



#### PySAL: Python Spatial Analysis Library



PySAL is an open source cross-platform library for geospatial data science with an emphasis on geospatial vector data written in Python. It supports the development of high level applications for spatial analysis, such as

- detection of spatial clusters, hot-spots, and outliers
- construction of graphs from spatial data
- spatial regression and statistical modeling on geographically embedded networks
- spatial econometrics
- exploratory spatio-temporal data analysis

#### PySAL Components

- **explore** - modules to conduct exploratory analysis of spatial and spatio-temporal data, including statistical testing on points, networks, and polygonal lattices. Also includes methods for spatial inequality, distributional dynamics, and segregation.
- **viz** - visualize patterns in spatial data to detect clusters, outliers, and hot-spots.
- **model** - model spatial relationships in data with a variety of linear, generalized-linear, generalized-additive, and nonlinear models.
- **lib** - solve a wide variety of computational geometry problems:
  - graph construction from polygonal lattices, lines, and points.
  - construction and interactive editing of spatial weights matrices & graphs
  - computation of alpha shapes, spatial indices, and spatial-topological relationships
  - reading and writing of sparse graph data, as well as pure python readers of spatial vector data.

Details are available in the [PySAL api](#).

For background information see [\[RA07\]](#).

# II. 実証技術の概要 > 12. ABCI

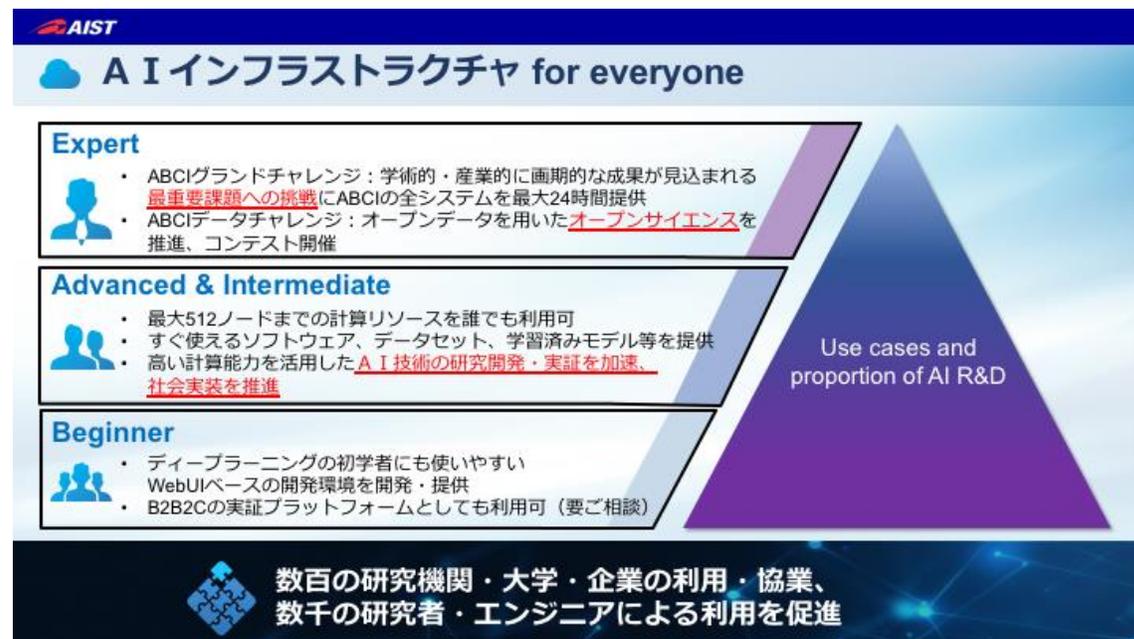
## ABCIについて

国立研究開発法人 産業技術総合研究所が構築・運営するAI向け計算プラットフォーム

### 概要

項目	詳細
名称	ABCI (エービーシーアイ)
概要	<ul style="list-style-type: none"> <li>国立研究開発法人 産業技術総合研究所が構築・運営をしている世界最大規模の人工知能処理向け計算プラットフォーム</li> <li>正式名称はAI橋渡しクラウド (AI Bridging Cloud Infrastructure)</li> </ul>
主な機能	<ul style="list-style-type: none"> <li>AIモデルの開発インフラ</li> </ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"> <li>AIモデルの開発インフラ</li> </ul>

### 産総研による説明資料抜粋



**AIST AI インフラストラクチャ for everyone**

**Expert**

- ABCIブランドチャレンジ: 学術的・産業的に画期的な成果が見込まれる **最重要課題への挑戦**にABCIの全システムを最大24時間提供
- ABCIデータチャレンジ: オープンデータを用いた **オープンサイエンス**を推進、コンテスト開催

**Advanced & Intermediate**

- 最大512ノードまでの計算リソースを誰でも利用可
- すぐ使えるソフトウェア、データセット、学習済みモデル等を提供
- 高い計算能力を活用した **AI技術の研究開発・実証を加速、社会実装を推進**

**Beginner**

- ディープラーニングの初学者にも使いやすい WebUIベースの開発環境を開発・提供
- B2B2Cの実証プラットフォームとしても利用可 (要ご相談)

Use cases and proportion of AI R&D

数百の研究機関・大学・企業の利用・協業、数千の研究者・エンジニアによる利用を促進

# II. 実証技術の概要 > 13. QGIS

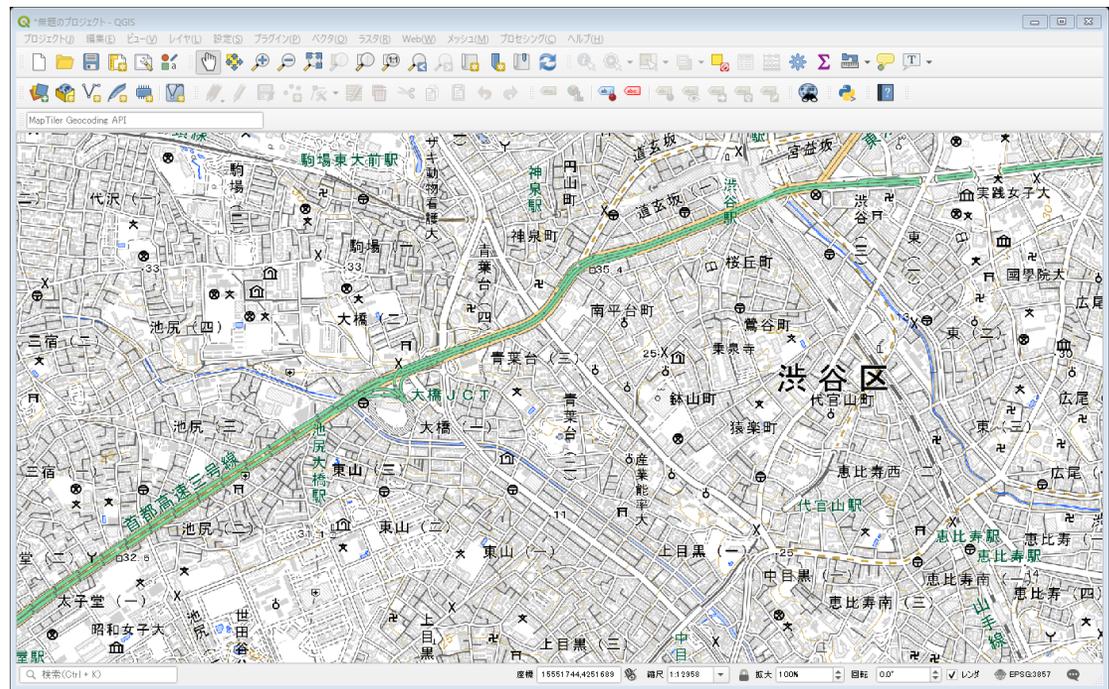
## QGISについて

豊富なファイル形式の空間情報の参照・加工・分析等が可能なGISフリーソフトウェア

### 概要

### QGIS画面イメージ

項目	詳細
名称	QGIS (キュージーアイエス)
概要	<ul style="list-style-type: none"> <li>● 主要なGISフリーソフトウェア</li> </ul>
主な機能	<ul style="list-style-type: none"> <li>● GISデータの閲覧</li> <li>● 形状・属性等の加工編集</li> <li>● 形状・属性等を利用した空間解析</li> <li>● GIS関連ファイルの入出力</li> </ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"> <li>● 教師データの作成</li> <li>● 画像や成果物の閲覧</li> </ul>



# II. 実証技術の概要 > 14. ERDAS IMAGINE

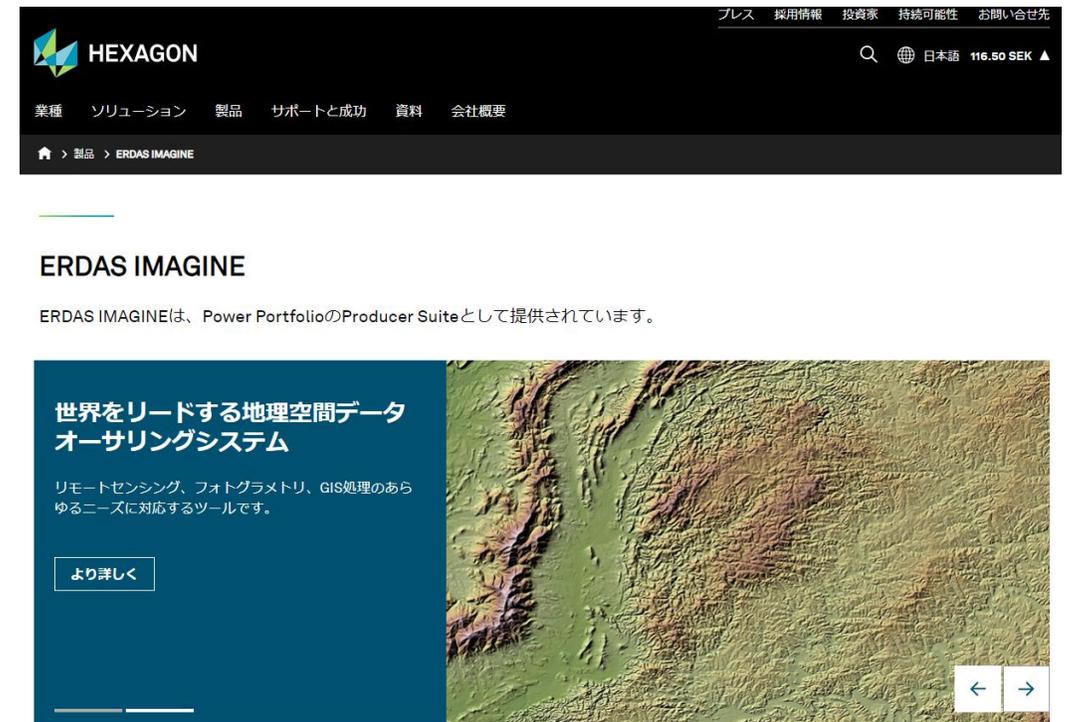
## ERDAS IMAGINEについて

衛星画像等のリモートセンシング画像の加工・分析が可能な主要ソフトウェア

### 概要

項目	詳細
名称	ERDAS IMAGINE (ア-ダスイマジン)
概要	<ul style="list-style-type: none"> <li>衛星画像等のリモートセンシング画像の加工・分析が可能な主要ソフトウェア</li> </ul>
主な機能	<ul style="list-style-type: none"> <li>リモートセンシングデータの閲覧</li> <li>幾何補正・属性等の加工編集</li> <li>バンド間演算等を利用した分析</li> <li>GIS関連・衛星関連ファイルの入出力</li> </ul>
本ユースケースで利用する機能	<ul style="list-style-type: none"> <li>ALOS-3衛星シミュレーション画像の作成</li> </ul>

### ERDAS IMAGINE公式HP



## II. 実証技術の概要 > 15. Tellus Tellusについて

日本発のオープン&フリーの衛星データプラットフォーム

### 概要

項目	詳細
名称	Tellus (テルース)
概要	<ul style="list-style-type: none"> <li>日本発のオープン&amp;フリーの衛星プラットフォーム</li> <li>Tellus Marketを提供しており、衛星画像、API、アドイン等のマーケット環境を提供している（有償・無償）</li> </ul>
主なプロダクト	<ul style="list-style-type: none"> <li>Tellusマーケット</li> </ul>
本ユースケースでの利用方法	<ul style="list-style-type: none"> <li>今後、本実証で開発したAPIプログラム等を活用して、Tellus上でサービス化を実施予定</li> </ul>

### Tellus公式サイト



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

実証実験では対象都市の3D都市モデル・航空写真・ALOS-3衛星シミュレーション画像を準備し、AIモデルを試作・精度検証した上で、更新優先度マップの有用性を自治体・民間企業にヒアリングする

データ収集・  
シミュレーション画像  
加工

- 新旧2時期の画像から変化箇所を抽出する上で、データ収集が可能である対象エリアを選定する
- 旧時期画像として、PLATEAU搭載済の航空写真を収集し、3D都市モデルデータを収集・加工する
- 新时期画像として、旧時期画像より撮影時期の新しい航空写真やASNARO-1衛星画像を収集し、解像度80cmのALOS-3シミュレーション画像として加工する

AI教師データの整備

- 更新優先度マップのAIモデル開発に必要な教師データセットを作成する
- 収集・加工して整備済の新旧時期のペア画像を対象に作成する
- 建物の変化等を目視で確認しながら、変化箇所（新築・滅失・建替）の建物形状を作成する

AIモデル開発

- 深層学習技術を取入れて、入力画像の取得条件に合わせたAIモデルを開発し、更新優先度マップ（メッシュデータ）を出力するプログラムを開発する

更新優先度マップ  
精度検証

- 検証用に収集・加工・作成された新旧時期の画像データ・更新優先度マップをPLATEAU VIEW上で閲覧可能にする
- 教師データの一部エリアを用いて、更新優先度マップの精度を検証する

自治体・民間企業  
ヒアリング

- 更新優先度マップの有用性について、ユーザーヒアリングを実施する
  - 自治体には、都市計画基本図等の整備において部分更新の実現性についての意見を調査する
  - 民間企業には、自社整備データの更新作業の効率化や更新地域の優先付け等について意見を調査する



## Ⅲ. 実証システム > 2. 想定事業機会 想定事業機会

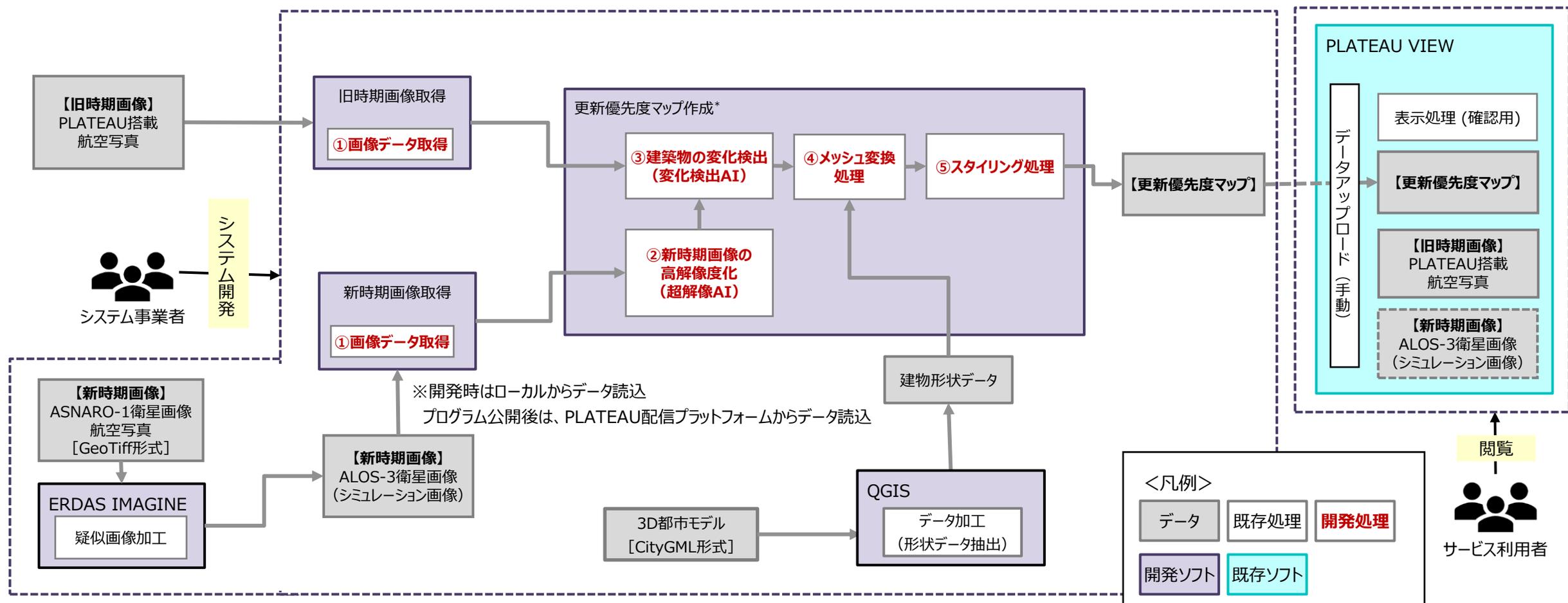
3D都市モデルや地図関連データの作成・更新・提供を担う民間事業者や自治体が想定利用者となり、ALOS-3衛星画像を含めた変化検出サービスとしての提供が事業機会として見込まれる

項目	内容
利用事業者	<ul style="list-style-type: none"><li>• 自治体</li><li>• 民間事業者（地図関連サービス会社）</li></ul>
提供価値	<ul style="list-style-type: none"><li>• 3D都市モデルや地図関連データの更新候補箇所の抽出</li><li>• 地図関連データの更新に必要な航空写真アーカイブ等の調達効率化</li><li>• PLATEAU（航空写真・3D都市モデル）のリソース活用</li><li>• 災害時における広域な変化箇所の迅速な把握</li></ul>
サービス仮説	<ul style="list-style-type: none"><li>• サブスクリプションサービス<ul style="list-style-type: none"><li>- PCやタブレットから閲覧可能なクラウドサービスとして、ALOS-3衛星画像と合わせたサブスクリプション型のマネタイズ</li></ul></li><li>• 受託型サービス<ul style="list-style-type: none"><li>- ALOS-3衛星画像の選定、変化検出、成果物までを利用者の個別ニーズに合わせて実施する受託型のマネタイズ</li></ul></li></ul>

# 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

新旧2時期の画像を変化検出のための入力データとして使用し、更新優先度マップを作成する

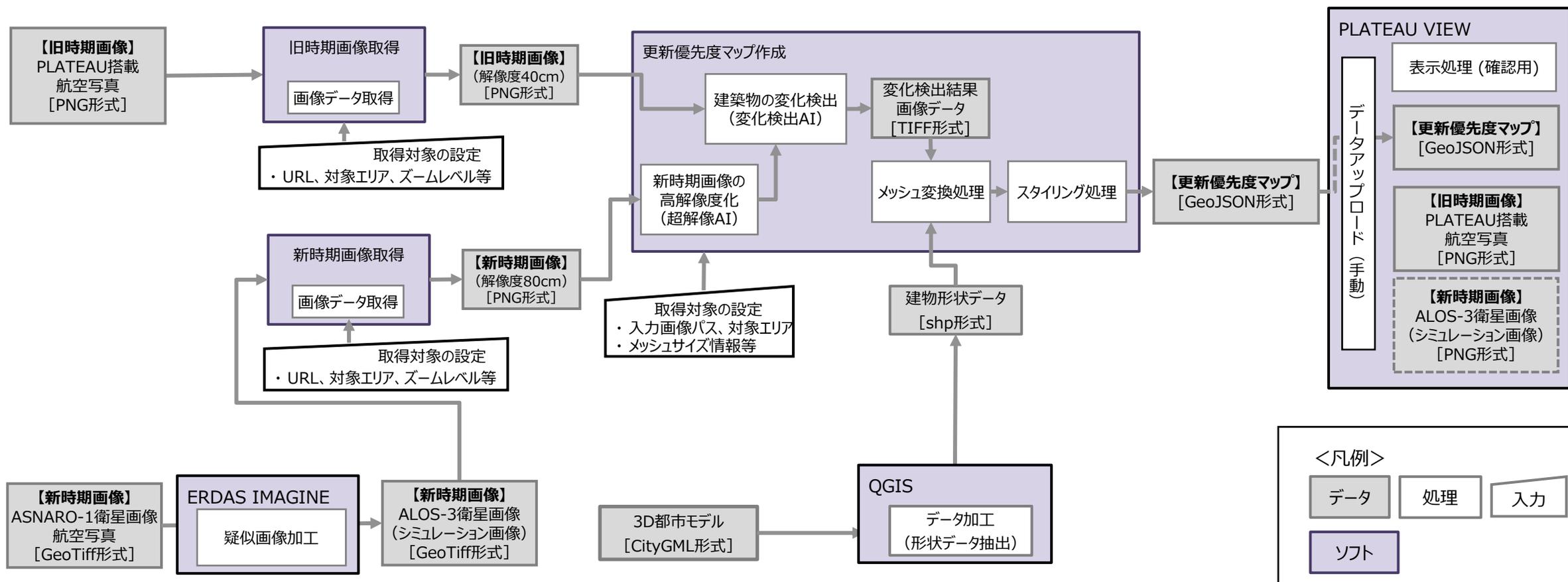


\* 計算処理用サーバ (データの読込処理・更新優先度マップの生成等) のスペック: (仮想CPU: 4コア、メモリ: 4 GB、HDD: SSD 1TB)



# 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

新旧2時期の画像を変化検出のための入力データとして使用し、更新優先度マップを作成する



# Ⅲ. 実証システム > 4. システム機能

## システム機能一覧

システム機能は以下のとおり

赤太字：開発部分

機能名	内容	本番環境	開発環境
①画像データ取得	<ul style="list-style-type: none"> <li>PLATEAUのストリーミング配信から旧時期の航空写真を取得</li> <li>ローカル環境からALOS-3衛星シミュレーション画像を取得*</li> </ul>	○	×
②新時期画像の高解像度化 (超解像AI)	<ul style="list-style-type: none"> <li>新時期のALOS-3シミュレーション画像に対して、旧時期の航空写真の解像度と合わせるために超解像処理を実施</li> </ul>	○	○
③建築物の変化検出 (変化検出AI)	<ul style="list-style-type: none"> <li>旧時期の航空写真と超解像処理で高解像度化した新時期のALOS-3シミュレーション画像を比較して、建築物の変化を検出</li> </ul>	○	○
④メッシュ変換処理	<ul style="list-style-type: none"> <li>建築物の検出結果をメッシュデータに変換</li> <li>メッシュごとに変化率を算出し、更新優先度を決定</li> </ul>	○	×
⑤スタイリング処理	<ul style="list-style-type: none"> <li>メッシュデータに対して、更新優先度に応じた色付けを実施</li> </ul>	○	×
疑似画像加工	<ul style="list-style-type: none"> <li>航空写真および衛星画像からALOS-3シミュレーション画像を作成</li> </ul>	×	○
データ加工（形状データ抽出）	<ul style="list-style-type: none"> <li>Project PLATEAUにて公開されているCityGML形式の建築物データから家屋の形状（LOD0）を抽出し、GISで読み込める形式で保存</li> </ul>	×	○

\* プログラム公開後は、PLATEAUのストリーミング配信からALOS-3衛星シミュレーション画像を取得

# Ⅲ. 実証システム > 4. システム機能

## ① 画像データ取得

機能仕様

項目	詳細
機能名	画像データ取得
機能概要	地図タイル画像 (XYZ Tiles) 形式で配信されているデータをダウンロードし保存
入力データ仕様	3次メッシュコードが記載されたテキストファイル (文字コードはUTF-8推奨) 3次メッシュポリゴン (shp形式)
出力データ仕様	3次メッシュの範囲に対応するPNG形式の画像
利用するアルゴリズム	-

利用ライブラリ

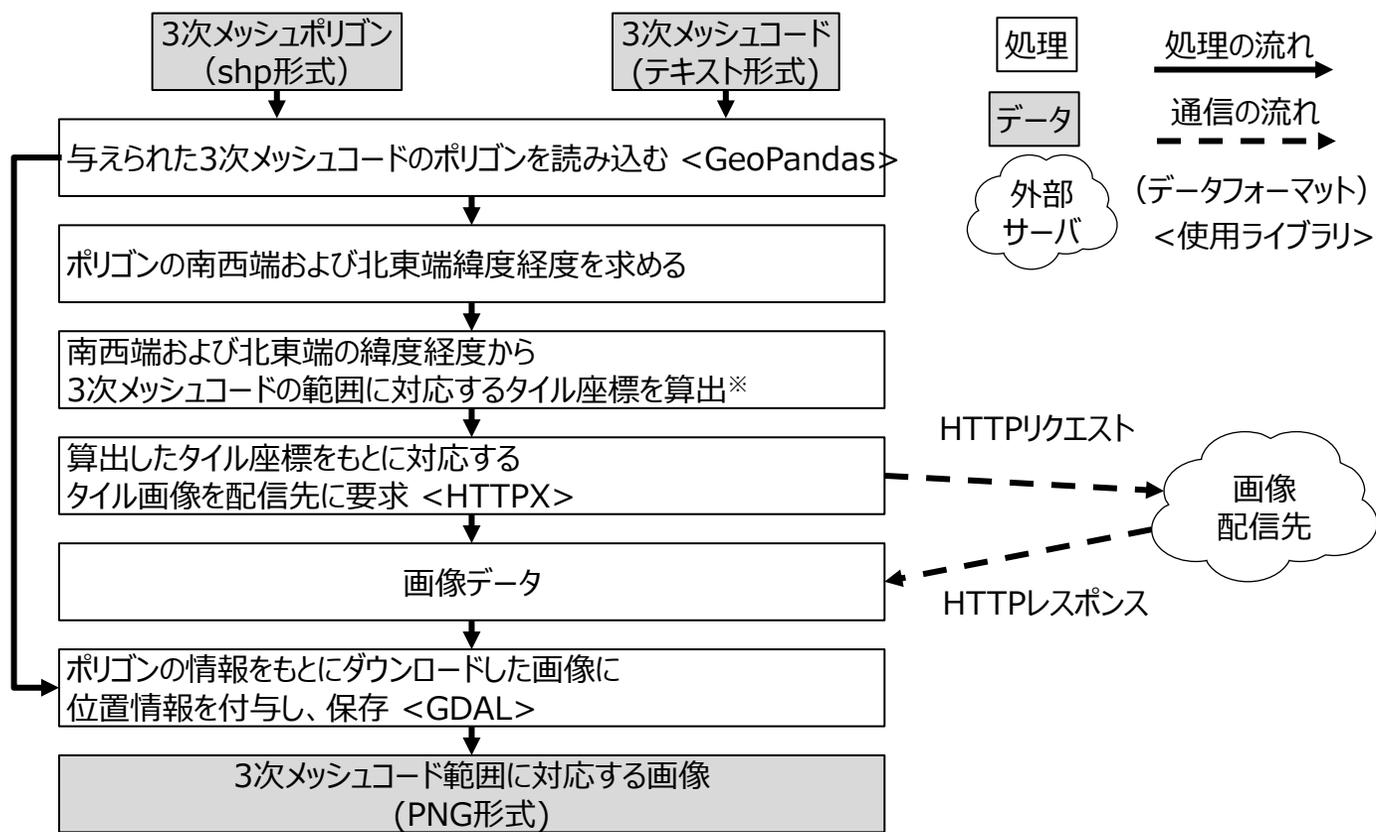
利用ライブラリ		用途
GeoPandas	地理空間情報データ操作ライブラリ	3次メッシュポリゴン情報の取得
GDAL	地理空間情報データ操作ライブラリ	取得した画像への地理情報を付与
HTTPX	HTTPクライアントライブラリ	外部サーバからのタイル画像取得

# Ⅲ. 実証システム > 4. システム機能

## ① 画像データ取得 | 処理フロー

旧時期および新時期のデータ流れは以下のとおり

旧時期および新時期のデータ取得処理の流れ



緯度経度からタイル座標の算出方法

緯度  $lat$ 、経度  $lon$  の地点に対応するズームレベル  $z$  での X 方向のタイル座標を  $x$ 、Y 方向のタイル座標を  $y$ 、と仮定すると、以下の数式で求まる。

$$x = \left\lfloor \frac{lon + 180}{360} \cdot 2^z \right\rfloor$$

$$y = \left\lfloor \left( 1 - \frac{\ln \left( \tan \left( lat \cdot \frac{\pi}{180} \right) + \frac{1}{\cos \left( lat \cdot \frac{\pi}{180} \right)} \right)}{\pi} \right) \cdot 2^{z-1} \right\rfloor$$

基本的に、旧時期データにおけるズームレベルは19、新時期データにおけるズームレベルは18を指定。

# Ⅲ. 実証システム > 4. システム機能

## ① 画像データ取得 | 使用方法

旧時期画像および新時期画像の取得

### 実行コマンドに関する仕様

項目	内容
実行するコマンド名	python download_tile_image.py [scene] --config [config]
引数名	scene: 新旧時期の指定。旧時期の場合はold、新時期の場合はnewを指定 config: 設定用のYAMLファイルのパスを指定
結果	png形式の画像を保存

### 設定用YAMLファイルの仕様

キー	説明
url	旧時期画像または新時期画像タイルデータの配信URL
mesh_code	ダウンロードする範囲の3次メッシュコードが記載されたテキストファイルのパス
Zoom_level	ダウンロードする画像のズームレベル 基本的に、旧時期の場合は19、新時期の場合は18を指定
output	画像の保存先ディレクトリのパス

### Ⅲ. 実証システム > 4. システム機能

## ②新時期画像の高解像度化（超解像AI）

機能仕様

項目	詳細
機能名	新時期画像の高解像度化
機能概要	新時期のALOS-3衛星シミュレーション画像に対して、旧時期の航空写真の解像度と同じになるように高解像度化する
入力データ仕様	ALOS-3衛星シミュレーション画像 (解像度80cm、PNG形式)
出力データ仕様	ALOS-3衛星シミュレーション画像を超解像処理し、高解像度化した画像 (解像度40cm)
利用するアルゴリズム	ESRGAN

利用ライブラリ

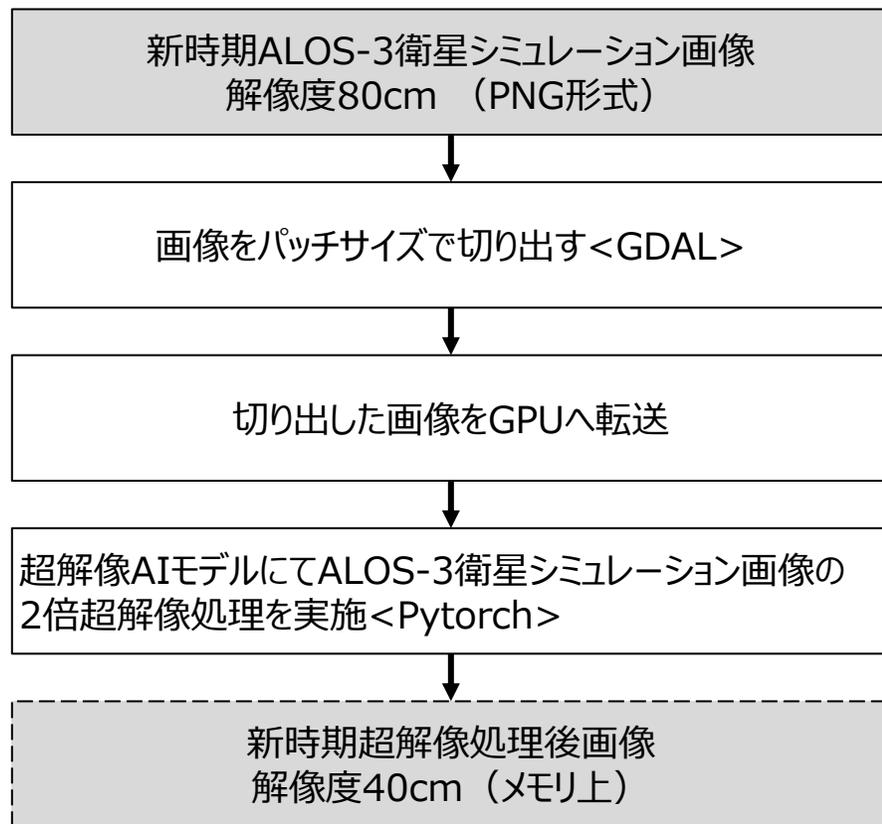
利用ライブラリ		用途
CUDA	GPU向け汎用並列計算環境	AIをGPU上で動作させるための環境
PyTorch	Deep Learningのフレームワーク	AIの構造定義や学習/判読処理
OpenCV	画像処理/機械学習ライブラリ	AIに入力するための画像加工
GDAL	地理空間情報データ操作ライブラリ	変化検出結果に地理情報を付与
Rasterio	地理空間情報画像操作ライブラリ	変化検出結果をGISデータに変換
GeoPandas	地理空間情報データ操作ライブラリ	GISデータを用いてメッシュデータを生成
MLFlow	AI実験管理用ライブラリ	AI学習結果・学習時設定パラメータの管理

### Ⅲ. 実証システム > 4. システム機能

## ②新時期画像の高解像度化（超解像AI） | 処理フロー

超解像AIおよび変化検出AIの流れは以下のとおり

#### 新時期画像の高解像度化



#### <凡例>

処理

データ

中間データ

処理の流れ

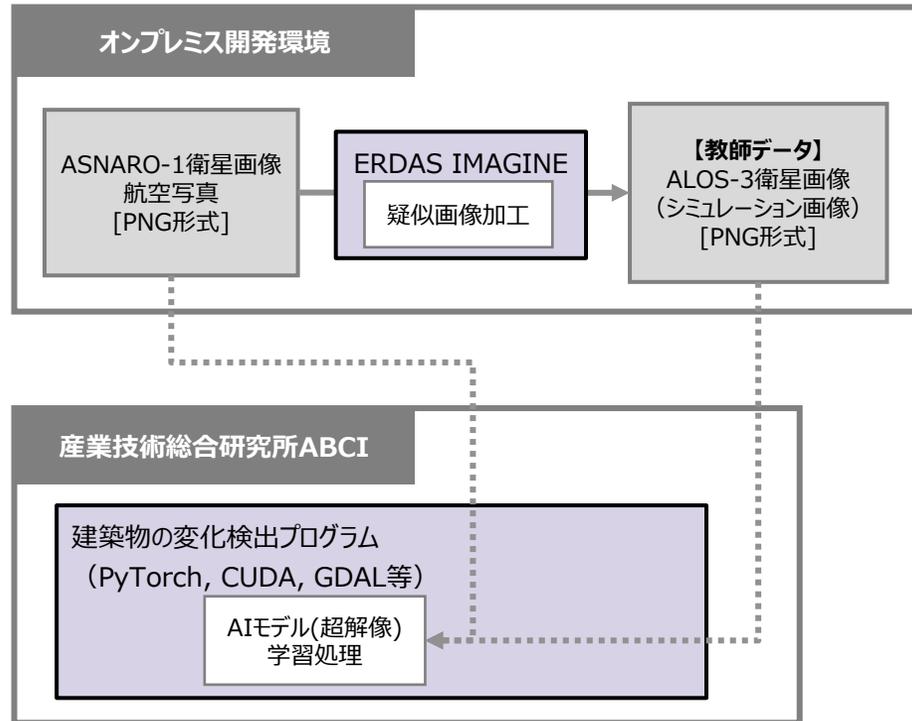
(データフォーマット)

<使用ライブラリ>

### Ⅲ. 実証システム > 4. システム機能

## ②新時期画像の高解像度化（超解像AI） | 学習方法

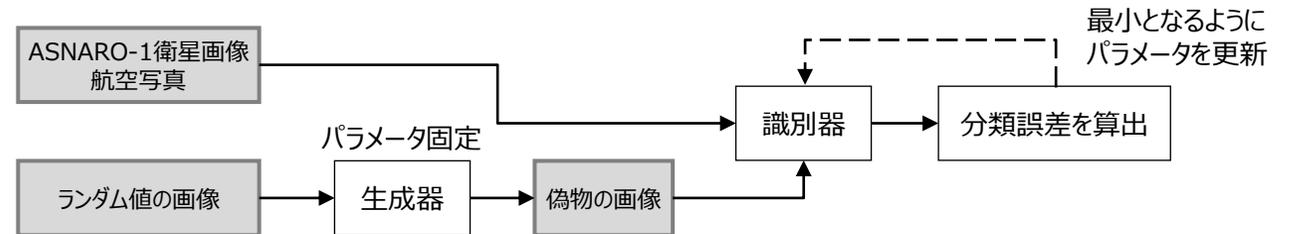
ALOS-3衛星シミュレーション画像（解像度80cm）を入力して、生成される超解像処理画像が元画像（解像度40cm）を再現するよう学習



- 超解像AIに使用しているモデルは、生成器と識別器の2つのネットワークから成り立っている
  - 生成器: ランダムな値が格納されている画像データから、目的とする画像の偽物画像を生成
  - 識別器: 入力された画像が生成器が作成した偽物画像か本物の画像かを判定

#### 識別器の学習

- 本物のASNARO-1衛星画像もしくは航空写真と生成器が作成した偽物の画像を
  - 識別器に入力し、識別器の分類誤差が最小になるように識別器のパラメータを更新
- 識別器の分類精度を高めることが目的



#### 生成器の学習

- ランダム値の画像データを生成し、識別器にて本物か偽物かを分類
  - 識別器の分類誤差が最大になるように生成器のパラメータを更新
- 生成器が本物の画像と同等の画像を再現できるようにすることが目的

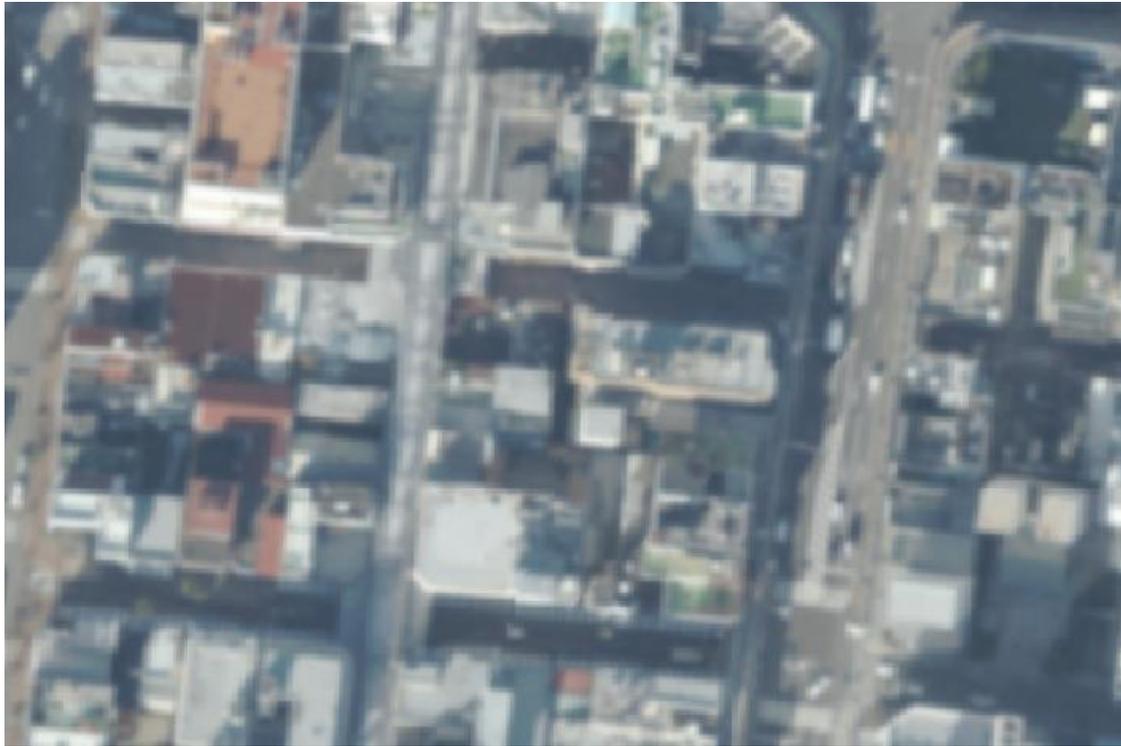


### Ⅲ. 実証システム > 4. システム機能

## ② 新時期画像の高解像度化（超解像AI） | 超解像処理画像

超解像処理画像のサンプル（解像度80cm→40cm）

ALOS-3衛星シミュレーション画像（解像度80cm）



超解像処理画像（解像度40cm）



# Ⅲ. 実証システム > 4. システム機能

## 【参考】AI開発・学習システム | ソフトウェア

開発で利用したソフトウェアの開発環境は以下のとおり

種別	名称	バージョン	内容
OS	Ubuntu	20.04 LTE	Linux Distribution
ミドルウェア	miniconda	4.9.2	Pythonライブラリの管理
ライブラリ	CUDA	11.3	GPU向け汎用並列計算プラットフォーム (AIをGPUで動作させるために必要)
	lxml	4.8.0	xml/html形式データ操作ライブラリ (CityGMLからジオメトリなどの情報の抽出に使用)
	GeoPandas	0.10.2	地理空間情報データ操作ライブラリ (GISデータの加工のために使用)
	GDAL	3.4.3	地理空間情報データ操作ライブラリ (GISデータから画像に変換 / 地理情報の付与に使用)
	Rasterio	1.2.10	地理空間情報画像操作ライブラリ (変化検出結果からGISデータへの変換に使用)
	PyTorch	1.11.0	Deep Learningフレームワーク (AIの構造定義や学習/判読処理のために使用)
	MLFlow	1.26.1	AI実験管理用ライブラリ (AI学習結果・学習時設定パラメータ管理に使用)
	OpenCV	4.5.5	画像処理/機械学習ライブラリ (AIに入力するための画像加工に使用)

## Ⅲ. 実証システム > 4. システム機能

# 【参考】AI開発・学習システム | ハードウェア

開発で利用したハードウェアの開発環境は以下のとおり

種別	スペック	内容
GPUサーバ	<ul style="list-style-type: none"><li>仮想CPU : 4コア</li><li>メモリ : 56GB</li><li>GPU : NVIDIA A100×1</li></ul>	AIモデルの学習・評価用

## Ⅲ. 実証システム > 4. システム機能

### ③ 建築物の変化検出（変化検出AI）

機能仕様

項目	詳細
機能名	建築物の変化検出
機能概要	同じ場所が撮影されている旧時期画像と新时期画像から建築物の変化を抽出
入力データ仕様	位置情報が付与されている旧時期および新时期の画像 (RGB3バンド、Byte、解像度40cm)
出力データ仕様	建築物の変化確率 (0~1) が各ピクセルの値として格納された画像 (1バンド、Float32、解像度40cm)
利用するアルゴリズム	U-Net (Encoder部分をSiamese化)

利用ライブラリ

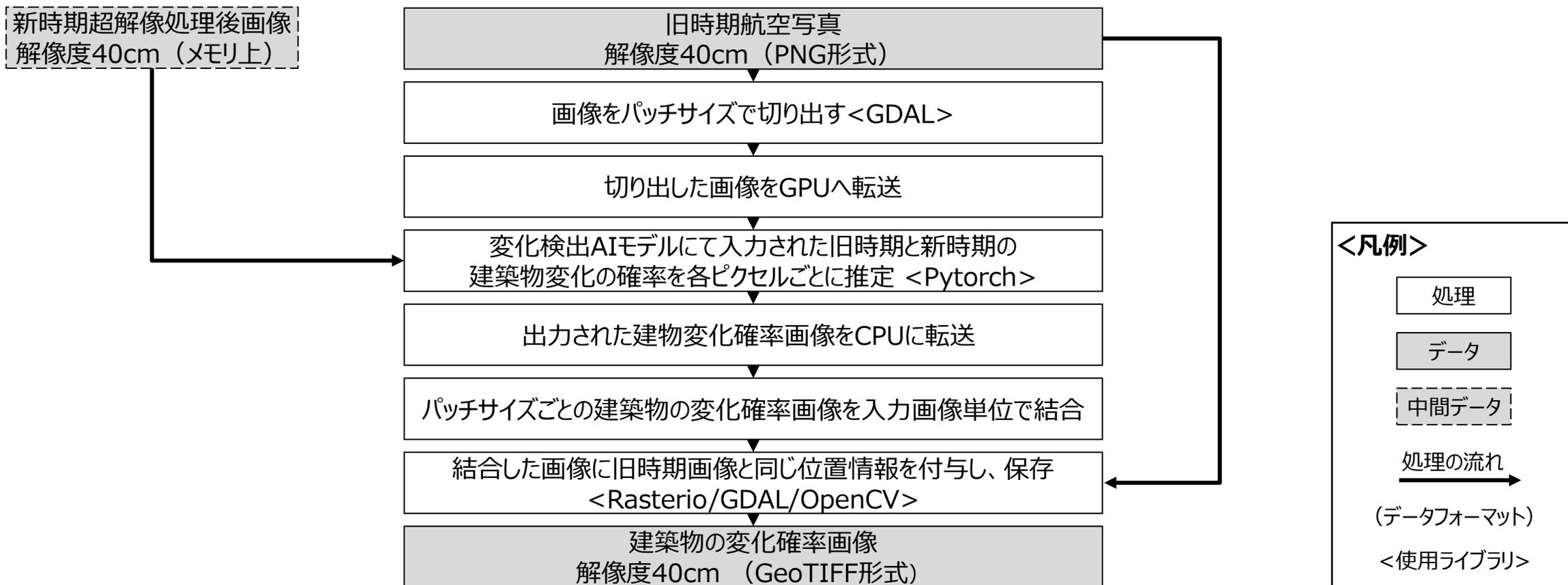
利用ライブラリ		用途
CUDA	GPU向け汎用並列計算環境	AIをGPU上で動作させるための環境
PyTorch	Deep Learningのフレームワーク	AIの構造定義や学習/判読処理
OpenCV	画像処理/機械学習ライブラリ	AIに入力するための画像加工
GDAL	地理空間情報データ操作ライブラリ	変化検出結果に地理情報を付与
Rasterio	地理空間情報画像操作ライブラリ	変化検出結果をGISデータに変換
GeoPandas	地理空間情報データ操作ライブラリ	GISデータを用いてメッシュデータを生成
MLFlow	AI実験管理用ライブラリ	AI学習結果・学習時設定パラメータの管理

### Ⅲ. 実証システム > 4. システム機能

## ③ 建築物の変化検出（変化検出AI） | 処理フロー

超解像AIおよび変化検出AIの流れは以下のとおり

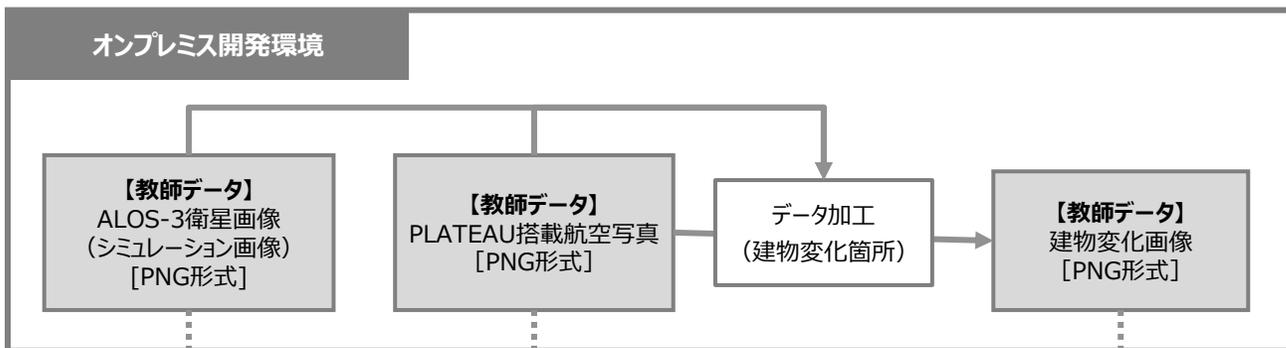
#### 建築物の変化検出



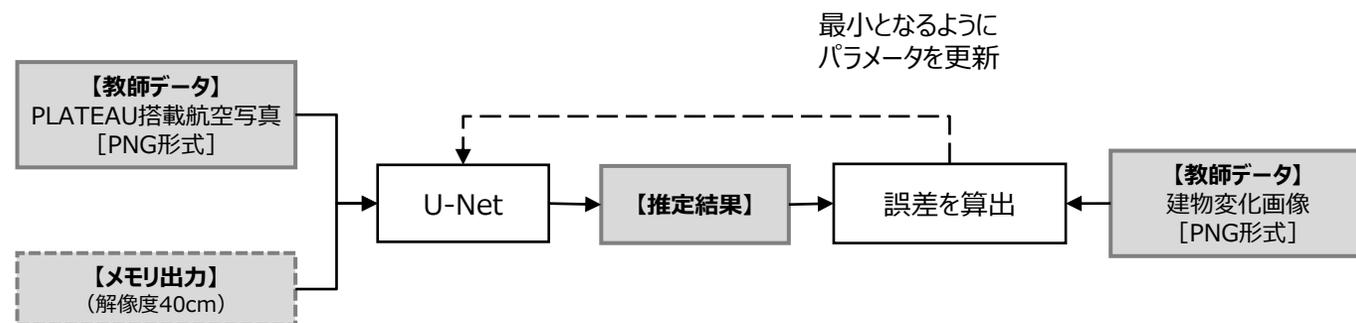
### Ⅲ. 実証システム > 4. システム機能

## ③ 建築物の変化検出（変化検出AI） | 学習方法

旧時期画像と新時期画像から変化検出された結果が、教師データに近付くよう学習



PLATEAUにて公開されている航空写真と超解像処理したALOS-3衛星シミュレーション画像を入力し、モデル（U-Net）推定結果と教師データの誤差が最小になるようにモデルのパラメータを更新  
→ モデルの出力を教師データに近づけることが目的



## Ⅲ. 実証システム > 4. システム機能

# ③ 建築物の変化検出（変化検出AI） | 使用方法

建築物の変化検出の使用方法（新時期画像の高解像度化の使用方法を含む）

### 実行コマンドに関する仕様

項目	内容
実行するコマンド名	python generate_probmap.py --config [config]
引数名	config: 設定用のYAMLファイルのパスを指定
結果	建築物変化検出結果をGeoTIFF形式で出力

### 設定用YAMLファイルの仕様

キー	説明
patch_size	判読時のパッチサイズ（デフォルトは1024）
batch_size	判読時のバッチサイズ デフォルトは1
cd_weight	変化検出モデルの重みのパス
sr_weight	超解像モデルの重みのパス
output	建築物変化確率画像の保存先ディレクトリのパス

### Ⅲ. 実証システム > 4. システム機能

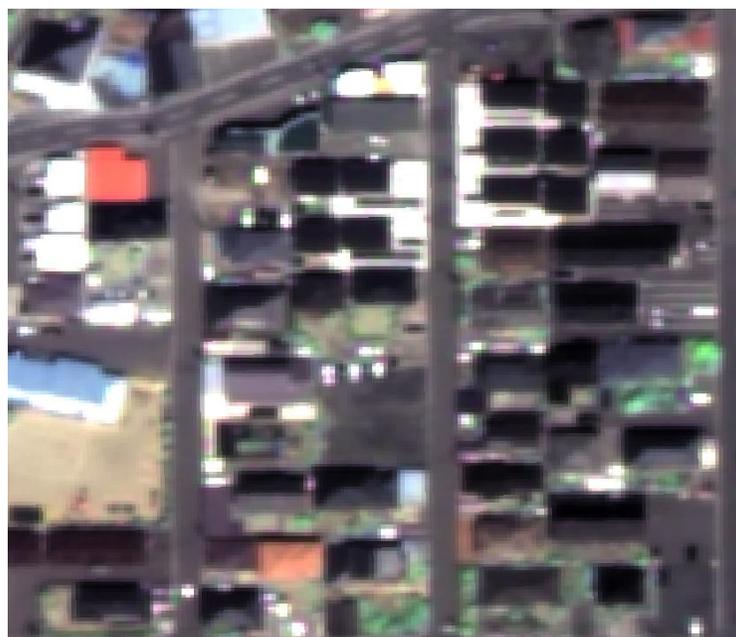
## ③ 建築物の変化検出（変化検出AI） | 変化検出結果の例

建築物の変化検出結果のサンプル

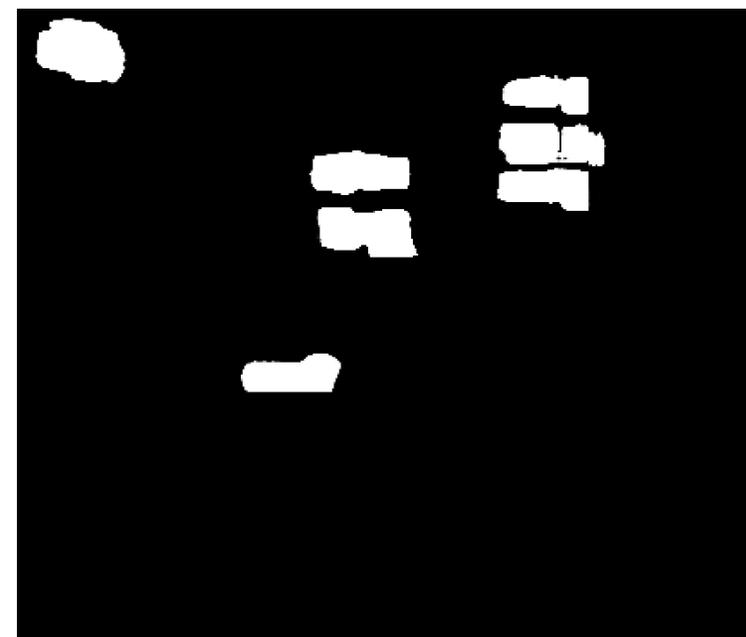
入力（旧時期航空写真）



入力（新时期ALOS-3衛星シミュレーション画像）



出力（建築物の変化結果画像）



# Ⅲ. 実証システム > 4. システム機能

## ④ メッシュ変換処理

機能仕様

項目	詳細
機能名	メッシュ変換処理
機能概要	建築物の変化検出結果画像から地域メッシュごとに変化率を算出し、更新優先度を決定
入力データ仕様	建築物の変化確率画像 (解像度40cm、GeoTIFF形式)
出力データ仕様	更新優先度が格納された地域メッシュの ポリゴンデータ (GeoJSON形式)
利用するアルゴリズム	自然分類

利用ライブラリ

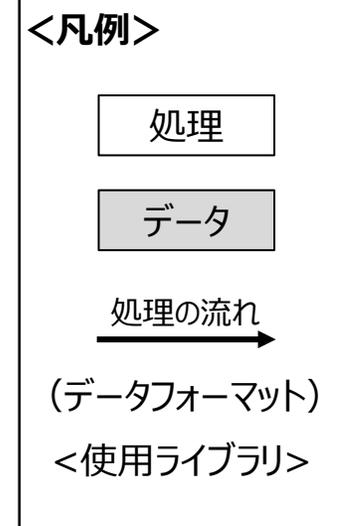
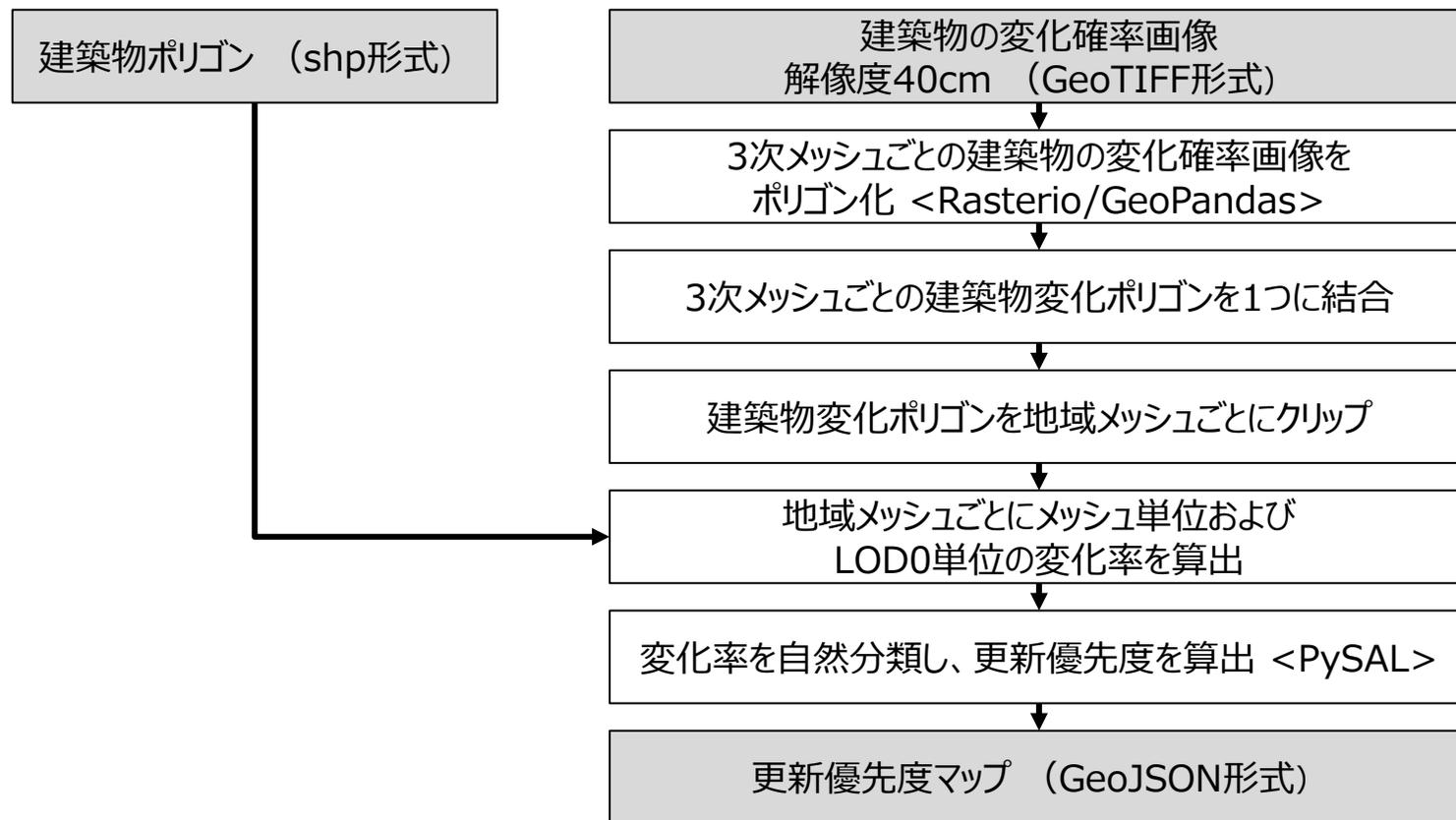
利用ライブラリ		用途
Rasterio	地理空間情報画像操作ライブラリ	変化検出結果をGISデータに変換
GeoPandas	地理空間情報データ操作ライブラリ	GISデータを用いてメッシュデータを生成
PySAL	空間分析用ライブラリ	メッシュごとの更新優先度の計算

# Ⅲ. 実証システム > 4. システム機能

## ④ メッシュ変換処理 | 処理フロー

メッシュ変換処理の流れは以下のとおり

### メッシュ変換処理



## Ⅲ. 実証システム > 4. システム機能

# ④メッシュ変換処理 | 使用方法

### 更新優先度マップのメッシュ生成

#### 実行コマンドに関する仕様

項目	内容
実行するコマンド名	python create_mesh.py --config [config]
引数名	config: 設定用のYAMLファイルのパスを指定
結果	地域メッシュごとに更新優先度を格納したGeoJSONファイルを出力

#### 設定用YAMLファイルの仕様

キー	内容
bldg	LOD0単位で変化率を算出する際にもととなる建物ポリゴンのパス
threshold	既存の分類閾値が記載されたCSVファイルのパス
epsg	更新優先度マップ作成範囲の平面直角座標系のEPSGコード
output	保存先ディレクトリのパス
filename	出力する更新優先度マップのファイル名

# Ⅲ. 実証システム > 4. システム機能

## ⑤ スタイリング処理

機能仕様

項目	詳細
機能名	スタイリング処理
機能概要	PLATEAU VIEWなどsimplestyle-specに対応したWebGISで表示する際の色付けを反映する処理
入力データ仕様	更新優先度マップ (GeoJSON形式)
出力データ仕様	更新優先度マップ (GeoJSON形式)
利用するアルゴリズム	-

利用ライブラリ

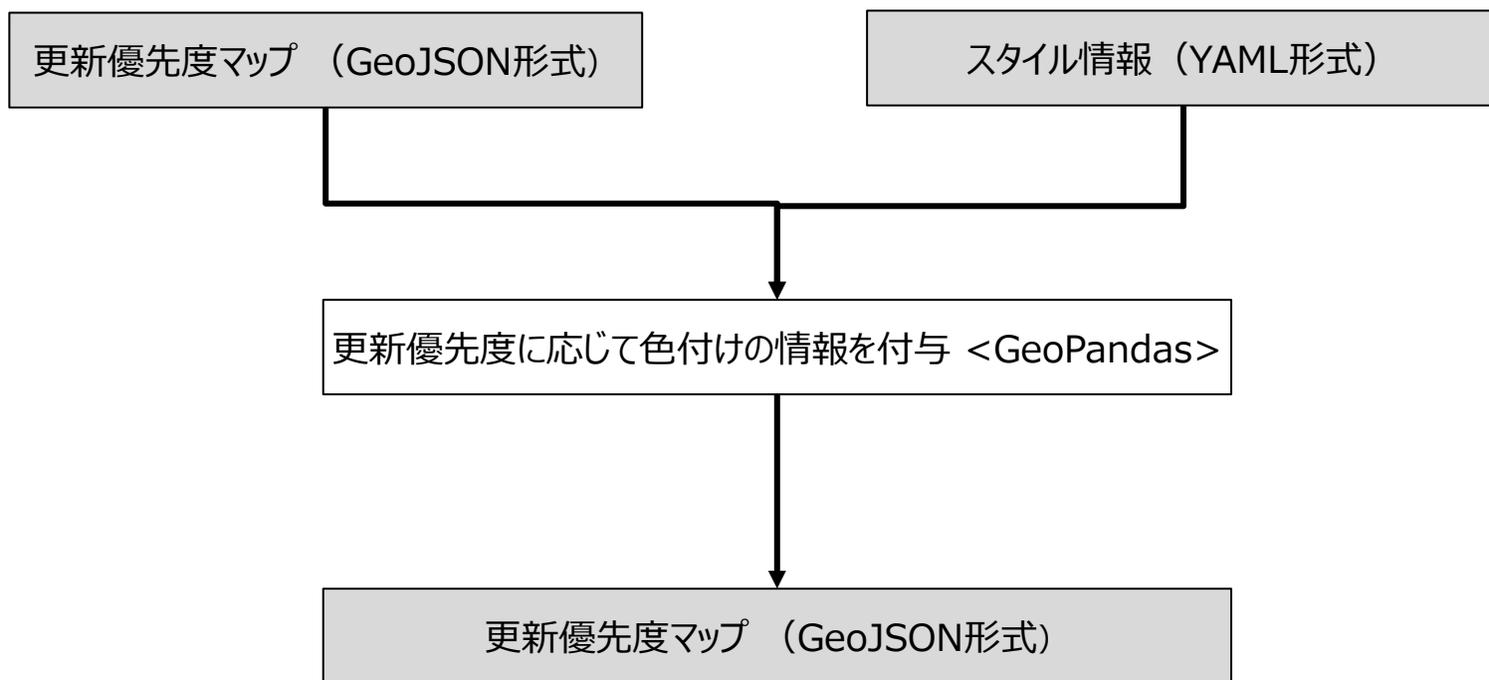
利用ライブラリ		用途
GeoPandas	地理空間情報データ操作ライブラリ	色付けの情報を付与

# Ⅲ. 実証システム > 4. システム機能

## ⑤ スタイリング処理 | 処理フロー

メッシュ変換処理の流れは以下のとおり

### スタイリング処理



#### <凡例>

処理

データ

処理の流れ

(データフォーマット)

<使用ライブラリ>

## Ⅲ. 実証システム > 4. システム機能

# ⑤ スタイリング処理 | 使用方法 (1/2)

### 更新優先度マップのスタイリング

#### 実行コマンドに関する仕様

項目	内容
実行するコマンド名	python style_mesh.py --config [config]
引数名	scene: 新旧時期の指定。新时期の場合はnewを指定
結果	スタイル（色付け）が反映されたGeoJSON形式の更新優先度マップを出力

#### 設定用YAMLファイルの仕様

キー	内容
output_dir	更新優先度マップの保存先ディレクトリのパス
denom	変化率算出時の分母（mesh: メッシュ単位、lod0: LOD0単位）
num_classes	更新優先度マップの分類数

## Ⅲ. 実証システム > 4. システム機能

# ⑤スタイリング処理 | 使用方法 (2/2)

### 更新優先度マップのスタイリング

#### 設定用YAMLファイルの仕様

キー	内容
stroke	更新優先度マップにおけるメッシュの枠線の色 16進カラーコードで記載
stroke-width	更新優先度マップにおけるメッシュの枠線の太さ 単位はpx
stroke-opacity	更新優先度マップにおけるメッシュの枠線の透明度 0 (完全に透過) ~1 (不透過) の間で10進小数で記載
fill	更新優先度マップにおけるメッシュの塗りつぶし色 16進カラーコードで記載
fill-opacity	更新優先度マップにおけるメッシュの塗りつぶしの透過度 0 (完全に透過) ~1 (不透過) の間で10進小数で記載

# Ⅲ. 実証システム > 5. アルゴリズム

## アルゴリズム一覧

高速かつ省メモリな超解像AIモデルと2つの画像情報から変化を抽出する変化検出AIモデルを開発した

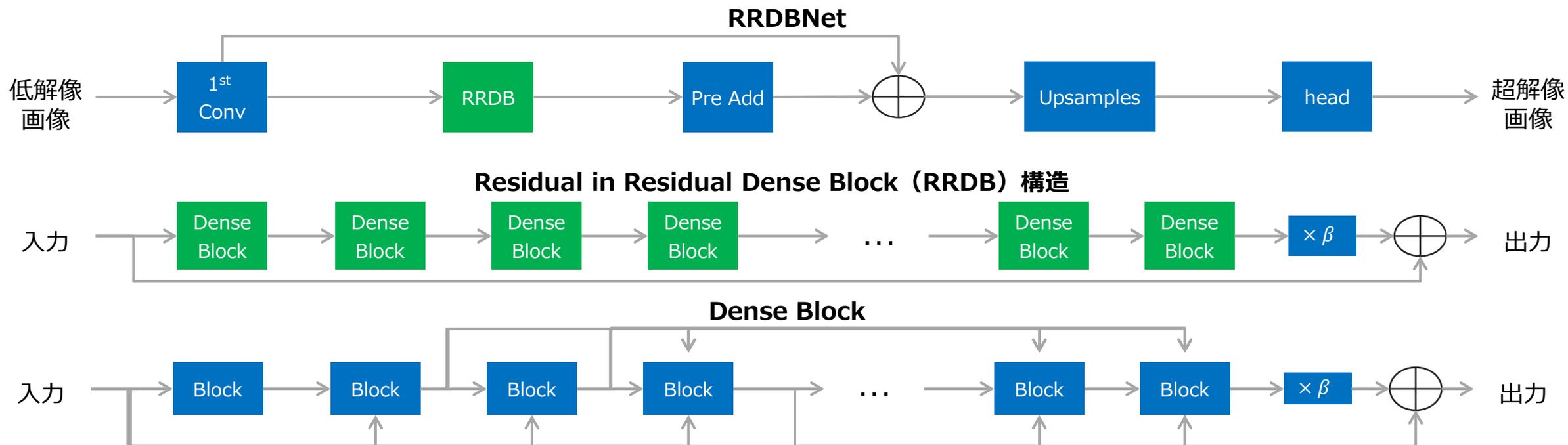
AIモデル	内容
超解像AIモデル	新時期ALOS-3衛星シミュレーション画像を旧時期航空写真の解像度となるように高解像度化する
変化検出AIモデル	旧時期航空写真と超解像処理後の新時期画像から建築物の変化を抽出する

# Ⅲ. 実証システム > 5. アルゴリズム

## 超解像AIモデル

ESRGAN\*をベースとした高速かつ省メモリな超解像AIモデル（RRDBNet）を使用する

- Residual in Residual Dense Block（RRDB）構造を採用することで超解像画像の品質を向上
  - より多くの層と接続を持つことがAIモデル精度向上に繋がるという発想から提案された構造
  - 複数の残差構造（Residual in Residual）によって層数を増やすとともに、全ての層に特徴を伝搬させる接続を持つブロック（Dense Block）を用いて接続を増加



\* <https://arxiv.org/abs/1809.00219>

# Ⅲ. 実証システム > 5. アルゴリズム

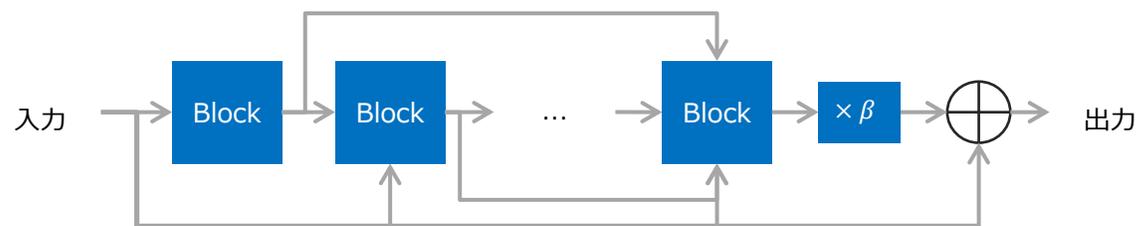
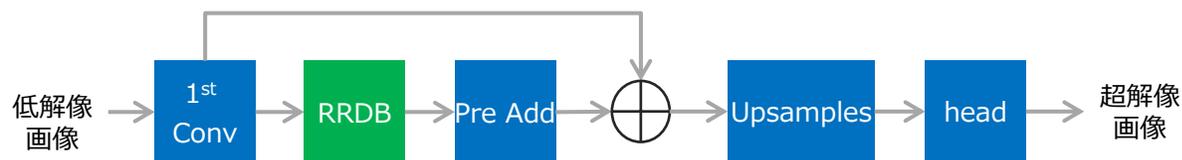
## 超解像AIモデル

超解像AIモデル（実装コード、動作部分のみ抜粋）

```
class RRDBNet(nn.Module):
    def forward(self, x: torch.Tensor) -> torch.Tensor:
        x_f = self.first(x)
        x_res = self.pre_add(self.residual_blocks(x_f))
        x_add = torch.add(x_f, x_res)
        return self.head(self.upsamples(x_add)).sigmoid()
```

```
class RRDB(nn.Module):
    def forward(self, x):
        residual = x
        for block in self.dense_blocks:
            residual = block(residual)
        return x + self.weight * residual
```

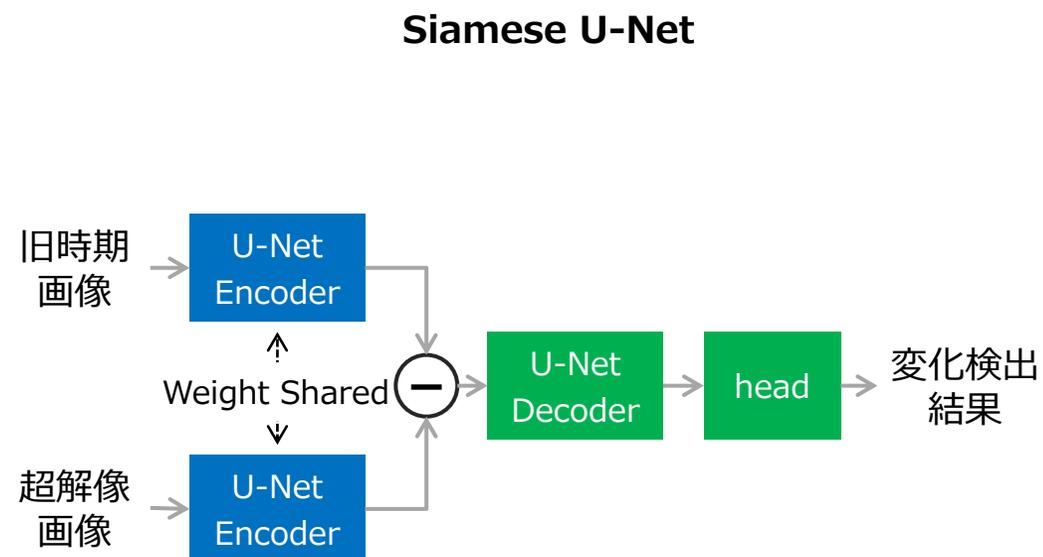
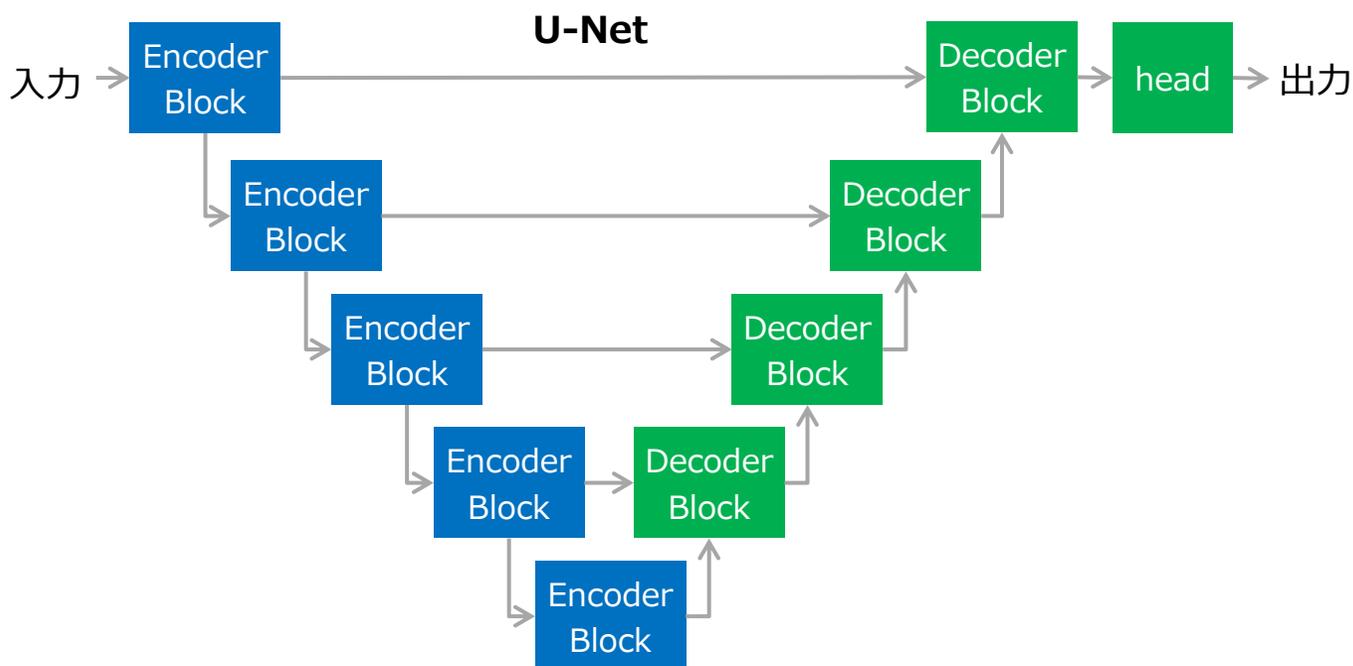
```
class DenseResidualBlock(nn.Module, ModelMixin):
    def forward(self, x: torch.Tensor) -> torch.Tensor:
        residual = x
        for i, block in enumerate(self.blocks):
            out = block(residual)
            residual = torch.cat([residual, out], dim=1) if i < len(self.blocks) - 1 else out
        return x + self.weight * residual
```



# Ⅲ. 実証システム > 5. アルゴリズム 変化検出AIモデル

U-Net\*をSiamese構造にして変化検出を行なえるよう対応する

- U-Net\*をSiamese構造にして変化検出を行なえるよう対応
  - 2枚の入力画像に対して、それぞれ用のCNNから構成されるネットワーク構造
  - 双方のパラメータを共通化しておくことにより、画像ペア同士が似ているか等を判断する特徴を抽出可能
- 旧時期・超解像画像から特徴量を抽出（Encoder）、抽出した特徴量から変化箇所を検出（Decoder）
  - Encoderの重みはそれぞれで共有されたものを使用



\* <https://arxiv.org/abs/1505.04597>

# Ⅲ. 実証システム > 5. アルゴリズム

## 変化検出AIモデル

変化検出AIモデル（実装コード、動作部分のみ抜粋、一部改変）

```
class SiamDiff(nn.Module, ModelMixin):
```

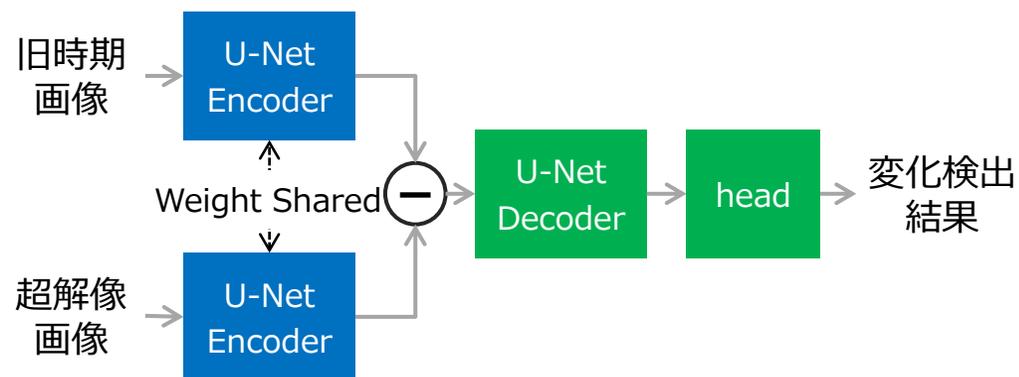
```
    def forward(self, x_old: torch.Tensor, x_new: torch.Tensor) -> torch.Tensor:
        x_old_res = self.unet_encoder(x_old)
        x_new_res = self.unet_encoder(x_new)
        diff = [x_n - x_o for x_n, x_o in zip(x_new_res, x_old_res)]
        x_dec = self.unet_decoder(diff[-1], diff[:-1])[-1]
        return self.head(x_dec)
```

```
class UNetEncoder(nn.Module, ModelMixin):
```

```
    def forward(self, x: torch.Tensor) -> list[torch.Tensor]:
        output = []
        for block in self.encoder_blocks:
            x = block(x)
            output.append(x)
        return output
```

```
class UNetDecoder(nn.Module, ModelMixin):
```

```
    def forward(self, x: torch.Tensor, x_enc: list[torch.Tensor]) -> list[torch.Tensor]:
        output = []
        for block, enc in zip(self.decoder_blocks, reversed(x_enc)):
            x = block(x, enc)
            output.append(x)
        return output
```



## Ⅲ. 実証システム > 5. アルゴリズム 変化検出AIモデル

超解像AIは、ALOS-3衛星画像（解像度80cm）の解像度を上げることで、旧時期画像の航空写真（解像度40cm）との変化検出精度を向上させることを目的として利用

超解像処理前の画像（元画像20cm）



超解像画像処理後（元画像20→10cmに超解像処理済）



### 旧時期画像（40cm程度）と新时期画像（80cm）の空間解像度の違いの解消

- 旧時期画像の航空写真と新时期画像では、約2倍空間解像度が異なり、変化検出モデルの入力データのために、解像度を合わせる必要がある
- 解像度の低い方（80cm）に合わせるパターンもあるが、旧時期画像が持っている40cm解像度の情報が少なくなってしまうので、得策ではない
- 解像度の低い方（80cm）の解像度を上げて40cmにすると、通常の拡大処理ではノイズ（ジャギー）やボケ等が発生してしまうので、建物の形状等を再現する超解像処理を適用する
  - 建物形状がクリアになるとともに、ノイズが軽減され、変化検出モデルの精度向上が期待できる

# 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

本実証で活用した3D都市モデルは以下のとおり

地物	地物型	属性区分	属性名	内容
建築物LOD2	bldg:Building	空間属性	bldg:lod0RoofEdge	建築物の屋根面の外形線平面
			bldg:lod0FootPrint	建築物の接地面の外形線平面

# Ⅲ. 実証システム > 6. データ > ①活用データ PLATEAU公開済の3D都市モデル

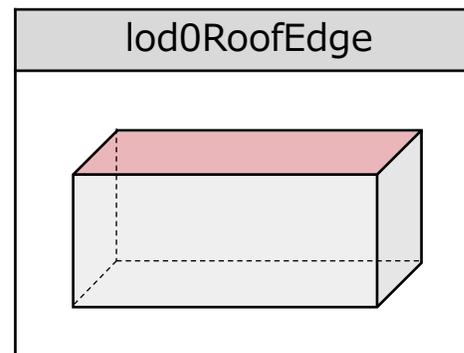
G空間情報センターにて公開されている3D都市モデル（CityGML形式）から建物のLOD0の空間属性を示す「lod0FootPrint」もしくは「lod0RoofEdge」の情報をポリゴン（面のGISデータ）に変換し使用した

ポリゴンに変換する際に使用した空間属性

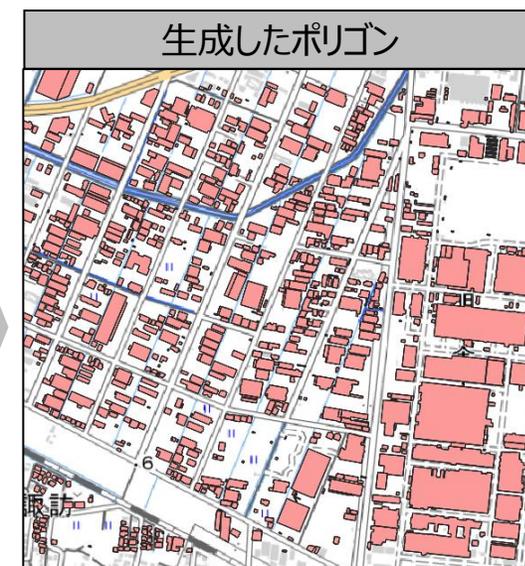
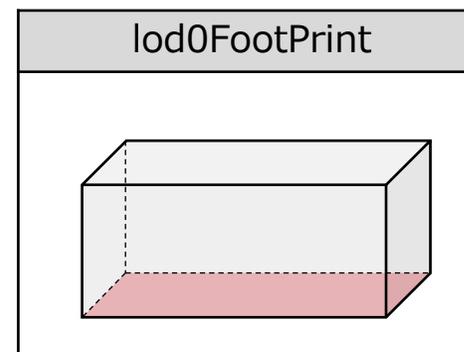
空間属性→ポリゴンへの変換イメージ

対象都市	使用した空間属性
札幌市	lod0RoofEdge
郡山市	lod0FootPrint
東京都23区	lod0RoofEdge
横浜市	lod0FootPrint
沼津市	lod0RoofEdge
掛川市	lod0FootPrint
大阪市	lod0FootPrint
豊中市	lod0RoofEdge

- 使用した空間属性
  - 「lod0RoofEdge」：建築物の屋根面
  - 「lod0FootPrint」：建築物の接地面
- 本実証では、航空写真および衛星画像から変化を抽出するため、**屋根の外形線で示された要素である「lod0RoofEdge」**をなるべく用いた



または



# 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

本実証で活用したデータは以下のとおり

活用データ	内容	データ形式	出所
PLATEAU公開済の航空写真	<ul style="list-style-type: none"> <li>更新優先度マップ作成時の入力データとして利用し、新旧2時期の変化検出時の旧時期データとなる</li> <li>PLATEAUとしてストリーミング公開済の都市から実証エリアを選定した</li> </ul>	PNG形式	G空間情報センター
航空写真	<ul style="list-style-type: none"> <li>更新優先度マップ作成時の入力データ（新時期用）の元データとなる</li> <li>旧時期（PLATEAU公開済 航空写真）の撮影時期よりも撮影時期の新しい航空写真のオープンデータを選定した</li> </ul>	TIFF形式	G空間情報センター パスコ
ASNARO-1衛星画像	<ul style="list-style-type: none"> <li>更新優先度マップ作成時の入力データ（新時期用）の元データとなる</li> <li>旧時期（PLATEAU公開済 航空写真）の撮影時期よりも撮影時期の新しい航空写真のオープンデータを選定した</li> </ul>	TIFF形式	パスコ



## Ⅲ. 実証システム > 6. データ > ①活用データ

# PLATEAU公開済の航空写真

PLATEAUとしてストリーミング公開済の都市から8都市を実証エリアとして選定し、それぞれの航空写真を更新優先度マップ作成時の旧時期データとして利用する

対象都市	撮影時期	解像度	画像面積*1	概算容量*2 (PNG形式)	備考
札幌市	2019年	40cm	135km <sup>2</sup>	4.1GB	
郡山市	2016年	40cm	862km <sup>2</sup>	8.5GB	ズームレベル19の画像が配信されていなかったため、ズームレベル18で記載
東京都23区	2018年	50cm	695km <sup>2</sup>	16GB	
横浜市	2019年	40cm	505km <sup>2</sup>	14GB	
沼津市	2017年	40cm	266km <sup>2</sup>	5.1GB	
掛川市	2019年	40cm	265km <sup>2</sup>	8.2GB	
大阪市	2020年	100cm	269km <sup>2</sup>	441MB	航空写真の空間解像度が100cmのため、ズームレベル17で記載
豊中市	2019年	40cm	58km <sup>2</sup>	1.3GB	

\*1 PLATEAU公開済の3D都市モデルの整備範囲と航空写真の公開範囲には、行政界の縁周辺の差異のため、画像面積と行政面積は正確には一致しない

\*2 概算容量は、備考欄に記載が無い限りストリーミング配信のズームレベルを19（解像度30cm相当）で記載

## Ⅲ. 実証システム > 6. データ > ①活用データ

# 航空写真

実証エリアから3都市を選定し、旧時期（PLATEAU公開済の航空写真）の撮影時期よりも新しい航空写真のオープンデータをそれぞれ更新優先度マップ作成時の新时期データとして準備する

対象都市	撮影時期	画像面積	出所・備考
沼津市	2019～2021年	266km <sup>2</sup>	G空間情報センター（静岡県オープンデータ）
掛川市	2021年	265km <sup>2</sup>	G空間情報センター（静岡県オープンデータ）
豊中市	2022年	36km <sup>2</sup>	パスコ調達



# Ⅲ. 実証システム > 6. データ > ①活用データ

## ASNARO-1衛星画像

実証エリアから5都市を選定し、旧時期（PLATEAU公開済の航空写真）の撮影時期よりも新しいASNARO-1衛星画像のアーカイブデータをそれぞれ更新優先度マップ作成時の新时期データとして準備する

対象都市	撮影日	撮影角度 (入射角)	画像面積	出所・備考
札幌市	2022/06/14	12.17°	100km <sup>2</sup>	パスコ
郡山市	2022/05/18	3.50°	100km <sup>2</sup>	
東京都23区	2020/11/21	4.98°	470km <sup>2</sup>	パスコ 撮影範囲が広い ため、ASNARO-1衛星の複数シーンを利用
	2021/11/27	5.55°		
	2022/01/09	2.05°		
横浜市	2020/11/21	4.98°	160km <sup>2</sup>	
	2022/01/09	2.05°		
大阪市	2022/04/20	14.80°	176km <sup>2</sup>	
	2022/06/24	14.83°		

# Ⅲ. 実証システム > 6. データ > ②データ処理 データ処理一覧

システムに入力するデータ (データ形式)	目的	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)
ALOS-3衛星シミュレーション画像 (航空写真ベース) (TIFF形式)	<ul style="list-style-type: none"> <li>更新優先度マップ作成時の入力データとして利用し、新旧2時期の変化検出時の新時期データとなる</li> </ul>	<ul style="list-style-type: none"> <li>ALOS-3衛星画像が80cm解像度となるので、解像度を80cmにダウンサンプリングすることでシミュレーション画像を作成</li> </ul>	ERDAS IMAGINE	<ul style="list-style-type: none"> <li>PLATEAU公開済の航空写真 (PNG形式)</li> </ul>
ALOS-3衛星シミュレーション画像 (ASNARO-1衛星画像ベース) (TIFF形式)	<ul style="list-style-type: none"> <li>更新優先度マップ作成時の入力データとして利用し、新旧2時期の変化検出時の新時期データとなる</li> </ul>	<ul style="list-style-type: none"> <li>ALOS-3衛星画像が80cm解像度となるので、解像度を80cmにダウンサンプリングすることでシミュレーション画像を作成</li> </ul>	ERDAS IMAGINE	<ul style="list-style-type: none"> <li>ASNARO-1衛星画像 (TIFF形式)</li> </ul>
教師データ (建物変化) (PNG形式)	<ul style="list-style-type: none"> <li>AIモデルの学習用データや評価データとして利用される</li> <li>新旧2時期の画像データを比較して、建物の変化箇所を2値化して作成した画像</li> </ul>	<ul style="list-style-type: none"> <li>新旧2時期の画像データを目視で比較して、建物の変化箇所を2値画像として作成</li> </ul>	QGIS	<ul style="list-style-type: none"> <li>PLATEAU公開済の航空写真 (PNG形式)</li> <li>ALOS-3衛星シミュレーション画像 (TIFF形式)</li> </ul>
変化検出結果画像データ (PNG形式)	<ul style="list-style-type: none"> <li>更新優先度マップのメッシュ表現の元データとして利用される</li> </ul>	<ul style="list-style-type: none"> <li>更新優先度マップのAIモデル (変化検出) により、建物の変化情報を2値化画像として作成</li> </ul>	建築物の変化検出	<ul style="list-style-type: none"> <li>PLATEAU公開済の航空写真 (PNG形式)</li> <li>ALOS-3衛星シミュレーション画像 (TIFF形式)</li> </ul>



## Ⅲ. 実証システム &gt; 6. データ &gt; ②データ処理

## ALOS-3衛星シミュレーション画像 | 航空写真ベース (1/4)

実証エリアから3都市を選定し、旧時期（PLATEAU公開済の航空写真）の撮影時期よりも新しい航空写真のオープンデータをそれぞれ更新優先度マップ作成時の新时期データとして利用する

対象都市	撮影時期	解像度*1 (リサンプリング後)	画像面積	概算容量*2 (PNG形式)	出所・備考
沼津市	2019～2021年	80cm	266km <sup>2</sup>	2.4GB	G空間情報センター（静岡県オープンデータ）
掛川市	2021年	80cm	265km <sup>2</sup>	2.3GB	G空間情報センター（静岡県オープンデータ）
豊中市	2022年	80cm	36km <sup>2</sup>	0.3GB	パスコ調達

\*1 ALOS-3衛星画像が80cmの解像度となるので、各航空写真の解像度を80cmにダウンサンプリングすることでシミュレーション画像として作成  
(作成方法はASNAO-1衛星画像を用いた時と同様)

\*2 概算容量は、公開時のストリーミング配信を想定して、ズームレベル18（解像度60cm相当）の容量として記載

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | 航空写真ベース (2/4)

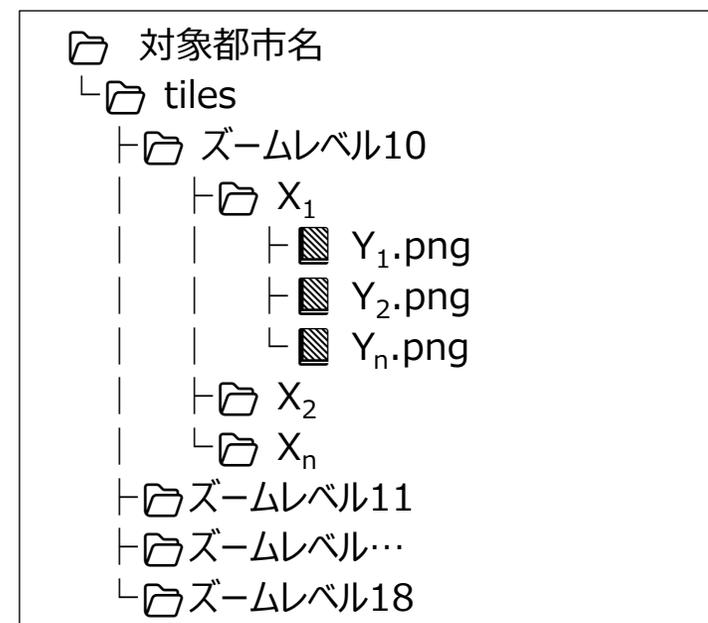
ALOS-3衛星画像の解像度は80cmであるが、データ公開後のストリーミング配信を考慮して、PLATEAU航空写真と同じデータ仕様で作成した

ALOS-3衛星シミュレーション画像のデータ仕様

項目	説明
配信形式	地図タイル画像 (XYZ Tiles) 形式
タイル1枚の画像サイズ	256×256ピクセル
画像フォーマット	PNG (Portable Network Graphics)
投影法	Webメルカトル (EPSG : 3857)
ズームレベル	10 (解像度153m相当) ~18 (解像度0.6m相当)
配信URL	<code>https://{FQDN}/tiles/{z}/{x}/{y}.png</code>

- データ作成のズームレベルは80cmがベースとなるため、解像度60cm相当のズームレベル10~18で設定した (PLATEAU航空写真のズームレベルは10~19)

各データのフォルダ構成



- ズームレベル別にフォルダを配置
- $X_1 \sim X_n$ および $Y_1 \sim Y_n$ は、地理院タイルに準拠するタイル座標

## Ⅲ. 実証システム &gt; 6. データ &gt; ②データ処理

# ALOS-3衛星シミュレーション画像 | 航空写真ベース (3/4)

オリジナル航空写真（解像度20cm）と比較すると、ALOS-3衛星シミュレーション画像（解像度80cm）の方が建物の輪郭が不明瞭になることが確認できる

ALOS-3衛星シミュレーション画像のサンプル

対象都市	オリジナル航空写真（解像度20cm）	ALOS-3衛星シミュレーション画像（解像度80cm）
静岡県沼津市 (沼津駅周辺)		
静岡県掛川市 (市街地)		

## Ⅲ. 実証システム &gt; 6. データ &gt; ②データ処理

## ALOS-3衛星シミュレーション画像 | 航空写真ベース (4/4)

オリジナル航空写真（解像度12cm）と比較すると、ALOS-3衛星シミュレーション画像（解像度80cm）の方が建物の輪郭が不明瞭になることが確認できる

ALOS-3衛星シミュレーション画像のサンプル

対象都市	オリジナル航空写真（解像度12cm）	ALOS-3衛星シミュレーション画像（解像度80cm）
大阪府豊中市 (豊中市役所周辺)		

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (1/9)

実証エリアから5都市を選定し、旧時期（PLATEAU公開済の航空写真）の撮影時期よりも新しいASNARO-1衛星画像のアーカイブデータをそれぞれ更新優先度マップ作成時の新时期データとして利用する

対象都市	撮影日	撮影角度 (入射角)	解像度*1 (リサンプリング後)	画像面積	概算容量*2 (PNG形式)	出所・備考
札幌市	2022/06/14	12.17°	80cm	100km <sup>2</sup>	0.9GB	パスコ
郡山市	2022/05/18	3.50°		100km <sup>2</sup>	0.9GB	
東京都23区	2020/11/21	4.98°		470km <sup>2</sup>	4.3GB	パスコ 撮影範囲が広い ため、ASNARO-1衛星 の複数シーンを利用
	2021/11/27	5.55°				
	2022/01/09	2.05°				
横浜市	2020/11/21	4.98°		160km <sup>2</sup>	1.5GB	
	2022/01/09	2.05°				
大阪市	2022/04/20	14.80°		176km <sup>2</sup>	1.6GB	
	2022/06/24	14.83°				

\*1 ALOS-3衛星画像が80cmの解像度となるので、各ASNARO-1衛星画像の解像度を80cmにダウンサンプリングすることでシミュレーション画像として作成（作成方法は航空写真を用いた時と同様）

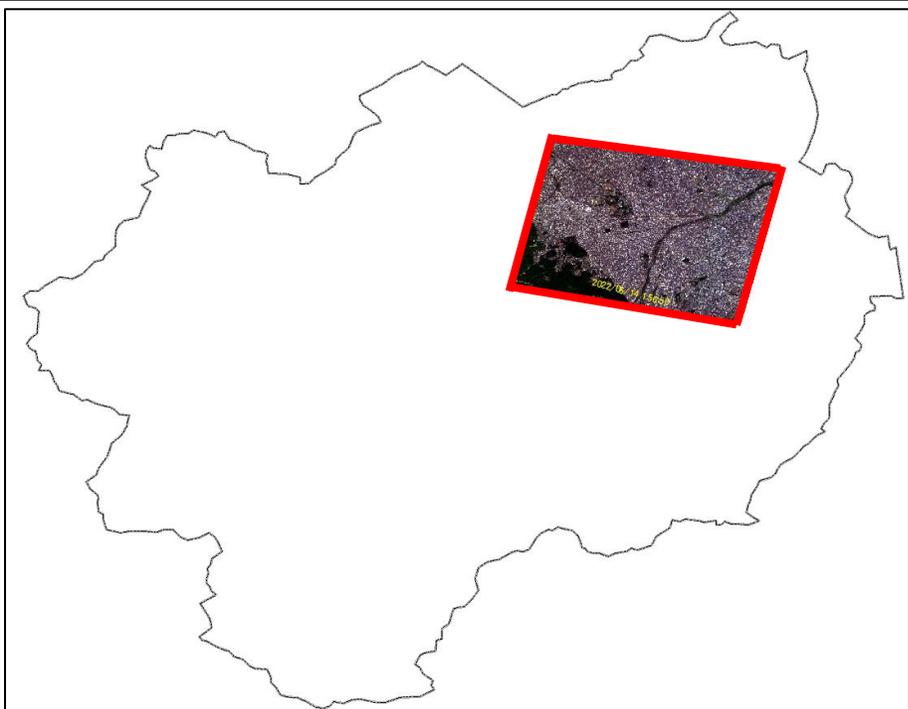
\*2 概算容量は、公開時のストリーミング配信を想定して、ズームレベル18（解像度60cm相当）の容量として記載

# Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (2/9)

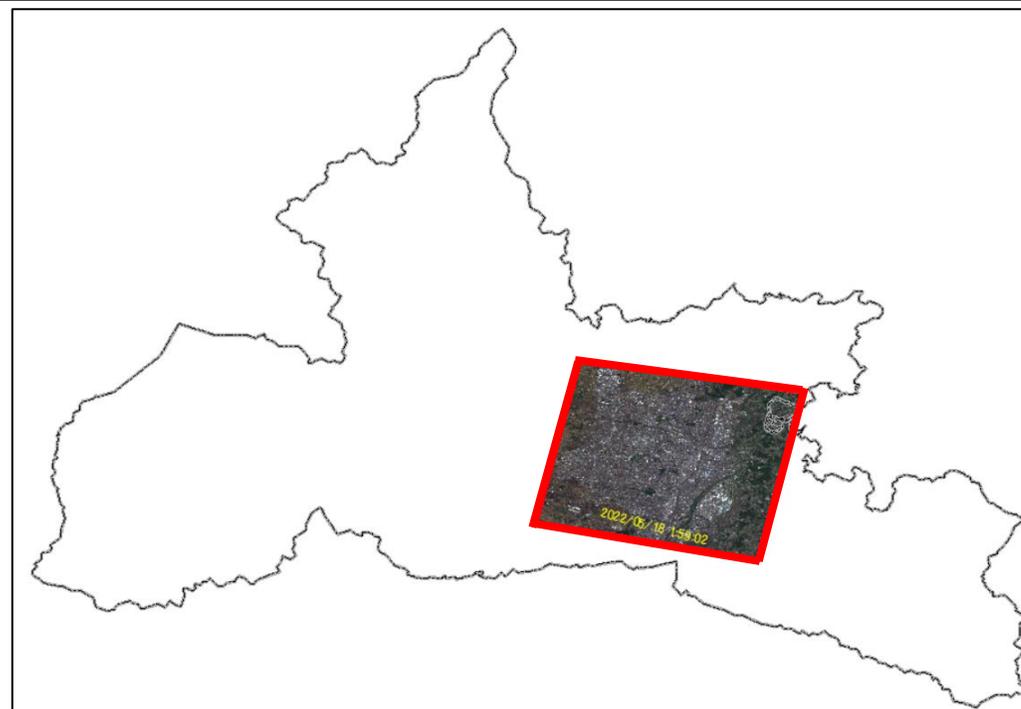
ASNARO-1衛星画像のアーカイブデータの範囲は以下のとおり

北海道札幌市のデータ範囲イメージ



- 撮影日 : 2022/06/14
- 撮影角度 (入射角) : 12.175°
- 雲量 : 0%

福島県郡山市のデータ範囲イメージ



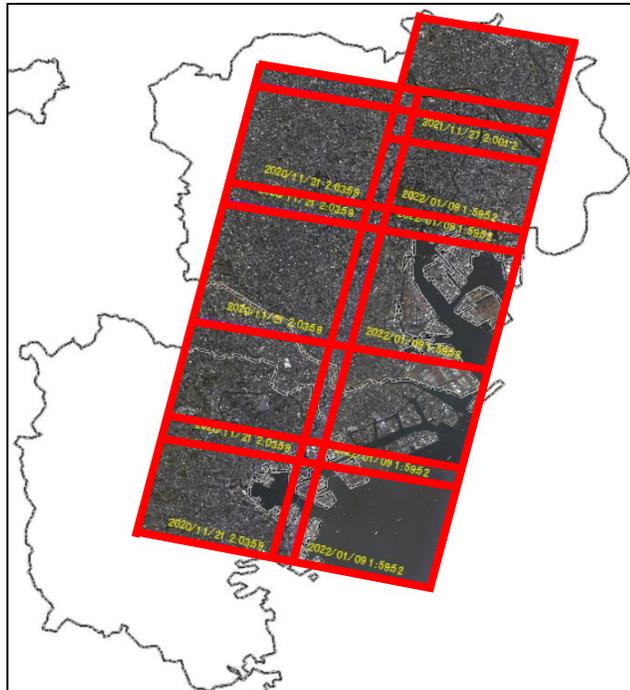
- 撮影日 : 2022/05/18
- 撮影角度 (入射角) : 3.507°
- 雲量 : 0%

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (3/9)

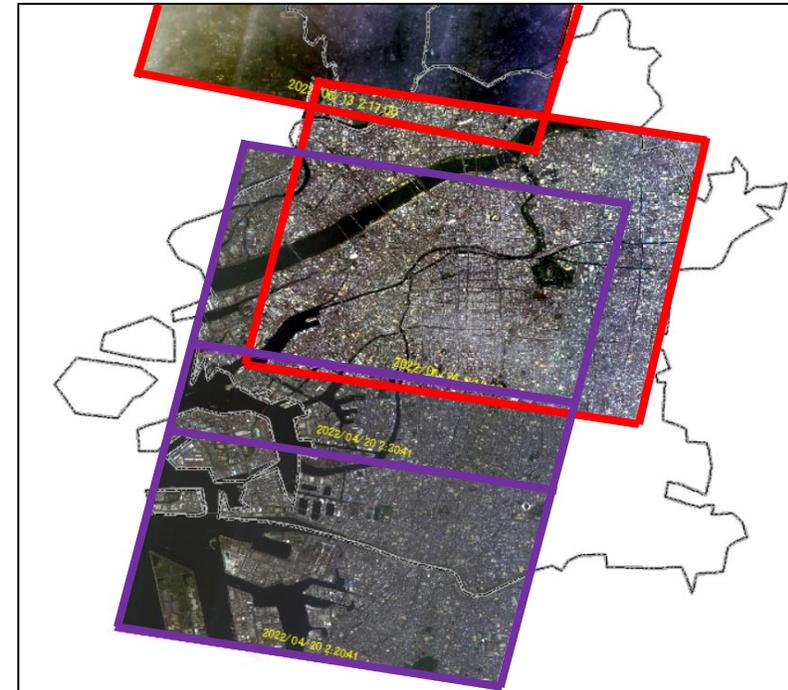
ASNARO-1衛星画像のアーカイブデータの範囲は以下のとおり

東京都23区・横浜市のデータ範囲イメージ



- 撮影日：2020/11/21～2022/01/09
- 撮影角度（入射角）：10°未満
- 雲量：0%

大阪府大阪市のデータ範囲イメージ



- 撮影日：2022/04/20（紫枠）、2022/06/24（赤枠）
- 撮影角度（入射角）：14.805°（紫枠）、14.833°（赤枠）
- 雲量：0%

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (4/9)

ASNARO-1衛星画像（解像度50cm）と比較すると、ALOS-3衛星シミュレーション画像（解像度80cm）の方が建物の輪郭が不明瞭になることが確認できる

ALOS-3衛星シミュレーション画像のサンプル (1/3)

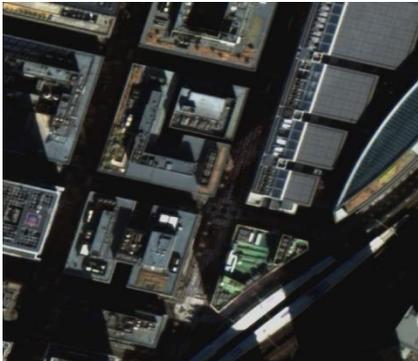
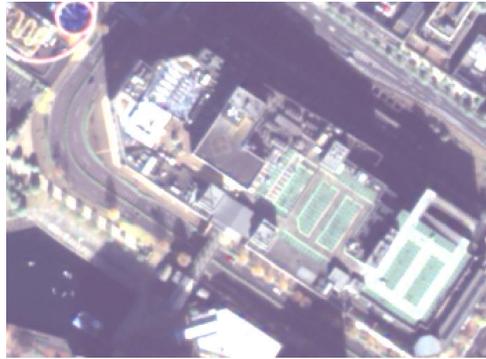
対象都市	撮影日	撮影角度 (入射角)	ASNARO-1衛星画像 (解像度50cm)	ALOS-3衛星シミュレーション画像 (解像度80cm)
北海道札幌市 (札幌駅周辺)	2022/06/14	12.17°		
福島県郡山市 (市役所周辺)	2022/05/18	3.50°		

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (5/9)

ASNARO-1衛星画像（解像度50cm）と比較すると、ALOS-3衛星シミュレーション画像（解像度80cm）の方が建物の輪郭が不明瞭になることが確認できる

ALOS-3衛星シミュレーション画像のサンプル (2/3)

対象都市	撮影日	撮影角度 (入射角)	ASNARO-1衛星画像 (解像度50cm)	ALOS-3衛星シミュレーション画像 (解像度80cm)
東京都23区 (東京駅周辺)	2022/01/09	2.05°		
神奈川県横浜市 (みなとみらい周辺)	2020/11/21	4.98°		

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (6/9)

ASNARO-1衛星画像（解像度50cm）と比較すると、ALOS-3衛星シミュレーション画像（解像度80cm）の方が建物の輪郭が不明瞭になることが確認できる

ALOS-3衛星シミュレーション画像のサンプル (3/3)

対象都市	撮影日	撮影角度 (入射角)	ASNARO-1衛星画像 (解像度50cm)	ALOS-3衛星シミュレーション画像 (解像度80cm)
大阪府大阪市 (京セラドーム大阪周辺)	2022/04/20	14.80°		

### Ⅲ. 実証システム > 6. データ > ②データ処理

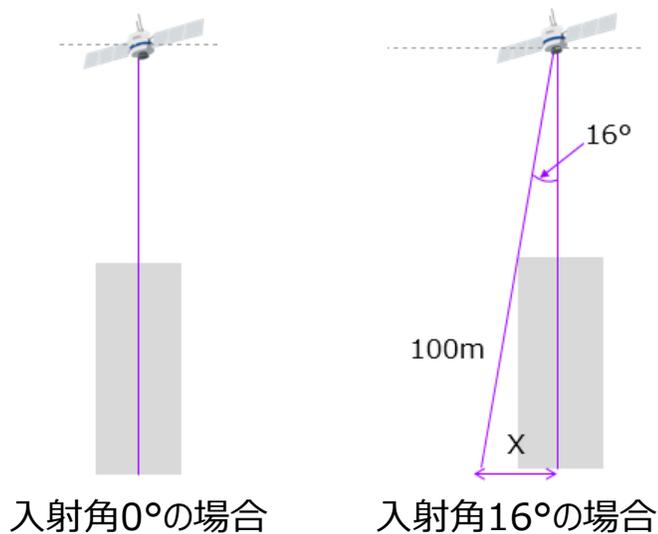
## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (7/9)



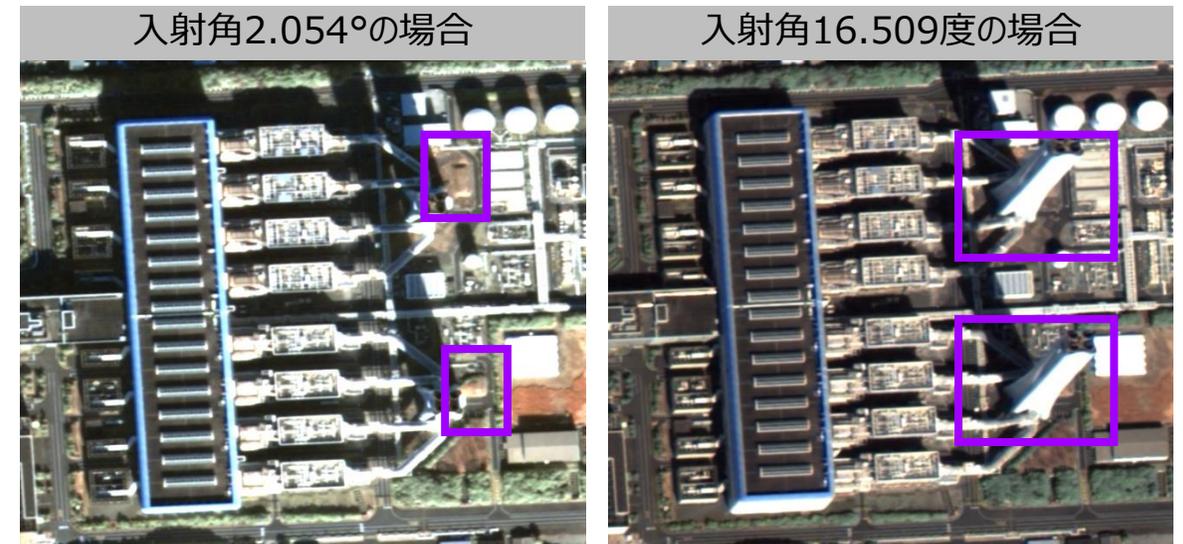
PLATEAU  
by MLIT

撮影時の入射角が大きいと、高層建築物の側面が映り込んでしまい倒れ込むような画像になってしまうため、影響を最小限に抑えるよう入射角の小さいASNARO-1衛星画像を選定した

撮影角度（入射角）の違いによる影響



横浜火力発電所の排気塔（高さ200m）の倒れ込みイメージ



- 入射角16°で撮影した場合、100mの建築物の倒れ込みは約29mとなる
- 計算式は以下の通り  
 $X = \tan 16^\circ \times 100\text{m} = 0.28675 \times 100\text{m} = 28.675\text{m} (\approx 29\text{m})$

- ALOS-3衛星画像は、航空写真と同様に入射角0°の直下視で撮影するのが基本だが、今回のシミュレーション画像ではASNARO-1衛星を用いるため入射角が発生する
- 建築物の倒れ込みが発生した場合、オルソ画像（倒れ込みのない画像）に加工しても影響を完全に排除することはできない

### Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (8/9)

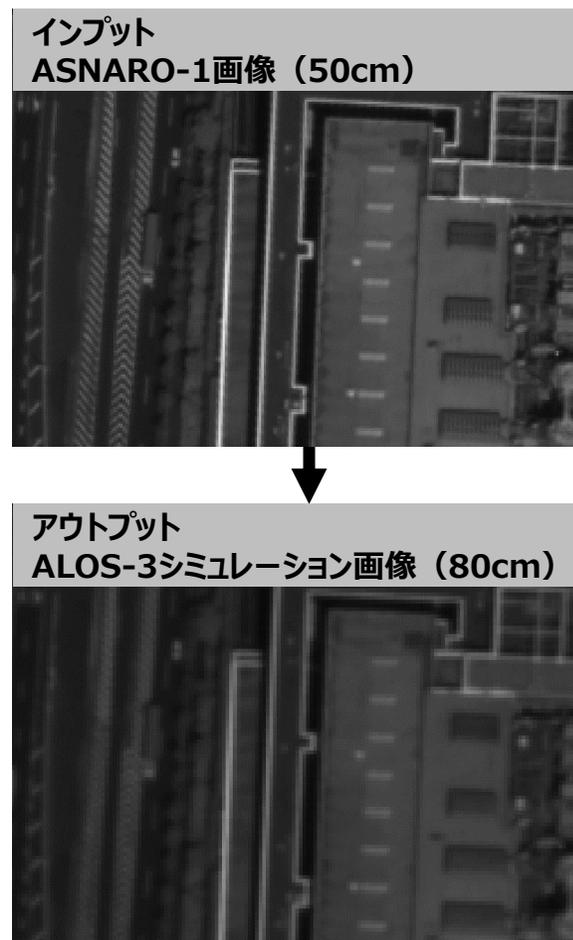
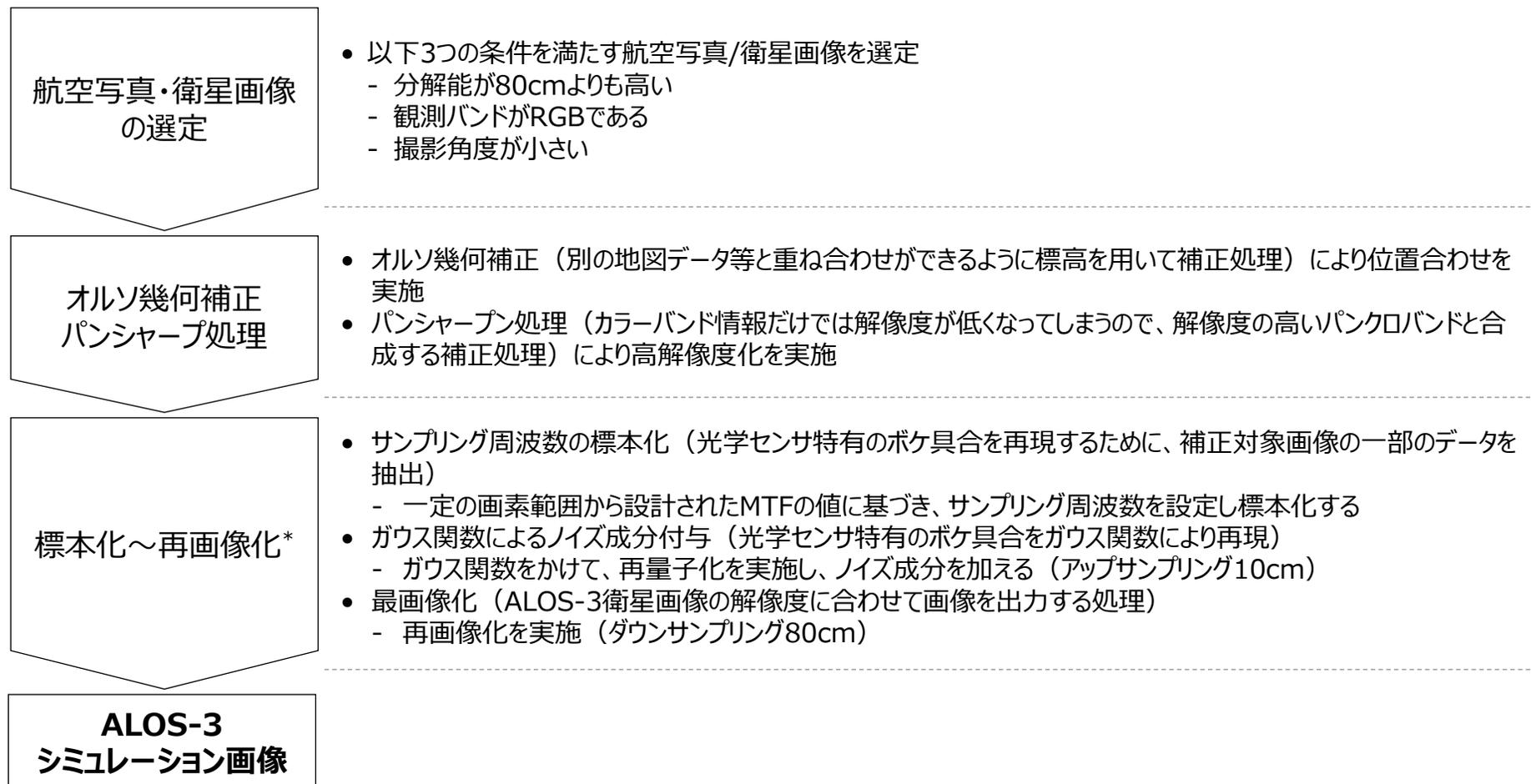
ALOS-3衛星シミュレーション画像を作成する上での影響は以下のとおり

項目		ASNARO-1衛星	ALOS-3衛星	影響の有無・加工処理等による差異の解消
軌道		太陽同期準回帰軌道 (高度約504km)	太陽同期準回帰軌道 (高度約670km)	<ul style="list-style-type: none"> <li>影響なし</li> </ul>
観測幅		観測幅10km	観測幅70km	<ul style="list-style-type: none"> <li>影響あり (実証としては影響なし)</li> <li>広域の時は、複数時期に撮影してカバー</li> </ul>
分解能 観測バンド	パンクロ	0.5m	0.8m	<ul style="list-style-type: none"> <li>影響はあるが、加工処理により解消</li> <li>リサンプリングにより解像度を合わせる</li> </ul>
	マルチ	2.0m (4バンド)	3.2m (6バンド)	
波長帯	パンクロ	450~860nm	520~760nm	<ul style="list-style-type: none"> <li>影響なし</li> <li>AIアルゴリズム内では、RGBカラーを使用する</li> </ul>
	コースタルブルー	400~450nm	400~450nm	
	ブルー (B)	450~520nm	450~500nm	
	グリーン (G)	520~600nm	520~600nm	
	レッド (R)	630~690nm	610~690nm	
	レッドエッジ	705~745nm	690~740nm	
	近赤外線	760~860nm	760~890nm	
MTF (シャープネス)	パンクロ	0.05以上	0.1以上	<ul style="list-style-type: none"> <li>影響はあるが、加工処理により解消</li> <li>MTF処理で合わせる</li> </ul>
	マルチ	0.05以上	0.2以上	
位置精度		CE90@25m	RMSE@10m (計画)	<ul style="list-style-type: none"> <li>影響はあるが、位置合わせ加工により解消</li> </ul>

# Ⅲ. 実証システム > 6. データ > ②データ処理

## ALOS-3衛星シミュレーション画像 | ASNARO-1衛星画像ベース (9/9)

ALOS-3衛星シミュレーション画像の作成フローは以下のとおり



\* RGB画像それぞれを実施する



# Ⅲ. 実証システム > 6. データ > ②データ処理

## 教師データ (1/2)

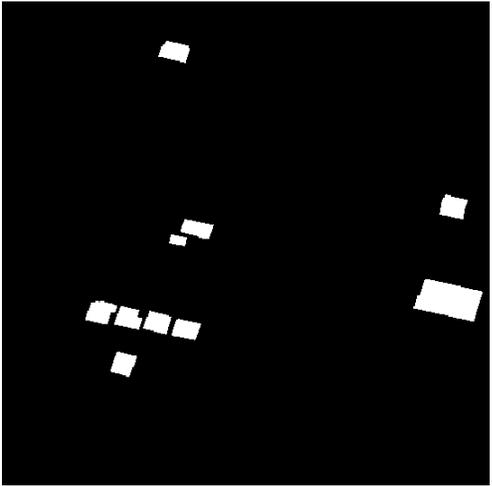
新旧2時期の画像データを比較して、建物の変化箇所を2値画像（白黒画像）として作成した画像データを、AIモデルの教師データ（学習用データや評価データ）として利用する

対象都市	撮影時期 (旧時期)	撮影時期 (新时期)	画像サイズ* (1枚あたり)	画像枚数 (区画数)	変化箇所数	概算容量 (PNG形式)
札幌市	2019年	2022年	256×256画素	266枚	755	173MB
郡山市	2016年	2022年		783枚	6,746	571MB
東京都23区	2018年	2020～2022年		1,073枚	4,554	588MB
横浜市	2019年	2020～2022年		124枚	204	75MB
沼津市	2017年	2019年		243枚	811	137MB
掛川市	2019年	2022年		324枚	797	197MB
大阪市	2020年	2022年		205枚	1,147	110MB
豊中市	2019年	2021年		189枚	1,285	114MB

\* 旧時期画像の画像サイズのみ512×512画素

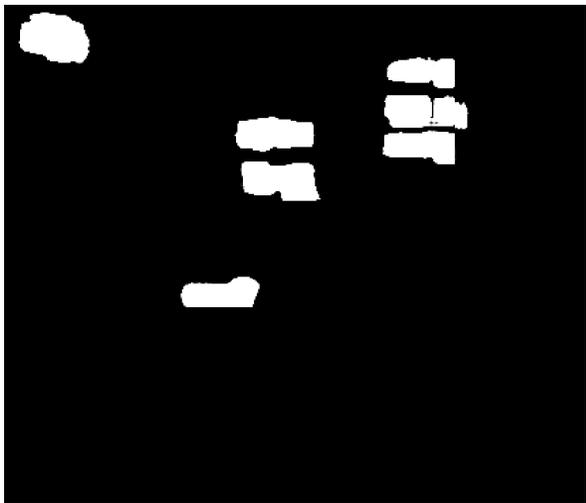
# Ⅲ. 実証システム > 6. データ > ②データ処理 教師データ (2/2)

変化検出用AIモデルの教師データのサンプルは以下のとおり

項目	旧時期画像 (PLATEAU公開済みの航空写真)	新时期画像 (ALOS-3衛星シミュレーション画像)	ラベル画像 (建物変化箇所ラベル画像)
画像イメージ			
画像形式	PNG形式	PNG形式	PNG形式
画像チャンネル	RGB 3チャンネル	RGB 3チャンネル	Grayscale 1チャンネル
画像サイズ [pixel]	512 x 512	256 x 256	256 x 256
画像ビット数	Unsigned 8bit	Unsigned 8bit	Unsigned 8bit
空間解像度 [cm/pixel]	40	80	80

# Ⅲ. 実証システム > 6. データ > ②データ処理 変化検出結果画像データ

建築物の変化検出結果画像のサンプルは以下のとおり

入出力	入力		出力
項目	旧時期画像 (PLATEAU公開済みの航空写真)	新時期画像 (ALOS-3衛星シミュレーション画像)	建築物の変化検出結果 (建築物の変化確率画像)
画像イメージ			
画像形式	PNG形式	PNG形式	TIFF形式
画像チャンネル	RGB 3チャンネル	RGB 3チャンネル	Grayscale 1チャンネル
画像ビット数	Unsigned 8bit	Unsigned 8bit	Float32
空間解像度 [cm/pixel]	40	80	40

# Ⅲ. 実証システム > 6. データ > ③ 出力データ 出力データ一覧

本システムの出力データは以下のとおり

出力データ	内容	データ形式
更新優先度マップ	<ul style="list-style-type: none"><li>メッシュで更新優先度を示したデータ</li><li>更新優先度は優先度の低い方から順に1~5の5段階で分類される</li></ul>	GeoJSON形式

# Ⅲ. 実証システム > 6. データ > ③出力データ 更新優先度マップ (1/6)

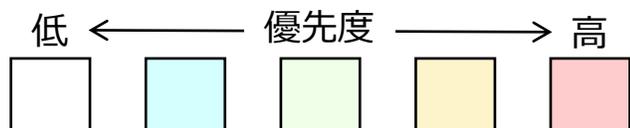
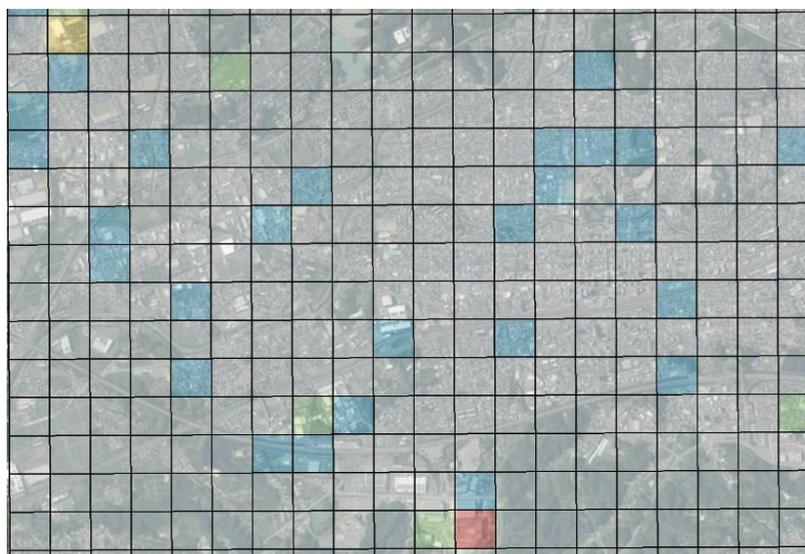
1つの対象都市につき、3つのメッシュサイズと2つの閾値区分があるため、6つの更新優先度マップを整備

対象都市	メッシュサイズ	閾値区分	整備面積	概算容量 (GeoJSON形式)
北海道札幌市	<ul style="list-style-type: none"> <li>3次メッシュ (約1km四方)</li> <li>4次メッシュ (約500m四方)</li> <li>5次メッシュ (約250m四方)</li> </ul>	<ul style="list-style-type: none"> <li>メッシュ単位</li> <li>LOD0単位</li> </ul>	100km <sup>2</sup>	2.5MB
福島県郡山市			100km <sup>2</sup>	2.5MB
東京都23区			470km <sup>2</sup>	11.8MB
神奈川県横浜市			160km <sup>2</sup>	4.0MB
静岡県沼津市			266km <sup>2</sup>	6.7MB
静岡県掛川市			265km <sup>2</sup>	6.7MB
大阪府大阪市			176km <sup>2</sup>	4.4MB
大阪府豊中市			36km <sup>2</sup>	0.9MB

# Ⅲ. 実証システム > 6. データ > ③ 出力データ 更新優先度マップ (2/6)

更新優先度は5段階に設定し、マップ上で赤色に表示される優先度5が最も更新優先度が高い

サンプルイメージ (5次メッシュ、メッシュ単位)



データ仕様

項目	値
フォーマット	GeoJSON形式
メッシュサイズ	<ul style="list-style-type: none"> <li>3次メッシュ (約1km四方)</li> <li>4次メッシュ (約500m四方)</li> <li>5次メッシュ (約250m四方)</li> </ul>
投影法	地理座標系 (EPSG : 4326)
優先度の分類数	5
分類方法	変化率を5段階で自然分類

フィールド定義

フィールド名	説明
更新優先度	<ul style="list-style-type: none"> <li>更新優先度を示す値</li> <li>1が最も優先度が低く、5が最も高い</li> </ul>
stroke*	<ul style="list-style-type: none"> <li>メッシュの外周線の色</li> <li>6桁の16進数カラーコードで記載</li> </ul>
stroke-width*	<ul style="list-style-type: none"> <li>メッシュの外周線の幅</li> <li>単位はピクセル</li> </ul>
stroke-opacity*	<ul style="list-style-type: none"> <li>メッシュの外周線の透明度</li> <li>0 (完全に透明) ~1 (不透過) の範囲で10進数小数で記載</li> </ul>
fill*	<ul style="list-style-type: none"> <li>メッシュの塗りつぶし色</li> <li>6桁の16進数カラーコードで記載</li> </ul>
fill-opacity*	<ul style="list-style-type: none"> <li>メッシュの塗りつぶし色の透明度</li> <li>0 (完全に透明) ~1 (不透過) の範囲で10進数小数で記載</li> </ul>

\* : PLATEAU VIEW上では非表示の項目

# Ⅲ. 実証システム > 6. データ > ③ 出力データ 更新優先度マップ (3/6)

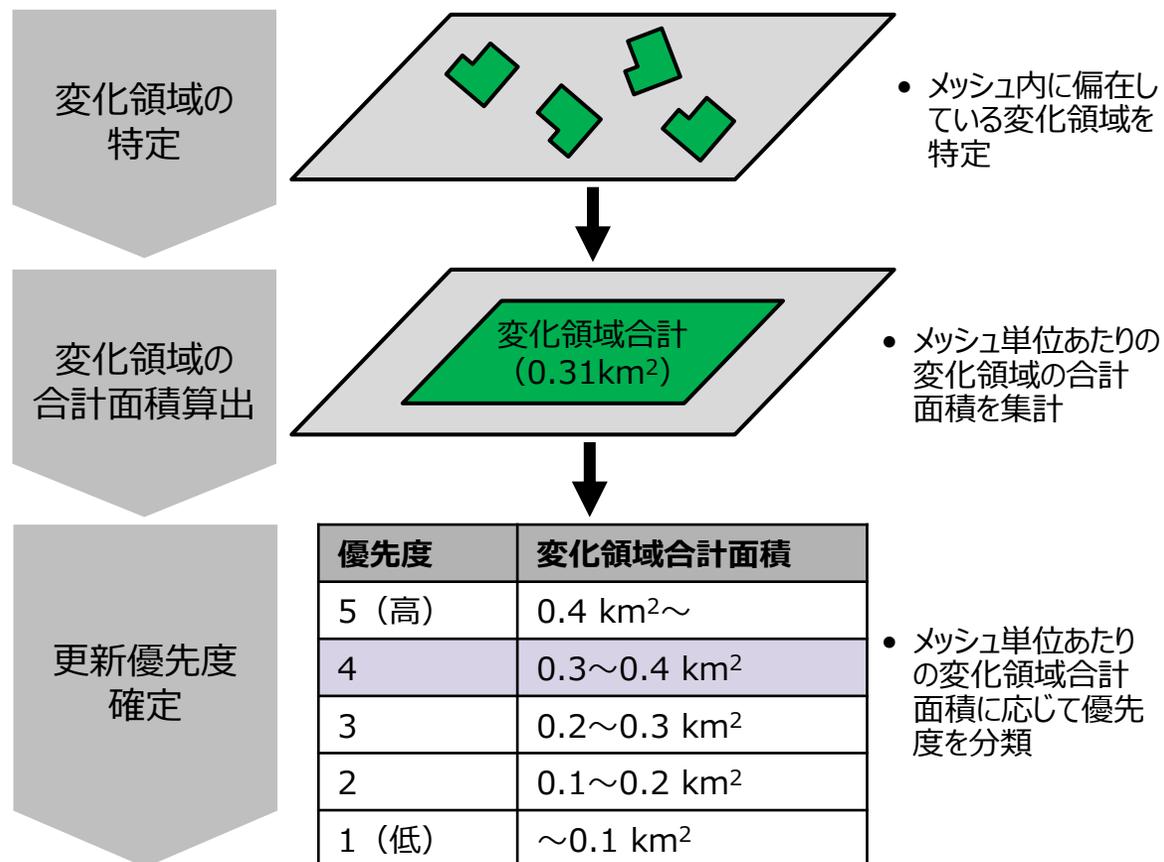
更新優先度マップ (GeoJSON形式) のサンプル

```
{
  "type": "FeatureCollection",
  "name": "sample",
  "crs": { "type": "name", "properties": { "name": "urn:ogc:def:crs:OGC:1.3:CRS84" } },
  "features": [
    { "type": "Feature", "properties": { "更新優先度": 4, "stroke": "#000000", "stroke-width": 2.0, "stroke-opacity": 1.0, "fill": "#fee480", "fill-opacity": 0.4 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ 138.8375, 35.11666666666667 ], [ 138.8375, 35.11875 ], [ 138.840625, 35.11875 ], [ 138.840625, 35.11666666666667 ], [ 138.8375, 35.11666666666667 ] ] ] } },
    { "type": "Feature", "properties": { "更新優先度": 2, "stroke": "#000000", "stroke-width": 2.0, "stroke-opacity": 1.0, "fill": "#98e1ff", "fill-opacity": 0.4 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ 138.840625, 35.11666666666667 ], [ 138.840625, 35.11875 ], [ 138.84375, 35.11875 ], [ 138.84375, 35.11666666666667 ], [ 138.840625, 35.11666666666667 ] ] ] } },
    { "type": "Feature", "properties": { "更新優先度": 5, "stroke": "#fe8080", "stroke-width": 2.0, "stroke-opacity": 1.0, "fill": "#fee480", "fill-opacity": 0.4 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ 138.8375, 35.11875 ], [ 138.8375, 35.12083333333333 ], [ 138.840625, 35.12083333333333 ], [ 138.840625, 35.11875 ], [ 138.8375, 35.11875 ] ] ] } },
    { "type": "Feature", "properties": { "更新優先度": 3, "stroke": "#daffc7", "stroke-width": 2.0, "stroke-opacity": 1.0, "fill": "#fee480", "fill-opacity": 0.4 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ 138.840625, 35.11875 ], [ 138.840625, 35.12083333333333 ], [ 138.84375, 35.12083333333333 ], [ 138.84375, 35.11875 ], [ 138.840625, 35.11875 ] ] ] } },
    { "type": "Feature", "properties": { "更新優先度": 1, "stroke": "#000000", "stroke-width": 2.0, "stroke-opacity": 1.0, "fill": "#ffffff", "fill-opacity": 0.4 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ 138.84375, 35.11666666666667 ], [ 138.84375, 35.11875 ], [ 138.846875, 35.11875 ], [ 138.846875, 35.11666666666667 ], [ 138.84375, 35.11666666666667 ] ] ] } },
    { "type": "Feature", "properties": { "更新優先度": 2, "stroke": "#000000", "stroke-width": 2.0, "stroke-opacity": 1.0, "fill": "#98e1ff", "fill-opacity": 0.4 }, "geometry": { "type": "Polygon", "coordinates": [ [ [ 138.84375, 35.11875 ], [ 138.84375, 35.12083333333333 ], [ 138.846875, 35.12083333333333 ], [ 138.846875, 35.11875 ], [ 138.84375, 35.11875 ] ] ] } }
  ]
}
```

# Ⅲ. 実証システム > 6. データ > ③ 出力データ 更新優先度マップ (4/6)

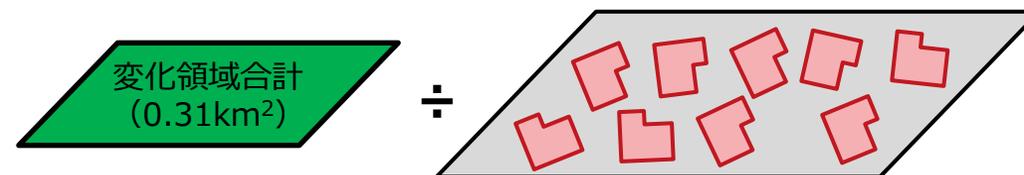
集計対象のメッシュに注目した場合、建物の変化領域はメッシュ内に偏在しているため、変化領域面積の合計をメッシュ単位で算出し、変化率を2種類の定義で計算する

更新優先度算出フロー



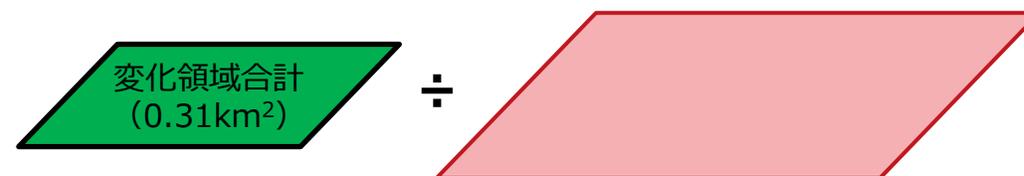
変化率の2種類の定義

建物LOD0面積単位)  $\text{変化率} = \frac{\text{変化があった面積}}{\text{メッシュ1つあたりの建物LOD0ポリゴンの面積}}$



- 郊外等で建物数が少ない地域において、面積の小さい変化でも優先度を高く評価できる
- 元々の建物数が少なすぎると変化率が過剰に評価される可能性がある

メッシュ面積単位)  $\text{変化率} = \frac{\text{変化があった面積}}{\text{メッシュ1つあたりの面積}}$



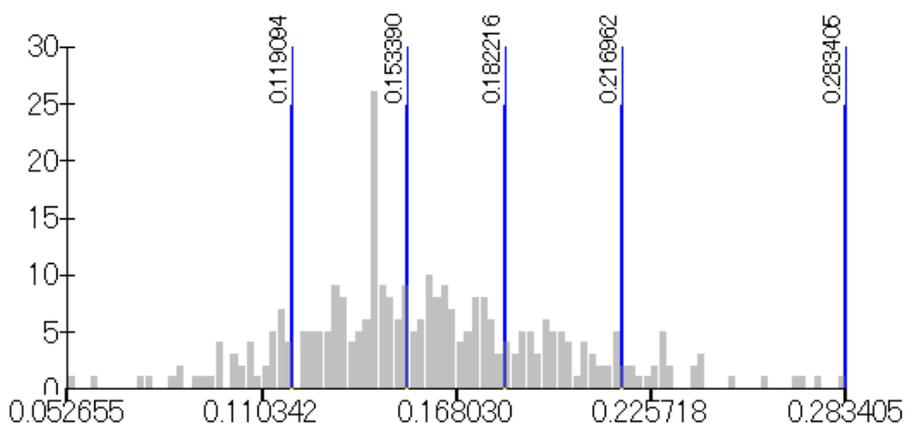
- 地域性によらず同じ基準で評価が可能であり、都市域において利便性が高い
- 郊外等で建物数が少ない地域の変化を過少に評価される可能性がある

# Ⅲ. 実証システム > 6. データ > ③出力データ 更新優先度マップ (5/6)

更新優先度については、変化率を対象都市ごとに自然分類することにより5段階に分類

## 自然分類の考え方

- 同じ分類クラス内の分散を小さく、異なる分類クラス間の分散を大きくするように閾値を設定する分類手法
- データの変化量が比較的大きい箇所に閾値が設定されるため、視覚的に判断しやすいことが特徴
- 右の表と下図は、変化率を対象にして、各メッシュの数値を5段階で自然分類を実施したイメージである (数値はダミー)

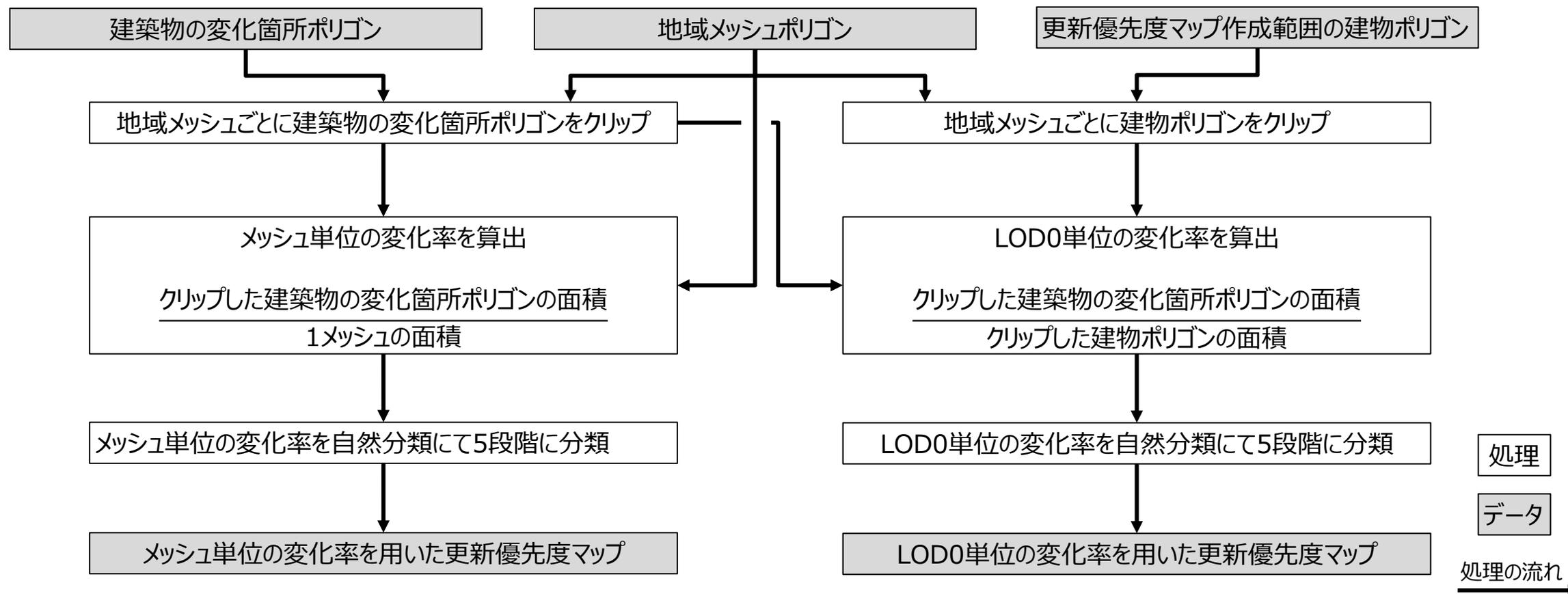


## 分類イメージ

メッシュID	変化率	分類
1	0.053	分類 1 (0~0.119)
2	0.062	
3	0.075	
42	0.119	分類 2 (0.119~0.153)
43	0.122	
44	0.123	
153	0.155	分類 3 (0.153~0.182)
154	0.155	
155	0.156	
233	0.184	分類 4 (0.182~0.216)
234	0.184	
235	0.184	
295	0.226	分類 5 (0.216~)
296	0.228	
297	0.228	

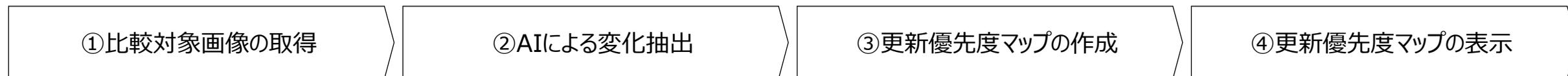
# Ⅲ. 実証システム > 6. データ > ③ 出力データ 更新優先度マップ (6/6)

更新優先度は以下の処理により付与する



# Ⅲ. 実証システム > 7. ユーザインタフェース 更新優先度マップ作成フロー (1/4)

更新優先度マップ作成のフローは以下のとおり

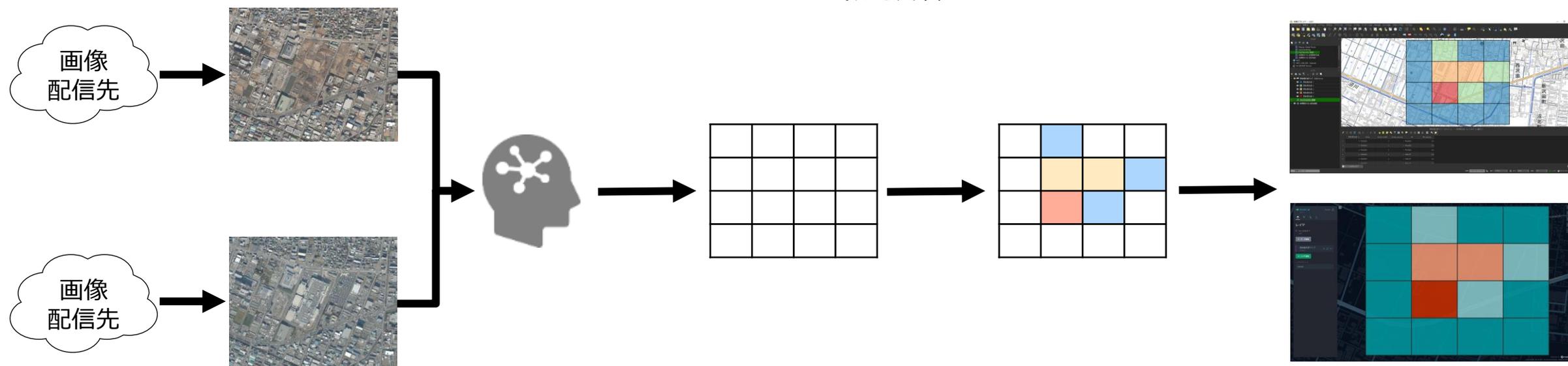


- 画像配信先から対象範囲の新旧画像をダウンロード

- ダウンロードした画像からAIを用いて変化抽出し、変化率を算出

- 変化率を自然分類で分類し更新優先度を算出
- 更新優先度に対して描画時のスタイルを反映

- PLATEAU VIEW等で表示



# Ⅲ. 実証システム > 7. ユーザインタフェース 更新優先度マップ作成フロー (2/4)

## ① 比較対象画像の取得

### STEP1: 旧時期航空写真のダウンロード

旧時期航空写真をダウンロードするには、以下のコマンドを実行する

```
python download_tile_image.py old
```

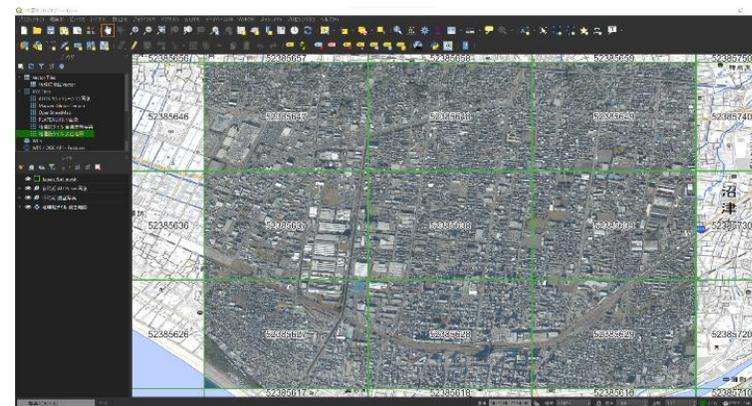
### STEP2: 新时期ALOS-3衛星シミュレーション画像のダウンロード

新时期航空写真をダウンロードするには、以下のコマンドを実行する

```
python download_tile_image.py new
```

### STEP3: ダウンロードした画像の確認

GISなどで表示して、指定された範囲がダウンロードされているか確認する



# Ⅲ. 実証システム > 7. ユーザインタフェース 更新優先度マップ作成フロー (3/4)

## ②AIによる変化抽出

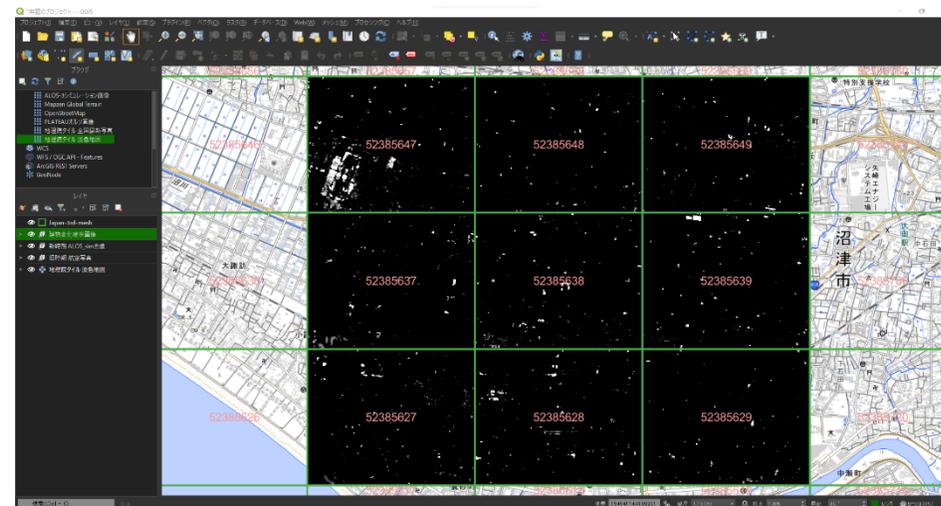
### STEP1: 建築物の変化確率画像の生成

以下のコマンドを実行して、旧時期航空写真と新时期ALOS-3シミュレーション画像から建築物の変化確率画像を生成する

```
python generate_probmap.py
```

### STEP2: 生成した建築物の変化確率画像の確認

保存先のフォルダの中身をlsコマンドで確認する  
 もしくは、GISで生成した建築物の変化確率画像を表示し、指定した範囲に建築物の変化確率画像が生成されていることを確認する



# Ⅲ. 実証システム > 7. ユーザインタフェース 更新優先度マップ作成フロー (4/4)

## ③更新優先度マップの作成 / ④更新優先度マップの表示

### STEP1: 更新優先度マップの生成

以下のコマンドを実行して、更新優先度マップのメッシュを生成する

```
python create_mesh.py
```

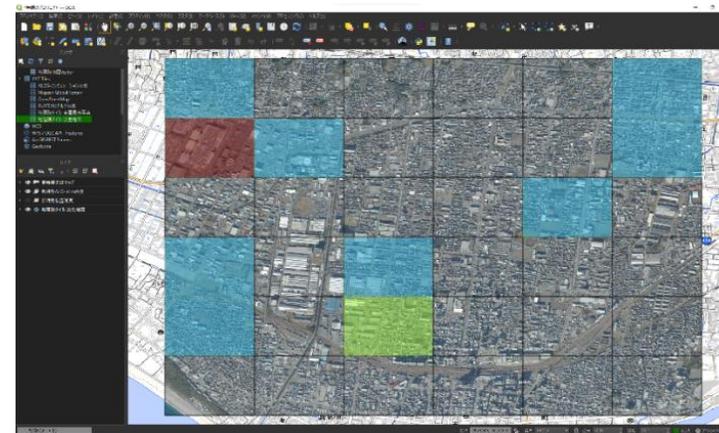
### STEP2: 更新優先度マップのスタイリング

以下のコマンドを実行して、更新優先度に応じて色付けされたメッシュを生成する

```
python style_mesh.py
```

### STEP3: 生成した更新優先度マップの表示

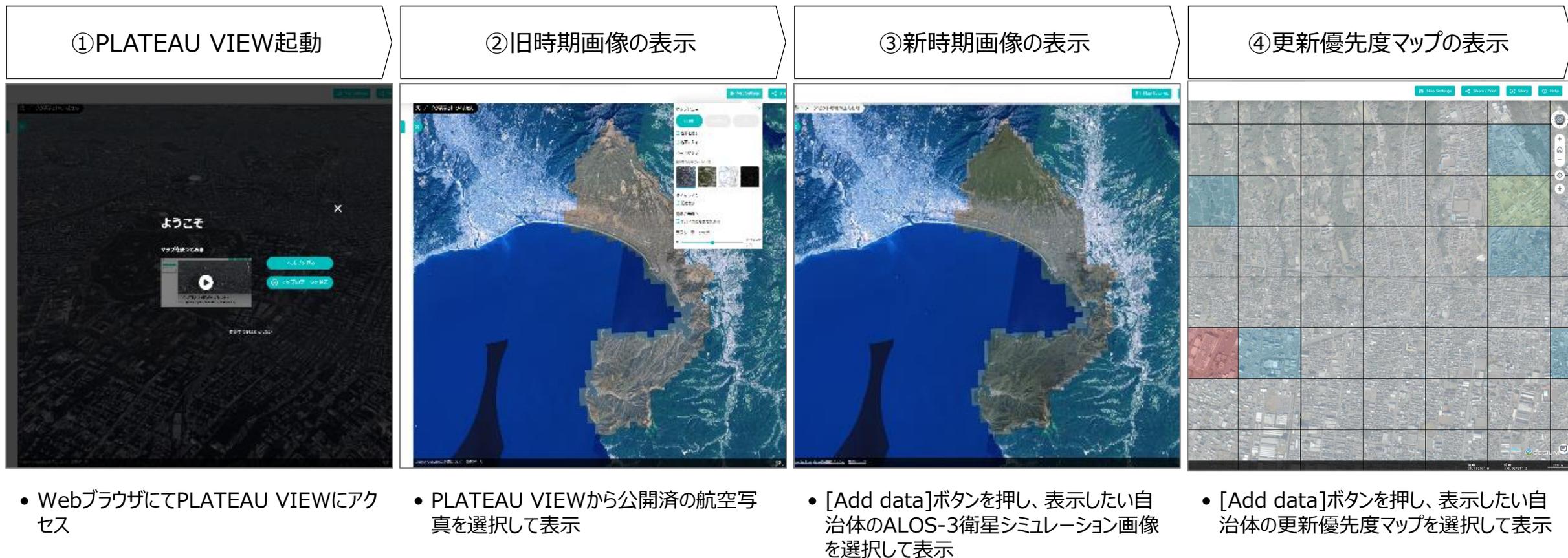
GISで更新優先度マップを表示し、指定した範囲が作成されていることを確認する  
 PLATEAU VIEWでの表示方法は次ページ以降を参照



# Ⅲ. 実証システム > 7. ユーザインタフェース

## PLATEAU VIEWによる更新優先度マップ確認フロー

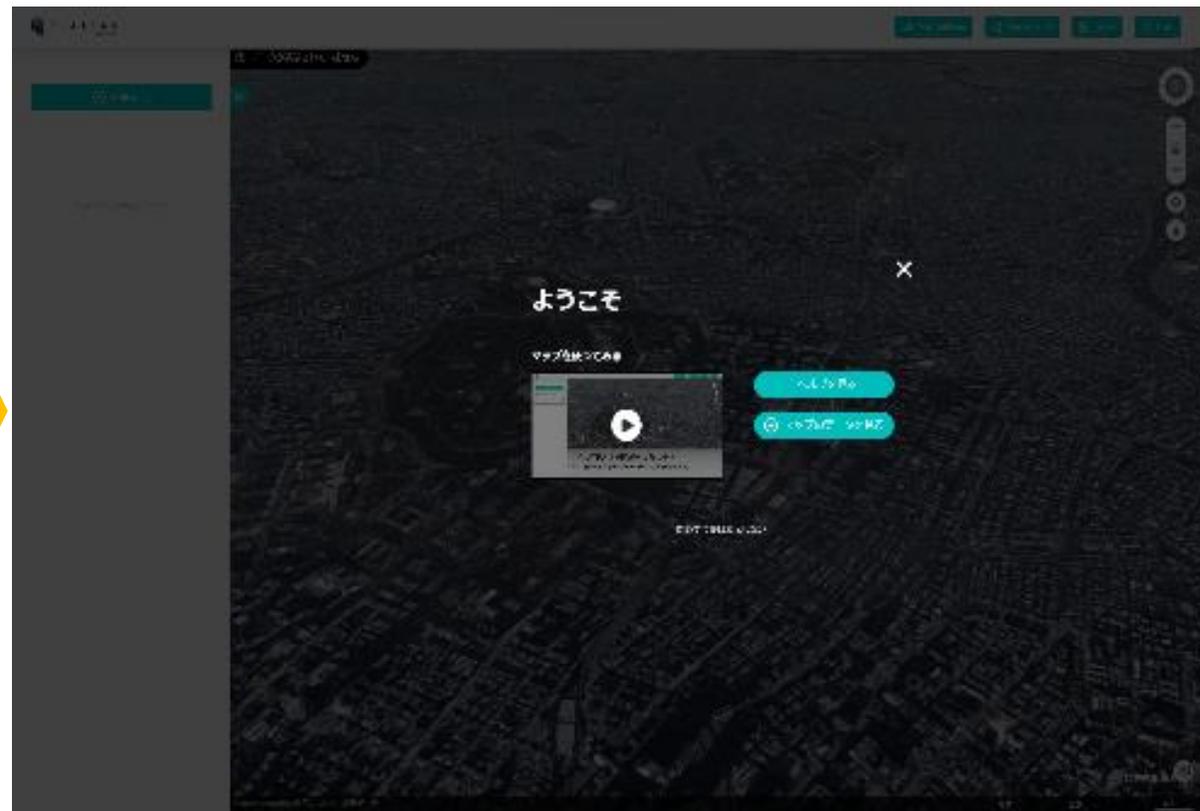
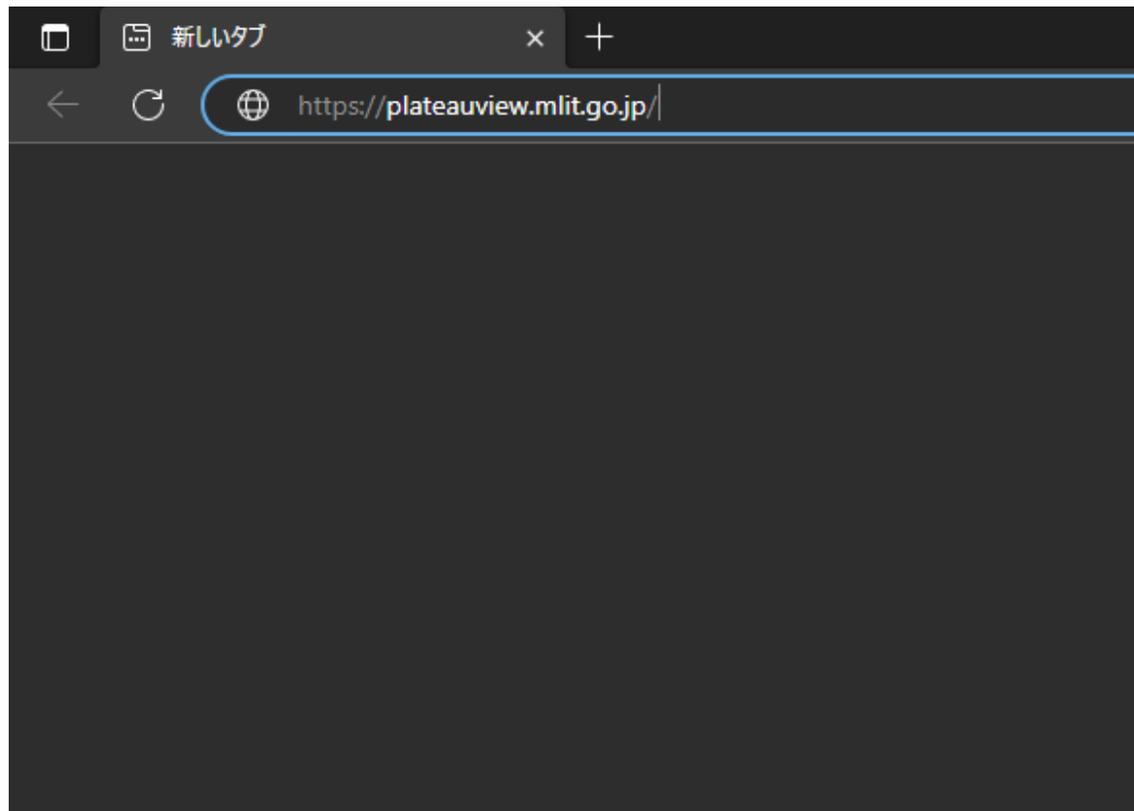
PLATEAU VIEWによる更新優先度マップの確認フローは以下のとおり



## Ⅲ. 実証システム > 7. ユーザインタフェース

### ① PLATEAU VIEW起動

WebブラウザにてPLATEAU VIEWのURLを入力してアクセスする

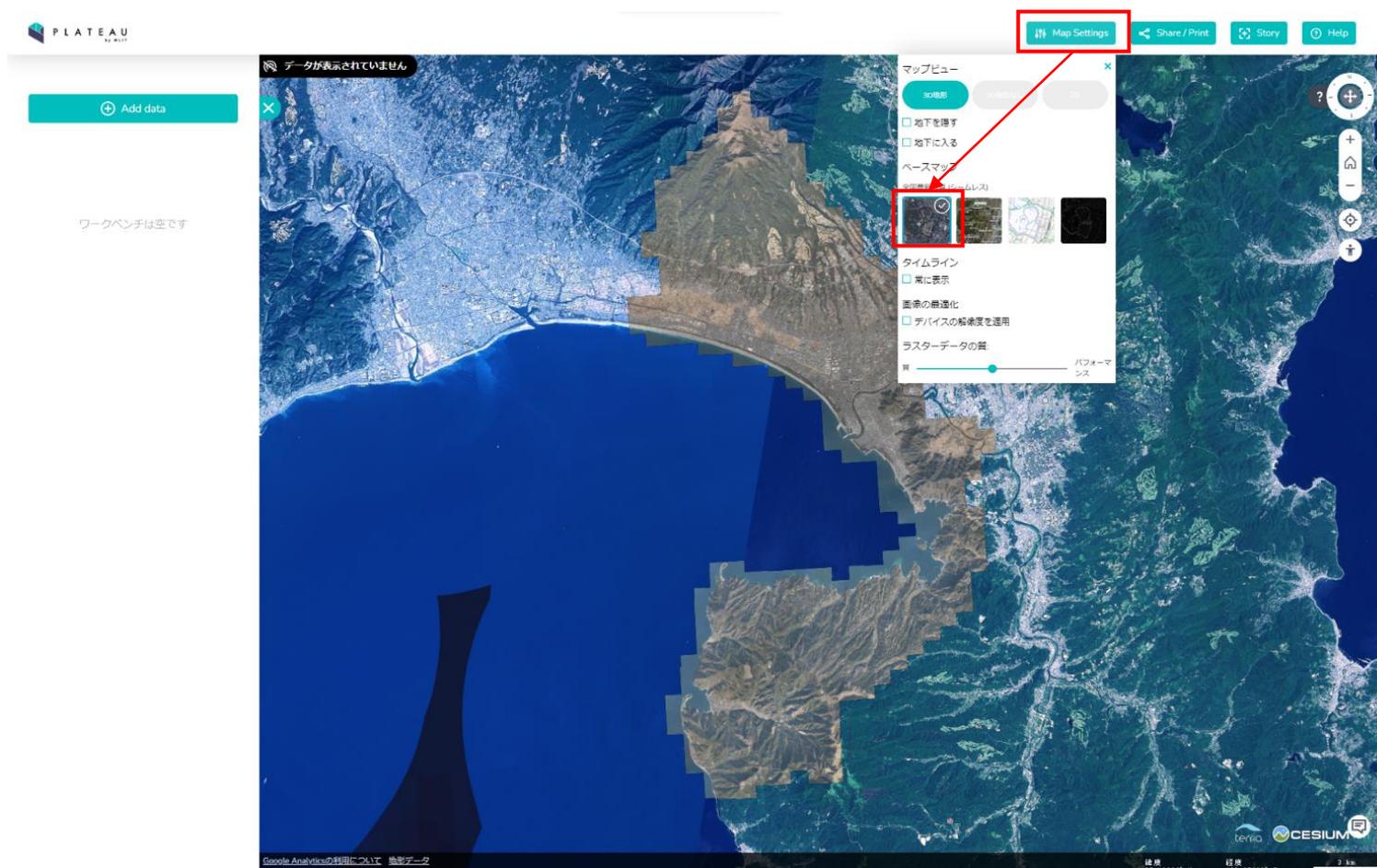




## Ⅲ. 実証システム > 7. ユーザインタフェース

### ② 旧時期画像の表示

PLATEAU VIEW上の[Map Setting]ボタンから「全国最新写真（シームレス）」を選択し、ベースマップから公開済の航空写真を選択して表示する

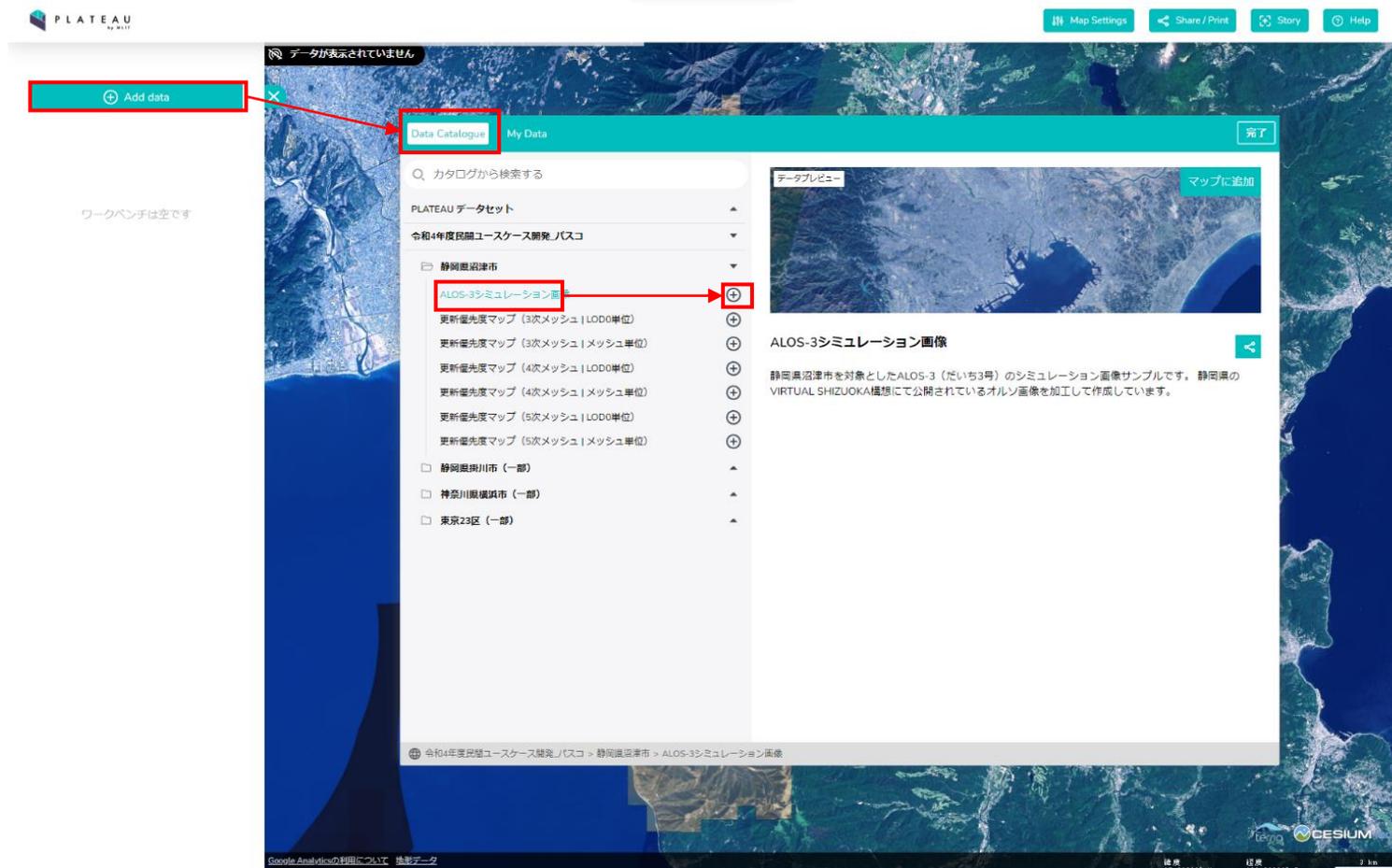




# Ⅲ. 実証システム > 7. ユーザインタフェース

## ③ 新時期画像の表示 (1/2)

[Add data]ボタンから、表示したい自治体のALOS-3衛星シミュレーション画像（オープンデータ公開後）を選択して表示する

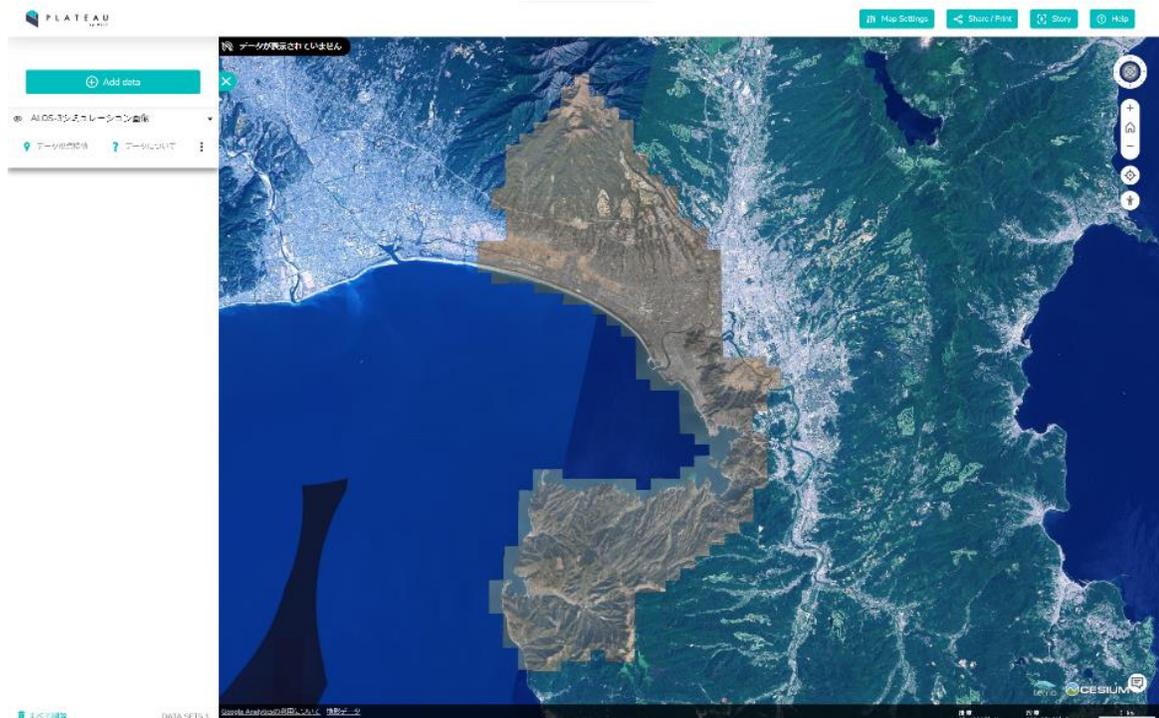


# Ⅲ. 実証システム > 7. ユーザインタフェース

## ③ 新時期画像の表示 (2/2)

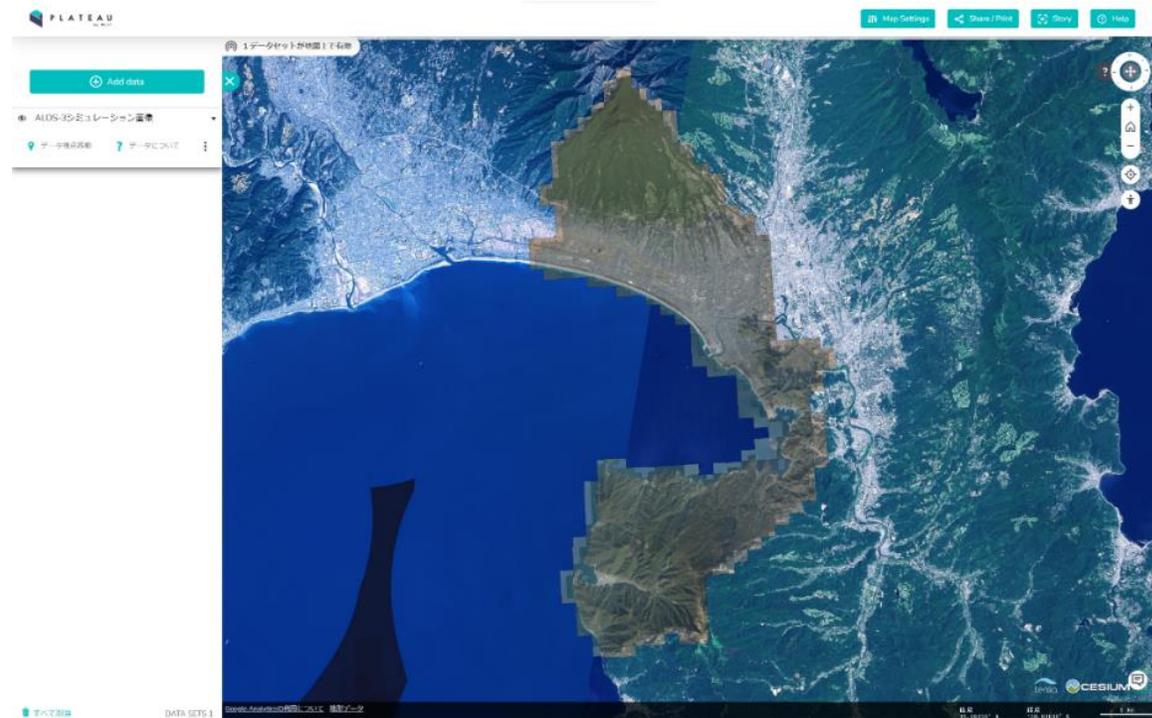
ALOS-3衛星シミュレーション画像のレイヤを表示/非表示に切り替えることにより、旧時期と新時期画像の表示切替えが可能

ALOS-3シミュレーション画像非表示 (旧時期画像)



 ALOS-3シミュレーション画像

ALOS-3シミュレーション画像表示 (新時期画像)

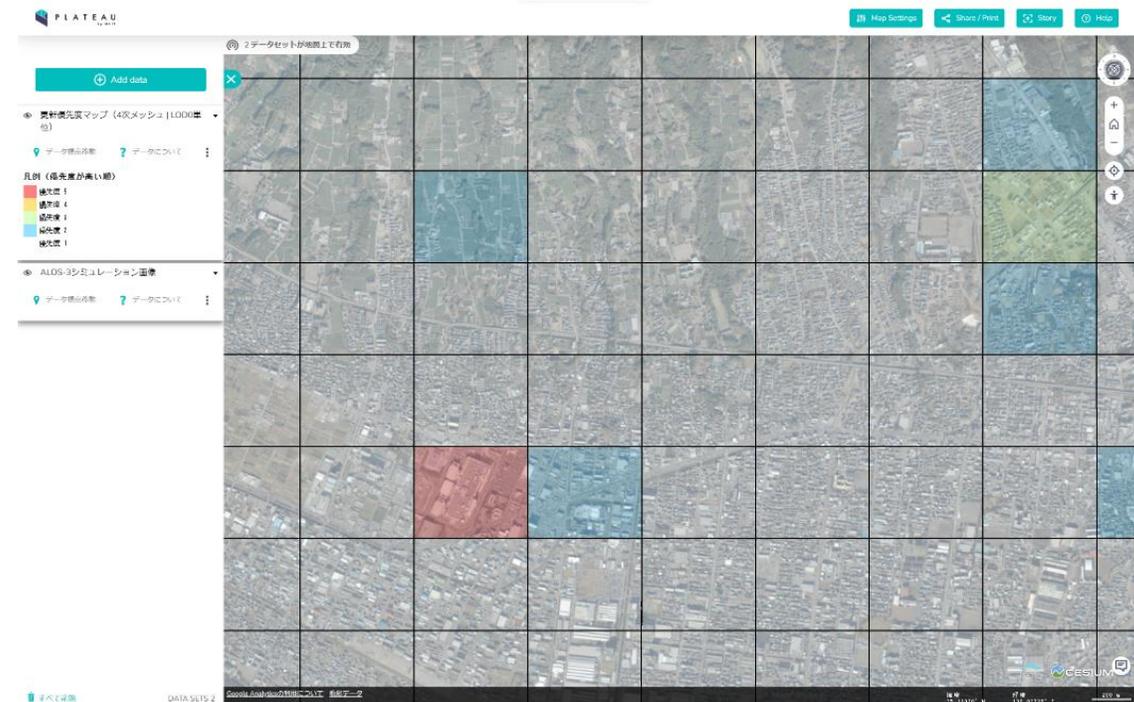
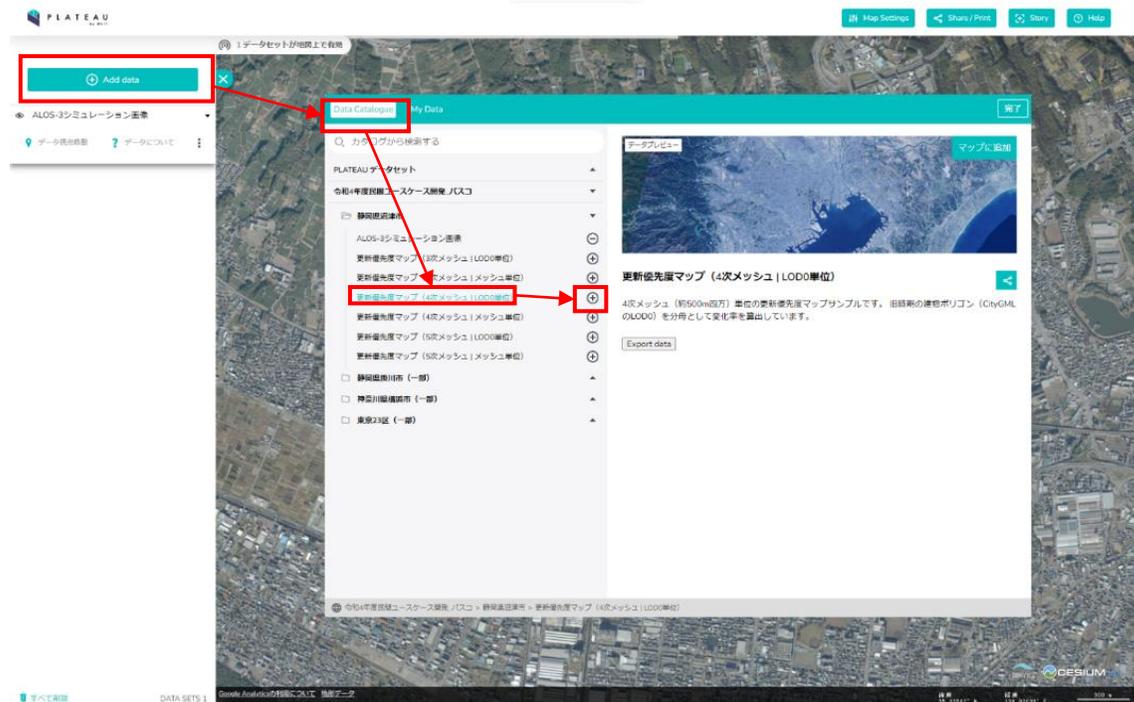


 ALOS-3シミュレーション画像

# Ⅲ. 実証システム > 7. ユーザインタフェース

## ④ 更新優先度マップの表示

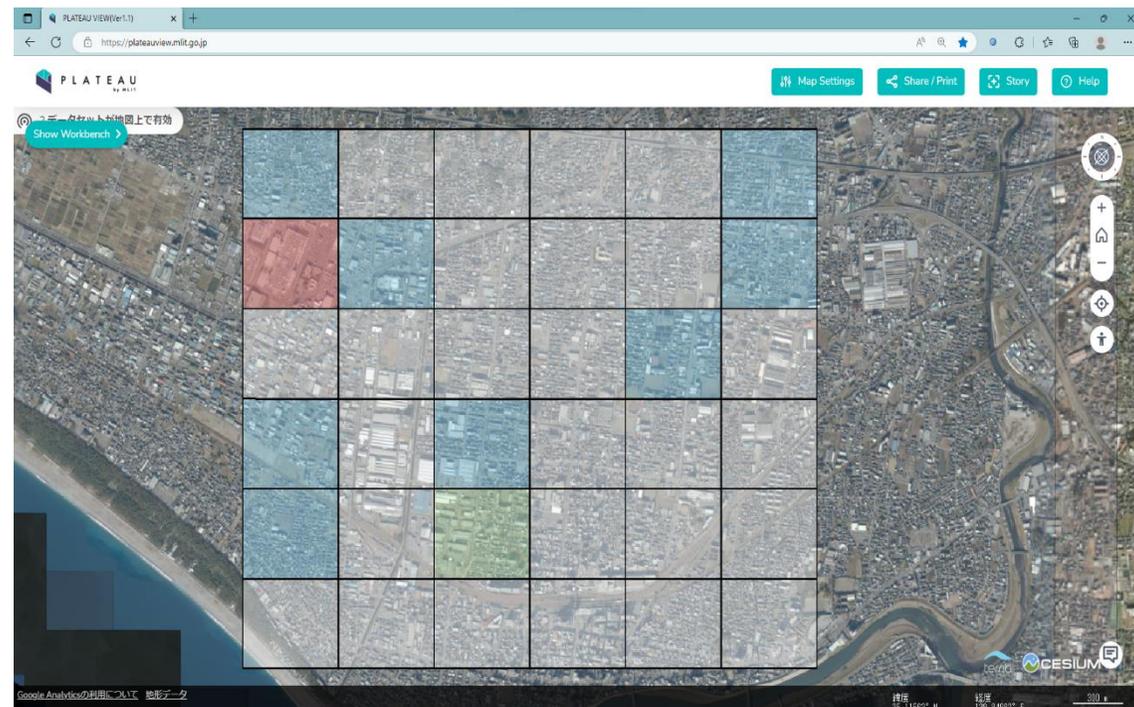
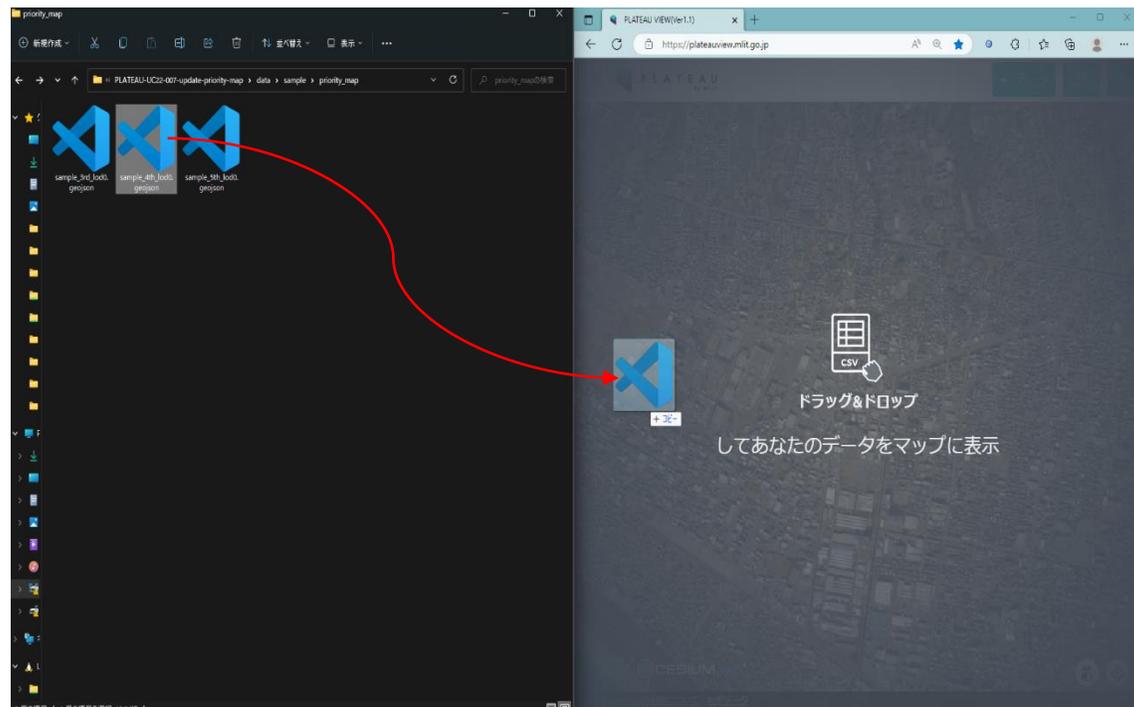
[Add data]ボタンを押し、表示したい自治体の更新優先度マップを選択して表示する



### Ⅲ. 実証システム > 7. ユーザインタフェース

## ④ 更新優先度マップの表示（ユーザ作成データ）

PLATEAU VIEWの画面にユーザ自身作成が作成した更新優先度マップをドラッグ&ドロップして表示



# 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

システムテストの結果は以下のとおり

試験項目	確認内容	結果
本実証内でのデモ環境の性能要件	<b>更新優先度マップの表示：1自治体あたり10秒以内</b> <ul style="list-style-type: none"> <li>試験対象エリア：静岡県沼津市</li> <li>動作環境               <ul style="list-style-type: none"> <li>CPU：Intel Core i5-8250U</li> <li>CPUコア数：4</li> <li>CPUクロック周波数：1.80GHz</li> <li>メモリ容量：8.0GB</li> <li>ウェブブラウザ：Microsoft Edge バージョン 109.0.1518.61</li> </ul> </li> </ul>	<b>OK</b> <ul style="list-style-type: none"> <li>ALOS-3シミュレーション画像表示               <ul style="list-style-type: none"> <li>6.7 [sec]</li> </ul> </li> <li>更新優先度マップ表示               <ul style="list-style-type: none"> <li>3次メッシュ：2.9 [sec]</li> <li>4次メッシュ：3.1 [sec]</li> <li>5次メッシュ：3.4 [sec]</li> </ul> </li> </ul>
各プログラムの性能要件	<b>更新優先度マップの出力：3次メッシュ（1km<sup>2</sup>）あたり10分以内（判読処理対象の画像データ取得にかかる処理時間を除く）</b> <ul style="list-style-type: none"> <li>動作環境               <ul style="list-style-type: none"> <li>仮想CPUコア数：4</li> <li>メモリ容量：60GB</li> <li>GPU：NVIDIA V100</li> </ul> </li> <li>入力データ               <ul style="list-style-type: none"> <li>旧時期3次メッシュ画像（3136 x 2092[pix]）</li> <li>新时期3次メッシュ画像（1568 x 1046[pix]）</li> </ul> </li> </ul>	<b>OK</b> <ul style="list-style-type: none"> <li>変化検出結果生成               <ul style="list-style-type: none"> <li>16.27[sec]</li> </ul> </li> <li>更新優先度マップ               <ul style="list-style-type: none"> <li>3次メッシュ生成：5.43[sec]</li> <li>4次メッシュ生成：5.66[sec]</li> <li>5次メッシュ生成：5.95[sec]</li> </ul> </li> </ul>

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. 更新優先度マップにおけるAIモデルの精度検証 > ① 検証内容 検証内容

精度検証として、閾値による分類精度と2分類での建物変化の検出精度に分けて検証を行う

## 検証の概要

## KPI

### 検証目的

- 更新優先度マップ全体の分類精度を検証する
- 建物変化の有無の観点で再現率を評価し、KPIの達成可否を検証する

### 検証内容およびその方法

- ① 更新優先度マップにおける閾値分類結果の精度
  - AI判読結果と教師データ双方から作成した更新優先度マップにおいて、以下の3つの観点から更新優先度の分類が正しいか評価を実施する
    - **メッシュサイズ別:** メッシュサイズごとの精度評価を検証
      - メッシュサイズを変化させ、そのマイクロ平均F1値を比較
    - **地域別:** 地域に関係なく同等の精度があるかを検証
      - 全地域および郡山市、東京23区、大阪市のマイクロ平均F1値を比較
    - **超解像処理:** 超解像処理による精度向上を検証
      - 超解像処理の有無以外の条件を変化させずマイクロ平均F1値を比較
- ② 建物変化の検出精度
  - 建物変化を検出できるかという観点で、建物変化が一番少ない優先度1とそれ以外の2~5の2分類において再現率を比較する
  - 2分類とすることで、建物変化検出の精度をより正確に評価可能である

No.	評価指標・KPI	目標値	目標値の設定理由
1	マイクロ平均F1値	75%	<ul style="list-style-type: none"> <li>• 閾値分類数を5分類に設定したので、個々の閾値項目の精度ではなく、分類全体の精度を評価する</li> <li>• 閾値の境界に近い数値の誤分類もある程度は許容される想定として、目標値を設定した</li> </ul>
2	再現率	90%	<ul style="list-style-type: none"> <li>• 建物変化の有無を評価する観点で、見逃しを少なくすることが重視されるので、チューニング等により実現可能性がある目標値を設定した</li> </ul>
3	適合率	80%	<ul style="list-style-type: none"> <li>• 建物変化の有無を評価する観点では、再現率が重要視される</li> <li>• ただし、適合率が低すぎると結果を利用する際にユーザー側での再確認の負荷が高まるため、チューニング等により実現可能性がある目標値を設定した</li> </ul>

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ① 検証内容

## AI判読結果の評価指標一覧

本検証では、閾値による5分類の評価はマイクロ平均F1値を使い、建物変化検出の有無（2分類）の評価は再現率・適合率を利用する

評価指標	詳細	本ユースケースでの利用方法
マイクロ平均F1値	<ul style="list-style-type: none"> <li>多クラス分類において、特定のデータの偏りを考慮せずに、全ての分類クラスの結果を評価する平均算出法であるマイクロ平均を用いた評価手法</li> <li>分類クラスを超えて全体の精度を知ることが可能</li> </ul>	<ul style="list-style-type: none"> <li>更新優先度マップは5分類となり、各クラスでは少数クラスもあるため、分類全体の精度評価することを重視してマイクロ平均F1値での評価を行う</li> </ul>
マクロ平均F1値	<ul style="list-style-type: none"> <li>多クラス分類において、特定のデータの偏りの有無によらず個々の分類クラス毎の精度から算術平均をベースに評価する手法</li> <li>少数クラスがある場合は、少数クラスの異常値の影響を受けやすい</li> </ul>	
再現率	<ul style="list-style-type: none"> <li>判読結果において、見逃しをしていない程度を把握するための指標</li> <li>正解データのうち、AIが正しく判読した割合で表される</li> <li>数値が高いほど、見逃しが少ないことを表す</li> </ul>	<ul style="list-style-type: none"> <li>建物変化検出有無の検証において、双方を利用して評価を行う                     <ul style="list-style-type: none"> <li>- 再現率：建物変化の見逃し（変化している建物を検出できない）の精度の評価に利用</li> <li>- 適合率：建物変化の誤検出（変化していない建物まで検出する）の程度の評価に利用</li> </ul> </li> </ul>
適合率	<ul style="list-style-type: none"> <li>判読結果において、誤検出をしていない程度を把握するための指標</li> <li>AI判読結果のうち、AIが正しく判読した割合で表される</li> <li>数値が高いほど、誤検出が少ないことを表す</li> </ul>	

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ① 検証内容

## 評価指標の算出方法 (1/2)

再現率は判読結果のどれだけ正解できたか（見逃しがないか）の指標、適合率は誤検出の少なさ（検出したものは正解だったのか）の指標として算出される

### 再現率の算出方法

$$\text{再現率}[\%] = \frac{\text{AI判読結果が正解した数}}{\text{正解データ数}} \times 100$$

### 適合率の算出方法

$$\text{適合率}[\%] = \frac{\text{AI判読結果が正解した数}}{\text{AI判読結果数}} \times 100$$

例) 以下の結果における分類1の再現率

		AI判読結果	
		分類1	分類2
正解データ	分類1	2,046	326
	分類2	104	496

分類1における正解データ数は

$$2046 + 326 = 2372$$

そのうち、分類1と正しく分類しているのは2046

よって、再現率は

$$\text{再現率}[\%] = \frac{\text{AI判読結果が正解した数}}{\text{正解データ数}} = \frac{2046}{2372} \times 100 \approx 86.3\%$$

例) 以下の結果における分類1の適合率

		AI判読結果	
		分類1	分類2
正解データ	分類1	2,046	326
	分類2	104	496

分類1におけるAI判読結果数は

$$2046 + 104 = 2150$$

そのうち、分類1と正しく分類しているのは2046

よって、適合率は

$$\text{適合率}[\%] = \frac{\text{AI判読結果が正解した数}}{\text{AI判読結果数}} = \frac{2046}{2150} \times 100 \approx 95.2\%$$

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ① 検証内容

## 評価指標の算出方法 (2/2)

マイクロ平均F1値は、マイクロ平均法による適合率と再現率の調和平均から算出される

### マイクロ平均F1値の算出方法

- クラスごとに計算せずに結果全体から算出する

例) 以下の結果におけるマイクロ平均F1値

		AI判読結果		
		分類1	分類2	分類3
正解データ	分類1	61	8	0
	分類2	7	25	4
	分類3	2	9	13

- 表の黄色部分が正解となるので、マイクロ平均を用いた
- 適合率と再現率、その調和平均であるF1値は以下ようになる

$$\begin{aligned} \text{マイクロ平均適合率mp}[\%]^* \\ &= \frac{61+25+13}{(61+9)+(25+17)+(13+14)} \times 100 \approx 71.2 \end{aligned}$$

$$\begin{aligned} \text{マイクロ平均再現率mf}[\%]^* \\ &= \frac{61+25+13}{(61+8)+(25+11)+(13+11)} \times 100 \approx 76.7 \end{aligned}$$

- マイクロ平均F1値はmpとmfの調和平均  $(2/(1/mp+1/mf))$  であるため、73.8%となる
- \*丸括弧内の第1項が正解の数、第2項が過抽出もしくは抽出漏れの数を示す

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ① 検証内容 検証シナリオ

メッシュサイズ250m、超解像有、5分類、マイクロ平均F1値での評価を基本として、各条件を変更した10のシナリオで検証を行う

条件変更項目

No.	検証内容	エリア	メッシュサイズ	超解像処理	優先度分類数	評価手法
—	ベースケース	全地域	250m	有	5	マイクロ平均F1値
1	メッシュサイズ別		1km			
2			500m			
3			250m			
4	地域的な汎化性能（地域別）	全地域				
5		福島県郡山市				
6		東京23区				
7		大阪府大阪市				
8	超解像処理の分類精度検証			無		
9				有		
10	建物変化検出の分類精度検証			有	2	再現率、適合率

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 結果サマリ



PLATEAU  
by MLIT

メッシュサイズ細分化のマイクロ平均F1値の向上効果、画像精度に起因すると思われる各地域別の特徴、超解像処理による精度向上が確認できた

検証内容	評価指標・KPI	達成度・結果 (青字：達成、赤字：未達成)		示唆
		項目	結果	
メッシュサイズ別 (1km、500m、250m)	マイクロ平均F1値 (75%以上)	1km	62.05%	<ul style="list-style-type: none"> <li>メッシュサイズを細かくするにつれマイクロ平均F1値は向上する傾向が見られた</li> <li>メッシュサイズ250mにおいて、マイクロ平均F1値がKPIの75%を超えた</li> </ul>
		500m	70.92%	
		250m	84.08%	
地域的な汎化性能 (全地域、郡山市、 東京23区、大阪市)	マイクロ平均F1値 (75%以上)	全地域	84.08%	<ul style="list-style-type: none"> <li>3つの各地域において、マイクロ平均F1値がKPIの90%を超えた</li> <li>大阪市の精度が低い原因として、大阪市の旧時期航空写真は空間解像度が1mになるので、他の地域の画像より条件が悪かったことが考えられる</li> </ul>
		郡山市	96.18%	
		東京23区	81.85%	
		大阪市	75.00%	
超解像処理 (超解像処理の有無)	マイクロ平均F1値 (75%以上)	超解像処理なし	82.60%	<ul style="list-style-type: none"> <li>超解像処理を加えることで全体的な精度が向上した</li> <li>超解像処理を加えることで、更新優先度2～5のメッシュの再現率が向上した</li> </ul>
		超解像処理あり	84.08%	
建物変化の有無 (優先度2分類)	<ul style="list-style-type: none"> <li>再現率 (90%以上)</li> <li>適合率 (80%以上)</li> </ul>	2分類	再現率 : 87.02% 適合率 : 80.81%	<ul style="list-style-type: none"> <li>KPIとしていた再現率90%に至らなかったが、適合率は80%以上を達成したため、改善は必要なものの実用面ではユーザーが利用しやすい(検出漏れが少ないが誤検出あり → ユーザーは検出されたもののみをチェックすればよい) と考えられる</li> </ul>

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 メッシュサイズ別

メッシュサイズを細かくするにつれマイクロ平均F1値が向上する傾向が見られ、メッシュサイズ250mではマイクロ平均F1値がKPIの75%を超え84.08%となった

メッシュサイズ1km

評価指標：マイクロ平均F1値=62.05% (KPIの75%を達成できず)

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	33	17	3	2	4
	優先度2	7	48	15	3	1
	優先度3	1	4	20	6	1
	優先度4	0	0	2	7	5
	優先度5	0	0	1	2	13

メッシュサイズ500m

評価指標：マイクロ平均F1値=70.92% (KPIの75%を達成できず)

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	322	68	7	2	21
	優先度2	25	156	38	2	3
	優先度3	2	15	36	19	3
	優先度4	1	0	2	10	11
	優先度5	0	0	1	1	15

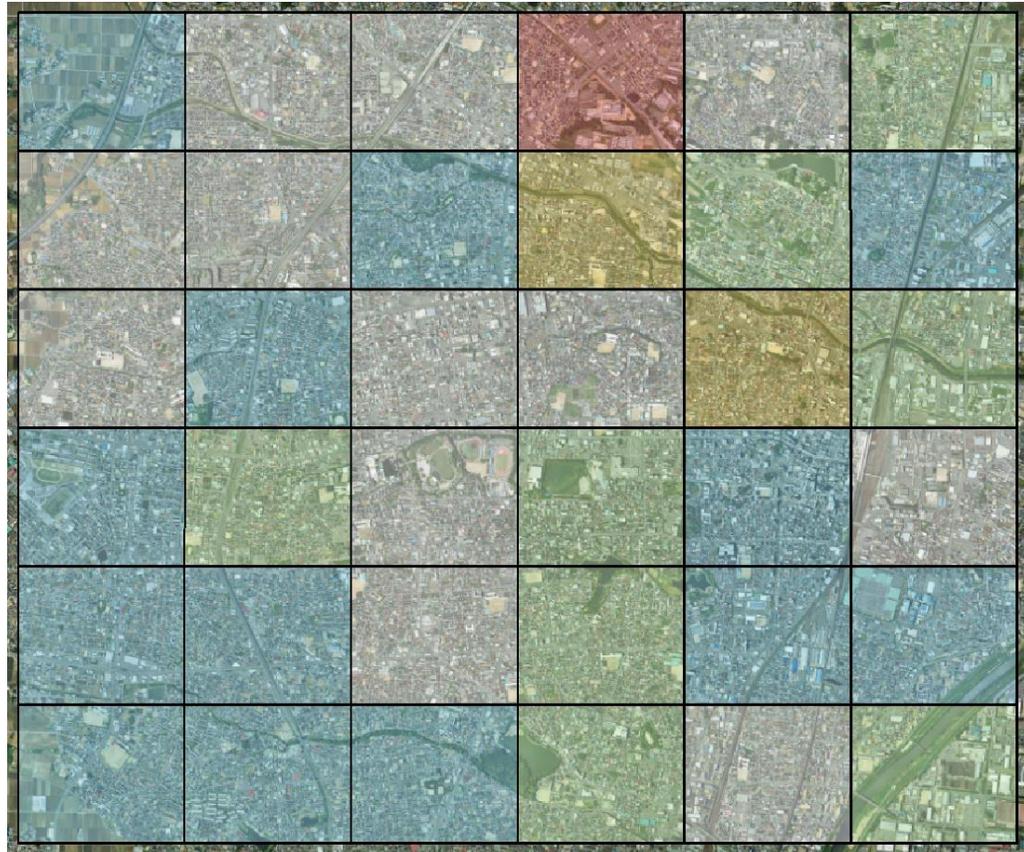
メッシュサイズ250m

評価指標：マイクロ平均F1値=84.08% (KPIの75%を達成)

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	2,046	270	25	5	26
	優先度2	99	290	65	4	2
	優先度3	2	18	50	17	6
	優先度4	2	3	2	13	8
	優先度5	1	1	3	0	14

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 メッシュサイズ別

正解 (メッシュサイズ1km)

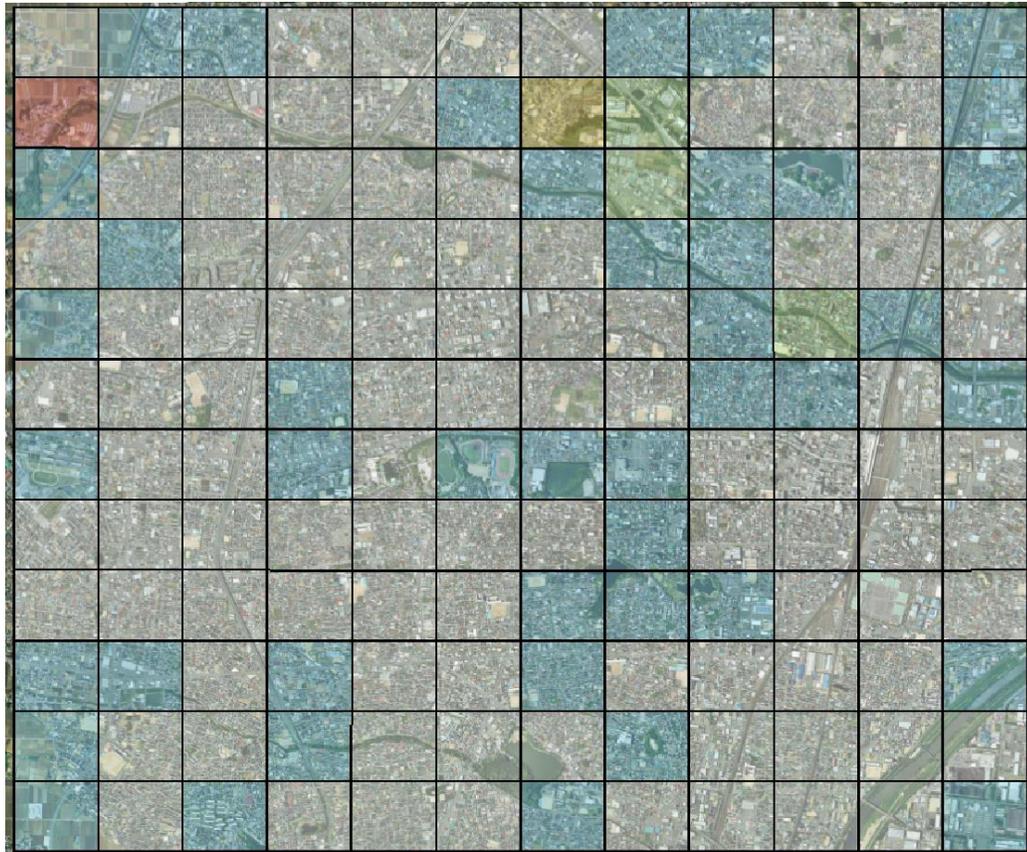


判読結果 (メッシュサイズ1km)

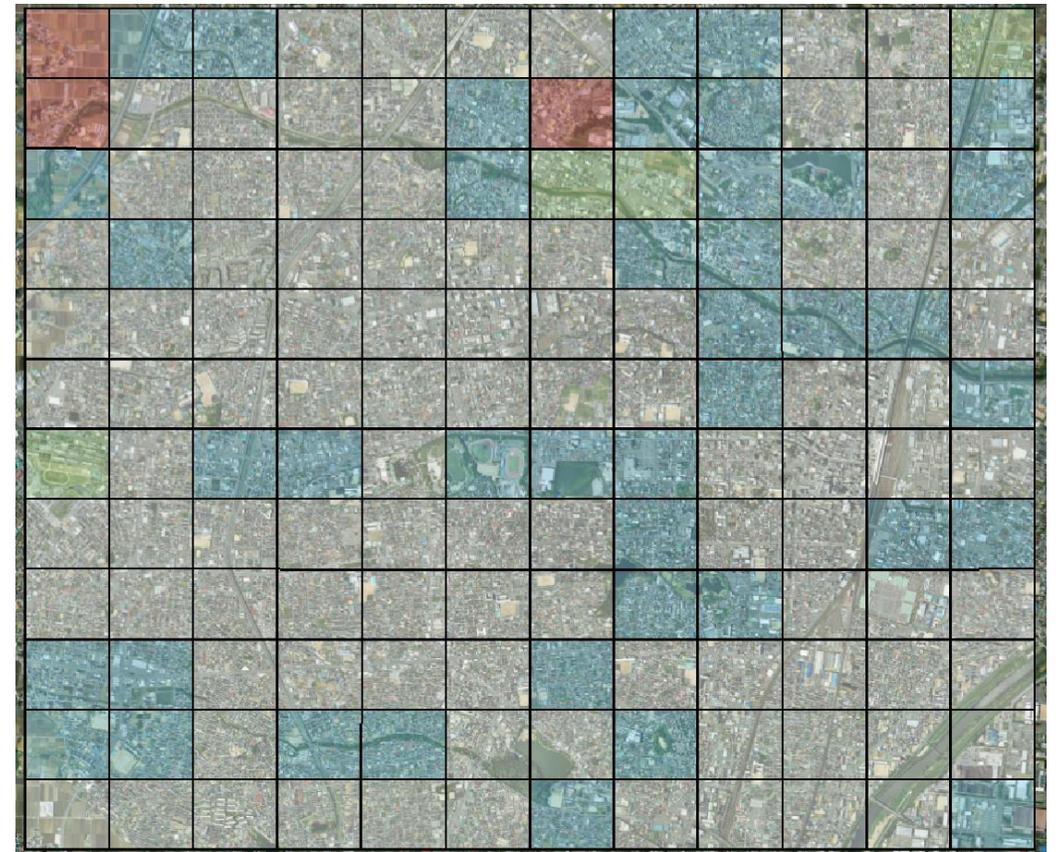


# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 メッシュサイズ別

正解 (メッシュサイズ500m)



判読結果 (メッシュサイズ500m)

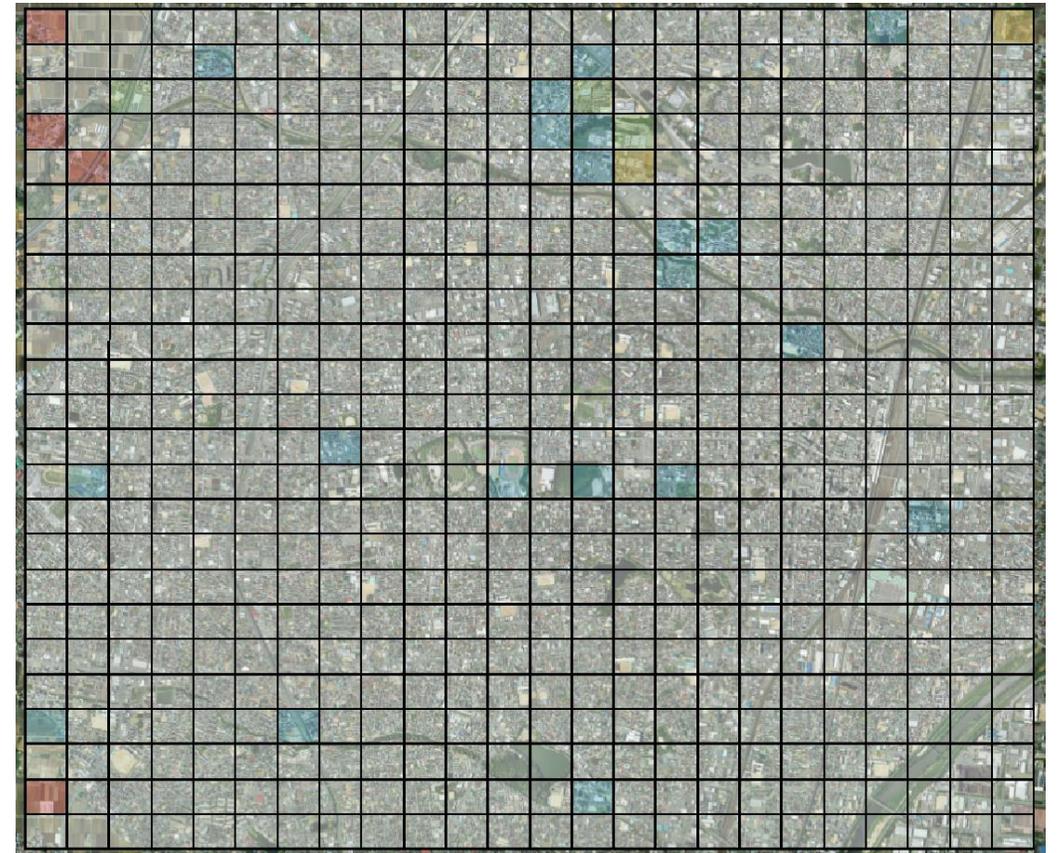


# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 メッシュサイズ別

正解 (メッシュサイズ250m)



判読結果 (メッシュサイズ250m)



# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ②検証結果

## 地域的な汎化性能（地域別）

全ての地域において、マイクロ平均F1値がKPIの75%を超えた

### 全地域

評価指標：マイクロ平均F1値=84.08%（KPIの75%を達成）

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	2,104	204	16	2	46
	優先度2	81	315	60	3	1
	優先度3	1	10	60	19	3
	優先度4	2	1	2	10	13
	優先度5	4	1	2	2	10

### 郡山市

評価指標：マイクロ平均F1値=96.18%（KPIの75%を達成）

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	537	6	0	0	2
	優先度2	4	14	2	1	0
	優先度3	1	0	1	1	0
	優先度4	1	0	0	0	0
	優先度5	4	0	0	0	2

### 東京23区

評価指標：マイクロ平均F1値=81.85%（KPIの75%を達成）

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	665	112	2	0	2
	優先度2	24	137	29	1	0
	優先度3	0	2	17	3	1
	優先度4	0	1	1	3	5
	優先度5	0	0	0	0	3

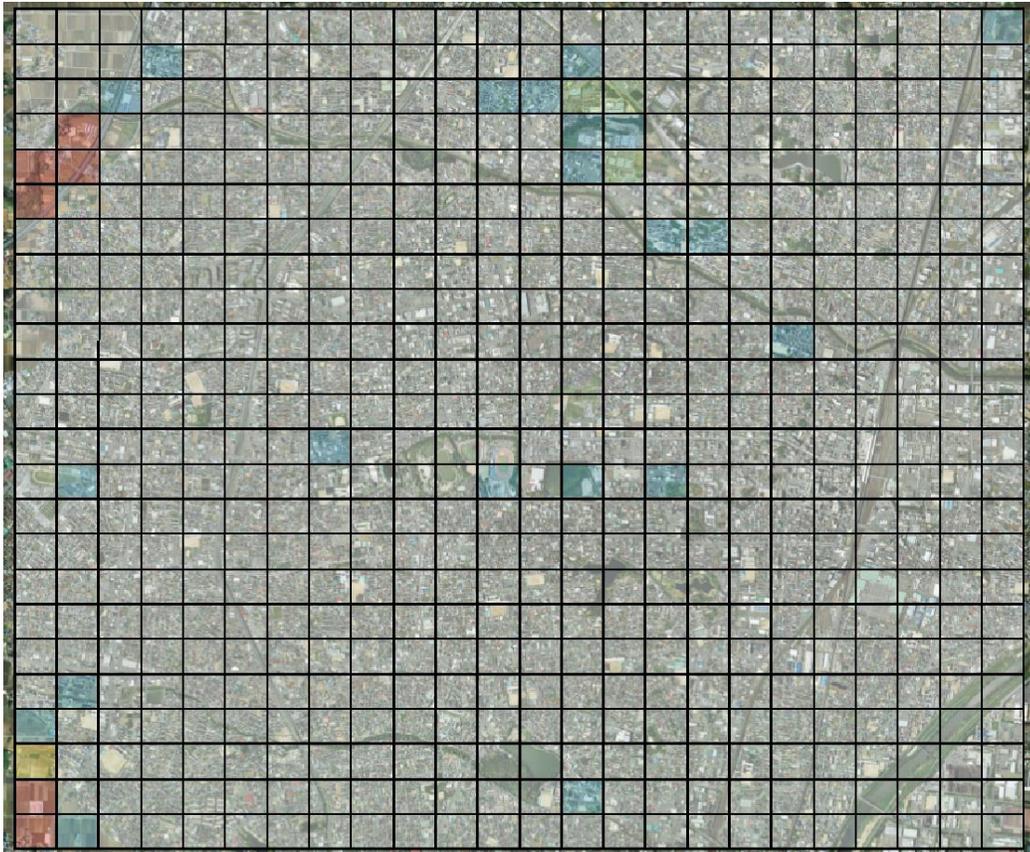
### 大阪市

評価指標：マイクロ平均F1値=75.00%（KPIの75%を達成）

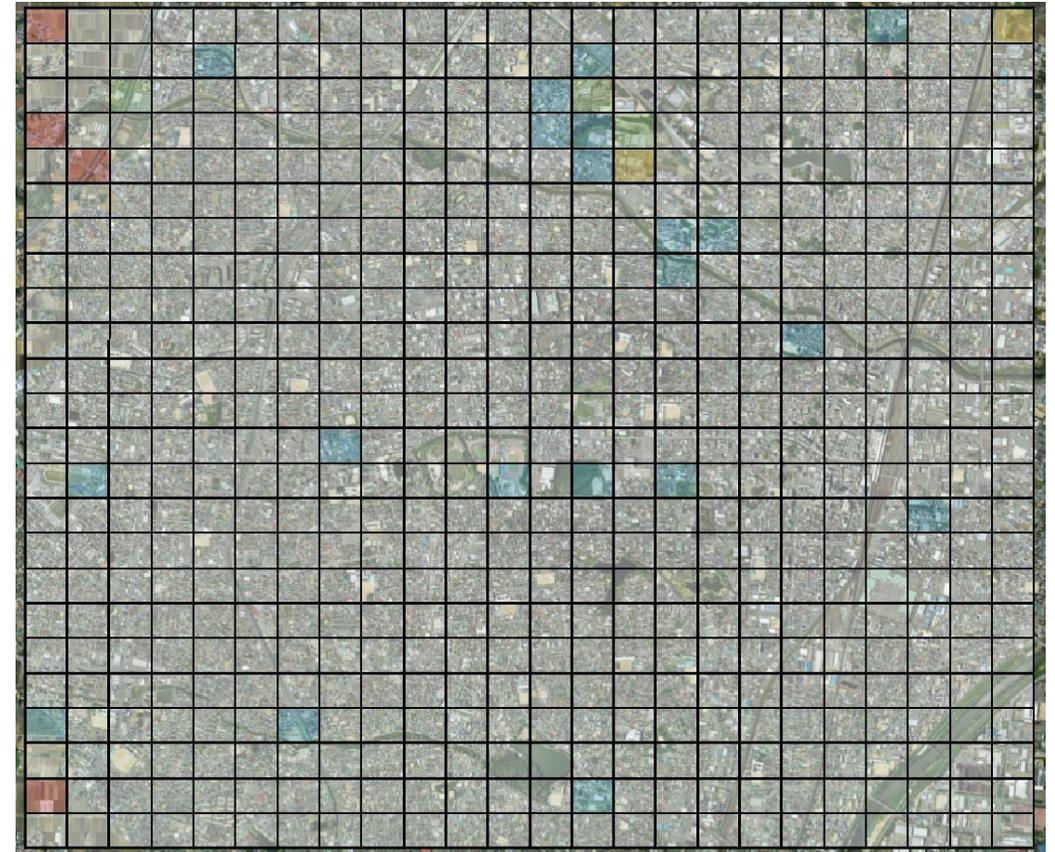
		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	96	18	2	1	1
	優先度2	18	49	4	1	1
	優先度3	0	1	7	1	0
	優先度4	1	0	0	1	1
	優先度5	0	0	1	0	0

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 地域的な汎化性能（地域別）

正解（郡山市）



判読結果（郡山市）

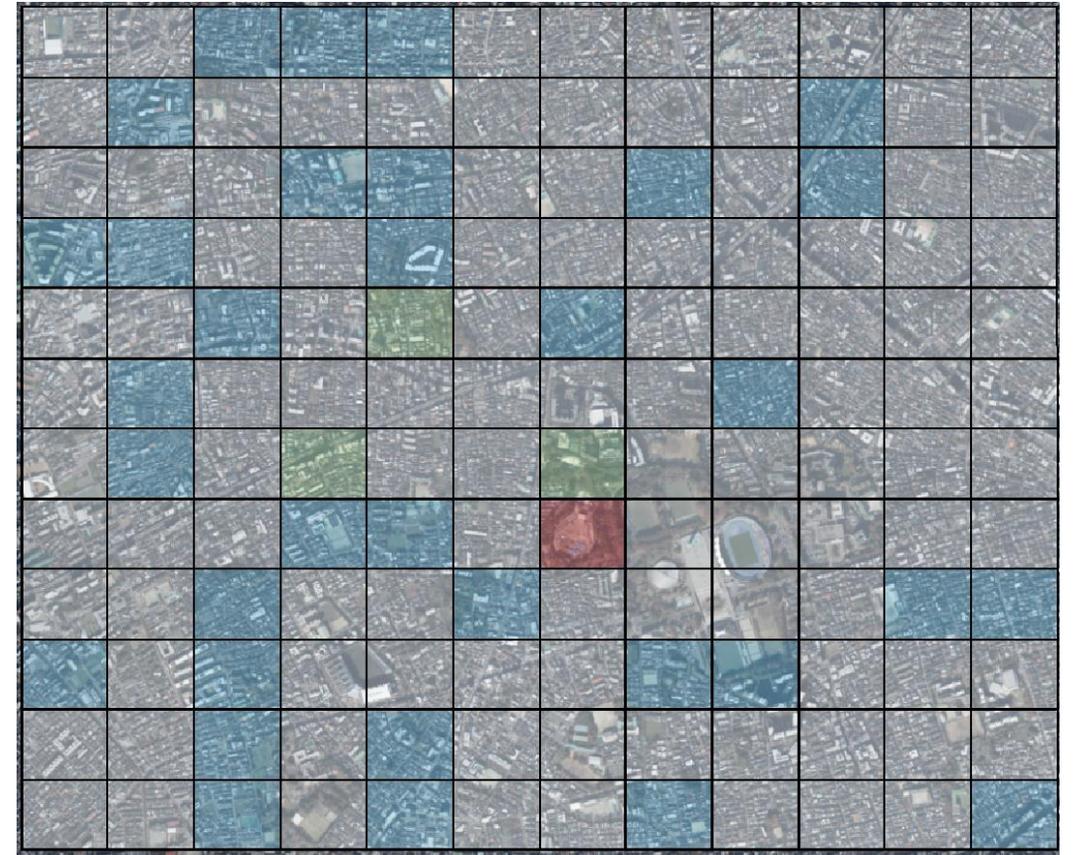


# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 地域的な汎化性能（地域別）

正解（東京23区）



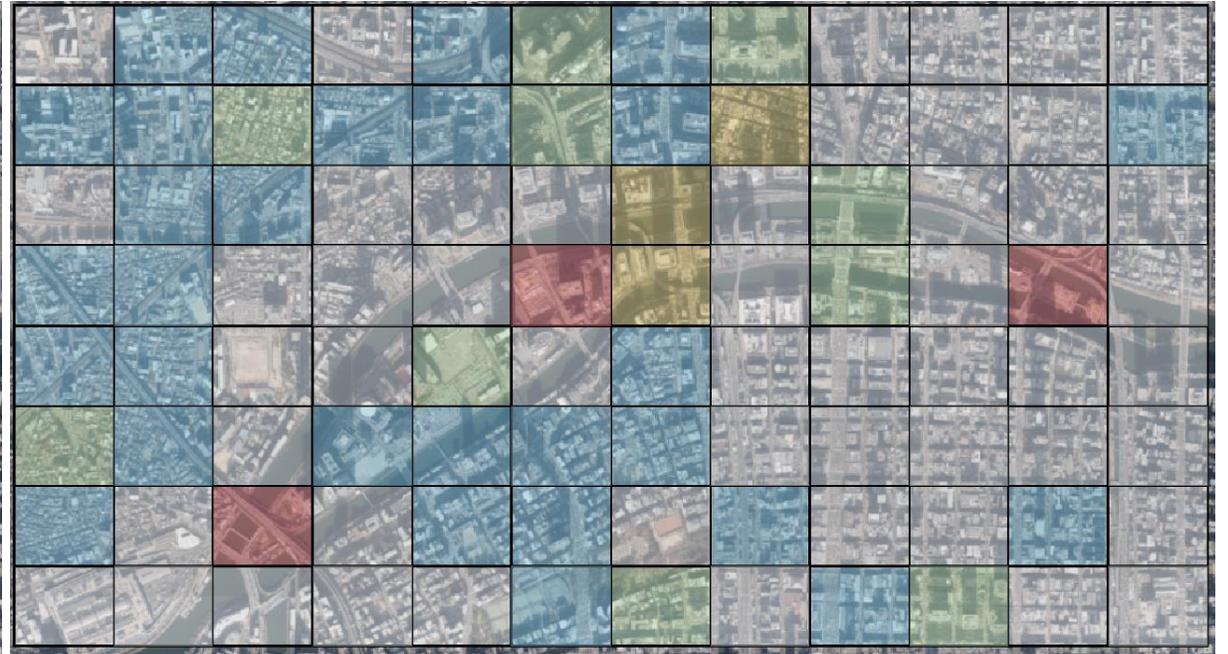
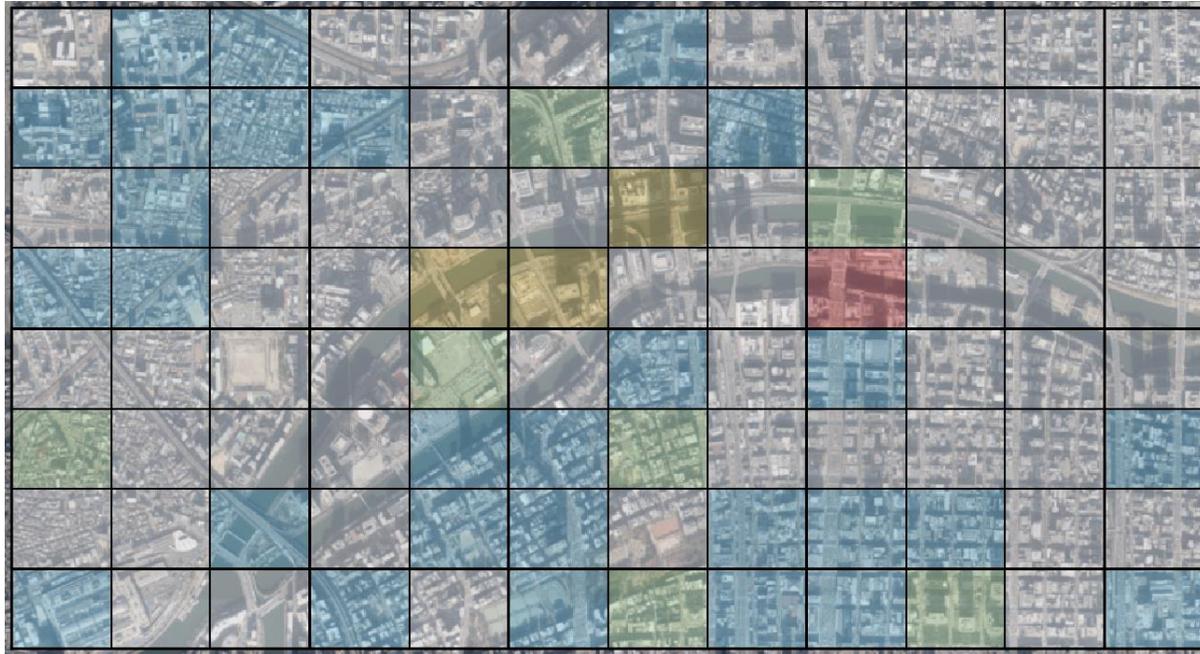
判読結果（東京23区）



# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 地域的な汎化性能（地域別）

正解（大阪市）

判読結果（大阪市）



# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果

## 超解像処理の有無

超解像処理を実施することで全体的に精度が向上した。中でも優先度2~5の分類精度向上の効果がある

超解像処理前 (=比較用参考)

超解像処理後 (=全地域250mメッシュと同じ)

評価指標 : **マイクロ平均F1値=82.06% (KPIの75%を達成)**

評価指標 : **マイクロ平均F1値=84.08% (KPIの75%を達成)**

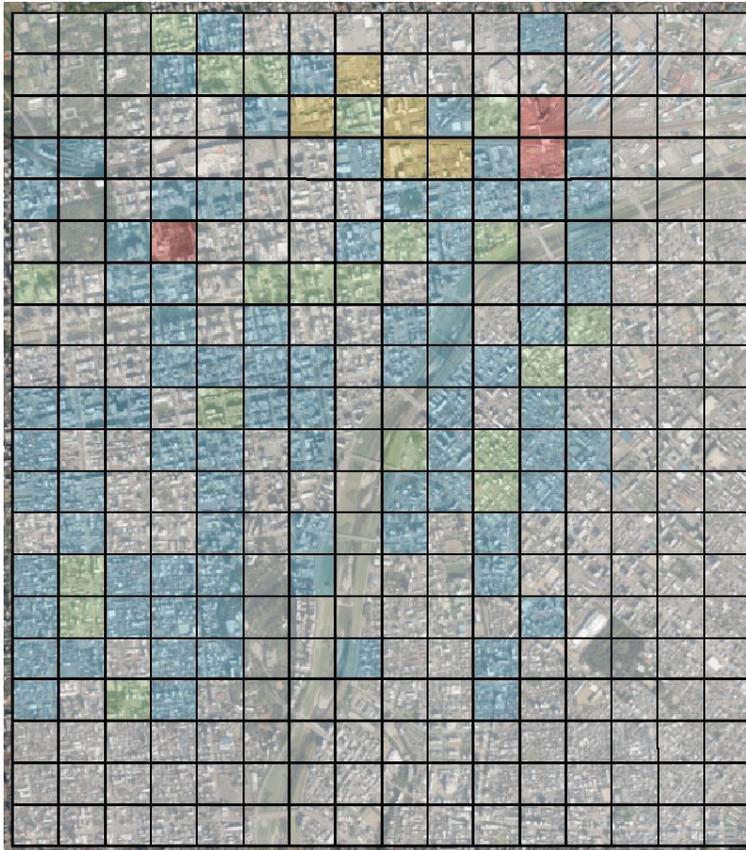
前提条件 : 超解像処理後の方が精度が高い

		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	2,239	100	15	4	14
	優先度2	268	156	34	0	2
	優先度3	15	22	44	9	3
	優先度4	1	3	4	8	12
	優先度5	6	2	2	1	8

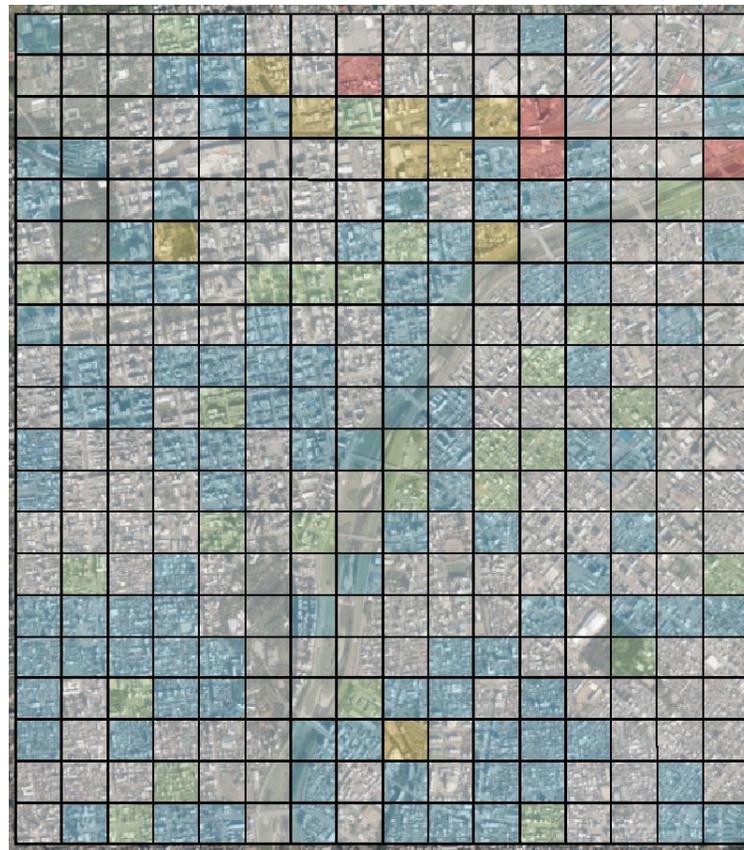
		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	2,104	204	16	2	46
	優先度2	81	315	60	3	1
	優先度3	1	10	60	19	3
	優先度4	2	1	2	10	13
	優先度5	4	1	2	2	10

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 超解像処理の有無

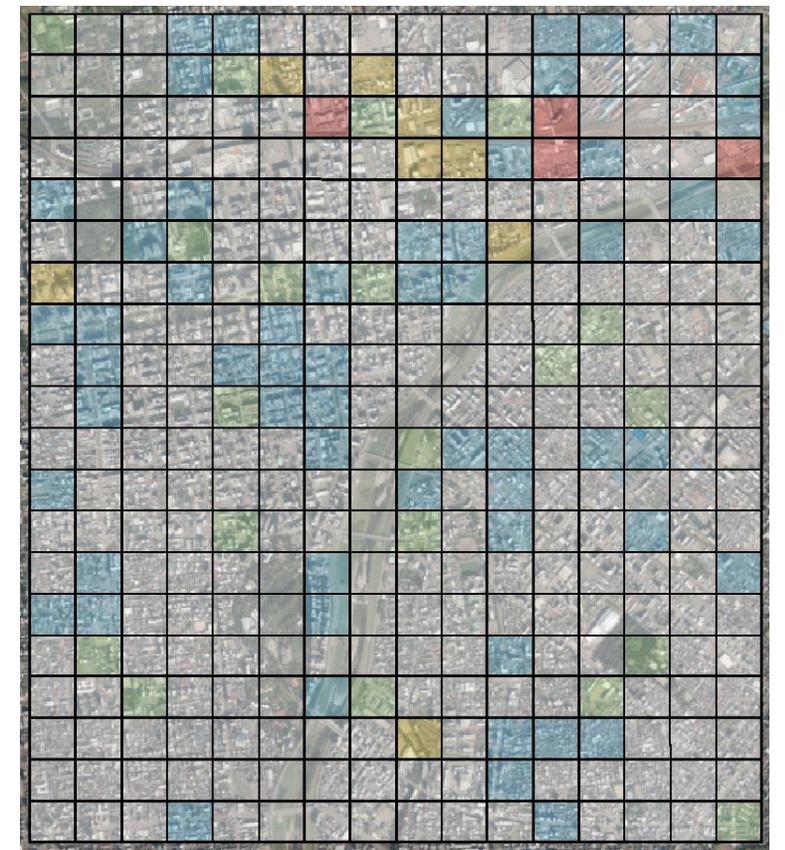
正解



判読結果 (超解像有)



判読結果 (超解像なし)



# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果

## 建物変化の有無

変化の有無のみの検出（2分類）で評価を行った結果、再現率87%、適合率80%というKPIに近い精度が得られていることが確認できた

更新優先度1と2～5の2分類

5分類（=参考用）

評価指標：再現率=87.02%（KPIの90%を達成できず）

評価指標：マイクロ平均F1値=84.04%（KPIの75%を達成）

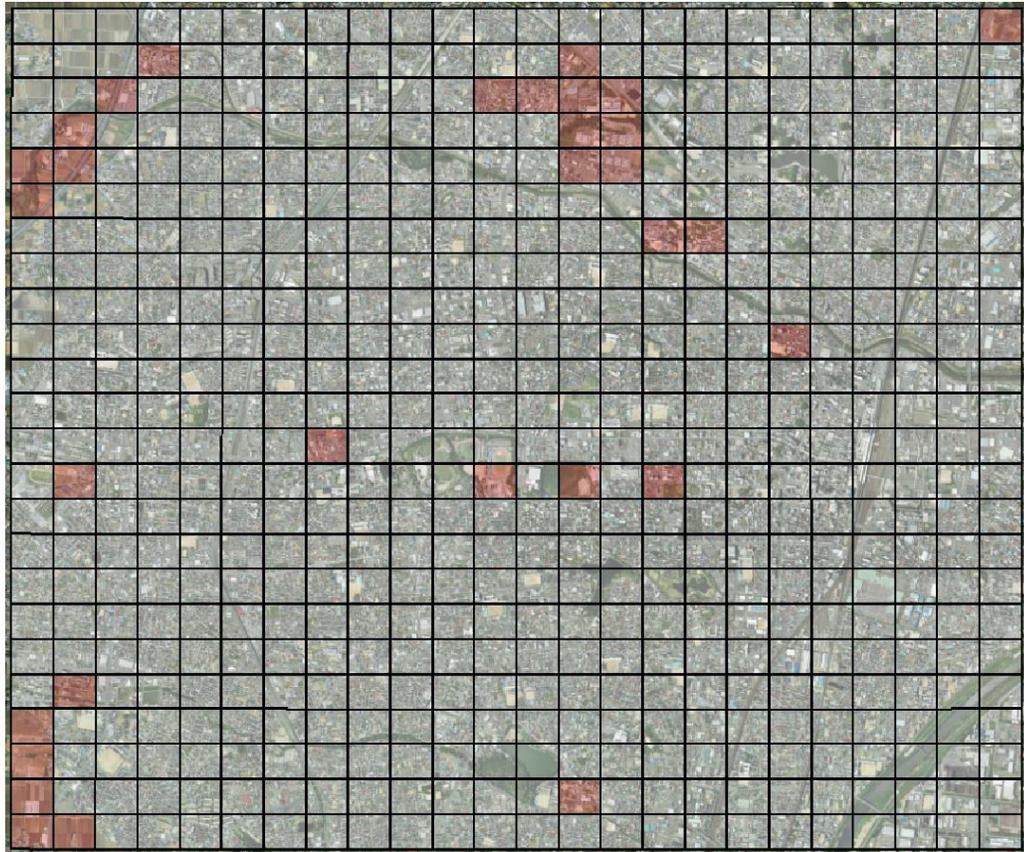
適合率=80.81%（KPIの80%を達成）

		AIモデルの推計結果	
		優先度1	優先度2～5
実際の区分	優先度1	2,104	268
	優先度2～5	88	512

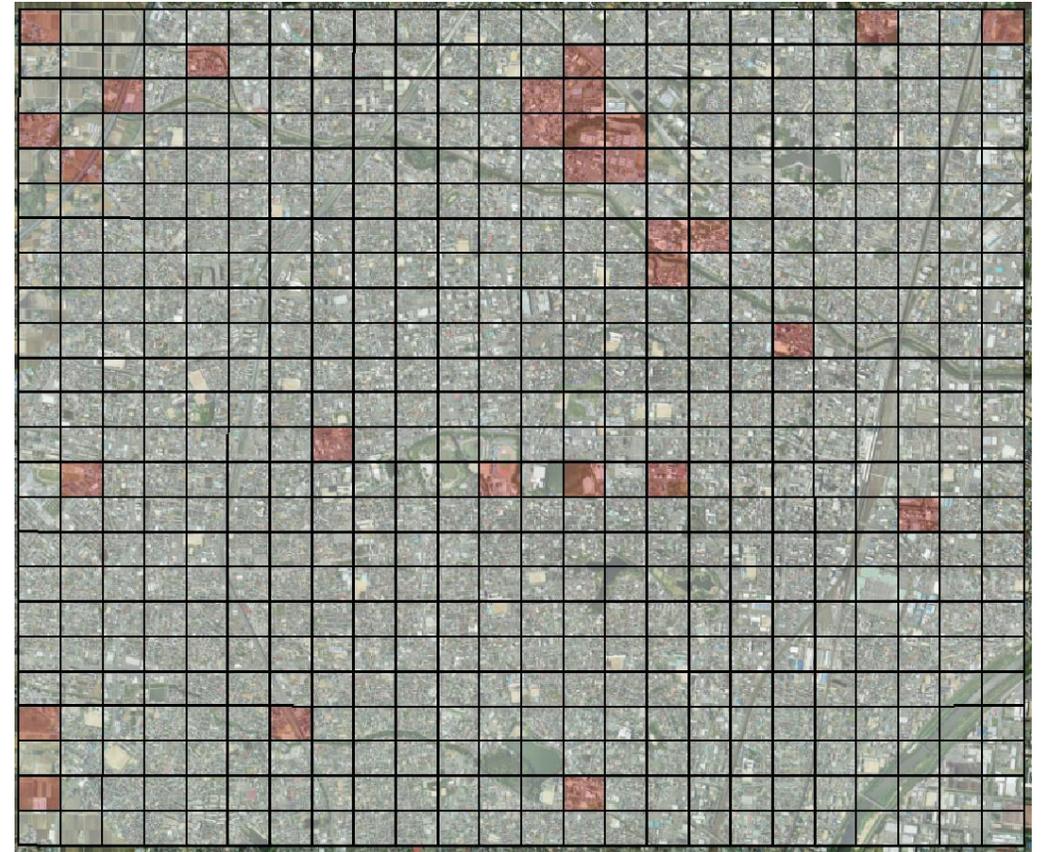
		AIモデルの推計結果				
		優先度1	優先度2	優先度3	優先度4	優先度5
実際の区分	優先度1	2,104	204	16	2	46
	優先度2	81	315	60	3	1
	優先度3	1	10	60	19	3
	優先度4	2	1	2	10	13
	優先度5	4	1	2	2	10

# IV. 実証技術の検証 > 1. 更新優先度マップにおけるAIモデルの精度検証 > ② 検証結果 建物変化の有無

正解 (2分類)



判読結果 (2分類)



# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容

## ヒアリング実施概要

更新優先度マップの利用可能性に関するヒアリング検証の実施概要は以下の通り

<b>目的</b>	コスト低減・運用・品質・表現方法等の観点から更新優先度マップの利用可能性を検証する
<b>実施期間</b>	2022年12月～2023年1月
<b>実施形式</b>	<ul style="list-style-type: none"> <li>• オンライン</li> <li>• 対面（横浜市、沼津市）</li> </ul>
<b>ヒアリング先</b>	<ul style="list-style-type: none"> <li>• 横浜市（2名）</li> <li>• 東京都（3名）</li> <li>• 沼津市（3名）</li> <li>• 地図関連サービス事業者：A社（3名）</li> <li>• 地図関連サービス事業者：B社（3名）</li> </ul>
<b>実施内容</b>	<ul style="list-style-type: none"> <li>• ALOS-3衛星の紹介</li> <li>• PLATEAU実証プロジェクト、更新優先度マップの紹介</li> <li>• PLATEAU VIEWによる更新優先度マップのデモ実施</li> <li>• ヒアリングシートによるヒアリングの実施</li> <li>• 質疑応答</li> </ul>

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容 ヒアリングの様子

対面形式で実施したヒアリングの様子は以下のとおり





# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容 ヒアリング項目

コスト低減・運用・品質・表現方法等の観点からヒアリングを行い、更新優先度マップの利用可能性を検証する

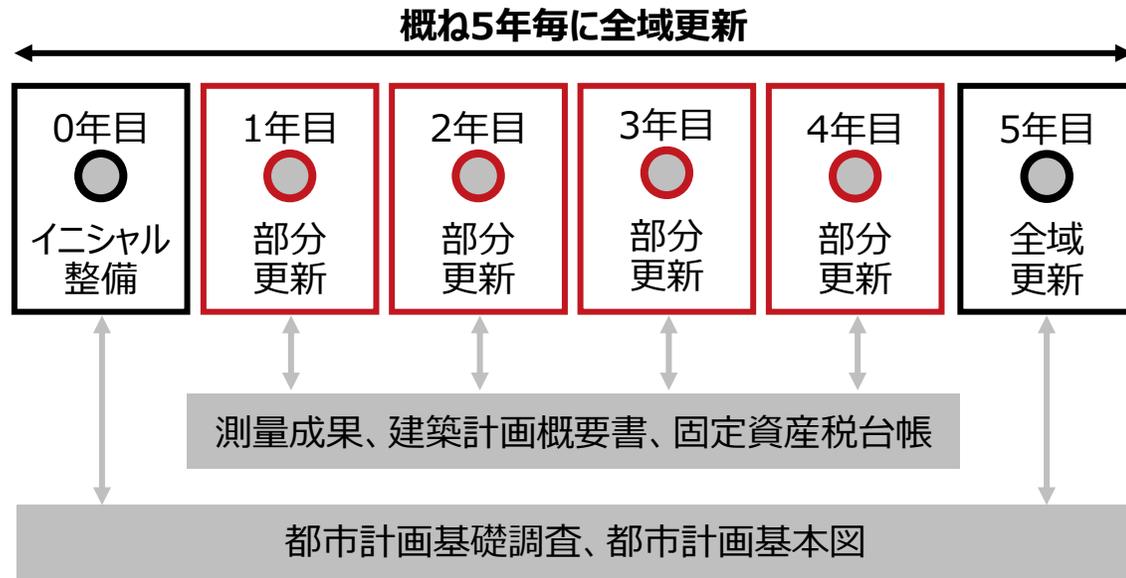
分類	ヒアリング項目	備考・注意点
全般	既存業務フローと業務課題を確認	—
コスト	データ更新時のコスト試算の前提条件	<ul style="list-style-type: none"> <li>前提条件の妥当性やコスト試算が不足している項目を確認</li> </ul>
	コスト削減効果の期待	<ul style="list-style-type: none"> <li>コスト削減効果を期待できそうか？</li> </ul>
運用	更新優先度マップの提供範囲と頻度	<ul style="list-style-type: none"> <li>提供範囲の要求</li> <li>最低限の更新頻度の要求（例：最低年1回）</li> </ul>
	更新回数（年1回）の部分更新の実現可能性	<ul style="list-style-type: none"> <li>更新頻度を上げるためには、どのような工夫が必要か？</li> </ul>
	更新業務の判断基準	<ul style="list-style-type: none"> <li>どの程度の変化規模が確認できれば、更新実施を判断可能になるか？</li> <li>メッシュ情報のみで判断・評価が可能か？</li> </ul>
品質	精度要求の下限	<ul style="list-style-type: none"> <li>再現率・適合率はどの程度ないと利用が進まないか？ （100%は達成できない前提での精度要求の下限は？）</li> </ul>
更新優先度マップの 表現方法	メッシュサイズの大きさ・他の集計方法の必要性	<ul style="list-style-type: none"> <li>メッシュサイズの大きさ（500m/1km/2.5km）は妥当か？</li> <li>他の集計方法（自治体の北部・東部等）が必要か？</li> </ul>
	属性項目の区分方法	<ul style="list-style-type: none"> <li>変化あり・変化なしの2分類の妥当性を確認</li> <li>属性項目（新築/滅失/建替等）の区分の重要性</li> </ul>
	閾値の区分方法	<ul style="list-style-type: none"> <li>閾値の基準となる分母の考え方の妥当性を確認 <ul style="list-style-type: none"> <li>- A案) 変化率 = 変化検出された面積 ÷ メッシュ内のLOD0の面積</li> <li>- B案) 変化率 = 変化検出された面積 ÷ メッシュ全体の面積</li> </ul> </li> </ul>

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容 コスト削減効果試算の条件 (1/5)

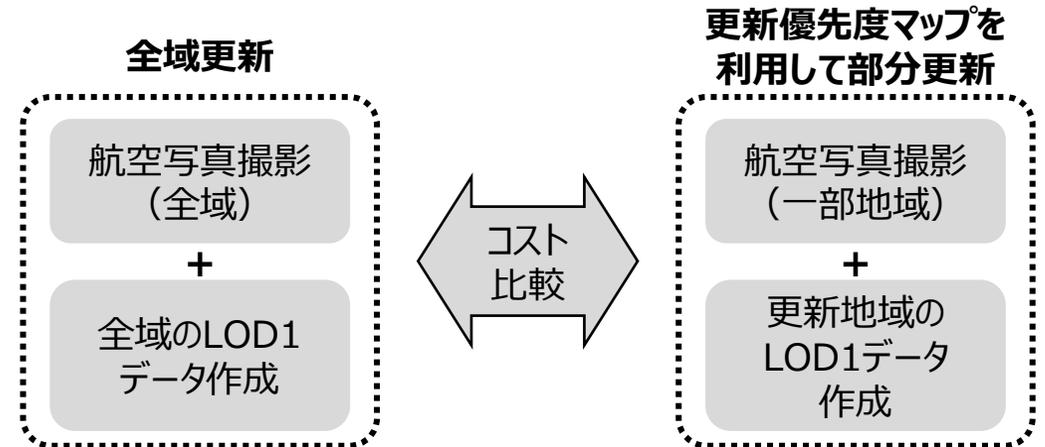
自治体には概ね5年毎に3D都市モデルの全域更新を実施してもらう想定で、全域更新した場合と更新優先度マップを活用して部分更新した場合の更新費用を比較することでコスト削減効果を試算する

自治体の3D都市モデル更新サイクル (利用想定イメージ)

コスト削減効果の比較対象



- 3D都市モデルの全域更新と部分更新とにかかる更新費用を比較する  
- 「更新費用」=「航空写真撮影費用」+「LOD1作成費用」
- 民間企業からの更新ニーズを考慮して、LOD1データの更新を実施



- 3D都市モデルは約5年毎に全域更新し、その他各年は更新優先度マップを活用して部分更新のみ実施
- 部分更新と同時期に、測量結果、建築計画概要書、固定資産税台帳等も合わせて更新を実施
- 全域更新と同時期に、都市計画基礎調査、都市計画基本図の更新を実施

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容

## コスト削減効果試算の条件 (2/5)

LOD1の作成費用は建物・道路のみを対象とし、国土交通省が提供する「3D都市モデル整備費用試算ツール」から人口規模と作成範囲を入力することで算出可能

項目	3D都市モデル整備費用の試算に必要な情報 (赤字：入力必須情報)	備考・留意点
基本情報	<ul style="list-style-type: none"> <li>自治体コード</li> <li>都道府県名</li> <li>市区町村名</li> <li><b>人口規模 (万人)</b></li> <li>自治体面積 (km<sup>2</sup>)</li> <li>政令市等の別</li> <li>都市計画区域 (km<sup>2</sup>)</li> <li>用途地域 (km<sup>2</sup>)</li> </ul>	<ul style="list-style-type: none"> <li>対象エリアの棟数を直ちに算出することが難しいため、人口規模に応じて1km<sup>2</sup>内の建物頭数を仮定して価格を設定している</li> </ul>
利用可能なデータ	<ul style="list-style-type: none"> <li>都市計画基本図の作成年度</li> <li>航空写真の撮影年度</li> <li>都市計画決定情報 (GIS) の作成年度</li> <li>災害リスク情報 (CSV又はGISデータ) の作成年度</li> </ul>	<ul style="list-style-type: none"> <li>「利用可能なデータ」は、3D都市モデルの元となるデータのこと</li> <li>本ツールは3D都市モデルのデータ作成費用のみを対象とするため、実際は「利用可能なデータ」の取得・作成費用が別途必要</li> </ul>
作成範囲	<ul style="list-style-type: none"> <li><b>建築物モデルLOD1 (km<sup>2</sup>)</b></li> <li><b>建築物LOD2 (km<sup>2</sup>)</b></li> <li><b>道路LOD1 (km<sup>2</sup>)</b></li> <li><b>都市計画決定情報LOD1</b></li> <li><b>土地利用現況LOD1</b></li> <li><b>災害リスク情報LOD1 (個数)</b></li> </ul>	<ul style="list-style-type: none"> <li>「建築物モデルLOD1」、「建築物LOD2」、「道路LOD1」については、対象範囲を記入</li> <li>「都市計画決定情報LOD1」については、対象とするか否かを記入</li> <li>「災害リスク情報LOD1」については、対象となる洪水浸水想定区域図の数を記入</li> </ul>

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容 コスト削減効果試算の条件 (3/5)

更新優先度マップを活用して部分更新を行う際の作業フローとコスト試算は以下の通り

部分更新の作業フロー

コスト試算例（民間企業が撮影した航空写真を使用する場合）

	No.	項目	地上解像度等	初期費用*1	単価 (km <sup>2</sup> )	数量 (km <sup>2</sup> )	計*1 (税抜)
更新度優先マップによる更新箇所の選定	1	航空写真オルソ画像データ	8cm	¥ 50,000	5,000	3	¥ 65,000
	2	DSMデータ	8cm	¥ 50,000	25,000	3	¥ 125,000
3D都市モデル整備に必要なリソースデータの調達 (VDP*2を3km <sup>2</sup> を使用する場合)	3	リソースデータ小計					¥ 190,000
	4	LOD1 (建物・道路) 更新費用	2500レベル	別途		3	¥ 450,000
3D都市モデル (LOD1建物・道路) の作成	5	合計					¥ 640,000

### 【試算の前提条件】

- 航空写真オルソ画像データ、DSMデータの費用は地上解像度に応じて単価が変動する
- LOD1の更新は建物・道路のみを想定しており、属性項目の入力には別途追加費用がかかる
- LOD1の更新費用は、パスコの複数自治体の実績をベースに平均的なコストとして試算した
- 建物密度等によってLOD1 (建物、道路) の費用は変動する
- VDPのオルソ画像データの納品は対象外とし、ストリーミングでの使用料のみで試算
- 実際の見積りに当たっては、データ作成事業者から通常の手続きに従って見積りが必要
- 実際のデータ作成に当たっては、事前調査や経年変化部分の修正作業、プロジェクト管理、資料作成等の費用が別途必要

\*1 一般的な概算値として算出

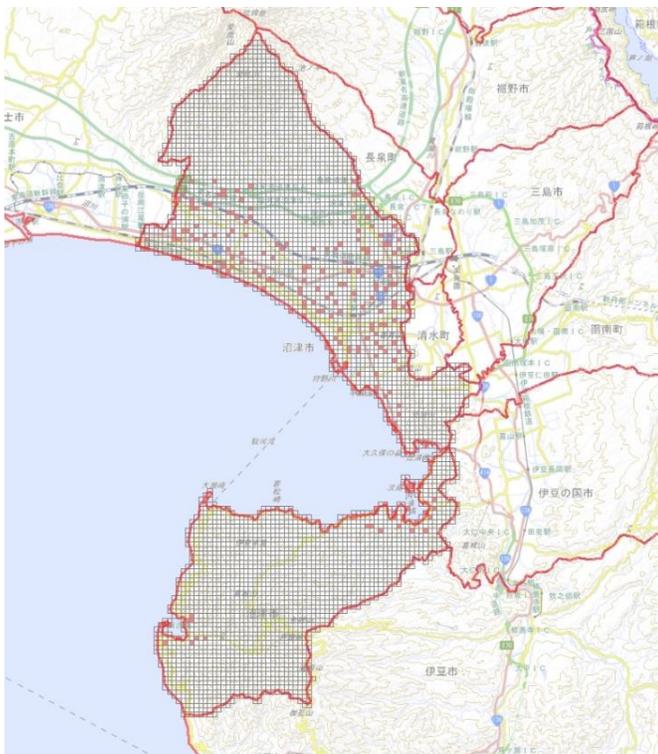
\*2 Vexcel Data Programは地上解像度が8cmの航空写真オルソ画像データとDSMとなる。最小購入面積は1図郭(東西2km×南北1.5km:3km)からで、WMTS形式のストリーミングで閲覧可能

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容 コスト削減効果試算の条件 (4/5)

自治体ヒアリングの際に提示するコスト削減効果のシミュレーションは以下の通り

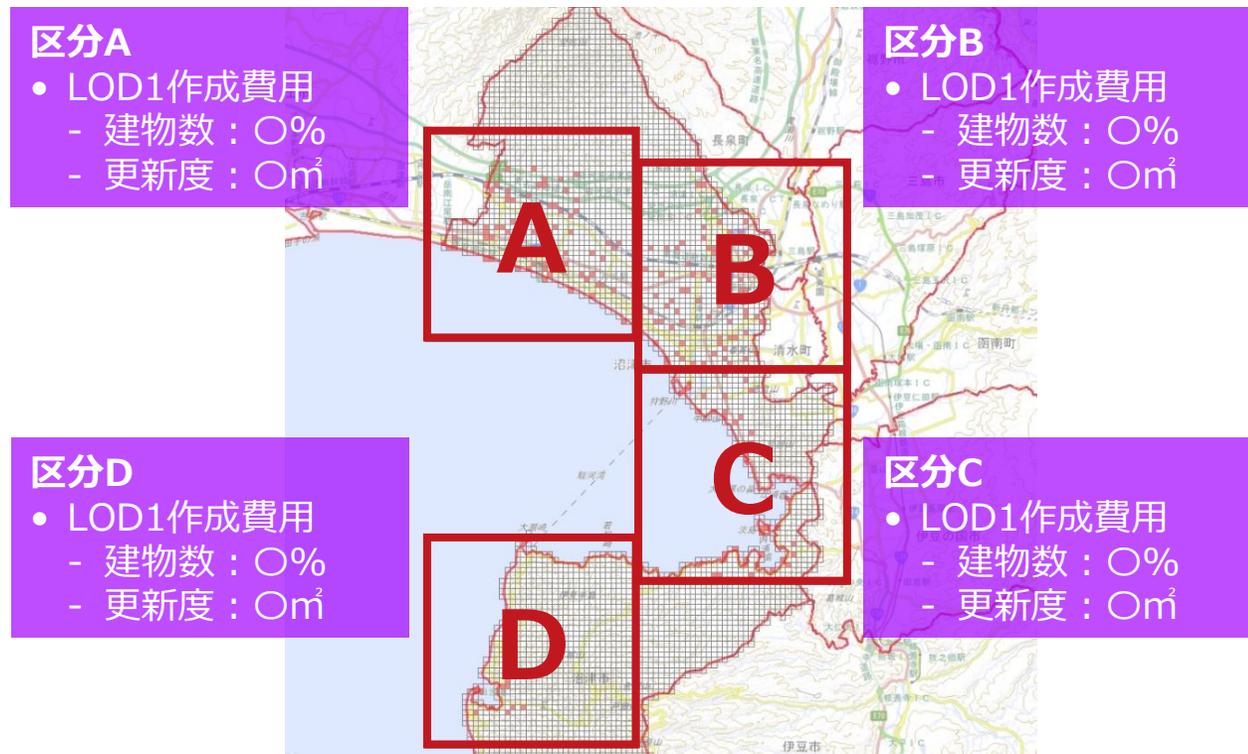
全域更新の場合

- 撮影費用\*2：1,000万円
- 3D都市モデル（LOD1データ）作成費用：347～416万円/年



部分更新の1回あたりの場合（1地区\*10km四方）

- 撮影費用\*2：167万円
- 3D都市モデル（LOD1データ）作成費用：建物数と更新度で変動



**区分A**

- LOD1作成費用
- 建物数：0%
- 更新度：0m<sup>2</sup>

**区分B**

- LOD1作成費用
- 建物数：0%
- 更新度：0m<sup>2</sup>

**区分D**

- LOD1作成費用
- 建物数：0%
- 更新度：0m<sup>2</sup>

**区分C**

- LOD1作成費用
- 建物数：0%
- 更新度：0m<sup>2</sup>

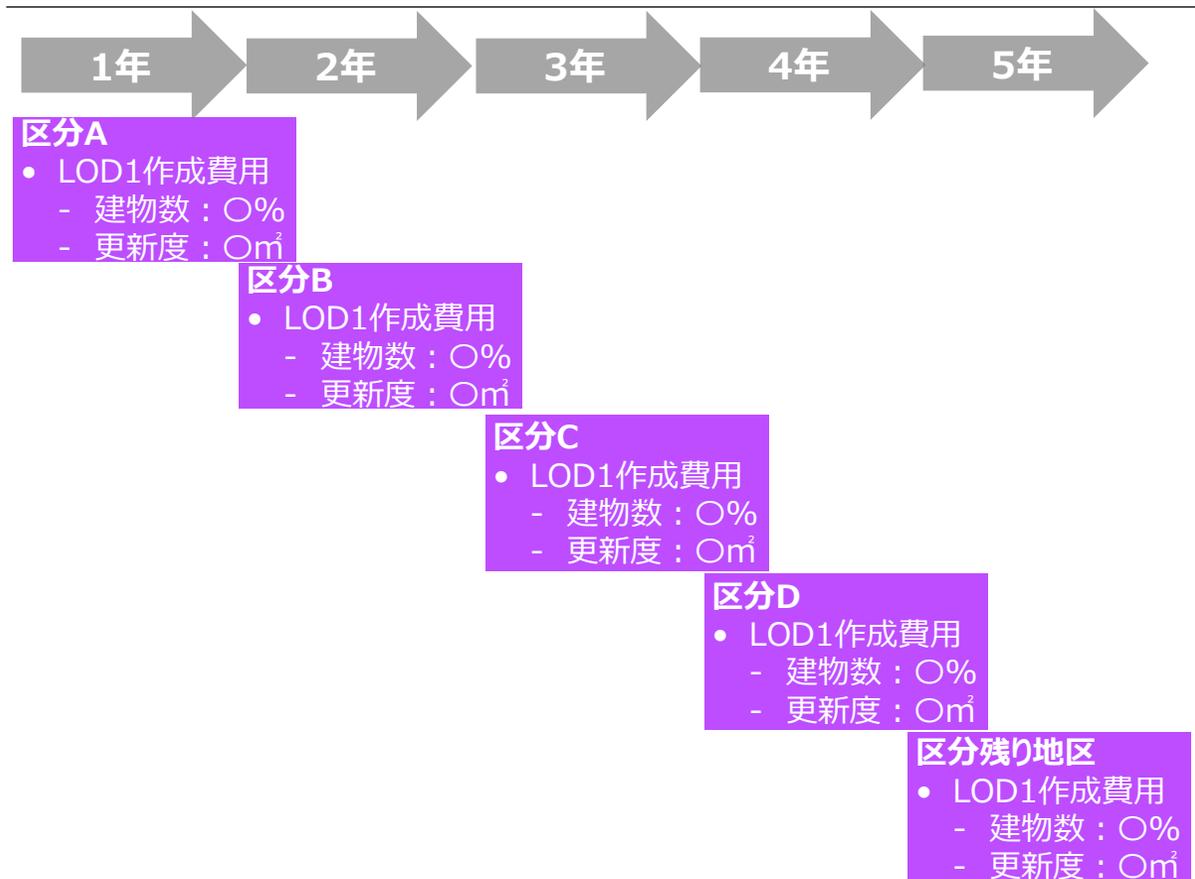
\*1 区分A～Dの区割りは、航空飛行コースの効率を勘案して10km四方に設定し、範囲は更新優先度マップを勘案して移動し設定する

\*2 撮影費用は、一般的な概算値として算出

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ① 検証内容 コスト削減効果試算の条件 (5/5)

自治体ヒアリングの際に提示するコスト削減効果のシミュレーションは以下の通り

部分更新の実施イメージ (5年間) \*1



部分更新の場合 (5年間のトータルコスト)

- 撮影費用\*2 : 167万円/年×4 + 残り地区 = 1,000万円
- 3D都市モデル (LOD1データ) 作成費用 : 347~416万円/年

5年間のトータルコストとしては、5年に1度の全域更新と、順次更新していく部分更新方法とは、ほぼ同等になる見込み

一方で、現実社会と3D都市モデルの時点のズレが解消されていくことで、利便性が向上する

\*1 区分A~Dの区割り、航空飛行コースの効率を勘案して10km四方に設定し、範囲は更新優先度マップを勘案して移動し設定する

\*2 撮影費用は、一般的な概算値として算出

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ② 検証結果 サマリ

3D都市モデル更新の必要性把握には有用であるものの、実際の更新業務を想定した場合、現状では部分更新を行うスキームがなくオペレーション上の課題があることが分かった

KPI		結果	示唆
部分更新の可能性	自治体	<ul style="list-style-type: none"> <li>部分更新箇所の情報を把握できれば、部分更新のニーズは十分にある</li> <li>年1回の部分更新のためには自治体内の複数部署での調整が必要</li> </ul>	<ul style="list-style-type: none"> <li>部分更新をした箇所は、次の更新対象箇所から外すため、全体での更新コストを削減もしくは以前と同等にする必要がある</li> <li>部分更新のためだけに新たに航空写真を撮影することは現実的ではないので、他業務と合わせてリソースデータを取得することでコスト低減を図り、更新頻度を高めることが必要</li> </ul>
	民間企業 (地図関連サービス)	<ul style="list-style-type: none"> <li>更新優先度マップを活用することで、自社製品更新のための航空写真アーカイブの調達が効率化される</li> </ul>	
更新優先度マップの提供範囲と頻度		<ul style="list-style-type: none"> <li>データ更新目的であれば、年1回の更新情報で良い</li> <li>代替の更新情報の得にくい郊外の方が、ニーズが高い</li> </ul>	<ul style="list-style-type: none"> <li>原則年1回の更新情報で問題ないが、大規模開発や災害発生時には個別での更新情報の提供ニーズが高い</li> </ul>
精度要求		<ul style="list-style-type: none"> <li>建物データ更新には、建物単位の変化情報が必要</li> <li>建物データ更新に利用する航空写真の調達判断には高い精度要求はないため、利用の可能性あり</li> </ul>	<ul style="list-style-type: none"> <li>1棟1棟の建物データ更新のためには、固定資産業務等での撮影成果や高解像度衛星・航空写真アーカイブ画像の限定エリアでの調達等が必要</li> <li>更新エリアの選定には、図郭単位でも利用可能</li> </ul>
その他のニーズ		<ul style="list-style-type: none"> <li>建物の変化情報以外にも、道路・植生等の変化情報がほしいとのニーズ有り</li> </ul>	<ul style="list-style-type: none"> <li>建物以外にも変化情報の提供範囲を広げることで、より多くのユーザー利用が期待できる</li> </ul>

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ② 検証結果 自治体へのヒアリング結果 (1/2)

更新の可視化については棟単位での把握の要望はあるものの有用性は評価されている。ただし、3D都市モデルの更新業務に対しては、メッシュ単位での更新オペレーションが現時点で存在しないことが課題

分類	ヒアリング項目	ヒアリング結果
全般	既存業務フロー・業務の課題を確認	<ul style="list-style-type: none"> <li>3D都市モデルのみを更新するという予算はないので、基本図と合わせた更新になる。現在、市内を数年かけて更新している。使われ方によっては、更新優先度マップで変化が大きい箇所が分かれば、更新するニーズはある</li> <li>更新体制は未定である。市区町村が担当する場合もありうるし、全体を実施する場合は都道府県が対応することもあると思う。ただし、都度更新ということは難しく、5年間隔になる</li> <li>部分更新の実施の有無は別として、集中的に更新する際の材料になる</li> <li>ピンポイントのエリアで3D都市モデルを活用しようという場合が多い。そのために部分的に更新するのは有効だと思う。一方で、いずれかの時点では全体で一斉に更新する必要がある</li> </ul>
コスト	データ更新時のコスト試算の前提条件	<ul style="list-style-type: none"> <li>他部署で毎年撮影している航空写真を利用することになる</li> <li>様々な原典となる航空写真があるが、何を扱うか整理する必要がある</li> <li>基本図の更新では、都市計画全体だと、特別緑地・生産緑地・道路等も更新対象となる。3D都市モデルのみを更新する予算はないので、基本図更新と合わせることになる</li> <li>メッシュの中でどの建物が変化したのかを確認するのは目視確認になってしまうので、どの建物が変化しているかが分かるのであれば、使いやすくなる</li> <li>更新優先度マップは部分更新には有効だと思うが、費用負担は難しい（無償なら参考に使用する）</li> <li>3D都市モデル更新時に別途、航空写真撮影を実施するのは現実的でないので、既存の航空写真などで対応となる</li> </ul>
	コスト削減効果の期待	<ul style="list-style-type: none"> <li>5年更新より、部分更新を実施しておいた箇所においては、費用がかからないなどの効果が必要</li> <li>無償でこのようなユースケースのサービスの範囲が広がっていくことを期待している</li> <li>変化しているメッシュの国土基本図修正図化では、予察業務を省くことができるかもしれない</li> </ul>

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ② 検証結果 自治体へのヒアリング結果 (2/2)

更新の可視化については棟単位での把握の要望はあるものの有用性は評価されている。ただし、3D都市モデルの更新業務に対しては、メッシュ単位での更新オペレーションが現時点で存在しないことが課題

分類	ヒアリング項目	ヒアリング結果
運用	更新優先度マップの提供範囲と頻度	<ul style="list-style-type: none"> <li>道路や植生の変化も確認したい</li> </ul>
	更新回数（年1回）の部分更新の実現可能性	<ul style="list-style-type: none"> <li>ALOS-3衛星が観測頻度（年5～8回）までの更新は必要ない</li> <li>変化箇所を確認できる頻度が高いのはありがたいが、行政としては、年単位で事業を行うために、年1回の事業判断になる</li> <li>5年に1度の都市計画基本図の更新も難しさがある。まずは、年1回程度の更新の確認をしたい。その次に、災害時における被災箇所の変化を確認したい</li> </ul>
	更新業務の判断基準	<ul style="list-style-type: none"> <li>更新優先度がある基準以上の箇所を対象にして更新を実施するなどの基準を決めることになる</li> <li>都市計画基本図の更新は、大きな建物が完成してから更新すれば良いという意見もある。または、中心市街地だけ見ておくということはあるかもしれない。前回の更新は、新東名が完成したときに実施した。5年毎の全域一斉時に更新するよりも、頻度高く更新した方がコストを抑えられるという説明が可能なら良い</li> </ul>
品質	精度要求の下限	<ul style="list-style-type: none"> <li>1棟の変化が分かると良い。面積ではなく棟数が必要</li> <li>予察では1棟も漏らせない。（100%更新がない地域が分かると良い）</li> </ul>
更新優先度マップの表現方法	メッシュサイズの大きさ・他の集計方法が必要性	<ul style="list-style-type: none"> <li>図郭の中で把握するより、ポイントで分かった方が良いと思う。例えば、250mメッシュを利用する場合、1kmメッシュや基本図の図郭内のどこが変化したかがより細かく把握できる。また、統計的に使うこともできる</li> <li>断定はできないが、メッシュサイズは小さい方が良いと思う</li> </ul>
	属性項目の区分方法	<ul style="list-style-type: none"> <li>基本図の更新目的であれば、変化あり・なしのみで、詳細な属性（例えば、新築・滅失・改築の区分）はなくても良い</li> <li>新築・滅失・改築は、区分できる方が良い。また、属性（商業施設、住宅、工場等）が追加されると、なお良い。（建物属性とリンク）</li> </ul>
	閾値の区分方法	<ul style="list-style-type: none"> <li>現時点では判断は難しいが、どちらもあれば良い</li> <li>A案、B案の両方がある方が良い。（郊外で1棟も漏らしたくない場合は、A案を使うなどで使い分けるため）</li> </ul>

# IV. 実証技術の検証 > 2. 実証システムの価値検証 > ② 検証結果

## 地図関連サービス事業者へのヒアリング結果

民間利用の観点では、1つの統計情報として変化検出が可視化されたデータは有用であるとのこと

分類	ヒアリング項目	ヒアリング結果
全般	既存業務フロー・業務の課題を確認	<ul style="list-style-type: none"> <li>自社で整理している変化箇所の情報を基にして航空写真を調達しているため、更新優先度マップの情報があると有用である</li> <li>自社製品更新のための調査で、都市部は年間1回以上更新している。郊外は2～3年に1回の更新</li> </ul>
コスト	データ更新時のコスト試算の前提条件	<ul style="list-style-type: none"> <li>無償を希望するが、有償でも価値があると思う。（業務全体でも費用対効果次第）</li> <li>変化点は社内でも蓄積しており、ある程度エリアでの変化は把握している</li> <li>一棟単位での変化抽出が出来れば大変有用と考える</li> </ul>
	コスト削減効果の期待	<ul style="list-style-type: none"> <li>更新が少ない箇所（図郭）は、航空写真を調達しないことでコスト削減が考えられる</li> </ul>
運用	更新優先度マップの提供範囲と頻度	<ul style="list-style-type: none"> <li>年に1度あると良い</li> <li>郊外だと更新周期が長い為、エリアレベルでの変化でも役に立つかもしれない</li> <li>市レベルだと1年に1回、町村レベルだと2～3年に1回の更新頻度なので、場所は田舎の方が更新優先度マップのニーズは高いと思う。（更新頻度が遅い地域）</li> <li>建物以外にも道路等の変化情報が欲しい</li> </ul>
品質	精度要求の下限	<ul style="list-style-type: none"> <li>統計情報なので、それほど高い精度は求めることはない。あくまで参考情報として利用する</li> <li>影や雲量、積雪は影響するのか確認をしたい</li> </ul>
更新優先度マップの表現方法	メッシュサイズの大きさ・他の集計方法が必要性	<ul style="list-style-type: none"> <li>更新作業用の航空写真の調達図郭と合ったサイズがあると良い</li> <li>市区町村単位で把握したい</li> </ul>
	属性項目の区分方法	<ul style="list-style-type: none"> <li>新築、建替、滅失の属性があると良い</li> </ul>
	閾値の区分方法	<ul style="list-style-type: none"> <li>A案とB案の両方</li> <li>現状では判断がつかないが、田舎の方がニーズが高そうなのでA案で良いと思う</li> </ul>

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

ALOS-3衛星では高い位置精度の画像提供が可能になり、またデータ更新箇所は将来的には棟単位での把握が可能となる見込み。更に、3Dモデルにより変化検出の精度が向上する

項目	想定される技術面での優位性
ALOS-3衛星画像の活用	<p><b>衛星画像では倒れこみを避けた高い位置精度の画像提供が可能</b></p> <ul style="list-style-type: none"> <li>ALOS-3衛星画像の撮影能力の特徴として、70km幅の連続撮影が可能であり、直下視を基本としていることが挙げられる</li> <li>これらの特徴によって、他の衛星画像で生じる建物の倒れ込み（写真の周辺部ほど、写真中心から外側へ傾くような現象）が少なく、位置精度の高いオルソ画像を提供することが可能である</li> </ul>
データ更新箇所の把握	<p><b>現状でも短期間で団地やエリアレベルの変化を把握でき、超解像処理が進めば棟単位での把握が可能になる見込み</b></p> <ul style="list-style-type: none"> <li>PLATEAU公開済の航空写真と最新時期のALOS-3衛星画像を準備すれば、AIモデルを適用することで、短時間で大まかな変化箇所の把握が可能になる</li> <li>ALOS-3衛星の80cmの分解能では、戸建建物レベルの1棟1棟の詳細な変化を得ることはまだ難しいが、集合住宅・団地・大型の開発エリア等の変化が集中する箇所を高い優先度として判定可能になる</li> <li>今後、ALOS-3衛星画像への超解像処理等の高度化がさらに進むことで、将来的には戸建建物レベルの詳細な変化を正確に捉えられるようになる可能性がある</li> </ul>
既存3D都市モデル等の活用	<p><b>異なる時期の2画像に3D都市モデルを追加することで、変化検出の精度が向上する</b></p> <ul style="list-style-type: none"> <li>異なる時期の2画像だけを入力して変化検出するよりも、既存の3D都市モデルデータを活用し既知の建物箇所に対する変化検出処理を追加することで、変化の精度向上が可能になる</li> </ul>



## V. 成果と課題 > 1. 今年度の実証で得られた成果

### ② 3D都市モデルによるビジネス面での優位性

ALOS-3衛星は全国の市町村データの年1回以上の更新が可能であり、また更新優先度マップの活用で事業者や自治体の地図データ更新が効率化する。更に、種々の3D都市モデルの活用で変化把握が容易になる

項目	想定されるビジネス面での優位性
ALOS-3衛星画像の活用	<b>全国の市町村用に必要なデータの年1回以上の更新が可能で、かつ他衛星に比べ低コスト</b> <ul style="list-style-type: none"><li>• 全国の全域を年5～8回撮影可能な撮影能力を生かし、全市町村でも年1回の部分更新を実現可能にするための画像提供が可能である</li><li>• 衛星画像の調達価格は、他の高解像度衛星に比較して割安であり、価格競争力がある</li></ul>
データ更新箇所の把握	<b>更新優先度マップの活用で、民間関連サービス事業者や自治体の地図更新業務の効率化・低コスト化が可能</b> <ul style="list-style-type: none"><li>• 民間関連サービス事業者等は、現状は地図データ更新のために航空写真アーカイブを調達している。よって、更新優先度マップを利用し航空写真アーカイブの調達を効率化することのニーズが高い</li><li>• また、自治体が調達している航空写真・高解像度衛星がある場合は、部分更新作業を含めたトータルコストの低減を提示することで、利用促進が期待できる</li></ul>
既存3D都市モデルの建物面積の活用	<b>航空写真以外も含めた3D都市モデルにより、変化の視覚的把握が容易になる</b> <ul style="list-style-type: none"><li>• 更新優先度マップの表現方法に既存の3D都市モデルの建物面積を用いることで、特に郊外の過疎地域での変化を視覚的に把握しやすくなる（郊外地域の更新を重視し、地域性を考慮した利用が可能）</li></ul>



## 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

今後の民間・自治体双方での活用に向け、プラットフォーム上での利便性向上、後続業務も含めた更新業務全体に対応可能なソリューション開発が必要。衛星画像の利用機会が増加することで検証も進む

項目	活用にあたっての課題
プラットフォーム搭載等による利便性向上	<p><b>ユーザーが衛星画像に容易にアクセスできる環境を整備し、PLATEAU・Tellus等の相互利用を進めるためには、プラットフォーム上にAPIやAIモデルを実装することが必要</b></p> <ul style="list-style-type: none"> <li>● 本実証では、限定地域でのALOS-3衛星シミュレーション画像によるAIモデル開発に留まっている</li> <li>● 今後、ALOS-3衛星に近い解像度の画像がTellusマーケット等のプラットフォーム上で流通するようになった際、ユーザーが容易に更新優先度マップにアクセスできるようになるためには、Tellus APIやアドイン等によってAIモデルもプラットフォーム上に実装することが必要</li> <li>● ユーザーが衛星画像にアクセスしやすい環境が整備できれば、PLATEAU・Tellus等の相互利用が大幅に促進されることが考えられる</li> </ul>
更新業務全体におけるソリューション提供	<p><b>変化箇所の把握後に必要となる高解像度画像の部分調達の支援や、より細かな属性・条件へ対応するためのソリューションが必要</b></p> <ul style="list-style-type: none"> <li>● 変化箇所の大まかな把握後に、民間事業者や自治体の地図データの実際の部分更新をする際には、高解像度衛星画像・航空写真アーカイブ画像の部分調達等がケース毎に必要な</li> <li>● そのため、ユーザーニーズに応じて更新業務全体におけるソリューションや提案をブラッシュアップする必要がある</li> <li>● 建物変化の属性付与（新築・滅失・増築や、商業施設・工場の情報）や、建物以外の道路・植生等の変化検出のニーズもあるので、詳細な属性付与や対象地物の拡大を図っていくことが期待される</li> </ul>
精度向上	<p><b>精度向上のために3D都市モデルの更に洗練された連携方法や、実際の衛星画像データを利用した検証の実施が必要</b></p> <ul style="list-style-type: none"> <li>● 変化検出のAIモデルの精度を継続的に高めていく必要がある</li> <li>● 2時期の画像だけを入力して変化検出するよりも、既存の3D都市モデルデータを追加することで、既知の建物箇所に対する変化検出も可能になり、変化抽出の精度向上が期待できる</li> <li>● また、ALOS-3衛星の実際の衛星画像を対象にした検証は本実証期間内では実施できなかったため、今後の検証を通して実利用を推進していくことが望まれる</li> </ul>

## V. 成果と課題 > 2. 今後の取り組みに向けた課題

# 【参考】ALOS-3衛星画像での3D都市モデルの更新可能性

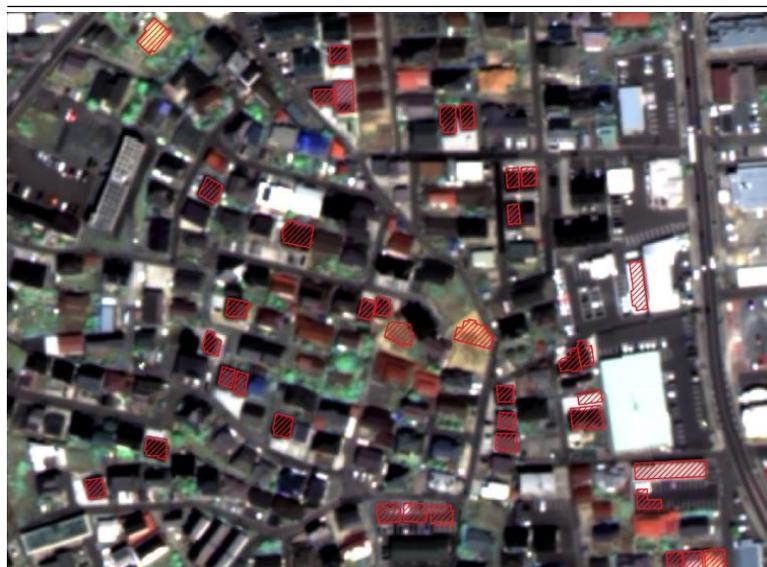
更新優先度マップでの変化箇所の把握後、地図データの実際の部分更新には高解像度衛星画像・航空写真アーカイブの部分調達等が必要になるが、今後の技術開発でALOS-3衛星に近い解像度の画像自体で代用できる可能性がある

福島県郡山市 密集した住宅街エリアのヒートマップ表示 (サンプル)

旧時期航空写真



新时期衛星画像+教師データ (  )



AI判読結果 ヒートマップ表示



現段階において、ALOS-3衛星画像の解像度では1棟1棟の区別にはまだ課題があるが、技術向上を図りながら表現方法の工夫 (例：ヒートマップ表示) も図りながら、ALOS-3衛星に近い解像度の画像自体でソリューション提供ができるようになることも期待される



# 用語集 (1/7)

用語	内容	
ア行	ASNARO-1衛星	<ul style="list-style-type: none"><li>2014年に打上げた国産の小型光学衛星</li><li>直下視で10km×10kmシーンを解像度50cmで撮影可能</li></ul>
	ESRGAN	<ul style="list-style-type: none"><li>2018年に発表された深層学習モデル</li><li>画像の高解像度化である超解像手法の一つとして利用されている</li></ul>
	EPSG	<ul style="list-style-type: none"><li>ESPG (European Petroleum Survey Group) によって作成された測地系、座標系、地図投影法等をコード体系にしたもの</li><li>各種GISソフト等で使用されている</li></ul>
	ウェブメルカトル	<ul style="list-style-type: none"><li>地図投影法の一つで、緯度経度が直行するが高緯度の面積が大きくなる特徴などがある</li><li>始めはGoogle社によって策定されたが、現在は多くのマップ (Bing Maps、地理院地図等) でも利用されている</li><li>ESPGコードでは3857</li></ul>
	ABCI	<ul style="list-style-type: none"><li>国立研究開発法人 産業技術総合研究所が構築・運営するAI向け計算プラットフォーム</li></ul>
	ALOS-3衛星	<ul style="list-style-type: none"><li>2023年運用開始を予定していた国産光学衛星</li><li>直下視で観測幅70kmの広域を解像度80cmで連続撮影可能</li></ul>
	MTF	<ul style="list-style-type: none"><li>Modulation Transfer Function (変調伝達関数) の略</li><li>レンズの性能を評価するパラメータで、被写体の持つコントラストをどの程度忠実に再現できるかを表現する</li></ul>
	オルソ処理	<ul style="list-style-type: none"><li>地形の凸凹や写真の傾きを補正すること</li><li>通常の写真画像は中心投影なので、撮影角度や地形の標高の影響により、歪んだ形状となっている</li><li>オルソ補正によりGISの背景データとして重畳できるようになる</li></ul>

# 用語集 (2/7)

用語	内容
カ行	回帰日数 <ul style="list-style-type: none"> <li>人工衛星の軌道において、同一の地点の上空までに衛星が戻るまでの日数</li> <li>このような軌道を準回帰軌道という</li> </ul>
	解像度 <ul style="list-style-type: none"> <li>地上解像度と同義</li> <li>衛星や航空写真において、画像上の一画素の一边を地上における距離で表したもの</li> </ul>
	観測幅 <ul style="list-style-type: none"> <li>人工衛星が軌道に沿って観測する際に、観測できる幅（軌道と直行方向における幅）</li> <li>走査幅ともいう</li> </ul>
	幾何補正 <ul style="list-style-type: none"> <li>地図データや画像等の歪みを補正し、他の地図データと重畳できるように加工する処理のこと</li> </ul>
	軌道 <ul style="list-style-type: none"> <li>人工衛星が地球を周回する際の筋道</li> <li>地球観測衛星は、極軌道（極を通る軌道）の一種である太陽同期準回帰軌道を利用している</li> <li>太陽同期軌道は軌道面と太陽のなす角度がほぼ一定の軌道であり、準回帰軌道は同一地点では同一時刻に観測される軌道を示す</li> </ul>
	基盤地図 <ul style="list-style-type: none"> <li>国土地理院が中心になって整備をしている電子地図</li> <li>国・地方公共団体・民間事業者等が地図整備をする際に基準とする地図</li> </ul>
	QGIS <ul style="list-style-type: none"> <li>Quantum Geographical Information Systemの略</li> <li>フリーの地理空間情報データの閲覧、編集、分析機能を有するクロスプラットフォームのオープンソースソフトウェア・GISソフト</li> <li>有料・高額なGISソフトに近い機能・操作性を備えており、機能の追加も無料のプラグインで行うことができる</li> </ul>
	教師データ <ul style="list-style-type: none"> <li>機械学習で学習や検証等に用いるためのデータセット</li> <li>画像を用いた教師データの場合は、画像データと画像データに対応したラベル（画像内の領域を分類したものや画像に名称を付けたり種類がある）を付与したペアになったデータセットが必要になる</li> </ul>

# 用語集 (3/7)

用語		内容
カ行	航空写真	<ul style="list-style-type: none"> <li>空中写真と同義</li> <li>空中から固定カメラを使用した垂直に撮影された画像で、複数の航空写真を標定してステレオ観測することで、地表物の位置や高さを計測して地形図を作成できるように撮影されたもの</li> </ul>
サ行	再現率	<ul style="list-style-type: none"> <li>リコール (Recall) と同義</li> <li>見逃しをしていない程度を把握するための指標で、「正しく陽性と予測できたデータ数」÷「実際に陽性であるデータ数」で計算される</li> <li>数値が高い方が、見逃しが少ないことを意味する</li> </ul>
	GeoJSON	<ul style="list-style-type: none"> <li>JSONを用いて空間データをエンコードし、非空間属性を関連付けるファイルフォーマット</li> </ul>
	自然分類	<ul style="list-style-type: none"> <li>クラス内の分散を小さくし、クラス間の分散を大きくするように分類クラスを設定することで、データの変化量が比較的大きい箇所に閾値を設定し分類する手法</li> <li>視覚的に判断しやすいことが特徴</li> </ul>
	深層学習	<ul style="list-style-type: none"> <li>現在の第3次人工知能ブームの中心的な技術</li> <li>画像認識・テキスト解析・音声認識等の複数の領域で、研究開発・実務利用が進んでいる</li> </ul>
	ズームレベル	<ul style="list-style-type: none"> <li>WebGISにおいて、地図上に表示される世界の範囲を決定する指標</li> <li>低いズームレベルでは広い範囲が大まかに表示され、高いズームレベルになるほど狭い範囲を詳細に表すようになる</li> </ul>
	図郭	<ul style="list-style-type: none"> <li>国土地理院によって整備された国土基本図を元に、ある基準に従って分割された個々の地図範囲</li> <li>国土基本図の図郭には地図情報レベルが5段階 (50000、5000、2500、1000、500) ある</li> <li>例えば、地図情報レベル2500では、東西方向2km・南北方向1.5kmの図郭になる</li> </ul>

# 用語集 (4/7)

用語	内容
タ行	<ul style="list-style-type: none"> <li>● 極軌道（極を通る軌道）の一種で、軌道面と太陽のなす角度がほぼ一定の軌道</li> </ul>
	<ul style="list-style-type: none"> <li>● 画像データの画素数を間引くことで、ファイルサイズを小さくする処理</li> <li>● オリジナル画像に比べて情報量が減ってしまうため、オリジナルの品質（見た目）をなるべく損なわないように、間引く画像処理方法は複数手法がある</li> </ul>
	<ul style="list-style-type: none"> <li>● 統計への利用を目的に、緯度・経度に基づいて日本全国の地域をすき間なくほぼ同じ大きさの網の目（メッシュ）に区分したもの</li> <li>● 一辺の長さが約80kmの1次メッシュから約250mの5次メッシュまでが定められている</li> </ul>
	<ul style="list-style-type: none"> <li>● 解像度と同義</li> <li>● 衛星や航空写真において、画像上の一画素の一辺を地上における距離で表したもの</li> </ul>
	<ul style="list-style-type: none"> <li>● 3次元である地球上の位置を緯度と経度で表現する座標系</li> </ul>
	<ul style="list-style-type: none"> <li>● Webメルカトル地図を256×256（もしくは512×512）ピクセルの正方形で切り分けた画像からなるタイル</li> <li>● 各タイル画像にはズームレベル（z）とX方向、Y方向の位置（x,y）が決められており、x, y, zの数値を指定することで一意のズームレベルおよび範囲を示すことができる</li> </ul>
	<ul style="list-style-type: none"> <li>● 解像度の低い画像の解像度を上げて、解像度がより高い画像を作成する手法</li> <li>● 様々な手法があるが、単画像を対象にした場合、深層学習モデルを応用した手法では、従来手法よりも画質が良い成果を得られる事例が多くなってきている</li> </ul>
	<ul style="list-style-type: none"> <li>● 人工衛星から地表面に対して、垂直に撮影すること</li> <li>● 単画像では、建物等の倒れ込みや画像の歪みを最小限度に抑えることができる</li> </ul>

# 用語集 (5/7)

用語		内容
タ行	適合率	<ul style="list-style-type: none"> <li>• プレシジョン (Precision) と同義</li> <li>• 誤検出をしていない程度を把握するための指標で、「正しく陽性と予測できたデータ数」÷「陽性と予測したデータ数」で計算される</li> <li>• 数値が高い方が、誤検出が少ないことを示す</li> </ul>
	Tellus	<ul style="list-style-type: none"> <li>• 日本発のオープン&amp;フリーの衛星データプラットフォーム</li> <li>• 無償・有償で衛星データ・API・アドインの利用が可能なマーケット環境である、Tellus Marketを提供している</li> </ul>
	投影法	<ul style="list-style-type: none"> <li>• 球形の地球を平面で表現するための変換手法</li> <li>• 曲面を平面で表現する完全な方法はないため、地図投影法には、距離を維持する方法、面積を維持する方法、方位を維持する方法等の多くの種類が存在する</li> </ul>
	都市計画基本図	<ul style="list-style-type: none"> <li>• 都市計画法第14条の規定による都市計画の図書としての総括図、計画図等の基本となる地形図</li> <li>• 地方公共団体の行政区域内の都市計画の内容を示しており、一般に、地形図に家屋形状・都市計画道路・用途地域等を色分けして表示される</li> <li>• 縮尺は、地方公共団体の面積によって異なる</li> </ul>
ナ行	入射角	<ul style="list-style-type: none"> <li>• 光がある媒質 (例：空気) から別の媒質 (例：地面・土) に入るときに、境界面で反射と屈折が起こる境界面が平面の場合、その法線と入射光線のなす角を入射角と呼ぶ</li> <li>• 入射角が小さい程、直下視に近い撮影角度になる</li> </ul>
ハ行	PyTorch	<ul style="list-style-type: none"> <li>• 深層学習モデル開発に利用される主要フレームワークの一つ</li> <li>• Facebookの人工知能研究開発グループによって開発された</li> </ul>
	波長帯	<ul style="list-style-type: none"> <li>• 波長域と同義</li> <li>• 光 (電磁波を含む) は、波長・周波数により可視光・近赤外・熱赤外・マイクロ波等に分類される</li> <li>• 人工衛星に搭載されている各センサは、対応する波長帯を観測するように設計されている</li> </ul>

# 用語集 (6/7)

用語	内容
八行	パンクロマチック <ul style="list-style-type: none"> <li>● 地形や人工構造物の形状をなるべく高い分解能で観測するために、太陽光の反射強度が高くなる波長帯（可視域から近赤外）を1つ設ける</li> <li>● この分解能が高い1つのバンドのこと</li> <li>● パンクロマチックバンド、パンクロバンドともいう</li> </ul>
	パンシャープン <ul style="list-style-type: none"> <li>● 分解能の高いパンクロマチックとカラーバンドであるマルチスペクトルバンドを合成して、パンクロマチックと同じ分解能のカラー画像を作成すること</li> </ul>
	反射率 <ul style="list-style-type: none"> <li>● 物体に照射する光の放射束に対してはね返った光の放射束の比</li> </ul>
	バンド <ul style="list-style-type: none"> <li>● 人工衛星の各センサから作成される画像が、各バンドに対応する</li> <li>● 1バンド画像はモノクロ画像になるが、RGBに3バンドを割り当てることで、トゥルーカラー画像や疑似カラー画像を作成することができる</li> </ul>
	分解能 <ul style="list-style-type: none"> <li>● 地上解像度と同義</li> <li>● 衛星や航空写真において、画像上の一画素の一边を地上における距離で表したもの</li> <li>● なお、波長分解能という呼称は、マルチスペクトルセンサの観測波長帯をどの程度細かく細分化できるかでセンサ性能を評価する際に使われる</li> </ul>
	分類手法 <ul style="list-style-type: none"> <li>● 連続変数データセットに対して、ある閾値を設けて、複数（通常は3～10程度）に分類すること</li> <li>● 分類手法には、等差分（閾値の間隔が同じ値）や等数分類（各分類のデータ数が同数）や自然分類（類似のデータをまとめて視覚効果を高めた分類）など複数が存在する</li> </ul>

# 用語集 (7/7)

用語		内容
マ行	マイクロ平均	<ul style="list-style-type: none"> <li>特定のクラスのデータ数の偏りを考慮せずに、すべてのクラスの結果を重み付けなく評価する平均の取り方</li> <li>クラスを超えて全体の精度を知ることができる</li> </ul>
	マルチスペクトル	<ul style="list-style-type: none"> <li>可視光・近赤外・熱赤外・マイクロ波等の波長帯毎に、複数のセンサで観測すること</li> </ul>
	メタデータ	<ul style="list-style-type: none"> <li>データセットに対して、データの属性や関連情報等を記載したデータ</li> <li>データの効率的な管理・検索を図る上で重要</li> </ul>
	メッシュ	<ul style="list-style-type: none"> <li>地図を決められた大きさに分割、格子状に区切ったもの</li> </ul>
ヤ行	U-Net	<ul style="list-style-type: none"> <li>画像から特徴を抽出する部分（エンコーダ）と、抽出された特徴をエンコーダと逆の処理を行うことで入力画像と同じサイズの確率画像を出力する部分（デコーダ）からなる画像の領域分割用の深層学習モデル</li> <li>各層において、エンコーダで抽出された特徴をデコーダに伝える構造をすることにより、精度を向上していることが特徴</li> <li>アルファベットの「U」のような形をしていることから、U-Netと命名された</li> </ul>
ラ行	ラスターデータ	<ul style="list-style-type: none"> <li>グリッド上に構成されたデータのこと</li> <li>通常の写真などの画像データは、ラスターデータになる</li> </ul>
	リモートセンシング	<ul style="list-style-type: none"> <li>対象物から離れた位置から各種センサにより、対象物の特性を計測する技術</li> <li>航空機や人工衛星に搭載した各種センサで地表面等を計測すること</li> </ul>
	レイヤ	<ul style="list-style-type: none"> <li>地図データでGISソフト上で一つ一つの主題毎に重ね合わせることができる個別データ</li> <li>道路レイヤ・建物レイヤ・河川レイヤのように分かれる</li> </ul>

# 3D都市モデルの更新優先度マップ<sup>o</sup> 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：株式会社パスコ**

本報告書は、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られません。