建物のエッジ検出に基づく映像と 2.5D 地図マッチング

粟野 友貴† 西村 拓哉†

†株式会社 NTT データ 〒135-8671 東京都江東区豊洲 3-3-9

E-mail: †{Yuki.Awano, Takuya.Nishimura}@nttdata.com

あらまし 自動運転などで用いられる自己位置推定において，GPS から取得した姿勢と位置を補正するために，画像内に映る建物と建物情報を含む地図をマッチングさせる手法の研究がされている．近年では，セマンティックセグメンテーションによって画像内の建物を抽出し，2.5D 地図とマッチングさせる手法がいくつか提案されている．しかしながら，前述の手法では，前方に障害物がある場合は建物が２つに分断されてマッチングできない．また複雑な形状をした建物は単純な形状情報しか含まない 2.5D 地図とマッチングできないといった課題がある．

そこで本研究では，建物のセグメンテーション結果を用いるだけでなく，建物の 4 隅について学習したエッジ検出モデルを用いることで，2.5D 地図に対応した情報を取得し，より多くの場面でマッチングさせることが可能な手法を提案する．

キーワード 深層学習，画像認識，ランドマーク

1. はじめに

当社では，都市空間情報の収集，分析を行っている[1]．建物に対して画像と地図情報をマッチングさせる手法を確立できれば，人やモノがどの建物周辺で活動しているか映像から認識できるようになり，都市空間上で起こる様々な分析に役立てることが可能となる．この建物のようなランドマークを検出して地図情報とマッチングさせる手法は，自動運転などにおける自己位置推定精度を向上させるための位置補正や，AR における広告の掲示領域推定を目的として研究されてきた．

マッチングに用いられる地図情報として，2.5D 地図上に登録されているテクスチャ情報のない建物が用いられる．画像から建物の外形および他の建物との位置関係を認識することができれば，この 2.5D 地図内の建物とマッチングすることが可能となる．しかしながら，実世界に存在する建物は多種多様で，表面の模様（テクスチャ）が複雑ということもあり，単純な方法で街中に複数ある建物を認識することは難しい．

建物のマッチングにおける課題は２つある．一つ目の課題として 2.5D 地図内の建物とマッチングさせるため，画像内の建物を個別に認識する必要があるが，前述したとおり建物のテクスチャが複雑で難しいことが挙げられる．従来手法では，画像の輝度勾配を元に直線を検出し，その中から建物のエッジを建物の幾何的特徴などから選抜される．この手法の場合，人の目で見て直線があることが分かっていても，木々などの障害物が直線を少しでも隠したり，想定していた条件と異なったりする場合，建物のエッジをとることが難しくなる．近年では，深層学習モデルを用いて建物の特徴形状を検出するモデルが提案されている[5]．具体的には，ピクセル単位で物体を認識するセマンティックセグメンテーションを用いて建物の領域，エッジを検出する．ただこの方法では，エッジもセグメンテーションとして検出されるため，エッジの端点が得られず，建物が密集していることでエッジが接する場合や，誤差による検出領域の途切れが存在すると建物の輪郭を把握することが難しくなる．2 つ目の課題として，画像内で検出した建物を 2.5D 地図とマッチングさせる際に度重なる反復計算が必要となる．従来手法では，GPS などのセンサで初期推定された姿勢と位置から，2.5D 地図の建物を画像内に投影し，画像内の建物のエッジとマッチングする．投影した建物の端とエッジを合わせる必要があるため，このマッチングには度重なる反復計算で位置の微調整が必要となる．

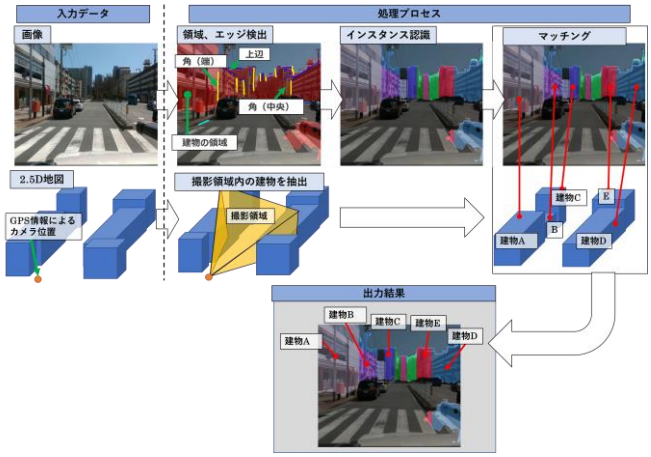

図 1 提案手法の概要

そこで本稿では，1 つ目の画像内の建物を検出する課題に対して，CNN ベースでエッジを検出するモデルを構築することで，建物の上辺，側面などを一つずつ検出し，画像内の建物の立体的な情報および個別の領域情報（インスタンス）を認識する手法を提案する．

また 2 つ目の課題に対して，従来手法と異なり，本稿では画像内の建物をインスタンスとして認識できている．そのため，投影した建物の面積と一定値以上重なり合うインスタンスをマッチングすればいいため，比較的少ない試行回数でマッチングすることが可能となる．本稿では一回の試行回数でもマッチングできることを確認した．図 1 に本提案手法の概要を示す．

以降，2 章にて関連文献，3 章に本稿の提案手法，4 章に実験内容と結果，5 章に結論を記載した．

2. 関連文献

画像と 2.5D 地図内の建物をマッチングする手法に関連する文献を本章で記載する．

[2]は建物の角をマッチング情報として用いるが，画像の水平方向に輝度勾配変化の大きい直線から角を取得する手法となっている．実際の場面では，建物のテクスチャが複雑であったり，障害物などが数多く存在したりするため，水平方向に輝度勾配の大きい箇所は角以外にも存在し，その中から角のみ取得することは難しい．[3]では建物上空の輪郭をマッチング情報として用いるが，建物が密集している場面では上空がほとんど見えない，また電線が通ることの多い日本の街中では正確に建物上空の輪郭を取るのは難しい．[4]では初期位置から周辺にある 2D 地図の建物の角群を画像に投影し，投影先が最も鉛直直線らしい組み合わせを任意範囲内で探索する．この手法では，画像内の建物の角の大半が他の地物と被ることなく見えて，かつ電柱などの鉛直に長く伸びる他の地物がないという条件を満たす必要がある．[5]では 2.5D 地図から画像へ投影した建物の領域とエッジがセマンティックセグメンテーションで検出した領域と被るように CNN モデルで位置の補正量を推定する．興味深い手法ではあるが，補正量も学習する必要がある．[6]では SLAM で生成した 3D 点群と 2.5D 地図の建物の差分が最小となるように探索する．この手法では，様々な地物が存在する街中では通用しない．

本稿では[7]で提案された四角形の物体検出手法を参考に，エッジ検出モデルを作成することで，従来手法では困難であった街中の建物のエッジを検出可能とし，セマンティックセグメンテーションで検出した建物の領域と組み合わせることで，建物のインスタンスを取得し，よりシンプルなマッチング手法を提案する．

3. 提案手法

入力画像I_{input}，2.5D 地図，GPS 情報からI_{input}内の建物と地図上の建物をマッチングさせる手法を本章で提案する．全体の流れとして，まず画像I_{input}内の建物を 1 つずつ検出するインスタンス認識を行う．次に，インスタンス認識された建物情報と GPS 情報から地図内の建物とマッチングする．

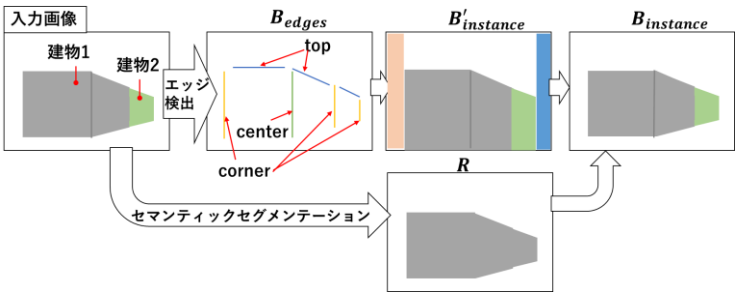

図 2　建物のインスタンス認識手順

ここで扱う 2.5D 地図とは，xy 軸平面を地面，z 軸を上空への方向とした時に，xy 軸平面上での建物の輪郭情報があり，その輪郭が一定の状態で z 軸方向への高さ情報を含み，建物表面のテクスチャ情報が”ない”建物のことを指す．

3.1. 建物のインスタンス認識

建物のインスタンス認識手法を記す．手順としては図 2 のように，画像内の建物をセマンティックセグメンテーションで検出する．次に建物のエッジを本稿で提案するエッジ検出モデルで検出する．最後に，セグメンテーション領域とエッジの関係から建物のインスタンスを推定する．

建物のセマンティックセグメンテーションには深層学習モデルの中で比較的小規模なモデルかつ高精度な予測が可能な U-Net[8]を用いた．

CNN ベースの物体検出モデルを用いて，建物のエッジを個別に検出し，その部位を認識する．一般的に，物体検出モデルは物体がどこにあるのか検出することは可能だが，バウンディングボックス（BB）内の細部の情報（ここではエッジの端点がどこにあるのか）までは検出する機能はない．また物体検出ではなく，エッジの端点を直接出力するモデルの場合，事前にエッジの数を指定する必要があることから，エッジの数が画像毎に変動する場合は用いることはできない．そこで，提案手法では[7]で提案された四角形の物体検出手法と同様に，物体検出モデルの regression 出力層に，BB 内の検出したい形状情報に関する特徴量変数を加える．

ネットワーク構造を図 3 に示す．ベースとした構造は Faster-RCNN[9] を用いた[7]と異なり，EfficientDet[10]を用いた．これは 2-stage である Faster-RCNN よりも 1-stage である EfficientDet の方が，学習負荷が小さいためである．EffidicientDet と異なる点として，図 4 に示すように regression 出力層にエッジの端点に関係する変数を加える．

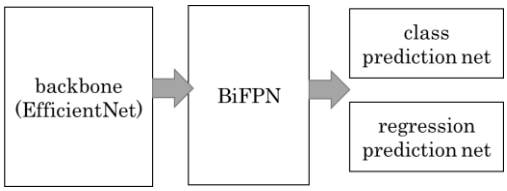

図 3　エッジ検出モデルのネットワーク構造．基本的な構造は EfficientDet と同一．BiFPN：weighted Bi-directional Feature Pyramid Network．規則的かつ効率的にマルチスケールの特徴量を融合するネットワーク．

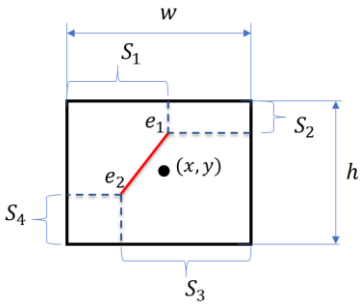

図 4　エッジ検出モデルの regression 出力層の変数 $\left(x, y, w, h, \frac{S_1}{w}, \frac{S_2}{h}, \frac{S_3}{w}, \frac{S_4}{h}\right)$．矩形に関する情報$(x, y, w, h)$とエッジの端点に関する情報$e_1 = \left(\frac{S_1}{w}, \frac{S_2}{h}\right), e_2 = \left(\frac{S_3}{w}, \frac{S_4}{h}\right)$を出力する

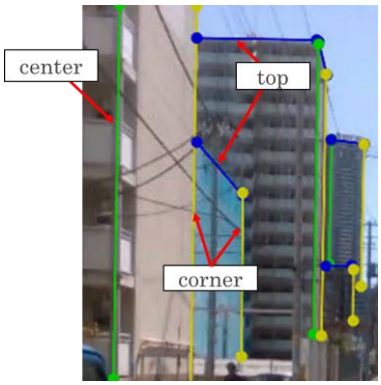

図 5　建物エッジのクラス．緑線：center，黄線：corner，青線：top

建物のエッジは図 5 のように建物の上辺(top)，視覚的に建物の端となる 4 隅（corner），視覚的に建物内側にくる 4 隅（center）の 3 種類に分類した．

セマンティックセグメンテーションおよびエッジ検出モデルから得られた建物の領域およびエッジの関係からインスタンスを認識する．まずは検出したエッジを元に建物の下辺以外の輪郭リストB_{edges}を推定する．

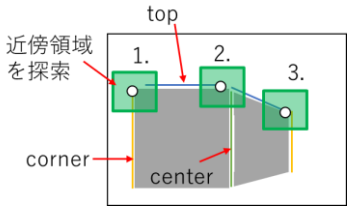

図 6　建物の上半分の輪郭認識手法．図中の数字は探索していく順番となる．

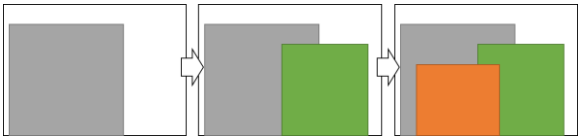

図 7　低い建物が前方に来ている時の輪郭認識手法．高さの大きい順に建物の領域を重ねていく．

図 6 のように corner, top, center で接続できそうなエッジ同士を探索する．左側の corner から探索をはじめて，右側の corner が見つかれば探索を終了し，B_{edges}に格納する．建物が見切れている場合，その建物の top はないため，見切れているという情報を登録し，corner のみ登録する．

輪郭リストB_{edges}が生成できた後，ここまでは建物の一番外側の輪郭しか認識されてなく，より低い建物が前面に来ているときの下側の輪郭を認識できていない．そこで，輪郭要素B^i_{edges}で囲まれる領域を$B'^i_{instance}$とし，図 7 のように$B'_{instance}$の中から，高さの大きい順に重ねていくことで，後からより低い建物領域に上書きされ，下側の輪郭が得られる．

最後に，エッジ群から得られた領域要素$B'^i_{instance}$とセマンティックセグメンテーションで得られた建物の領域Rを掛け合わせることで，建物のインスタンス$B^i_{instance}$を取得する．

3.2. 画像中の建物と 2.5D 地図中のマッチング手法

前節でインスタンス認識された建物と 2.5D 地図をマッチングする．まず GPS から取得した移動履歴から進行方向を推定し，進行方向上にある 2.5D 地図の建物を画像に投影する．投影された建物と最も重なる 3.1 の建物インスタンス$B^i_{instance}$をその建物として割り当てる．

進行方向dは GPS のnまでの移動履歴を用いて下式のように求める．移動履歴の新しい方が進行方向に寄与するように重みづけを行った．

$$d = \frac{1}{\sum w(n,i)} \sum_{i=1}^{n} w(n,i)(s_0 - s_i)$$

$$w(n,i) = n - i + 1$$

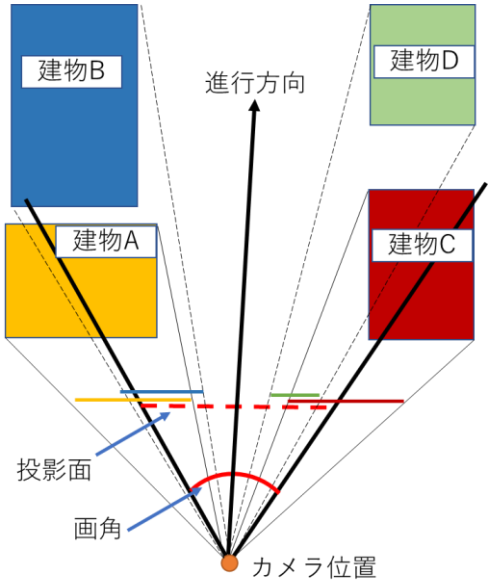

図 8　カメラ位置と 2.5D 地図から建物を 2 次元画像に投影

ここでs_0:現在位置，s_i:i ステップ前の位置である．

算出した進行方向d，2.5D 地図情報を用いて画像に建物を投影する．ここでカメラは水平に置かれていることとする．この仮定から事前に把握しているカメラの画角と現在位置と建物との位置関係から画像に建物を投影することが可能となる．図 8 のように，カメラ位置に進行方向に近く，法線方向からは遠い角とカメラ位置から進行方向に遠く，法線方向からは近い角の間を投影面に投影する．進行方向に遠い順に建物を画像に上書き投影していくことで,前後関係を維持する．

最後に，2 つの領域の重なり度合いを評価する指標である Intersection over Union (IoU)を用いて，投影された建物との IoU が最も高いインスタンス$B^i_{instance}$をその建物として割り当てる．

4. 実験内容と結果

建物のインスタンス認識結果，2.5D 地図とのマッチング結果について記載する．なお，検証用データとして，豊洲周辺で撮影した車載カメラ映像を用いた．撮影時にカメラが約 5° 時計回りに傾いていたため，画像を 5° 反時計回りに回転させてから認識をさせた．

建物の領域，エッジ検出モデルの学習についてまずは記載する．U-Net の学習には，オープンソースである ADE20K の中で建物を含むデータ(6049 枚)のみ用いた．エッジ検出モデルの学習には，関東圏の街中の画像 116 枚に対してアノテーションを実施し，学習を行った．

続いて建物のインスタンス認識結果について記載する．提案手法の比較対象としたベースラインは，画像に対して水平方向の平均輝度勾配が一定値以上の列を建物の端（エッジ）とし，提案手法で用いたセグメンテーションをエッジで区切った領域を建物のインスタンスとした．

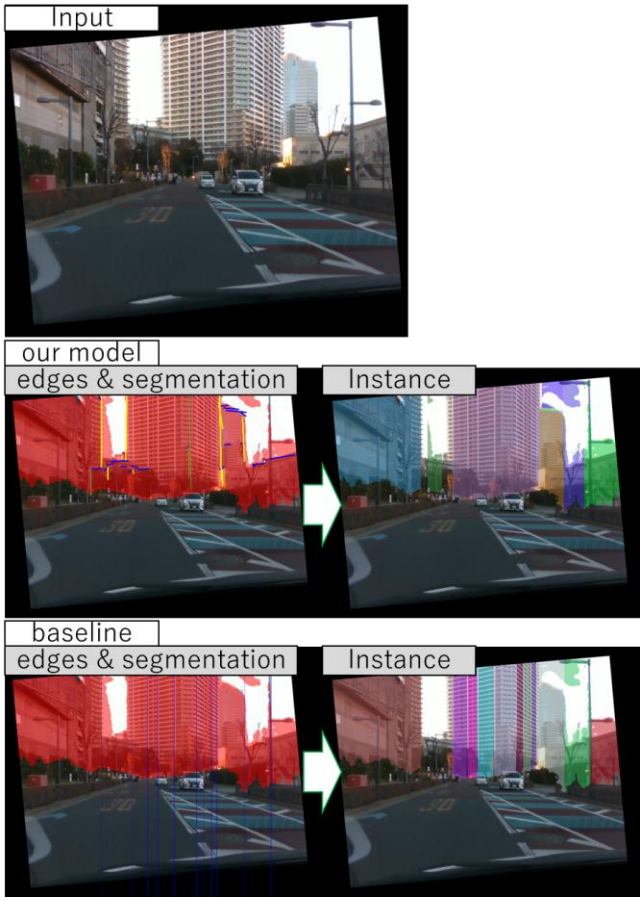

図 9　建物のインスタンス認識結果の例

提案手法とベースラインの検出例を図 9 に示す．ベースラインでは図 9 の Input 画像内の中央の建物（our model の Instance 結果で薄紫色の領域）のテクスチャをエッジとして何本か検出してしまっているが，提案手法では検出せずに建物の両端のみ検出していることが確認できる（建物左側はインスタンス認識時に一部欠けてしまっているが，その下側前方にある建物のエッジの影響を受けたため）．また Input 画像内の中央にある建物の右隣の建物（our model の Instance 結果で黄色の領域）に対して，提案手法では建物の高さを top エッジ（青線）検出結果から正しく認識できていることが確認できる．

インスタンス認識精度（検出率）を図 10 に示す．検証に用いた画像枚数は 42 枚である．検出率は正解ラベル（建物）の総数に対して，推定結果と正解ラベルとの Intersection over Union (IoU)が一定値以上超えた建物の割合とした．また画像の中には遠く離れた小さな建物も映っている場合があるが，本検出結果の用途

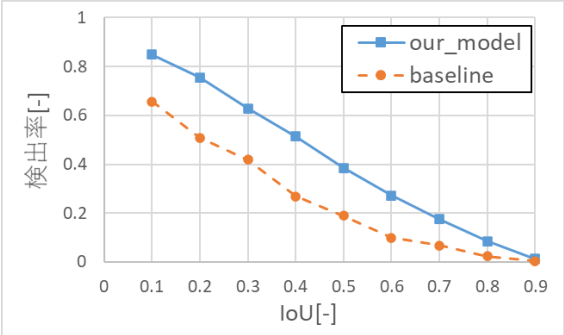

図 10　インスタンス認識精度．正解ラベルの総数に対して，推定結果と正解ラベルとの IoU が一定値以上超えた建物の割合を検出率とした．IoU が高いほど，正解ラベルとの重なりが大きい．

図 11 建物の区切りが明確でないシーン．左図：入力画像，中図：人によるアノテーション，右図：本手法のインスタンス検出結果

としては，画像内の中で比較的近く，サイズが大きい建物が主に活用される．そのため，評価対象となる建物は領域面積が画像面積サイズ 0.64%以上に限定した．図 10 から IoU が 0.4 の時，提案手法の方がベースラインよりも 24.5%検出率が高く，いずれの IoU でも提案手法の方がベースラインよりも精度が高いことを確認した．

図 11 の右側の建物のように画像からだけでは人の目でも建物の分かれ目，どこで隣接する建物の区切りがあるのか，分からないことがあった．そのような状況に対処するため，建物の領域を区分するエッジを検出する際に，カメラ位置周辺の地図情報もモデルに入力することが今後必要になると思われる．

図 12 に画像中の建物と 2.5D 地図中の建物をマッチングさせた結果を示す．なお，地図情報として OpenStreetMap を用いた[11]．最寄りの建物 3 つに対して，マッチングできることを確認した．

5. おわりに

エッジ検出結果に基づいた画像と 2.5D 地図の建物をマッチングさせる新規手法を示した．CNN ベースのエッジ検出モデルを用いることで，様々な地物が混在する街中でも建物のエッジを検出することが可能となった．また，検出エッジは建物の上辺か 4 隅なのか分類できていることから，建物の立体的な構造を把握す

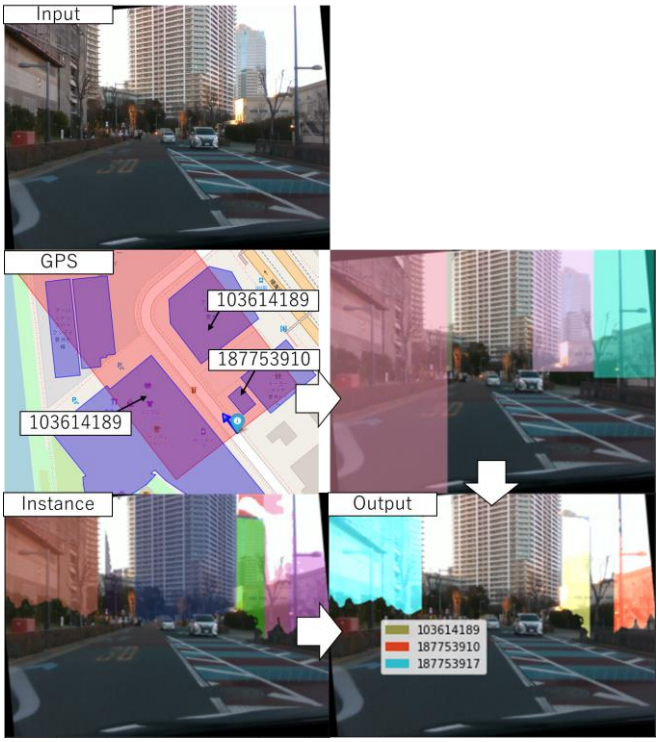

図 12　画像中の建物と 2.5D 地図中のマッチング結果（地図情報の図として OpenStreetMap を用いた）

ることができ，従来手法では得ることのできないインスタンスの認識が可能となった．インスタンス認識ができることによって，従来手法のように 2.5D 地図の建物を画像に投影することなく，よりもシンプルな方法でマッチングを行うことができるようになった．今後は，地図情報もエッジ検出モデルの入力情報として，より正確に建物を認識する手法を検討していきたい．

参　考　文　献

[1] “空間データ活用　～モノと位置とで見える未来～”, https://www.nttdata.com/jp/ja/data-insight/2020/033001/ (2021.02.11 確認)

[2] C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit, Instant Outdoor Localization and SLAM Initialization from 2.5D Maps, ISMAR, 2015.

[3] M. Bansal and K. Daniilidis. Geometric Urban Geo-Localization, CVPR, 2014.

[4] J.Yuan and A. M. Cheriyadat, Combining Maps and Street Level Images for Building Height and Facade Estimation, CVPR, 2016.

[5] A. Armagan, M. Hirzer, P. M. Roth and V. Lepetit, Learning to Align Semantic Segmentation and 2.5D Maps for Geolocalization, CVPR, 2017.

[6] R. Liu, J. Zhang, S. Chen and C. Arth, Towards SLAM-based Outdoor Localization using Poor GPU and 2.5D Building Models, ISMAR, 2019.

[7] Y. Xu, M. Fu, Q. Wang, Y. Wang, K. Chen, G. Xia and X. Bai, Glinding vertex on the horizontal bounding box for multi-oriented object detection,

TPAMI, 2020.

[8] O. Ronneberger, P. Fischer and T. Brox, U-Net: Convolutional Networks for Biomedical Image Segmentation, arXiv:1505.04597v1, 2015.

[9] S. Ren, K. He, R. Girshick, and J. Sun, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, PAMI, 2017

[10] M. Tan, R. Pang and Q. V. Le, EfficientDet: Scalable and Efficient Object Detection, CVPR, 2020.

[11] “OpenStreetMap ライセンス情報”, https://www.openstreetmap.org/copyright (2021.02.11 確認)