G-57

深層学習を用いた写真からの高精度地域推定

High-Accuracy Geolocation Estimation from Photos Using Deep Learning

平田 麟太朗† 篠崎 隆志‡ 井口 信和‡
Rintaro Hirata Takashi Shinozaki Nobukazu Iguchi

1. はじめに

オープンソースインテリジェンス（OSINT）は、セキュリティ分野に限らず、広く情報収集の手段として重要性を増している。その中でも、地理的情報を取得するタスクを GEOINT と言うが、これは法執行機関から企業、個人まで幅広い層に活用されうる技術である [1]。 GEOINT において、画像から位置情報を推定する技術は、写真サービスの位置推定、災害対応、犯罪捜査、歴史・文化研究、都市計画など、多岐にわたる分野での応用が可能である。近年、深層学習の進歩により、画像認識技術は飛躍的な進歩を遂げ、画像からの位置推定技術においても、章 2 で述べるように深層学習を用いた手法が主流となりつつある [2]–[8]。

しかし、従来の深層学習を用いた位置推定手法は、実際の位置と推定した位置の誤差が大きくなる問題を抱えており、実用的な精度を達成することが困難であった（表 1）。特に都市部以外の地域や、著名なランドマークが写っていない画像において誤差が大きくなる傾向にあった。これは、学習用データセットに地域的な偏りがあったこと（図 2）、そして位置推定に有効な細かな地理的特徴を捉えきれていなかったことなどが原因として考えられる。さらに、従来手法では、モデルの推論根拠が不明なままであり、推論結果の信頼性を評価することが困難であった。そのため、OSINT 等の現実世界と一致させるような用途において、推論結果をそのまま利用することに抵抗があるケースも見られた。これは、モデルの推論結果が、人間が理解しやすい形で提示されていないことが原因であると考える。

これらの課題を克服するため、本研究では、OpenStreetMap [9] や国勢調査 [10] 等の公開されている地理空間データを用いて、地域的な偏りが少なく（図 4）、位置推定に有効な地理的特徴を多く含む画像データセットを構築する。次に、Vision Transformer [11] を画像エンコーダに、Vicente らによる GeoCLIP で提案されているランダムフーリエ変換機構 [2] を位置エンコーダに用いたモデルでマルチタスク学習を行い、画像エンコーダーに地理的特徴を抽出するよう学習させる。最後に、学習済みモデルの画像エンコーダーとベクトルデータベースを用いて画像の特徴量を用いる検索システムを構築し、推論結果の根拠となる類似画像を提示することで、推論結果の信頼性を向上させる。

2. 関連研究

画像からの位置推定技術は、近年活発に研究されている分野であり、様々な手法が提案されている [2]–[8]。大きく分けて、検索ベースの手法と分類ベースの二つの手法に分けられる。

検索ベースの手法は、入力画像と類似する画像を大規模なデータベースから検索し、その画像のメタデータから位置情報を推定する手法である。Hays ら [12] が提案した IM2GPS は、手動で作成した特徴量を用いた画像検索によって位置情報を推定する手法であり、Vo ら [12] は、深層学習を用いて学習した特徴量を用いることで、IM2GPS の精度を向上させた。これらの手法は、データベースに含まれる画像が多いほど精度が向上するという利点を持つ。しかし、埋め込み作成の計算コストが高かったり、大規模なデータベースが必要だったりする点が課題として挙げられる。

一方、分類ベースの手法は、地球の表面を複数の地理的なセルに分割し、入力画像がどのセルに属するかを分類する問題として位置推定を行う手法である。最も先行した研究として、Weyand らによる PlaNet という手法がある [8] 。これは、畳み込みニューラルネットワークを用いた位置情報の分類をを提案している。Müller-Budack らは、PlaNet に階層的な分類とシーン認識を導入することで、精度を向上させた [5] 。分類ベースの手法は、画像検索ベースの手法と比較して、計算コストが低く、大規模なデータベースを必要としない点が利点である。しかし、セルの分割方法によって精度が大きく左右される点や、地理的な特徴を十分に考慮できていなかった点が課題として残る。

従来の両手法では、 章 1 で述べたように、特に都市部以外の地域において誤差が大きくなる傾向があった。この傾向は、Clark らの研究で作成された GWS15k というテストデータセット [6] を用いた結果からも明らかである。GWS15k は、全世界を均等にグリッドでサンプルしたデータセットである。このデータセットは都市部以外にもデータが分布しており、これを用いて検証されたモデルは軒並み精度が低い結果となっている（表 1）。そこで、本研究では分類ベースの手法と検索ベースの手法の両方を採用する。さらに、OpenStreetMap 等の地理空間データを用いて、より地理的な特徴を反映した高密度なデータセットを構築することで、この課題の解決を目指す。

表 1: GWS15k データセットにおける位置推定精度の比較（[13] から引用）

モデル	nkm 以内の正解率（%）		
	1km	25km	200km
PIGEOTTO [3]	0.7%	9.2%	31.2%
GeoDecoder [6]	0.7%	1.5%	8.7%
GeoCLIP [2]	0.6%	3.1%	16.9%
Translocator [7]	0.5%	1.1%	8.0%
ISNs [5]	0.1%	0.6%	4.2%

† 近畿大学情報学部, Faculty of Informatics, Kindai University
‡ 近畿大学情報学部/情報学研究所, Faculty of Informatics, Cyber Informatics Research Institute, Kindai University

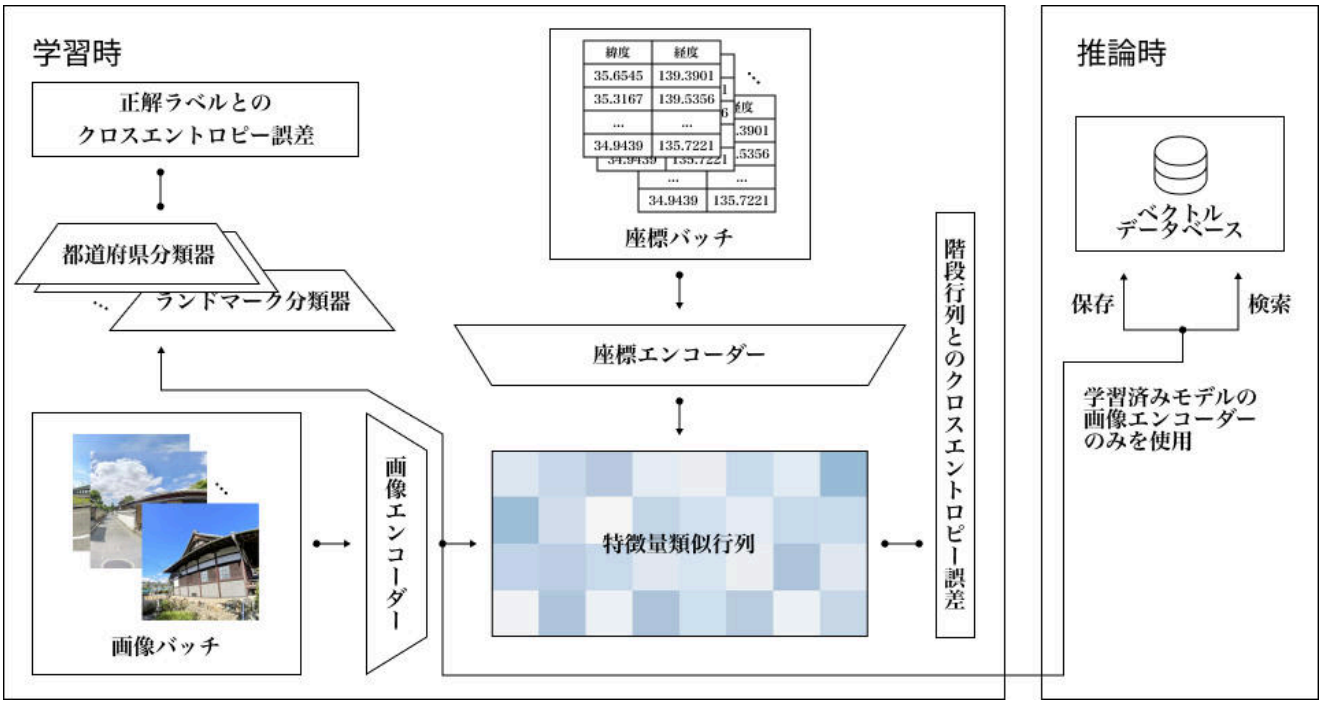

図 1: モデルのアーキテクチャ

3. 提案手法

従来の位置推定手法で用いられてきたデータセットは、地域分布に偏りがあり（図 2）、位置推定に有効な地理的特徴を含まない画像が多く含まれていた（図 3）。これらの問題点を解決するため、本研究では、OpenStreetMap のデータを用いて、高密度な学習用データセットを構築した。

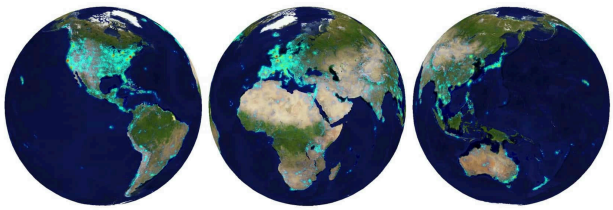

図 2: IM2GPS データセットの地域分布（[12] から引用）

図 3: 有効な地理的特徴を含まない画像の例

3.1. 高密度なデータセットの構築

高密度なデータセットを構築するために、Geofabrik [14] で公開されている日本全域の OpenStreetMap のダンプデータを活用した。このデータは PBF 形式であり、Quack-OSM [15] を用いて道路やランドマークを GeoPandas [16] の表形式に変換した。道路については長さと種類に応じて等間隔で、ランドマークについては等間隔グリッドになるように座標をサンプリングした。ただし、サンプリングする座標については少なくとも 5km 間隔以上、多くとも 500m 間隔以下にはならないようなアルゴリズムを開発し、慎重に点群を作成した。主要道路や大きいランドマークであればあるほどサンプル数は多くなるが、そうでない道路、ランドマークについても一定のサンプル数を確保するようにし、データの偏りをなるべく抑制することを目指した。最終的に道路やランドマークから作成した点群を一つにまとめ、近傍の点は DBSCAN アルゴリズム [17] を用いてクラスタリングし、各クラスタの重心を取ることで、最終的なサンプル点を作成した（図 4）。

図 4: 作成したデータセットの地域分布

次に、サンプルした点について、GoogleMap [18]、Mapillary [19] 等のストリートビュー画像データベースから、東西南北の画像を取得し、パノラマ画像として結合することで、より広範囲な地理的特徴を捉えられるようにした。さらに、先行研究で用いられてきた IM2GPS データセット [20], YFCC データセット [21]、 mp16 データセット [22] や、先行研究では使用されていなかった [23] を活用した。これらのデータセットはそれぞれ形式が異なるため、quality 値が 90 の webp 画像に再エンコードし、緯度経度を EXIF として埋め込む前処理を行った。章 3.2 で述べるように、本研究は Vision Transformer の Large モデルを凍結せずに学習するため、既存研究に比べて大幅に学習コストがかかると予想された。そのため、リソースの制限からこれらの中から日本国内の画像のみを抽出して学習に用いた。解像度については、学習時にランダムな場所、比率、サイズで切り出しを行うため、この時点では統一していない。

さらに、データ拡張として、画像のサイズに応じたランダムな切り出しと、ランダムなアフィン変換、一般的な画像拡張の適用、ランダムノイズの付加を TorchVision [24] を用いて行った。ランダムなアフィン変換は、角度変換なし、スケールは 1.0 倍から 1.1 倍の間でランダムに、せん断は-15 度から 15 度の間でランダムに適用した。一般的な画像拡張では、Torchvision のデフォルト設定である 3 つの変換を、同一画像に独立して 3 回適用し、3 枚の別々の拡張が適用された画像を生成、最後にその 3 つの画像を合成することで行っている。ランダムノイズは画像ごとに 50% の確率で適用され、各ピクセルの値を-10 から 10 の範囲で一様乱数で増減させている。これらのデータ拡張の設定値は、目視での確認によって地域的な特徴が損なわれない値を慎重に決定した（図 5)。

各画像には、国勢調査 [10] による町丁目単位の境界データ、OpenStreetMap の道路、ランドマークデータを空間結合し、ラベルを付与した。道路とランドマークに関しては、画像の位置から 20m 以内のものを隣接していると判定した。これは、目視で確認した結果、20m 以内であれば道路やランドマークの特徴が画像に現れていると判断したためである。隣接する道路やランドマークが複数存在する場合は、最も近いもの 1 つをラベルとして付与した。隣接していない画像についてはラベルを-1 として損失の計算から除外した。結果として 表 2 に示すようなラベルサンプル数の偏りが生じたデータセットを構築した。

表 2: データセットにおけるラベルサンプル数の偏り

ラベルの種類	標準偏差	最小数	最大数
都道府県	13.44	4480	117857
市区町村	522.18	1	13212
町丁目	5578.86	10	3340
道路	523.08	11	182
ランドマーク	1631.36	11	1490

この構築したデータセットを 80:15:5 の比率で学習用、バリデーション用、テスト用として分割した。バリデーション用、テスト用データセットは、正確な検証を行うためにも、すべての市区町村が同等の割合で含まれるように、慎重に分割した。最終的に 902858 枚の画像、容量にして 256GB のデータセットを構築した。

3.2. Vision Transformer と GeoCLIP を用いたマルチタスク学習

本研究では、Vision Transformer と GeoCLIP の位置エンコーダを用いたマルチタスク学習を行い、画像と位置情報を効率的に学習する（図 1)。

画像エンコーダには、HuggingFace [25] で timm/vit_large_patch14_clip_224.laion2b_ft_in12k_in1k [26] として公開されている Vision Transformer ベースの事前学習済みモデルを用いた。このモデルは、Large サイズの ViT をベースに、CLIP [27]、Laion [28]、ImageNet12K、ImageNet1K [29] で事前学習されており、約 300M パラメータを持つ。Vision Transformer (ViT)は一般的な CNN よりもデータセットの量に応じて精度がスケールされていると言われている [30] ことから、本実装により適切だと考えられる。本研究では ViT の Large モデルを用い、出力層の直前の 1024 次元の特徴ベクトルを取り出し、新規の全結合層を用いて 512 次元に圧縮したものを画像エンコーダーとしての出力とした。

位置エンコーダには、GeoCLIP で提案されているランダムフーリエ変換機構を用いた [2] 。この機構は、緯度経度をランダムフーリエ特徴に変換することで、地理的な近接性を高次元ベクトル空間上で表現することを可能にすると言わ

図 5: 作成したデータセットのデータ拡張済み画像の例

れている。ランダムフーリエ特徴の次元数は、GeoCLIP の論文で推奨されている 512 次元を用いた。位置エンコーダは 5 層の MLP で構成されており、各層の次元数は 1024 次元である。活性化関数には ReLU を用いた。GeoCLIP の位置エンコーダは、入力座標に対して複数の異なるスケールでランダムフーリエ特徴を計算することで、地理的な情報の階層表現を学習する。本研究では、GeoCLIP の論文で提案されている指数関数的なシグマ値割り当て戦略を採用し、シグマ値として 1, 4, 16, 64, 256 の 5 つを用いた。これらの値は大きいほど近傍の地理的特徴を区別でき、小さいほど長距離の地域的特徴を区別できため、日本の地理的スケールに合わせて設定した。キューサイズはは 8192 を用い、元論文よりも大きい値を採用した。

マルチタスク学習では、画像エンコーダの出力である 512 次元の特徴ベクトルを入力とし、都道府県、市区町村、町丁目、道路、ランドマークを推論する全結合層を作成した。この全結合層における出力層の次元数は、各タスクのクラス数に対応する。これにより、モデルは地理的な階層構造を学習し、より高精度な位置推定が可能になると期待できる。また、道路やランドマークのようなセマンティックな情報を学習することで、ガードレールや標識など、位置推定に有効な細かな特徴を捉えることができると考えられる。

モデル全体の損失として、各タスクのクロスエントロピー誤差の合計を用いた。この損失計算では、タスクごとの重み付けは行わなかった。クロスエントロピー誤差の計算では、各ラベルの出現数から重みを計算し、出現数が多いラベルの重みは低く、出現数の少ないラベルの重みは高くなるように設定した (1)。これにより、モデルの推論がデータセットの分布に偏りにくくなると期待できる。

$$w_i = \left(\frac{n_{\text{samples}}}{n_{\text{classes}} \cdot n_i} \right) \cdot \left(\frac{n_{\text{classes}}}{\sum_{j=1}^{n_{\text{classes}}} w_j} \right) \quad (1)$$

モデルの学習は、複数 CPU コアと複数 GPU を用いて、分散データ並列戦略で実施した。 最適化関数には RAdam [31] を用い、バッチサイズは 256、学習率は 1e-4、エポック数は最大 100 とした。早期終了の条件として、5 エポック連続して検証時の損失が改善しない場合は学習を終了するように設定した。1 エポック約 1500 ステップであり、実際には 74 エポックで収束した（図 6）。学習は NVIDIA 社製 GPU である A100 80GB を 2 台搭載した Linux ワークステーションで行った。CPU は AMD 社製 EPYC 7443P(24cores, 2.85GHz)、メモリ 512GB、OS は Ubuntu 20.04 LTS で、Python v3.10.13、PyTorch v2.3.1、TorchVision v0.18.1 を用い、数値精度には bf16-mixed を用いた。

最終的なモデル単体での予測精度は、都道府県で 76%、市区町村で 35%、町丁目で 11%、道路で 18%、ランドマークで 12% であった。

3.3. 学習済みモデルを用いた画像検索システムの構築

学習済みモデルの画像エンコーダを用いて、学習用、バリデーション用データセットの全画像について特徴量を推論し、緯度経度、画像パスのメタデータと共にベクトルデータベースに保存した。ベクトルデータベースには、Qdrant [32] を用いた。Qdrant は、ローカルにデータベースを保存することができ、コサイン類似度による検索が可能なオープンソースのベクトルデータベースである。Qdrant の選択理由は、軽量かつ高速であること、Python API が使いやすいこと、ローカル環境での運用が容易であることなどが挙げられる。

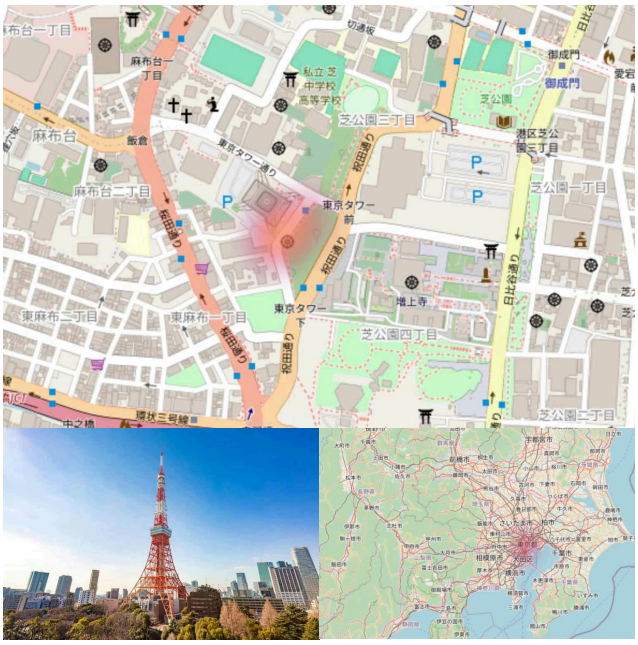

図 7: ヒートマップ表示の例
入力画像 [34] 、広域および詳細ヒートマップ

構築した画像検索システムでは、入力画像の特徴量を計算し、ベクトルデータベースに保存されている画像特徴量とのコサイン類似度を計算することで、入力画像と類似する画像を検索する。検索結果として、上位 100 件の埋め込みと、それに対応する緯度経度、画像パスを取得し、取得した緯度経度を用いて、folium [33] を用いてヒートマップを作成する（図 7）。ヒートマップは、類似画像の地理的な分布を可視化することで、モデルの推論根拠をユーザーに提示する。コサイン類似度を用いた理由は、高次元ベクトル空間において、ベクトルの方向が類似している画像ほど、意味的に近い画像である可能性が高いと考えられるためである。

4. 実験

構築したテストデータとベクトルデータベースを用いて位置推定精度を評価した。具体的には、まずテスト用画像をモデルに入力し、位置情報を推定する。次に、推定された位置

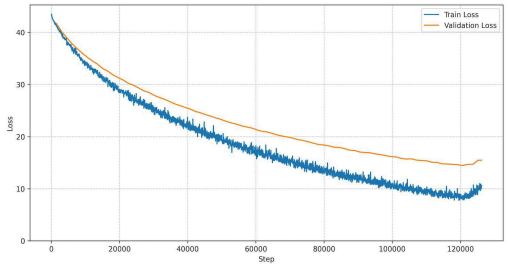

図 6: 学習時、バリデーション時の損失グラフ

情報に基づいて、ベクトルデータベースから類似画像を 100 件検索する。検索された類似画像の緯度経度から重心を計算し、これを最終的な推定位置とする。最後に、実際の座標との誤差距離を計算することで、位置推定精度を評価した（表 3）

誤差が 1km 以内の確率は 20.52%、25km 以内の確率は 83.65%、100km 以内の確率は 91.38% であった。これは、先行研究の他のモデルに比べて顕著な改善であると言える。従来手法では、都市部やランドマークの写っていない画像において誤差が大きくなる傾向があったが、本研究では、OpenStreetMap 等の地理空間データを用いた高精度な学習用データセットを構築することで、これらの問題を抑制し、高精度な位置推定を実現することができた。また、前章で述べたように、テストセットにすべての市区町村を含むデータを用いることで、よりモデルの性能を正確に測定できるようになった。

表 3: 作成したテストデータセットにおける精度の比較

モデル	誤差中央値	nkm 以内の正解率（%）		
		1km	25km	100km
OSV5M [4]	339km	0.3%	13.2%	27.9%
GeoCLIP [2]	413km	0.04%	1.3%	10.0%
本研究	**25km**	**20.5%**	**83.7%**	**91.4%**

本研究の偏り抑制能力を示すため、それぞれの都道府県から 1000 枚の画像をランダムにサンプルし、それぞれの推論結果の割合を計算した。その結果、上位 5 都道府県は 図 4 に示す通りである。また、東京都や大阪府といった都市圏はどちらも下位 2 位であった。また、47 都道府県全体での割合における標準偏差は 0.8325 であった。これにより、本研究は都道府県ごとの推論結果の偏りを抑制する能力があることが示された。

図 4: 都道府県ごとの推論結果の偏り（上位 5 都道府県）

都道府県	推論割合	$\frac{1}{47}$（2.13%）に対する比率
兵庫県	5.63%	264%
三重県	3.57%	168%
千葉県	3.35%	157%
茨城県	3.09%	145%
長野県	3.03%	142%

提案手法の実行時間は、画像 1 枚あたり平均 1.012±1.026 秒であった(n=100)。この実行速度は、実用的なアプリケーションに十分な速度であると考えられる。

5. 考察

本研究では、高精度な位置推定と推論根拠の明確化を実現する新たな手法を提案した。実験の結果、提案手法は従来手法を凌駕する精度を達成し、画像検索システムによって推論結果の根拠を明確化することができた。

提案手法は、従来手法と比較して、OpenStreetMap 等の地理空間データを用いた高精度な学習用データセットを構築することで、地理的特徴の類似性や学習用データセットの偏りによる誤差を抑制している点、Vision Transformer と GeoCLIP の位置エンコーダを用いたマルチタスク学習によって、画像と位置情報を効率的に学習し、地理的な階層構造を捉えることができている点、学習済みモデルの画像特徴量を用いた画像検索システムによって、推論結果の根拠となる類似画像を提示することで、推論結果の信頼性を向上させている点において優れている。特に、Vision Transformer と GeoCLIP の位置エンコーダを用いたマルチタスク学習が精度向上に大きく貢献していると考えられる。Vision Transformer は画像全体の特徴を効率的に学習し、GeoCLIP の位置エンコーダは地理的な近接性を高次元ベクトル空間上で表現することで、高精度な位置推定を実現している。また、道路やランドマークのようなセマンティックな情報を学習することで、ガードレールや標識など、位置推定に有効な細かな特徴を捉えることができていると考えられる。

本研究の限界として、夜間や悪天候時の画像、屋内の画像、航空写真などが学習データセットに含まれていないことから、これらの条件下では位置推定の精度が大きく下がる可能性がある。今後は、これらの条件下での位置推定精度を向上させるために、より大きなデータセットの利用や、より適切なデータ拡張、さらには画像セグメンテーションの適用による、画像解釈情報の追加などの手法を検討していく。

本研究では、リソースの制限から日本国内の画像のみを対象としたが、提案手法を他国に適用することで、世界規模での位置推定が可能になる。さらに、より多くの地理的特徴を考慮したり、時間帯、天候などの画像の撮影条件や、テキスト情報、都市データなどを考慮することで、さらなる精度向上が期待できる。

6. まとめ

本研究では、深層学習を用いた写真からの高精度地域推定手法を提案した。地理空間データを用いた高精度な学習用データセットの構築、Vision Transformer と GeoCLIP の位置エンコーダを用いたマルチタスク学習、学習済みモデルの画像特徴量を用いた画像検索システムの構築といった 3 つの要素を組み合わせることで、従来手法の課題であった位置推定精度の低さを大幅に改善した。また、推論根拠の不明確さも解決し、推論根拠を明確化することができた。

本研究の学術的な貢献は、地理空間データを用いて、地域的な偏りが少なく、位置推定に有効な地理的特徴を含む画像データセットを構築したこと、Vision Transformer と GeoCLIP の位置エンコーダを用いたマルチタスク学習を行い、画像エンコーダーに地理的特徴を抽出するよう学習させることで、高精度な位置推定を実現したこと、学習済みモデルの画像特徴量を用いた画像検索システムを構築し、推論結果の根拠となる類似画像を提示することで、推論結果の信頼性を向上させたことの 3 点である。

本研究は、GEOINT などのタスクにおいて非常に有望であるが、個々人のプライバシーに悪影響を与える可能性があるため、データセットやモデルの重みなどは慎重に公開を検討しなければならない。

提案手法をさらに発展させるためには、他国への適用、さらなる精度向上、他の技術との融合などが考えられる。本研究では、日本国内の画像のみを対象としたが、提案手法を他国に適用することで、世界規模での位置推定が可能になる。また、より多くの地理的特徴を考慮したり、時間帯、天候などの画像の撮影条件 や、テキスト情報、都市データなどを考慮することで、さらなる精度向上が期待できる。

参考文献

[1] Clark, R. M.: Geospatial Intelligence: Origins and Evolution, *ISBN:164712011X*, (2020).

[2] Cepeda, V. V., Nayak, G. K., and Shah, M.: GeoCLIP: Clip-Inspired Alignment between Locations and Images for Effective Worldwide Geo-localization, *arXiv:2309.1602*, (2023).

[3] Haas, L., Skreta, M., and Alberti, S.: PIGEON: Predicting Image Geolocations, *Computer Vision and Pattern Recognition (CVPR)*, (2023).

[4] Astruc, G., Dufour, N., Siglidis, I., Aronssohn, C., Bouia, N., Fu, S., Loiseau, R., Nguyen, V. N., Raude, C., Vincent, E., Xu, L., Zhou, H., and Landrieu, L.: OpenStreetView-5M: The Many Roads to Global Visual Geolocation, *Proceedings of the European Conference on Computer Vision (ECCV)*, (2024).

[5] Muller-Budack, E., Pustu-Iren, K., and Ewerth, R.: Geolocation Estimation of Photos using a Hierarchical Model and Scene Classification, *Proceedings of the European Conference on Computer Vision (ECCV)*, (2018).

[6] Clark, B., Kerrigan, A., Kulkarni, P. P., Cepeda, V. V., and Shah, M.: Where We Are and What We're Looking At: Query Based Worldwide Image Geo-localization Using Hierarchies and Scenes, *Computer Vision and Pattern Recognition (CVPR)*, (2023).

[7] Pramanick, S., Nowara, E. M., Gleason, J., Castillo, C. D., and Chellappa, R.: Where in the World is this Image? Transformer-based Geo-localization in the Wild, *Proceedings of the European Conference on Computer Vision (ECCV)*, (2022).

[8] Seo, P. H., Weyand, T., Sim, J., and Han, B.: CPlaNet: Enhancing Image Geolocalization by Combinatorial Partitioning of Maps, *arXiv:1808.0213*, (2018).

[9] OpenStreetMap, URL: https://www.openstreetmap.org,

[10] 2020 年度国勢調査, URL: https://www.e-stat.go.jp,

[11] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N.: An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, *International Conference on Learning Representations (ICLR)*, (2021).

[12] Vo, N., Jacobs, N., and Hays, J.: Revisiting IM2GPS in the Deep Learning Era, *arXiv:1705.04838*, (2017).

[13] GWS15k Benchmark (Photo geolocation estimation) | Papers With Code, URL: https://paperswithcode.com/sota/photo-geolocation-estimation-on-gws15k,

[14] Geofabrik, URL: https://www.geofabrik.de,

[15] QuackOSM, URL: https://kraina-ai.github.io/quackosm,

[16] GeoPandas, URL: https://geopandas.org,

[17] Ester, M., Kriegel, H., Sander, J., and Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise, *ISBN:1-57735-004-9*, (1996).

[18] Google Maps, URL: https://www.google.com/maps,

[19] Mapillary, URL: https://www.mapillary.com,

[20] IM2GPS DataSet, URL: http://graphics.cs.cmu.edu/projects/im2gps,

[21] YFCC DataSet, URL: https://multimediacommons.wordpress.com/yfcc100m-core-dataset,

[22] MP16 DataSet, URL: http://www.multimediaeval.org/mediaeval2016,

[23] Google Landmarks V2 DataSet, URL: https://github.com/cvdfoundation/google-landmark,

[24] TorchVision, URL: https://pytorch.org/vision/stable,

[25] Hugging Face, URL: https://huggingface.co,

[26] PyTorch Image Models, URL: https://timm.fast.ai,

[27] Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., and Sutskever, I.: Learning Transferable Visual Models From Natural Language Supervision, *arXiv:2103.0002*, (2021).

[28] Laion, URL: https://laion.ai,

[29] Deng, J., Dong, W., Socher, R., Li-Jia, L., Li, K., and Fei-Fei, L.: ImageNet: A large-scale hierarchical image database, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2009).

[30] Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D.: Scaling Laws for Neural Language Models, *arXiv:2001.08361*, (2020).

[31] Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J.: On the Variance of the Adaptive Learning Rate and Beyond, *arXiv:1908.03265*, (2019).

[32] Qdrant, URL: https://qdrant.ai,

[33] Folium, URL: https://python-visualization.github.io/folium,

[34] アソビュー！, URL: https://www.asoview.com/base/155288,