Image2GPS: Predicting Geographic Location from Images

Team Members: Michelle Brown, Anushka Sheoran, Anoushka Menon

CIS 4190/5190: Applied Machine Learning

Fall 2025

1. Introduction

This project addresses the Image2GPS task from CIS 4190/5190 Applied Machine Learning: predicting the geographic location (latitude, longitude) at which an image was captured. The problem is framed as a supervised regression task, where visual cues such as buildings, walkways, greenery, and lighting conditions are leveraged to infer location. Our work focuses on building a robust image-based geolocation system under limited data constraints, while exploring how transfer learning and careful optimization can significantly improve performance.

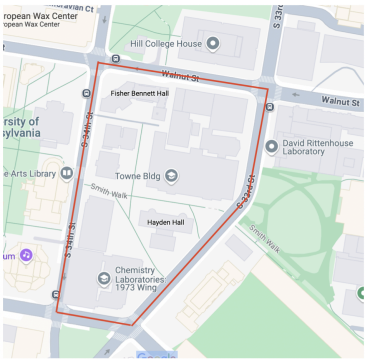
We begin with a simple ResNet-based baseline that establishes a reference point for image-to-GPS regression. From this starting point, we iteratively improve model design by introducing stricter data splitting, safer data augmentations, validation-based checkpointing, and stronger optimization strategies. These changes lead to substantial reductions in localization error, demonstrating that methodological rigor can yield large gains even without increasing model complexity.

Building on this foundation, we explore higher-capacity architectures and alternative inference paradigms. In particular, we evaluate deeper convolutional backbones (ResNet-50) and a modern ConvNeXt architecture, examining the trade-offs between model capacity, regularization, and generalization. Finally, we extend beyond pure end-to-end regression by incorporating a feature-based k-nearest neighbors (k-NN) retrieval approach, which leverages learned visual embeddings to improve robustness, especially under distribution shift.

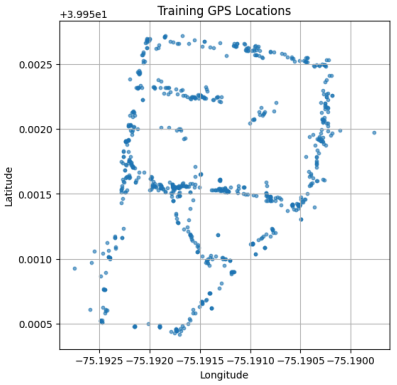
Across these experiments, we evaluate models using both coordinate-space error metrics (MAE and RMSE in degrees) and geodesic distance in meters, providing interpretable measures of real-world localization accuracy. Our best-performing models reduce average localization error by more than 50% relative to the baseline and demonstrate improved stability on an external dataset, highlighting the benefits of combining representation learning with retrieval-based inference. Overall, this project illustrates a clear progression from baseline regression to more robust hybrid approaches, offering practical insights into how architectural choices, training protocols, and evaluation strategies affect performance in real-world image geolocation tasks.

2. Core Components

2.1 Data Collection



We curated a custom dataset of images captured on the University of Pennsylvania campus, from 33rd and Walnut to 34th and Spruce st, delineated to the right. Images were taken using standard iPhone cameras in a controlled manner (vertical orientation with no zoom) and paired with GPS coordinates extracted from EXIF metadata. Multiple viewpoints were captured at the same location to improve robustness, by rotating while capturing images in the same location.



To ensure data quality, we removed blurry or low-resolution images, images with missing or inconsistent EXIF metadata, tilted images, and images whose main focus were cars or humans. Some images were converted to JPG (from HEIC) format to ensure consistency. For each image, EXIF GPS metadata is extracted using the exifread library. Latitude and longitude values, originally stored in degrees–minutes–seconds (DMS) format, are converted to decimal degrees. The extracted GPS information is saved to a structured CSV file (picture_metadata.csv) containing the image filename and its corresponding latitude and longitude. A subset of samples was manually inspected to verify correct alignment between images and GPS labels. Before training, we performed a sanity check on the dataset by plotting the GPS coordinates of all

training images. The resulting latitude–longitude scatter plot to the right matched the expected campus layout, confirming correct label extraction and alignment.

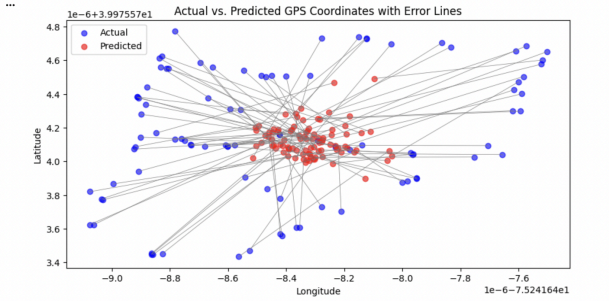
The full dataset is randomly shuffled and split into training (80%), validation (10%), and test (10%) subsets using a fixed random seed to ensure reproducibility. For each split, images are copied into separate directories, and a corresponding metadata.csv file is created that maps image filenames to GPS coordinates. This directory-based structure enables straightforward loading with standard dataset utilities. Finally, the processed dataset, containing images, split-specific metadata files, and a README documenting dataset structure and intended use, is uploaded to the Hugging Face Hub. This allows the dataset to be easily shared, versioned, and loaded using the Hugging Face datasets library for model training and evaluation

2.2 Model Design Considerations

2.2.1 Baseline Model

As a baseline, we trained a ResNet-18–based image-to-GPS regression model that predicts continuous latitude and longitude coordinates from input images. The final fully connected layer was modified to output two values, and the model was trained end-to-end using mean squared error (MSE) loss on normalized GPS coordinates. Images were resized to 224×224 and normalized using ImageNet statistics, with standard data augmentations applied during training.

Model performance was evaluated by converting predictions back to real-world coordinates and computing geodesic distance in meters. On the test set, the model achieved an average localization error of 81.70 meters from the true GPS location. In coordinate space, this corresponds to a mean absolute error (MAE) of approximately 0.000535° and a root mean squared error (RMSE) of approximately 0.000653°. These results establish a reasonable baseline and demonstrate that visual features contain meaningful geographic information, while leaving substantial room for improvement with more advanced architectures or training strategies. The image shows that predicted values cluster towards the middle (mean of the points) compared to the actual values.

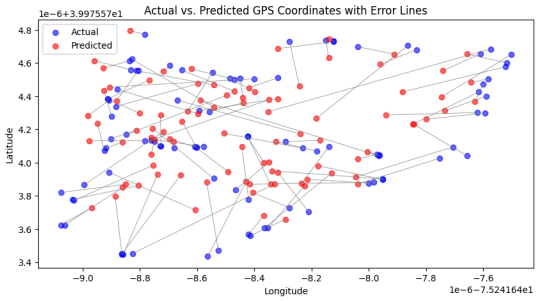


2.2.2 Baseline Refinements (ResNet-18)

As an initial exploration, we refined the baseline ResNet-18 model in several ways involving data handling, training strategy, and evaluation rigor. Unlike the baseline, which effectively trains and evaluates using a single test split, this model uses three distinct splits (train, validation, test). Model selection is performed based on validation RMSE, and only the best-performing checkpoint is evaluated on the held-out test set. This avoids information leakage and provides a more reliable estimate of generalization performance.

The baseline applies aggressive augmentations such as random cropping, rotations, and flips. In contrast, this model uses minimal and geographically safe augmentations, limited primarily to color jitter, while preserving spatial structure. This is important for a geolocation task, where excessive spatial transformations can distort location cues rather than improve robustness.

This model uses a lower learning rate, weight decay, and a more aggressive learning rate decay schedule, resulting in more stable convergence. Training runs for more epochs, and the model explicitly tracks and retains the best validation checkpoint, rather than simply using the final epoch as in the baseline. The code supports both full fine-tuning and a freeze–unfreeze training strategy, where the ResNet backbone is initially frozen and later



unfrozen for gentle fine-tuning. Although experiments showed that full fine-tuning performed best, this framework allowed systematic evaluation of training strategies beyond the baseline's single fixed approach.

Beyond standard coordinate-space MAE and RMSE, this model consistently reports geodesic distance errors in meters, including average and median distance. It also evaluates performance on the external Hugging Face dataset, providing an additional test of out-of-distribution generalization that was not present in the baseline. Relative to the baseline (81.7 m average error), the improved ResNet-18 model cuts localization error by over half on the test set while maintaining reasonable performance under external distribution shift.

2.2.3 ResNet-50

To assess whether increased model capacity improves geolocation accuracy, we evaluated deeper architecture ResNet-50, extending the refined ResNet-18 setup. Compared to ResNet-18 (11.7M parameters), ResNet-50 (25.6M) substantially increased model capacity and learnable parameters, and increased depth from 18 to 50 layers.

We compared two fine-tuning strategies for ResNet-50: baseline, where the entire pretrained network was fine-tuned from the start, and freeze–unfreeze, where only the regression head was trained initially before unfreezing the full model. After a hyperparameter grid search, the baseline ResNet-50 configuration used AdamW with a learning rate of 3e-4, weight decay 1e-4, a StepLR scheduler (step size = 4, $\gamma = 0.1$), and was trained end-to-end for 12 epochs. For the freeze–unfreeze strategy, Stage 1 trained only the final regression head for 4 epochs using AdamW (lr = 5e-4, weight decay = 1e-4) with StepLR (step size = 3, $\gamma = 0.1$), followed by Stage 2 fine-tuning the full network for 8 epochs at a reduced learning rate of 3e-4. In both cases, the best model checkpoint was selected based on validation RMSE in meters computed using geodesic distance.

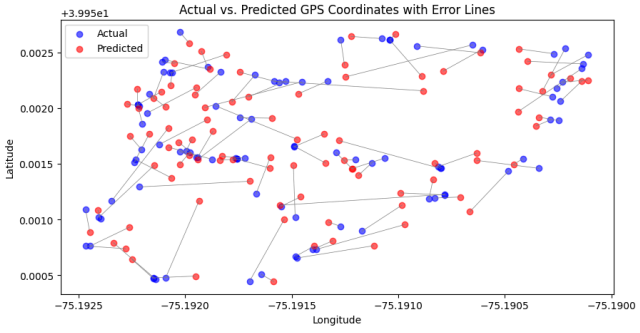
The baseline ResNet-50 model showed steady convergence, with validation RMSE decreasing from 99.16 m at epoch 1 to a best value of 45.46 m. On the held-out test set, this model achieved an average geodesic error of 30.69 m (RMSE 0.000268°), while generalizing to an external dataset with an average error of 60.36 m. Across all ResNet-50 experiments, this baseline configuration produced the lowest and most stable validation and test errors, making it the most optimized ResNet-50 variant evaluated.

2.2.4 ConvNeXt + k-NN Model:

While the ResNet-50 model is a pure end-to-end regression model, this final approach reframes the problem as a hybrid feature-learning + non-parametric retrieval task. The ResNet-50 backbone is replaced with ConvNeXt-Tiny, a modern convolutional architecture inspired by Vision Transformers. ConvNeXt uses larger receptive fields, depthwise convolutions, and simplified normalization, producing stronger global visual representations than traditional ResNet architectures for the same input resolution.

This model supports two inference modes: direct regression, where the ConvNeXt head predicts latitude and longitude (similar to ResNet-50), and feature-based k-nearest neighbors (k-NN), where the model first extracts high-level image embeddings and then predicts GPS coordinates by averaging the locations of the k most similar training images in feature space.

Furthermore, the k-NN component is non-parametric. Predictions depend directly on the training data at inference time, allowing the model to adapt better to novel images that resemble known locations, reduce large regression errors by anchoring predictions to nearby examples, and trade bias for variance through the choice of k.



On the external dataset, k-NN inference outperforms direct regression across all reported metrics. The average geodesic error drops from 49.93 m to 44.44 m, while the median error decreases substantially from 42.84 m to 30.91 m. This large reduction in median error indicates that k-NN predictions are more stable and less prone to large outliers, suggesting improved robustness under domain shift.

On the project's held-out test set, both methods perform strongly, but k-NN still provides a modest improvement in average error, reducing it from 24.39 m to 22.98 m. Median errors are nearly identical (19.45 m vs. 19.69 m), indicating that for in-distribution data, the learned regression head already performs near optimally, with k-NN offering only marginal gains.

3. Exploratory components

3.1 Code

The primary motivation for our code-level exploration was to improve model performance and stability under limited data, where architectural changes alone may not be sufficient. First, we conducted hyperparameter grid searches over learning rate, weight decay, and learning rate decay schedules. We evaluated combinations of Adam, AdamW, and different decay factors, selecting configurations based on validation RMSE in meters rather than training loss alone. This validation-driven selection consistently led to more stable convergence and lower geodesic error.

Second, we experimented with regularization strategies, particularly dropout in the final fully connected regression head for higher-capacity models such as ResNet-50. Dropout was implemented only at the head level to avoid disrupting learned spatial features in earlier layers. This reduced overfitting and improved validation performance, especially when using deeper backbones.

Third, we evaluated alternative loss functions, including Huber loss, which is less sensitive to large errors than mean squared error. This experiment was motivated by the presence of occasional large localization errors that could disproportionately influence training. While Huber loss improved robustness to outliers in some settings, mean squared error ultimately provided more consistent validation performance and lower geodesic distance on the test set, and was therefore retained for the final models.

Finally, we implemented training control mechanisms such as validation-based checkpointing and learning rate scheduling. Instead of defaulting to the final training epoch, we selected models based on the lowest validation RMSE.

3.2 Technique

Our exploration incrementally increased model capacity and studied how different inductive biases affect geolocation under limited data. Starting from a refined ResNet-18 baseline, we emphasized geographically safe image augmentations and validation-based model selection, which reduced average localization error by over 50% relative to the original baseline.

Scaling to ResNet-50 tested whether deeper residual representations improve spatial reasoning; while validation RMSE improved steadily during training, final gains were moderate, indicating diminishing returns from depth alone in a geographically constrained setting. Motivated by recent literature showing that learned visual embeddings often cluster semantically and spatially similar images, we introduced a ConvNeXt + k-NN hybrid approach that decouples representation learning from prediction.

By performing non-parametric k-NN retrieval in feature space, this method anchors predictions to visually similar training examples, substantially reducing median error and improving robustness under domain shift. Overall, results show that while deeper regression models improve performance up to a point, feature-space retrieval provides complementary benefits, particularly in reducing large errors and stabilizing predictions on out-of-distribution data.

Our final model constitutes a hybrid ensemble that combines a parametric deep network with a non-parametric retrieval component. The ConvNeXt backbone serves as a learned feature extractor, and k-NN performs instance-based inference by combining predictions from similar training examples. This approach ensembles learned representations and data-driven retrieval, leveraging the strengths of both regression-based prediction and example-based reasoning. This design improves robustness and reduces large errors, particularly under distribution shift, while retaining strong in-distribution performance.

Model	Architectural Components	Test Avg Error (m)	Test Median Error (m)	External Avg Error (m)
Baseline ResNet-18)	Shallow residual CNN with BasicBlocks and direct end-to-end coordinate regression	~81.7	—	—
Refined ResNet-18	ResNet-18 backbone with a regression head, paired with geographically safe image augmentations and validation-based checkpoint selection	~40	—	—
ResNet-50	Deeper residual network using bottleneck blocks (1×1–3×3–1×1 convolutions) to increase representational capacity	30.69	27.57	60.36
ConvNeXt + kNN	Hybrid architecture combining learned ConvNeXt embeddings with non-parametric k-nearest neighbor retrieval in feature space.	24.39	19.45	49.93

3.3 Comparison

State-of-the-art approaches to image-based geolocation, such as Google's PlaNet model (Weyand et al., 2016), achieve extremely strong performance by reframing geolocation as a classification problem over discretized geographic regions. PlaNet is trained on approximately 29.7 million geotagged photo albums from Google+, totaling hundreds of millions of images, and leverages large-scale deep networks to achieve what the authors describe as "sometimes superhuman" localization accuracy.

While highly effective, such methods depend on massive labeled datasets and substantial computational resources, which limits their applicability in constrained or custom-data settings. In contrast, our approach focuses on small-scale, user-collected data, using fewer than 1,000 images and a regression-based ResNet-50 pipeline. Rather than attempting global-scale geolocation, our model targets local or regional prediction, where fine-grained spatial distinctions are more relevant than worldwide coverage.

Despite its simplicity, our method incorporates several core ideas common in state-of-the-art systems, including transfer learning from large image models, progressive fine-tuning, and spatially meaningful evaluation metrics. Across both in-distribution and external datasets, our model achieves average localization errors in the range of approximately 30–70 meters, depending on domain shift. These results demonstrate that many of the benefits of state-of-the-art geolocation models can be retained even in low-resource settings, without requiring large-scale data collection or complex classification pipelines.

Overall, this comparison highlights a trade-off between global accuracy at scale and practical deployability, positioning our approach as a lightweight and effective alternative for localized geolocation tasks and research scenarios where massive datasets are unavailable.

4. Project contributions

All team members participated in data collection, taking photographs on our phones that were combined into the final dataset, and collaborated throughout the project on model design and evaluation.

Anushka managed dataset preparation for submission, handled leaderboard uploads, and tracked experimental progress across model iterations.

Anoushka organized and synthesized project progress, maintaining a centralized reference document used throughout experimentation and report writing.

Michelle led model implementation and integration, developing the unified Jupyter notebook that consolidated all architectural changes, hyperparameter tuning experiments, optimization strategies, and exploratory components.

5. Extra credit

Link to video: https://www.youtube.com/watch?v=Ot_5HKtLQQU

6. Reference

Weyand, T., Kostrikov, I., & Philbin, J. (2016). *PlaNet—Photo geolocation with convolutional neural networks*. arXiv preprint arXiv:1602.05314.
https://arxiv.org/abs/1602.05314

Link to our dataset: https://huggingface.co/datasets/AnoushkaMenon/CIS519_Image2GPS
Leaderboard submission alias: ama_test

Link to GitHub repo: https://github.com/asheorann/Image_Geolocation_ML_Model/