Training Models For GeoGuessr

1st Nicholas Scarfe
University of Adelaide
Adelaide, Australia
Nicholas.scarfe@gmail.com

***Abstract*—The task of predicting the geographic location of an image based solely on its visual content is a challenging open problem in computer vision. In this study, we investigate whether geotagged imagery can be used to train neural networks to regress latitude and longitude directly from input images. Using the OpenStreetView-5M dataset, which contains over five million street-view images with associated GPS coordinates and timestamps, we trained several neural network architectures, including feed forward network, convolutional network, and a transformer network. All images were resized to 224×224 for consistency and computational feasibility, and temporal metadata was incorporated into the models. Training was optimized using mean squared error (MSE) loss while recording mean absolute error (MAE) for interpretability, with trigonometric decompositions applied to longitude to address discontinuities at the antimeridian. Experimental results show that fully connected networks suffered from instability and overfitting, while CNNs achieved moderate accuracy with limited benefit from trigonometric transformations. The vision transformer consistently outperformed other models, achieving an MAE of 30.3 degrees despite limited fine-tuning. After fine-tuning the transformer model it reached a MAE of 19.5 degrees. Our work demonstrates the feasibility of regression-based approaches for image geolocation and provides initial benchmarks for incorporating temporal metadata in large-scale geospatial prediction.**

I. Introduction

A. Context

Geotagged images contain both visual content and associated geographic coordinates, making them a valuable resource for location-based tasks. With the emergence of large-scale datasets such as the 5 million geotagged streetview images provided by OpenStreetView, it becomes possible to explore the relationship between image content and its geographical origin. Leveraging this type of multimodal data allows for the development of machine learning systems that can infer location information directly from visual input.

B. Motivation

The ability to predict the latitude and longitude of untagged images has significant applications in areas such as autonomous navigation, geographic information systems, and digital forensics. However, the problem is inherently difficult due to the variability in visual features across regions and the need for models to generalize beyond memorizing specific locations. By approaching this challenge with large-scale data and deep learning methods, we aim to understand the feasibility and limitations of geolocation prediction from raw images.

C. Proposed Solution

We investigate whether a neural network can be trained to regress geographic coordinates (latitude and longitude) from street-level images. Using a dataset of 5 million geotagged streetview images, we preprocess all inputs by resizing them to 224×224 to reduce computational complexity while maintaining visual fidelity. Alongside visual data, we incorporate timestamp metadata by converting UTC values into local time zones, providing temporal context. The model is trained to output continuous latitude and longitude values, which are directly compared to the ground-truth coordinates of the input image. All of this is done to answer the question, "Can geotagged images be used to train a neural network to predict the latitude and longitude of untagged images?".

D. Contributions

- Designed and implemented a regression-based neural network framework for predicting latitude and longitude from geotagged streetview images.
- Applied big data preprocessing techniques to handle a dataset of 5 million high-definition images with metadata.
- Evaluated multiple neural network architectures to analyze the effectiveness of visual and temporal cues in geolocation prediction.

II. Literature Review

Early efforts such as IM2GPS [7] demonstrated that image content alone could provide cues for geolocation but relied heavily on nearest-neighbor search in a large reference database, limiting generalization. PlaNet [12] improved on this with a CNN trained to classify the globe into geographic cells, achieving significant gains but being constrained by cell resolution and classification boundaries. Later, Revisiting IM2GPS [11] showed that deep features combined with nearest-neighbor retrieval could outperform PlaNet in some cases, emphasizing the role of representation learning. More recently, GeoCLIP [3] introduced contrastive learning between images and geographic coordinates, enabling flexible embedding-based geolocation, while OSV-5M [6] contributed a large-scale benchmark dataset with state-of-the-art baselines, facilitating robust evaluation across methods.

In contrast, our study proposes a regression-based approach rather than classification or retrieval, directly predicting latitude and longitude from visual input. We introduced both baseline fully connected and CNN models, alongside a trigonometric longitude decomposition to better handle cyclicity at the an-

TABLE I
COMPARISON OF STUDIES ON PREDICTING GEOLOCATION FROM IMAGES.

Study	Dataset Size	Model Architecture	Evaluation Metric(s)	Key Findings
Weyand et al. (2016) – PlaNet	126M Flickr + Street View images	Inception CNN + classification over geocells	Median error distance (km)	Achieved median error of 1131 km
Hays & Efros (2008) – IM2GPS	6M Flickr images	Nearest neighbor retrieval (hand-crafted features)	Distance error (km)	Median error $\approx$ 2500 km
Vo et al. (2017) – Local Feature Aggregation	91M Flickr + Google images	CNN features + NetVLAD pooling	Median distance error	Reduced median error to $\approx$ 1,131 km; improved robustness via feature aggregation
Müller-Budack et al. (2018) – Time-Aware Geolocation	14M Flickr images	Multi-modal CNN (visual + temporal features)	Mean/median error distance	Temporal metadata improved accuracy, especially for seasonal/landmark images
Seo et al. (2020) – Cross-View Geo-localization	CVUSA dataset (ground + aerial pairs)	Siamese CNN with shared weights	Recall@K, median error	Leveraged cross-view learning; improved fine-grained localization within cities
Our Study (2025) – OpenStreetView Regression	5M Street View images	MLP, CNN variants, Vision Transformer	MAE (degrees), MSE	Transformer V2 achieved performance with MAE of 19.5° $\approx$ 2500km

timeridian. Moreover, we explored a vision transformer model, showing its potential to capture global spatial relationships in imagery, even under computational constraints. While our best-performing model (transformer v2, MAE = 19.5 degrees) only had a rough distance error of 2500km, performing worse compared to some of the other models, the approach highlights the viability of regression methods for fine-grained geolocation because of our inability to both train on the total dataset and to a sufficient epoch depth due to computational constraints.

III. RESEARCH METHODOLOGY

Our methodology for predicting latitude and longitude from geotagged images consists of five main phases: data preprocessing, feature engineering, model design and selection, model training and refinement, and evaluation. Each phase was designed to ensure that the model could learn meaningful patterns from images and metadata without unintentionally exploiting shortcuts such as time zone leakage or dataset biases.

A. Experimental Results

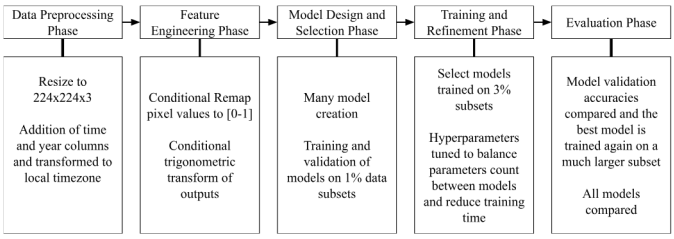

Fig. 1.

B. Data Preprocessing Phase

In the data preprocessing phase, all input images were resized to 224×224×3 to reduce computational complexity while maintaining sufficient detail for feature extraction. This size was chosen to match the input requirements of widely used image models and to make training feasible on available hardware. Images retained their original color channels, as color differences in vegetation, terrain, or sky conditions could provide valuable geographic cues. The raw pixel values, originally ranging from 0 to 255, were normalized to a [0, 1] range using floating-point values to stabilize training and avoid exploding gradients. Metadata processing included converting timestamps from UTC to local time using each image's latitude and longitude with a timezone mapping library, thereby eliminating the possibility of longitude leakage. While certain artifacts such as vehicle dashboards and windshield reflections were present in an estimated 1–5% of the dataset, these were not explicitly removed due to the size of the corpus. We assume these cases to be sufficiently distributed across the dataset so as not to bias the model significantly, though some risk of overfitting to individual vehicles remains.

C. Feature Engineering Phase

The feature engineering phase focused on transforming temporal and geographic metadata into formats suitable for learning. Local time was encoded as a continuous floating point value between 0 and 1, where 0.5 corresponds to noon, ensuring that the cyclical nature of time could be captured while avoiding direct use of date-time objects. Seasonality was represented by scaling the day of the year (excluding the year itself) into the same [0, 1] range, computed as the fraction of seconds elapsed through the year. This encoding enabled the model to learn seasonal cues, such as snow in July indicating a southern hemisphere location, without introducing calendar-related biases. All geographic outputs were left unscaled while other times latitude was scaled to the range [-1, 1], while longitude was expressed as a pair of values, $\cos(\lambda)$ and $\sin(\lambda)$, thereby avoiding discontinuities at ±180° and allowing smoother regression across the globe.

Testing these two scaling methods allowed us to determine if the discontinuity in longitude affected model performance.

D. Model Design and Selection Phase

In the model design and selection phase, we evaluated four different neural network families to balance interpretability, training feasibility, and representational power. The baseline was a simple multilayer perceptron (MLP) with four hidden layers of sizes 256, 128, 64, and 32, using ReLU activation. This model was chosen as a benchmark against which more complex architectures could be compared. Then, convolutional neural networks (CNN) were tested in two configurations. The first, a CNN with constant filter, employed three convolution layers with 64 filters each, using 3×3 kernels followed by batch normalization and 2×2 max pooling. The second, a scaling-filter CNN, used the same kernel sizes but doubled the number of filters in each layer, starting at 32 and ending at 128. The scaling design encouraged earlier layers to capture coarse global patterns and deeper layers to focus on fine-grained details. Both CNN variants were trained using the transformed latitude and longitude outputs. Finally, we implemented a vision transformer (ViT) pretrained on ImageNet-21k, which processed 224×224 inputs into embeddings. These embeddings were passed into a four-layer MLP with fully connected hidden layers of sizes 128, 128, 64, and 64. All models combined image encodings with the engineered temporal and seasonal features immediately after encoding. A final transformer model was also trained with transformer training enabled; this was initially disabled due to time constraints as it massively increased training time.

E. Training and Refinement Phase

The training and refinement phase was conducted on Google Colab TPUs (v2-8) with 300 GB of RAM, which provided a balance between sufficient memory for large-scale data handling and computational efficiency at reasonable cost. Due to resource constraints, preliminary experiments were performed on 1% of the dataset (50,000 images), trained for 10 epochs with a validation split 20%. Training times varied by architecture, ranging from approximately 20 minutes per epoch for the MLP to 40 minutes for the vision transformer. To test scalability, selected models were trained on 3% of the dataset (150,000 images) for five epochs under the same validation protocol. Hyperparameters such as layer sizes and filter counts were tuned iteratively to maintain a balance between training feasibility and model expressiveness, with the final model architectures representing the best compromise observed in preliminary trials.

F. Evaluation Phase

In the evaluation phase, the performance of the model was assessed by comparing the validation errors in different architectures and feature representations. This comparative evaluation was intended not only to identify the best-performing model but also to test whether architectural complexity translated into meaningful performance improvements over the baseline. Special attention was paid to the impact of output transformations, particularly the use of cosine and sine encodings for longitude, to determine whether these modifications improved prediction stability and accuracy.

IV. Experimental Evaluation

A. Experimental Setup

All experiments were conducted using Google Colab's TPU v2-8 configuration with 300 GB of available RAM. This platform offered a cost-effective balance between computational speed and dataset capacity, consuming approximately 1.71 compute units per hour. The dataset consisted of five million geotagged street-view images from OpenStreetView, each accompanied by timestamp metadata. For practical reasons, experiments were restricted to subsets of the data: an initial 1% sample (50,000 images) was used for model comparison over 10 epochs, while extended tests were conducted on a 3% sample (150,000 images) for five epochs to assess scalability. In all cases, 20% of the training subset was reserved as a validation set to monitor generalization performance. Training configurations were kept consistent across models to ensure comparability. Images were input at a resolution of 224×224 with normalized pixel values, and metadata was provided in its engineered form as described in Section 3. Optimization was carried out using a standard setup (batch size of 64, optimizer of Adam, default parameters), which remained constant across experiments to isolate the effect of architecture choice. Evaluation was conducted by measuring validation error in predicted latitude and longitude, either directly for regression outputs or through reconstructed geographic coordinates when using cosine-sine transformations. This setup allowed for controlled comparisons of architecture design, metadata integration, and output representation strategies. The final transformer model (named transformer V2) was trained on 3% samples until that particular sample stopped improving validation accuracy. The model was then trained on the next 3% sample until finally reaching a validation accuracy that did not improve with swapping samples. This was done using transformer V2 because it showed the best potential for improvement. Only this model was trained this extensively due to the computational cost required.

B. Experimental Results

V. Discussion

The results of our experiments highlight several important insights into the feasibility of predicting latitude and longitude from street-view images. Across all models, performance stabilized within a mean absolute error (MAE) of approximately 30–34 degrees, which, while coarse, demonstrates that image content contains sufficient information to estimate location at a continental or regional level. Among the tested models, the Vision Transformer achieved the lowest validation MAE at 30.3 degrees, marginally outperforming both convolutional networks and the fully connected baseline. This suggests that transformer architectures, even when only partially fine-tuned, are particularly well-suited to learning spatial cues and

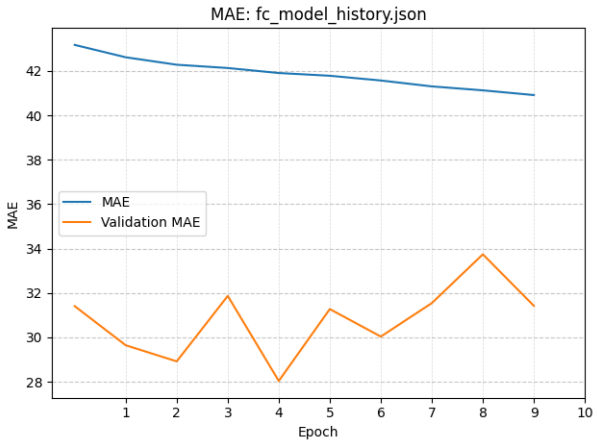

Fig. 2. Feed Forward Network training on 1% dataset for 10 epochs.

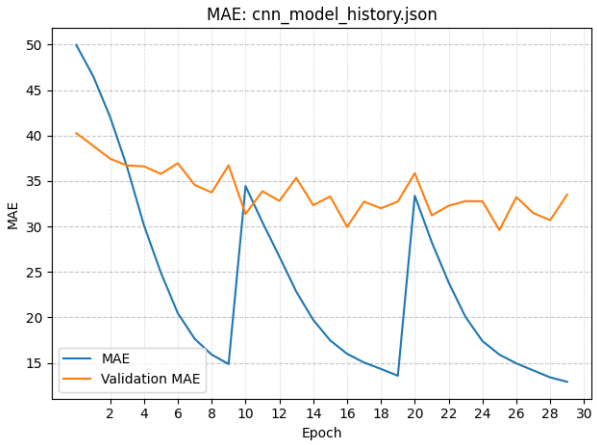

Fig. 3. CNN (constant filter size) training on 3 separate 1% datasets for 10 epochs each.

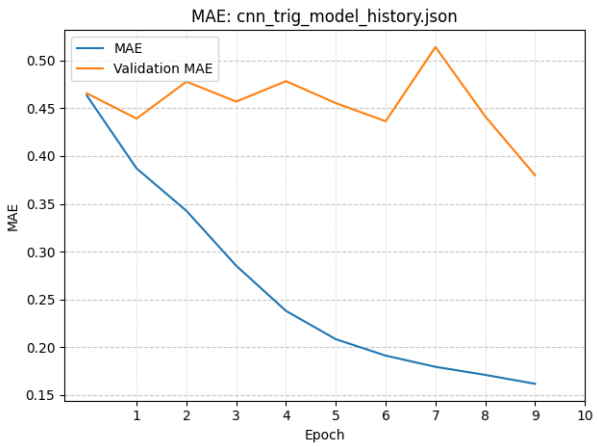

Fig. 4. CNN (constant filter size) training on 1% dataset for 10 epochs using transformed and scaled outputs.

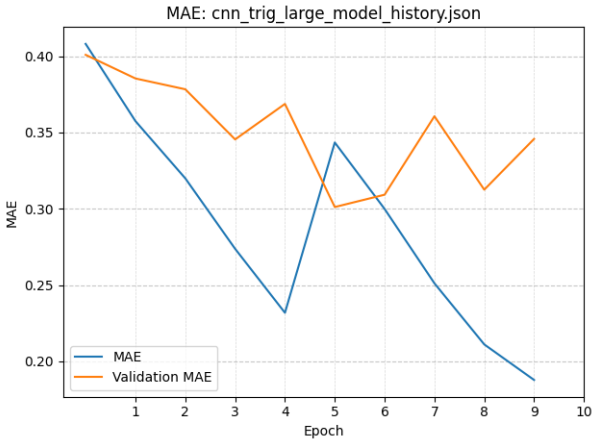

Fig. 5. CNN (increasing number of filters) training on 1% dataset for 10 epochs using transformed and scaled outputs.

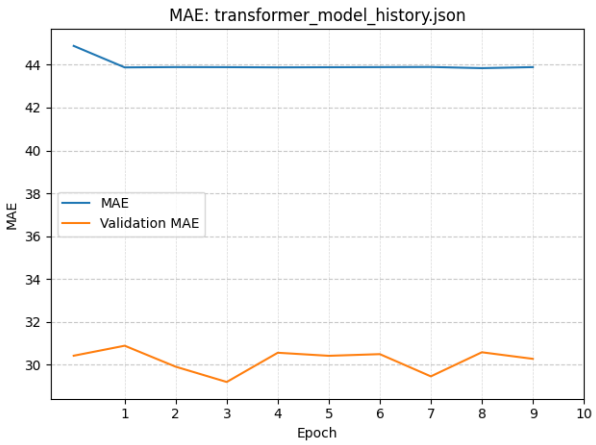

Fig. 6. CV Transformer training on 1% dataset for 10 epochs without encoding fine tuning

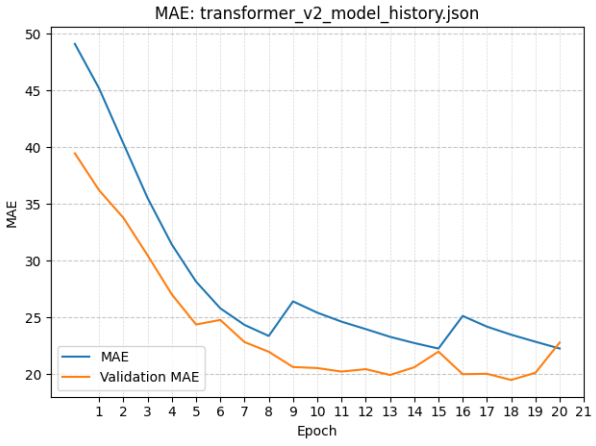

Fig. 7. Transformer V2: The final fine-tuned model trained on 3% datasets for 8, 7, and 5 epochs respectively. Only switching to the next 3% when the validation error increased

contextual relationships within images. Upon further training using the transformer model while allowing the transformer component to train (V2), we reached a minimum validation MAE of 19.5 degrees, that being an average error of around 2500km.

The benefits of architectural choices were not always clear-cut. The fully connected MLP performed comparably to the CNNs, though its instability during training suggests overfitting driven by its very large first layer. The constant-filter CNN and its trigonometric variant demonstrated incremental improvement, but the effect of longitude decomposition into cosine and sine was less pronounced than expected. This may be due to the fact that discontinuity at the antimeridian affects relatively few geographic regions, thus limiting its impact on overall performance. This led us not to use the output decomposition method in our final model. The scaling CNN with increasing filter sizes offered slightly better results than the baseline CNN, consistent with the intuition that deeper layers capture finer details.

The transformer model consistently outperformed all other approaches from the beginning of training, even though only its fully connected layers were trainable during the later testing stage. This indicates that the pretrained ViT embeddings already encode robust geographic cues from natural scenes. Despite the promising results, the first transformer's performance plateaued around the 30-degree MAE threshold, a limit not surpassed by any model tested during the testing section. It is reasonable to expect that fine-tuning the transformer component itself could lead to further improvements, though this comes at significant computational cost, as a single epoch on 3% of the dataset required over three hours of TPU training. For both researchers and practitioners, these findings emphasize the trade-off between accuracy and resource availability: transformer-based approaches appear most effective, but scaling them to global geolocation remains challenging. To test this assumption we trained the final model as described over 9% of the dataset with 20 epochs eventually reaching our best error of 19.5 degrees.

VI. Limitations

There are several key limitations to the current study. First, training was restricted to small subsets of the dataset (1–3%), primarily due to computational resource constraints. This limited the ability of models to generalize across the full geographic and environmental diversity of the dataset. Second, the transformer model was not fully fine-tuned during initial testing due to its high training cost. As a result, its performance represents only a lower bound of its potential which was later explored although not fully as it was only trained on 9% of the dataset and only for 20 epochs. Third, while metadata such as time and season were included, other potentially useful contextual features (e.g., compass orientation, weather conditions) were excluded, which may have constrained the model's ability to disambiguate visually similar locations. Finally, evaluation was performed using global mean absolute error, which, while interpretable, may obscure performance differences at smaller geographic scales, such as city or country-level accuracy.

VII. Conclusion

This study investigated the use of deep learning models to predict latitude and longitude from geotagged street-view images. We implemented and compared a range of architectures, including a feed forward network, two variants of convolutional neural networks, and a vision transformer. Models were trained on subsets of a five-million-image dataset, with engineered temporal and seasonal features incorporated to reduce bias and enhance learning. Results showed that all models converged to a validation MAE in the range of 30–34 degrees, with the Vision Transformer consistently outperforming the others despite being only partially fine-tuned. This transformer model was eventually fine-tuned to a final validation MAE 19.5 degrees.

These findings demonstrate both the potential and the difficulty of image-based geolocation. While achieving continent-level accuracy is feasible, finer-grained localization remains out of reach without additional features, larger-scale training, or more advanced architectures. For future work, we suggest expanding training to larger proportions of the dataset over more epochs, fine-tuning transformer encoders to leverage their representational power, and incorporating additional contextual metadata to improve disambiguation. Since the encoding portion of the transformer was so computationally costly we would also recommend deepening the transformer network to better capitalize on the learning potential of the extracted features rather than passing those features through a simple feed forward network. A more nuanced evaluation framework that considers accuracy at multiple geographic resolutions would also provide clearer insight into model performance. This could potentially allow combining multiple transformer models together, the first to decide the general location and the subsequent models could be trained on specific regions or continents. Taken together, this study provides a foundation for advancing the field of image-based geolocation and highlights the critical importance of balancing accuracy, computational feasibility, and data scale.

References

[1] Hatem Abedi. Large dataset of geotagged images, 2022. Accessed: 2025-06-14.

[2] Guillaume Astruc, Nicolas Dufour, Ioannis Siglidis, Constantin Aronssohn, Nacim Bouia, Stephanie Fu, Romain Loiseau, Van Nguyen Nguyen, Charles Raude, Elliot Vincent, Lintao Xu, Hongyu Zhou, and Loic Landrieu. Openstreetview-5m: The many roads to global visual geolocation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21967–21977, 2024.

[3] Vicente Vivanco Cepeda, Gaurav Kumar Nayak, and Mubarak Shah. GeoCLIP: Clip-Inspired Alignment between Locations and Images for Effective Worldwide Geo-localization. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 8690–8701. Curran Associates, Inc., 2023.

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *International Conference on Learning Representations (ICLR)*, 2021.

[5] Natural Earth. 1:10m cultural vectors – countries, n.d. Accessed: 2025-07-06.

[6] Ioannis Siglidis Constantin Aronssohn Nacim Bouia Stephanie Fu Romain Loiseau Van Nguyen Nguyen Charles Raude Elliot Vincent Lintao XU Hongyu Zhou Loic Landrieu Guillaume Astruc, Nicolas Dufour. OpenStreetView-5M: The Many Roads to Global Visual Geolocation, 2024. CVPR 2024 paper; 5.1M street-view images dataset and benchmarks.

[7] James H. Hays and Alexei A. Efros. IM2GPS: Estimating Geographic Information from a Single Image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2008.

[8] Hatem Mousselly-Sergieh, Daniel Watzinger, Bastian Huber, Mario Döller, Elöd Egyed-Zsigmond, and Harald Kosch. World-wide scale geotagged image dataset for automatic image annotation and reverse geotagging. In *Proceedings of the 5th ACM Multimedia Systems Conference*, pages 47–52, 2014.

[9] Nicholas Scarfe. Big data analysis project. https://github.com/NicholasScarfe/Big_Data_Analysis_Project_Nscarfe.git, 2025. Accessed: 2025-08-17.

[10] Michael Schmitt, Lloyd Haydn Hughes, Chunping Qiu, and Xiao Xiang Zhu. Sen12ms – a curated dataset of georeferenced multi-spectral sentinel-1/2 imagery for deep learning and data fusion, 2019. Accessed: 2025-06-14.

[11] Nam Vo, Nathan Jacobs, and James Hays. Revisiting IM2GPS in the Deep Learning Era. *arXiv preprint arXiv:1705.04838*, 2017.

[12] Tobias Weyand, Ilya Kostrikov, and James Philbin. PlaNet - Photo Geolocation with Convolutional Neural Networks. In *European Conference on Computer Vision (ECCV)*, volume 9912 of *Lecture Notes in Computer Science*, pages 37–55. Springer International Publishing, 2016.