Presence-Only Geographical Priors for Fine-Grained Image Classification

Oisin Mac Aodha Elijah Cole Pietro Perona
Caltech
www.vision.caltech.edu/~macaodha/projects/geopriors

Abstract

Appearance information alone is often not sufficient to accurately differentiate between fine-grained visual categories. Human experts make use of additional cues such as where, and when, a given image was taken in order to inform their final decision. This contextual information is readily available in many online image collections but has been underutilized by existing image classifiers that focus solely on making predictions based on the image contents.

We propose an efficient spatio-temporal prior, that when conditioned on a geographical location and time, estimates the probability that a given object category occurs at that location. Our prior is trained from presence-only observation data and jointly models object categories, their spatio-temporal distributions, and photographer biases. Experiments performed on multiple challenging image classification datasets show that combining our prior with the predictions from image classifiers results in a large improvement in final classification performance.

1. Introduction

Correctly classifying objects into different fine-grained visual categories is a challenging problem. In contrast to generic object recognition, it can require knowledge of subtle features that are essential for differentiating between visually similar categories. However, without having access to additional information that may not be present in an image, many categories can be visually indistinguishable. For example, the two toad species in Fig. 1 are similar in appearance but tend to be found in very different locations in Europe. Knowing *where* a given image was taken can provide a strong prior for *what* objects it may contain.

Most images that are captured and shared online today also come with additional metadata in the form of: *where* they were taken, *when* they were taken, and *who* captured them. This information not only offers the possibility of helping to resolve ambiguous cases for image classification, but can also enable us to generate predictions of where, and when, different objects are likely to be observed.

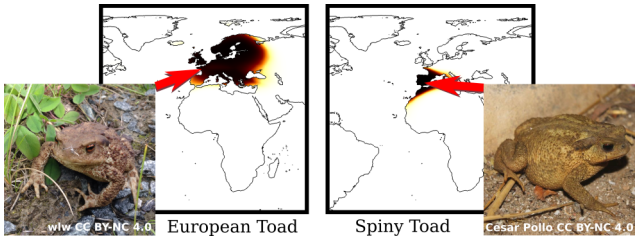

Figure 1: Differentiating between two visually similar categories such as the European (left) and Spiny (right) Toads can be challenging without additional context. To address this problem, we propose a spatio-temporal prior that encodes where, and when, a given category is likely to occur. For a known test location our prior predicts how likely it is for each category to be present. Darker colors indicate locations that are more likely to contain the object of interest.

Existing work that uses location information to improve classification performance either discretizes the input data into spatio-temporal volumes [5], store the entire training set in memory at inference time [59], or jointly train deep images classifiers while along with corresponding location information [50]. Methods that discretize or store the raw training data do not scale well in terms of memory use, and jointly training image classifiers with location information necessitates that location information is present at test time - which may not always be the case. We instead take inspiration from species distribution modeling (SDM) [18], and model a separate geographical prior that can be combined with the predictions of *any* image classifier. However, unlike many approaches to SDM that assume they have access to presence and absence information at training time (*e.g.* [51]), we make a more general assumption that only presence information is available *i.e.* we know where the categories have been observed, but have *no* data regarding where they are not found.

In this work we make the following contributions: (i) An efficient spatio-temporal prior that jointly models the relationship between location, time of year, photographer, and the presence of multiple different object categories. (ii) A novel presence-only training loss to capture these rela-

tionships. (iii) Experiments that show that combining the probabilistic predictions of image classifiers with our prior significantly improves the test time performance on challenging fine-grained image datasets.

2. Related Work

Here we discuss work related to spatio-temporal models that encode the location of a set of discrete object categories. We do not address methods that have explored other uses of location information such as inferring where an image was taken given only the raw pixels [26, 55], or methods that use location to disambiguate visually similar places for image localization [54].

Fine-Grained Image Classification

Correctly determining which one of multiple possible fine-grained categories is present in an image requires understanding the relationship between subtle visual features and the corresponding image-level category label *e.g*. [56, 31, 61, 53]. Existing approaches have investigated the modeling of parts [37, 65, 8, 64, 29], higher order feature interactions [36, 21], attention mechanisms [60, 66, 57], noisy web data [33], novel training losses [12], and pairwise category information [15]. Orthogonal to those works, we propose a spatio-temporal prior that can be combined with the probabilistic predictions of any image classifier to improve the final classification performance.

Location and Classification

A small number of approaches have explored the use of location information to improve image classification at test time. Berg *et al*. [5] proposed a spatio-temporal prior that when combined with the output of an image classifier increased the accuracy of bird species classification. Their approach discretized location and time into spatio-temporal cubes and used an adaptive kernel density estimator to represent the distribution of each species independently. Also in the context of predicting the presence of different biological species, Wittich *et al*. [59] evaluated different nearest neighbor based lookup strategies for retrieving the most relevant instances from a training set of geo-tagged observations. These approaches are inefficient in terms their memory requirements as they necessitate storing either the entire training set or a discretized version of it. Existing repositories of citizen science data (*e.g*. [48, 2, 1]) can contain on the order of tens of millions of observations making them prohibitively large to store and retrieve on mobile devices. Choosing the correct discretization is challenging [43], and incorrect choices can significantly affect the final performance [34, 42]. A key benefit of our approach is that discretization is not required.

Tang *et al*. [50] explored different feature encodings for incorporating location information directly into deep neural networks at training time. This included raw location features (*i.e*. longitude and latitude), demographic information collected via a census, user provided hash-tags, and geographical map features (*e.g*. land use estimates). The disadvantage of their method is that it assumes that location information is present at test time and that all the required features can be computed for a given test location. Furthermore, they cannot use location information that does not have an associated image. They also need to retrain their entire model if new location data is collected. We instead propose an efficient spatio-temporal prior that jointly models the spatial distribution of multiple object categories that can be trained independently of the image classifier.

Spatio-Temporal Distribution Modelling

Our goal is to estimate the spatio-temporal distribution of a set of object categories. Related to this, there is a rich literature exploring models for estimating the distribution of biological specimens across geographic space and time [28]. This is referred to as species distribution modelling or environmental niche modelling. Broadly, these methods can be divided into two groups, those that use *presence-absence* and those that use *presence-only* information [25].

Making a presence-absence observation at a given location requires that every species from a predefined set of interest be confirmed as either being present or absent for that sampling event. In practice, this kind of data is onerous to collect because it requires intense survey effort to confirm that a species is truly absent with a high degree of certainty [39]. However, once this data is collected it can be combined with standard supervised classification approaches such as logistic regression [25], probit regression [47], Gaussian processes [23], decision trees [18], and neural networks [62, 44, 41], among others [16]. Presence-absence data is also compatible with traditional multi-label learning [7, 63, 11, 58]. Recently deep models have been applied to this problem in order to jointly model the location preferences of different species [24, 10, 51, 6] and human sampling biases [9].

In contrast, a presence-only (*i.e*. incidental) observation may be recorded wherever an object of interest is encountered - *without* requiring any absences to be verified. While presence-only data can be much easier to collect, the lack of absence information makes it more difficult to model. This limitation is typically dealt with in one of three different ways. The first approach is to generate 'pseudo-negatives' and then apply one of the presence-absence approaches from above. As no true negative information is available, these approaches randomly sample a set of locations and make the assumption that these locations are absences *e.g*. [17, 45, 3]. The second commonly used approach is to train a highly regularized model directly on the presence-only data *e.g*. by fitting a maximum entropy distribution [46] or a

low-rank model [19]. These methods make the assumption that the model should be able to explain data where it has been observed and should be uncertain elsewhere. Finally, and most related to our work, are the approaches that use additional information such as the detectability of a given species and a photographer's propensity to image them *e.g.* [40, 22].

Unlike many of the classic approaches for spatio-temporal distribution modelling, in this work we jointly learn a continuous spatio-temporal prior for each category of interest using a neural network to amortize the computation. In contrast to previous deep distribution models *e.g.* [24, 10, 51], we do not require presence-absence data or additional environmental features as input. We instead exploit the structure that exists in online image repositories such as those collected by citizen scientists to jointly model objects [24, 20], their locations, and photographer biases.

3. Methods

Here we outline our spatio-temporal prior, which models the geographical and temporal distribution of a set of object categories and photographers. During training we assume that we have access to a set of tuples $\mathcal{D} = \{(I_i, \mathbf{x}_i, y_i, p_i) | i = 1, ..., N\}$, where I_i is an image, $y_i \in \{1, ..., C\}$ is the corresponding class label, $\mathbf{x}_i = [\text{lon}_i, \text{lat}_i, \text{time}_i]$ represents the location (longitude and latitude) and time the image was taken, and p_i is the individual, *i.e.* photographer, who captured the image. Note that the location does not need to be captured alongside the image. $\mathcal{D}$ can be assembled from unrelated image and location datasets as long as both contain the same categories.

At test time, given an image and where and when it was taken we aim to estimate which category it contains *i.e.* $P(y|I, \mathbf{x})$. One approach is to model the joint distribution $P(I, \mathbf{x})$ as in [50], but this necessitates that the location information is *always* available at test time. Instead, inspired by [5], we can incorporate location information as a Bayesian spatio-temporal prior. If we assume that I and $\mathbf{x}$ are conditionally independent given y, then

$$P(y|I, \mathbf{x}) = \frac{P(I, \mathbf{x}|y)P(y)}{P(I, \mathbf{x})} \quad (1)$$

$$= \frac{P(I)P(\mathbf{x})}{P(I, \mathbf{x})} \frac{P(y|I)P(y|\mathbf{x})}{P(y)} \quad (2)$$

$$\propto P(y|I)P(y|\mathbf{x}), \quad (3)$$

where we assume a uniform prior $P(y) = 1/C$ for $y \in \{1 \ldots, C\}$. In reality an image may contain location information unrelated to the class label (*e.g.* the background), but we assume this factorization is valid. By factoring the distribution in this way we can represent the image classifier, $P(y|I)$, and spatio-temporal prior, $P(y|\mathbf{x})$, separately. Note that at test time we do not assume that we have any knowledge of the individual p who captured the image. For $P(y|I)$ we can use any discriminative model that produces a probabilistic output *e.g.* a convolutional neural network. In this work we focus our attention on representing $P(y|\mathbf{x})$.

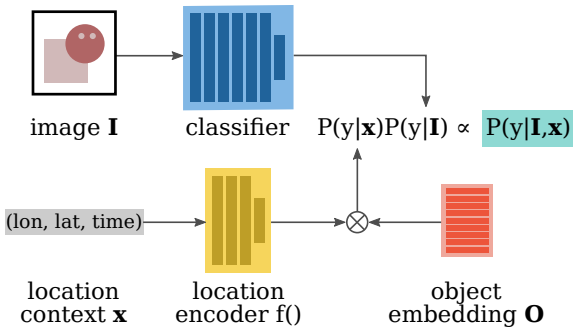

Figure 2: **Inference time.** Our goal is to estimate if an object category y is present in an input image $\mathbf{I}$. At test time we make use of additional spatio-temporal information $\mathbf{x}$ in the form of where and when the image was taken.

Presence-Absence Loss

As we are modeling the spatio-temporal prior independently from the image classifier our training data is now of the form $\mathcal{D} = \{(\mathbf{x}_i, y_i, p_i) | i = 1, ..., N\}$. In the ideal case we would have complete information consisting of where and when a given category has both been observed to be present and been observed to be *not* present *e.g.* as in [10, 51]. Then instead of $y_i \in 1, ..., C$, each spatio-temporal location $\mathbf{x}_i$ would be associated with a binary multi-label vector $\mathbf{y}_i = [y_i^1, ..., y_i^C]$ where each entry $y_i^c \in \{0, 1\}$ indicates whether or not category c is present at $\mathbf{x}_i$. This formulation results in a standard multi-label learning problem, so we can estimate the parameters of the spatio-temporal model by solving

$$\max_\theta \sum_{i=1}^{N} \sum_{c=1}^{C} y_i^c \log(\hat{y}_i^c) + (1 - y_i^c) \log(1 - \hat{y}_i^c), \quad (4)$$

where we define $\hat{y}_i^c = P(y_i^c | \mathbf{x}_i)$ and P is parameterized by θ. However, as discussed previously, presence-absence information is both difficult and time consuming to acquire in real world settings.

Presence-Only Loss

In this work we explore the more challenging presence-only setting, where each spatio-temporal location $\mathbf{x}_i$ is associated with a binary label $y_i \in \{1, ..., C\}$ indicating which single category was observed. In essence, we have a label vector $\mathbf{y}_i$ where there is only one affirmative entry, *i.e.*

$y_i^c = 1$ for some c, and the remaining entries are unknown. In this setting, Eqn. 4 can be written as

$$\max_{\theta} \sum_{i=1}^{N} \log(\hat{y}_i^{c_i}) + A_i, \tag{5}$$

where A_i represents a proxy absence term for the i^{th} training example and c_i is the corresponding observed category. Now the question becomes how to choose A_i.

One common approach for representing A_i is to generate 'pseudo-negatives' [3] by randomly sampling absence data from some parametric distribution. For instance, one might set

$$A_i = \log(1 - P(y_i|\mathbf{r}_i)). \tag{6}$$

where $\mathbf{r}_i$ is a randomly selected spatio-temporal location with $[\text{lon}(\mathbf{r}_i), \text{lat}(\mathbf{r}_i)] \sim \text{Unif}(\mathbb{S}^2)$ and $\text{time}(\mathbf{r}_i) \sim \text{Unif}([0, 1])$. The implicit assumption is that each category (whether man-made or naturally occurring) occurs in a relatively small subset of $\mathbb{S}^2 \times [0, 1]$, so the probability of a category occurring at a randomly chosen location $\mathbf{r} \in \mathbb{S}^2 \times [0, 1]$ is small as well. To the extent that this assumption holds, these pseudo-negatives are likely to be valid.

An alternative approach is to instead sample absences over locations and times where the presence data for other categories occurs. In this case we would set A_i according to Eqn. 6 but sample negative locations from the positive occurrence locations *i.e.* $\mathbf{r}_i \sim \text{Unif}(\{\mathbf{x}_1, \ldots, \mathbf{x}_N\})$. This biases the training towards regions that contain valid data.

3.1. Our Approach

In this section we outline how we model and train our spatio-temporal prior $P(y|\mathbf{x})$.

Location and Object Embedding

In many contexts, different objects do not occur independently at a given spatio-temporal location. Knowing that object A is present may provide information regarding the presence or absence of object B at the same place and time. Similarly, different spatio-temporal locations are not independent, and may share commonalities. We exploit this structure to encode low dimensional embeddings of objects and spatio-temporal locations.

Taking inspiration from [10], we model our spatio-temporal prior as $P(y|\mathbf{x}) \propto s(f(\mathbf{x})\mathbf{O})$. Here, $f : \mathcal{R}^3 \rightarrow \mathbb{R}^D$ is a multi-layered fully-connected neural network that maps a spatio-temporal location $\mathbf{x}$ to a D dimensional embedding vector. $\mathbf{O} \in \mathcal{R}^{D \times C}$ represents an object embedding matrix, where each column is a different category. The product $f(\mathbf{x})\mathbf{O}$ results in a C dimensional vector, where each element represents the affinity that a spatio-temporal location $\mathbf{x}$ has for category y. The intuition is that we are representing spatio-temporal locations and object categories in a shared embedding space where the inner product between the embedding of a location $\mathbf{x}$ and an object y is large if y is likely to occur at location $\mathbf{x}$. Finally, $s()$ is an entry-wise sigmoid operation to ensure that the resulting prediction are in the range $[0, 1]$.

Photographer Embedding

In online image collections we often have access to additional information at training time in the form of the photographer $p \in \mathcal{P}$ who captured the image. To see why this information is valuable, consider the following example. Suppose a photographer p visits location $\mathbf{x}$ and does *not* report object y. If p has never taken an image of an object like y, then this non-report gives us little information. However, if p has a history of reporting categories similar to object y, then this constitutes weak evidence that y might actually be absent at that location. Thus, we can interpret the same presence-only information in different ways depending on the individual who provides it.

To capture photographer biases, we embed photographers into the same shared embedding space as the objects and locations. This is achieved by learning a photographer embedding matrix $\mathbf{P} \in \mathcal{R}^{D \times |\mathcal{P}|}$ at training time. The intuition is that, like the objects, photographers have affinities for particular locations and times, and share similarities in their spatio-temporal patterns with other photographers. This enables us to represent both a photographer's preference for a given location $P(p|\mathbf{x}) \propto s(f(\mathbf{x})\mathbf{P}))$, and a photographer's affinity for a given object category $P(y|p) \propto s(\mathbf{O}^T\mathbf{P})$. Once trained, the photographer embeddings $\mathbf{P}$ are not required at test time, see Fig. 2.

Joint Embedding Loss

Our goal at training time is to estimate the set of parameters $\theta = [\theta_f, \mathbf{O}, \mathbf{P}]$, where θ_f denotes the weights of the location embedding network $f()$, $\mathbf{O}$ is the category embedding matrix, and $\mathbf{P}$ is the photographer embedding matrix.

We start with the intuition that our model should be conservative: if a category y has been observed at the spatio-temporal location $\mathbf{x}$ in the training set, then $s(f(\mathbf{x})\mathbf{O}_{:,y})$ should be close to 1, otherwise it should be close to 0. Here, $\mathbf{O}_{:,y}$ indicates the y^{th} column of $\mathbf{O}$. We rely on the location embedding function $f()$ to interpolate between presence locations. This is conservative in the sense that it assumes that an object is absent if it has not been observed. This is a very strong assumption, but it enables the spatio-temporal prior to be aggressive in down-weighting incorrect predictions from the image classifier.

Our first loss encourages the model to predict the presence of objects where they have been observed in the training set and down weight their likelihood where they have

not been observed:

$$\begin{aligned}\mathcal{L}_{o\_loc}(\mathbf{x}, \mathbf{r}, \mathbf{O}, y) =& \lambda \log(s(f(\mathbf{x})\mathbf{O}_{:,y})) + \\ & \sum_{\substack{i=1 \\ i \neq y}}^{C} \log(1 - s(f(\mathbf{x})\mathbf{O}_{:,i})) + \\ & \sum_{i=1}^{C} \log(1 - s(f(\mathbf{r})\mathbf{O}_{:,i})).\end{aligned} \tag{7}$$

λ is a hyperparameter used to weight the positive observations and $\mathbf{r}$ is a uniformly random spatio-temporal datapoint. Next, we want the affinity between a photographer p and a location $\mathbf{x}$ be high if p was present at $\mathbf{x}$, and low otherwise:

$$\begin{aligned}\mathcal{L}_{p\_loc}(\mathbf{x}, \mathbf{r}, \mathbf{P}, p) =& \log(s(f(\mathbf{x})\mathbf{P}_{:,p})) + \\ & \log(1 - s(f(\mathbf{r})\mathbf{P}_{:,p})).\end{aligned} \tag{8}$$

We assume that a photographer has a low affinity for a category unless they have ever observed it:

$$\begin{aligned}\mathcal{L}_{p\_o}(\mathbf{O}, \mathbf{P}, y, p) =& \lambda \log(s(\mathbf{O}_{:,y}^T \mathbf{P}_{:,p})) + \\ & \sum_{\substack{i=1 \\ i \neq y}}^{C} \log(1 - s(f(\mathbf{O}_{:,i}^T \mathbf{P}_{:,p})).\end{aligned} \tag{9}$$

Finally, to estimate the parameters of our prior we simply maximize

$$\mathcal{L} = \mathcal{L}_{o\_loc} + \mathcal{L}_{p\_loc} + \mathcal{L}_{p\_o}, \tag{10}$$

by iterating over each of the datapoints in the training set.

4. Experiments

We evaluate the effectiveness of our spatio-temporal prior by performing experiments on several image classification datasets that have location and time information. We choose image classification because for other domains (*e.g.* species distribution modeling) it is not possible to obtain accurate ground truth information regarding the true spatio-temporal distributions of the categories of interest.

4.1. Datasets

While location metadata is readily available for online image collections, many popular image classification datasets do not contain this information *e.g.* [56, 52, 14, 35]. However, datasets containing images of different species of plants and animals are readily available with location, time, and photographer information. To this end, we perform experiments on the iNaturalist 2017 and 2018 (iNat2017 and iNat2018) species classification datasets which contain images collected and annotated by citizen scientists [53]. They have 5,089 and 8,142 categories respectively. While [5] evaluated their location prior on the BirdSnap dataset, the images and location metadata used are not provided by the authors. We recollect the images and location data from the web using the original image URLs. Despite the dataset consisting of images of species commonly found in North America, when we recollected the images and locations we found that the original images are from all over the world and 40% were missing location. Like [5], we also simulate location metadata for BirdSnap [5] and another fine-grained dataset of birds, NABirds [52], by associating each image with a species observation from eBird [48]. Our train locations and photographers are sampled from eBird 2015, and the test set is from 2016. BirdSnap and NABirds contain images from 500 and 555 different species of North America birds. Finally, we also perform experiments on YFCC100M-GEO100 [50] (YFCC). YFCC contains 100 everyday object categories with associated locations, but no date or photographer information is provided. Unlike the other datasets, some of the object categories in YFCC are not geographically distinct *e.g.* 'band', 'ford', or 'ipod'.

4.2. Implementation Details

Our location encoder $f()$ is a fully-connected neural network consisting of an input layer, followed by multiple residual layers [27], and a final output embedding layer. We jointly train the location encoder, along with the photographer and object embeddings using Adam [32] for 30 epochs with a batch size of 1024, using dropout to prevent overfitting. The dimensionality of the shared embedding space is set to $D = 256$. When weighting the positive instances during training we set λ to the number of categories. To counteract the heavily imbalanced nature of many of the datasets, we limit the maximum number of datapoints for each category per epoch. We set the maximum number of datapoints to 100, and for each epoch we randomly select a different subset for each category. The only exception is for YFCC, where capping the data hurt performance. Full details of our network architecture and training procedure are available in the supplementary material.

Unless otherwise specified, at test time, our model takes three inputs – longitude, latitude, and day of the year, specifying where and when the image of interest was captured. For these three input features $\mathbf{x}$ we explored different methods for 'wrapping' the coordinates *i.e.* an observation taken on December 31^{st} should result in a similar embedding to one captured on January 1^{st}. Similarly, we want geographical coordinates to wrap around the earth. To achieve this, for each input dimension l of $\mathbf{x}$ we perform the mapping $[\sin(\pi x^l), \cos(\pi x^l)]$, resulting in two numbers for each dimension. Here, we assume that each dimension of the input has been normalized to the range $x^l \in [-1, 1]$.

For the image classifiers $P(y|I)$ we fine-tune a separate InceptionV3 [49] network for each of the datasets beginning with ImageNet initialized weights [14]. Unless otherwise specified, we use an input image resolution of 299×299.

	YFCC	BirdSnap	BirdSnap†	NABirds†	iNat2017			iNat2018		
$P(y\|\mathbf{x})$ - Prior Type	Test	Test	Test	Test	Val	Test Pu	Test Pr	Val	Test Pu	Test Pr
No Prior (*i.e.* uniform)	50.15	70.07	70.07	76.08	63.27	64.16	63.63	60.20	50.17	50.33
Nearest Neighbor (num)	**51.78**	70.82	77.76	79.99	65.35	66.07	65.61	68.72	54.50	54.53
Nearest Neighbor (spatial)	51.21	71.57	77.98	80.79	65.55	66.72	66.14	67.51	53.70	53.83
Discretized Grid	51.19	71.13	77.19	79.58	65.49	66.62	66.07	67.02	53.28	53.49
Adaptive Kernel [5]	51.47	71.57	78.65	81.11	64.86	65.83	65.59	65.23	53.17	53.21
Ours no date	50.70	71.66	78.65	81.15	69.34	70.62	70.18	72.41	57.68	**57.84**
Ours full	-	**71.84**	**79.58**	**81.50**	**69.60**	**70.83**	**70.51**	**72.68**	**58.44**	**57.84**

Table 1: **Classification accuracy.** Results after combining image classification predictions $P(y|I)$ with different spatio-temporal priors $P(y|\mathbf{x})$. All results are top 1 accuracy with classifier predictions extracted from an InceptionV3 [49] network fine-tuned on each of the respective datasets. † indicates that simulated locations, dates, and photographers from the eBird dataset [48] are used. The baseline algorithms do not use date information.

	Top1	Top3	Top5
iNat2017 - InceptionV3 299×299			
No Prior (*i.e.* uniform)	63.27	79.82	84.51
Ours no wrap encode	69.48	84.43	88.15
Ours no photographer	69.39	83.97	87.71
Ours no date	69.34	84.16	87.89
Ours full	69.60	84.41	88.07
iNat2018 - InceptionV3 299×299			
No Prior (*i.e.* uniform)	60.20	77.90	83.29
Ours no wrap encode	72.12	87.00	90.52
Ours no photographer	72.84	87.30	90.75
Ours no date	72.41	87.19	90.60
Ours full	72.68	87.26	90.79
iNat2018 - InceptionV3 520×520			
No Prior (*i.e.* uniform)	66.18	83.32	88.04
Ours no wrap encode	77.09	90.68	93.54
Ours no photographer	77.64	90.82	93.52
Ours no date	77.41	90.80	93.58
Ours full	77.49	90.85	93.57

Table 2: **Ablation.** Classification accuracy for different variants of our prior on the iNat2017 and iNat2018 [53] validation sets. In the case of iNat2018, we still observe improvements when combining our prior with a more powerful image classifier - see rows 'InceptionV3 520×520'.

4.3. Quantitative Evaluation

In this section we evaluate how much our spatio-temporal prior improves image classification performance. We focus on comparing to methods that can avail of presence-only data and those that use a dedicated spatio-temporal prior as opposed to approaches that jointly train the image classifier with location information *e.g.* [50]. This is because for many datasets, location and time information is not always available at test time for all images.

Table 1 contains a comparison of our approach to alternative methods that utilize location information. For the baseline methods we normalize their predictions so that they sum to one. Additionally, we found that adding a weak uniform prior to their outputs significantly improves their performance. This adds robustness in cases where there are no objects from the training set present near the test locations. The lack of a uniform prior explains the poor results for nearest neighbor based approaches in [50]. For each of the baseline algorithms we select their hyperparameters (*e.g.* the number of neighbors) on a held out validation set for each dataset. When location information is not available at test time, we assume a uniform prior over the categories.

Our model performs on par, or better, than the baselines across all datasets. The advantage of our approach is that it is computationally efficient at test time. Compared to lookup based methods, it only requires a forward pass through a compact fully-connected neural network. In addition, it also captures structural information such as object and photographer biases. One failure case that is worth noting are the results on YFCC [50]. Here, we observe that all methods perform similar to the uniform prior *i.e.* just using the output of the image classifier. This can be explained by the relative lack of spatio-temporal structure in the object categories present in the dataset. Again, this is consistent with the findings in [50], where the authors had to use additional features to produce an increase in performance.

4.3.1 Ablation Study

In Table 2 we compare the performance of different variants of our model on iNat2017 and iNat2018 [53]. Again, across all metrics there is a large increase in performance compared to the baseline uniform prior. In some cases, we even observe that there is an additional boost in performance when we explicitly model photographers.

Training fine-grained image classifiers with larger input images can significantly increase classification performance [13]. We observe that the benefit of our spatio-temporal prior is still apparent even when we use a more powerful classifier that has been training for longer with larger images. This increase in accuracy is also present when we evaluate performance using more lenient evaluation metrics *i.e.* Top5 vs. Top1 accuracy. This is significant because it highlights that for some datasets the performance boost provided by the spatio-temporal prior is orthogonal to improvements in the underlying image classifier.

(a) Location embedding (b) Photographer location affinity

Figure 3: **Spatial predictions.** (a) Embeddings for each location on the earth for a model trained on iNat2018 [53]. We observe that the embeddings appears to capture information related to climate zones, despite not being trained on any climate data. (b) Log plot of estimated photographer location preferences. Brighter colors indicate that more photographers have captured images in that location. We can see that there is a large bias towards North America and Western Europe.

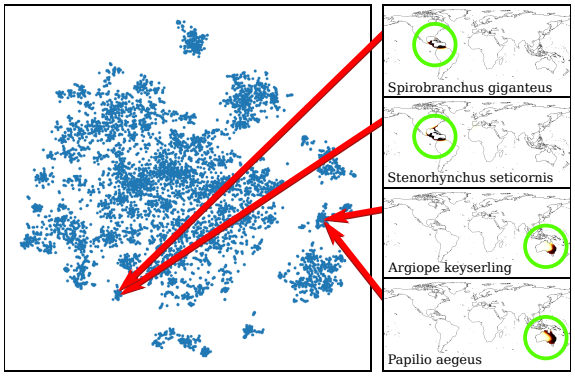

Figure 4: **Object embedding.** t-SNE [38] plot of the learned embedding $\mathbf{O}$ for all 8,142 categories from iNat2018 [53]. The location in the object embedding space encodes a category's preferences for a particular geographical region. We observe that categories that have similar spatio-temporal distributions tend to be nearby.

4.4. Qualitative Evaluation

Our model captures the relationship between objects, locations, and photographers. In Fig. 3 (a) we can see the resulting embedding for each input location from our prior trained on iNat2018 [53]. By applying the embedding function $f()$ to each location we can generate its D dimensional embedding vector. We then use ICA [30] to project the embedded features to a three dimensional space and mask out the ocean for visualization. Perhaps as expected, there is low frequency structure in the resulting image *i.e.* nearby locations tend to support similar objects. One advantage of our approach is that we are not restricted to a fixed discretization. As a result we can generate embeddings for any location and time. In Fig. 4 we visualize our learned object embedding $\mathbf{O}$. Objects that have similar spatio-temporal distributions tend to result in similar embedding vectors.

Distinct from other work, our prior also models the relationship between photographers and locations, and photographers and object categories. In Fig. 3 (b) we plot the estimated affinity for each input location across all photographers *i.e.* $\sum_p s(f(\mathbf{x})\mathbf{P}_{:,p})$. We only show results for photographers who provided at least 100 observations in the iNat2018 [53] training set, resulting in 634 individuals. In Fig. 5 we display the estimated affinity for each object category for a set of photographers *i.e.* $P(y|p) \propto s(\mathbf{O}^T\mathbf{P})$. We observe that the embedding captures the similarity in object affinity held by different photographers.

Finally, in Fig. 6 we use our prior to generate spatio-temporal predictions for several different species from iNat2018 [53]. Each image is generated by querying every location on the surface of the earth, on a specified day of the year, to generate $P(y = y^*|\mathbf{x})$ for the category of interest y^*. In practice, we evaluate 1000×2000 spatial locations for each time point (*e.g.* first day of the month). This step is very efficient as we can pre-compute $f(\mathbf{x})$ for every location, independent of the category of interest. Again, for visualization we mask out the predictions over the ocean.

4.5. Limitations

We are limited by the quality of the provided location data *e.g.* it can be inaccurate or intentionally obfuscated. We also make strong assumptions about a photographer's affinity for an individual object category. In practice, these interactions may be complex *i.e.* once a photographer captures an image of a particular category they may be less likely to take an image of the same object in the near future. There are also known spatial biases in the types of citizen science data we use [4, 9]. However, this may not be a major issue as we can assume that the distribution of test locations and dates is similar those observed during training.

5. Conclusion

We introduce a spatio-temporal prior to help disambiguate fine-grained categories resulting in improved test time image classification performance. In addition to helping image classification, our model also naturally captures the relationships between locations and objects, objects and objects, photographers and objects, and photographers and

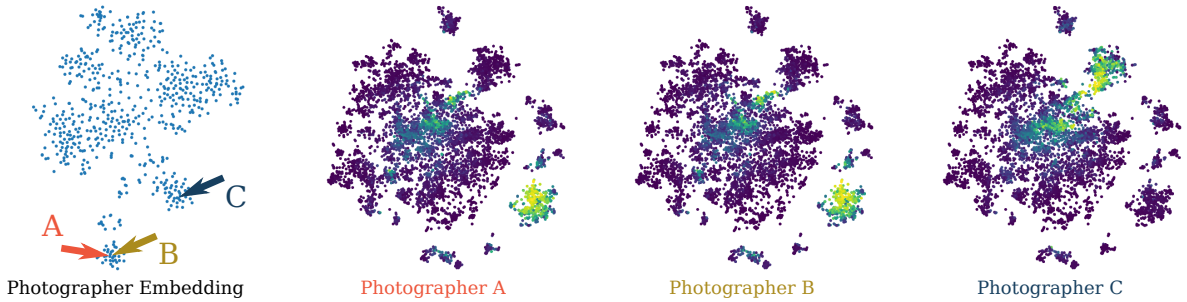

Figure 5: **Photographer object affinity**. On the left we see a t-SNE [38] plot of the photographer embeddings $\mathbf{P}$ for iNat2018 [53]. The three plots on the right depict the predicted affinities for three different photographers (A,B, and C) visualized on the category embedding from Fig. 4. Brighter colors indicate a higher affinity for a given category. We observe that individuals that are close in the photographer embedding space $\mathbf{P}$ (*e.g.* A and B) have similar category affinities, compared to those that are far away (*e.g.* C).

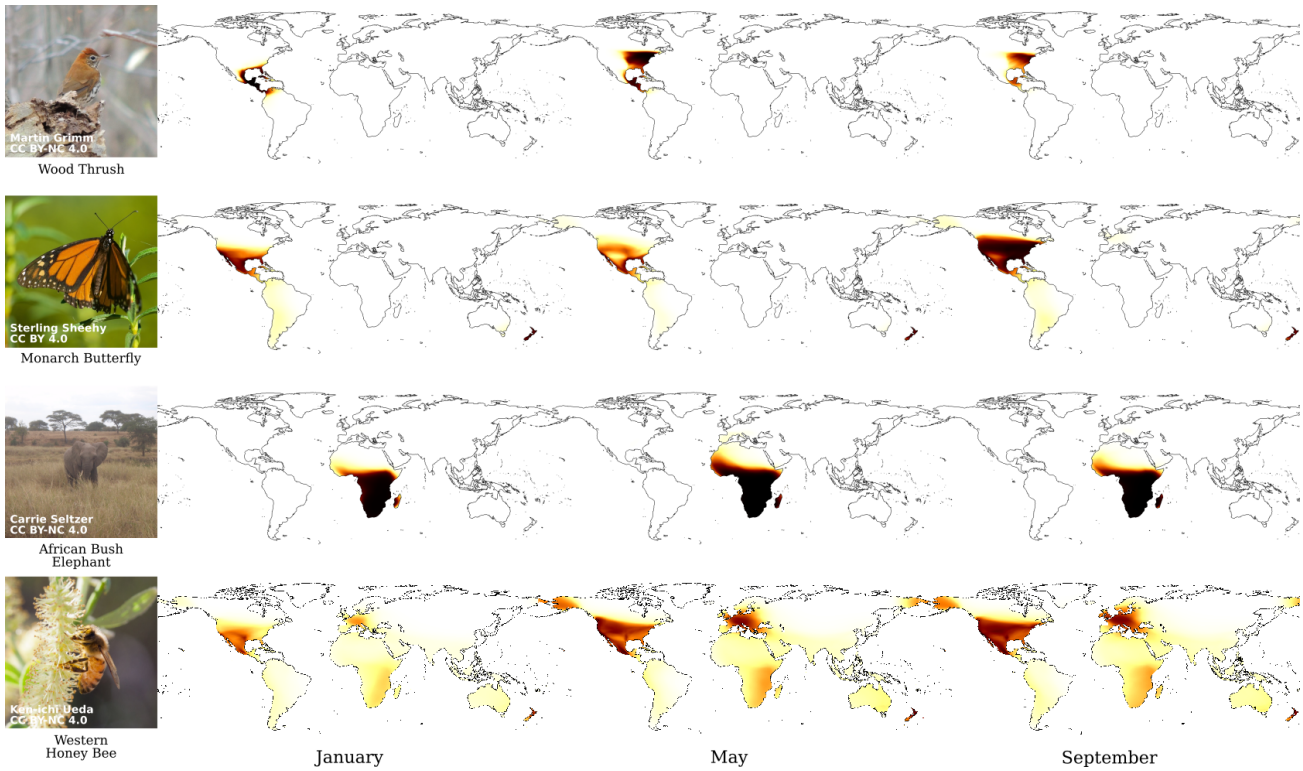

Figure 6: **Spatio-temporal predictions**. Predicted distributions for several object categories for three different time points using our full model trained on iNat2018 [53]. In the first two rows, we see that our model captures seasonal migratory behaviors. Darker colors indicate locations where the categories are predicted to be found. On the bottom row, our model correctly predicts that the Western Honey Bee can be found on several different continents. It is worth noting that the results are affected by geographical sampling biases in the iNat2018 dataset.

locations in an interpretable manner. Importantly, our prior is efficient at test time, both in terms of model size and inference speed, and scales to large numbers of categories.

Acknowledgements This work was supported by a Google Focused Research Award and an NSF Graduate Research Fellowship (Grant No. DGE1745301). We also thank Grant Van Horn and Serge Belongie for helpful discussions, along with NVIDIA and AWS for their kind donations.

References

[1] GBIF - www.gbif.org. 2019. 2

[2] iNaturalist - www.inaturalist.org. 2019. 2

[3] M. Barbet-Massin, F. Jiguet, C. H. Albert, and W. Thuiller. Selecting pseudo-absences for species distribution models: how, where and how many? *Methods in Ecology and Evolution*, 2012. 2, 4

[4] J. Beck, M. Böller, A. Erhardt, and W. Schwanghart. Spatial bias in the gbif database and its effect on modeling species' geographic distributions. *Ecological Informatics*, 2014. 7

[5] T. Berg, J. Liu, S. Woo Lee, M. L. Alexander, D. W. Jacobs, and P. N. Belhumeur. Birdsnap: Large-scale fine-grained visual categorization of birds. In *CVPR*, 2014. 1, 2, 3, 5, 6

[6] C. Botella, A. Joly, P. Bonnet, P. Monestiez, and F. Munoz. A deep learning approach to species distribution modelling. In *Multimedia Tools and Applications for Environmental & Biodiversity Informatics*. 2018. 2

[7] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification. *Pattern Recognition*, 2004. 2

[8] S. Branson, G. Van Horn, S. Belongie, and P. Perona. Bird species categorization using pose normalized deep convolutional nets. In *BMVC*, 2014. 2

[9] D. Chen and C. P. Gomes. Bias reduction via end-to-end shift learning: Application to citizen science. In *AAAI*, 2019. 2, 7

[10] D. Chen, Y. Xue, S. Chen, D. Fink, and C. Gomes. Deep multi-species embedding. In *IJCAI*, 2017. 2, 3, 4

[11] Y.-n. Chen and H.-t. Lin. Feature-aware label space dimension reduction for multi-label classification. In *NeurIPS*, 2012. 2

[12] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019. 2

[13] Y. Cui, Y. Song, C. Sun, A. Howard, and S. Belongie. Large scale fine-grained categorization and domain-specific transfer learning. In *CVPR*, 2018. 6

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[15] A. Dubey, O. Gupta, P. Guo, R. Raskar, R. Farrell, and N. Naik. Pairwise confusion for fine-grained visual classification. In *ECCV*, 2018. 2

[16] J. Elith and C. H. Graham. Do they? how do they? why do they differ? on finding reasons for differing performances of species distribution models. *Ecography*, 2009. 2

[17] R. Engler, A. Guisan, and L. Rechsteiner. An improved approach for predicting the distribution of rare and endangered species from occurrence and pseudo-absence data. *Journal of applied ecology*, 2004. 2

[18] D. Fink, W. M. Hochachka, B. Zuckerberg, D. W. Winkler, B. Shaby, M. A. Munson, G. Hooker, M. Riedewald, D. Sheldon, and S. Kelling. Spatiotemporal exploratory models for broad-scale survey data. *Ecological Applications*, 2010. 1, 2

[19] W. Fithian and R. Mazumder. Flexible low-rank statistical modeling with missing data and side information. *Statistical Science*, 2018. 2

[20] S. Franceschini, E. Gandola, M. Martinoli, L. Tancioni, and M. Scardi. Cascaded neural networks improving fish species prediction accuracy: the role of the biotic information. *Scientific Reports*, 2018. 3

[21] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell. Compact bilinear pooling. In *CVPR*, 2016. 2

[22] G. E. Garrard, M. A. McCarthy, N. S. Williams, S. A. Bekessy, and B. A. Wintle. A general model of detectability using species traits. *Methods in Ecology and Evolution*, 2013. 3

[23] N. Golding and B. V. Purse. Fast and flexible bayesian species distribution modelling using gaussian processes. *Methods in Ecology and Evolution*, 2016. 2

[24] D. J. Harris. Generating realistic assemblages with a joint species distribution model. *Methods in Ecology and Evolution*, 2015. 2, 3

[25] T. Hastie and W. Fithian. Inference from presence-only data; the ongoing controversy. *Ecography*, 2013. 2

[26] J. Hays and A. A. Efros. Im2gps: estimating geographic information from a single image. In *CVPR*, 2008. 2

[27] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[28] T. M. Hegel, S. A. Cushman, J. Evans, and F. Huettmann. Current state of the art for statistical modelling of species distributions. In *Spatial Complexity, Informatics, and Wildlife Conservation*. 2010. 2

[29] S. Huang, Z. Xu, D. Tao, and Y. Zhang. Part-stacked cnn for fine-grained visual categorization. In *CVPR*, 2016. 2

[30] A. Hyvärinen and E. Oja. Independent component analysis: algorithms and applications. *Neural networks*, 2000. 7

[31] A. Khosla, N. Jayadevaprakash, B. Yao, and F.-F. Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *CVPR Workshop on Fine-Grained Visual Categorization*, 2011. 2

[32] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv*, 2014. 5

[33] J. Krause, B. Sapp, A. Howard, H. Zhou, A. Toshev, T. Duerig, J. Philbin, and L. Fei-Fei. The unreasonable effectiveness of noisy data for fine-grained recognition. In *ECCV*, 2016. 2

[34] A. M. Lechner, W. T. Langford, S. D. Jones, S. A. Bekessy, and A. Gordon. Investigating speciesenvironment relationships at multiple scales: Differentiating between intrinsic scale and the modifiable areal unit problem. *Ecological Complexity*, 2012. 2

[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[36] T.-Y. Lin, A. RoyChowdhury, and S. Maji. Bilinear cnn models for fine-grained visual recognition. In *ICCV*, 2015. 2

[37] J. Liu, A. Kanazawa, D. Jacobs, and P. Belhumeur. Dog breed classification using part localization. In *ECCV*, 2012. 2

[38] L. v. d. Maaten and G. Hinton. Visualizing data using t-sne. *JMLR*, 2008. 7, 8

[39] D. I. MacKenzie. What are the issues with presence-absence data for wildlife managers? *The Journal of Wildlife Management*, 2005. 2

[40] D. I. MacKenzie, J. D. Nichols, G. B. Lachman, S. Droege, J. Andrew Royle, and C. A. Langtimm. Estimating site occupancy rates when detection probabilities are less than one. *Ecology*, 2002. 3

[41] S. Mastrorillo, S. Lek, F. Dauba, and A. Belaud. The use of artificial neural networks to predict the presence of small-bodied fish in a river. *Freshwater Biology*, 2003. 2

[42] J. Moat, S. P. Bachman, R. Field, and D. S. Boyd. Refining area of occupancy to address the modifiable areal unit problem in ecology and conservation. *Conservation Biology*, 2018. 2

[43] S. Openshaw. The modifiable areal unit problem. 1983. 2

[44] S. L. Özesmi and U. Özesmi. An artificial neural network approach to spatial habitat modelling with interspecific interaction. *Ecological Modelling*, 1999. 2

[45] S. J. Phillips, M. Dudík, J. Elith, C. H. Graham, A. Lehmann, J. Leathwick, and S. Ferrier. Sample selection bias and presence-only distribution models: implications for background and pseudo-absence data. *Ecological applications*, 2009. 2

[46] S. J. Phillips, M. Dudík, and R. E. Schapire. A maximum entropy approach to species distribution modeling. In *ICML*, 2004. 2

[47] L. J. Pollock, R. Tingley, W. K. Morris, N. Golding, R. B. O'Hara, K. M. Parris, P. A. Vesk, and M. A. McCarthy. Understanding co-occurrence by modelling species simultaneously with a joint species distribution model (jsdm). *Methods in Ecology and Evolution*, 2014. 2

[48] B. L. Sullivan, C. L. Wood, M. J. Iliff, R. E. Bonney, D. Fink, and S. Kelling. ebird: A citizen-based bird observation network in the biological sciences. *Biological Conservation*, 2009. 2, 5, 6

[49] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 5, 6

[50] K. Tang, M. Paluri, L. Fei-Fei, R. Fergus, and L. Bourdev. Improving image classification with location context. In *ICCV*, 2015. 1, 2, 3, 5, 6

[51] L. Tang, Y. Xue, D. Chen, and C. P. Gomes. Multi-entity dependence learning with rich context via conditional variational auto-encoder. In *AAAI*, 2018. 1, 2, 3

[52] G. Van Horn, S. Branson, R. Farrell, S. Haber, J. Barry, P. Ipeirotis, P. Perona, and S. Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *CVPR*, 2015. 5

[53] G. Van Horn, O. Mac Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. Belongie. The inaturalist species classification and detection dataset. In *CVPR*, 2018. 2, 5, 6, 7, 8, 11

[54] K. Vishal, C. Jawahar, and V. Chari. Accurate localization by fusing images and gps signals. In *CVPR Workshop*, 2015. 2

[55] N. Vo, N. Jacobs, and J. Hays. Revisiting im2gps in the deep learning era. In *ICCV*, 2017. 2

[56] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 2, 5

[57] F. Wang, M. Jiang, C. Qian, S. Yang, C. Li, H. Zhang, X. Wang, and X. Tang. Residual attention network for image classification. In *CVPR*, 2017. 2

[58] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu. Cnn-rnn: A unified framework for multi-label image classification. In *CVPR*, 2016. 2

[59] H. C. Wittich, M. Seeland, J. Wäldchen, M. Rzanny, and P. Mäder. Recommending plant taxa for supporting on-site species identification. *BMC Bioinformatics*, 2018. 1, 2

[60] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *CVPR*, 2015. 2

[61] L. Yang, P. Luo, C. Change Loy, and X. Tang. A large-scale car dataset for fine-grained categorization and verification. In *CVPR*, 2015. 2

[62] P. Yen, F. Huettmann, and F. Cooke. A large-scale model for the at-sea distribution and abundance of marbled murrelets (*Brachyramphus marmoratus*) during the breeding season in coastal british columbia, canada. *Ecological Modelling*, 2004. 2

[63] M.-L. Zhang and Z.-H. Zhou. Multi-label neural networks with applications to functional genomics and text categorization. *IEEE Transactions on Knowledge and Data Engineering*, 2006. 2

[64] N. Zhang, J. Donahue, R. Girshick, and T. Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, 2014. 2

[65] N. Zhang, R. Farrell, F. Iandola, and T. Darrell. Deformable part descriptors for fine-grained recognition and attribute prediction. In *ICCV*, 2013. 2

[66] B. Zhao, X. Wu, J. Feng, Q. Peng, and S. Yan. Diversified visual attention networks for fine-grained object classification. *IEEE Transactions on Multimedia*, 2017. 2

Supplementary Material

Here, we present additional analysis of the results and details of the model presented in the main paper.

A. Supplementary Results

In Fig. 7 we observe the per-category performance changes on iNat2018 [53] resulting from using our prior compared to using no spatio-temporal prior *i.e.* the raw output of the classifier. On the right of the plot we see large improvements for several categories *e.g.* the 'Angulate Tortoise' who is predicted to be found in South Africa. Capturing this geographic specialization enables the prior to rule out instances of this category in other locations. On the left of the plot we see a small number of cases where the performance decreases after applying the prior *e.g.* the 'White-dotted Groundling'. In this particular instance, there are only a small number of training locations (13) for this category, potentially causing the spatio-temporal prior to underestimate its true range.

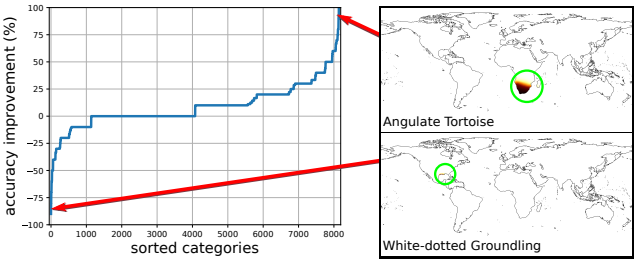

Figure 7: **Accuracy improvement.** Here we see the categories in iNat2018 sorted by how much their absolute classification accuracy improves after applying our spatio-temporal prior. Values greater than 0 indicate categories where the spatio-temporal prior improves accuracy.

B. Training Details

In Fig. 8 we illustrate the architecture of our location encoder $f()$. As described in the main paper, each coordinate x^l of the input spatio-temporal location vector $\mathbf{x}$ is mapped to $[\sin(\pi x^l), \cos(\pi x^l)]$, resulting in two numbers for each input dimension. This is then passed through an initial fully connected layer, follower by a series of residual blocks, each consisting of two fully connected layers with a dropout layer in between. In practice, we set the number of hidden units in each fully connected layer and the output embedding to 256. In total we use four residual blocks (*i.e.* $B = 4$ in Fig. 8).

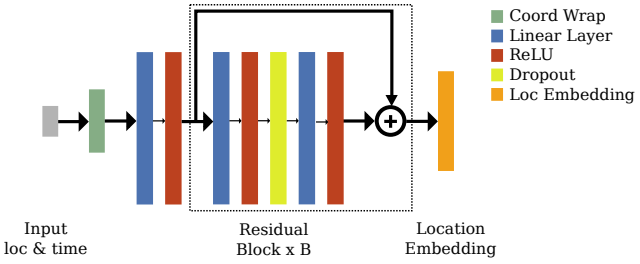

Figure 8: **Location encoder.** Our location encoder $f()$ is a fully connected neural network.