

EcoScape: A Model for Species-Specific Functional Habitat Connectivity

UC Santa Cruz Technical Report UCSC-SOE-23-10

December 7, 2023

Natalia Ocampo-Peñuela^{1*†}, Luca de Alfaro^{2*†}, Jasmine Tai²,
Natalie Valett², Coen Adler², Artie Nazarov²

¹Environmental Studies, University of California, Santa Cruz, 1156 High St, Santa Cruz, 95064, California, United States.

²Computer Science and Engineering, University of California, Santa Cruz, 1156 High St, Santa Cruz, 95064, California, United States.

*Corresponding author(s). E-mail(s): nocampop@ucsc.edu;
luca@ucsc.edu;

Contributing authors: cjtai@ucsc.edu; nvalett@ucsc.edu;
ctadler@ucsc.edu; anazarov@ucsc.edu;

[†]These authors contributed equally to this work.

Abstract

Habitat connectivity is essential to maintain viable and healthy populations due to increased genetic flow, enhanced functional habitat, and repopulation potential [1–4]. Conserving, enhancing, and restoring habitat connectivity has been identified as a priority to achieve global targets of halting extinctions and safeguarding 30% of our planet [5]. Computational constraints of existing connectivity models have restricted these analyses to coarse scales and to a few charismatic species and/or landscapes [6–8].

We introduce EcoScape, an efficient algorithm for computing the pixel-level functional connectivity of species habitats. Informed by metapopulation theory, EcoScape computes the connectivity of each habitat pixel as the probability that the pixel can be repopulated via propagation from other habitat locations, taking into account the species' ecological preferences and dispersal abilities [9]. We provide a methodology based on open science data for applying EcoScape

to any bird species. We show that for two forest/woodland bird species in California, EcoScape habitat connectivity values are highly correlated with relative abundances reported in eBird [10]. EcoScape leverages Graphic Processing Units (GPUs), and is hundreds of times faster than existing connectivity models. This opens the way to mapping habitat connectivity at large scale and fine resolutions for multiple species, offering guidance for current and future conservation efforts.

1 Main

Habitat connectivity is essential for the long-term viability of animal and plant populations because it enhances species movement in the landscape, increasing the amount of functional habitat, and ultimately maintaining genetic diversity [11–14]. Conserving and enhancing habitat connectivity has become increasingly important to allow shifting distributions and species adaptation under future climatic scenarios [15–17] and has thus been prioritized in global frameworks such as the Convention on Biological Diversity (CBD) [18].

Conservation planning benefits from spatially-explicit models of habitat connectivity to inform decisions and evaluate outcomes [19]. To reach the CBD goal of protecting 30% of the planet to effectively safeguard biodiversity, we need to prioritize areas that protect and enhance landscape connectivity for as many species as possible, while considering future climatic scenarios [15, 16]. However, current efforts to map habitat connectivity in current and future scenarios have focused on a few charismatic species (e.g. large mammals) [20], limited spatial scales (e.g. specific landscapes) [21], or only within protected areas [22].

Computational constraints have limited the scope and scale at which habitat connectivity models are applied. Currently, widely used connectivity models are based on electrical circuit theory [6, 23–26] and graph theory [7, 8]. These models are limited in the scale and magnitude of data they can compute, and the speed at which they can compute it [27, 28]; see recent advances using the Julia language [25, 28].

To make connectivity modeling a truly useful mechanism for informing conservation decisions, faster connectivity models are needed to allow modeling for large regions and suites of species. As a step in this direction, we present EcoScape, an algorithm for computing connectivity that is order of magnitudes faster than prior approaches. EcoScape implements a propagation-based approach that is rooted in metapopulation theory [29] and can be implemented very efficiently using Graphical Processing Units (GPUs). In metapopulation theory, a species' population is viewed as consisting of individual subpopulations, each inhabiting a habitat patch. The local subpopulations are unstable, and can locally become extinct; thus the long-term survival of the species depends on the ability of the patches to be repopulated via dispersal from other source patches [30, 31].

EcoScape computes the *connectivity* of each individual habitat pixel as the probability that the pixel can be repopulated from other habitat pixels. To estimate this probability, EcoScape repeatedly simulates the species' propagation from randomly-chosen habitat locations, or seeds. The simulations take into account species-specific

information, such as the dispersal distance of the species, its affinity for particular landcovers, and its gap-crossing ability over unfavorable landcover. The sum of the connectivity for all pixels within a species habitat yields the *functionally connected habitat* (FCH).

We validate EcoScape’s approach on two California forest/woodland birds: Acorn Woodpecker (*Melanerpes formicivorus*) and Steller’s Jay (*Cyanocitta stelleri*). These species are easily detected and frequently observed non-migratory species that have close association with forest/woodland, allowing for precise area-of-habitat estimation [32]. We show that for these species, the connectivity computed by EcoScape is highly correlated with relative abundance estimates derived from eBird records [10]. To the best of our knowledge, this represents the first time a connectivity model has been validated via fine-grained observations over an extended region.

EcoScape is hundreds of times faster than prior connectivity models [8, 25, 26, 28]. The efficiency of EcoScape is due to two factors. The propagation model has been crafted to take full advantage of the high parallelism provided by modern graphic processing units (GPUs) and machine-learning frameworks. Further, algorithms inspired by circuit theory need to consider both current *sources* and *sinks*. Propagation models only require *sources*, allowing for faster and highly parallel computation as they do not need to iterate through all *sinks*.

Finally, we provide a workflow that draws on the power of open biodiversity data from global and citizen-science platforms such as the International Union for the Conservation of Nature (IUCN) [33], eBird [10], and AVONET [34] to enable the study of all bird species. This eliminates one of the chief bottlenecks to the study of habitat connectivity, namely, the fact that the extensive input required for each species is generally not available [28].

Via efficient algorithms, and data workflows that can be applied to any bird species, we aim to bring *continental-scale connectivity analysis* within reach: the study of habitat connectivity at fine spatial resolutions, over continental extents, and for a large suite of species.

2 EcoScape: Habitat Connectivity as Habitat Repopulation

EcoScape computes the connectivity of habitat pixels as the probability that the pixels can be repopulated via propagation from other habitat pixels.

The input to EcoScape consists of two spatial raster layers: a *habitat* layer h and a *landscape matrix permeability* layer p . A habitat pixel h_{ij} at coordinates i, j can have value 0 (non-habitat) or 1 (habitat). A permeability pixel p_{ij} has values in the interval $[0, 1]$: 0 indicates no possibility of species propagation, and 1 full possibility of propagation. Thus, permeability is inversely related to the standard matrix resistance used in most existing connectivity methods [6, 24]. We assume that the permeability in the habitat is $p = 1$.

The model also takes as input the dispersal distance d and the gap-crossing distance g of the species. A gap-crossing distance of g indicates that birds can fly over up to $g - 1$ pixels regardless of landcover class. The dispersal distance d indicates how far the

species can propagate from seeds. Both distances are expressed in pixels; we assume that d is a multiple of g , or $d = ng$. A good rule of thumb is to choose a pixel size that is only somewhat smaller than the gap-crossing g ; in the study, we used $300m$ pixels and $g = 2$.

EcoScape outputs two raster layers: a *connectivity* raster c , where $c_{ij} \in [0, 1]$ represents the probability that a habitat pixel ij can be repopulated, and a *flow* raster f , where f_{ij} represents the amount of habitat repopulation that happened due to propagation through pixel ij .

To compute connectivity, EcoScape repeatedly simulates how the species can repopulate its habitat by propagating from randomly-chosen seed locations. In each simulation, a habitat pixel is chosen as a seed randomly and independently with probability $1/d^2$, so that there is on average one seed for every d^2 -sized habitat region. EcoScape initializes a *repopulation* raster layer r , setting it to 1 at seed pixels and to 0 everywhere else. EcoScape then simulates n rounds of propagation (so $d = ng$), updating in each round the repopulation r_{ij} of pixel ij via:

$$r_{ij} := \max \left\{ r_{ij}, p_{ij} \cdot \max_{kl \in N_g(ij)} (X_{kl} \cdot r_{kl}) \right\}, \quad (1)$$

where $p_{ij} \in [0, 1]$ is the landscape matrix permeability of pixel ij , $N_g(ij)$ is a g -sized neighborhood of ij , and $X_{kl} \in [0.9, 1]$ is a random number chosen independently for each kl . The neighborhood $N_g(ij)$ consists of the pixels that are within gap-crossing distance g from ij ; thus, in each round the birds can propagate up to g pixels. To obtain an efficient implementation, we let $N_g(ij)$ be the square centered around ij and with edge $2g + 1$, that is, $N_g(ij) = \{kl \mid i - g \leq k \leq i + g, j - g \leq l \leq j + g\}$. The random factor X_{kl} models the probability that the population at pixel kl will propagate in the round. The random factor acts as a tie-breaker for the max operator. Further, the randomness at each simulation step ensures that multiple propagation paths are chosen, to avoid highlighting a single least-cost path [6, 14, 35].

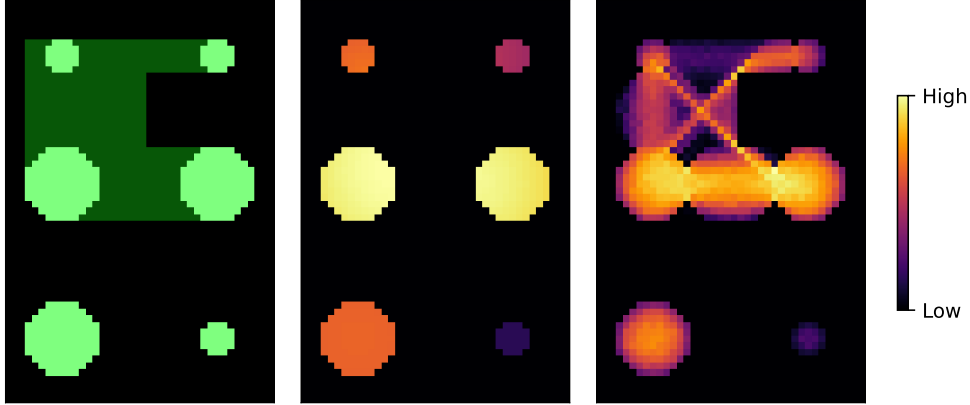
The propagation (1) is loosely inspired by diffusion processes [9] and allows efficient implementation on GPUs via a max-convolution computation in the machine-learning framework PyTorch [36].

The *connectivity* layer c is obtained by computing the average \bar{r} of r over many simulations and by clipping the result to the habitat via $c = h\bar{r}$ (i.e., $c_{ij} = h_{ij}\bar{r}_{ij}$ at every pixel ij). Thus, $c_{ij} = 0$ if $h_{ij} = 0$, and $0 \leq c_{ij} \leq 1$ if $h_{ij} = 1$. The total area of habitat is $H = a \sum_{ij} h_{ij}$, where a is the area of a pixel. We define the *functionally connected habitat* (FCH) as $FCH = a \sum_{ij} c_{ij}$: in this sum, every habitat pixel contributes an amount proportional to its area, and to the probability with which it can be repopulated. The ratio FCH/H expresses FCH as a percentage of the area of habitat that is functionally connected.

The flow layer is computed via

$$f_{ij} = p_{ij} \frac{\partial FCH}{\partial p_{ij}}. \quad (2)$$

The *flow* f_{ij} through pixel ij represents the amount of repopulation due to propagation through ij , as shown in the methods. Unlike connectivity, the *flow* can be non-zero outside of the habitat. The *flow* value at pixels outside the habitat yields information



(A) Habitat and landscape matrix (B) Connectivity layer (C) Flow layer

Fig. 1: EcoScape connectivity computation on an artificial 40×60 -pixel habitat and landscape matrix. (A): habitat patches (in light green) are connected by high permeability matrix pixels (in dark green). (B): Connectivity layer. Larger patches have higher connectivity; for equal-sized patches, connected patches have greater connectivity than isolated ones. (C): Flow layer. Multiple paths connect habitat patches according to the landscape matrix permeability.

on the role of landscape matrix pixels in maintaining connectivity. The *flow* can be computed via back-propagation in PyTorch, considering FCH as the output, and p as a layer of model parameters.

The efficiency of EcoScape depends on two factors. First, propagation (1) and flow computation (2) can be implemented on modern machine-learning frameworks, and take advantage of the large parallelism offered by GPU computation. Second, we argue that propagation models for connectivity are inherently more efficient than circuit-based ones. For example, to compute the connectivity of a pixel, Omniscape must designate the pixel as a sink, select current sources, and solve the circuit equations to compute the current into the pixel [26]; the process must be repeated for each pixel. Propagation and diffusion, in contrast, can use just sources to compute repopulation simultaneously at all pixels.

The EcoScape computation of *connectivity* and *flow* is illustrated on an artificial landscape in Figure 1. We have implemented the algorithms in the Python package [37]; where we also provide sample Python notebooks that demonstrate how to produce *connectivity* and *flow* layers for example species.

Functionally Connected Habitat and Equivalent Connected Area.

The *functionally connected habitat* (FCH) computed by EcoScape is closely related to the widely used Equivalent Connected Area (ECA) [38]. Consider habitat patches H_1, \dots, H_n , where patch H_k has area A_k , $1 \leq k \leq n$, and assume as a simplification

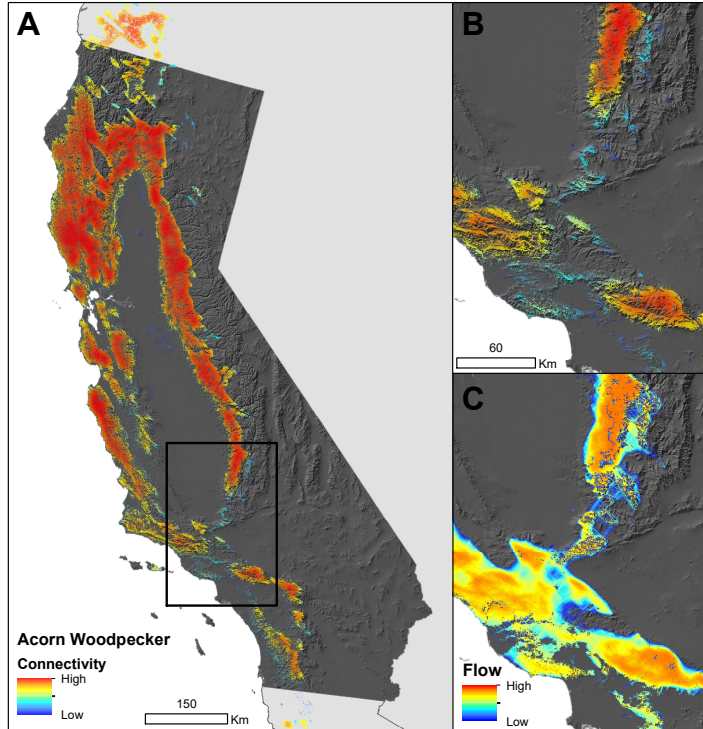


Fig. 2: Connectivity (A, B) and flow (C) layers for the Acorn Woodpecker (*Melanerpes formicivorus*) in California using a 600m gap-crossing distance and 12 km dispersal distance calculated using EcoScape. Panel A shows the bounding box expanded in panels B and C

that the probability with which a bird can propagate from a patch H_k to patch H_l is equal to P_{kl} , and is independent on the precise source and destination locations in the patches. A seed occurs in patch H_k with probability proportional to A_k , and such seed can repopulate the area A_l of H_l with probability P_{kl} . Thus, the contribution of seeds in H_k for the repopulation of H_l is $A_l A_k P_{kl}$, and the total connectivity is proportional to $\sum_{kl} A_k A_l P_{kl}$, which is the ECA. Thus, the *functionally connected habitat* (FCH) computed by EcoScape is consistent with established ideas in landscape connectivity and metapopulation theory.

3 Results

We used EcoScape to compute the *connectivity* and *flow* layers for two non-migratory bird species: Acorn Woodpecker (*Melanerpes formicivorus*) and Steller's Jay (*Cyanocitta stelleri*). The species were selected for their association with forest/woodland, and ease of detection, facilitating the validation with citizen-science data. We studied these species in California at a resolution of 300m pixels. We obtained

the *habitat* layer by refining eBird range maps [39] based on habitat preference data from IUCN [40, 41] to produce an Area of Habitat map [32]. We produced the *landscape matrix permeability* layer using a global habitat class map [40], species habitat preferences from IUCN, and habitat use frequency from eBird observations [10] (see Methods for details). We used 600m (2 pixels) as the gap-crossing distance for both species. The dispersal distances were derived from the species’ Hand-Wing Index (HWI) [34], which is a good proxy for dispersal [42]. We used 12 km for Acorn Woodpecker and 1.8 km for Steller’s Jay. The layers and parameters used for these two example species were obtained from open data sources in a fully automated approach, which can be extended to more species (see Methods).

Figure 2 depicts the *connectivity* (A, B) and *flow* (C) layers for the Acorn Woodpecker. Habitat patches that are larger, and closer to other large patches, have higher connectivity values, as expected; while small, isolated patches have connectivity values closer to zero. The *flow* layer shows the importance of habitat and non-habitat pixels to the connectivity of the Acorn Woodpecker and is useful to identify existing stepping stones and corridors that can be prioritized for conservation. In the area of study, EcoScape estimated the FCH as 50,558 km² (76% of the habitat) for the Acorn Woodpecker, and 68,915 km² (89% of the habitat) for the Steller’s Jay.

Precision.

EcoScape computes the *connectivity* and *flow* layers as the average of multiple simulations. We computed the standard deviation of the connectivity at every pixel. In our study conditions, averaging 400 simulations resulted in fewer than 1% of pixels having standard deviation greater than 0.02, or 2% of the overall connectivity value range [0, 1]; no pixel had standard deviation greater than 0.05.

Correlation with eBird data.

We validated the *connectivity* layers computed by EcoScape (N=10,000 simulations) by comparing them with relative abundance of birds, measured as the frequency of observation for a species, derived from the citizen-science platform eBird [10]. eBird is a citizen-science platform in which birdwatchers log sightings in *checklists* with observation events consisting of: date, geographical location, distance traveled, and count for each species observed, among other data. We only considered complete eBird checklists with a maximum traveling distance of 2 km, during the breeding season (April to June), from 2012 to 2018 to match the 2015 landcover used [40] and to exclude the influence of the 2020 California wildfires (N=185,996 checklists). We grouped the checklists using a ≈ 1 km grid, and computed the average number of individuals of the species reported per checklist at each grid location within the study area (Acorn Woodpecker: N=6,818 grid locations; Steller’s Jay: N=5,806). This provides an observation frequency at each of the above grid locations. We then subdivided the connectivity values [0, 1] into 10 intervals [0, 0.1), [0.1, 0.2), ..., [0.9, 1], and we computed the average observation frequency of the locations falling in each connectivity interval. Finally, we measured the coefficient of determination R^2 between the average observation frequencies in the 10 intervals, and the 10 interval endpoints, using as weights the number of locations falling in each connectivity interval. The results indicate that

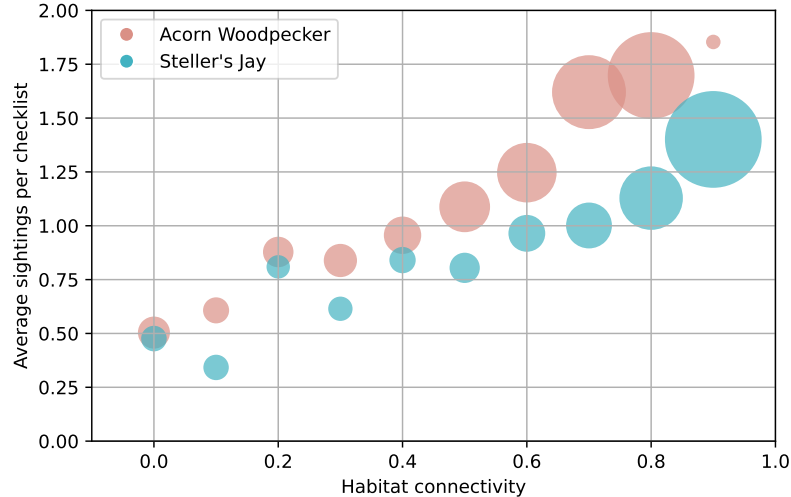


Fig. 3: Average number of individuals reported per checklist within the species habitat as a function of habitat connectivity at the checklist location. The area of a circle is proportional to the number of habitat grid locations with checklists that fall within the connectivity interval (total grid locations: Acorn Woodpecker: $N=6,818$; Steller’s Jay: $N=5,806$). The connectivity was computed with EcoScape using gap-crossing distance $g = 600m$, and dispersal distance of 12 km for Acorn Woodpecker and 1.8 km for Steller’s Jay.

average relative bird abundances are strongly correlated with connectivity, with a coefficient of determination R^2 of 0.95 for Acorn Woodpecker and 0.90 for Steller’s Jay. The relationship between sightings and connectivity for the two species is illustrated in Figure 3; the data backing the figure is provided in the supplementary material. In the supplementary material, we provide measurements of R^2 for many values of the gap-crossing distance g and number of crossings n . Intriguingly, the results show that R^2 remains high for $d = ng$ equal to or below the dispersal distance predicted by the Hand-Wing Index, and then declines, suggesting that longer distances do not explain bird sightings.

Comparison with Omniscap and Patch Area.

Omniscap [26] is the habitat connectivity tool whose output can be most directly compared to that of EcoScape. Omniscap uses a notion of analysis radius which is equivalent to EcoScape’s dispersal distance, and computes connectivity as the amount of current that can flow to a pixel from habitat locations within analysis radius from the pixel. We have run Omniscap on the study area, using conductance values proportional to our landscape matrix permeability (see the Methods, Section 5.5.2). The results (Table 1) indicate that Omniscap’s and EcoScape’s outputs are both highly correlated with eBird records; EcoScape is faster by a factor of about 100. We view the fact that EcoScape’s results roughly match those of a widely-used tool such as

Species	EcoScape		Omniscape		Patch Area		N
	R^2	Time	R^2	Time	R^2	Time	
Acorn Woodpecker	0.95	16s	0.79	1866s	0.40	21s	6,818
Steller’s Jay	0.90	5s	0.97	772s	0.16	30s	5,806

Table 1: Time required to run EcoScape (400 simulations), Omniscape, and patch-area computation, along with the coefficient of determination R^2 between output (connectivity, current, and log patch area, respectively) and average sightings per checklist reported in eBird, and number N of grid locations using in the estimation of R^2 . The computation used Google Colab [43] with A100 GPU (EcoScape and patch area) and an Intel i7 2.3GHz CPU (Omniscape). In the case of patch size, the time reported is an upper bound.

Omniscape [44] as additional validation of the proposed method. We note that the parameters used by both tools can be fine-tuned, and the correlation values vary with the precise values of landscape matrix permeability and conductance.

To test if our connectivity approach is more informative than merely considering habitat patch size, we implemented a fast GPU-based algorithm for the computation of patch sizes. Due to the very large range of patch sizes (from 0.1 km² to over 10,000 km²), we use the log patch size as the model output. The results (Table 1) indicate that while patch size can be computed efficiently, the correlation with eBird records is weaker (full data backing this analysis can be found in the supplementary material).

Relation with other connectivity models.

The potential of diffusion to represent animal movement has been explored in [9, 45], where the results are validated on a species of butterflies at a landscape scale.

Several habitat connectivity models relying on circuit and graph-theory are computationally intensive, when compared to EcoScape. Efforts to speed them up have included coding in the Julia language [25, 28], and stand-alone packages such as Gflow [46] and ConScape [28]. The fastest current approach is ConScape [28], which can process a 10,000 pixel territory in 72 seconds; EcoScape can process the same extent in 0.02 seconds, or over 1,000 times faster.

The outputs of EcoScape represent two layers that usually require running two different pieces of software. Furthermore, EcoScape’s *Functionally Connected Habitat* is calculated much faster and at higher resolution than the related Equivalent Connected Area produced by graph-theory algorithms such as CONEFOR [8]. The *flow* layer produced by EcoScape is akin to least cost path corridors identified by software such as LinkageMapper [47] and Omniscape [26], but it is produced during the same run as the *connectivity* layer.

4 Discussion

EcoScape is a novel, propagation-based, pixel-level, and species-specific algorithm to map functional habitat connectivity and flow at fine resolutions and continental scales.

Conserving, enhancing, and increasing habitat connectivity has been identified as an important mechanism to combat habitat loss, fragmentation, and climate change impacts [16, 48]. To achieve global targets of increased connectivity at landscape scales, we must produce spatially-explicit models that highlight areas of high value for conserving and restoring connectivity for multiple species [49, 50]. Current efforts to address this challenge have fallen short because they only account for a few charismatic and well-studied species (e.g. pumas [51], jaguars [52]), or are limited to recommendations only within protected areas [22].

EcoScape contributes to this task by significantly speeding up the computation of functional habitat connectivity, making it practical to map hundreds of species at once. The speed of EcoScape also facilitates large-scale scenario exploration for connectivity, e.g., with respect to future climate conditions [17] by replacing the habitat and landcover layers with projected species distributions or habitat loss or gain, to understand how current decisions will play out in the future (for example see [53]).

The workflow we developed relies on open data from IUCN, eBird, and AVONET, and provides users Python packages for every step of the process, providing a starting point for the study of any bird species globally. Dispersal distances, in particular, are challenging to estimate for most species and the lack of information has limited the application of connectivity models for lesser-known species. Though not perfect, we propose using the Hand-Wing index [42] to estimate dispersal as an input for EcoScape for birds. A similar idea has been explored with body mass for mammals [54], and applied to existing connectivity models [55].

We also present the first validation of a habitat connectivity model via large-scale, freely available citizen science data. Existing connectivity models have been tested via simulation experiments [56], using animal tracking data [57, 58], and more rarely with genetic data [59]. The limited availability of tracking data, which currently exists mostly for large and charismatic fauna [60], has been an obstacle to the validation of connectivity models for lesser-known species. In contrast, the validation framework we introduced can be used for any bird species with sufficient eBird data [10], and potentially for other taxa using data from iNaturalist [61] or other taxon-specific citizen science platforms (e.g. eMammal).

As ecology and conservation enter the world of big data [27], technological challenges arise from processing and taking full advantage of the fine resolution and large quantity of data. EcoScape addresses this challenge by providing users an efficient, easy to use, and open way to map habitat connectivity and flow at fine resolutions over continental scales. These advances will allow ecologists and conservation planners to fully consider a spectrum of lesser-known species and diverse scenarios in their conservation plans.

Data accessibility statement.

All the data required for replicating the results is available at [10.5281/zenodo.8395864](https://doi.org/10.5281/zenodo.8395864).

Code availability statement.

The process for reproducing the results, and all the required code, are available on github.com/ecoscape-earth/ecoscape-connectivity-paper-reproduction and archived at [10.5281/zenodo.8395874](https://zenodo.org/record/8395874).

All code is open source. To promote usage and experimentation, the code is organized into modules that can be installed via the standard Python package manager `pip`. The package `ecoscape-connectivity` provides the EcoScape algorithm implementation, while `ecoscape-layers` enables the preparation of the input habitat and landscape matrix raster layers using IUCN and eBird data. These two packages rely in turn on the package `scgt` (for *Santa Cruz Geographical Toolkit*), which simplifies working on geographical raster layers. The packages include links to their open-source code repositories and documentation.

Acknowledgments.

We thank Dr. Tyler Sorensen for helpful discussions on adapting algorithms to GPU computation, and Dr. Santiago Claramunt for helpful discussions to parameterize dispersal distances using the Hand-Wing Index. We thank Oscar Godinez for initial discussions exploring the comparison of EcoScape to existing connectivity models. Dr. Chris Wilmers, Dr. Reza Goljani Amirkhiz, and Whitney Hansen for helpful comments on the manuscript.

Funding statement.

This work was partly funded by a Google Research gift.

Author contribution statement.

NOP and LdA jointly conceived and directed the study; NOP led the ecological component of the study, and LdA led the computer science one. JT was the main contributor to the code and process that produced the geographical raster layers used in the study; JT and NV jointly developed the codebase for working on geographical raster layers that is used throughout the paper. LdA, CA and AN contributed to algorithm development and testing. NOP, LdA and NV contributed to the statistical validation of the results. LdA led the data archiving and reproducibility work, with help from JT. NOP and LdA led the writing of the manuscript, with contributions from all other authors.

5 Methods

The overall data and processing flow used in this paper is illustrated in Figure 4. First, we prepare the inputs used by the EcoScape algorithm, namely, the spatial layers encoding the habitat and landscape matrix permeability. Next, we run the EcoScape algorithm on these data, using as parameters a given gap-crossing and dispersal distance. Lastly, we analyze the connectivity layer produced by EcoScape, correlating it with bird observations from the eBird citizen-science platform [10, 62], yielding the validation for our proposed connectivity approach. Everything needed to reproduce the results of this paper is provided in an open GitHub repository [63], accompanied by detailed step-by-step instructions. The portions of code of general interest, such as

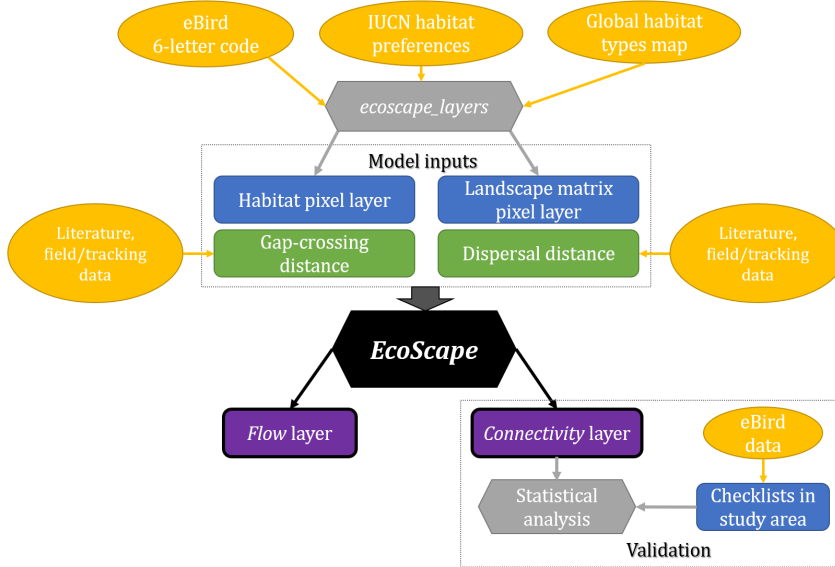


Fig. 4: Flow chart of methods for running the functional habitat connectivity model EcoScape, illustrating specific steps taken in the present analysis. The central square shows model inputs including spatial layers as blue rectangles and numerical inputs as green rectangles; yellow circles show user-provided inputs and our suggested sources; gray hexagon show sister Python package `ecoscape_layers` which is provided to calculate the spatial model input layers; purple rectangles depict model output layers (Connectivity and Flow). Bottom right rectangle shows the validation of the connectivity model used based on eBird data.

the EcoScape algorithm and the input layer preparation, have been published in open-source Python packages that can be installed via the standard `pip` Python package installer [37, 64]. In [63], we also provide sample Python notebooks that allow anyone to experiment with the input layer generation and algorithms on Google Colab [21], without requiring any installation.

We first describe technical aspects of the EcoScape algorithm and its implementation that expand on what is found in the main body of the paper. Then, we describe the preparation of the inputs: habitat and landscape matrix permeability layers, and gap-crossing and dispersal distance. Last, we describe the detailed techniques we used for the statistical validation of the connectivity layer using citizen-science data from eBird [10].

5.1 The EcoScape Algorithm And Its Implementation

EcoScape inputs.

The EcoScape algorithm takes three inputs:

- A boolean (binary) habitat layer h , so that for a pixel ij , $h_{ij} = 1$ if the pixel is in the habitat, and $h_{ij} = 0$ if it is outside.
- A landscape matrix permeability layer p , with $0 \leq p_{ij} \leq 1$: we assume $p_{ij} = 1$ in the habitat, and $p_{ij} = 0$ for landcovers with high resistance for the species; intermediate values correspond to suitable but not ideal landcover for the species.
- A gap-crossing distance g , expressed in number of pixels, and a number of gaps n , so that the dispersal distance d is $d = n \cdot g$.

In practice, in our implementation, we do not provide p directly. Rather, we provide a landcover layer t , where t_{ij} is an integer, describing the landcover class at pixel ij , along with a permeability table D , giving the permeability $D(t) \in [0, 1]$ for every landcover class t . The permeability layer is then obtained as $p_{ij} = D(t_{ij})$. The advantage of this approach is that, when studying multiple species on the same study area (as done in this paper), we can use the same landcover layer; only the permeability table needs to be modified to achieve the appropriate landscape matrix layer. Further, one can experiment with different permeability values without having to rebuild a raster layer.

The EcoScape algorithm.

The EcoScape algorithm computes the connectivity of a habitat pixel as the probability that the pixel can be repopulated from other habitat locations via propagation from randomly-chosen seeds. To do so, EcoScape performs a number of repopulation simulations. At the beginning of each simulation, EcoScape initializes the repopulation raster layer r via

$$r_{ij} = h_{ij} \cdot \delta(Y_{ij} < 1/d^2),$$

where Y_{ij} is a random variable with value uniformly distributed in $[0, 1]$ sampled independently at each pixel, d is the dispersal distance, and δ is a *characteristic function* so that $\delta(\phi)$ is 1 if ϕ is true, and 0 otherwise. Thus, in each $d \times d$ square there is on average one seed.

Once the raster layer r is initialized, EcoScape performs n iterations of the propagation update (1). In the update, as mentioned, the random variable X is used to break ties at random between source locations where c has the same value. If we did not use such a random tie-break, the nature of the GPU implementation would favor a fixed origin for the birds (in our experiments, the upper-left corner), leading to large artifacts and invalidating the output. The randomness also ensures that the propagation is multi-path, rather than using exclusively the path of least resistance. We let X assume values uniformly at random in the open interval $(0.9, 1)$, sampled independently at every pixel. The fact that $X < 1$ can be interpreted as the fact that birds at a pixel have a small probability, at each propagation step, of choosing not to propagate in the round.

The random tie-break can also be understood as follows. Let $r^{(0)}$ be the repopulation layer after seed initialization, and let $r^{(m)}$ be the layer after the m -th iteration of (1), so that (1) can be rewritten as

$$r_{ij}^{(m+1)} = \max \left\{ r_{ij}^{(m)}, p_{ij} \cdot \max_{kl \in N_g(ij)} (X_{kl} \cdot r_{kl}^{(m)}) \right\}. \quad (3)$$

Then, it is easy to see that the sequence $r^{(0)}, r^{(1)}, r^{(2)}, \dots$ is monotonically increasing, so that at each ij , $r_{ij}^{(0)} \leq r_{ij}^{(1)} \leq r_{ij}^{(2)} \leq \dots$.

To track where the repopulation came from, we can define a *source mapping* S , so that the repopulation at pixel ij came from neighboring pixel $S(ij)$. We define this mapping as follows. Initially, $S(ij) = \perp$ at all ij , where \perp is a special symbol indicating that the source is undefined (the population did not come from another location). When we update the value at ij , that is, when $r_{ij}^{(m)} < r_{ij}^{(m+1)}$, there is a unique neighbor $kl \in N_g(ij)$ that provides the argmax in (3): this kl can be thought of as the *source* of the increased bird population at ij , and therefore, we set $S(ij) := kl$. When the value at ij is unchanged, we leave $S(ij)$ unchanged.

The source mapping S defines a graph, with an edge from ij to kl when $S(ij) = kl$. It is easy to see that this graph is a collection of trees, each tree having root in a repopulation seed. Indeed, there can be no loops in the graph: $S(ij) = kl$ implies $r_{ij}^{(m+1)} < r_{kl}^{(m)}$ at the time $m + 1$ of the last update of kl , and by monotonicity, $r_{ij}^{(n)} < r_{kl}^{(n)}$ after the last update n . Thus, the EcoScape algorithm can be understood as a way to generate random dispersal trees rooted in the repopulation seeds. This alternate characterization is essential to the computation of the flow layer. Finally, the connectivity layer is obtained by clipping the repopulation to the habitat via $c = h r^{(n)}$, where multiplication is pixel-wise.

The flow layer.

The flow at a pixel indicates how much of the connectivity is due to movement through the pixel. To see this, let $c_{ij} = c_{ij}^{(n)}$ be the connectivity, and let S be the source mapping defined above. The locations that owe their repopulation to ij consists of ij , its immediate descendants in the propagation trees, or the kl with $S(kl) = ij$, and their descendants of any level: we call this the *downstream region* of ij (see Figure 5). Precisely, the downstream region of ij is the set of pixels $U(ij)$ that have a path in the graph S to ij . Pixels in $U(ij)$ were repopulated through ij .

If we change the permeability p_{ij} of ij by a small amount δ (small, so as not to change the source mapping), the connectivity in the downstream region of ij will change correspondingly, by $\delta(\partial FCH / \partial p_{ij})$, where $FCH = a \sum_{ij} c_{ij}$. If we keep S fixed, that is, if we fix the flow taken by the updates (3), the total habitat repopulation of the downstream region of ij is $p_{ij}(\partial FCH / \partial p_{ij})$, or:

$$f_{ij} = a \sum_{kl \in U(ij)} c_{kl} = p_{ij} \frac{\partial FCH}{\partial p_{ij}}. \quad (4)$$

This provides a way of computing the flow f_{ij} through a pixel ij using back-propagation in a machine-learning framework.

Implementation.

The EcoScape connectivity algorithm is implemented on top of PyTorch, one of the most widely-used and powerful machine-learning frameworks [36]. The [ecoscape-connectivity](#) Python package provides the algorithm, and can be installed with `pip`. We encode the EcoScape computation in PyTorch by having the initial repopulation

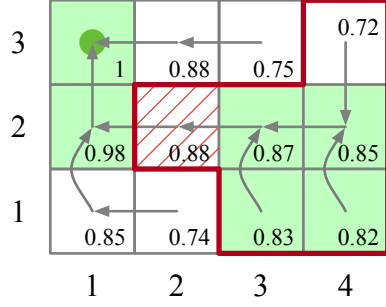


Fig. 5: A repopulation layer r . The value r_{ij} is indicated in each pixel ij ; pixels in the habitat are depicted in green, and pixel $(1,3)$ is the (only) seed, with $r_{1,3} = 1$. The source mapping S is depicted with arrows: for instance, we have $S(1,2) = (1,3)$ (pixel $(1,2)$ points to the sink $(1,3)$) and $S(2,2) = (1,2)$. Pixel $(2,2)$ is highlighted with red cross-hatching, and its downstream region $U(2,2) = \{(2,2), (3,1), (3,2), (4,1), (4,2), (4,3)\}$ is shown with a bold red border. We have $f_{2,2} = 0.87 + 0.85 + 0.83 + 0.82 = 3.37$, so that the flow at $(2,2)$ corresponds to the habitat repopulation that flowed through $(2,2)$.

layer $r^{(0)}$, consisting of the repopulation seeds, be the input. The landscape matrix permeability p is a model parameter. The PyTorch model computes n multiplication by a random layer followed by max-pool operations, and finally a multiplication by p , corresponding to the updates (3). The output is the connectivity layer $c = h s^{(n)}$. As p is a model parameter in PyTorch, we can let $FCH = a \sum_{ij} c_{ij}$ and compute $\partial FCH / \partial p_{ij}$ via back-propagation; the flow layer is obtained via a multiplication by p_{ij} , according to (4).

When the input layers are sufficiently small, they can be fed at once to the GPU; otherwise, they can be fed in separate tiles.

When interested in both connectivity and flow layers, we divided our 3716×5377 study area in 2048×2048 tiles, each with a border of 256 pixels around it (some tiles were smaller, at the extremes of the area). We fed these tiles, of overall size $(2048 + 2 \cdot 256) \times (2048 + 2 \cdot 256)$ pixels, to an A100 NVidia GPU. The GPU processed these tiles, and we used the output results only from the core part, excluding the border. The purpose of the border was to provide context for the processing of the core: in particular, seeds in the border region can cause repopulation in the core. Thus, it is necessary for the border region to be larger than the dispersal distance: this was easily true in our setting, as the longest dispersal distance we considered was of $2 \cdot 20 = 40$ pixels.

When only the connectivity layer is desired, the PyTorch implementation can be run in *forward*-mode, without storing the information required for back-propagation. This speeds the computation and reduces the memory requirements. In this case, our study area of 3716×5377 pixels could be sent at once to an NVidia A100 GPU.

Connectivity and flow layers are obtained as the average of many simulations. As the values produced by each simulation are statistically independent, the standard deviation of the result decreases with the square root of the number of simulations.

While connectivity is bounded to the $[0, 1]$ interval, the flow through a pixel is not bounded (see Figure 5), and can assume large values if the pixel is helpful in repopulating a large territory. To accommodate for the value range, the tool outputs $20 \log_{10}(1 + f_{ij})$ rather than f_{ij} at each pixel ij , and such logarithmic scale has also been used in our figures.

Limitations.

A limitation of the current algorithm consists in the use of a neighborhood $N_g(ij)$ that is square in shape: $kl \in N_g(ij)$ if $|k - i| \leq g$ and $|l - j| < g$, where g is the gap-crossing distance in pixels. We chose to use square neighborhoods for the sake of efficiency, as they matched well the max-pool operators available in PyTorch. For larger values of the gap-distance g , a round neighborhood may be preferable. We experimented with variants, but for the small values of g considered in this paper more complex algorithms did not lead to superior results, at least as measured by the fit with citizen-scientist species observations.

5.2 eBird Data

The preparation of layers and the validation we present relied on bird records from the citizen-science database eBird [10] downloaded January 2023 [62]. eBird data consists of a collection of *checklists*. Each checklist is compiled by a birder visiting a location, and counting how many birds of each species they were able to identify. Among other data, a checklist contains:

- a location expressed as latitude/longitude pair,
- a date,
- the type of checklist, which can be stationary, traveling, or incidental (the latter used for observations when the main goal was not birding),
- for traveling checklists, a length of travel,
- whether the checklist was a complete or partial list of observed species,
- and finally, a list of observed species, each with the count of individuals observed.

In this study, we filtered the checklists, keeping only those that:

- were located in California (our area of study),
- took place in 2012-2018, so as to match the time-frame of our landscape matrix layer, which was derived from satellite observations in 2015,
- were traveling, as incidental checklists are unreliable, and we did not want to consider stationary checklists, which are a minority and might behave differently from traveling ones,
- were of length not exceeding 2 km,
- were complete.

The choice of limiting the analysis to checklists not exceeding 2 km was done to balance data availability with the location accuracy of sightings. The shorter the distance

traveled, the more accurately we know the locations of the birds sighted, but the fewer the checklists we select.

Once filtered, we clustered the checklists into a discrete set of *locations* according to their geographical coordinates. For each checklist, we truncated latitude and longitude to two decimal digits, forming a string: each location would correspond to one such string. For instance, an observation occurring at coordinates (37.428992, -122.015710) would be represented by the string “37.42;-122.01”, and all locations with this string would be associated with the same location. Due to the rounding, a location corresponds to an area of about 1 km², which was an approximate match for our checklist length upper-bound of 2 km (during a checklist, observers do not always travel in straight lines).

5.3 Generation of the Input Layers

EcoScape requires as input the habitat and permeability raster layers in GeoTIFF format, as well as gap-crossing and dispersal distances. EcoScape can work with any permeability and habitat layers; the only requirement being that they are specified in the same coordinate reference system (CRS). In particular, EcoScape can use as habitat boolean raster layers produced by any species distribution model (SDM), and can use as permeability raster layers any continuous landscape suitability map.

To develop the results of this paper, and to facilitate similar studies on other regions and bird species, we created a Python package, [ecoscape-layers](#) [64], that automates the generation of these layers using freely available data products from eBird and IUCN. In our code repository for reproduction, we provide Python notebooks that produce the inputs used for this paper [63]. Further, as part of the documentation for the [ecoscape-layers](#) package, we provide sample code for generating input layers for other species and regions.

To run the package, users need API tokens for eBird <https://science.ebird.org/en/status-and-trends/download-data> and IUCN <https://apiv3.iucnredlist.org/api/v3/token>. Users also need to specify the desired raster projection depending on their study area; the bounds used to define the study area as coordinates; and the six-letter eBird codes of the species of interest. These can be obtained as documented in the software packages.

Landscape matrix layer.

We prepare the landscape matrix layer using the Jung et al. Level 2 habitat map from 2015 [40]. This map classifies global land covers into the habitat types as described by the IUCN Red List Habitats Classification Scheme [41].

Habitat layer.

The EcoScape habitat layer is similar to an Area of Habitat map [32] which depicts the suitable area within a species’ range based on habitat and elevation preferences. In this study we do not consider elevation and only refined the range map by suitable habitat due to inconsistencies in elevation limits.

To create the habitat layer, the `ecoscape-layers` package uses the landscape matrix layer produced above in conjunction with the range maps that can be downloaded from eBird at <https://science.ebird.org/en/status-and-trends/range-maps>. For methodological details on how eBird range maps are computed, see Fink and collaborators [39].

We refine the eBird ranges according to landcover class, using the habitat preference categories downloaded from the IUCN Red List [33], in combination with the landscape matrix layers produced above. Users can decide what classifies as a habitat; the options available are: selecting all forest habitat types (for forest specialist birds), selecting only habitat types classified as “major importance” by the IUCN, or selecting habitat types classified as “major importance” and/or “suitable” by the IUCN. Based on the user-defined refining method, `ecoscape-layers` will create a binary habitat layer of pixels that fall within the species’ range map and are classified as habitat (value 1), and those in the range map but not classified as habitat (value 0).

Landscape matrix permeability.

The IUCN Red List habitat preferences provided for a bird species include the habitat code, major importance, suitability, and seasonality for each habitat type listed for a species. For each species, the `ecoscape-layers` package builds a table that associates each landcover class with its permeability. Landcover classes deemed “suitable” and of “major importance” as habitats are assigned a permeability of 1 (no resistance), landcover classes that are “suitable” (but not “major importance”) get a permeability of 0.9, and all other landcovers receive a permeability of 0.

We then fine-tuned these values according to eBird sightings. This step was necessary, as IUCN occasionally did not list as suitable some landcover classes where in fact the species are often recorded, and that thus, are clearly permeable to the species (even when they may not be suitable as breeding habitat).

To do so, we considered all checklists as selected in Section 5.2; note that the checklists have year-round occurrence, and are not restricted to the breeding season. To measure where the sightings occurred, we created a new *area of sightings* geographical raster layer, with the same spatial resolution as the landscape matrix layer, with pixels initially set to value 0. We then considered all checklists; if a checklist reported a sighting of the species, we set to 1 all pixels of the *area of sightings* layer within 1 km of the checklist location. We then counted, for each landcover class i , the number K_i of pixels in the *area of sightings* layer with value 1, where the corresponding landcover pixel has value i . This gave us a measure of the area of landcover i where sightings occurred, and avoids over-counting sightings that occur very close to each other.

Let I_H be the landcover classes that are part of the habitat, and let $K = \max_{i \in I_H} K_i$ be the maximum sighting area for any habitat landcover class. For a landcover class i , let p_i^{IUCN} be the permeability determined via IUCN. In our study, we set the permeability of i via $p = \min(1, \max(p_i^{\text{IUCN}}, K_i/K))$, thus giving high permeability to the landcovers where a species is often sighted. Supplemental Table 1 in the supplementary materials reports the permeability of all landcover classes with permeability over 0.01; landcover classes with lower permeability do not meaningfully influence the results. The reproduction datasets contain full permeability results.

Layers for Acorn Woodpecker and Steller’s Jay.

For the study, we selected two forest/woodland associated bird species: Acorn Woodpecker (*Melanerpes formicivorus*), and Steller’s Jay (*Cyanocitta stelleri*). These species were chosen due to their resident status, affinity to wooded habitats, abundant eBird sighting records, and high detectability.

Our study region encompasses California, plus a 200 km buffer zone around it to provide surrounding information for all California locations. We use a spatial resolution of 300m pixels, reasoning that the gap crossing distance of all these birds is likely to be greater than 300m. The layers used for the experiments were of 3716×5377 pixels; they are provided as part of the reproducibility dataset [63].

For Acorn Woodpecker, we added to the list of suitable landcover classes the landcover type 308, *Mediterranean type shrubby vegetation*. In California, this landcover class typically corresponds to sparse oak woodland, which is prime habitat for Acorn Woodpeckers.

5.4 Gap-crossing and dispersal distance

To run EcoScape, users must provide values of species movement as gap-crossing distance and dispersal distance, both expressed in pixels. As the name implies, the gap-crossing distance is the longest amount of unfavorable landcover that a bird is willing to fly over during dispersal. A bird with gap-crossing distance g can fly over up to $g - 1$ pixels of the landscape matrix regardless of its permeability. The dispersal distance d is the distance a species most commonly travels when dispersing. We do not specify d directly, but rather, we specify the number n of gap-crossings, so that $d = ng$.

If users have data on gap-crossing and/or dispersal distance for their study species, such as from bird banding or movement studies, this is the ideal data to input into EcoScape. However, these values are hard to encounter in the literature and only exist for very few well-studied birds [65]. The recent release of the global AVONET database [34], and proofs-of-concept for the Hand-Wing Index and its relationship to gap-crossing abilities, such as those by Claramunt and collaborators in Amazonian forest fragments [66], provide a framework for using the HWI as a proxy for bird dispersal [42, 67]. If dispersal distance for the species of interest is not known, we provide a way to calculate it from the HWI (provided freely in the global database AVONET [34]). Using a dataset of known bird HWI and average dispersal distance, we trained a linear regression model to predict dispersal distance. The trained linear model predicts the dispersal distance d via the formula:

$$d = 0.734 \cdot HWI - 8.777 \tag{5}$$

The gap-crossing distance has also shown to have a relationship to the HWI, but this has only been explored for road crossing between forest fragments [66]. Given this uncertainty, we use a gap-crossing of 2 pixels (600m) for all species in this study, except in the sensitivity analysis reported below. We report the gap-crossing and dispersal distance used in our example species in Table 2 and note that the dispersal distance was calculated based on the HWI. As our knowledge of bird dispersal expands with

Species	Gap Distance	N. gaps	Dispersal
Acorn Woodpecker	2px (600m)	20	12 km
Steller's Jay	2px (600m)	3	1.8 km

Table 2: Parameters for the EcoScape runs that correspond to the dispersal distance estimated based on the Hand-Wing Index (HWI).

increasingly smaller and cheaper GPS transmitters [60], EcoScape will be ready to receive parameters on both gap-crossing and dispersal distance.

5.5 Validation with Citizen-Science Data

We validated the EcoScape connectivity layer for our example species by measuring the correlation between the connectivity at a pixel, and the relative bird abundance at that pixel, measured as the frequency of observation for the species. For validation, we filtered the checklists as described in Section 5.2, and then we further considered only the checklists that took place in the months April-May-June, as this is breeding season for the species we considered. The filtering yielded $N=185,996$ checklists, and $N=18,336$ locations in California with at least some checklist.

5.5.1 Ecoscape

We validated the connectivity layers for both EcoScape and existing algorithm Omniscape [26] using eBird observations, and detail the methods below.

Given a habitat layer h with values in $\{0, 1\}$ and a connectivity layer c computed via EcoScape or Omniscape, we compute the correlation between c and eBird sightings as follows. We consider the 18,336 locations with checklists in California. For each location l , and for each of the two species considered in the study, we compute the average number b_l of individuals per species seen in a checklist at l .

Since connectivity is computed in the habitat only, we then filtered the locations, keeping only those that occurred in the habitat of each species. Precisely, for each location l , let ij be the pixel coordinates of the center of l : if any of the 3×3 pixels around ij are in the habitat, we consider l to also be in the habitat. This yields 6,818 locations for the Acorn Woodpecker, and 5,806 for the Steller's Jay.

Next, we sampled the values of the habitat and connectivity layers at each location. We identified the pixel coordinates ij corresponding to the location center, and we computed c_l (the connectivity of the location) as the maximum connectivity in the habitat, in the 3×3 pixels (thus, $900\text{m} \times 900\text{m}$) around ij . Precisely, remembering that the habitat has values in $\{0, 1\}$:

$$c_l = \max_{\substack{i-1 \leq k \leq i+1 \\ j-1 \leq m \leq j+1}} h_{km} c_{km} .$$

Like most citizen-science data, eBird observations are affected by various types of noise [68, 69]. Bird sightings are recorded by volunteers with variable skills [70], under

variable conditions (length of outing, time of day, and so on), and in locations that have different characteristics. Thus, we do not directly compare the connectivity with the eBird-reported sightings at each location. Rather, we de-noise the data by binning locations according to their connectivity. A detailed statistical justification for this is reported in Section A.2 in the supplementary material.

We divide the habitat locations into 10 bins $[0, 0.1), [0.1, 0.2), \dots, [0.9, 1)$, according to their connectivity. Let the bins be $B_0, B_{0.1}, \dots, B_{0.9}$, labeled according to their lower bound. A location l with $c_l = 0.12$, l will be put in $B_{0.1}$. Let $|B|$ be the number of locations that are in bin B . For each bin B , we compute the average number of individuals per species per checklist at locations in the bin: $b_B = E[b_l \mid l \in B]$. For $x = 0, 0.1, 0.2, \dots, 0.9$, we measure the correlation between x and b_{B_x} , that is, between the bin’s connectivity x , and the average number b_{B_x} of bird sightings per checklist in the bin. Specifically, we compute a linear fit between x and b_{B_x} , where point (x, b_{B_x}) has weight $|B_x|$ equal to the number of locations in B_x , and we report the *coefficient of determination* R^2 of such linear fit. This provides a measure of the predictive power of connectivity with regards to bird observations. The average sightings per checklist, and the number of locations, that fall in each connectivity interval are reported in Supplemental Table 3 in the supplementary material for the two species in the study. The coefficient of determination R^2 is computed from the data in that table.

We performed a sensitivity analysis, reported in Section A.4 of the supplementary material, to examine how the results depend on the choice of gap-crossing and dispersal distances. The analysis shows that for Acorn Woodpecker and Steller’s Jay, R^2 is high for dispersal distances up to those computed via the hand-wing index (see Section 5.4).

5.5.2 Omniscap

Table 3 provides the details of the Omniscap runs that have been used in our comparison with EcoScap. The block size reported in the table corresponds to the granularity with which Omniscap considers sources for the current flow. A block size of 5 for the Acorn Woodpecker implies that only the central pixel in each 5×5 tile is used as a source. Omniscap’s running time is roughly proportional to the number of sources, and hence, to the inverse of the square of the block size. A block size of 5 is perhaps a bit coarse, and favors Omniscap in the timing comparison. In EcoScap, a seed density of 4 corresponds for the Acorn Woodpecker to 1 pixel every $40^2 = 1600$ for each of the 400 simulations; hence, every pixel is sampled with roughly 1/4 probability; this compares with a 1/25 sampling probability in Omniscap. Due to these considerations, and more in general, due to the large number of parameters involved in these simulations, the comparison we present between Omniscap and EcoScap has value only as it pertains to selected parameter choices.

For Omniscap, we used Julia 1.5.5 and Omniscap 0.5.8, run on Intel i7 2.3GHz processor. For EcoScap, we relied on an A100 GPU on Google Colab. To obtain the conductivity values Omniscap needs, we have taken our permeability layer p , with values in $[0, 1]$, and rescaled it to the interval $[1, 11]$ via $1 + 10p$. Table 1 gives a detailed comparison of the running time of EcoScap and Omniscap, in runs with parameters according to Tables 2 and 3.

Species	Radius	Blocksize
Acorn Woodpecker	40px (12 km)	5
Steller's Jay	6px (1.8 km)	1

Table 3: Parameters used in the Omniscape runs that we used in the comparison with EcoScape.

Once the current values are obtained, we divide the current values in 10 intervals, as we do for EcoScape. The intervals are equal-sized, and we choose their size so that the 10th interval contains a fraction 0.1 of the habitat locations with eBird checklists (thus, the 10th interval starts at the 90% quantile). Proceeding in analogy of what done for connectivity, we compute the average sightings per checklist in each of the 10 current intervals. Supplemental Table 4 in the supplementary materials reports the average sightings per checklist, and the number of locations, that fall in each current interval. From this data, we can compute the coefficient of determination R^2 as the coefficient of determination of the average sightings vs. the current in the interval, using the number of locations as weights. The results, reported in Table 1, indicate that the current computed by Omniscape correlates well with eBird bird frequency observations.

5.5.3 Patch Size

To study the correlation between habitat patch size and species sightings reported in eBird, we computed the size of all habitat patches, considering habitat pixels connected (and thus belonging to the same patch) if they shared an edge or a corner. In the study conditions, patch sizes range from 1 pixel (0.09 km²) to over 10,000 km², and a direct correlation between patch size and species observations would not have yielded useful results. Rather, we studied the correlation between log-size, which is more uniformly distributed, and sightings. Thus, we produced a connectivity-as-patch-size raster layer c , where c_{ij} is equal to the log patch size to which pixel ij belongs. In this way, the same setup we used for studying the correlation of EcoScape and Omniscape output with sightings, enabled us to study the correlation of log patch size and sightings.

To group locations according to intervals of patch size, we formed equal-sized intervals in log-patch size. As left endpoint of the smallest interval, we took the patch size corresponding to the area of 1 pixel, or 0.09 km². We then considered equal-spaced intervals in log-patch size, assigning a patch size of size s to the interval $[\log s]$, where \log is the natural logarithm. This yielded 14 total intervals, and 12 non-empty intervals, for the patch sizes in the area of study for both species, which is comparable to the 10 intervals used for connectivity and current. We compute R^2 as the coefficient of determination between average sightings in the interval, and log-patch size corresponding to the interval, using as weight the number of locations that fall in patches with that size interval; this is the analogous of what we do for connectivity and current. The detailed results are reported in the supplementary materials, Section A.6.

We note that while the algorithm we used to compute connected patches is reasonably efficient, it is by no means state of the art. The problem of computing connected

image regions is important in image processing, and much work has been devoted to speeding up the algorithm, including via low-level GPU programming; for a review of approaches see, e.g., [71, 72].

References

- [1] Kareiva, P. & Wennergren, U. Connecting landscape patterns to ecosystem and population processes. *Nature* **373**, 299–302 (1995). Publisher: Nature Publishing Group UK London.
- [2] Ricketts, T. H. The matrix matters: effective isolation in fragmented landscapes. *The American Naturalist* **158**, 87–99 (2001). Publisher: The University of Chicago Press.
- [3] Crooks, K. R. & Sanjayan, M. *Connectivity conservation* Vol. 14 (Cambridge University Press, 2006).
- [4] Rudnick, D. A. *et al.* The role of landscape connectivity in planning and implementing conservation and restoration priorities. *Issues in ecology* 1–23 (2012). Publisher: Ecological Society of America.
- [5] on Biological Diversity, C. COP 15 - Report of the Conference of the Parties to the Convention on Biological Diversity on its fifteenth meeting (Part I). Tech. Rep. (2022).
- [6] McRae, B. H., Dickson, B. G., Keitt, T. H. & Shah, V. B. Using Circuit Theory to Model Connectivity in Ecology, Evolution, and Conservation. *Ecology* **89**, 2712–2724 (2008). URL <https://onlinelibrary.wiley.com/doi/abs/10.1890/07-1861.1>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1890/07-1861.1>.
- [7] Urban, D. & Keitt, T. Landscape connectivity: a graph-theoretic perspective. *Ecology* **82**, 1205–1218 (2001). Publisher: Wiley Online Library.
- [8] Saura, S. & Torné, J. Conefor Sensinode 2.2: a software package for quantifying the importance of habitat patches for landscape connectivity. *Environmental modelling & software* **24**, 135–139 (2009). Publisher: Elsevier.
- [9] Ovaskainen, O. *et al.* An empirical test of a diffusion model: predicting clouded apollo movements in a novel environment. *The American Naturalist* **171**, 610–619 (2008). Publisher: The University of Chicago Press.
- [10] Sullivan, B. L. *et al.* eBird: A citizen-based bird observation network in the biological sciences. *Biological conservation* **142**, 2282–2292 (2009). Tex.ids=sullivanEBirdCitizenbasedBird2009a publisher: Elsevier.
- [11] Taylor, P. D., Fahrig, L., Henein, K. & Merriam, G. Connectivity is a vital element of landscape structure. *Oikos* 571–573 (1993). ISBN: 0030-1299 Publisher: JSTOR.
- [12] Beier, P. & Noss, R. F. Do habitat corridors provide connectivity? *Conservation biology* **12**, 1241–1252 (1998). ISBN: 0888-8892 Publisher: Wiley Online Library.

- [13] Chase, J. M., Blowes, S. A., Knight, T. M., Gerstner, K. & May, F. Ecosystem decay exacerbates biodiversity loss with habitat loss. *Nature* **584**, 238–243 (2020). URL <https://www.nature.com/articles/s41586-020-2531-2>. Number: 7820 Publisher: Nature Publishing Group.
- [14] McRae, B. H. & Beier, P. Circuit theory predicts gene flow in plant and animal populations. *Proceedings of the National Academy of Sciences* **104**, 19885–19890 (2007). ISBN: 0027-8424 Publisher: National Acad Sciences.
- [15] Heller, N. E. & Zavaleta, E. S. Biodiversity management in the face of climate change: A review of 22 years of recommendations. *Biological Conservation* **142**, 14–32 (2009). URL <https://linkinghub.elsevier.com/retrieve/pii/S000632070800387X>.
- [16] McGuire, J. L., Lawler, J. J., McRae, B. H., Nuñez, T. A. & Theobald, D. M. Achieving climate connectivity in a fragmented landscape. *Proceedings of the National Academy of Sciences* **113**, 7195–7200 (2016). ISBN: 0027-8424 Publisher: National Acad Sciences.
- [17] Littlefield, C. E., Krosby, M., Michalak, J. L. & Lawler, J. J. Connectivity for species on the move: supporting climate-driven range shifts. *Frontiers in Ecology and the Environment* **17**, 270–278 (2019). ISBN: 1540-9295 Publisher: Wiley Online Library.
- [18] UN, I. Convention on biological diversity. Tech. Rep. (1992). URL <https://www.ipcc.ch/site/assets/uploads/2001/09/doc4f.pdf>.
- [19] Keeley, A. T., Beier, P. & Jenness, J. S. Connectivity metrics for conservation planning and monitoring. *Biological Conservation* **255**, 109008 (2021). ISBN: 0006-3207 Publisher: Elsevier.
- [20] Ernest, H. B. *et al.* Genetic structure of mountain lion (*Puma concolor*) populations in California. *Conservation Genetics* **4**, 353–366 (2003). ISBN: 1566-0621 Publisher: Springer.
- [21] Correa Ayram, C. A., Mendoza, M. E., Etter, A. & Salicrup, D. R. P. Habitat connectivity in biodiversity conservation: A review of recent studies and applications. *Progress in Physical Geography* **40**, 7–37 (2016). Publisher: SAGE Publications Sage UK: London, England.
- [22] Brennan, A. *et al.* Functional connectivity of the world’s protected areas. *Science* **376**, 1101–1104 (2022). ISBN: 0036-8075 Publisher: American Association for the Advancement of Science.
- [23] McRae, B. H. Isolation by resistance. *Evolution* **60**, 1551–1561 (2006). Publisher: Wiley Online Library.

- [24] McRae, B. *et al.* Conserving Nature’s Stage: Mapping Omnidirectional Connectivity for Resilient Terrestrial Landscapes in the Pacific Northwest. Tech. Rep. (2016). URL <http://rgdoi.net/10.13140/RG.2.1.4158.6166>. Publisher: The Nature Conservancy.
- [25] Hall, K. R. *et al.* Circuitscape in Julia: Empowering Dynamic Approaches to Connectivity Assessment. *Land* **10**, 301 (2021). URL <https://www.mdpi.com/2073-445X/10/3/301>.
- [26] Landau, V., Shah, V., Anantharaman, R. & Hall, K. Omniscape.jl: Software to compute omnidirectional landscape connectivity. *Journal of Open Source Software* **6**, 2829 (2021). URL <https://joss.theoj.org/papers/10.21105/joss.02829>.
- [27] Farley, S. S., Dawson, A., Goring, S. J. & Williams, J. W. Situating Ecology as a Big-Data Science: Current Advances, Challenges, and Solutions. *BioScience* **68**, 563–576 (2018). URL <https://doi.org/10.1093/biosci/biy068>.
- [28] Van Moorter, B. *et al.* Accelerating advances in landscape connectivity modelling with the ConScape library. *Methods in Ecology and Evolution* **14**, 133–145 (2022). URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.13850>. Tex.ids= vanmoorterAcceleratingAdvancesLandscape2023 .eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/2041-210X.13850>.
- [29] Hanski, I. Metapopulation dynamics. *Nature* **396**, 41–49 (1998). URL <http://www.nature.com/articles/23876>. Bandiera_abtest: a Cg-type: Nature Research Journals Number: 6706 Primary_atype: Reviews Publisher: Nature Publishing Group.
- [30] Hanski, I. & Ovaskainen, O. The metapopulation capacity of a fragmented landscape. *Nature* **404**, 755–758 (2000). Publisher: Nature Publishing Group UK London.
- [31] Hanski, I. & Ovaskainen, O. Extinction debt at extinction threshold. *Conservation biology* **16**, 666–673 (2002). Publisher: Wiley Online Library.
- [32] Brooks, T. M. *et al.* Measuring terrestrial area of habitat (AOH) and its utility for the IUCN Red List. *Trends in ecology & evolution* **34**, 977–986 (2019). ISBN: 0169-5347 Publisher: Elsevier.
- [33] Hilton-Taylor, C. & Brackett, D. *2000 IUCN red list of threatened species* (Gland, 2000). URL <https://policycommons.net/artifacts/1372154/2000-iucn-red-list-of-threatened-species/1986326/>. Publisher: IUCN: International Union for Conservation of Nature.
- [34] Tobias, J. A. *et al.* AVONET: morphological, ecological and geographical data for all birds. *Ecology Letters* **25**, 581–597 (2022). ISBN: 1461-023X Publisher: Wiley Online Library.

- [35] Panzacchi, M. *et al.* Predicting the continuum between corridors and barriers to animal movements using Step Selection Functions and Randomized Shortest Paths. *Journal of Animal Ecology* **85**, 32–42 (2016). ISBN: 0021-8790 Publisher: Wiley Online Library.
- [36] Paszke, A. *et al.* Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* **32** (2019). Tex.ids=paszkePytorchImperativeStyle2019a.
- [37] de Alfaro, L. *et al.* EcoScape Connectivity: Python Package (2023). URL <https://pypi.org/project/ecoscape-connectivity>.
- [38] Saura, S., Estreguil, C., Mouton, C. & Rodríguez-Freire, M. Network analysis to assess landscape connectivity trends: application to European forests (1990–2000). *Ecological Indicators* **11**, 407–416 (2011). Publisher: Elsevier.
- [39] Fink, D. *et al.* eBird Status and Trends (2020). URL <https://ebird.org/science/status-and-trends/data-access>. Institution: Cornell Lab of Ornithology.
- [40] Jung, M. *et al.* A global map of terrestrial habitat types (2020). URL <https://zenodo.org/record/4058819>.
- [41] IUCN. Habitats Classification Scheme (Version 3.1) (2023). URL <https://www.iucnredlist.org/resources/habitat-classification-scheme>.
- [42] Sheard, C. *et al.* Ecological drivers of global gradients in avian dispersal inferred from wing morphology. *Nature communications* **11**, 2463 (2020). ISBN: 2041-1723 Publisher: Nature Publishing Group UK London.
- [43] Bisong, E. & Bisong, E. Google colabouratory. *Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners* 59–64 (2019). Tex.ids=bisongGoogleColabouratory2019a publisher: Springer.
- [44] Dickson, B. G. *et al.* Circuit-theory applications to connectivity science and conservation. *Conservation Biology* **33**, 239–249 (2019). URL <https://onlinelibrary.wiley.com/doi/10.1111/cobi.13230>.
- [45] Ovaskainen, O. & Cornell, S. J. Biased movement at a boundary and conditional occupancy times for diffusion processes. *Journal of applied probability* **40**, 557–580 (2003). URL <https://www.cambridge.org/core/journals/journal-of-applied-probability/article/biased-movement-at-a-boundary-and-conditional-occupancy-times-for-diffusion-processes/395C5239D084ADB9E6657295CCF05FBC>. Publisher: Cambridge University Press.
- [46] Leonard, P. B. *et al.* gflow: software for modelling circuit theory-based connectivity at any scale. *Methods in Ecology and Evolution* **8**, 519–526 (2017).

URL <https://onlinelibrary.wiley.com/doi/abs/10.1111/2041-210X.12689>. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1111/2041-210X.12689>.

- [47] McRae, B. H. & Kavanagh, D. M. Linkage mapper connectivity analysis software. *The Nature Conservancy, Seattle WA* (2011).
- [48] Nuñez, T. A. *et al.* Connectivity planning to address climate change. *Conservation Biology* **27**, 407–416 (2013). Publisher: Wiley Online Library.
- [49] Koen, E. L., Bowman, J., Sadowski, C. & Walpole, A. A. Landscape connectivity for wildlife: development and validation of multispecies linkage maps. *Methods in Ecology and Evolution* **5**, 626–633 (2014). ISBN: 2041-210X Publisher: Wiley Online Library.
- [50] DeMatteo, K. E. *et al.* A multispecies corridor in a fragmented landscape: Evaluating effectiveness and identifying high-priority target areas. *Plos one* **18**, e0283258 (2023). ISBN: 1932-6203 Publisher: Public Library of Science San Francisco, CA USA.
- [51] Dickson, B. G., Roemer, G. W., McRae, B. H. & Rundall, J. M. Models of regional habitat quality and connectivity for pumas (*Puma concolor*) in the southwestern United States. *PloS one* **8**, e81898 (2013). ISBN: 1932-6203 Publisher: Public Library of Science San Francisco, USA.
- [52] Rodríguez-Soto, C., Monroy-Vilchis, O. & Zarco-González, M. M. Corridors for jaguar (*Panthera onca*) in Mexico: conservation strategies. *Journal for Nature Conservation* **21**, 438–443 (2013). ISBN: 1617-1381 Publisher: Elsevier.
- [53] Yesuf, G. U., Brown, K. A., Walford, N. S., Rakotoarisoa, S. E. & Rufino, M. C. Predicting range shifts for critically endangered plants: Is habitat connectivity irrelevant or necessary? *Biological Conservation* **256**, 109033 (2021). ISBN: 0006-3207 Publisher: Elsevier.
- [54] Sutherland, G. D., Harestad, A. S., Price, K. & Lertzman, K. Scaling of Natal Dispersal Distances in Terrestrial Birds and Mammals. *Conservation Ecology* **4**, art16 (2000). URL <http://www.ecologyandsociety.org/vol4/iss1/art16/>.
- [55] Ocampo-Peñuela, N., Garcia-Ulloa, J., Kornecki, I., Philipson, C. D. & Ghazoul, J. Impacts of four decades of forest loss on vertebrate functional habitat on Borneo. *Frontiers in Forests and Global Change* **3**, 53 (2020). ISBN: 2624-893X Publisher: Frontiers Media SA.
- [56] Unnithan Kumar, S. & Cushman, S. A. Connectivity modelling in conservation science: a comparative evaluation. *Scientific Reports* **12**, 16680 (2022). ISBN: 2045-2322 Publisher: Nature Publishing Group UK London.

- [57] Pullinger, M. G. & Johnson, C. J. Maintaining or restoring connectivity of modified landscapes: evaluating the least-cost path model with multiple sources of ecological information. *Landscape Ecology* **25**, 1547–1560 (2010). ISBN: 0921-2973 Publisher: Springer.
- [58] Zeller, K. A. *et al.* Are all data types and connectivity models created equal? Validating common connectivity approaches with dispersal data. *Diversity and distributions* **24**, 868–879 (2018). ISBN: 1366-9516 Publisher: Wiley Online Library.
- [59] Mateo-Sánchez, M. C. *et al.* Estimating effective landscape distances and movement corridors: comparison of habitat and genetic data. *Ecosphere* **6**, 1–16 (2015). ISBN: 2150-8925 Publisher: Wiley Online Library.
- [60] Wikelski, M. *et al.* Going wild: what a global small-animal tracking system could do for experimental biologists. *Journal of Experimental Biology* **210**, 181–186 (2007). ISBN: 1477-9145 Publisher: Company of Biologists.
- [61] INaturalist.
- [62] eBird. eBird: An online database of bird distribution and abundance [web application] (2023). URL <https://www.ebird.org>. Accessed March 2023.
- [63] de Alfaro, L. *et al.* EcoScape Connectivity: Paper Reproduction (2023). URL <https://github.com/ecoscape-earth/ecoscape-connectivity-paper-reproduction>.
- [64] Tai, J. *et al.* EcoScape Habitat Generation: Python Package (2023). URL <https://pypi.org/project/ecoscape-connectivity>.
- [65] Fraser, K. C. *et al.* Tracking the conservation promise of movement ecology. *Frontiers in Ecology and Evolution* **6**, 150 (2018). ISBN: 2296-701X Publisher: Frontiers.
- [66] Claramunt, S., Hong, M. & Bravo, A. The effect of flight efficiency on gap-crossing ability in Amazonian forest birds. *Biotropica* **54**, 860–868 (2022). Publisher: Wiley Online Library.
- [67] Arango, A. *et al.* Hand-Wing Index as a surrogate for dispersal ability: the case of the Emberizoidea (Aves: Passeriformes) radiation. *Biological Journal of the Linnean Society* **137**, 137–144 (2022). Publisher: Oxford University Press UK.
- [68] Kosmala, M., Wiggins, A., Swanson, A. & Simmons, B. Assessing data quality in citizen science. *Frontiers in Ecology and the Environment* **14**, 551–560 (2016). Publisher: Wiley Online Library.
- [69] Callaghan, C. T., Rowley, J. J. L., Cornwell, W. K., Poore, A. G. B. & Major, R. E. Improving big citizen science data: Moving beyond haphazard sampling.

PLOS Biology **17**, e3000357 (2019). URL <https://journals.plos.org/plosbiology/article?id=10.1371/journal.pbio.3000357>.

- [70] Yu, J., Wong, W.-K. & Hutchinson, R. A. *Modeling Experts and Novices in Citizen Science Data for Species Distribution Modeling*, 1157–1162 (2010). ISSN: 2374-8486.
- [71] Hennequin, A., Lacassagne, L., Cabaret, L. & Meunier, Q. *A new Direct Connected Component Labeling and Analysis Algorithms for GPUs*, 76–81 (2018). URL <https://ieeexplore.ieee.org/abstract/document/8596835>.
- [72] Lemaitre, F., Hennequin, A. & Lacassagne, L. *How to speed Connected Component Labeling up with SIMD RLE algorithms*, WPMVP'20, 1–8 (Association for Computing Machinery, New York, NY, USA, 2020). URL <https://dl.acm.org/doi/10.1145/3380479.3380481>.

A Supplementary Material

A.1 Landcover Permeability

Supplemental Table 1 reports the permeability used in the study, for all IUCN landcover classes whose permeability was greater than 0.01. If the permeability is smaller than 0.01, no meaningful propagation can occur. The reproduction datasets contain full data on all landcover permeabilities, including those smaller than 0.01.

A.2 De-noising eBird observations

We provide here some statistical justification for the procedure we have employed to de-noise eBird checklist data. We recall that we first cluster eBird checklists into *locations* according to their geographical coordinates, using rounded-off values of latitude and longitude to attribute each checklist to a discrete location.

Once this is done, a simple way to de-noise the observations consists in restricting the consideration to locations with at least a certain number of checklists, say 10 or 20, and computing the average number of birds per checklist in each location. However, it turns out that relying on locations with many checklists for the validation is rather ineffective, for two reasons. First, there are relatively few locations with many checklists, and they tend not to occur in the middle of the forest habitats of our species, but at the borders, where roads and cities occur. Among our total of 18,336 locations in California, only 3,169 had more than 10 checklists.

Second, and more interestingly, it turns out that even from a statistical point of view, it is not useful to restrict consideration to locations with many checklists. We can think of the noise associated with observations as the sum of two kinds of noise: *checklist noise* and *location noise*.

Landcover Code	Landcover Name	Acorn Woodpecker	Steller's Jay
104	1.4. Forest – Temperate	1.000	1.000
105	1.5. Forest – Subtropical/tropical dry	1.000	0.900
201	2.1. Savanna - Dry	0.034	0.000
304	3.4. Shrubland –Temperate	0.359	0.324
305	3.5. Shrubland – Subtropical/tropical dry	0.099	0.052
308	3.8. Shrubland – Mediterranean-type shrubby veg...	1.000	0.212
401	4.1. Grassland – Tundra	0.002	0.015
404	4.4. Grassland – Temperate	0.900	0.040
405	4.5. Grassland – Subtropical/tropical dry	0.900	0.001
500	5. Wetlands (inland)	0.032	0.017
505	5.5. Wetlands (inland) – Permanent freshwater ...	0.045	0.032
900	9. Marine Neritic	0.015	0.006
1401	14.1 Arable Land	0.272	0.038
1402	14.2 Pastureland	1.000	0.221
1403	14.3 Plantations	0.909	0.259

Supplemental Table 1: IUCN landcover classes and their permeability for Acorn Woodpecker and Steller's Jay. All landcover classes with permeability greater than 0.01 for one or both species are listed.

Species	Checklist noise	Location noise
Acorn Woodpecker	1.52	2.42
Steller's Jay	1.29	1.77

Supplemental Table 2: Checklist and location noise for the species under consideration.

Checklist noise is the noise in checklists occurring at the same location; in other words, it is the standard deviation of the number of birds per checklist, in checklists taken at the same location. To measure the checklist noise, we compute at each location the standard deviation of the number of birds per checklist at that location. We then average the standard deviation at all locations with at least 10 checklists.

Location noise is the standard deviation in average number of birds per checklist measured on locations that are similar, for some measure of similarity. To measure the location noise, we considered only locations with at least 10 checklists. For these locations, we computed the average number of birds per checklist. We then considered two location similar if their connectivity value were similar, computing the connectivity with the parameters chosen for Figure 3. Hence, we divided the locations in 10 bins $[0, 0.1), [0.1, 0.2), \dots, [0.9, 1]$ of connectivity. For each bin, we computed the standard deviation of the locations' number of birds per checklist. We then computed the average over all bins.

It turns out that for our species, location noise is greater than checklist noise; the values are reported in Supplemental Table 2. In other words, there is stronger agreement between the number of birds seen in individual checklists compiled at the same location, than there is about the average number of birds per checklist compiled at equivalent locations from the point of view of connectivity. This most likely happens because we have focused on species that are very easy to detect: if Acorn Woodpeckers are breeding in a location, they are very likely to be detected and correctly recorded by most observers, lowering checklist noise.

Averaging the checklists taken at the same location is effective for dealing with checklist noise, but ineffective for location noise. As location noise is predominant, we are better off considering all locations: this far improves coverage (from 3,169 locations with at least 10 checklists, to 18,336 locations), and more importantly, improves coverage in the middle of the forested regions that constitute our habitats.

Supplemental Figure 1 provides another view into this phenomenon. The figure reports the standard deviation of the reported species sightings across locations that have the same connectivity, according to the location's connectivity, and according to the number of checklists available at the location. As we can see, more checklists per location does not translate into lower standard deviation; the variability of sightings across locations is not primarily a function of the number of checklists available. It is likely that other factors, from availability of nutrients, to characteristics of the location (sun exposure, humidity, amount of human traffic, and so forth) play a larger role.

Connectivity	Sightings	N	Connectivity	Sightings	N
0.000	0.504	272	0.000	0.476	168
0.100	0.607	175	0.100	0.342	166
0.200	0.878	245	0.200	0.809	139
0.300	0.839	294	0.300	0.615	151
0.400	0.956	377	0.400	0.841	179
0.500	1.088	714	0.500	0.805	241
0.600	1.246	993	0.600	0.965	365
0.700	1.620	1549	0.700	1.001	578
0.800	1.698	2151	0.800	1.128	1133
0.900	1.853	48	0.900	1.401	2686

(A) Acorn Woodpecker (B) Steller’s Jay

Supplemental Table 3: Data underlying Figure 3. *Connectivity* is the lower end-point of each connectivity interval considered (e.g., the interval $[0, 0.1)$ is listed as 0); *Sightings* is the average of the number of sightings per checklist at locations in that connectivity interval, and N is the number of locations averaged.

A.3 Sightings vs. Connectivity, Underlying Data

Supplemental Table 3 provides the data underlying Figure 3.

A.4 EcoScape Runs: Sensitivity Analysis

To analyze the sensitivity of EcoScape runs with respect to the values of gap and dispersal distance, we have run it for our species for gap distances $g = 1, 2, \dots, 6$, and for numbers of gap crossings n ranging from 2 to 40. Supplemental Figures 2A and 2B report the coefficient of determination R^2 between connectivity and sightings for each parameter combination for the species considered in the study.

The results present a threshold effect: the coefficient of determination is high until the dispersal distance $d = ng$ reaches a critical threshold, then decreases. We hypothesize that the distance threshold corresponds to the effective dispersal distance for the species, that is, the distance at which dispersal routinely contributes to population abundance. The threshold corresponds roughly to the dispersal distance obtained via the Hand-Wing Index approach (see Section 5.4).

A.5 Omniscape Runs: Current Flow vs. Sightings

In Supplemental Figure 3, we report the average number of sightings vs. cumulative current in habitat locations, as computed by Omniscape; the underlying data is provided in Supplemental Table 4. This figure is the equivalent of Figure 3 for EcoScape. To produce Supplemental Figure 3, we set the 0.9 value to the 90% percentile of the current range in the habitat for each species; this renormalization makes the graphs comparable with the ones for EcoScape.

Current, mA	Sightings	N	Current, mA	Sightings	N
0.000	0.413	204	0.000	0.156	33
11.449	0.879	554	2.247	0.936	6
22.899	0.962	619	4.495	0.643	118
34.348	0.929	614	6.742	0.570	318
45.798	1.395	713	8.989	0.848	500
57.247	1.656	682	11.237	0.917	598
68.697	1.469	796	13.484	1.014	751
80.146	1.472	895	15.731	1.219	948
91.595	1.772	1059	17.978	1.366	1953
103.045	1.620	682	20.226	1.435	581

(A) Acorn Woodpecker (B) Steller's Jay

Supplemental Table 4: Data underlying Supplemental Figure 3. *Current* is the lower endpoint of each cumulative current interval considered (we use equal-sized intervals); *Sightings* is the average of the number of sightings per checklist at locations in that connectivity interval, and *N* is the number of locations averaged.

A.6 Patch Size vs. Sightings

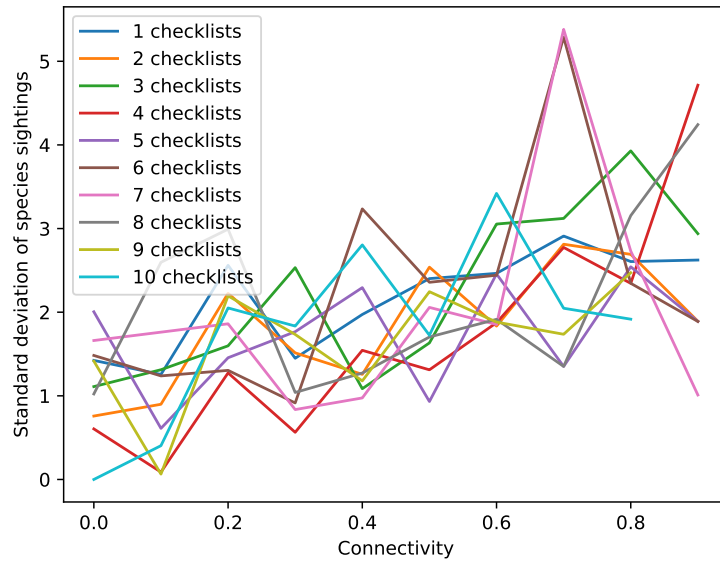
Supplemental Figure 4 depicts the relationship between average species sightings in eBird checklists, and size of habitat patches; the underlying data is provided in Supplemental Table 5. The figure is computed with the same methodology as Figure 3, using log patch size as connectivity measure.

Patch size, km ²	Sightings	N	M	Patch size, km ²	Sightings	N	M
0.090	0.658	821	13850	0.090	0.667	485	23322
0.245	1.101	465	3818	0.245	0.718	274	6535
0.665	1.096	286	1189	0.665	0.935	216	2447
1.808	1.295	193	370	1.808	1.014	168	926
4.914	1.804	136	157	4.914	0.860	173	404
13.357	1.184	155	64	13.357	1.337	153	161
36.309	1.867	120	40	36.309	1.088	206	74
98.697	1.527	194	23	98.697	1.216	149	26
268.286	0.834	646	9	268.286	0.762	249	9
729.278	1.792	1117	6	729.278	1.960	689	6
1982.382	1.901	750	4	1982.382	1.108	381	3
5388.673	—	0	0	5388.673	—	0	2
14647.931	1.447	1935	1	14647.931	—	0	0
39817.205	—	0	0	39817.205	1.144	2663	1

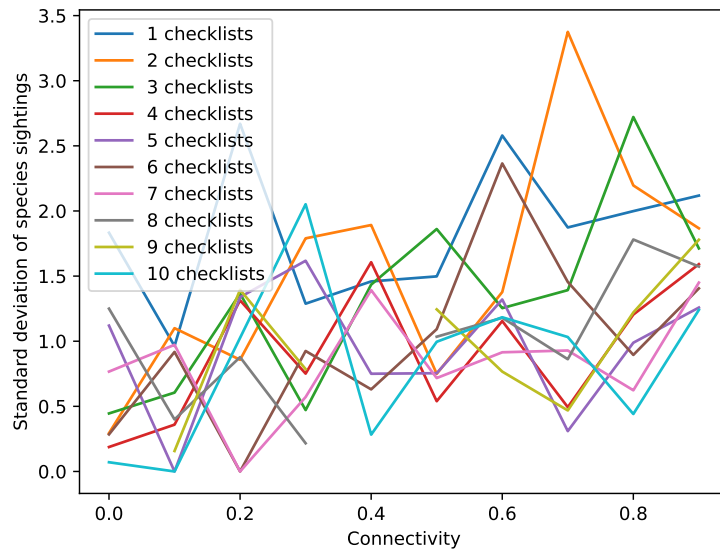
(A) Acorn Woodpecker

(B) Steller's Jay

Supplemental Table 5: Data underlying Supplemental Figure 4. *Patch size* is the lower endpoint of each patch size interval considered, in km² (we used intervals of size 1 in natural-logarithm scale, with the smallest interval starting at 1 pixel area or 0.09 km²); *Sightings* is the average of the number of sightings per checklist at locations in that connectivity interval, *N* is the number of locations averaged, and *M* is the number of patches having size in that interval. Note that for some intervals of patch size, there were no patches, and hence no locations nor sightings.

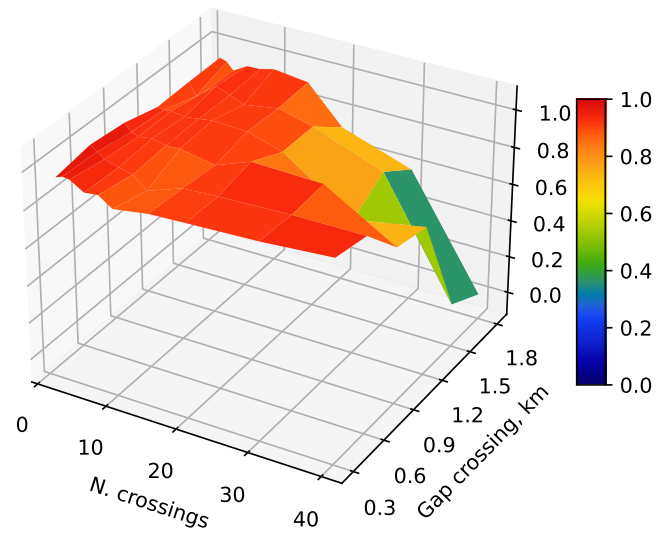


(A) Acorn Woodpecker

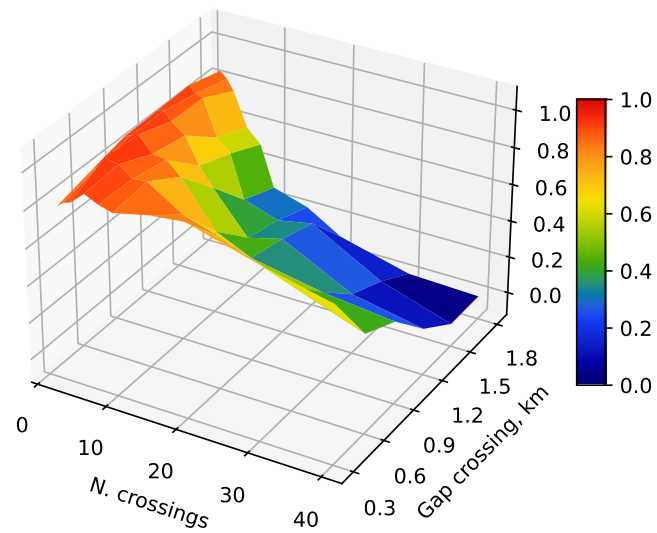


(B) Steller's Jay

Supplemental Figure 1: Standard deviation of bird sightings at each location (measured as average over checklists at that location) according to the connectivity of the habitat, and the number of checklists, at the location.

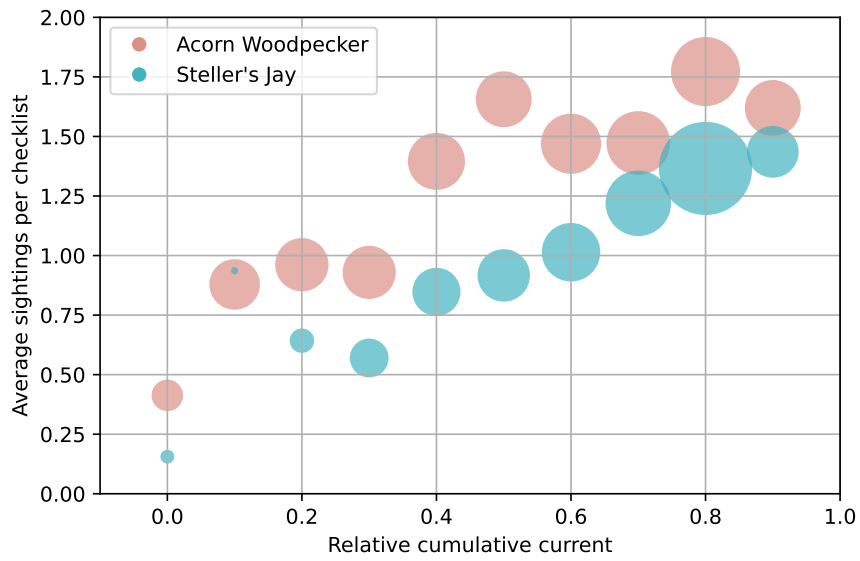


(A) Acorn Woodpecker

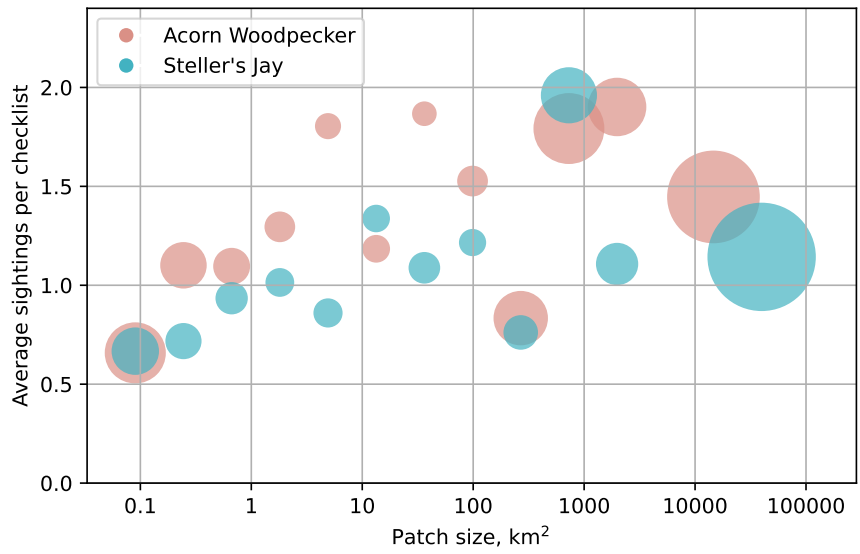


(B) Steller's Jay

Supplemental Figure 2: Coefficient of determination for connectivity as a predictor of average bird sightings per checklist, according to the gap crossing distance g (expressed in km; every pixel is 300m) and the number of gap crossings in a dispersal.



Supplemental Figure 3: Average sightings per checklist vs. relative cumulative current in habitat as computed by Omniscap.



Supplemental Figure 4: Average species sightings per eBird checklist, vs. size of habitat patches.