Image Sequence Geolocation with Human Travel Priors

Evangelos Kalogerakis[†], Olga Vesselova[†], James Hays[*], Alexei A. Efros[*], Aaron Hertzmann[†]

[†]University of Toronto, [*]Carnegie Mellon University

Abstract

This paper presents a method for estimating geographic location for sequences of time-stamped photographs. A prior distribution over travel describes the likelihood of traveling from one location to another during a given time interval. This distribution is based on a training database of 6 million photographs from Flickr.com. An image likelihood for each location is defined by matching a test photograph against the training database. Inferring location for images in a test sequence is then performed using the Forward-Backward algorithm, and the model can be adapted to individual users as well. Using temporal constraints allows our method to geolocate images without recognizable landmarks, and images with no geographic cues whatsoever. This method achieves a substantial performance improvement over the best-available baseline, and geolocates some users' images with near-perfect accuracy.

1. Introduction

This paper considers the problem of geolocating a sequence of time-stamped photographs taken by a single individual. We wish to determine where on Earth each picture was taken. A key observation of our work is that both the image data and the temporal data provide valuable cues to location. Consider, for example, a single image of a sea: it may be impossible — even for a human — to tell where the picture was taken, except that it must have been taken on one of the Earth's seas. However, suppose we know that the same photographer also took a picture containing the Acropolis, two hours later. Now we know much more about the first picture: it must be a sea within two hour's travel of Athens. As we add more images to the sequence, each of these images can help resolve the locations of the others. Considering multiple images together may even make it possible to geolocate them even when none of them contain recognizable landmarks.

This paper describes a method for geolocating image sequences using both image and temporal information. We introduce a model comprising two key components: a human travel prior, and an image likelihood. The human travel prior is estimated from a training set of georeferenced photos from Flickr.com (Figure 1). The image likelihood term is obtained by matching images to the training database. The model is a Hidden Markov Model variant, and inference of geographic tags for new sequences can be performed efficiently using the Forward-Backward algorithm. Performance can be improved by learning models specific to individual test users. Our method greatly improves geolocation results over single-image matching, achieving near-perfect accuracy for a few users, and significant improvements for most others. While we use a simple geographic cue based on image matching, our method could be combined with other cues such as geometric registration or high-level object recognition.

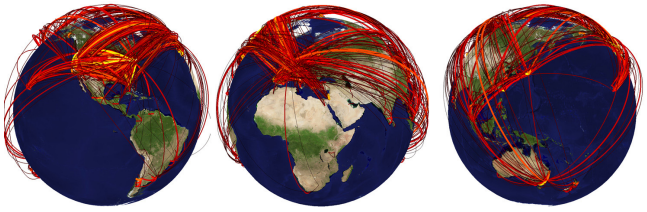

Figure 1. Common trips in the Flickr database, based on pairs of consecutive images separated by at least five days. Arcs are colored according to frequency.

Image sequence geolocation has many potential applications. First, automatic geolocation would provide users an easy way to tag their image collections and to visualize and share their own travel. Second, photo-sharing websites would be able to auto-tag all images for easy searching, sharing, and annotation, thereby allowing other users to find photos of a particular location. Third, new computer vision algorithms are increasingly making use of geotags when available, e.g., [4, 18]. Fourth, there is increasing scientific interest in models of human movement [2, 11], and online databases provide a rich source of movement data. One important application is to forecast the spread of future epidemics [6, 13]. Likewise, travel data is of immediate value to urban planners. For example, analysis of geotagged Flickr imagery provides valuable information about tourism [9], supplementing current data-gathering methods such as hotel and museum surveys [10]. Hence, effective sequence geotagging could be of interest to many disciplines, opening up new application areas for computer vision.

Despite the availability of existing geotagging tools, including GPS sensors and manual tagging, we find that only a minority of the billions of extant photos have geotags, and many manually-defined tags are vague or incorrect [15]. Though some cities are amply documented by geotagged photos, much of the world is not. Furthermore, some applications require separating users by categories such as nationality [9], which requires much larger datasets. It is conceivable that GPS-equipped cameras could eventually become prevalent, but there remain economic and technological barriers. Hence, there is a clear value to algorithms that can bootstrap from existing databases for geolocation.

2. Related Work

Geographic referencing of photographs is an emerging research topic in computer vision. Most existing methods focus on tagging single images in isolation, and vary in coverage from urban to regional to the entire Earth. Urban localization systems employ detailed databases of a particular city, and can potentially provide very detailed localization within that city. Zhang et al. [19] match image keypoints followed by geometric alignment, while Schindler et al. [17] match 2D patterns on 3D façades. At a regional level, Cristani et al. [8] learn models of image features for distinguishing outdoor images, which they apply to discriminate among regions of Southeastern France. At the global scale, Hays and Efros [12] compute location distributions by low-level image matching to a georeferenced database. Crandall et al. [7] identify landmarks based on image data, metadata, and other photos taken within a 15-minute window. Despite these exciting initial results, current methods are not capable of reliable geotagging of arbitrary images, and many images will be inherently ambiguous (e.g., a picture taken within a restaurant). Our work shows how single-image matching can be combined with sequence data to dramatically improve accuracy. In principle, our approach could be combined with any or all of the cues from the single-image methods above.

Our work is related to previous efforts to organize photo collections, for example, through geometric alignment [16, 18], by clustering photos into sequences of events [4], or both [3]. Our work is the first to exploit temporal constraints for geolocation. We do not explicitly cluster images into discrete events, instead using a more flexible travel model. Our work is also similar in spirit to the development of prior models for person tracking, but we focus on movement at much larger scales than previous work.

A major component of our work is modeling human travel distributions. Recent activity in this area has analyzed human travel distributions using random walk models. Brockmann et al. [2] obtained travel data from wheresgeorge.com, a website that tracks the movements of US dollar bills based on serial numbers entered by users. Movement statistics from this data were shown to follow a Lévy flight model, a stochastic process typified by many short-distance trips but also occasional long-distance trips. This data does not track individual users, only bills as they change hands. Airline traffic data provides another source of aggregate travel data [6, 13], but for only one mode of transport. González et al. [11] study individual human mobility using mobile phone traces, describing variation across individuals by employing a truncated Lévy flight model. However, their model ignores dependence on start and end location, which, as we show, is significant for worldwide travel. Additionally, for personal photographs, travel time and destination may be correlated (e.g., for individuals that photograph only when traveling); ours is the first study of the statistics of worldwide Flickr travel data.

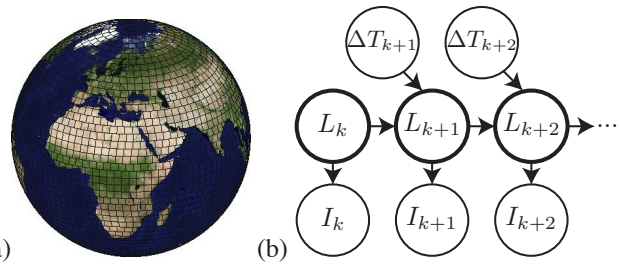

Figure 2. (a) Discretization of the world into 3186 location bins L_k. (b) We employ a modified Hidden Markov Model, in which travel between location bins L_k and L_{k+1} depends on the time interval ΔT_{k+1} between them. Each photograph I_k depends on the location L_k in which it was taken. Given input images $I_{1:N}$ and intervals $\Delta T_{2:N}$, our goal is to estimate the location bins L_k.

3. Overview

Our geolocation algorithm takes as input a sequence of N photographs $I_{1:N}$, and the time intervals $\Delta T_{2:N}$ between them. Specifically, ΔT_k is the elapsed time between image I_k and I_{k-1}. The goal is to estimate the locations $L_{1:N}$ for the images. We discretize the Earth into 3186 bins of roughly 400 km $\times$ 400 km (Figure 2(a)); each location L_k must correspond to one of these bins. Note that we do not assume that the original image timestamps are correct, only that the intervals between them are. Hence, the method does not require that the camera's clock is set correctly.

We assume a Markov prior model for travel conditioned on travel times (Section 4). This distribution describes the possible locations of photo k, given the location of photo $k-1$ and the time interval ΔT_k between them. Our image likelihood $p(I_k|L_k = i)$ is obtained by matching to a georeferenced database (Section 5). The complete graphical model is shown in Figure 2(b). This is a variant of a Hidden Markov Model in which different steps have different transition probabilities. Hence, inference can be performed using the Forward-Backward algorithm, yielding marginal distributions over location $P(L_k = i|I_{1:N}, \Delta T_{2:N})$, as de-

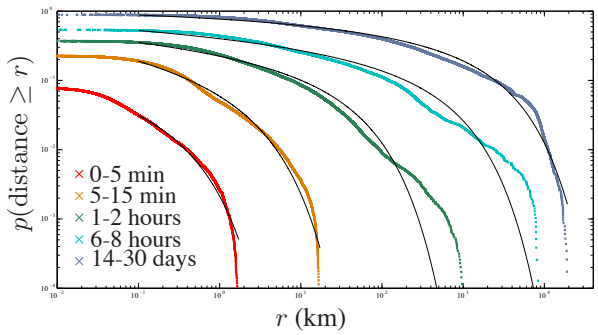

Figure 3. Log-log CDFs of travel distance between consecutive photographs in the Flickr database, illustrating the heavy-tailed nature of travel. Most pictures are separated by small distances, but a few by large distances. The distribution depends on the interval ΔT between pictures. Black curves are maximum likelihood fits of truncated Lévy flight models [5, 11].

scribed in Section 6. Furthmore, the model can be adapted to individual users, using an EM-like algorithm. Some additional model parameters are set by cross-validation.

4. Priors for Human Travel

We now consider the travel distribution that describes the possible location of photo k given the location of photo $k-1$ and the time interval ΔT_k between them. Our analysis is based on a dataset of 6 million georeferenced photographs from Flickr.com (details in Section 7).

Distance distributions. Recent studies of human travel have employed models based on the Lévy flight, a stochastic process from statistical physics [2, 11]. This model assumes a heavy-tailed distribution over trips: most trips cover short distances, but a few trips travel long distances. Figure 3 shows a plot of travel distances we obtained from Flickr data, along with truncated Lévy flight models fit by maximum likelihood [5, 11]. While the overall fit is reasonable, it is not perfect. The model does not capture falloff of the data at about 20,000 km (since it is impossible for the distance between two pictures to be greater than half the circumference of the Earth), nor does it capture travel distances below a minimum distance (in this case, 100 km). Most importantly, the above models are invariant to where travel begins or ends. In practice, we might expect that travel from Paris to New York is more likely than Paris to Iceland, or Paris to the North Sea, even though both destinations are closer than New York.

Spatially-variant model. To address these issues, we propose an empirical travel model that depends on start, destination, and travel time. Figures 1 and 4 visualize the empirical travel distributions for various starting points, showing behavior that depends significantly on both starting and ending location. This histogram can be used to build a spatially-variant travel distribution as:

$$P_{ij\tau} \equiv P(L_k = j | L_{k-1} = i, \Delta T_k = \tau) = \frac{N_{ij\tau}}{\sum_j N_{ij\tau}} \quad (1)$$

where $N_{ij\tau}$ is the number of consecutive image pairs in the database that start at location i, end at location j, with time interval τ between them.

Regularization. Even though we begin with a database of 6 million photographs, we nonetheless find that some parts of the spatially-variant model are undersampled. For example, although we have 3186 location bins on Earth, the database only contains 299 image pairs starting in Kansas and separated by 14-30 days. Using this empirical distribution alone will lead to zero probabilities assigned to plausible trips. We resolve this issue by regularizing with a distance-based distribution. Specifically, we discretize distances and time intervals into bins. We then estimate a travel distribution from the histogram as:

$$q_{d,\tau} = \frac{N_{d\tau}}{\sum_d N_{d\tau}} \quad (2)$$

where $N_{d\tau}$ is the number of distances between photographs in distance bin d and interval bin τ. Then, $q_{d,\tau}$ is the probability that picture k is distance d from picture $k-1$, given that they are separated by time interval τ. Then, the regularized model is:

$$P_{ij\tau} = \frac{N_{ij\tau} + \lambda_q q_{d(i,j)\tau}}{\sum_j \left(N_{ij\tau} + \lambda_q q_{d(i,j)\tau}\right)} \quad (3)$$

where $d(i,j)$ is the distance between locations i and j, and λ_q is a regularization weight. We perform an additional regularization, obtaining the final travel probability as:

$$P'_{ij\tau} \propto P_{ij\tau} + \sum_a P_{ia(\tau-1)} P_{aj(\tau-1)} \quad (4)$$

which helps fill-in undersampled long-distance travel bins based on the distribution from the next-shorter time interval.

Single-image prior. We determine a prior over initial image location by counting the number of images N_i taken at each location:

$$P(L = i) = \frac{N_i + \lambda_L}{\sum_i (N_i + \lambda_L)} \quad (5)$$

where λ_L is a regularization constant added to allow starting at locations not in the training set. Figure 5 shows the empirical distribution.

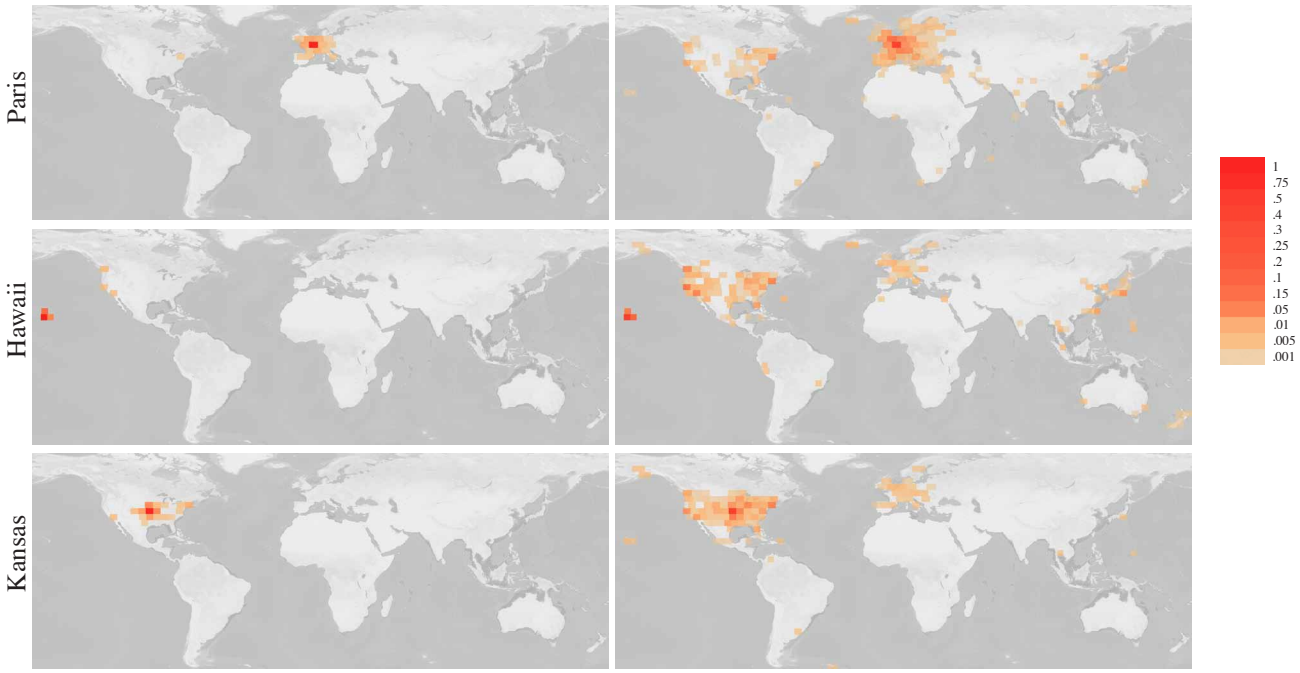

Figure 4. Spatially-variant travel statistics $\left(N_{ij\tau}/\sum_j N_{ij\tau}\right)$ for three starting points and two time intervals, plotted in the log scale. Statistics come from 6 million georeferenced Flickr photographs. Note that there is significant dependence on start and end location, not captured by previous travel models based on Lévy flights. The same log-scale colorbar is used for all distribution plots in this paper.

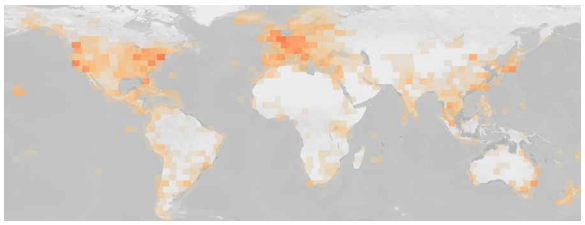

Figure 5. Empirical distribution of image location $(N_i/\sum N_i)$ in the Flickr database. The bin with the largest number of photos is London, representing 5% of the photos.

5. Image Likelihood

We now define an image likelihood term that, given a location bin, describes a distribution over possible images for this location. Schemes based on generative models for image features [8] or geometric feature-matching [17, 19] are presently very limited in geographic scope. Instead, we use a non-parametric likelihood based on matching our database of Flickr images, inspired by the method of Hays and Efros [12]. However, unlike in that work, we must compute a probability distribution, rather directly returning a single location estimate.

For a test image I_k, we first obtain the M most-similar training images I_m. Each image is represented by a descriptor comprising the Gist descriptor [14], a color histogram, a texton histogram, and straight line statistics [12]. A similarity score $D(I_k, I_m)$ is computed as the L^2-distance between image descriptors. We then define the likelihood that the correct bin is i based on all matches $\mathcal{M}_{ik}$ in that bin as:

$$P(L_k = i|I_k) \propto \left(\sum_{m\in\mathcal{M}_{ik}} w_{km}\right) + \lambda_C \qquad (6)$$

based on a normalized matching score

$$w_{km} = e^{-\lambda_w D(I_k,I_m)} / \sum_{\ell=1}^{M} e^{-\lambda_w D(I_k,I_\ell)} \qquad (7)$$

where λ_C is a regularization constant that allows unmatched bins to have nonzero probability.

The image likelihood is then defined by applying Bayes Rule in reverse of its normal application, and substituting in Equations 5 and 6:

$$p(I_k|L=i) = \frac{P(L_k = i|I_k)p(I_k)}{P(L_k = i)} \qquad (8)$$

$$\propto \frac{(\sum_m w_{km}) + \lambda_C}{N_i + \lambda_L} \qquad (9)$$

since $p(I_k)$ is constant for a given image I_k. The numerator normalizes the likelihood, so that more popular locations (which will naturally have more matches) do not have high likelihood solely due to their popularity.

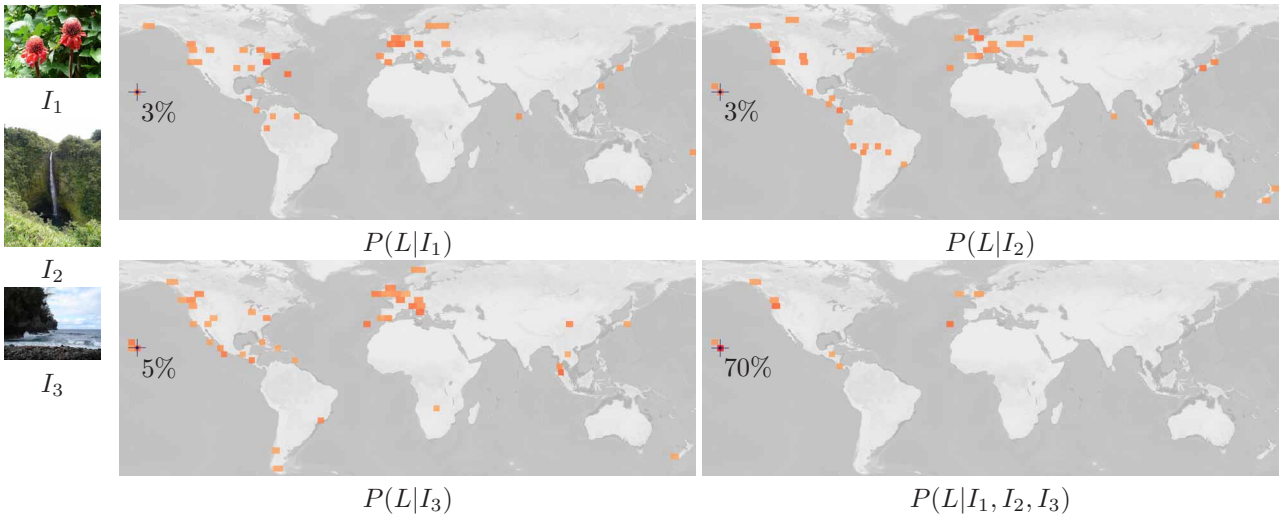

Figure 6. Toy dataset consisting of three photos of Hawaii, illustrating geolocation without landmarks. The ground truth location is denoted with a cross. The raw matches are very noisy, assigning 3-5% probability to the correct bin; none of the images are located correctly on their own. We compute the posterior probability over location (requiring that all three images come from the same bin), and a clear peak emerges in the correct location with posterior probability 70%: this is the bin that best matches all three images. (Note that all distributions are plotted on a log scale.) Adding more images improves the result further.

While these distributions can be very ambiguous for individual images, combining them can yield meaningful estimates. For example, Figure 6 shows the individual posteriors for three images of the same location, together with the joint posterior for all three images. While none of the images can be geolocated based on their individual matches, considering them together yields the correct estimate.

6. Geolocating a New Sequence

Given a new image sequence $I_{1:N}$ with timestamps $T_{1:N}$, our goal is to geolocate each image as accurately as possible. The time intervals $\Delta T_{2:N}$ are first computed by subtraction. The marginal distributions over image locations

$$\gamma_{ki} \equiv P(L_k = i | I_{1:N}, \Delta T_{2:N}) \tag{10}$$

are then computed with the Forward-Backward algorithm [1].

There are a few important implementation details. First, unlike the basic HMM model, the transition probabilities vary at each k. However, it is trivial to modify the Forward-Backward algorithm to handle this. Second, the image likelihood is not normalized, since we cannot directly compute $p(I_k)$. However, the output of Forward-Backward can be normalized by summing γ over locations i for each k. Finally, since the transition probabilities are matrices of size 3186×3186, direct application of Forward-Backward would be very slow. However, the transition matrices are also very sparse (since, for short-duration trips, most destinations have zero probability). Hence, the inner loops of Forward-Backward can be implemented efficiently using sparse matrix multiplication. Figure 7 shows a toy example of computing γ for a two-image sequence, illustrating the value of the travel model.

User-Specific Learning. It often occurs that a test photo does not have any good matches in the training database, but is similar to another test photo from the same location that does have good matches. We can exploit this observation by learning a user-specific image likelihood model for a test sequence. Our algorithm is based on Expectation-Maximization. Conceptually, the algorithm alternates between computing the location distribution γ for each image, and then inserting (or replacing) these images in the training set, location-weighted by γ. In practice, we can fold these steps together into multiple iterations of Forward-Backward. The first Forward-Backward iteration is run normally, as above. Then, for subsequent passes, the image conditional (Equation 6) is replaced with:

$$P(L_k = i | I_k) \propto \sum_{m \in \mathcal{M}_{ik}} w_{km} + \sum_{n=1}^{N} \gamma_{ni} w_{kn} + \lambda_C \tag{11}$$

where w_{kn} is the image matching score between test images k and n, and γ_{ni} is the result of the previous Forward-Backward pass. The output of this algorithm is the final γ distribution. This algorithm is not guaranteed to converge; nonetheless, we find that running 3 iterations of Forward-Backward significantly improves prediction performance. (It is possible to define a convergent version of this algorithm, which we leave for future work.)

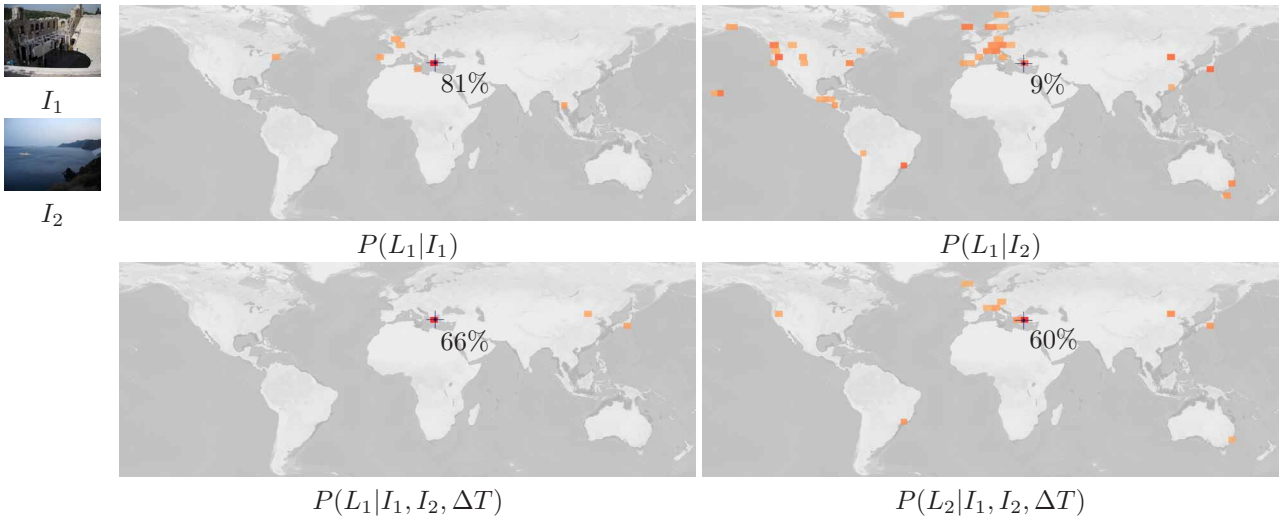

Figure 7. Toy dataset consisting of two images separated by two hours and several hundred kilometers, illustrating the benefit of a travel model. The first image is from the Acropolis at Athens, and is easily geolocated based on single-image matches. The second image, taken in an adjacent bin at Santorini, cannot be geolocated based on single-image matches: the probability at the correct bin is 9%. However, the posterior distribution γ (bottom row) corrects this, assigning probability 60% at the correct location in the second image. While other nearby bins in the Mediterranean would also be reasonable interpretations for the second image, using the entire sequence from which these two images were taken yields very high-confidence predictions at the correct locations.

Location estimation. For many applications, we must output a single latitude and longitude estimate for each image. To do so, we first convert the posterior γ for image k into a continuous PDF:

$$p_k(\mathbf{x}) = \sum_i \gamma_{ki} u_i(\mathbf{x}) \tag{12}$$

where $\mathbf{x}$ denotes a location on the Earth, and u_i is a uniform distribution over bin i. The MAP estimate would be to pick any point within the bin with the largest value of γ. However, it may be preferable to pick a high-probability region where two neighboring bins have high probability (e.g., when a city straddles two bins).

Instead, we use a location estimator that maximizes the probability of being near the correct answer. We specify a distance threshold R. The posterior probability that a location estimate $\mathbf{y}$ is within R of the actual location $\mathbf{x}$ is:

$$P(||\mathbf{y} - \mathbf{x}|| \leq R) \quad = \quad \int_{||\mathbf{x}-\mathbf{y}|| \leq R} p_k(\mathbf{x}) d\mathbf{x} \tag{13}$$

The optimal estimate $\mathbf{y}^*$ maximizes this probability and represents the location with the most probability mass within radius R. This estimator converges to MAP as $R \to 0$. We compute this estimate by a numerical approximation. Specifically, we represent $p_k(\mathbf{x})$ as an image, and compute the posterior probability by convolution of p_k with a disc of radius R (ignoring error due to boundaries and distortion). The estimate $\mathbf{y}^*$ is then the pixel location with the largest value.

Cross Validation. We estimate the parameters λ_C, λ_L, λ_w, λ_q, and M by cross-validation [1]. Cross-validation searches for parameter settings that maximize an estimation score on a validation set of geotagged images. For each set of parameters, location estimates $\mathbf{y}^*$ are computed for all validation images. The score is the percentage of images for which the estimates are within distance R of their true locations. Cross validation returns the set of parameters with the best score.

7. Experiments

We used the IM2GPS data [12] as our training database, which includes about 6 million geotagged images from Flickr.com, posted up to November 2007, and filtered to remove some images inappropriate for matching. For learning the travel priors, we used additional heuristics to remove users with implausible travel, such as users that appear to travel 100 km in under 45 minutes. For testing, we downloaded images from Flickr.com posted after November 2007. We filtered out inappropriate images by the same criteria as above, as well as removing users that had more than 300 pictures in a single location, users that visited at least 3 locations with less than 3 pictures each, users that had more the 1300 pictures total, and users with obviously incorrect geotags. We split the remaining images into a validation set of 6 users (comprising 2005 photos), and a test set of 20 users (4117 photos). These two sets are visualized in Figure 8. All results reported are scores for the test set. As a baseline, we compare our method (SEQ) to single-

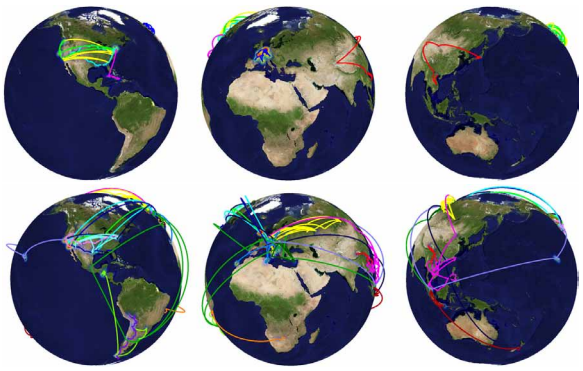

Figure 8. Trajectories of the users in the validation set (top) and test set (bottom).

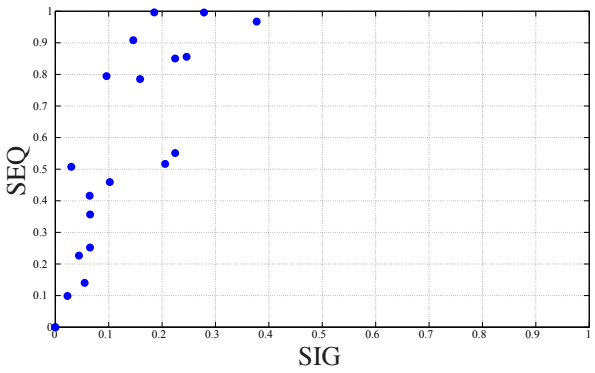

Figure 9. Per-user results. Each dot corresponds to one of the test users, with the single-image (SIG) score on the horizontal axis and sequence (SEQ) score on the vertical axis. SEQ performs about four times better than SIG for any given user. SEQ often performs above 80%, whereas SIG never performs above 40%.

image geolocation (SIG), in which Equation 6 is applied for each image independently (using adapation based on EM for both methods). We performed cross-validation to obtain the best parameters for each algorithm.[1] We report results as the percentage of images for which the location estimate is within $R = 400$ km of the ground-truth location.

We find that SEQ performs dramatically better than SIG, obtaining near-perfect results for some users. For example, SEQ geolocates the images from three of the users with more than 95% accuracy (two of which are shown in Figure 10), whereas single-image geolocation achieves 29% accuracy across the same users.

Across all 4117 images from all test users, the average performance of SEQ is 58%, as compared to 15% for SIG. Note that this the data many images with no obvious geographic cues whatsoever. A baseline algorithm that always returns London (the most common bin in the training set) yields 3% accuracy, and nearest-neighbors [12] yields 10%.

It is possible that SEQ's results amount to matching only images with unique imagery (such as landmarks) and then smoothing locations for the remaining images. We tested this hypothesis as follows. We defined the *distinctive images* to be those correctly geolocated by SIG, according to ground truth. We then replaced the image likelihoods with delta-functions at the correct locations for distinctive images, and with uniform distributions for non-distinctive images. We then re-ran SEQ using these modified likelihoods. The average score dropped from 58% to 39%, thus contradicting the hypothesis. We also measured the content of the non-distinctive images by replacing the likelihoods of only the distinctive images with uniform distributions, yielding a score of 27.5%, which is still well above that of SIG. These tests show that the algorithm uses information from all of the images—landmark matching alone appears to be suboptimal for sequence geolocation.

The quality of sequence geolocation depends on the single-image cues. This relationship is illustrated in Figure 9. For example, there were two users that travelled to locations poorly represented in the training database (including Siberia, Kazakhstan, and Zimbabwe), and their images had almost no good matches. However, when there are correct matches, SEQ dramatically improves performance. For any given user, SEQ performs about four times better than SIG, suggesting that our method can be thought as a way to boost the performance of any single-image cue.

We found that user-specific learning improved both SIG and SEQ by an average of 6% over not learning. We also found that predictions using only the distance distribution prior were worse than the spatially-variant prior, evidenced by the very small values of λ_q selected by cross-validation.

8. Discussion

Our results show that incorporating temporal information into geolocation can dramatically improve accuracy. Even images with no apparent geographic cues can be geolocated, so long as they occur alongside more informative images. We show how careful choice of movement priors can yield more realistic models.

Our work represents a first attempt at sequence geolocation, and there are many opportunities for future research. Our geolocation is fairly coarse due to the binning we chose, but the data supports much finer discretization in many areas. Exploiting other geographic cues ought to improve performance, such as geometirc models of specific landmarks [17, 19], and other meta-data associated with the imagery [7]. We obtain a 4-fold improvement over single-image matching, and better geographic cues can be directly incorporated into the model. We anticipate the use of geolocation for obtaining valuable data for the study of human behavior in multiple disciplines [2, 6, 9, 10, 11, 13, 15].

[1] The estimated values for the full algorithm were: $\lambda_q = 9.9729(10)^{-4}$, $\lambda_C = 0.0244$, $\lambda_L = 0.0521$, $\lambda_w = 7.5$, and $M = 60$.

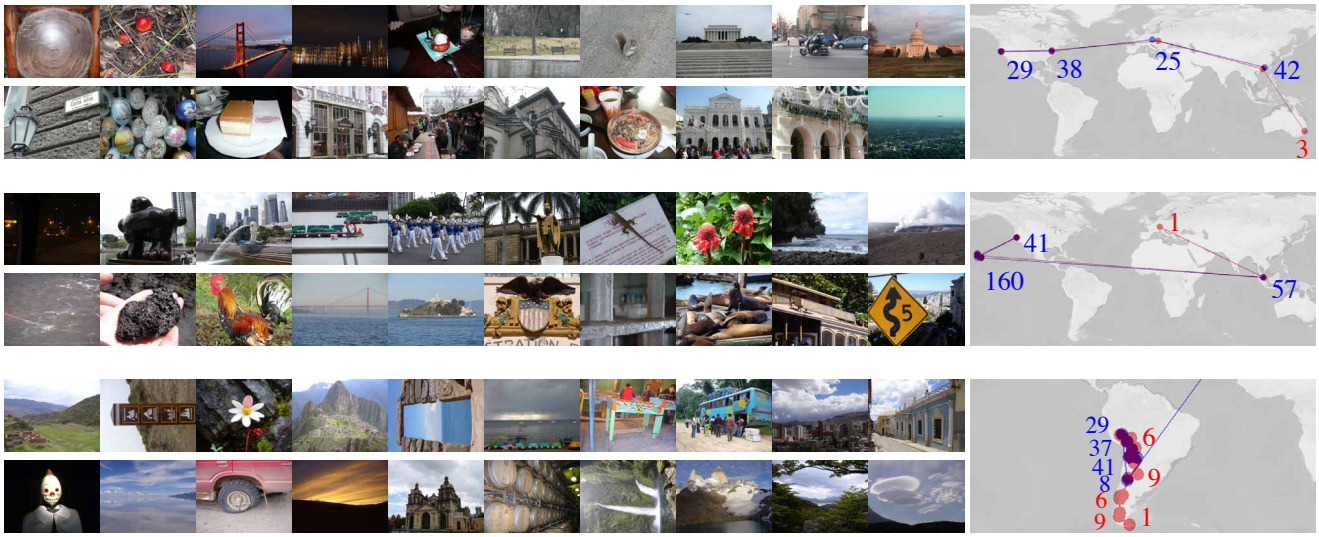

Figure 10. Sample images from three of our test users, and their routes. Ground-truth routes are shown in red, and routes estimated with SEQ in blue. The number of images in each location is shown, with blue numbers indicating correctly-tagged regions, and red indicating errors. **Top:** A user with 137 photos of San Francisco, Washington DC, Budapest, Macau, and Sydney. SEQ geolocates this sequence with 97.8% accuracy, as compared to 37.7% with SIG. The only errors are in Sydney, which is shown in only three aerial views. **Middle:** A user with 259 photos from Switzerland, Singapore, Hawaii, and San Francisco. SEQ geolocates this sequence with 99.6% accuracy, as compared to 18.5% with SIG. The only error is in Switzerland, which is only shown in a single blurry night-time photo. **Bottom:** A user with 146 photos from South America. SEQ geolocates this sequence with 79% accuracy, as compared to 10% with SIG. The algorithm incorrectly labels the last leg of the trip as in the United Kingdom.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[2] D. Brockmann, L. Hufnagel, and T. Geisel. The Scaling Laws of Human Travel. *Nature*, 439(7075):462–465, 2006.

[3] M. Brown and D. G. Lowe. Automatic Panoramic Image Stitching using Invariant Features. *IJCV*, 74(1), 2007.

[4] L. Cao, J. Luo, H. Kautz, and T. S. Huang. Annotating Collections of Photos Using Hierarchical Event and Scene Models. In *Proc. CVPR*, 2008.

[5] A. Clauset, C. R. Shalizi, and M. E. J. Newman. Power-law distributions in empirical data. *SIAM Review*, 2009. To appear; http://arxiv.org/abs/0706.1062.

[6] V. Colizza, A. Barrat, M. Barthelmy, A.-J. Valleron, and A. Vespignani. Modeling the Worldwide Spread of Pandemic Influenza: Baseline Case and Containment Interventions. *PLoS Med*, 4(1), 2007.

[7] D. Crandall, L. Backstrom, D. Huttenlocher, and J. Kleinberg. Mapping the World's Photos. In *Proc. WWW*, 2009.

[8] M. Cristani, A. Perina, U. Castellani, and V. Murino. Geolocated image analysis using latent representations. In *Proc. CVPR*, Apr 2008.

[9] F. Girardin, F. Calabrese, F. Fiore, C. Ratti, and J. Blat. Digital Footprinting: Uncovering Tourists with User-Generated Content. *Pervasive Computing*, 7(4):36–43, 2008.

[10] F. Girardin, F. D. Fiore, C. Ratti, and J. Blat. Leveraging explicitly disclosed location information to understand tourist dynamics: a case study. *J. of Location Based Services*, 2(1):41–56, Mar 2008.

[11] M. C. González, C. A. Hidalgo, and A.-L. Barabási. Understanding individual human mobility patterns. *Nature*, 453(7196):779–782, Jun 2008.

[12] J. Hays and A. A. Efros. IM2GPS: estimating geographic information from a single image. In *CVPR*, 2008.

[13] L. Hufnagel, D. Brockmann, and T. Geisel. Forecast and control of epidemics in a globalized world. *PNAS*, 101(24):15124–15129, Oct. 2004.

[14] A. Oliva and A. Torralba. Building the gist of a scene: The role of global image features in recognition. *Visual Perception, Progress in Brain Research*, 155, 2006.

[15] T. Rattenbury and M. Naaman. Methods for Extracting Place Semantics from Flickr Tags. *ACM Trans. Web*, 3(1), 2009.

[16] F. Schaffalitzky and A. Zisserman. Multi-view matching for unordered image sets or "How do I organize my holiday snaps?". In *Proc. ECCV*, 2002.

[17] G. Schindler, P. Krishnamurthy, R. Lublinerman, Y. Liu, and F. Dellaert. Detecting and Matching Repeated Patterns for Automatic Geo-tagging in Urban Environments. In *Proc. CVPR*, Mar 2008.

[18] N. Snavely, S. M. Seitz, and R. Szeliski. Photo Tourism: Exploring Photo Collections in 3D. *ACM Trans. on Graphics*, 25(3):835–846, July 2006.

[19] W. Zhang and J. Kosecka. Image Based Localization in Urban Environments. In *Proc. 3DPVT*, May 2006.

Acknowledgements. This work was supported by CFI, CIFAR, Google, Microsoft Research, NSERC, NSF grants IIS-0546547 and CCF-0541230, and the Ontario MRI. Thanks to Clauset et al. for providing their power-law fitting code online.