Modeling pet behavior during unsupervised walks using a GPS tracker on a domestic cat†

Miguel Castaño Arranz
Division of Signals and Systems
Luleå University of Technology,
SE-971 87 Luleå, Sweden
mail:miguel.castano@ltu.se

***Abstract*—A method for modeling the behavior of a domestic animal in an unsupervised walk is presented. The subject of studies has been a domestic cat. GPS technology has been used to track the animal and determine a set of locations which the animal usually visits during unsupervised walks. The transition between these locations has been modeled as a discrete Markov chain.**
[1]

I. Introduction

Recently GPS location systems for pets have been developed for tracking their unsupervised walks. These technologies are usually combined with a web service for accessing the data on real time as well as for storing the data logged during the walks.

The motivation for the web services is clearly the immediate tracking of the position of the pet. Storing the logged data gives the opportunity to retrieve this information in order to understand routines of the pet. The studying of the routines can be motivated by curiosity, by a better interaction with the animal and also by an additional layer of security which is in this way generated. Additionally, the study of this routines through Markov chains can provide with tools to detect abnormal situations where the pet can be endangered and to know where are the most likely places to look for your animal in case of failure of the GPS system. Previous research on monitoring and identifying cat activities includes the paper [4], where accelerometers are placed on cats in order to identify their activities.

II. Cat Description

Cat behavior is described in [1]. Prior to the selection of an appropriate modeling scheme, the routines of a domestic cat during its walking were investigated, and the following facts have been observed:

- The cat has a set of favorite places that usually visits during its walks.
- The cat might discover new favorite places or might abandon old ones.
- There are some places where the cat is likely to spend larger amounts of time and some others where it is likely to spend shorter time. This suggests the possibility to identify expected residence times for each of the places.
- After leaving home, the cat is most likely to initiate its walk visiting some places than others.
- The transition between places can be described in a probabilistic framework. For example, after visiting the neighbors home, the cat usually gets food and water and it is then likely that the cat will continue its walk instead of returning home.

The first two observations indicate that it is appropriate to identify the places that the cat visits using clustering algorithms on the GPS data. The selected clustering algorithm must include a forgetting factor to be able to adapt to the long term dynamics of the cat. These dynamics include the discover of new favorite places and the abandoning of old ones.

The last three observations indicate that the walk of the cat can be modeled in a probabilistic framework using discrete-time Markov chains (DTMC). Each of the states in the Markov chain would reflect the cat being at one of the identified favorite places.

The cat under observation has been found to have the following favorite places:

1. Home. Where the cat returns after the walk
2. Forest. Where the cat uses to run and hunt.
3. Backyard. Where the cat plays with other cats.
4. Hiding place. Where the cat hides when it is scared.
5. Neighbors home. The cat usually visits this family which has another cat, and the cat usually gets some food and water.

Notice that, when describing the places, we naturally associated an activity to each place. This relationship between places and activities has been used in [3].

III. Model of the cat using Markov chains.

A DTMC is random process which describes the transition between certain states. The probabilities of going from a certain state at time n to another state at time $n+1$ are collected in the so called transition matrix. In our model, the state X_i corresponds to the cat being in the favorite location X_i. The following transition matrix with sampling time equals to 1.5 min has been arbitrary selected trying to reflect the

[1]† This version of the paper is work in progress.

behavior of the observed cat Lilla Tiger.

$$TM = \begin{pmatrix} 1.0000 & 0 & 0 & 0 & 0 \\ 0.0069 & 0.9769 & 0.0115 & 0.0023 & 0.0023 \\ 0.0090 & 0.0105 & 0.9700 & 0.0075 & 0.0030 \\ 0.0274 & 0.0056 & 0.0056 & 0.9571 & 0.0043 \\ 0.0210 & 0.0570 & 0.0570 & 0.0150 & 0.8500 \end{pmatrix} \tag{1}$$

This matrix collects the probabilities $TM_{ij} = P(X_i(n+1)|X_j(n-1))$. The state X_i is the cat being at the favorite places described i, being the favorite places ennumerated in the previous section. The initial probability of this Markov chain is:

$$P_0 = [0, 0.35, 0.35, 0.1, 0.2] \tag{2}$$

This type Markov chain is called stationary Markov chain. This means that the random process is memoryless, and the next state depends only on the current one with no importance of the previous history data. On the contrary, in a Markov chain of order m, the future state would depend on the previous m states. Using a non-stationary Markov chain will provide a better fit at the cost of an increase in the number of parameters. This increase in the number of parameters reduces the flexibility of the model, hindering its estimation and interpretation. The goal of this paper is to obtain a model with a reduced set of data, which can be interpreted to obtain indications about the routines of a cat. Therefore it is of interest to reduce the number of model parameters and a stationary Markov chain has been selected.

This Markov chain is an absorbing chain, being the first state (Home) the absorbing state. This means that there will be a moment in time when this state is reached, and after this moment the state will not be left.

This Markov chain has been simulated to create a history data of the cat walks. The result from the first random walks has been depicted in Fig. 1

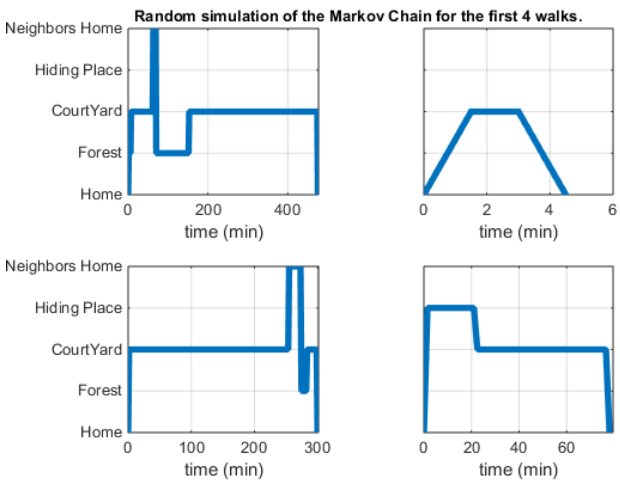


Fig. 1. Simulations of the Markov chain describing the cat walk

IV. Simulation GPS locations from the Markov Chain

This simulation has to now be translated to location data of the cat walking and visiting these places. These locations have been defined in a map depicted in Fig. 2. At each time step, a random position has been generated within the corresponding favorite place. The sequence of random positions has been passed to a moving average filter. The effect of the moving average filter is twofold: first there is a smoothing the data sequence when walking within a favorite place, and second there will be samples generated in the place between favorite locations when a transition takes place. Examples of GPS locations g from walks are depicted in Fig. IV. These locations have been generated from the simulated walks depicted in Fig. 1.

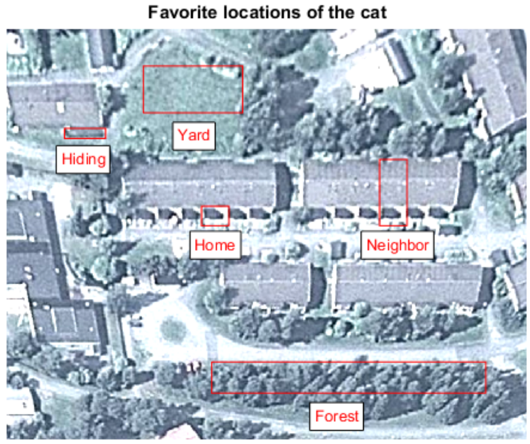
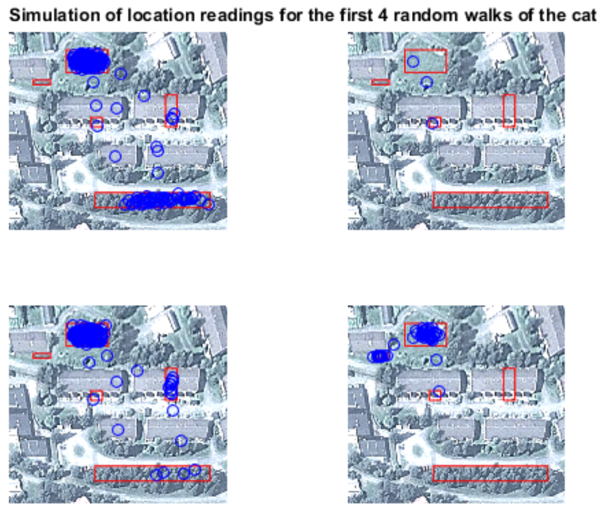


Fig. 2. Map with the favorite locations of the cat



V. CLUSTERING ALGORITHM.

The location data generated in the previous section will be the input for the clustering algorithm to identify the favorite places of the cat.

A. Requirements for the clustering algorithm.

- The number of places visited by the cat and therefore the number of clusters are initially unknown.
- Points can be assigned to no cluster. Some location points are merely a walk or a transition between two of the visited favorite places.
- The algorithm should be noise-tolerant and resistant to outliers. In order to deal with the inherent uncertainty of the GPS system.
- New GPS data can be added to update the clustering partition. Data from new walks should be merged with the previously generated clusters. Clusters related to places which are no longer visited have to be forgotten and new places with their related cluster have to be discovered.

A density-bsed clustering algorithm such as Density-Based Spatial Clustering of Applications with Noise (DBSCAN). This algorithm can deal with the previous requirements with the exception of the addition of new GPS data. For this purpose, the standard DBSCAN algorithm was adapted to perform iterations with new data with the addition of a forgetting factor.

B. Description of the DBSCAN clustering method

The DBSCAN algorithm uses the concept of Eps-neighborhood of a point, which is the set of points that lie closer than a distance Eps from the point. If a point has no points closer than the distance Eps, then the point belongs to no cluster. Each cluster has two kinds of points: border points and core points. A point is a core point if it has a minimum number of points in its Eps-neighborhood. This number is chosen and denoted as $MinPts$.

DBSCAN has been introduced in [2] and used in [5] for the discovery of personal gazzetters.

The DBSCAN algorithm is starts with a random point in the database and determine its Eps-neighborhood. Three alternatives can happen:

- If the point is a core point, then its cluster can be reconstructed by finding the Eps-neighborhood of all the points in the cluster. We remove all the points of the generated cluster from the database.
- If the point is a border point, then we do nothing.
- If the point has no other points in its Eps-neighborhood, we remove it from the database.

We continue visiting points in the database until the database is left empty or all the points have been visited. The solution of the clustering algorithm is unique for given values of Eps and $MinPts$. The selection of these values is usually performed depending on the scales of the problem and the density of points.

C. Adaptation of the DBSCAN clustering method.

With every new batch of location data obtained from a new walk, we have to be able to update the clustering with the new data. This update has to be done with keeping only relevant data in each iteration, and also including a forgetting routine. The forgetting routine has to be used to allow clusters to disappear if they are not regularly visited. In this way, favorite places which are not visited by the cat anymore will gradually disappear, also there will be occasions in which a small new clusters without special significance are generated and will quickly disappear.

After a set of GPS locations is analyzed, only the GPS positions which where classified in a cluster will be stored for the next iterations. When data from a new walk is available, the data set is merged with the stored data which resulted from the previous iteration. Every certain number of iterations, we initiate the forgetting routine. In this forgetting routine, a fraction of the data from each cluster is randomly deleted. This also limits the amount of data stored between the iterations.

The results of applying the clustering algorithm to a series of walks is depicted in Fig. V-C. We can see in this figure (walk 58) how generated clusters with no relation to a favorite place are quickly forgotten. We can also see how sometimes a favorite place is identified as 2 (or more) separated clusters, which will later merge. It is also possible that two favorite places which are very close will merge and be identified as an unique cluster, but the forgetting routine will be able to segregate these two cluster again. Additionally, if a favorite place is not visited recently, the cluster will tend to be reduced and occasionally disappear.

VI. DETERMINING THE MARKOV PARAMETERS FROM THE GSP DATA AND THE CLUSTERS

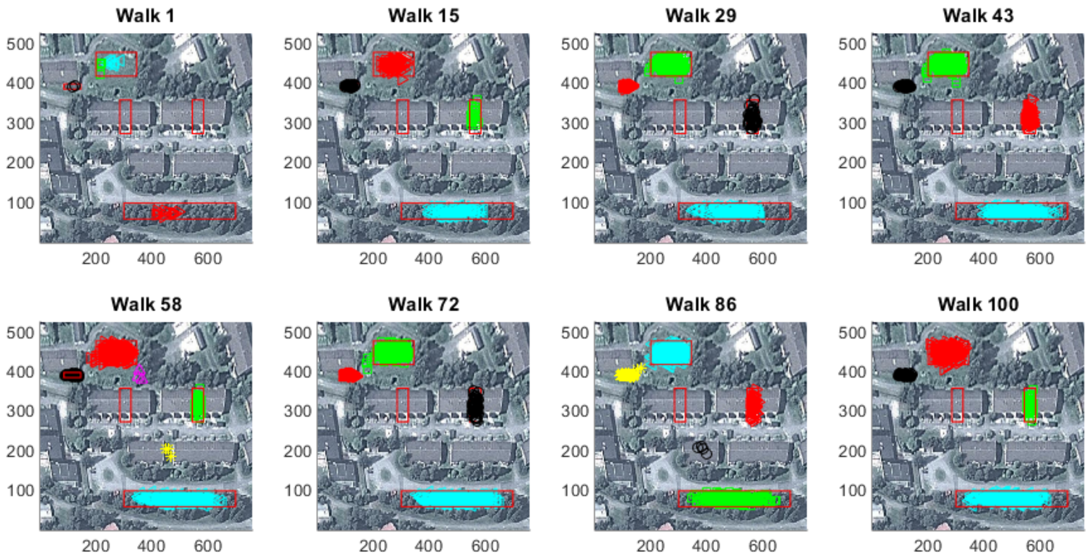
For retrieving the Markov chain associated to the walk of the cat, we need to have history data on the transition between states. To obtain data in this format from the GPS readings, we first need to label the GPS points form the walk sequences according to the clusters to which they belong.

To facilitate the labeling of the data, we first create the convex hull for each of the clusters. A GPS point for the walk sequence is labeled as belonging to a cluster, if it is included in the convex hull associated to the cluster.

After this classification of the GPS data, there will be GPS points which are not belonging to any of the clusters. Since a cluster is generated at the places where the cat spends a significant amount of time, the GPS points associated to no cluster are considered to be a transient behavior in moving between the favorite places. We decided in this work to label the transient points with the same label as the closest sample in time with a label.

For the modeling of the Markov chain, we count the number of transitions between any combination of states and store it in a count matrix C matrix as follows:

where the C_{ij} element of the matrix is the number of times that a transition between the state X_i and the state X_j was observed in the data. The diagonal elements C_{ii} count the



number of times that the state X_i was observed at the time instant n and observed again at the time instant $n+1$.

Normalizing the rows of the count matrix C so that the sum of elements on each row is equal to 1, gives the estimation of the transition matrix P.

After 90 simulations of cat walks, the following estimation of P has been obtained:

$$\hat{P} = \begin{pmatrix} 1.0000 & 0 & 0 & 0 & 0 \\ 0.0084 & 0.9779 & 0.0094 & 0.0022 & 0.0022 \\ 0.0076 & 0.0098 & 0.9721 & 0.0072 & 0.0033 \\ 0.0294 & 0.0098 & 0.0079 & 0.9490 & 0.0039 \\ 0.0135 & 0.0432 & 0.0378 & 0.0162 & 0.8892 \end{pmatrix} \quad (3)$$

VII. Interpreting Markov Chains.

a) Residence time.: The residence time (T_i) is the time that the cat stays in a location s_i. The probabilistic event of the cat that staying or leaving location s_i is a Bernouilli process which occurs at the each time epoch of the DTMC with probability P_{ii} of staying and probability $1 - P_{ii}$ of success (leaving). The number of epochs until the Bernuilli process is successful (the cat leaves the location) follows then a geometric distribution Geom$(1 - P_{ii})$. It follows from the geometric distribution, that the expected value and variance of the residence time are:

$$E(T_i) = \frac{T}{1 - P_{ii}}; var(T_i) = \frac{T \cdot P_{ii}}{(1 - P_{ii})^2} \quad (4)$$

Where T is the sampling time at which the DTMC is evaluated.

b) Expected time to return home: Being the first state X_1 home, we can calculate the expected number of steps until returning home depending on the first visited place X_i by solving the system of linear equations $(\mathcal{P} - I) \cdot E(\tau_1) = B$, where $\mathcal{P}_{ij} = P_{i+1,j+1}$, and B is a column matrix with all elements equal to -1. $E(\tau_1) = [E(\tau_{12}); E(\tau_{13}), E(\tau_{14}), E(\tau_{15})]$ is a vector with the expected amount of steps until returning from each of the states. That is:

$$\begin{pmatrix} -0.0221 & 0.0094 & 0.0022 & 0.0022 \\ 0.0098 & -0.0279 & 0.0072 & 0.0033 \\ 0.0098 & 0.0079 & -0.0510 & 0.0039 \\ 0.0432 & 0.0378 & 0.0162 & -0.1108 \end{pmatrix} \begin{pmatrix} \tau_{12} \\ \tau_{13} \\ \tau_{14} \end{pmatrix} = -1 \quad (5)$$

which results in the expected steps to return home: $E(\tau_{12}) = 102.2053, E(\tau_{13}) = 98.5883, E(\tau_{14}) = 61.4622, E(\tau_{15}) = 91.5681$. These expected values multiplied by the sampling time $T = 1.5min$ lead to the following average times to return home.

c) Next visited place:

d) Tree diagram:

e) : Probability of being in state s' at step n when in state s at step $m < n$.

$$p_{s,s'}(m,n) = Pr\{X(n) = s'|X(m) = s\} = Pr\{X(n-m) = s'|X(0) = \quad (6)$$

The probability to move from s to s' in n¿0 steps is:

$$p_{s,s'}(n) = \sum_{s''} p_{s,s''}(l) \cdot p_{s'',s}(n-l); forall 0 \leq l \leq n \quad (7)$$

f) Expected transitions: Let P be the transition matrix of a Markov chain. The ij-th entry $p_{ij}^{(n)}$ of the matrix P^n gives the probability hat the Markov chain, starting in state s_i will be in state s_j after n steps. RealLife.tex

VIII. Conclusions

We have introduced a new method for identification of pet routines during an unsupervised walk. The method requires a GPS system attached to the pet which is logging data from the walks.

The identification is performed in two steps. First a clustering algorithm is applied on the GPS data resulting in an identification of clusters which are related to the favorite places of the animal. A forgetting routine implemented in the clustering algorithm implies that a location has to be regularly visited in order to form part of the model.

In a second step, a Markov chain model is identified from the GPS data. The number of states of the Markov model is the number of clusters identified in the previous step, being the each of the states representing the cat at one of the clusters.

The results have been applied on simulations from a realistic behavior of a cat. The application on real data is still part of future work. The tracking device distributed by tractive is currently being used to generate a database with real life location data from a cat.

REFERENCES

[1] David G Barratt. Home range size, habitat utilisation and movement patterns of suburban and farm cats felis catus. *Ecography*, 20(3):271–280, 1997.

[2] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press, 1996.

[3] Basel Kikhia, Andrey Boytsov, Josef Hallberg, Zaheer ul Hussain Sani, Håkan Jonsson, and Kåre Synnes. *Structuring and Presenting Lifelogs Based on Location Data*, pages 133–144. Springer International Publishing, Cham, 2014.

[4] Shinichi Watanabe, Masako Izawa, Akiko Kato, Yan Ropert-Coudert, and Yasuhiko Naito. A new technique for monitoring the detailed behaviour of terrestrial animals: A case study with the domestic cat. *Applied Animal Behaviour Science*, 94($1-2$):117 $--$ 131, 2005.

[5] Changqing Zhou, Dan Frankowski, Pamela Ludford, Shashi Shekhar, and Loren Terveen. Discovering personal gazetteers: An interactive clustering approach. In *Proceedings of the 12th Annual ACM International Workshop on Geographic Information Systems*, GIS '04, pages 266–273, New York, NY, USA, 2004. ACM.