Segmentation of Geolocalized Trajectories using Exponential Moving Average

Soda Cisse* — Peggy Cellier** — Olivier Ridoux*

* Université de Rennes 1, IRISA
** INSA Rennes, IRISA
Campus Beaulieu, 35042 Rennes cedex, FRANCE
Soda.Mareme.Cisse@irisa.fr, Peggy.Cellier@irisa.fr, Olivier.Ridoux@irisa.fr

RÉSUMÉ. De nos jours, une quantité importante de données décrivant les trajectoires d'objets mobiles est rendue disponible par la généralisation des capteurs de géolocalisation. Des informations pertinentes, comme les trajets les plus utilisés par les enfants pour aller à l'école ou les rues les plus empruntées par les travailleurs le matin, peuvent être extraites de ces données permettant, par exemple, de repenser l'espace urbain. Une trajectoire est représentée par un ensemble de points \((x, y, t)\) où \(x\) et \(y\) sont les coordonnées géographiques d'un objet et \(t\) une date. Ces données sont difficiles à explorer et interpréter dans leur forme initiale, c'est-à-dire sous la forme d'un ensemble de points \((x, y, t)\), car elles sont bruitées, irrégulièrement échantillonnées et de trop bas niveau. Une première étape pour les rendre utilisables est de les ré-échantillonner, les lisser et les segmenter en segments de plus haut niveau (e.g., "stops" et "moves") afin de rendre plus facile à interpréter. Dans cet article, nous proposons une méthode s'appuyant sur le calcul de moyennes mobiles exponentielles pour segmenter des trajectoires en segments “accélération” ou “ralentissement”. Les expériences que nous avons menées montrent l'efficacité de la moyenne mobile exponentielle comme fonction de lissage. De plus, elles montrent aussi que la différence entre deux moyennes mobiles exponentielles avec des pondérations différentes permet de découvrir des segments d’accélération et de ralentissement pertinents.

ABSTRACT. Nowadays, large sets of data describing trajectories of mobile objects are made available by the generalization of geolocalization sensors. Relevant information, for instance the most used routes by children to go to school or the most extensively used streets in the morning by workers, can be extracted from this amount of available data allowing, for example, to reconsider the urban space. A trajectory is represented by a set of points \((x, y, t)\) where \(x\) and \(y\) are the geographic coordinates of a mobile object and \(t\) is a date. These data are difficult to explore and interpret in their raw form, i.e. in the form of points \((x, y, t)\), because they are noisy, irregularly sampled and too low level. A first step to make them usable is to resample the data, smooth it, and then to segment it into higher level segments (e.g. "stops" and "moves") that give a better grip for interpretation than the raw coordinates. In this paper, we propose a method for the segmentation of these trajectories in accelerate/decelerate segments which is based on the computation of exponential moving averages (EMA). We have conducted experiments where the exponential moving average proves to be an efficient smoothing function, and the difference between two EMA of different weights proves to discover significant accelerating-decelerating segments.

MOTS-CLÉS : géolocalisation, traces, échantillonnage, lissage, segmentation
KEYWORDS : geolocalization, tracks, sampling, smoothing, segmentation
1. Introduction

Nowadays, large sets of data describing trajectories of mobile objects are made available by the generalization of geolocalisation sensors. Relevant information, for instance the most used routes by children to go to school or the most extensively used streets in the morning by workers, can be extracted, for example, to reconsider the urban space. Spacapietra et al. [1] define a trajectory as the evolution of the position (perceived as a point) of an object (e.g., person, animal, vehicle) that is moving in space. These data are difficult to explore and interpret in their raw form, i.e. in the form of points \((x, y, t)\), because they are noisy, irregularly sampled and too low level. Noisy means that they suffer from errors. For instance, even at a stop, the measured coordinates of an object are not constant. Irregularly sampled is simply the observation that time intervals between two consecutive samples may vary a lot. For instance, in our experiments, we have observed a ratio of more than two hundreds between the greatest and the smallest intervals in a truck trajectory, and judging from the extreme positions of the greatest intervals, these large gaps do not correspond to a stop of the truck. Low level means that \((x, y, t)\) points tell little about what a mobile object is really doing. To know more, segments of consecutive positions must be aggregated into segments that represent phases of the life of the mobile object, for instance, "move", "stop", "load", "unload" for a truck, or "move", "at school", "at home" for a pupil. Although our long term objective is to use raw trajectories plus contextual data to mine the semantics of these trajectories, in this article we do not consider contextual data and focus on the raw trajectories.

A first step to make trajectories usable is to resample the data, smooth it, and then segment it into higher level segments (e.g. "stop" and "move") that give a better grip for interpretation than the raw coordinates. Known methods for data cleaning and smoothing use semantic trajectories [2], the maximum speed of the moving object [3], Kalman filters [4], or the Douglas-Pecker algorithm [5] to compute less bulky trajectories without distorting their quality. The most important phase after cleaning trajectories is their segmentation. The most used segmentation model is that of Spacapietra et al. [1] which divides the trajectories into "moves" and "stops" segments. An important drawback of these methods of segmentation is that they need to fix an a priori threshold in order to recognize "stop" segments in spite of the noise. However, the threshold depends a lot on the type of the mobile object, e.g. trucks or pedestrians, or on the scale of the application, e.g. tourism applications where a "stop" in a town may allow moving around in the town.

In this paper, we propose a method for the segmentation of these trajectories in accelerate/decelerate segments which is based on the computation of exponential moving averages (EMA). Experiments shows EMA is an efficient smoothing function, and the difference between two EMAs of different weights proves to discover significant accelerating-decelerating segments, hence "stops" and "moves".

This article is structured as follows: Section 2 describes the cleaning phase and an original way of segmenting trajectories, with experiments. In Section 3, we compare our proposal with existing works on segmentation of trajectories. Section 4 presents future works and the conclusion of this article.

2. Identifying accelerate-decelerate segments

In this section we present how we address the problem of noisy and irregularly sampled data by resampling. Then we explain how we tackle the problem of low level of
data by automatically discovering accelerating-decelerating segments in trajectories. Finally, we report the results of experiments that have been conducted on truck trajectories.

2.1. Resampling trajectories

The GPS raw trajectories of moving objects are represented by a points \((x, y, t)\) where \(x\) and \(y\) respectively represent the latitude and the longitude of the moving object position at time \(t\). These positions are located in the geodetic coordinate system WGS 84 where latitude and longitude are angular coordinates. The angular coordinates are apriori transformed into planar coordinates using the Lambert93 projection.

Raw mobility data is often polluted with errors caused by loss of signal and sensor imprecisions. Experiments with real data show that signal losses are frequent and cause large gaps in the signal. Several methods allow for cleaning raw mobility data and reducing dataset size and noise. However, only a few propose a way to complete a dataset when there are missing points. Our algorithm takes as input a raw trajectory and produces as output a resampled trajectory according to a fixed sampling period. The period can be chosen by the user according to the applications, e.g. depending on mobile object type (pedestrians, vehicles, animals, natural phenomena, etc), or it may be fixed according to a compression objective, e.g. have one million samples. It uses a barycenter interpolation to resample a trajectory.

Barycentric resampling is defined as follows. Let \((x', y', t')\) be a resampled point. It falls in between two consecutive original points \((x, y, t)\) and \((x'', y'', t'')\). The date \(t'\) is fixed by the resampling period, and the position \((x', y')\) is the barycenter of \((x, y)\) and \((x'', y'')\) according to \((t' - t)/(t'' - t)\).

Resampling interpolates missing samples under the assumption of uniform speed during the gap. This is a fair assumption for high frequency sample rates, but certainly false for large intervals of missing samples during which instantaneous speed may change a lot. So, we recommend to resample at a lower rate than the nominal sampling rate of the original trajectory. For instance, when the original nominal sampling period was \(T\), the resampling period should not be more than \(2 \times T\).

2.2. Exponential Moving Average for Segmentation

In the literature, most of segmentation methods [6], [7] and [2] use a priori thresholds for identifying segments; our method does not. It is based on a method used in financial forecasting which consists in comparing two floating averages, one of which is computed relatively to a small time window, the short range trend, the other is computed relatively to a long time window, the long range trend. The idea is that when the long range trend does better than the short range trend we are experiencing a slow down. Vice-versa when the short range trend does better than the long range trend. Comparing the two floating averages indicates periods that the financial jargon calls "bull" and "bear" periods, where optimism and pessimism are in order. The periods start and stop where the graphs of the
two floating averages cross each others. In short, this method smoothes the data, because of the averaging effect, and computes a kind of derivative of the input signal.

Averaging a track of sample points is a way of smoothing it. Note that it requires a really periodic sampling rate to be meaningful, otherwise a naive computation would apply implicitly null weights to missing samples. However, if an equal weight is given to every past point the averaged signal will lag way behind the actual signal. So, more recent points must be given a larger weight than ancient points. If the average is computed by arithmetic mean, the standard solution is to fix a window size, say $N$, inside which samples have weight $\frac{1}{N}$, and outside which samples have weight 0. The greater is the $N$ the greater is the lag, but the better is the smoothing. Another approach is to give exponentially decreasing weights to samples of the past. This is called Exponential Moving Average (EMA) and is formalized as follows:

$$EMA_t = \alpha \times sample_t + (1 - \alpha) \times EMA_{t-1}$$

where $\alpha$ is chosen in $[0, 1]$. The greater is the $\alpha$ the lesser is the weight of the past, the lesser is the lag of the averaged signal with respect to the original signal. So the $\alpha$ plays a rôle similar to the window size, but in reverse. An important difference is that arithmetic mean gives uniform weights to every points in the window, while EMA gives larger weights to more recent points.

Constant $\alpha$ cannot be exactly correlated with a fixed window size, but a rapid computation shows that given a window size $N$, an $\alpha$ equals to $\frac{2}{N+1}$ ensures that 90% of the exponential weights are given to the samples inside the window. Conversely, given an $\alpha$, more than 90% of the highest weights are given to the $N = \frac{2}{\alpha} - 1$ latest samples. So, it is frequent to characterize the $\alpha$ of an EMA by the size of the fixed window average it corresponds to up to 90%. It is believed that it is easier to envisage the impact of a window size than to envisage the impact of an $\alpha$ value. This is what we do in the sequel when we speak of window sizes of 2, 8, etc.; the window sizes are then converted into $\alpha$ coefficients $\frac{2}{3}, \frac{2}{5},$ etc. To be clear we call them pseudo-windows.

EMA has good formal properties. In particular, it lags behind the original signal less than a fixed width floating window arithmetic mean does. An other advantage of EMA is that it is very easy to compute. In particular, it does not require to bufferize past samples like arithmetic mean does. Computing EMA at time $t$ only requires to know the sample at time $t$ and EMA at previous time.

We plan to apply the EMA in an innovative way that is related to our goal of having no a priori knowledge on thresholds and scales of the application. The main idea is to analyze the same trajectory, the same input signal, at multiple scales simultaneously. The different scales are defined as powers of the smallest scale, thus yielding scales of different size order, for instance, 10, $10^2$ and $10^3$, etc, or 2, $2^2$, $2^3$, etc. These scales will be used as the pseudo-window sizes that characterize EMA. So doing, we expect to obtain a compact but rich view of a trajectory. Rich means that though the input signal is compressed in
few segments, it is still possible to distinguish staying in a place while moving inside it, and stopping in that place for a while. In the sequel we use binary scales, i.e. 2^2.

We also plan to consider at the same time two kinds of speeds. First, the instantaneous speed along the trajectory, in other words what the speedometer of a vehicle shows. Second, the vectorial speed between any two samples (x, y, t) and (x', y', t'). Both are computed as the euclidian distance between two points divided by the time that separates them, \(d((x', y'), (x, y))/(t' - t)\). However, instantaneous speed is computed using consecutive points under the assumption that the actual trajectory between the two points is a straight line, whereas vectorial speed can be computed using any two points under no assumption. Measuring vectorial speed at a high sample rate yields a fair approximation of the mobile's instantaneous speed, whereas measuring it at a low sample rate shows to what extent it remains in the same area.

To summarize, we propose to compute speed at different scales to capture different concepts of "stop" and "move", and to average it at different scales to capture different concepts of "accelerate" and "decelerate".

2.3. Experiments

In order to validate our intuition of a multi scale analysis, we have implemented the sampling algorithm and the multi scale EMA computation. We have tested the analysis on two series of real world trajectories. A first series comes from trucks equipped with GPS sensors. These are long range datasets, covering several months in France. In principles, these datasets are sampled at a 5 minute period but they suffer many data losses, with a ratio of more than a two hundred between the largest interval and the nominal sampling period. We have no explanation for this. They are also very noisy: for instance, the measured position of a stopped truck may drift a lot around its actual position. A second series comes from pedestrian equipped with GPS sensors. Their measured activity is limited to visiting an urban park. They are much shorter, a few hours, but they also suffer from data losses and noise. The nominal sampling rate is 5 seconds, and we have observed loss intervals of several hours.

The first step for the segmentation of these trajectories is their resampling. We have chosen to resample trucks trajectories at a 10 minute rate. This was chosen as a rate that compensate for data losses, but does not create too many artificial sample points. The resulting dataset is larger than the original, but not too much.

Figure 1 displays three different scale treatments on a truck trajectory that covers one month. All experiments are presented as follows. The lower half shows the instantaneous speed of the truck (thin line, high peaks mean high speed), plus the average (EMA) speeds for two pseudo-window sizes (thick lines). The upper half shows the differences between the two EMA speeds, and a black and gray line that represents the segmentation (black for "accelerate" and gray for "decelerate"). All diagrams are time-aligned.

The first experiment is for a 10 minute sampling rate and two pseudo-window size of 2 and 8. The result is that EMA fits very well the instantaneous speed, and black and gray segments fit very well the acceleration and deceleration periods, including very small segments that look more like the results of noise than meaningful data. The se-
Figure 1. Three different scale treatments of a truck trajectory
cond experiment is for a 60 minute sampling rate and the same pseudo-window sizes. The lowest sampling rate smoothes the datasets and less noise peaks are considered as segments. In both cases, the EMA speeds (the thick lines) were very close to the instantaneous speed. The third experiment combines the low sampling rate (60 minutes) and a very large pseudo-window size (256 instead of 8). The result is to compare a hourly instantaneous speed with a speed averaged on several days. Many segments are aggregated to form large “activity” segments that correspond to days, or even groups of days (see for instance the large black segment in the first quarter of the trajectory).

A first observation is that at high sampling rates or small pseudo-window sizes the sign of the difference of the short range trend minus the long range trend correctly detects acceleration and deceleration periods. A second observation is that as pseudo-window size increases, or as sampling period increases, EMA lags behind the actual signal more and more. The consequence is that the difference between the two trends tends to no longer represent acceleration, but simply activity. So, for large pseudo-windows or large sampling periods the EMA computation automatically detected “stop” and “move” segments instead of “accelerate” and “decelerate” segments.

We have observed the impact of scaling in two dimensions: resampling rate, and EMA pseudo-window sizes. Increasing scale in either dimension tends to smooth and flatten data. Indeed, peaks tend to be overseen by large period sampling, and to be averaged by large pseudo-window sizes. The bidimensional exploration displays a kind of wavefront along which the scaling parameters differ, but the result is qualitatively equivalent.

We have conducted several experiments with different scales and mobiles with different dynamics. For instance, a second class of mobile objects is pedestrians in an urban park. Their trajectories have been sampled at a 30 second and 300 second rates, and the segmentation done with the same pseudo-window sizes as above (2-8 up to 2-256). As for trucks, it detects accelerations precisely, but since the dynamics of pedestrians is much slower than of trucks the risk of under-sampling is more visible. For instance, the pseudo-window size ratio of 2 to 256 is too large for these rather slow and even-minded mobiles. However, this should not prevent from using such large scales. What is important is that among all the scale size-orders at least one is meaningful.

3. Related works

Few work has been done on the segmentation of trajectories. Anagnostopoulos et al. [10] are among the first to work on the segmentation of GPS traces. However they cut the trajectory in MBR (Minimum Bounding Rectangles) to segment it and not work directly on the points (x, y, t). Spaccapietra et al. [1] proposed a segmentation model composed of “stop” and “move” segments where: “stop” segments represent the phases of the trajectories where the mobile object is stopped; “move” segments are detected between two “stop” and indicate that the object is moving with a certain speed.

Several studies have been based on this model for the trajectories segmentation. Alvares et al. [6] proposed a method for segmenting a trajectory in “stops” and “moves”. Their method is based on a base of geographic places and they look into the trajectory all the sub-trajectories that intersect the geographic places with a given interval for greater than a given threshold time. So, this method uses context data and a priori thresholds. The method of Yan et al. [8] includes all the points of a trajectory whose instantaneous speed is lower than a threshold speed and in which the time interval between the first and the last point of this group of points is greater than a given time threshold. All these consecutive
points will be considered as a "stop" and points between two "stops" as a "move". So, this method still uses several a priori thresholds. This means calculating a new threshold for each type of moving object, and probably for every application, which may be restrictive when several different types of mobile objects are managed at the same time. Zheng et al. [7] have developed a framework for segmentation where they identify "walk segments" (segments where the moving object is moving) and use them to obtain "non-walk segments". This is the opposite strategy of the other previous works where "moves" are in fact "non-stops". Buchin et al. [9] present a theoretic framework to compute segments of a trajectory using criteria like speed, direction, etc.

4. Conclusion and future works

We have presented a new approach for segmenting geolocalized trajectories. It is fundamentally multi-scale, and uses very little a priori knowledge (only the sampling rate). It is based on an iterative usage of exponential moving averages at different scales. At low scale (i.e. high sampling rate or small pseudo-window size) it detects "acceleration" and "deceleration", whereas at high scale (i.e. low sampling rate or large pseudo-window size) it detects "activity" and "non-activity", more conventionally called "move" and "stop". Further work is needed to find a global representation of the multi-scale analysis (either internal, for programs, and visual, for users), and still further work is needed to combine this analysis with contextual data like maps or agendas.

5. Bibliographie