

LEARNING TRAJECTORY PATTERNS BY SEQUENTIAL PATTERN MINING FROM PROBABILISTIC DATABASES

Josky Aïzan¹, Cina Motamed² and Eugene C. Ezin³

¹Ecole Doctorale Sciences Exactes et Appliquées

²Laboratoire d'Informatique Signal et Image de la Côte d'Opale Université du Littoral Côte d'Opale, France

³Institut de Mathématiques et de Sciences Physiques
Université d'Abomey-Calavi, Bénin

ABSTRACT

In this paper, we use Sequential Pattern Mining from Probabilistic Databases to learn trajectory patterns. Trajectories which are a succession of points are firstly transformed into a succession of zones by grouping points to build the symbolic sequence database. For each zone we estimate a confidence level according to the amount of observations appearing during trajectory in the zone. The management of this confidence allows to reduce efficiently the volume of useful zones for the learning process. Finally, we applied a Sequential Pattern Mining algorithm on this probabilistic databases to bring out typical trajectories.

KEYWORDS

Trajectory Patterns, Data Mining, Sequential Pattern Mining, Probabilistic Databases

1. INTRODUCTION

The study of human activities and behaviour is an important research area in computer vision. Nowadays, automatic activities and behaviour understanding have gained great deal of attention. Using unsupervised methods, researchers try to observe a scene, learn prototypical activities and use prototypes for analysis. This approach has been of particular interest for surveillance [1],[2] and traffic monitoring [3]-[5] where methods for categorizing observed behavior, detecting abnormal actions for a quick response, and even predicting future occurrences are highly solicited.

Because of large amounts of data in use for these applications, it is difficult to manually analyze each individually which needs the use of unsupervised methods. In these cases, the data mining in general and the Sequential Pattern Mining (SPM) in particular appear as promising solutions. However, it is recognized that data obtained from a wide range of data sources is inherently uncertain [6], [7]. This paper is concerned with SPM in probabilistic databases [7], a popular framework for modeling uncertainty and its application to learning trajectory.

This work is organized as follows. In section 2, we present the state of art and related works on SPM and uncertain SPM. Section 3 describes problem statement while section 4 gives explanations about learning trajectory with uncertain SPM. Finally, in section 5, we present experimental results and their analysis. A conclusion ends this work with further directions.

2. STATE OF ART AND RELATED WORKS ON SPM AND UNCERTAIN SPM

The task of sequential pattern mining consists of discovering interesting subsequences in a set of sequences. The sequential ordering of events is taken into account unlike pattern mining introduced by Agrawal and Srikant [8] for finding frequent itemsets. The first sequential pattern mining algorithm is called AprioriAll [9]. The improved version of this algorithm is Generalized Sequential Pattern algorithm (GSP) [10]. These two algorithms are inspired by the Apriori algorithm for frequent itemset mining [8]. GSP algorithm uses a standard database representation, also called a horizontal database and performs a breadth-first search to discover frequent sequential patterns. In recent years, other algorithms have been designed to discover sequential patterns in sequence databases. The Spade algorithm [11] inspired by the Eclat algorithm [12] for frequent itemset mining, is an alternative algorithm that uses a depth-first search. It uses a vertical database representation rather than a horizontal database representation. The vertical representation of a sequence database indicates the itemsets where each item i appears in the sequence database [11],[13], [14]. For a given item, this information is called the IDList of the item.

Spam [13] is another algorithm that is an optimization of Spade and also performs a depth-first search using bit vector IDLists. Recently, the Spam algorithm [13] and Spade algorithm [11] were improved to obtain the CMSpam and CM-Spade algorithms [14] both based on the observations that Spam and Spade generate many candidate patterns and perform the join operation to create the IDList of each of them is costly. Besides depth-first search algorithms and vertical algorithms, another important type of algorithms for sequential pattern mining is pattern-growth algorithms. These algorithms are designed to address a limitation of the previously described algorithms, which is to generate candidate patterns that may not appear in the database.

Uncertainty in SPM can occur in three different aspects: the source (an event is recorded deterministically, but the source is not readily identifiable), the event (the source of the data is known, but the events are uncertain) and the time (only time is uncertain) may all be uncertain. Uncertainty in the time-stamp attribute was considered in [23] and seems not well-suited to the probabilistic database approach. In this paper, we focus on uncertainty in the source. SPM in probabilistic databases [7], [6] is a popular framework for modelling uncertainty. Recently several data mining and ranking problems have been studied in this framework, including top-k [15], [16], [17] and frequent itemset mining (FIM) [18]-[21]. The SPM problem in probabilistic databases has been studied in [22]. Also, SPM is studied in noisy sequences [24], but the model proposed there does not fit in the probabilistic database framework.

3. PROBLEM STATEMENT

3.1. Deterministic SPM

Let $S = \{1, \dots, p\}$ and $I = \{i_1, i_2, \dots, i_m\}$ be respectively a set of sources and a set of items. An event e is a set of items such that $e \subseteq I$. A sequence database $D = \langle s_1, s_2, \dots, s_p \rangle$ is an ordered list of sequences such that each $s_i \subseteq D$ is of the form (eid_i, e_i, σ_i) , where eid_i is a unique event-id, including a time-stamp (events are ordered by this time-stamp), e_i is an event and σ_i is a source.

A sequence is an ordered list of events $s = \langle e_1, e_2, \dots, e_n \rangle$ such that $e_k \subseteq I$ ($1 \leq k \leq n$). A sequence s is said to be of length k or a k -sequence if it contains k items, or in other words if $k = \sum_{j=1}^n |e_j|$. A sequence $s_a = \langle A_1, A_2, \dots, A_n \rangle$ is a subsequence of another sequence $s_b = \langle B_1, B_2, \dots, B_m \rangle$ denoted $s_a \preceq s_b$, if and only if there exist integers $1 < i_1 < i_2 < \dots < i_n < m$ such that $A_1 \subseteq B_{i_1}, A_2 \subseteq B_{i_2}, \dots, A_n \subseteq B_{i_n}$. Let $D_i = \{e | (eid, e, i) \in D\}$ be the sequence corresponding to a source i ordered by eid . For a sequence s and source i , let $X_i(s, D)$ be an indicator variable, whose value is 1 if s is a subsequence of sequence D_i , and 0 otherwise. For any

sequence s , define its support in D , denoted $Sup(s, D) = \sum_{i=1}^p X_i(s, D)$. The goal is to find all sequences s such that $Sup(s, D) \geq \theta p$ for some user-defined threshold $0 \leq \theta \leq 1$.

3.2. Source level uncertainty

Proposed by Muzammal and Raman [25], the Source Level Uncertainty (SLU) based on a probabilistic database D^p which is an ordered list of records $\langle r_1, \dots, r_n \rangle$ such that each $r_i \in D^p$ is of the form (eid, e, W) where eid is an event-id, e is an event and W is a probability distribution over S ; the lists ordered by eid . The distribution W consists of pairs (σ, c) , where $\sigma \in S$ and $0 < c \leq 1$ is the confidence that the event e is associated with source σ ; we assume $\sum_{(\sigma, c) \in W} c = 1$. An example can be found in Table 1.

A possible word D^* of D^p is generated by assigning each event ei to one of the possible sources $\sigma_i \in W_i$. Thus every record $r_i = (eid_i, e_i, W_i) \in D^p$ takes the form $r'_i = (eid_i, e_i, \sigma_i)$ in D^* . We get the complete set of possible words by enumerating all such possible combinations.

Table 1. Source level uncertain database.

eid	event	W
e ₁	(a,d)	(X:0.6)(Y:0.4)
e ₂	(a)	(Z:1.0)
e ₃	(a,b)	(X:0.3)(Y:0.2)(Z:0.5)
e ₄	(b,c)	(X:0.7)(Z:0.3)

Table 2. A database transform to p-sequence.

	P-sequence
D_X^p	(a, d : 0:6)(a, b : 0:3)(b, c : 0:7)
D_Y^p	(a, d : 0:4)(a, b : 0:2)
D_Z^p	(a : 1:0)(a, b : 0:5)(b, c : 0:3)

$Pr[D^*] = \prod_{i=1}^n Pr_{W_i}[\sigma_i]$ is the probability of a possible word D^* . For example, from the database of Table 1, a possible word D^* can be generated by assigning events e_1, e_3 and e_4 to X with probabilities 0.6, 0.3 and 0.7 respectively, and e_2 to Z with probability 1.0. Thus, $Pr[D^*] = 0.6 \times 1.0 \times 0.3 \times 0.7 = 0.126$. The support of a sequence in a possible word are well-defined because every possible word is a (deterministic) database. The definition of the expected support of a sequence s in D^p follows easily:

$$ES(s, D^p) = \sum_{D^* \in PW(D^p)} Pr[D^*] \times Sup(s, D^*) \quad (1)$$

The problem we consider is: Given a probabilistic database D^p , determine all sequences s such that $ES(s, D^p) \geq \theta m$, for some user-specified threshold $\theta, 0 \leq \theta \leq 1$. Since there are potentially an exponential number of possible words, it is infeasible to compute $ES(s, D^p)$ directly using Equation 1. Next, we show how to do this computation more efficiently.

3.3. Computing Expected Support

A sequence of the form $\langle (e_1, c_1), \dots, (e_k, c_k) \rangle$, where e_j is an event and c_j is a confidence value is called p-sequence. It's analogous to a source sequence in classical SPM. For examples, we write a p-sequence $\langle \{(a, b), 0.3\}, \{(b, c), 0.7\} \rangle$ as $(a, b : 0.3)(b, c : 0.7)$. An SLU database D^p is a collection of p-sequences D_1^p, \dots, D_m^p , where D_i^p is the p-sequence of source i , and contains a list of pairs (e_k, c_k) with $1 \leq k \leq n$, where e_k are those events in D^p that have non-zero confidence of being assigned to source i , ordered by eid (see Table 2). However, the p-sequences corresponding

to different sources are not independent. Thus, one may view an SLU event database as a collection of p-sequences with dependencies in the form of x-tuples [26]. Nevertheless, Muzammal and Raman [25] showed that we can still process the p-sequences independently for the purposes of expected support computation:

$$\begin{aligned}
 ES(s, D^p) &= \sum_{D^* \in PW(D^p)} Pr[D^*] \times Sup(s, D^*) \\
 &= \sum_{D^*} Pr[D^*] \times \sum_{i=1}^m X_i(s, D^*) \\
 &= \sum_{i=1}^m \sum_{D^*} Pr[D^*] \times X_i(s, D^*) \\
 &= \sum_{i=1}^m E[X_i(s, D^p)] \tag{2}
 \end{aligned}$$

where E denotes the expected value of a random variable. Since X_i is a 0–1 variable, $E[X_i(s, D^p)] = Pr[s \preceq D_i^p]$, and we calculate the right-hand expression, which refer to as the source support probability.

Computing the Source Support Probability. Given a p-sequence $D_i^p = \langle (e_1, c_1), \dots, (e_r, c_r) \rangle$ and a sequence $s = \langle s_1, s_2, \dots, s_q \rangle$, a $(q + 1) \times (r + 1)$ matrix $A_{i,s}[0..q][0..r]$ is created (the subscripts on A are omitted when the source and sequence are clear from the context). For $1 \leq k \leq q$ and $1 \leq l \leq r$, $A[k, l]$ will contain $Pr[\langle s_1, \dots, s_k \rangle \preceq \langle (e_1, c_1), \dots, (e_l, c_l) \rangle]$. Set $A[0, l] = 1$ for all l , $0 \leq l \leq r$ and $A[k, 0] = 0$ for all $1 \leq k \leq q$, and compute the other values row-by-row. For $1 \leq k \leq q$ and $1 \leq l \leq r$, define:

$$C_{k,l}^* = \begin{cases} C_l & \text{if } s_k \subseteq e_l \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

The interpretation of Equation 3 is that $C_{k,l}^*$ is the probability that e_l allows the element s_k to be matched in source i ; this is 0 if $s_k \not\subseteq e_l$, and is otherwise equal to the probability that e_l is associated with source i . Now Equation 4 is used.

$$A[k, l] = (1 - C_{k,l}^*) \times A[k, l - 1] + C_{k,l}^* \times A[k - 1, l - 1]. \tag{4}$$

Table 3 shows the computation of the source support probability of sequence $s = (a)(b)$ for source X in the probabilistic database of Table 1. Similarly, we can compute $Pr[s \preceq D_Y^p] = 0.08$ and $Pr[s \preceq D_Z^p] = 0.35$. So, the expected support of (a)(b) in the database of Table 1 is $0.558 + 0.08 + 0.35 = 1.288$.

The reason Equation 3 is correct is that if $s_k \not\subseteq e_l$ then the probability that $\langle s_1, \dots, s_k \rangle \preceq \langle e_1, \dots, e_l \rangle$ is the same as the probability that $\langle s_1, \dots, s_k \rangle \preceq \langle e_1, \dots, e_{l-1} \rangle$ (note that if $s_k \not\subseteq e_l$ then $C_{k,l}^* = 0$ and $A[k, l] = A[k, l - 1]$). Otherwise, $C_{k,l}^* = C_l$ and two disjoint sets of possible words have to be considered: those where e_l is not associated with source i (the first term in Equation 3) and those where it is (the second term in Equation 3). In summary, given a p-sequence D_i^p and a sequence s , by applying Equation 3 repeatedly, $Pr[s \preceq D_i^p]$, is correctly computed.

4. LEARNING TRAJECTORY WITH UNCERTAIN SPM

The types of sequential data commonly used in data mining are time-series and sequences [27]. A time-series is an ordered list of numbers, while a sequence is an ordered list of nominal values (symbols). The problem of sequential pattern mining was originally designed to be applied to sequences [10]. However, it can also be applied to time-series after converting time-series to sequences using discretization techniques.

To build the database of sequences we used a database where trajectories are represented by a succession of points (X, Y) in pixel.

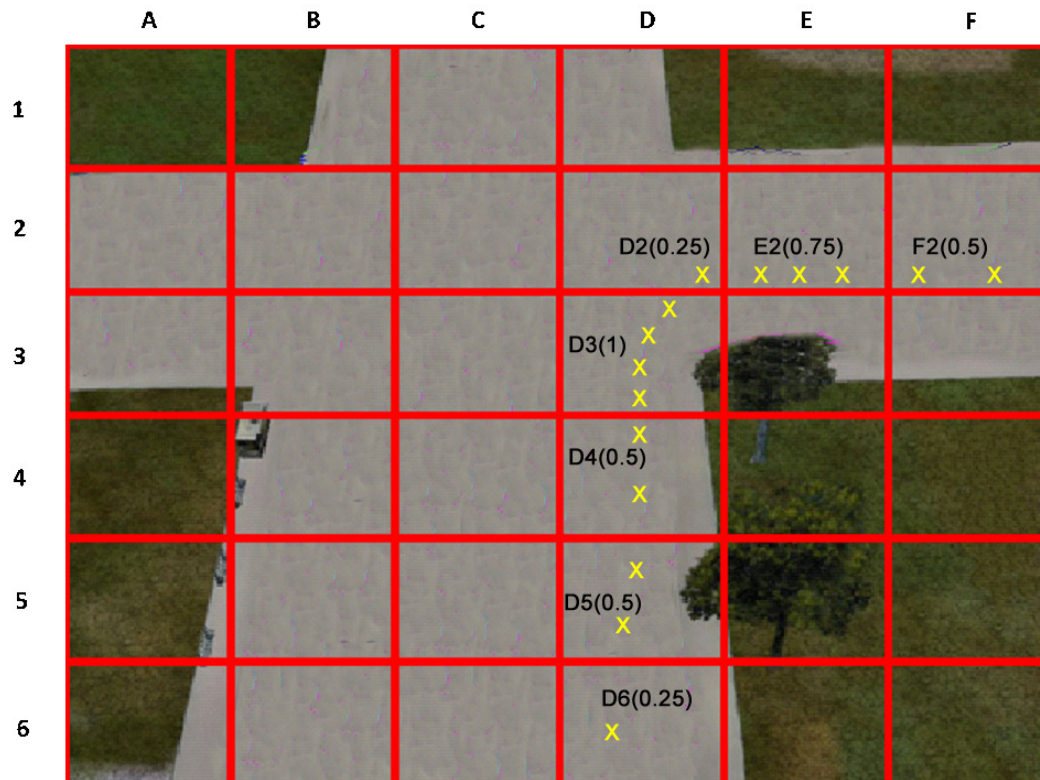


Figure 1. Illustration of trajectory sequence (D6D5D4D3D2E2F2) with the confidence of the symbols

Trajectories are transformed into a sequence of zones by grouping points to build the symbolic sequence database. For example, if we consider intervals of 50 pixels, the points whose X -coordinates belong to the interval $[0, 50[$ are grouped in zone A . Those whose X -coordinates belong to the interval $]50, 100[$ are grouped in zone B and so on. Points whose Y -coordinates belong to the interval $[0, 50[$ are grouped in zone 1. Those whose Y -coordinates belong to the interval $]50, 100[$ are grouped in zone 2 and so on. Therefore, a coordinate point $(50, 100)$ is in zone $A2$ and a coordinate point $(100, 50)$ is in zone $B1$ (see Figure 4 and Figure 5). For each trajectory crossing a zone, a symbol linked to the zone is generated and a confidence (uncertainty) level is estimated based on the number of points inside the zone. The confidence level represents the uncertainty of the symbol linked to this zone. This allows to generate the sequences with uncertain symbols (see Figure 1).

5. SIMULATION RESULTS AND DISCUSSIONS

In this section, we present the results we obtained in our work. We choose the CVRR Trajectory Clustering Dataset [28] for benchmarking trajectory clustering algorithms.

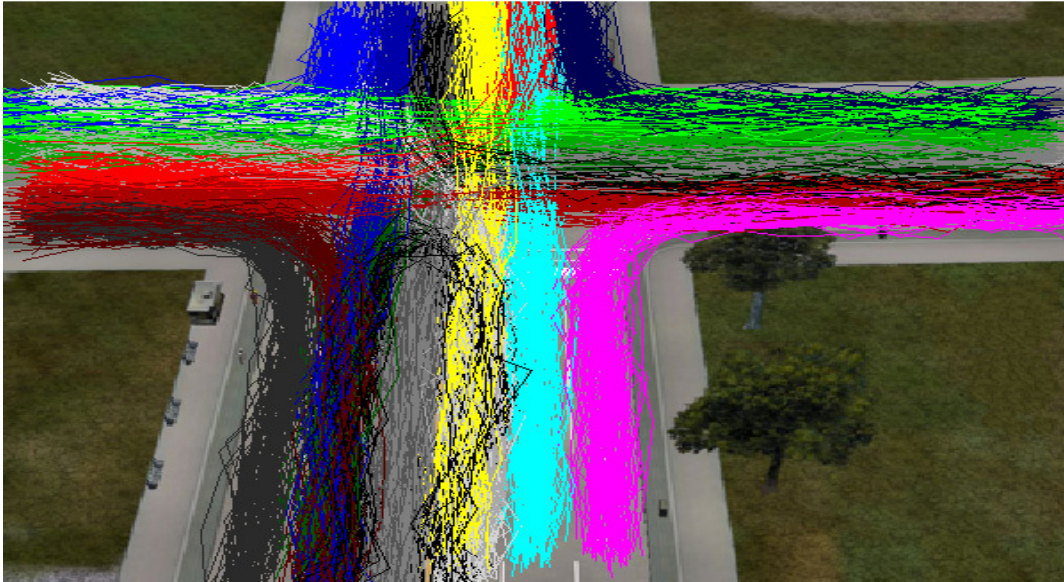


Figure 2. Cross trajectory dataset

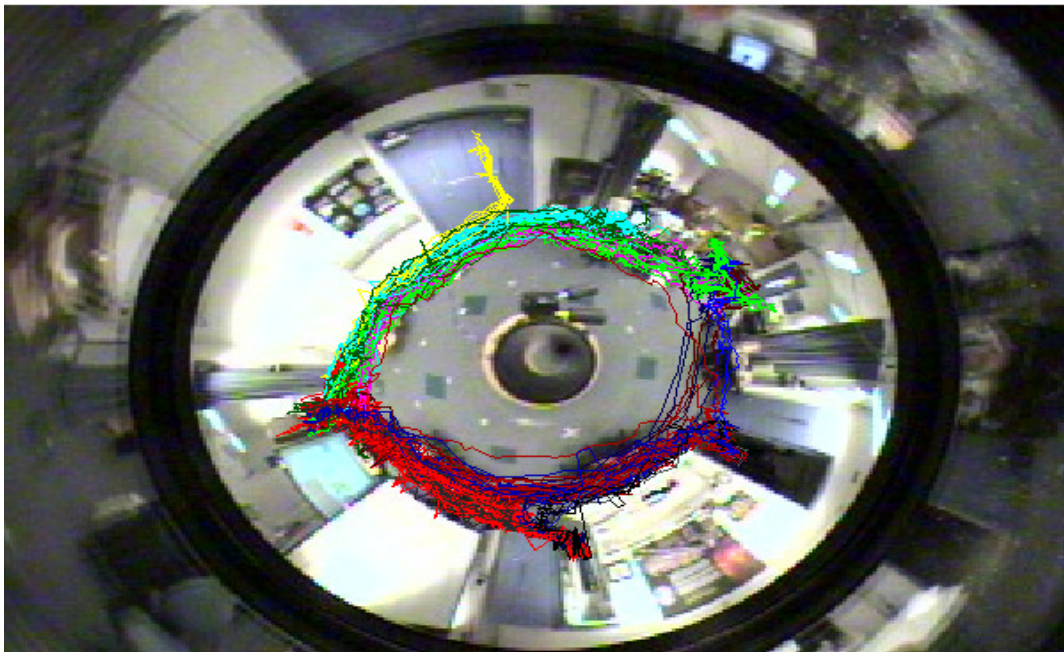


Figure 3. Labomni trajectory dataset

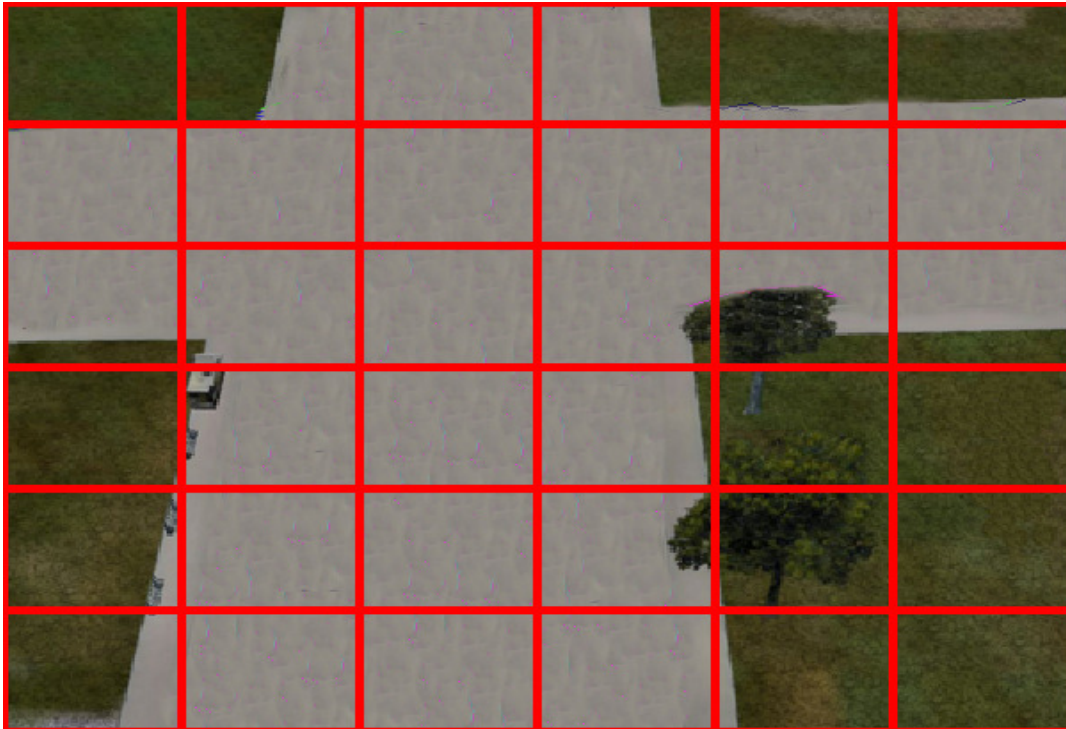


Figure 4. Cross labeled trajectory

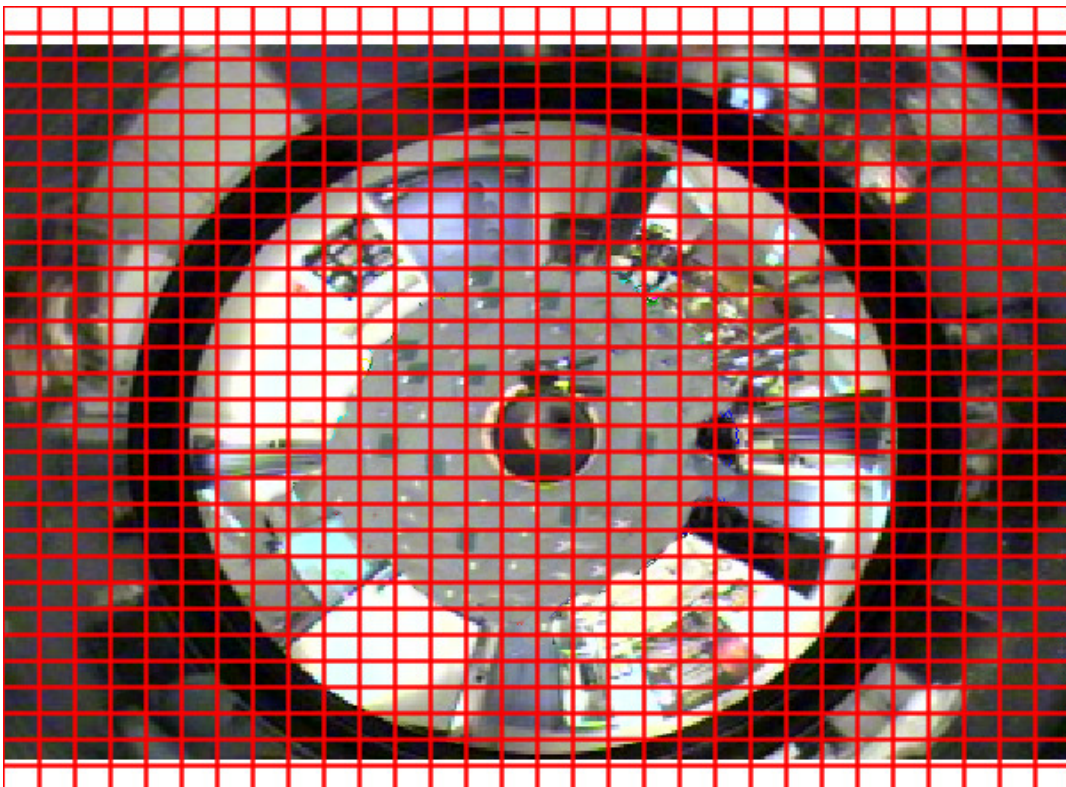


Figure 5. Labomni labeled trajectory

Table 3. Computing of the source support probability.

	$(a, d: 0.6)$	$(a, b: 0.3)$	$(b, c: 0.7)$
(a)	$0.4 \times 0 + 0.6 \times 1 = 0.6$	$0.7 \times 0.6 + 0.3 \times 1 = 0.72$	0.72
$(a)(b)$	0	$0.7 \times 0 + 0.3 \times 0.6 = 0.18$	$0.3 \times 0.18 + 0.7 \times 0.72 = 0.558$

5.1. CVRR trajectory clustering dataset

The full dataset of trajectory similarity/distance measures and clustering algorithms. In our case, we use CROSS (Figure 2) and LABOMNI (Figure 3) dataset.

The CROSS dataset contains a four way traffic intersection. Units are pixels.

The LABOMNI dataset examines humans rather than vehicles. An omni-directional camera was placed in the middle of a lab to observe trajectories from a less constrained environment than encountered by vehicle traffic. The participants were not aware of the data collection to ensure naturally occurring motion patterns. The trajectories have a long time duration and tend to have a large degree of overlap in the image plane. Units are pixels.

Table 4. Results of deterministic SPM and uncertain SPM.

Dataset	SPM frequent patterns	Uncertain SPM frequent patterns	Unreliable frequent patterns rate (%)	Reliable frequent patterns rate (%)
Cross	152	67	56%	44%
Labomni	31	17	45%	55%

5.2. Results and discussions

Our implementation in Java, is executed on a machine Intel(R) Core(TM) i7-7500U CPU @2.70 GHZ 2.90 GHZ running Windows 10. With a support value fixed to 0.05, the different results obtain are in Table 4.

With Cross dataset, there are 152 frequent sequences (obtained with deterministic database) of which 85 are considered unreliable, or a rate of 56% (obtained with the probabilistic database). With Labomni dataset, unlike in the case of Cross dataset, the unreliable frequent sequence rate (45%) is lower than the reliable frequent sequence rate.

It is noted from the results that in the two datasets used, the unreliable frequent sequence patterns rate is not equal to 0%. This result leads us to say that the deterministic SPM returns frequent sequences patterns that are not necessarily all reliable and justifies our choice on the uncertain SPM.

The results also show that the unreliable frequent sequence patterns rate of the Cross dataset is higher than that of the Labomni dataset. This could be explained by the fact that the data of the Labomni dataset are taken in an environment with less constraint than that of the Cross dataset where vehicle traffic is observed.

6. CONCLUSIONS

We have adapted a Sequential Pattern Mining algorithm for probabilistic databases to bring out typical trajectories. The management of the uncertainty of data help to focus on reliable part of the data. By using symbols with their uncertainties, the system estimates reliable frequent trajectory models by using the Sequential Pattern Mining algorithm.

For the future work, two possible extensions will be studied .The first one is the integration of temporal constraints (including the time uncertainty management) and the second extension is on the development of an online recognition system of sequential patterns in the context of uncertain observation and models.

REFERENCES

- [1] C. Stauffer and W. E. L. Grimson, Learning patterns of activity using real-time tracking, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no.8, pp. 747-757, Aug. 2000.
- [2] D. Makris and T. Ellis, Learning semantic scene models from observing activity in visual surveillance, *IEEE Trans. Syst., Man, Cybern. B*, vol.35, no. 3, pp. 397-408, Jun. 2005.
- [3] C. Piciarelli and G. L. Foresti, On-line trajectory clustering for anomalous events detection, *Pattern Recognition Letters*, vol. 27, no. 15, pp.1835-1842, Nov. 2006.
- [4] S. Atev, O. Masoud, and N. Papanikolopoulos, Learning traffic patterns at intersections by spectral clustering of motion trajectories, in *IEEE Conf. Intell. Robots and Systems*, Beijing, China, pp.485-486, Oct. 2006.
- [5] B. T. Morris and M. M. Trivedi, Learning, modeling, and classification of vehicle track patterns from live video, *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 425-437, Sep. 2008.
- [6] Aggarwal, C.C. (ed.): *Managing and Mining Uncertain Data*. Springer 2009.
- [7] Suciu, D., Dalvi, N.N.: Foundations of probabilistic answers to queries. In: Ozcan, F. (ed.) *SIGMOD Conference*. p. 963. ACM 2005.
- [8] R. Agrawal and R. Srikant, Fast algorithms for mining association rules, *The International Conference on Very Large Databases*, pp. 487-499, 1994.
- [9] R. Agrawal, and R. Srikant, Mining sequential patterns, *The International Conference on Data Engineering*, pp. 3-14, 1995.
- [10] R. Srikant, and R. Agrawal, Mining sequential patterns: Generalizations and performance improvements, *The International Conference on Extending Database Technology*, pp. 1-17, 1996.
- [11] M. J. Zaki, SPADE: An efficient algorithm for mining frequent sequences, *Machine learning*, vol. 42(1-2), pp. 31-60, 2001.
- [12] M. J. Zaki, Scalable algorithms for association mining, *IEEE Transactions on Knowledge and Data Engineering*, vol. 12(3), pp. 372-390, 2000.
- [13] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu, Sequential pattern mining using a bitmap representation, *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 429-435, 2002.
- [14] P. Fournier-Viger, A. Gomariz, M. Campos, and R. Thomas, Fast Vertical Mining of Sequential Patterns Using Co-occurrence Information, *The Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 40-52, 2014.
- [15] Hua, M., Pei, J., Zhang, W., Lin, X.: Ranking queries on uncertain data: a probabilistic threshold approach. In: Wang [21], pp. 673-686.
- [16] Zhang, Q., Li, F., Yi, K.: Finding frequent items in probabilistic data. In: Wang [21], pp. 819-832.

- [17] Cormode, G., Li, F., Yi, K.: Semantics of ranking queries for probabilistic data and expected ranks. In: ICDE. pp. 305316. IEEE 2009.
- [18] Aggarwal, C.C., Li, Y., Wang, J., Wang, J.: Frequent pattern mining with uncertain data. In: Elder et al. [9], pp. 2938.
- [19] Bernecker, T., Kriegel, H.P., Renz, M., Verhein, F., Züfle, A.: Probabilistic frequent itemset mining in uncertain databases. In: Elder et al. [9], pp. 119128.
- [20] Chui, C.K., Kao, B.: A decremental approach for mining frequent itemsets from uncertain data. In: PAKDD. pp. 6475 2008.
- [21] Chui, C.K., Kao, B., Hung, E.: Mining frequent itemsets from uncertain data. In: Zhou, Z.H., Li, H., Yang, Q. (eds.) PAKDD. LNCS, vol. 4426, pp. 4758. Springer 2007.
- [22] Muzammal, M., Raman, R.: On probabilistic models for uncertain sequential pattern mining. In: Cao, L., Feng, Y., Zhong, J. (eds.) ADMA(1). LNCS, vol. 6440, pp. 6072. Springer 2010.
- [23] Sun, X., Orlowska, M.E., Li, X.: Introducing uncertainty into pattern discovery in temporal event sequences. In: ICDM. pp. 299306. IEEE Computer Society 2003.
- [24] Yang, J., Wang, W., Yu, P.S., Han, J.: Mining long sequential patterns in a noisy environment. In: Franklin, M.J., Moon, B., Ailamaki, A. (eds.) SIGMOD Conference. pp. 406417. ACM 2002.
- [25] M. Muzammal, and R. Raman, "Mining sequential patterns from probabilistic databases," Knowledge and Information Systems, vol. 44(2), pp.325-358, 2015.
- [26] Cormode, G., Li, F., Yi, K.: Semantics of ranking queries for probabilistic data and expected ranks. In: ICDE. pp. 305316. IEEE 2009.
- [27] J. Han, J. Pei, and M. Kamber, Data mining: concepts and techniques, Amsterdam: Elsevier 2011.
- [28] B. T. Morris and M. M. Trivedi, Learning Trajectory Patterns by Clustering: Experimental Studies and Comparative Evaluation, in Proc. IEEE Inter. Conf. on Computer Vision and Pattern Recog., Maimi, Florida, June. 2009.

AUTHORS

Josky Aizan received a Bachelor Engineering degree from Ecole Nationale d'Economie Appliquée et de Management of Université d'Abomey-Calavi, Bénin in 2004 and Master degree in Computer Engineering in 2016 from Institut de Formation et de Recherche en Informatique of Université d'Abomey-Calavi, Bénin. He is currently a Ph.D student in Computer Engineering at Ecole Doctorale Sciences Exactes et Appliquées of Université d'Abomey-Calavi, Bénin and at Université du Littoral Côte d'Opale, France. His research interests include signal processing, image processing, video processing, machine learning and pattern analysis and recognition.



Cina Motamed is associate professor in Computer Science in the University of Littoral Cote d'Opale, Calais, France. He received his B.Sc. in mathematics, and M.Sc in Electrical Engineering and Computer Science from the University of Caen, France and the PhD degree in Computer Science from the University of Compiègne, France, in 1987, 1989, and 1992, respectively. Current research is concerned with the automatic visual surveillance of wide area scenes using computational vision. His research interests focus on the design of multicamera system for real-time multiobject tracking and human action recognition. He is recently focusing on the uncertainty management over the vision system by using graphical models, and beliefs propagation. He is also interested by unsupervised learning approaches for human activity recognition.



Eugene C. Ezin is a Full Professor in Computer science and Artificial Intelligence. He received his PhD degree with highest level of distinction in 2001 after research works carried out on neural networks and neural fuzzy systems for speech applications at the International Institute for Advanced Scientific Studies in Italy. He is a reviewer of many conferences and journals. His research interests include machine learning, expert systems, signal and image processing, high performance computing, cryptography, information systems and network security.

