Noname manuscript No.
(will be inserted by the editor)

Probabilistic Spatio-Temporal Retrieval in Smart Spaces

Vivek Menon ·
Bharat Jayaraman ·
Venu Govindaraju

Received: date / Accepted: date

Abstract A 'smart space' is one that automatically identifies and tracks its occupants using unobtrusive biometric modalities such as face, gait, and voice in an unconstrained fashion. Information retrieval in a smart space is concerned with the location and movement of people over time. Towards this end, we abstract a smart space by a probabilistic state transition system in which each state records the probabilities of presence of individuals in various zones of the smart space.

We carry out track-based reasoning on the states in order to determine more accurately the occupants of the smart space. This leads to a data model based upon an occupancy relation in which time is treated discretely, owing to the discrete nature of events, but probability is treated as a real-valued attribute. Using this data model, we show how to formulate a number of spatio-temporal queries, focusing on the computation of probabilities, an aspect that is novel to this model. We present queries both in SQL syntax and also in CLP(R), a constraint logic programming language (with reals) which facilitates succinct formulation of recursive queries.

We show that the answers to certain queries are better displayed in a graphical manner, especially the movement tracks of occupants of the smart space. We also define query-dependent *precision* and *recall* metrics in order to quantify how well the model is able to answer various spatio-temporal queries. We show that a query-

Vivek Menon
Amrita Vishwa Vidyapeetham, Kochi 682041, India
Tel: +91-484-2857943
Fax: +91-484-2857941
E-mail: vivek_menon@cb.amrita.edu

Bharat Jayaraman
Department of Computer Science and Engineering
University at Buffalo, Buffalo, NY 14260, USA
E-mail: bharat@buffalo.edu

Venu Govindaraju
Center for Unified Biometrics and Sensors
University at Buffalo, Buffalo, NY 14260, USA
E-mail: govind@buffalo.edu

dependent metric gives significantly better results for a class of occupancy-related queries compared with query-independent metrics.

Keywords Smart Spaces · Abstract Framework · Biometrics · Recognition · Retrieval · Precision · Recall · Data Model · Spatio-temporal Queries · CLP(R)

1 Introduction

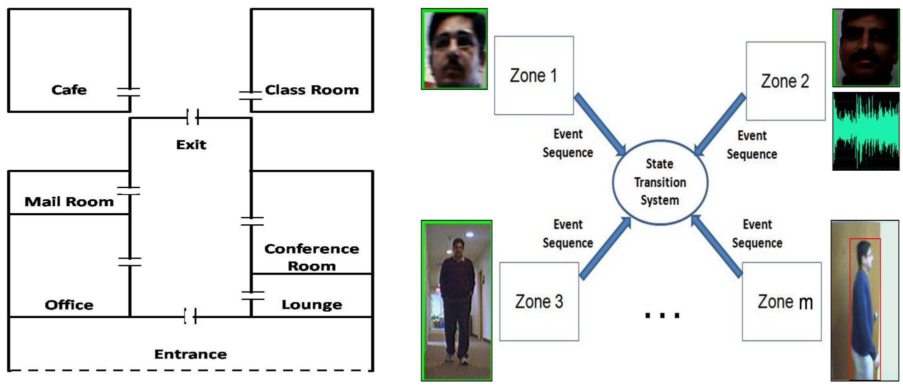
A smart space is a physical space embedded with intelligence and interfaced with humans in a natural way using vision, speech, gestures, and touch, rather than the traditional keyboard and mouse. The key to realizing this paradigm is identifying and tracking people in the space. The ability to identify and track people and answer questions about their whereabouts is critical to many applications. Such smart spaces are very important and beneficial in a number of settings, including homes for the elderly or disabled, office workplaces, and larger areas such as department stores, shopping complexes, train stations, and airports. In some spaces, most of the individuals are known or pre-registered (health-care monitoring) whereas in other spaces most of the individuals are unknown (homeland security).

Let us consider two scenarios from real-life incidents: (1) An elderly resident in an assisted living facility wears an RFID badge to facilitate continuous monitoring of his presence. On one occasion, he enters the elevator alone but gets trapped due to a power failure. The RFID signals transmitted by his badge are not in the range of any receiver. Only much later, when the elevator resumes its service, is he discovered. (2) An intruder has managed to gain illegal entry into a secure facility which is monitored by surveillance cameras. After an intruder alert has been raised, the security personnel set out to find the intruder and relies on inputs from the control room personnel monitoring the facility through multiple video feeds. As the intruder no longer appears on any of the video feeds, the search team has no other option but to search each room.

Automated approaches to transforming multimedia data from video surveillance feeds into a form suitable for information retrieval is a very challenging problem and spans multiple areas - video and audio processing, computer vision, spatio-temporal reasoning and data models. These scenarios also highlight the need for unobtrusive data gathering, where people go about their normal activities without being subject to a 'pause and declare' routine or the burden of RFID tags or badges. Identifying people from their face, gait and voice is more natural and less obtrusive and hence more suited in smart spaces. The overall goal of our research is to develop indoor smart spaces that can recognize and track their occupants as unobtrusively as possible and answer queries about their whereabouts. The sensors of interest in our work are video cameras and microphones that capture biometric modalities such as face, gait, and voice in an unconstrained fashion.

In our previous research, we have focused on multimodal approaches to biometric recognition (Menon et al, 2010) as well as the integration of recognition and reasoning in order to develop a more robust approach to identification and tracking (Menon et al, 2011, 2012a). This paper extends our most recent work (Menon et al, 2012b) on spatio temporal querying in smart spaces and discusses the results of information retrieval and performance of a smart space from a query-dependent perspective. While the basic data is about the location of individuals at various points in time, a fundamental property of this data is that it is probabilistic in nature, due to the inherent inexactness of biometric recognition. For example, during face recognition, camera angle, illumination and face expression could potentially impair the quality of recognition results and lead to significant errors in the overall recognition process. Thus the answers reported by the smart space in response to spatio-temporal queries are not certain but are typically qualified with probability values. We consider a number of different types of queries in this paper, such as 'When did X come to the conference room?', 'Did Y and Z meet in the high-security zone between 6 pm and 7 pm?', 'Where did X go between 2pm to 4pm?', etc.

We develop our spatio-temporal data model starting from an abstract state transition system: states, events, and a transition function (Menon et al, 2008). The state captures who is present in the different regions, or zones of the space. The state changes upon an event, i.e., the movement of an occupant from one zone to another. An event abstracts a biometric recognition step - whether it is face recognition, voice recognition, etc. - and is represented as a set of pairs $\langle o, p(o) \rangle$ where $p(o)$ is the probability that occupant o has been recognized at this event. Thus, the state information is also probabilistic in nature. As we described in our previous research (Menon et al, 2011, 2012a), there are different types of transition functions, each with different properties. But, generally a transition function takes as input a state (or a set of states) and an event, and determines the next state by assigning revised probabilities to the occupants based upon the probabilities in the event. Figure 1 depicts the architecture of a smart space.



(a) (b)

Fig. 1: Architecture of a Multimodal Smart Space

The state transition system model provides a natural basis for retrieval of answers in response to various queries about the whereabouts of occupants in the smart space. While the state information can be taken directly as the basis for information retrieval, we have found in our previous work (Menon et al, 2011) that there will be fewer spurious identifications if we to first perform a 'spatio-temporal track analysis' of the states. The basic idea is that the consecutive track elements of a valid occupant will mostly obey the zone adjacencies in the physical environment, whereas the tracks of spurious occupants will not have this property. Thus, identification is contingent upon the existence of a coherent track for the person with respect to zone adjacencies.

In this paper, we formulate a data model based upon an occupancy relation with a real-valued probability attribute and show how to formulate spatio-temporal queries using the well-known SQL database query language, focusing on the computation of probabilities, an aspect that is novel to this model. We then show how to formulate more complex queries in a constraint-based extension of logic programs, called CLP(R), which permits general recursive queries and reasoning over real-valued variables and arithmetic operations.

While the use of tables for reporting answers to relational queries is standard, we show that for many spatio-temporal queries it is more natural to depict the results of queries in a graphical manner. This is especially the case for depicting the trajectory of occupants across various zones over time. We show how a time-sequence diagram can be used for this purpose. We use graphs for reporting answers to occupancy queries and queries such as 'Did X and Y meet today in the lounge?'. We provide a few illustrations of this approach along with screen shots from our prototype system.

We also formulate precision and recall metrics in a query-dependent manner, since the performance of a smart space is ultimately determined by how well it can respond to the queries that are posed to it. We provide examples for calculation of query-dependent performance metrics based on results from simulation runs using our experimental prototype (Menon et al, 2010) of an eight-zone university building with 45 registered occupants where each of the frequented areas is mapped as a separate zone and named accordingly: entrance, office, mail room, lounge, conference room, classroom, cafeteria and exit as shown in Figure 1a.

We show that a query-dependent metric gives a significantly higher precision for a class of occupancy-related queries compared with a query-independent metric. The reason is that a query-dependent metric is not sensitive to every single event that occurs in the smart space and hence the degree of uncertainty is reduced when such events are not considered.

Our results confirm that the state transition model serves as a concise abstraction of a smart space and that spatio-temporal querying using CLP(R) is very effective in dealing with the query formulation involving probabilistic data from the state transition system. The rest of this paper is organized as follows. Related work is surveyed in section 2, the details of the data model and query formulation is discussed in section 3, constraint based queries are discussed in section 4. Query-dependent performance metrics are discussed in section 5 and conclusions in section 6.

2 Related Work

Traditional database management systems focused on handling precise data in applications such as payroll and inventory. The approach to queries in the domain of databases has been different from the information retrieval arena. SQL queries in databases are associated with a rich structure and a precise semantics that facilitates formulation of complex queries at a user level and complex optimizations at the system level. However, users are expected to have a detailed knowledge of the database to formulate queries successfully. On the other hand, query formulation in Information Retrieval (IR) is based a set of keywords which makes this process easy for casual users. Additionally, IR queries provide two important features, otherwise missing in databases: ranking of results and inclusion of uncertain matches, i.e., the results may include documents that may not match all the keywords in the query.

The data in smart spaces is fundamentally probabilistic and spatio-temporal in nature since people are moving between different zones over a period of time, and we are interested in their trajectories. Hence the data models and query languages of interest in a smart space are probabilistic and spatio-temporal. There has been considerable research on temporal queries (Snodgrass, 1987), spatial queries and spatio-temporal queriesAmicis et al (2011) over the past two decades. Location-based systems (Papandrea and Giordano, 2013) have been a major driver for the interest in moving object databases (MOD), and their associated data models, query languages, indexing, and uncertainty (de Caluwe et al, 2004; Guting and Schneider, 2005; Pelekis et al, 2008).

In addition to the challenges involved in spatio-temporal databases, research into probabilistic databases has gained momentum over the years due to the emergence of a broad range of applications that need to manage large and imprecise data sets in domains such as sensor networks (Deshpande et al, 2004) and various pervasive computing scenarios (Amoretti et al, 2013; Das et al, 2002; McCarthy and Anagnost, 2000). The conventional database management systems are incapable of handling large volumes of imprecise data associated with an increasing number of new applications, as imprecision is modeled in a probabilistic manner. The existing rich query languages coupled with some of the event detection engines such as Cayuga (Demers et al, 2006), SASE (Wu et al, 2006) or SnoopIB (Adaikkalavan and Chakravarthy, 2006) are capable of extracting sophisticated patterns from event streams, though these languages require the data to be precise.

A probabilistic database management system (ProbDMS) (Dalvi et al, 2009) stores large volumes of probabilistic data and supports complex queries in addition to the standard features supported by conventional database management systems. The major challenge in a ProbDMS is that it needs to exhibit scalability with increase in data volume, like conventional database management systems, as well as perform probabilistic inference. Special forms of probabilistic inference that occurs during query evaluation on relational probabilistic data have been proposed: lineage-based representations (Benjelloun et al, 2006), safe plans (Dalvi et al, 2006), algorithms for top-k queries (Re et al, 2007), and representations of views over probabilistic data (Re and Suciu, 2007). Recent work (Cormode and Garofalakis, 2007; Jayram et al, 2007; Re et al, 2008) on probabilistic data streams has investigated queries of varying complexity. Extensions to SQL with provision for uncertain matches and ranked results have been proposed in (Agrawal et al, 2003; Motro, 1988), though with certain restrictions.

Our research on querying in smart spaces (Menon et al, 2011) makes crucial use of probabilistic and temporal concepts, while the spatial issues are treated more in a qualitative (symbolic) than a quantitative (geometric) manner.

3 Data Model and Query Formulation

We begin with a presentation of the data model underlying our query language. While the data model is basically relational in nature, it departs from the standard relational model in the use of a real-valued attribute for the probability. Since the underlying events occur at discrete points in time, we assume that time is discretized as a totally ordered set of hour-minute points $(00:00, 00:01, \ldots, 23:59)$ over a 24-hour period. This can be extended to cover multiple days, months, and years in a straightforward way, as necessary. We assume that every biometric event occurs at one of these discrete time points, however, an event need not occur at every such time point.

Definition (State Relation): *Given a space with occupants* $O = o_1 \ldots o_n$*, zones* $Z = z_1 \ldots z_m$*, and biometric events occurring at distinct increasing times* $T = t_1 \ldots t_x$*, the states of the smart space can be represented as a relation* of the form $state(time,\ occupant,\ zone,\ probability)$*, where occupant* $\in O$*, zone* $\in Z$*, and time* $\in T$*, where* $T \subseteq \{00{:}00,\ 00{:}01,\ \ldots,\ 23{:}59\}$*, a discrete totally ordered set of time units. The attribute probability* $\in \mathcal{R}$*, the set of real numbers, and is functionally dependent on the other three attributes. The state relation satisfies the following integrity constraint:* $\forall t\, \forall i\ \Sigma\{p : (\exists z)\, state(t,\ o_i,\ z,\ p)\} = 1$.

The *state* relation shown in table 1a is based on the states of the smart space. The tuples in the *state* relation correspond to those time units at which the events occur. At each such time t, the state of a smart space is represented by a set of $m \times n$ tuples corresponding to all possible zone-occupant pairs. The *state* relation satisfies the integrity constraint that, for every occupant, the sum of the probabilities of the presence of this occupant across all zones is 1. There is an external zone where all occupants are assumed to be present initially when the smart space is empty. In figure 1, the external zone is not shown, and each person's probability of being in the external zone at any time is 1 - sum of the probabilities of being in one of the internal zones.

The raw data in the *state* relation can be improved through track-based reasoning, as discussed in (Menon et al, 2012a). Essentially we determine the track of an occupant o by selecting only tuples in which o appears the *state* relation. Now, given a graph representing the adjacency information of zones, we can check whether every transition of occupant o from a zone x to a zone y in the state relation corresponds to an edge in the graph. Due to errors in recognition, not every transition will agree with zone adjacency. Therefore, we consider o as a valid occupant if the number of erroneous transitions is a small percentage of all transitions pertaining to o, and we consider o as a spuriously identified occupant otherwise. From our experiments, we have found track-based reasoning to be very effective in minimizing the number of spuriously identified occupants, or false positives. The tuples for all spuriously identified occupants are removed from the *state* relation and the resulting relation is used to define an *occupancy* relation, as follows.

Definition (Occupancy Relation): *Given a smart space with* $O = o_1 \ldots o_n$*,* $Z = z_1 \ldots z_m$*, we define occupancy*$(start, end, person, zone, prob)$*, where start and end define a time interval. The attribute prob* $\in \mathcal{R}$*, the set of real numbers, is*

state(09:27, o_1, entrance, 0.17)
state(09:27, o_1, mail, 0.01)
state(09:27, o_1, office, 0.04)
state(09:27, o_1, conf_room, 0)
state(09:27, o_1, lounge, 0.61)
state(09:27, o_1, exit, 0)
state(09:27, o_1, classroom, 0)
state(09:27, o_1, cafeteria, 0)
state(09:27, o_2, entrance, 0.12)
...
state(09:27, o_2, cafeteria, 0)
...
...
...
state(09:27, o_{45}, entrance, 0.05)
...
state(09:27, o_{45}, cafeteria, 0)

(a) State Relation

occupancy(09:27, 09:29, o_1, entrance, 0.17)
occupancy(09:27, 09:29, o_1, mail, 0.01)
occupancy(09:27, 09:29, o_1, office, 0.04)
occupancy(09:27, 09:29, o_1, conf_room, 0)
occupancy(09:27, 09:29, o_1, lounge, 0.61)
occupancy(09:27, 09:29, o_1, exit, 0)
occupancy(09:27, 09:29, o_1, classroom, 0)
occupancy(09:27, 09:29, o_1, cafeteria, 0)
occupancy(09:27, 09:29, o_2, entrance, 0.12)
...
occupancy(09:27, 09:29, o_2, cafeteria, 0)
...
...
occupancy(09:27, 09:29, o_{10}, entrance, 0)
...
occupancy(09:27, 09:29, o_{10}, cafeteria, 0)

(b) Occupancy Relation

Table 1: Sample State and Occupancy Relations

functionally dependent on the other four attributes, and refers to the probability of presence of a person in a zone during a time interval.

Table 1b is a snapshot of the *occupancy relation*. This relation satisfies the integrity constraint that, for any given occupant o and time-interval, the sum of o's probabilities across all zones = 1 for this time-interval. It may be noted that the state relation is similar to a point-based representation and the occupancy relation is similar to an interval-based representation (Bohlen et al, 1998; Toman, 1996).

We first formulate a couple of simple queries in SQL (Structured Query Language) and then discuss in the next section the use of CLP (Constraint Logic Programming) for more complex queries, including recursive queries. Our focus will be on queries involving the computation of probabilities, as this is the novel part of the work. Below is the syntax of the most basic form of SQL queries:

`SELECT` *attributes* `FROM` *relations* `WHERE` *condition*

We will use the *occupancy* relation defined earlier as the basis for formulating queries. The tuples of the relations that satisfy the *condition* are selected and the relevant attributes are returned as the result. The *condition* is typically a conjunction of simpler tests that serve as a basis for tuple selection. There are numerous extensions to the basic syntax outlined above, in order to perform aggregate operations, grouping, ordering, etc.

A variety of queries can be posed to the smart space regarding the whereabouts of its occupants leading to different classes of response sets. The response set could be based on occupants, zones, or attributes of occupants such as probabilities of their likely presence, time of entry and exit to or from a zone. Additionally, the response set can include derived attributes such as duration of presence, tracks. These queries also could be based on details relating to a single occupant or multiple occupants.

As noted earlier, even though the queries do not make any reference to probabilities, the processing of the query will have to reason about them. A basic integrity constraint of the occupancy relation is that the sum of the probabilities across all zones = 1 at any given time. The queries will illustrate how to combine probabilities when reasoning over multiple time intervals. In combining probabilities, we often would need to known when someone was *not* in a particular zone at a particular time. This is simply 1 - probability of being present in the zone at that time.

Query 1. Did occupant o_3 visit the lounge at 3:00 pm?
Answer: For this query, it is necessary to check the occupancy relation for the presence of o_3 in the lounge at a time interval that contains 3:00 pm. If o_3 was detected to be in the lounge, at most one tuple in occupancy will satisfy the query. For simplicity, we assume overloaded comparison operators `<=`, `<`, `=`, etc., that are defined on time values.

```
SELECT  prob
FROM    occupancy
WHERE   person = o3 and
        zone = lounge and
        from <= 15:00 <= to
```

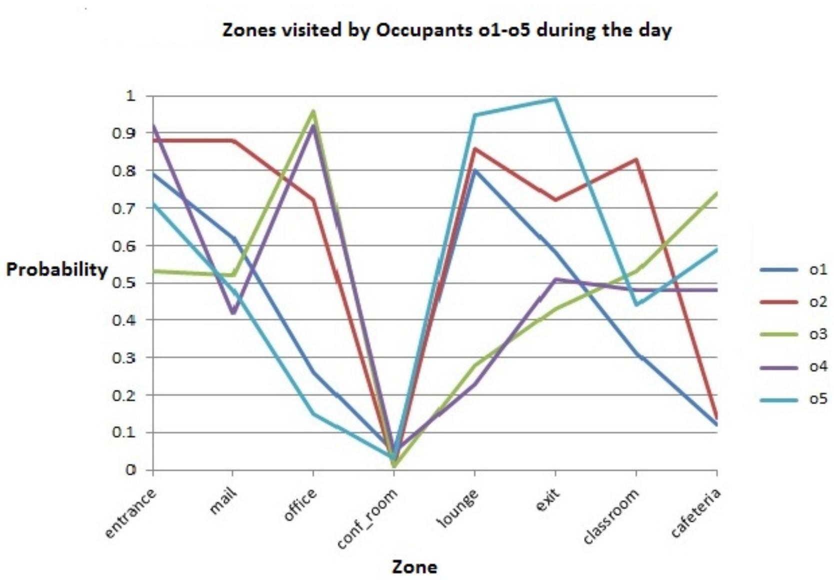


Fig. 2: Zones visited by occupants $o_1 ... o_5$

We can generalize this query for all the zones visited by one or more occupants during the day. Figure 2 shows a graphical depiction of all the zones visited by occupants $o_1 .., o_5$. In this graph, for each zone z and occupant o, the maximum probability of presence of o in z is reported – the maximum is taken over all the probabilities of the presence of o in z over the entire day. The data underlying this result was generated randomly by our simulator for smart spaces. (In figure 2, we can infer that none of these five occupants visited the conference room.)

Query 2. Was occupant o_6 in the lounge during 10:00 am to 11:00 am?
Answer: Since there could be multiple sub-intervals within 10:00 am to 11:00 am during which o_6 was in the lounge (with different probabilities), the answer to the query is 1 minus the product of the probabilities that he was not in the lounge during every such sub-interval. The probability that an occupant was not in the lounge at a given interval = 1 - probability that he was in the lounge during that interval.

```
(1 -
SELECT   PROD(1-prob) as prodprob
FROM     occupancy
WHERE    person = o6 and
         zone = lounge and
         10:00 <= from and to <= 11:00
)
```

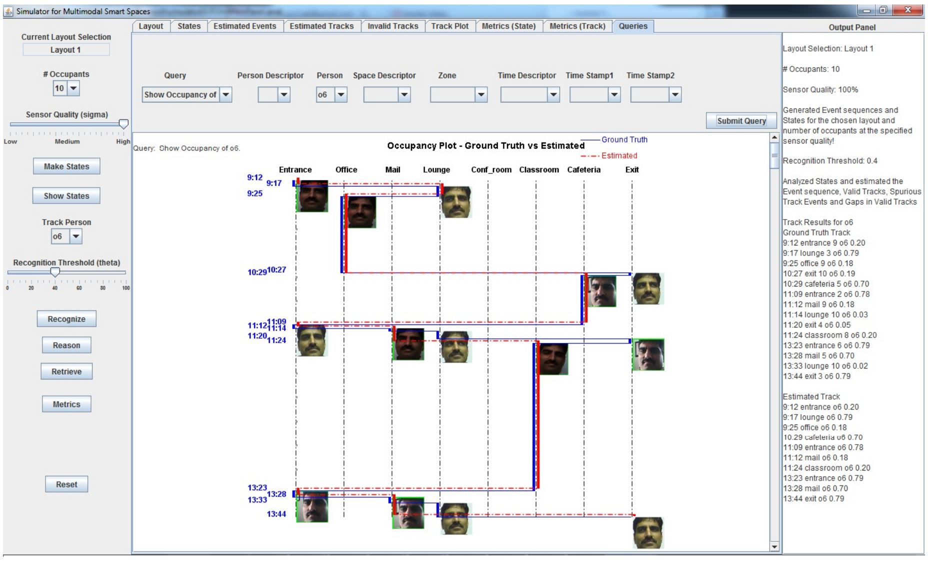


Fig. 3: Track of occupant o_6

Figure 3 is a screen-shot from our prototype and illustrates the occupancy track of occupant o_6 in the smart space. Modeled on lines of a UML time-sequence diagram, this track plot has zone locations along the x-axis and event time stamps along the y-axis. The solid blue lines correspond to the ground truth track and the dashed red lines indicate the estimated track inferred by the smart space. The horizontal lines capture the transition between zones and the vertical bars indicate the duration of presence in a zone. At each zone, the image captured by the camera at that zone is shown. In this example, the estimated track agrees with the ground truth for most part and deviates only for a few events.

4 Constraint Based Queries

Logic programs offer a more expressive query paradigm than SQL because they permit the formulation of general recursive queries. SQL offers the guarantee that all queries will terminate, an important requirement for a database query language. The subset of Horn clauses called Datalog, which is essentially Horn clauses without function symbols (Ceri et al, 1989), also has the strong termination property and has been studied extensively in the literature. Datalog supports recursive queries and has gained much interest in recent years with applications in a number of domains.

Since our underlying data model uses a real-valued attribute for probability along with operations for comparison and arithmetic, it is more natural to adopt the paradigm of constraint logic programming over reals, CLP(R) (Jaffar et al, 1992), rather than Horn clauses. Essentially, CLP(R) extends Horn clauses by generalizing unification to constraint satisfaction. Typically, CLP(R) systems provide solvers for linear equalities and inequalities; non-linear equations and inequations are deferred until one or more variables become bound and they become linear. They also support aggregation predicates, such as *min*, *max*, *sum*, *count*, etc., and we will make use of such operations in our formulations as well.

Definition (CLP(R)): *A CLP(R) program is a collection of rules, which are one of two forms:*

$p(\bar{t})$
$p(\bar{t}) :- p_1(\bar{t}_1) \ldots p_k(\bar{t}_k)$

where each p is a user-defined predicate and each $p_1 \ldots p_k$ may be user-defined or may be one of a pre-defined set of builtin constraint predicates, such as $\leq$, $\geq$, etc. The terms $\bar{t}$ and $\bar{t}_i$ for $1 \leq i \leq k$ include ordinary terms as well as terms composed from real numbers, variables, and the usual arithmetic operators.

Note that `<=` and `=>` are overloaded operators and we use them in this paper for also comparing time units. We also use the addition (+) and subtraction (-) operation over time units.

Query 3. Did o_1 and o_2 meet in the lounge today? Assume that "met" means "being in the same zone at the same time".
Answer: As in query 2, since there are multiple time intervals when o_1 and o_2 could have met, the probability that o_1 and o_2 met in the lounge today is 1 minus the probability that o_1 and o_2 did not meet in the lounge during any of the time intervals. For any interval, the probability of not having met in the lounge during this interval is 1 minus product of the probabilities of their being in the lounge during this interval – predicate `q3` returns this probability for every overlapping interval. We assume a built-in aggregate operation `prod` which multiplies the results of multiple solutions to a goal.

```
query3(1-Prob) :-
        prod(P, q3(P), Prob).
q3(1 - Prob1*Prob2) :-
        occupancy(F, T, o1, lounge, Prob1),
        occupancy(F, T, o2, lounge, Prob2).
```

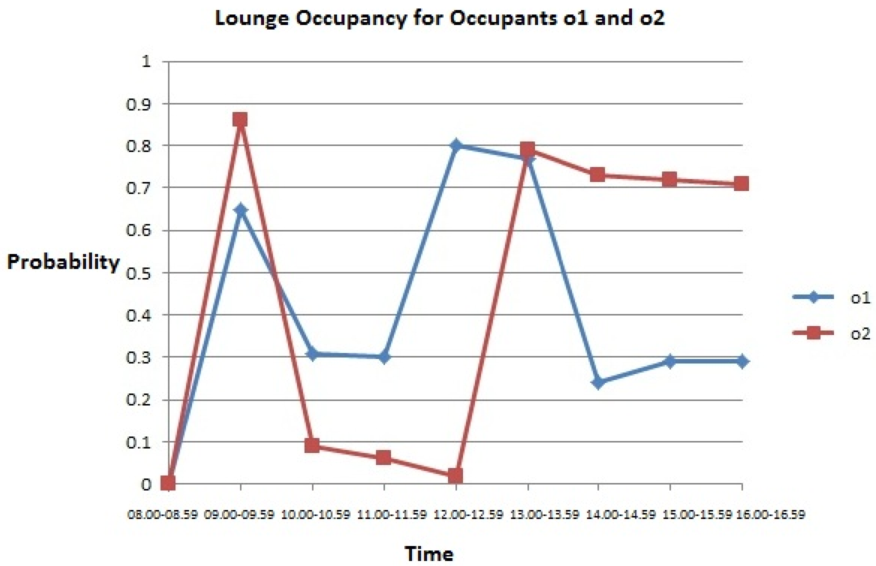


Fig. 4: Graphical output for Query 3: Did o_1 and o_2 meet in the lounge today?

Figure 4 shows the probability of presence (for both occupants o_1 and o_2) in the lounge through the day. The data for this output was again generated randomly by our simulator. From the graph, one can infer that o_1 and o_2 probably met at 9.00 am and again around 1.00 pm.

Query 4. What is the longest contiguous duration during which occupant o_1 was in the office?
Answer: We define the contiguous occupancy in a zone recursively and use this definition in order to define the longest duration. In order to obtain a meaningful response to this query we need to assume that a person is considered to be present in the office if his or her probability is greater than some value, say 0.5.

```
query4(Duration) :-
        setof(F,occupancy(F,_,o1,office,_), FromSet),
        max(D,q4(FromSet,D), Duration).
q4(FS,D) :-
        member(F,FS),
        contiguous(F,T,o1,office),
        D = T - F.
```

```
contiguous(F,T,P,Z) :-
        occupancy(F,T1,P,Z, Prob), Prob > 0.5,
        T2 = T1 + 0:01,
        contiguous(T2,T,P,Z).
contiguous(F,T,P,Z) :-
        occupancy(F,_,P,Z,Prob), Prob <= 0.5,
        T = F - 0:01.
```

It is straightforward to extend the above definition of `contiguous` so that the average probability during this period is also included. Other extensions include the incorporation of distances between adjacent zones and spatial queries that make use of this distance information. As can be seen from the above formulations and possible extensions, the CLP(R) is a powerful paradigm for probabilistic spatio-temporal queries.

5 Query-dependent Performance Metrics

The query-dependent characterization involves evaluating the performance of the model from an information retrieval perspective based on its ability to answer spatio-temporal queries about the space and its occupants. While the query-independent (Menon et al, 2010) approach is holistic and involves performance characterization at a system level, the scope of query-dependent performance characterization is restricted to the spatio-temporal dimensions that are either explicit or implicit from the query of interest. At a very granular level, the window of interest for evaluating the performance may only concern an occupant's presence in a particular zone at a specific point or interval of time, which maps to just one or few states of the state transition model.

The performance metrics for any given query of interest are defined in terms of the ground truth, which is a set of true answers associated with the query. The nature of the response set may vary depending on the type of query posed and includes attributes such as occupants, zones, probabilities of presence, time of occurrence or derived attributes such as duration of presence, tracks, etc., which are based on relations that can be defined as part of the data model. The response set involving the occupants in a zone is defined in terms of *recognition threshold* θ; only those persons with a probability $\geq \theta$ are assumed to be present. For a state where a person's probability in two or more zones is $\geq \theta$, the zone with the highest probability is taken as the zone of his presence.

Definition (Ground Truth): *Given n occupants O=$\{o_1 \ldots o_n\}$ and an event sequence $e_1 \ldots e_x$, then the ground truth, GT, is a sequence $o_{i_1} \ldots o_{i_x}$ where each index $i_1 \ldots i_x$ lies in the range $1 \ldots n$.*

The ground truth basically states which person was the true occupant in question for each biometric event. We first define occupancy-based precision and recall, as follows.

Definition (Precision Recall - Occupancy based): *Given a space with m zones, n occupants $O = \{o_1 \ldots o_n\}$, an event sequence $E = e_1 \ldots e_x$, and ground truth GT. For an occupancy-based query Q, suppose Rel_o is the set of relevant* *occupants that satisfies the query as per GT, and* Ret_o *is the set of retrieved occupants as per the data model and occupancy relation. Then,* $precision_o = |Ret_o \cap Rel_o| \,/\, |Ret_o|$ *and* $recall_o = |Ret_o \cap Rel_o| \,/\, |Rel_o|$.

This definition can be extended to queries that determine durations or time intervals. Essentially, each time interval $< t_1, t_2 >$ can be regarded as a discrete set of time points $\{t_1,\, t_1 + 0:01, \ldots t_2\}$. This leads to the following definition.

Definition (Precision Recall - Time based): *Given a space with* m *zones,* n *occupants* $O = \{o_1 \ldots o_n\}$*, an event sequence* $E = e_1 \ldots e_x$*, and ground truth* GT*. For an time-based query* Q*, suppose* Rel_t *is the set of relevant time units that satisfy the query as per* GT*, and* Ret_t *is the set of retrieved time units as per the data model and occupancy relation. Then,* $precision_t = |Ret_t \cap Rel_t| \,/\, |Ret_t|$ *and* $recall_t = |Ret_t \cap Rel_t| \,/\, |Rel_t|$.

We discuss query-dependent performance metrics by considering a fairly typical class of query and evaluating the occupancy based precision recall metrics for varying number of occupants (n = 5, 10, 15, 20, 25) while keeping the recognition threshold constant at θ=0.4. For example, a query of the form 'Who all were present in the cafeteria between 10:00 am - 11:00 am?' is an example of a spatio-temporal query that retrieves the set of occupants present in a zone (cafeteria) during a specified time interval (10:00 am - 11:00 am). We report performance metrics for this class of occupancy query by varying the zone of occupancy as well as the duration of occupancy. All zones in the smart space in our experimental setup were considered namely, entrance, mail, office, lounge, conf_room, exit, classroom, and cafeteria. The duration of occupancy over varying time intervals in an 8 hour workday were considered. The time intervals considered were 15 minutes, 30 minutes, 1 hour, 2 hours, 4 hours, and 8 hours.

No. of Occupants	Query-dependent Avg. Precision	Query-independent Avg. Precision
5	0.88	0.63
10	0.87	0.83
15	0.87	0.70
20	0.85	0.73
25	0.83	0.82

Table 2: Performance Metrics for Varying Number of Occupants

Table 2 reports the query-dependent precision metrics associated with this class of spatio-temporal query for varying number of occupants in the smart space and compares it with query-independent precision metrics reported in our previous work (Menon et al, 2012a). The query-dependent average precision is computed from precision values across all zones of the smart space over varying time intervals as discussed earlier. As typical queries are not concerned with every single event that occurred in the smart space, query-dependent average precision values are higher than query-independent precision values. (The recall metric did not exhibit appreciable difference between the two cases and hence is not presented here.)

6 Conclusions and Future Work

We have presented an approach to retrieving information on the whereabouts of occupants in a smart space. An important characteristic of such spaces is that the location of an occupant is fundamentally probabilistic in nature due to the inherent uncertainty associated with unconstrained biometric recognition using devices such as video cameras and microphones. Our data model captures this uncertainty in the occupancy relation, wherein each tuple records the probability that a given individual is present at a certain location for a certain duration. While probabilities are modeled as a real number in our data model, time is modeled discretely reflecting the discrete nature of the underlying events. We formulated probabilistic queries in SQL and CLP(R) to show how information can be retrieved and illustrate the results for some of the queries in a graphical manner. The state transition system model provides a natural basis for keeping track of the effect of various events that occur in the smart space. The states in turn serve as an effective basis for the retrieval of answers in response to various queries about the whereabouts of occupants in the smart space.

This paper makes two important contributions in spatio-temporal retrieval of information from a smart space:

1. We show different ways of presenting the result of spatio-temporal queries. While the tabular form of presenting views of relational data is standard and well-known, we show the usefulness of graphs and sequence diagrams in this paper. Sequence diagrams are especially useful in presenting the movements of people across zones of the smart space. When the results involve probabilities, we can also use graphs to report results, such as the probability distribution for the occupancy of an individual across various zones at a particular time.
2. We show how the performance of a smart space can be defined using precision-recall and tailored to the needs of information retrieval. In our earlier papers (Menon et al, 2010, 2011, 2012a), we presented precision-recall metrics for a smart space in a query-independent manner. Such an approach takes every event into account, whether or not it is pertinent to a particular query. Query-based metrics tend to provide a more realistic appraisal of the performance of a smart space, as it caters to the application at hand. From our experimental results, the smart space tends to provide significantly higher numbers in the precision metric for occupancy-related queries compared with query-independent precision metrics.

We have presented a variety of spatio-temporal queries in this paper. As part of our future work, we propose to investigate a class of interesting queries which may be called *what-if queries*, e.g., 'If A was known to be in the lounge at noon with certainty, what is the probability that B was also present at that time?' This query cannot be answered without re-initializing the state and occupancy relations. If A was in the lounge at noon with certainty it means that A was present with certainty in the zone preceding his entry to the lounge. Inductively, we can say that A was present with certainty in all zones in his track leading up to entrance to the security zone at or preceding noon. Thus, we need to define a revised event sequence in which the probability of A is 1.0 for each event in his track leading up to lounge at noon, and the probabilities of all other occupants for each such event is 0. The above query shows the deep interconnection between retrieval, reasoning, and recognition. A 'what-if' query that declares knowledge about an event causes the redefinition of one or more biometric events, thereby triggering the state transition system to compute a new set of states, which the retrieval system uses to determine a new occupancy relation for answering the query.

References

Adaikkalavan R, Chakravarthy S (2006) SnoopIB: Interval-based Event Specification and Detection for Active Databases. Data and Knowledge Engineering 59(1):139–165

Agrawal S, Chaudhuri S, Das G, Gionis A (2003) Automated Ranking of Database Query Results. In: Proc. of CIDR '03

Amicis R, Conti G, Piffer S, Prandi F (2011) Service Oriented Computing for Ambient Intelligence to Support Management of Transport Infrastructures. Journal of Ambient Intelligence and Humanized Computing 2(3):201–211

Amoretti M, Copelli S, Wientapper F, Furfari F, Lenzi S, Chessa S (2013) Sensor Data Fusion for Activity Monitoring in the PERSONA Ambient Assisted Living Project. Journal of Ambient Intelligence and Humanized Computing 4(1):67–84

Benjelloun O, Sarma AD, Hayworth C, Widom J (2006) An Introduction to ULDBs and the Trio System. IEEE Data Engineering Bulletin 29(1):5–16

Bohlen MH, Busatto R, Jensen CS (1998) Point-versus Interval-based Temporal Data Models. In: Proc. of 14th International Conference on Data Engineering, pp 192–200

de Caluwe R, deTre G, Bordogna G (2004) Spatio-Temporal Databases: Flexible Querying and Reasoning. Springer

Ceri S, Gottlob G, Tanca L (1989) What You Always Wanted to Know About Datalog (And Never Dared to Ask). IEEE Transactions of Knowledge and Data Engineering 1(1):146–166

Cormode G, Garofalakis M (2007) Sketching Probabilistic Data Streams. In: Proc. of SIGMOD '07, pp 281–292

Dalvi N, Re C, Suciu D (2006) Query Evaluation on Probabilistic Databases. IEEE Data Engineering Bulletin 29(1):25–31

Dalvi N, Re C, Suciu D (2009) Probabilistic Databases: Diamonds in the Dirt. Communications of the ACM 52(7):86–94

Das SK, Cook DJ, Battacharya A, Heierman EO, Lin T (2002) The Role of Prediction Algorithms in the MavHome Smart Home Architecture. IEEE Wireless Communications 9(6):77–84

Demers AJ, Gehrke J, Hong M, Riedewald M, White WM (2006) Towards Expressive Publish/Subscribe Systems. In: Proc. of 10th International Conference on Extending Database Technology (EBDT '06), pp 627–644

Deshpande A, Guestrin C, Madden S, Hellerstein JM, Hong W (2004) Model-driven Data Acquisition in Sensor Networks. In: Proc. of VLDB '04, pp 588–599

Guting R, Schneider M (2005) Moving Object Databases. Morgan Kaufmann

Jaffar J, Michaylov S, Stuckey P, Yap R (1992) The CLP(R) Language and System. ACM Transactions on Programming Languages and Systems (TOPLAS) 14(3):339–395

Jayram TS, Kale S, Vee E (2007) Efficient Aggregation Algorithms for Probabilistic Data. In: Proc. of Symposium on Discrete Algorithm (SODA '07), pp 346–355

McCarthy JF, Anagnost TD (2000) EVENTMANAGER: Support for the Peripheral Awareness of Events. In: Proc. of 2nd International Conference on Handheld and Ubiquitous Computing (HUC '00), pp 67–96

Menon V, Jayaraman B, Govindaraju V (2008) Biometrics Driven Smart Environments: Abstract Framework and Evaluation. In: Proc. of 5th International Conference on Ubiquitous Intelligence and Computing (UIC-08), Springer-Verlag, pp 75–89

Menon V, Jayaraman B, Govindaraju V (2010) Multimodal Identification and Tracking in Smart Environments. Journal of Personal and Ubiquitous Computing 14(8):685–694

Menon V, Jayaraman B, Govindaraju V (2011) The Three Rs of Cyberphysical Spaces. Computer 44(9):73–79

Menon V, Jayaraman B, Govindaraju V (2012a) Enhancing Biometric Recognition with Spatio-temporal Reasoning in Smart Environments. Journal of Personal and Ubiquitous Computing URL http://dx.doi.org/10.1007/s00779-012-0546-x

Menon V, Jayaraman B, Govindaraju V (2012b) Spatio-Temporal Querying in Smart Spaces. In: Proc. of 3rd International Conference on Ambient Systems, Networks and Technologies (ANT-2012), Elsevier Science, pp 366–373

Motro A (1988) Vague: A User Interface to Relational Databases that Permits Vague Queries. ACM Transactions on Information Systems 6(3):187–214

Papandrea M, Giordano S (2013) Location Prediction and Mobility Modelling for Enhanced Localization Solution. Journal of Ambient Intelligence and Humanized Computing DOI 10.1007/s12652-013-0175-x, URL http://link.springer.com/article/10.1007/s12652-013-0175-x

Pelekis N, Frentzos E, Giatrakos N, Theodoridis Y (2008) HERMES: Aggregative LBS via a Trajectory DB Engine. In: Proc. of 28th International Conference on Management of Data (SIGMOD 08)

Re C, Suciu D (2007) Materialized Views in Probabilistic Databases for Information Exchange and Query Optimization. In: Proc. of VLDB '07, pp 51–62

Re C, Dalvi N, Suciu D (2007) Efficient Top-k Query Evaluation on Probabilistic Data. In: Proc. of ICDE '07, pp 886–895

Re C, Letchner J, Balazinksa M, Suciu D (2008) Event Queries on Correlated Probabilistic Streams. In: Proc. of ACM SIGMOD International Conference on Management of Data, pp 715–728

Snodgrass R (1987) The Temporal Query Language TQuel. ACM Trans Database Syst 12(2):247–298

Toman D (1996) Point vs. Interval-based Query Languages for Temporal Databases. In: Proc.of 15th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS), pp 58–67

Wu E, Diao Y, Rizvi S (2006) High-performance Complex Event Processing over Streams. In: Proc. of ACM SIGMOD' 06, pp 407–418