



Open Archive TOULOUSE Archive Ouverte (OATAO)

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible.

This is an author-deposited version published in : <http://oatao.univ-toulouse.fr/>
Eprints ID : 18882

The contribution was presented at SAGEO 2016 :

<https://sageo2016.sciencesconf.org/>

To link to this article URL :

<https://hal.archives-ouvertes.fr/hal-01424734/document>

To cite this version : Jeveme Panta, Franck and Sèdes, Florence *Interrogation des données spatio-temporelles de géolocalisation indoor à partir des trajectoires hybrides*. (2016) In: Conférence internationale francophone Spatial Analytics and GEomatics (SAGEO 2016), 6 December 2016 - 9 December 2016 (Nice, France).

Any correspondence concerning this service should be sent to the repository administrator: staff-oatao@listes-diff.inp-toulouse.fr

Interrogation de données spatio-temporelles de géolocalisation indoor à partir de trajectoires hybrides

Franck Jeveme Panta, Florence Sèdes

IRIT, Université Paul Sabatier Toulouse 3
franck.panta@irit.fr, florence.sedes@irit.fr

RESUME. Le GPS (Global Positioning System) basé sur la visibilité directe entre le satellite et le récepteur, s'est imposé pour la localisation outdoor. En l'absence de signal GPS, la localisation d'une cible dans les environnements indoor peut se faire avec un réseau de capteurs. Ces réseaux déployés dans les bâtiments sont de plus en plus nombreux et servent à plusieurs applications basées sur la localisation : surveillance, détection, navigation, etc. Ces capteurs de localisation indoor génèrent une quantité importante d'informations de tracking. Le contexte d'application de ce travail est celui des systèmes de localisation indoor basés sur les cellules Wifi, sur les capteurs ICCARD et sur les caméras de vidéosurveillance. Dans un tel contexte, il n'existe aucun système global de référence similaire au GPS : les informations de localisation sont hétérogènes (positions géométriques et symboliques / multitude de systèmes de référence). Cet article présente un ensemble de démarches qui ont permis de concevoir et implémenter un framework utilisant les informations générées par les réseaux de capteurs de localisation déployés dans un environnement indoor, illustré dans le cadre du forensic.

ABSTRACT. GPS has become the reference for Outdoor positioning, where a direct uplink between the GPS satellite and a receiver device can be established. In the absence of GPS signal for Indoor positioning, the location of a target device requires the use of wireless sensors network. Nowadays sensor networks deployed in buildings are becoming more and more used for many applications based on the location: surveillance, detection, navigation, etc. These indoor locating sensors generate an important number of tracking information. The application context of this paper is indoor locations systems based on wireless cell, ICCARD sensors and video surveillance. In this context, there is no global reference system similar to GPS, the location information is heterogeneous (geometric and symbolic positions / various references systems). This paper presents sequences of steps which help to design and implement a framework using the information generated by the location sensor networks deployed in an indoor environment, , illustrated in the forensic.

MOTS-CLES : modèle de localisation indoor, dataset, requêtes spatio-temporelles, requête hybride.

KEYWORDS: indoor location model, dataset, spatio-temporal queries, hybrid query.

1. Introduction

La mise à disposition de l'utilisateur, via des requêtes spatio-temporelles, de données géo-localisées, basées sur la captation via des réseaux de capteurs, vise à localiser, identifier, extraire les métadonnées des capteurs, dispositifs ou objets connectés situés dans une région donnée, à une certaine distance d'autres objets ou capteurs, à l'intersection de trajectoires, etc. de telles applications sont basées sur la position de l'objet, qui peut être exprimée de deux façons, selon différents systèmes de référence : géométrique (coordonnées par rapport à un système de référence comme le système géodésique ou le plan d'un bâtiment) et symbolique (description plus sémantique par rapport à des points d'intérêt, adresse postale, réseaux de transport, pièces d'un bâtiment, etc.). Au moment du traitement de ces positions, traitement qui consiste à exprimer les prédicats par rapport à une référence commune, la principale différence entre les deux types d'expression des positions est la précision, une position géométrique pouvant être exprimée par un point tandis qu'une position symbolique est le plus souvent représentée par une géométrie (d'une pièce, d'un bâtiment ou d'une cellule RFID).

Dans les environnements outdoor, les applications utilisent dans la plupart des cas des modèles de localisation basés sur des capteurs GPS qui sont largement présents dans les systèmes embarqués (voitures, smartphones, etc.). Les environnements indoor sont de plus en plus complexes et il n'existe pas de système de référence unique auquel toute mesure peut se rapporter : les bâtiments des entreprises, infrastructures des métros, des aéroports, etc., génèrent un besoin important de gérer finement la localisation indoor en termes d'applications, sans qu'il n'existe encore de réponse satisfaisante et générique.

Les travaux présentés dans ce papier s'appuient sur les recherches menées dans notre équipe sur les problèmes de localisation outdoor, dans le cadre du forensic (Sedes et al. 2012, Codreanu 2015) et proposent une extension dans les environnements indoor avec des capteurs (appropriés) tels que bornes WiFi, lecteurs ICCARD et caméras de vidéosurveillance. Le problème principal dans le contexte indoor est l'hétérogénéité de l'information de localisation donnée par les différents capteurs. Cette hétérogénéité concerne les cas suivants :

- les positions sont soit géométriques soit symboliques: tandis que la localisation basée sur le wifi donne des positions géométriques par rapport au système de coordonnées à 2 dimensions, la localisation basée sur les cellules dans le cas du réseau cellulaire donne des positions symboliques, ce qui est aussi le cas avec les technologies telles que l'infrarouge (IR) et le RFID (Radio Frequency Identification) ;
- les positions sont exprimées par rapport à différents systèmes de coordonnées ou systèmes de référence.

Notre travail porte principalement sur ces deux points et consiste à développer une solution générique de modélisation et un framework de requête de trajectoire d'objets en environnement indoor.

La suite de ce papier est structurée comme suit : après un examen des travaux connexes concernant les modèles de localisation en indoor, les trajectoires et les types de requêtes spatiales dans la section 2, la section 3 présente la définition d'un modèle de requête trajectoire hybride en environnement indoor. Les différents algorithmes de recherche sont présentés à la section 4. A la section 5, l'architecture proposée pour le framework est présentée et enfin, la section 6 conclut et discute les recherches futures.

2. Etat de l'art

2.1. Les modèles de localisation en indoor

La localisation indoor vise à définir l'emplacement d'un utilisateur/objet dans un environnement intérieur. Un environnement indoor est en général constitué par un espace en trois dimensions (2D + étage par exemple) : ainsi, un édifice dans lequel vivent, travaillent, transitent, etc. des personnes passent habituellement leur temps (maisons, centres commerciaux, entreprises, gares, etc.) (Li et al., 2008). Pour représenter cet espace, les modèles spatiaux reposent sur la définition d'un modèle de topologie qui traduit l'espace, la définition de l'emplacement des objets et les trajectoires. Cette représentation est appelée modèle de localisation. (Leonhardt et U., 1998) définissent un modèle de localisation comme une représentation expressive, souple et efficace des informations de localisation.

Il existe deux classes de base de modèles de localisation :

- les modèles de localisation géométriques, dans lesquels les positions sont représentées par des points (ex : les points $P_1(x_1, y_1)$ et $P_2(x_2, y_2)$) (Parent et al., 2013) ;
- les modèles de localisations symboliques : dans ce cas les positions sont désignées par des noms ou symboles abstraits (ex: "Salle C1", "Salle C3") (Jensen et al., 2009).

Les modèles hybrides sont des combinaisons de modèles géométriques et symboliques (C. Becker et F. Darr, 2005). Nous nous intéressons aussi à la variation des positions dans le temps, C'est dans ce même ordre idée que nous présentons dans ce qui suit une revue de la littérature sur les trajectoires.

2.2. Définition des trajectoires

Une trajectoire est définie comme le changement de la position d'un objet mobile dans le temps. Une trajectoire peut être segmentée en fonction de plusieurs critères. Le segment d'une trajectoire est défini comme une sous-séquence maximale

de la trajectoire de telle façon que toutes les positions spatio-temporelles soient conformes à un prédicat donné (ex : direction constante) (Mountain et Raper, 2001).

(D. Codreanu, 2015) donne une définition formelle et générique d'une trajectoire et d'un segment de trajectoire. Une trajectoire (voir l'équation 1) est définie comme un tuple composé d'un identifiant de trajectoire, un identifiant d'objet, un ensemble de segments de trajectoires et un intervalle de temps global qui marque le "start" et le "end" de la trajectoire.

$$Tr = (tr_{id}, obj_{id}, \{u_k\}, [t_{start}, t_{end}]) \quad (1)$$

$$u_k = (u_{id}, \{position_i\} / 1 \leq i, [t_{kstart}, t_{kend}]) \quad (2)$$

Un segment (voir l'équation 2) de trajectoire est un tuple qui contient un identifiant de segment, un ensemble de positions homogènes et un intervalle de temps qui est optionnel.

Le mouvement des objets peut être effectué de façon libre (ex : le déplacement de l'aire) (Parent et al., 2006) ou contraint par exemple par l'architecture d'un bâtiment (Jensen et al., 2009) (ex : le mouvement d'une personne à l'intérieur d'une station de métro est contraint par la disposition des différents espaces et leurs connexions).

Dans ce qui suit, nous présentons les types de requêtes spatiales envisageables sur une trajectoire en environnement indoor.

2.3. Requêtes spatiales

Une requête spatiale est toute interrogation de l'utilisateur comportant au moins un élément de géolocalisation ou de situation spatiale (exprimée en termes de coordonnées géométriques ou de noms symboliques) et/ou une référence temporelle (date, heure, intervalle) (Ilarri et al., 2010). L'utilité de requêtes spatiales concernant la trajectoire passée des objets et les requêtes prédictives a été prouvée par une grande quantité de recherches et travaux commerciaux. Il existe plusieurs travaux concernant les requêtes spatiales comme : les requêtes de position (Becker et Durr, 2005) : retournant la position d'un objet mobile ou statique (position géométrique ou symbolique), les requêtes de région (Gu et al., 2009 ; Ay et al., 2008 ; Scuturici et Ejigu, 2006 ; Yuan et Schneider, 2010) : «quelles sont les pharmacies situées dans cette zone ?», les requêtes des voisins les plus proches (Zhang et al., 2003) : «quelles sont les maisons les plus proches du carrefour ?», les requêtes de navigation ou de trajectoire (Afyouni et al., 2012b): « quel est le plus court chemin pour arriver à la maison ?».

La plupart des ouvrages traitant les trajectoires d'objets visent à utiliser les anciennes trajectoires pour déceler les déplacements habituels et donc à prédire à l'avenir la position des objets. Moins de travaux considèrent la trajectoire orientée vers les requêtes de reconstitution et à notre connaissance, il n'y a pas de framework

de requête traitant les trajectoires hétérogènes (contenant à la fois des segments symboliques et géométriques) en environnement indoor. Dans ce qui suit, nous allons présenter un cas d'utilisation qui montre l'utilité d'une telle requête.

Supposons un bâtiment "intelligent" dans lequel chaque employé dispose d'un smartphone suivi par des systèmes de positionnement basés sur différents capteurs de localisation. D'autres types de capteurs surveillent la consommation d'énergie et la température partout dans le bâtiment.

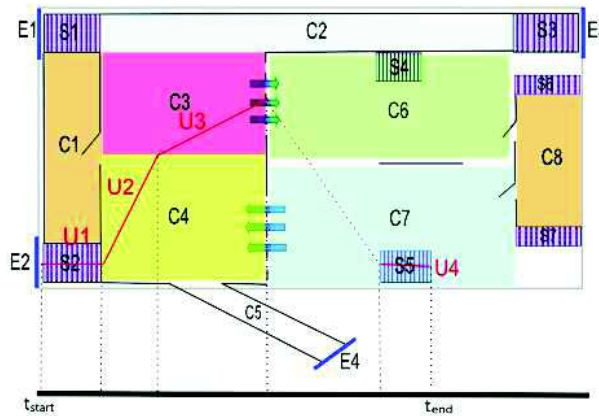


FIGURE 1. Exemple de trajectoire 1

La Figure 1 illustre la trajectoire d'une personne dans le bâtiment. La trajectoire est composée de quatre segments (u_1 , u_2 , u_3 et u_4). Il pourrait y avoir un manque d'information de position pour une certaine période de temps (une personne a fermé son smartphone et aucune information n'existe sur sa position). La ligne pointillée représente une lacune dans la définition de la trajectoire (appelé «trou») qui peut être "remplie" à l'aide des fonctions d'interpolation (Vazirgiannis et Wolfson, 2001). Sur la base de cette trajectoire, on veut trouver les différents opérateurs comme la moyenne de la consommation d'énergie, l'identification des capteurs situés sur la trajectoire ou à une certaine distance de la trajectoire.

Considérons un autre cas, tel un incident dans une zone de l'édifice. Une personne se déplace à l'intérieur d'une zone interdite. La génération de ses positions est basée sur la détection de son smartphone par les cellules WiFi environnantes. Ceci génère des coordonnées (x, y) au regard du plan du bâtiment. À un certain point, le signal est perdu et la seule information que nous avons est qu'il est entré dans une salle sécurisée (son badge a été validé à l'entrée de la salle). Dans ce cas, la trajectoire de la personne ressemble à celle de la Figure 2. Les deux premiers segments de la trajectoire sont u_1 et u_2 , il y a une période où l'objet a été perdu et la dernière partie de la trajectoire est u_3 qui est la chambre C1. Dans ce cas, nous

voulons localiser et identifier les caméras vidéo situées sur le parcours donné afin de visualiser les enregistrements qu'elles ont pu réaliser.

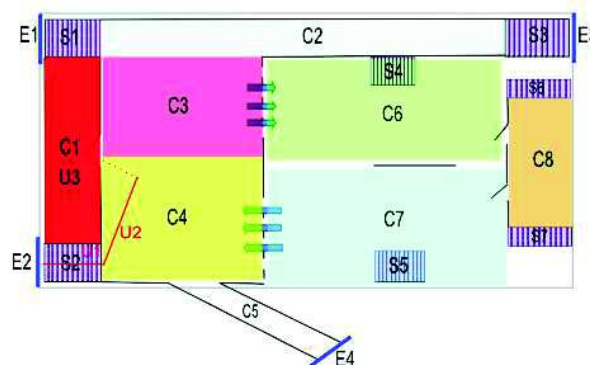


FIGURE 2. Exemple de trajectoire 2

Nous constatons dans ces deux cas que nous avons à faire à des trajectoires hétérogènes c'est-à-dire des trajectoires constituées de multiples positions (géométriques et symboliques) ayant des systèmes de coordonnées différents.

3. Définition d'un modèle de requête trajectoire hybride en environnement indoor

Selon les étapes de notre stratégie, nous allons commencer par définir un modèle de requête que le système va pouvoir supporter en se basant sur les deux exemples de cas d'utilisation présentés dans la section 2.3. L'innovation que nous introduisons dans la définition d'une trajectoire est la prise en compte du fait que chaque position est exprimée par rapport à un système de référence (ex : plan du bâtiment, réseau de lecteur ICCARD). Les positions sont homogènes à l'intérieur du segment (exprimées par rapport à un seul système de référence) mais les différents segments peuvent contenir des positions hétérogènes comme dans les exemples présentés à la section 2.3.

On définit une requête trajectoire hybride comme un tuple illustré ci-dessous :

$$T_r = (tr_{id}, \{u_k\}, [t_{start}, t_{end}]) \quad (3)$$

où tr_{id} est l'identifiant de la trajectoire, $\{u_k\}$, est l'ensemble des segments spatiaux constituant la trajectoire et $[t_{start}, t_{end}]$ représente l'intervalle de temps de la requête trajectoire.

La définition d'un segment de trajectoire est la suivante :

Interrogation de données spatio-temporelles

$$u_k = (u_{id}, refS_{id}, \{position_i\}) \quad (4)$$

où u_{id} est l'identifiant du segment, $refS_{id}$ est l'identifiant du système de référence par rapport auquel les positions sont définies et $position_i$ est l'ensemble des positions qui composent le segment de trajectoire.

Pour pouvoir gérer l'hétérogénéité des segments de trajectoire nous définissons la position d'un objet comme étant composée de deux grandes parties : une partie désignant le système de référence et une autre partie qui désigne la localisation par rapport au système de référence. L'idée va dans le même sens que celle proposée par (Jiang et Steenkiste, 2002) qui définissent la syntaxe d'une localisation indoor hybride basée sur la syntaxe d'un URI (Universal Resource Identifier).

Une requête trajectoire hybride est composée d'une partie spatiale et d'une partie temporelle. La partie spatiale est constituée d'une séquence de segments, chaque segment étant constitué d'une partie désignant le système de référence et une séquence de positions (géométriques ou symboliques) exprimées par rapport au système de référence correspondant. La Figure 3 présente un exemple de trajectoire hybride avec les différents types de position, de systèmes de référence et l'intervalle de temps de la requête.

```
{
  "query": {
    "spatial": [
      {
        "Floor8": [
          [128.53813, 368.34243],
          [190.89598, 357.04869],
          [140.32508, 345.56478],
          [136.62363, 322.94129],
          [153.10012, 305.00549],
          [172.11112, 288.10549]
        ]
      },
      {
        "ICCARD": ["0000110015010003"]
      }
    ],
    "temporal": {
      "start": "14-02-14 04:05:34",
      "end": "14-02-2014 05:50:10"
    }
  }
}
```

FIGURE 3. Exemple de requête trajectoire hybride

C'est cette requête qui donne la trajectoire suivie par le suspect. Les informations spatio-temporelles générées par les différents capteurs et sauvegardées seront interrogées par cette trajectoire pour déterminer les objets (personnes, dispositifs,

etc.) se trouvant sur celle-ci, ainsi que les caméras de vidéosurveillance susceptibles d'avoir filmé cette trajectoire pendant l'intervalle de temps de la requête.

Dans la section suivante, nous présentons les différents algorithmes de recherche des positions et des caméras.

4. Algorithmes de recherche

Le point d'entrée de notre système est une requête trajectoire hybride, constituée des positions géométriques et symboliques exprimées par rapport à différents systèmes de référence et d'un intervalle de temps. Une fois que cette requête est interprétée par un module interpréteur de requête, elle est exprimée comme un ensemble de points (p_1, p_2, \dots, p_n) et un intervalle de temps $[t_1, t_2]$. Ces points forment une séquence de segments spatiaux (u_1, u_2, \dots, u_n) pour ce qui concerne les positions géométriques et des cercles de centre (c_1, c_2, \dots, c_n) pour les différentes positions symboliques. Cette séquence de points avec l'information temporelle constitue l'entrée des algorithmes de recherche. Le but est :

- de proposer un ensemble d'objets (personne, dispositif, etc.) dont les positions ont «intersecté» la trajectoire donnée par la requête dans l'intervalle de temps $[t_1, t_2]$;
- de proposer un ensemble d'objets se trouvant dans la zone de couverture (portée) de chaque lecteur ICCARD de la requête trajectoire dans l'intervalle de temps $[t_1, t_2]$;
- de proposer une liste des caméras dont le champ de vue (qui peut être variable) a «intersecté» la trajectoire de la requête dans l'intervalle de temps.

4.1. Recherche des positions des objets liées aux bornes WiFi

A partir d'une trajectoire (séquence de segments u_k) donnée comme requête, il s'agit de retrouver les objets qui se trouvent « sur » cette trajectoire. Nous allons prendre en compte une incertitude car les points donnés dans la requête trajectoire peuvent ne pas être exacts (Alboody et al. 2009). Pour cela nous allons tracer la trajectoire (Poly-ligne) avec les points donnés dans la requête, puis effectuer une translation d'un côté comme de l'autre de cette trajectoire dans le but de construire une géométrie. Par la suite nous allons « intersecter » cette géométrie par l'ensemble de données (positions) WiFi enregistrées, qui va identifier l'ensemble des objets qui se trouvent sur la trajectoire. Rappelons qu'il faut « intersecter » la géométrie avec des données WiFi (positions des objets) dont le temps (DATETIME) est compris dans l'intervalle $[t_1, t_2]$ donné dans la requête.

Opérateurs :

Le résultat est l'ensemble des doublets: $R=\{r=(obj_i, pos_j)\}$; la fonction *construire_geometry*(u_k) permet de construire une géométrie à partir de la poly-ligne formée par les segments u_k ; la fonction *trouverObjet*($[t_1, t_2]$) renvoie la liste

des objets dont le temps (DATETIME) est compris dans l'intervalle $[t_1, t_2]$ ainsi que les positions correspondantes.

Algorithme :

Algorithme 1 : Algorithme de recherche des objets dont les positions ont intersecté la trajectoire de la requête

Entrées : une suite de segments de trajectoire : u_k et un intervalle de temps : $[t_1, t_2]$.
Sorties : la liste des objets qui ont intersecté la trajectoire et les positions correspondantes.

```

1:  $geometry \leftarrow construire\_geometry(u_k)$ 
2:  $listeObjet(obj_i, pos_i) \leftarrow trouverObjet([t_1, t_2])$ 
3: pour chaque  $obj_i$  de  $listeObjet(obj_i, pos_i)$  faire
4:     si  $pos_i$  intersecte  $geometry$  alors
5:          $ajouterResultat(obj_i, pos_i)$ 
6:     fin condition
7: fin boucle

```

4.2. Recherche des positions des objets liées aux lecteurs ICCARD

Dans ce cas, nous avons à faire à des positions symboliques qui sont des surfaces précises dans le bâtiment. Chaque lecteur ICCARD a un champ d'action ou bien une portée. Une fois la requête trajectoire passée dans le module interpréteur de requête, les noms des lecteurs sont représentés sur le plan du bâtiment par des points (entrées de l'algorithme) qui sont les positions affectées aux lecteurs ; à partir de ces points, sont construites des géométries. Ces géométries sont des cercles dont le diamètre est la portée de chaque lecteur. Le rayon de ces cercles est paramétrable. Les objets recherchés se trouvent donc dans la surface de ces cercles. On recherche donc ici les objets contenus dans les géométries construites.

Opérateurs :

Le résultat pour les lecteurs ICCARD est l'ensemble des doublets : $R=\{r(obj_i, pos_i)\}$; la fonction *construire_cercle* ($p_n, rayon_i$) permet de construire des cercles dont les centres sont les p_n et les rayons sont les $rayon_i$; la fonction *trouverObjet* ($[t_1, t_2]$) renvoie la liste des objets se trouvant dans la base de données dont le temps (DATETIME) est compris dans l'intervalle $[t_1, t_2]$ ainsi que les positions correspondantes.

Algorithme :

Algorithme 2 : Algorithme de recherche des objets contenus dans les géométries

Entrées : une suite de points : p_n et un intervalle de temps : $[t_1, t_2]$.
Sorties : la liste des objets contenus dans les géométries et les positions correspondantes.

```

1:  $listeObjet(obj_i, pos_i) \leftarrow trouverObjet([t_1, t_2])$ 
2: pour chaque  $p_n$  de la requête faire
3:      $geometry_i \leftarrow construire\_cercle(p_n, rayon_i)$ 
4:     pour chaque  $obj_i$  de  $listeObjet(obj_i, pos_i)$  faire
5:         si  $obj_i$  contenu dans  $geometry_i$  alors
6:              $ajouterResultat(obj_i, pos_i)$ 

```

```
7:      fin condition
8:      fin boucle
9:      fin boucle
```

4.3. Algorithme de sélection des caméras

(D. Codreanu, 2015) propose un algorithme de sélection des caméras fixes. Notre travail étant une extension en environnement indoor du sien, nous allons adapter cet algorithme dans ce cas.

Le but est de proposer à l'opérateur une liste de caméras susceptibles d'avoir filmé les segments de la trajectoire donnée comme requête. Pour chaque caméra on doit donner l'intervalle de temps pendant lequel elle a filmé la trajectoire.

L'algorithme se déroule en deux étapes, l'étape de filtrage et l'étape d'affinage des résultats :

- filtrage : on sélectionne les caméras qui se trouvent à une distance inférieure à la distance de visibilité maximale des segments de la trajectoire ;
- affinage : pour les caméras sélectionnées dans l'étape de filtrage, on calcule la géométrie des champs de vue, et on choisit celles dont les géométries touchent les segments de la trajectoire.

Le résultat est l'ensemble des triplets $R=\{r=(c_i, u_k, [t_a, t_b])\}$ où c_i est la caméra sélectionnée, u_k est le segment filmé et $[t_a, t_b]$ l'intervalle de temps pendant lequel la caméra a filmé le segment de la trajectoire.

Ces algorithmes sont implémentés dans le module correspondant au moteur de recherche de l'architecture du framework développé.

Dans la section suivante nous présentons cette architecture.

5. Architecture du framework et mise-en-oeuvre

Le framework développé implémente le modèle de requête trajectoire hybride et les algorithmes de recherche présentés dans les sections précédentes. L'implémentation du prototype a été réalisée en utilisant JDK 1.7 avec les drivers JDBC pour la connexion avec la base de données Oracle.

Les principaux modules constituant cette architecture sont :

- **Interface utilisateur** : constituée des différents blocs qui permettent l'interaction avec l'utilisateur, la visualisation des données et des résultats ;
- **Interpréteur de requête** : effectue la transformation de la requête trajectoire hybride soumise par l'utilisateur en requête spatio-temporelle (un ensemble de points 2D exprimés par rapport au plan du bâtiment formant des séquences de segments et/ou des géométries et un intervalle de temps) ;
- **Moteur de recherche** : implémente les algorithmes de recherche définis.

Interrogation de données spatio-temporelles

- **Stockage** : contient la base de données spatiale, la collection de données ainsi que des modules qui permettent l'interaction avec celles-ci ;
- **Générateur de requête SQL** : réalise la communication entre le moteur de recherche et le module de stockage ;
- **Collecte des données** : collecte les données générées par les différents capteurs de localisation et les organise dans un format donné.

Les relations entre les différents modules sont présentées à la Figure 4.

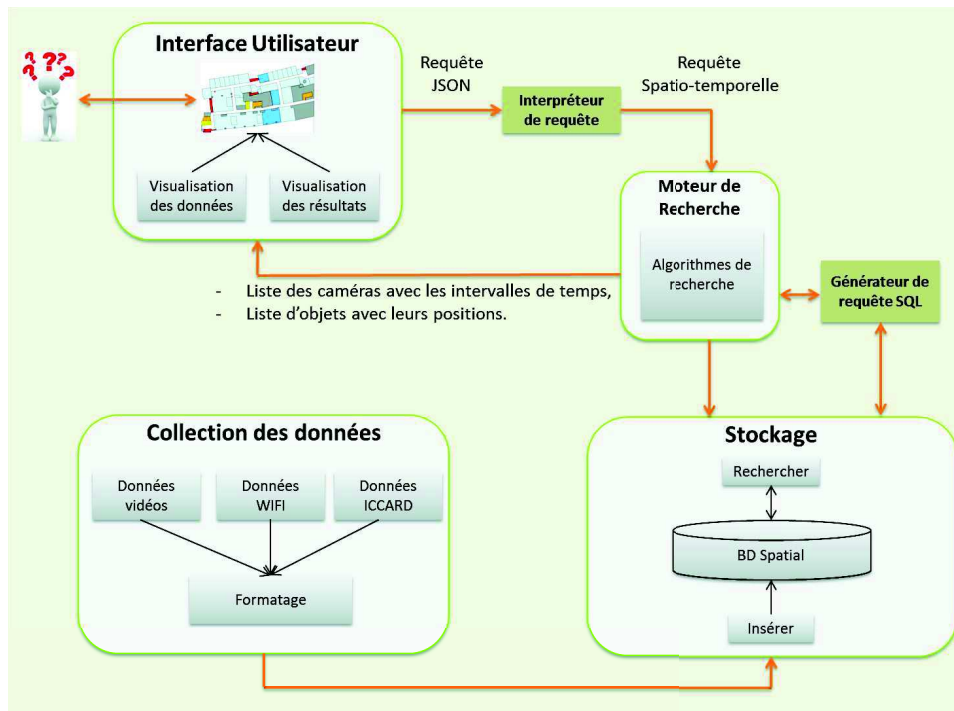


FIGURE 4. Architecture du framework

L'objectif est de mettre en œuvre la recherche d'objets via requête : recherche d'objets se trouvant sur une trajectoire donnée comme requête, d'objets se trouvant à la portée des lecteurs ICCARD, de caméras susceptibles d'avoir filmé la trajectoire pendant l'intervalle de temps donné par la requête.

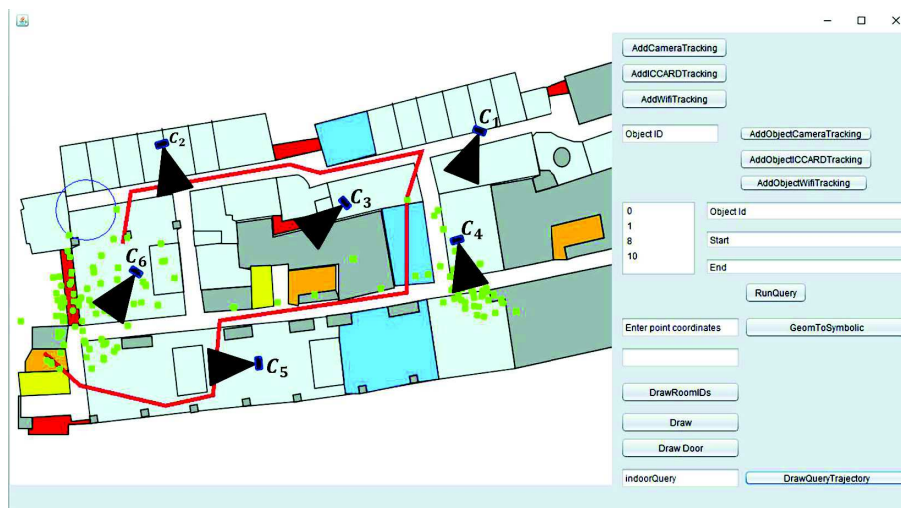


FIGURE 6. Application

L'interprétation de la requête de la Figure 3 est détaillée Figure 6 : la trajectoire (poly-ligne en rouge) représente le premier segment de la requête qui est constitué de l'ensemble des positions géométriques ([45;425], [92;468], [206;495], [269;481], [278;382], [526;345], [526;221], [546;158], [411;186], [371;175]) ; le cercle en bleu représente la portée du seul lecteur ICCARD ("0000110015010003") présent dans la requête. Les points en vert sont les positions des personnes générées par les bornes wifi et sauvegardées dans la base de données Oracle.

Les personnes qui nous intéressent sont celles qui se trouvent sur la trajectoire qui est la poly-ligne en rouge (en l'occurrence 6) et celles qui se trouvent à l'intérieur ou sur le cercle en bleu (en l'occurrence 2). Les caméras qui nous intéressent sont celles dont le champ de vue a «intersecté» la trajectoire (en l'occurrence C_2 et C_5).

L'implémentation des requêtes dans Oracle Spatial permet de visualiser ces résultats en retournant la liste des identifiants des personnes et leurs positions, la sélection des caméras qui ont «intersecté» les segments de la trajectoire ainsi que les intervalles de temps pendant lesquels ces champs de vue ont « intersecté » la trajectoire.

6. Conclusion

Nous avons présenté dans cet article notre contribution dans le domaine de la localisation indoor, comme une extension des précédents travaux sur la construction de trajectoires. Nous définissons un système de coordonnées en indoor qui peut être associé au plan du bâtiment, à un étage ou à chaque espace. Les positions sont

exprimées en utilisant les coordonnées 2D au regard du plan de l'étage, une troisième coordonnée est associée à chaque position dans le but d'indiquer le numéro de l'étage par exemple. Nous avons proposé un modèle de requête trajectoire hybride constitué des positions géométriques et symboliques exprimées par rapport à différents systèmes de référence. Les algorithmes développés permettent de rechercher les objets (personnes, dispositifs, etc.) se trouvant sur une trajectoire donnée comme requête sur un intervalle temporel donné, et de rechercher les caméras susceptibles d'avoir filmé cette trajectoire.

Notre travail peut s'instancier dans le contexte du forensic, par ex. lors d'une enquête (Codreanu et al., 2015). Nous envisageons d'étendre ce contexte afin d'être en mesure de traiter les requêtes en temps réel ou de prédire les trajectoires sur la base des statistiques réalisées sur des données stockées. Pour les systèmes de localisation indoor basés sur les caméras de vidéosurveillance, le résultat de l'opérateur proposé est une liste de caméras dont les intervalles de temps peuvent se recouvrir. Une perspective concerne donc l'ordonnement des résultats en fonction de la distance des caméras par rapport aux segments de la requête par exemple. Dans nos algorithmes, il est question de rechercher des relations topologiques entre les objets et les géométries. La question de la recherche de relations topologiques entre les régions de l'espace est d'actualité, surtout entre les régions de l'espace avec des incertitudes. Dans le traitement d'une requête, il faudra prendre en compte l'incertitude (fuzzy temporel et fuzzy spatial) (Sèdes et al., 2010).

Ces contributions ont permis de mettre sur pied les bases du développement d'un framework dont l'architecture a été présentée et qui peut être considéré comme un outil d'aide au requêtage hybride.

Bibliographie et références

- Afyouni, I., Ray, C., Ilarri, S., and Claramunt, C. (2012b). *Algorithms for continuous location-dependent and context-aware queries in indoor environments*. 20th Intl Conf on Advances in Geographic Information Systems, ACM SIGSPATIAL '12, pp 329-338.
- Ahed Alboody, Jordi Inglada, Florence Sèdes (2009) *Enriching the spatial reasoning system RCC8*. SIGSPATIAL Special 1(1): 14-20.
- Arslan Ay, S., Zimmermann, R., and Kim, S. H. (2008). *Viewable scene modeling for geospatial video search*. 16th ACM Intl Conf on Multimedia, MM '08, pp 309-318.
- Becker, C. and Durr, F. (2005). *On location models for ubiquitous computing*. Personal Ubiquitous Comput. 9(1) :20-31.
- C. Becker and F. Durr (2005). "On location models for ubiquitous computing," Personal and Ubiquitous Computing, vol. 9, pp. 20. [Online]. Available: <http://dx.doi.org/10.1007/s00779-004-0270-2>.
- D. Codreanu (2015). *Modélisation des métadonnées spatio-temporelles associées aux contenus vidéo et interrogation de ces métadonnées à partir des trajectoires hybrides : Application dans le contexte de la vidéosurveillance*. Thèse de doctorat, Université Paul Sabatier, mai 2015.

SAGEO'2016 – Nice, 6-9 décembre 2016

Dana Codreanu, André Péninou, Florence Sèdes, *Video Spatio-Temporal Filtering Based on Cameras and Target Objects Trajectories - Videosurveillance Forensic Framework*. ARES 2015: 611-617.

Gu, Y., Yu, G., Guo, N., and Chen, Y. (2009). *Probabilistic moving range query over rfid spatio-temporal data streams*. ACM CIKM '09, pp 1413-1416.

Ilarri, S., Mena, E., and Illarramendi, A. (2010). *Location dependent query processing : Where we are and where we are heading*. ACM Comput. Surv., 42(3) :12 :1–12 :73.

Jensen, C. S., Lu, H., and Yang, B. (2009). *Graph model based indoor tracking*. In Mobile Data Management: Systems, Services and Middleware, 2009. MDM '09. pp 122-131.

Jiang, C. and Steenkiste, P. (2002). *A hybrid location model with a computable location identifier for ubiquitous computing*. In Proceedings of the 4th International Conference on Ubiquitous Computing, UbiComp'02, pages 246–263, London, UK, UK. Springer-Verlag.

Leonhardt, U. (2008). *Supporting Location-Awareness in Open Distributed Systems*. PhD thesis, at Department of Computing, Imperial College of Science, Technology and Medicine, University of London (1998). pp 43-65.

Li, K.-J (2008). *Indoor space: A new notion of space*. In Proc. 8th International Symposium on Web and Wireless Geographical Information Systems (W2GIS) (2008), pp. 1–3. LoI: 10.1007/978-3-540-89903-71.

Mountain, D. and Raper, J. (2001). *Modelling human spatio-temporal behaviour: a challenge for location based services*. 6th International Conference on GeoComputation, pp. 2-9.

Parent, C., Spaccapietra, S., and Zim'anyi, E. (2006). *Conceptual Modeling for Traditional and Spatio-Temporal Applications : The MADS Approach*. Springer-Verlag.

Parent, C., Spaccapietra, S., Renso, C., Andrienko, G., Andrienko, N., Bogorny, V., Damiani, M. L., Gkoulalas-Divanis, A., Macedo, J., Pelekis, N., Theodoridis, Y., and Yan, Z. (2013). *Semantic trajectories modeling and analysis*. ACM Comput. Surv., 45(4):42:1-42:32.

Scuturici, V.-M. and Ejigu, D. (2006). *Positioning support in pervasive environments*. In 2006 ACS/IEEE International Conference on Pervasive Services, pp. 19–26.

Florence Sèdes, Ahed Alboody, Jordi Inglada (2010). *Modeling Topological Relations between Uncertain Spatial Regions in Geo-spatial Databases: Uncertain Intersection and Difference Topological Model*. DBKDA 2010, IEEE Computer Society, pp. 56-68.

Sèdes, F., Sulzer, J., Marraud, D., Mulat, C., and Cepas, B. (2012). *Intelligent Video Surveillance Systems, chapter A Posteriori Analysis for Investigative Purposes*. Wiley.

Vazirgiannis, M. and Wolfson, O. (2001). *A spatiotemporal model and language for moving objects on road networks*. In Proceedings of the 7th International Symposium on Advances in Spatial and Temporal Databases, SSTD'01, pages 20–35, London, UK, UK. Springer-Verlag.

Yuan, W. and Schneider, M. (2010). *Supporting continuous range queries in indoor space*. 11th Intl Conf on Mobile Data Management (MDM), pp 209–214. IEEE.

Zhang, J., Zhu, M., Papadias, D., Tao, Y., and Lee, D. L. (2003). *Location-based spatial queries*. 2003 ACM SIGMOD Intl Conf on Management of Data, SIGMOD'03, pp 443–454.