



HAL
open science

Video spatio-temporal filtering based on cameras and target objects trajectories - Videosurveillance forensic framework

Dana Codreanu, André Péninou, Florence Sèdes

► **To cite this version:**

Dana Codreanu, André Péninou, Florence Sèdes. Video spatio-temporal filtering based on cameras and target objects trajectories - Videosurveillance forensic framework. 10th International Conference on Availability, Reliability and Security (ARES 2015), Aug 2015, Toulouse, France. pp.611-617, 10.1109/ARES.2015.102 . hal-01343039

HAL Id: hal-01343039

<https://hal.science/hal-01343039v1>

Submitted on 7 Jul 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



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 : 15375

The contribution was presented at ARES 2015:
<http://www.ares-conference.eu/conference/>

To cite this version : Codreanu, Dana and Péninou, André and Sèdes, Florence
*Video spatio-temporal filtering based on cameras and target objects trajectories -
Videosurveillance forensic framework.* (2015) In: 10 th International Conference
on Availability, Reliability and Security (ARES 2015), 24 August 2015 - 28
August 2015 (Toulouse, France).

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

Video spatio-temporal filtering based on cameras and target objects trajectories - Videosurveillance forensic framework

Dana Codreanu, Andre Peninou, Florence Sedes
Université de Toulouse - IRIT - UMR 5505
118 Route de Narbonne
31062 Toulouse Cedex 9, France
Email: codreanu, peninou, sedes@irit.fr

Abstract—This paper presents our work about assisting videosurveillance agents in the search for particular video scenes of interest in transit network. This work has been developed based on requirements defined within different projects with the French National Police in a forensic goal. The video-surveillance agent inputs a query in the form of a hybrid trajectory (date, time, locations expressed with regards to different reference systems) and potentially some visual descriptions of the scene. The query processing starts with the interpretation of the hybrid trajectory and continues with a selection of a set of cameras likely to have filmed the spatial trajectory. The main contributions of this paper are: (1) a definition of the hybrid trajectory query concept, trajectory that is constituted of geometrical and symbolic segments represented with regards to different reference systems (e.g., geodesic system, road network), (2) a spatio-temporal filtering framework based on a spatio-temporal modeling of the transit network and associated cameras.

Keywords-metadata; spatio-temporal queries; videosurveillance systems; forensic tool;

I. INTRODUCTION

The application context of our work is related to videosurveillance systems. Our research was guided by different research projects in collaboration with the French National Police, SNCF, RATP and Thales Sécurité for a forensic application domain. In this context, the targeted systems are characterized by : (1) the big "variety" of content acquisition contexts (e.g., indoor,outdoor), (2) the big data volume and the lack of access to some content, (3) the multitude of system owners and the lack of standards, which leads to a heterogeneity of data and metadata formats generated by videosurveillance systems. Consequently, on one hand, the development of content based indexing tools generic and reliable in all contexts is problematic given the acquisition contexts diversity, the content volume and the lack of direct access to certain sources. On the other hand, the lack of metadata associated to the videos (tags, comments) makes the use of classical indexing approaches very difficult.

Public and private location nowadays rely heavily on cameras for surveillance and the number of surveillance cameras in service in public and private areas is increasing. Some estimations show that there are more than 400000 cameras in London and that only the RATP (Autonomous Operator of Parisian Transports) surveillance system comprises around 9000 fixed cameras and 19000 mobile cameras in Paris. But when needed, the content

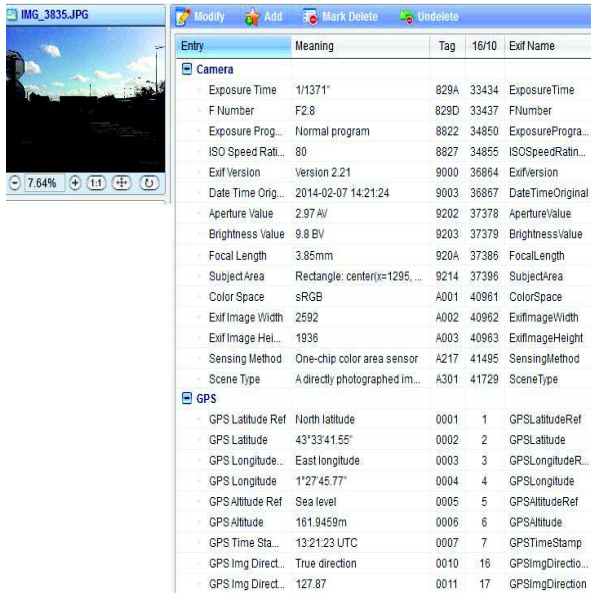
the surveillance videos is analyzed by human agents that have to watch the videos organized in a matrix called video wall. Very poor information are available for the agent that can only refer to the camera id and to his personal expertise in order to place the device with regards to the query's spatial elements given by the victim. Several studies show the cognitive overload coupled with boredom and fatigue lead to errors in addition of the prohibitive processing time. In that context, the main question is which tools can assist the human agents better do their work?

Many efforts to develop "intelligent" videosurveillance systems have been witnessed in the past years. The majority of these efforts aim at developing accurate content analysis tools [1] but the exhaustive execution of content analysis is resource intensive and gives poor results because of the video contents heterogeneity. The main idea we put forward is to use metadata from different sources (e.g., sensor generated data, technical characteristics) to pre-filter the video content and implement an "intelligent" content based retrieval.

With the development of new technologies, it becomes easy and relatively inexpensive to deploy different types of sensors (e.g., GPS sensors, compasses, accelerometers) associated to the cameras. The existent approaches show that based only on the spatial metadata and on the cameras characteristics, it is possible to extract precise and precious information concerning the filmed scene [2]. Nevertheless few approaches take into account the temporal dimension and propose generic data models and operators that can be used in a spatio-temporal trajectory based video retrieval context. For example, Figure 1 illustrates the metadata associated to an image taken with an Iphone4 camera.

When a person (e.g., victim of an aggression) files a complaint, she is asked to describe the elements that could help the human agents find the relevant video segments. The main elements of such description are: the location, the date and the time, the victim's trajectory and some distinguishing features that could be easily noticed in the video (e.g., clothes color, logos). Based on the spatial and temporal information and on their own knowledge concerning the cameras location, the surveillance agents select the cameras that could have filmed the victim's trajectory. Then, the filtered content is visualized in order to find the target scenes, objects (or people) and events.

Based on these observations, the contribution of this



Entry	Meaning	Tag	16/10	Exif Name
Camera				
Exposure Time	1/1371"	829A	33434	ExposureTime
F Number	F2.8	829D	33437	FNumber
Exposure Prog...	Normal program	8822	34850	ExposureProgra...
ISO Speed Rati...	80	8827	34855	ISOSpeedRatin...
Exif Version	Version 2.21	9000	36864	ExifVersion
Date Time Orig...	2014-02-07 14:21:24	9003	36867	DateTimeOriginal
Aperture Value	2.97 A/	9202	37378	ApertureValue
Brightness Value	9.8 BV	9203	37379	BrightnessValue
Focal Length	3.85mm	920A	37386	FocalLength
Subject Area	Rectangle: center(x)=1295,...	9214	37396	SubjectArea
Color Space	sRGB	A001	40961	ColorSpace
Exif Image Width	2592	A002	40962	ExifImageWidth
Exif Image Hei...	1936	A003	40963	ExifImageHeight
Sensing Method	One-chip color area sensor	A217	41495	SensingMethod
Scene Type	A directly photographed im...	A301	41729	SceneType
GPS				
GPS Latitude Ref	North latitude	0001	1	GPSLatitudeRef
GPS Latitude	43°33'41.55"	0002	2	GPSLatitude
GPS Longitude...	East longitude	0003	3	GPSLongitudeR...
GPS Longitude	1°27'45.77"	0004	4	GPSLongitude
GPS Altitude Ref	Sea level	0005	5	GPSAltitudeRef
GPS Altitude	161.9459m	0006	6	GPSAltitude
GPS Time Sta...	13:21:23 UTC	0007	7	GPSTimeStamp
GPS Img Direct...	True direction	0010	16	GPSImgDirectio...
GPS Img Direct...	127.87	0011	17	GPSImgDirection

Figure 1. Metadata example

paper concerns the video filtering and retrieval. We did an analysis of the current query processing mechanism within the videosurveillance systems that highlighted the fact that the entry point of any query is a trajectory reconstituted based on a person's positions and a time interval. These elements are used to select the videos recorded by the cameras that are likely to have filmed the scenery of interest. Consequently, the video retrieval is treated as a spatio-temporal data modelling problem. In this context, we have proposed the following elements :

- a definition of the hybrid trajectory query concept, trajectory that is constituted of geometrical and symbolic segments represented with regards to different reference systems (e.g., geodesic system, road network);
- a multilayer data model that integrates data concerning : the road network, the transportation network, the objects movement, the cameras fields of view changes;
- some operators that, based on a trajectory query and a time interval, select the fixed and mobile cameras whose field of view is likely to have filmed the query trajectory.

The paper is organized in the following manner : The section II presents a state of the art following two axes: (1) the research projects addressing the problem of intelligent video retrieval within videosurveillance systems and (2) the approaches that have associated spatio-temporal information to visual content for organizing, browsing or annotating. The section III presents the innovative query type that our spatio-temporal filtering framework processes. In section IV we present the data model our framework is based on and the section V explains in detail the specifications of the operators that we have defined.

II. RELATED WORK

There has been significant research on videosurveillance intelligent systems for both real time and a posteriori inquiry cases [3]. The majority of the research projects in this domain aim at developing content analysis tools. This tools are usually managed by a multimedia indexing module that can contain a fixed or a variable set of tools than can be executed on the entire multimedia collection or only on a filtered subcollection. We present in the following some research projects interested in video querying systems focusing on the way they filter the content before executing the video features extractors tools. The CANDELA project proposes a generic distributed architecture for video content analysis and retrieval [4]. The exhaustive content analysis is done in a distributed manner at acquisition time by a fixed set of tools. The CARETAKER project ¹ investigated techniques allowing the automatic extraction of relevant semantic metadata from raw multimedia. Nevertheless, there is no filtering of the content before the feature extraction. More related to our work, the VANAHEIM European project ², based on the human abnormal activity detection algorithms, proposed a technique for automatically filter (in real time) the videos to display on the videowall screens. Nevertheless, the filtering is based on a video analysis based learning process that supposes the utilization of a big volume of data and that is difficult to implement on a larger scale.

In the following, we present research works aiming to organize and retrieve video segments based on spatio-temporal information.

Many approaches focus on associating camera latitude /longitude and direction in order to browse or annotate images. [5] proposed to associate one image to it's spatial field of view. This association is based on the metadata collected from the sensors associated to the camera in order to hierarchically organize photo collections. [2] developed a mobile application for annotating images. Besides the camera location and direction information, [2] computed also a distance between the camera and the target object.

[6], proposes the SEVA system that annotates each frame of a video with the camera location, the timestamp and the identifiers of the objects that appear in that frame. Therefore this solution can only be applied in a controlled environment. In [7], an approach similar to SEVA is proposed with the following differences: (1) the objects don't have to transmit their positions and (2) the objects geometry is considered and not only their localisation. For each second of the video, two external databases (OpenStreetMaps and GeoDec) are queried in order to extract the objects (e.g., buildings, parcs) that are located within the camera's visible area. The system doesn't consider spatial queries but only keywords ones. Very related to our work, [5] propose a framework that associates each frame of a video with the geometry of the viewable scene based on

¹http://cordis.europa.eu/ist/kct/caretaker_synopsis.htm

²<http://www.vanaheim-project.eu/>

metadata collected from GPS and compass sensors. Based on a region query, the framework can return the video sequences that have intersected the video query region. The main difference between their framework and ours is that they don't address the multimedia retrieval process.

The interest for new efficient solutions for videos content retrieval indexing and retrieval based on information gathered from sensors is growing in the past years, as illustrated by the state of the art. From our knowledge, no approach has yet proposed a generic extensible model and a querying framework for fixed and mobile video sensors.

In the following we will present the type of query that our framework considers, the hybrid trajectory based one.

III. HYBRID TRAJECTORY BASED QUERY DEFINITION

The idea behind the concept of object trajectory originates in the need to capture the movement of an object in an area for a period of time. The *movement track* of an object (or raw movement, or raw trajectory) is defined basically as a sequence of spatio-temporal positions (i.e., positions that refer to the modifications of an object's spatial characteristics (coordinates and/or geometry) in time) [8]. Depending on the sensors that detected them, the positions can be either geometric (sequence of (x,y,t) triplets for 2D positions) [8] or symbolic (sequence of (rfidtag, t)) [9]. Also depending on the object's type and on the capability of the sensors associated to the object or embedded in the environment, additional data can be associated to the object's movement (e.g., for a mobile camera, it is interesting to capture information like orientation and field of view) [10].

Let us formalize the trajectory's definition. A trajectory is a sequence of segments and a time interval. A segment (u_k) is defined as sequence of homogeneous positions. The innovation that we introduce in the trajectory's definition is that we take into account the fact that a position is expressed with regards to a reference system (e.g., geodesic, road network). The positions are homogeneous within a segment (and expressed with regards to an unique reference system) but the different segments might contain heterogeneous positions.

We define a hybrid trajectory query based in the following :

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

where tr_{id} represents the trajectory's identifier, $\{u_k\}$ represents the set of spatial segments that compose the trajectory and $[t_{start}, t_{end}]$ represents the query's time interval.

The definition of a trajectory segment is given by the following equation :

$$u_k = (u_{id}, refS_{id}, \{position_i\}) \quad (2)$$

where u_{id} represents the segment's identifier, $refS_{id}$ is the reference system identifier with regards to which the positions are expressed and $\{position_i\}$ is the set of positions that compose the trajectory segment.

```
{
  "query": {
    "spatial": [
      {
        "WGS84": [
          [43.60571, 1.445331],
          [43.60933, 1.46292],
          [43.56366, 1.46156]
        ]
      },
      {
        "RRTLSE": {
          {"Route de Narbonne": 116}
          {"Route de Narbonne": 118}
        }
      }
    ],
    "temporal": {
      "start": "09-28-2014 18:50:00",
      "end": "09-28-2014 19:15:00"
    }
  }
}
```

Figure 2. Outdoor hybrid trajectory based query JSON example

In order to manage trajectory segments' heterogeneity, we define the object's position like consisting of two big parts: the part that designates the reference system and the part that designates the location with regards to the reference system. The idea is somehow similar to the one proposed by [11]. They define an hybrid indoor location syntax based on the URI (Universal Resource Identifier) syntax. Nevertheless, the notion of path that they use considers only the hierarchical aspect (a room is situated on a hallway that is part of a building's wing). We are interested in the trajectory concept, which implies a connection between the successive points. Also, they don't consider different reference systems, but only the plan of a building.

Based on the observations that we made, we present in the following two examples of *hybrid trajectory based query* consisting in symbolic and geometrical segments (see figure 2 and 3). Each trajectory based query has a spatial and a temporal part. The spatial part consists of a sequence of segments, each segment being made up of a part that designates the reference system and a sequence of positions (geometric and symbolic) expressed with regards to the corresponding reference system. Figures 2 and 3 illustrate hybrid trajectory based queries for outdoor and indoor environment. In both examples we can see the spatial/temporal structure and the locations expressed with regards to different reference systems identified by system identifiers that are known and managed by the data model (e.g., WGS84 - geodesic system, RRTLSE - Toulouse road network, RTTLSE - Toulouse transportation network).

The final need within the videosurveillance systems and the ultimate perspective of our research work is to develop a framework that would enable the processing of a seamless indoor-outdoor trajectory containing heterogeneous segments. In this paper, we focus on the processing of outdoor trajectories consisting of a mix of symbolic and

```

{
  "query": {
    "spatial": [
      {
        "Floor8": [
          [50.21627, 433.53088],
          [65.84485, 409.75342],
          [60.56707, 441.10219],
          [93.67555, 371.18614],
          [97.67467, 389.67915]
        ]
      },
      {
        "ICCARD": [0000110015010005]
      }
    ],
    "temporal": {
      "start": "10-02-14 12:55:34",
      "end": "10-02-2014 13:20:00"
    }
  }
}

```

Figure 3. Indoor hybrid trajectory based query JSON example

geometric segments.

In the following we will present the data definition that we use in order to process hybrid trajectory based query in outdoor environment.

IV. DATA DEFINITION

We proposed a model that integrates information concerning: (1) The road Network, (2) The transportation Network, and the objects and sensors that move in this environment (3) Objects and (4) Cameras. Our model has been described in [12]. In the following we will describe the data definitions used in our data modeling.

1) *The road network*: The *Road Network* definition is relatively simple. Given a real road network, its 2D modeling is represented by an ordered sequence of points situated on the network that have positions with regards to the geodesic system ($p = \langle \text{lat}, \text{long} \rangle$). Each two points define a line. All these lines give an approximation of the network real shape. The precision of this representation depends of the number of considered points. By considering a different granularity level, this modeling approach is equivalent to the graph based approach [13]. The chosen modeling approach enables to use different network granularity levels [14].

Definition 1: We define a road network based on a piecewise linear function $f : V \rightarrow R^2$. For each point of the vector returned by the query submitted to Google Maps, an interpolation point is added in the function definition. The linear interpolation consists basically in linking the ordered points. We will use the notation R_R to design the set of positions belonging to the road network.

Definition 2: We define a mapping function $\text{map0} : \text{positionsGPS} \rightarrow R_R$ which "translates" a GPS position in a road network position (e.g., 14 Montesquieu street). We define also the inverse function $\text{map0}^{-1} : R_R \rightarrow \text{positionsGPS}$ (Reverse Geocoding de Google Maps).

2) *The transportation network*: If the road network representation is pretty simple and canonical in the literature, the things are more complicated for the transportation network because of the spatial (the predefined trajectories) and temporal (the predefined timetable) constraints.

In order to define the transportation network modeling approach, we relayed on the GTFS format [15]. GTFS is the description format the most used by the agencies providing transportation services and also by the providers of GIS (Geographic information System) or multi-modal trajectory computation applications. In the following we present the data model formalization that we used to describe the transportation network with the enrichments that we brought to it. As in our work we were interested only in the bus network we will refer to it in our definitions.

Definition 3: The transportation network R_T is a tuple (R, Tr, St) where R is the set of routes (bus lines), a route being a sequence of tuples (R_{id}, r_{name}) where R_{id} designates the routes identifier and r_{name} its name (e.g., bus line number 2), Tr is the set of trips corresponding to the routes of R (a trip corresponds to a bus passage from the first station until the last one of a given line) and St represents the set of stations characterized by their position. The link between a bus trip and the bus stations that compose the trip is given by the Definition 5.

Definition 4: The bus trip is defined as a tuple $(Trip_{id}, R_{id}, H)$, where $Trip_{id}$ is the trip identifier (which aims at identifying the 5th bus that passed on a given route for example) and H contains the timetable corresponding to the trip.

Definition 5: The timetable H corresponding to a transportation network consists of a sequence of tuples $(S_{id}, S_{seq}, Tr_{id}, t_1, t_2)$ where S_{id} designates the station identifier, S_{seq} represents the station sequence number with regards to the trip identified by Tr_{id} , and t_1 and t_2 represent the bus arrival and departure time in the station S_{id} .

Definition 6: Let us define the mapping function $\text{map1} : \text{sections}(R_T) \rightarrow \text{positions}(R_R)$ that associates each section (segment defined by two consecutive bus stations) of the transportation network to a sequence of points on the road network.

3) *Objects*: In the following definitions, we will use a specific notation for the spatial and temporal data types (e.g., point, instant).

Definition 7: Let $\text{MO} = \{\text{mo}\}$ be the set of mobile objects / $\text{mo} =$ a mobile object (e.g., bus, car, person), $\text{id}(\text{mo})$ gives its identifier, $m\text{position}_i(\text{mo}) = (\text{position}(\text{mo})_i, t_i)$, $\text{position}(\text{mo})_i \in \text{point}$ and $\text{time}(m\text{position}_i) = t_i \in \text{instant}$ gives the object's position (which is a point) at t_i . $\text{TR}(\text{mo}_i)$ gives the object's trajectory (see definition 8).

Definition 8: The (discrete) trajectory of a mobile object is an ordered set of points and associated timestamps : $\text{TR} = \{(m\text{position}_i, t_i) | t_i < t_{i+1}\}$.

4) *Cameras*: In the following we will give the definitions related to the fixed and mobile cameras and their field of view (fov).

Definition 9: Let us define $\text{fov} = (\text{focalDistance}, \text{direc-}$

tion, visibleDistance, angleView, sensorSize, t_i), a vector of optical features of a camera c_j at t_i . Based on these features, the cameras field of view can be computed [16].

Definition 10: Let $FC = \{fc\}$ the set of fixed cameras/ fc is a fixed camera, $id(fc)=c_i$ represents its identifier, $position(c_i)$ gives its position/ $position(c_i) \in \underline{point}$ and $fov(c_i)$ gives the set of its field of view changes fov_i .

Definition 11: Let $MC = \{mc\}$ the set of mobile cameras/ mc is a mobile camera associated to a mobile object, $id(mc)=c_i$ represents its identifier and $mo(c_i) \in MO$ gives the mobile object to which the camera is associated. The camera's c_i trajectory will be the mobile object having the identifier $mo(c_i)$ one (see definitions 7 and 8).

Based on the data model that we presented we defined the following operator.

V. OPERATORS

Before specifying the operator that we developed in our work, we define the following functions and predicates:

- *geometry(point p, fov f): region* computes a cameras field of view geometry based on its position and its optical characteristics at a moment t ;
- *intersects(polyline seg, region g): boolean* checks if the geometry g intersects a road segment seg ;
- *intersects(point p, polyline seg): boolean* checks if a point p intersects a road segment seg ;
- *intersects(point p, set(polyline) tr): boolean* checks if a point p intersects a set of segments and returns true if $\exists seg_i$ in $tr / intersects(p, seg_i)$.

The entry point of our framework is a hybrid trajectory based query, composed of geometric and symbolic positions expressed with regards to different reference systems and a time interval. Once this query is interpreted, a query interpreter module with regards to our data model, the query is translated into:

- a spatial trajectory consisting of a sequence of segments (u_1, u_2, \dots, u_k) ;
- a time interval $[t_1, t_2]$.

These data can be extracted from the police inquiry data. The goal is to provide the human security agents with a list of video sequences that are likely to have filmed the searched trajectory. In order to do that, we have to search for the fixed cameras whose field of view (that can change) intersected the trajectory during the time interval and the mobile cameras likely to have filmed the researched scene.

Given the objectives, the goal is to find all the fixed cameras' fields of view that have intersected the trajectory during the interval $([t_1, t_2])$ and the mobile cameras likely to contain relevant informations for the trajectory query because their own trajectory intersected the query trajectory between t_1 et t_2 . In the following we will present separately the two selection operators, for the fixed cameras and the mobile cameras.

Let us define the *hasSeen* operator [17], [18]. Given the spatial trajectory $tr=(u_1, \dots, u_n)$ and the time interval $[t_1, t_2]$, *hasSeen*(tr, t_1, t_2) returns the set of cameras c_i ($1 \leq i \leq m$) associated to a segment u_k ($1 \leq k \leq n$) and a video extract between to time moments t_{start}^i and t_{end}^i

with $t_{start}^i \in [t_1, t_2]$ et $t_{end}^i \in [t_1, t_2]$. Each element of this set points out that the video sequence between t_{start}^i and t_{end}^i generated by the camera c_i has filmed the segment u_k .

$$hasSeen : u_1, u_2, \dots, u_n, [t_1, t_2] \rightarrow L$$

$$L = \begin{cases} c_1 : t_{start}^1 - > t_{end}^1, u_k (1 \leq k \leq n) \\ c_2 : t_{start}^2 - > t_{end}^2, u_k (1 \leq k \leq n) \\ \dots \\ c_m : t_{start}^m - > t_{end}^m, u_k (1 \leq k \leq n) \end{cases} \quad (3)$$

In the following, we will present the operator's specifications for the two cases, fixed and mobile cameras.

A. Fixed cameras

The result of the operator *hasSeen* for the fixed cameras is a set of triplets: $R = \{r = (c_i, u_k, [t_a, t_b])\}$, $c_i \in FixedCameras$, $u_k \in tr$, $t_1 \leq t_a, t_b \leq t_2$. The operator that we defined check what are the fixed cameras whose fields of view have intersected one of the query's spatial segments and between what time instants (t_a and t_b).

$r \in R \equiv$ It exists $fov_j \in fov(c_i)$ ($fov(c_i)$ is the set of instants when the field of view of the camera c_i changes such that:

$$\begin{aligned} & time(fov_j) \in [t_1, t_2] \\ & \wedge intersects(u_k, \\ & \quad geometry(position(c_i), fov_j)) \\ & \wedge t_a = time(fov_j) \\ & \wedge t_b = \min(time(succ(fov_j)), t_2) \end{aligned}$$

\vee

$$\begin{aligned} & time(fov_j) < t_1 \\ & \wedge t_1 \leq time(succ(fov_j)) \\ & \wedge intersects(u_k, \\ & \quad geometry(position(c_i), fov_i)) \\ & \wedge t_a = t_1 \\ & \wedge t_b = \min(time(succ(fov_i)), t_2) \end{aligned}$$

in the case where the change point is in the interval $[t_1, t_2]$, if the corresponding field of view geometry intersects one of the trajectory segments, then the result's time interval starts at the change moment and ends at the following change time point or at the end of the query time interval

in the case where the change happens before t_1 , and the next change time point is after t_1 , if the corresponding field of view geometry intersects one of the query segments, then the result time interval starts at t_1 and ends at the following change time point or at the end of the query time interval.

B. Mobile cameras

The operator *hasSeen* searches for the mobile cameras whose positions intersected one of the query segments and during what time interval (instants t_a and t_b). In the case of mobile cameras (installed on board of buses or police cars), their movement (sometimes with a high speed) doesn't allow the computation of the precise intersection between the field of view and the query segments. The

results only show that the mobile camera was located on the query segment during the query time interval.

The result of the hasSeen operator for the mobile cameras is the set of triplets: $R = \{r = (c_i, u_k, [t_a, t_b])\}$, $c_i \in MC$, $u_k \in tr$, $t_1 \leq t_a, t_b \leq t_2$.

$r \in R \equiv$ It exists $mp_j \in TR(mo(c_i))$, a position of the mobile object trajectory to which the mobile camera c_i is associated to such that:

$$\begin{aligned} & [time(mp_j(mo_i)) \in [t_1, t_2] \\ & \wedge \text{intersects}(\text{position}(mp_j(mo_i)), u_k) \\ & \wedge (\text{not intersects}(\text{prec}(\text{position}(mp_j(mo_i))), tr) \\ & \quad \vee (\text{intersects}(\text{prec}(\text{position}(mp_j(mo_i))), tr) \\ & \quad \wedge \text{prec}(\text{time}(mp_j(mo_i)), t_1)) \\ & \wedge t_a = \max(\text{prec}(\text{time}(mp_j(mo_i))), t_1) \\ & \wedge t_b = \min(\text{succ}(\text{time}(mp_j(mo_i))), t_2)] \\ & \vee \text{ // second part of the operator} \\ & [t_1 \leq \text{time}(mp_j(mo_i)) \\ & \wedge \text{time}(mp_j(mo_i)) \leq t_2 \\ & \wedge \text{intersects}(\text{position}(mp_j(mo_i)), u_k) \\ & \wedge \text{intersects}(\text{prec}(\text{position}(mp_j(mo_i))), tr) \\ & \wedge t_a = \text{time}(mp_j(mo_i)) \\ & \wedge t_b = \min(\text{succ}(\text{time}(mp_j(mo_i))), t_2)] \end{aligned}$$

in the case where the position of the object is on the query trajectory in the query interval $[t_1, t_2]$ and the position before isn't on the query trajectory, then the response time interval starts at the maximum between the moment of the last object's position update and t_1 and ends at the next position update or at the end of the query interval

in the case where the object's position is on the query trajectory during the query interval $[t_1, t_2]$ and the position before is also on the trajectory, then the response time interval starts at moment of the position's update and ends at the moment of the next update or at the end of the query interval.

VI. FORENSIC FRAMEWORK ARCHITECTURE

Figure 4 illustrates the forensic framework architecture that we have developed in our research work and that implements the data model and the operators described in the previous sections.

The main modules of the framework are :

- Terminal Interface: gathers all the bricks that implements the users interaction and the results and data visualization via a Google Maps API;
- Query Interpreter: implements the hybrid trajectory query submitted by the user into a spatio-temporal query (a spatial segments sequence projected on a urban road network and a time interval);
- Search Engine: implements the research operator defined in the previous section;
- Storage: contains the spatio-temporal database that stores the data model presented and the modules that enable the interaction with it;

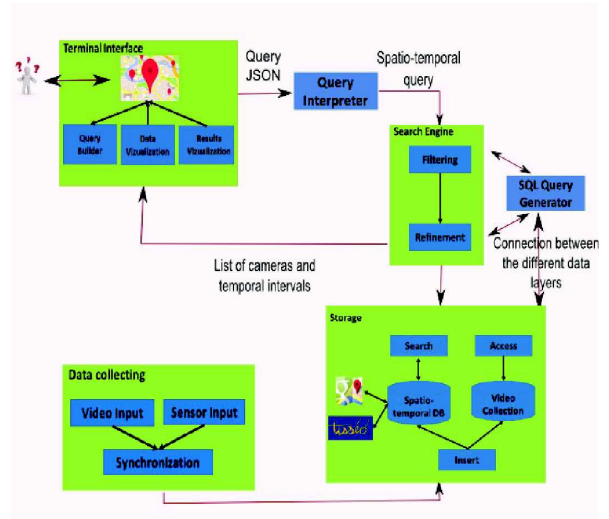


Figure 4. Forensic

- SQL Query Generator: enables the communication between the search engine and the storage module;
- Data Collecting: is in charge of the video and spatio-temporal data collecting produced by the cameras network and the sensors attached to them (GPS, compass, accelerometer);

VII. CONCLUSION

In this paper, we presented a spatio-temporal modelling approach of fixed and mobile cameras within a common transportation network. Taking our inspiration from the multilayer representation of the GIS (Geographical Information Systems), we model spatial information about the road and transportation infrastructures and mobile objects' trajectories in four independent layers: (1) Road network, (2) Transportation network, (3) Objects and (4) Cameras network. Based on this modeling approach we also proposed a generic architecture for a system that could assist the video surveillance operators in their research. Starting from a sequence of trajectory segments and a temporal interval, such system generates the list of cameras that could contain relevant information concerning the query (that "saw" the query's trajectory). The need of such assisting tools was identified within the National Project METHODEO. Among the project's partners, we mention the French National Police, Thales and the RATP (Autonomous Operator of Parisian Transports). Our approach has been validated and will be evaluated within the project. For now, our model considers only outdoor transportation and surveillance networks. We plan to extend it to indoor spaces also in order to model cameras inside train or subway stations for example. Our work has been done in the context of the a posteriori research in the case of a police inquiry. We would like to extend this context in order to be able to process real time queries or to predict trajectories based on some statistics based on the stored data (e.g., average speed on some road segments).

ACKNOWLEDGMENT

The authors would like to thank... more thanks here

REFERENCES

- [1] R. Cucchiara, "Multimedia surveillance systems," in *Proceedings of the Third ACM International Workshop on Video Surveillance and Sensor Networks*, ser. VSSN '05. ACM, 2005, pp. 3–10.
- [2] H. Debnath and C. Borcea, "Tagpix: Automatic real-time landscape photo tagging for smartphones," in *6th International Conference on MOBILE Wireless MiddleWARE, Operating Systems, and Applications*. Mobilware 2013, 2013.
- [3] F. Sedes, J. Sulzer, D. Marraud, C. Mulat, and B. Cepas, *Intelligent Video Surveillance Systems*. Wiley, 2012, ch. A Posteriori Analysis for Investigative Purposes.
- [4] P. Merkus, X. Desurmont, E. G. T. Jaspers, R. G. J. Wijnhoven, O. Caignart, J. f. Delaigle, and W. Favoreel, "Candela - integrated storage, analysis and distribution of video content for intelligent information systems," 2004.
- [5] B. Epshtein, E. Ofek, Y. Wexler, and P. Zhang, "Hierarchical photo organization using geo-relevance," in *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*, ser. GIS '07, 2007, pp. 1–7.
- [6] X. Liu, M. Corner, and P. Shenoy, "Seva: Sensor-enhanced video annotation," *ACM Trans. Multimedia Comput. Commun. Appl.*, pp. 1–26, 2009.
- [7] Z. Shen, S. Arslan Ay, S. H. Kim, and R. Zimmermann, "Automatic tag generation and ranking for sensor-rich outdoor videos," in *Proceedings of the 19th ACM International Conf. on Multimedia*, ser. MM '11, 2011, pp. 93–102.
- [8] C. Parent, S. Spaccapietra, C. Renso, G. Andrienko, N. Andrienko, V. Bogorny, M. L. Damiani, A. Gkoulalas-Divanis, J. Macedo, N. Pelekis, Y. Theodoridis, and Z. Yan, "Semantic trajectories modeling and analysis," *ACM Comput. Surv.*, vol. 45, no. 4, pp. 42:1–42:32, Aug. 2013.
- [9] C. S. Jensen, H. Lu, and B. Yang, "Graph model based indoor tracking," in *Mobile Data Management: Systems, Services and Middleware, 2009. MDM'09. Tenth International Conference on*, 2009, pp. 122–131.
- [10] S. Arslan Ay, R. Zimmermann, and S. H. Kim, "Viewable scene modeling for geospatial video search," in *Proceedings of the 16th ACM international conference on Multimedia*, ser. MM '08. New York, NY, USA: ACM, 2008, pp. 309–318.
- [11] C. Jiang and P. Steenkiste, "A hybrid location model with a computable location identifier for ubiquitous computing," in *Proceedings of the 4th International Conference on Ubiquitous Computing*, ser. UbiComp'02. London, UK, UK: Springer-Verlag, 2002, pp. 246–263.
- [12] D. Codreanu, A.-M. Manzat, and F. Sedes, "Mobile objects and sensors within a video surveillance system: Spatio-temporal model and queries," in *IMMoA*, ser. CEUR Workshop Proceedings, vol. 1075. CEUR-WS.org, 2013, pp. 52–59.
- [13] C. S. Jensen, J. Kolarvr, T. B. Pedersen, and I. Timko, "Nearest neighbor queries in road networks," in *Proceedings of the 11th ACM International Symposium on Advances in Geographic Information Systems*, ser. GIS '03. New York, NY, USA: ACM, 2003, pp. 1–8.
- [14] I. Sandu Popa, K. Zeitouni, V. Oria, D. Barth, and S. Vial, "Indexing in-network trajectory flows," *The VLDB Journal*, vol. 20, no. 5, pp. 643–669, 2011.
- [15] P. M. Nick Kizoom, "A transmodel based xml schema for the google transit feed specification with a gtfS / transmodel comparison," Kizoom, Tech. Rep., 2008.
- [16] S. Arslan Ay, S. H. Kim, and R. Zimmermann, "Generating synthetic meta-data for georeferenced video management," in *Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems*, ser. GIS '10. New York, NY, USA: ACM, 2010, pp. 280–289. [Online]. Available: <http://doi.acm.org/10.1145/1869790.1869830>
- [17] D. Codreanu, A. Péninou, and F. Sèdes, "Recherche d'extraits vidéos par reconstitution des trajectoires de caméras mobiles à partir d'un modèle spatio-temporel-application à la vidéosurveillance," in *INFormatique des Organisations et Systemes d'Information et de Decision (INFORSID)*, Lyon, France, 2014, pp. 11–26.
- [18] —, "Modélisation de trajectoires cible/caméra : requêtes spatio-temporelles dans le cadre de la vidéosurveillance (poster)," in *Journées Francophones Extraction et Gestion de Connaissances (EGC)*, vol. RNTI-E-26. Revue des Nouvelles Technologies de l'Information (RNTI), 2014, pp. 573–576.