REPAIRING PEOPLE TRAJECTORIES BASED ON POINT CLUSTERING

Duc-Phu CHAU(1)(2), François BREMOND(1), Etienne CORVEE(1), Monique THONNAT(1)

(1) Pulsar, INRIA Sophia Antipolis, France
(2) Department of Technology,
Phu Xuan Private University, Vietnam

Lisboa, 02 - 2009

INRIA
SOPHIA ANTIPOLIS

Outline

- **Introduction**
- **Approach proposed**: Learning lost and found zones
  - Offline Training Phase
    - Learning trajectory confidence using annotated trajectories
    - Computing automatically the lost and found zones using training trajectories
    - Defining zone triplets linking entry, lost and found zones
  - Online Testing Phase
    - Repairing lost trajectories based on zone triplets
- **Experimentation**
- **Conclusion and Future work**

Introduction [1]

- There are many methods proposed to track mobile objects (e.g. Using Kalman Filter, particle filters, probabilistic data association).
  - Obtained satisfactory results, but complex and long situations of occlusion are not addressed.
- Some works[1,2] have focused on modelling the scene in order to improve the tracking algorithm.
  - Require full trajectories
  - Cannot handle occlusions
  - Take into account only the shape and size of the objects, as they are detected on the 2D image plane.

[1] Fernyhough et al., *Buxton, B & Cipolla, R (editors) ECCV*, 1996.

[2] D.Makris et al., *IEEE Transactions on Systems*, Man and Cybernetics, 2005.

Introduction [2]

- Our motivation
  - Repairing lost trajectories thanks to a model of the scene.
  - Offline learning of the scene model
  - Propose a general method
    - Be able to model various types of scene
    - Require little a priori knowledge about the scene
- Our Approach
  - Define the confidence value for each trajectory and Filter out noisy trajectories
  - Learning the lost and found zones for a given scene
  - Detect and Repair online the lost trajectories

Introduction [3]

- Input
  - Trajectories obtained from a tracking algorithm
  - Zones where object can enter or leave the scene (Entry zone and Exit zone)
  - 3D Person Model
- Output
  - Trajectories after have been repaired

Offline training_
Trajectory confidence value [1]

Trajectory confidence value is computed by nine features:

1. Entry zone
2. Exit zone
3. Trajectory temporal duration
4. Trajectory spatial length
5. Type frequency: number of times the mobile object is classified as a 'person'.
6. Frailty: number of times that the trajectory is temporally lost.
7. Number of neighbours: neighbouring mobile objects at some special temporal instants.
8. Change-Size frequency: number of times the mobile object changes its size according to a predefined dimension variation threshold.
9. Change-Direction frequency: number of times the mobile object changes spatial direction.

$$CV = \sum_{i=1}^{5}(w_i * f_i) + \sum_{i=6}^{9}(w_i * (1 - f_i))$$

where,

CV = confidence value of the trajectory

w_i = the weight (importance) of feature i

f_i = the value of feature i after normalisation

Offline training_ Trajectory confidence value [2]

- Determine the Feature Weights using Supervised Learning
  - ‘Ground Truth’: defines the confidence value of a trajectory; manually annotated, value in interval [0,1]
    - **Complete trajectory**
    - **Incomplete trajectory**
    - **Unreliable trajectory**
    - **Noisy trajectory**

Offline training_
Trajectory confidence value [3]

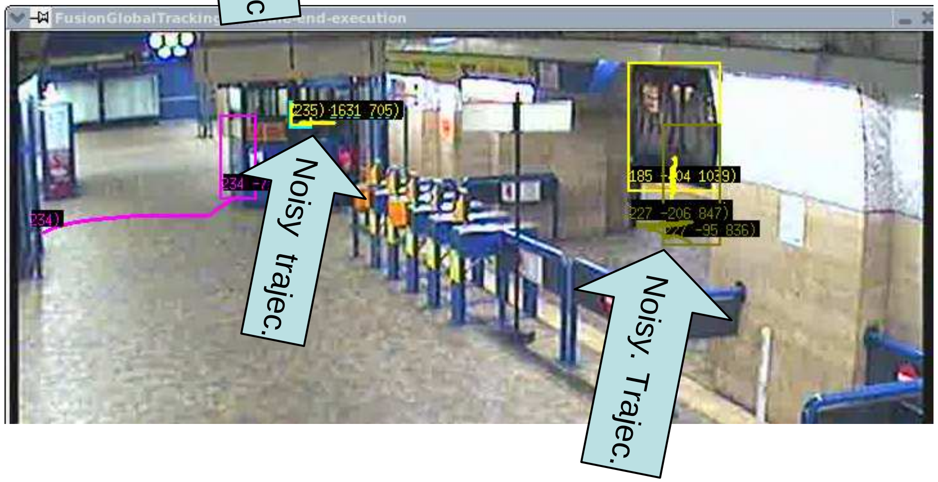
- **Complete trajectory (Complete trajec.)**: starts in an entry zone and ends in an exit zone ground Truth ≥ 0.8
- **Incomplete trajectory (Incomplete trajec.)**: does not start in an entry zone or does not end in an exit zone, 0.5 <= ground truth < 0.8

INRIA SOPHIA ANTIPOLIS

Offline training_
Trajectory confidence value [4]

- **Unreliable trajectory (Unreliable Trajec.)**: does not start in an entry zone and does not end in an exit zone, 0.2 <= ground truth < 0.5

- **Noisy trajectory (Noisy trajec.)**: does not correspond to the trajectory of a person, ground truth < 0.2



INRIA SOPHIA ANTIPOLIS

Offline training_ Trajectory confidence value [5]

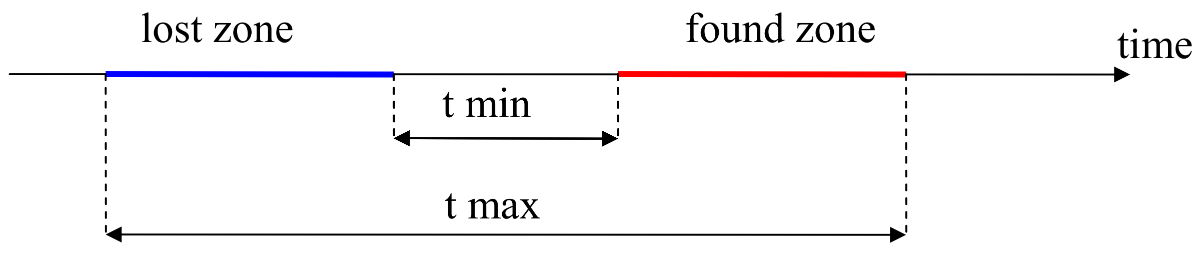
- We optimize the correspondence between ‘Ground Truth’ value and the learned confidence value using **genetic algorithm**.
- Filter out noisy trajectories using the confidence value before learning lost and found zones.

Offline training_
Computing the lost and found zones [1]

- Predefined zones in the scene
  - Entry zone (Start Zone): zone where the object can enter the scene.
  - Exit zone: zone where the object can leave the scene.
- Learned zones in the scene using KMeans clustering
  - Lost zone: zone where the tracking loses usually the objects and differing from exit zone
  - Found zone: zone where the tracking detects usually new objects and differing from entry zone.

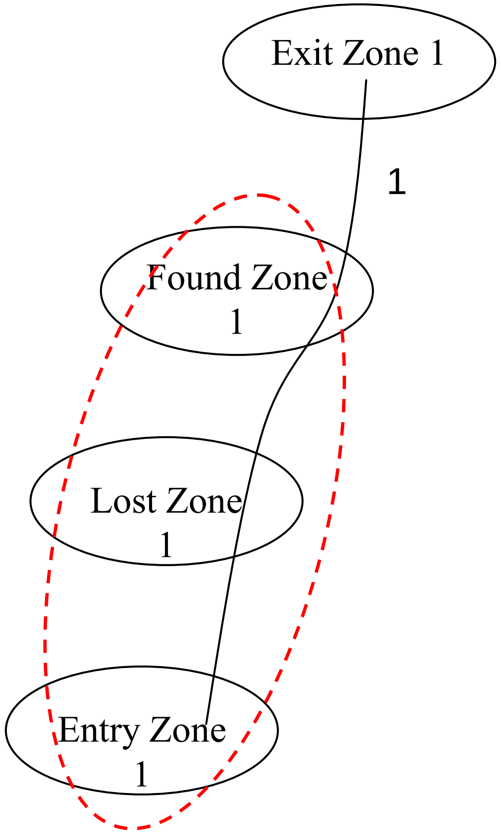
Offline training_
Computing the lost and found zones [2]

- Definition of a zone triplet for a set of trajectories
  - entry zone (EZ)
  - lost zone (LZ)
  - found zone (FZ)
  - ❖ minimum time from lost to found zone (t min)
  - ❖ maximum time from lost to found zone (t max)
- Help the system to online repair the lost trajectories.
- Represent the paths in the scene formed by complete trajectories.

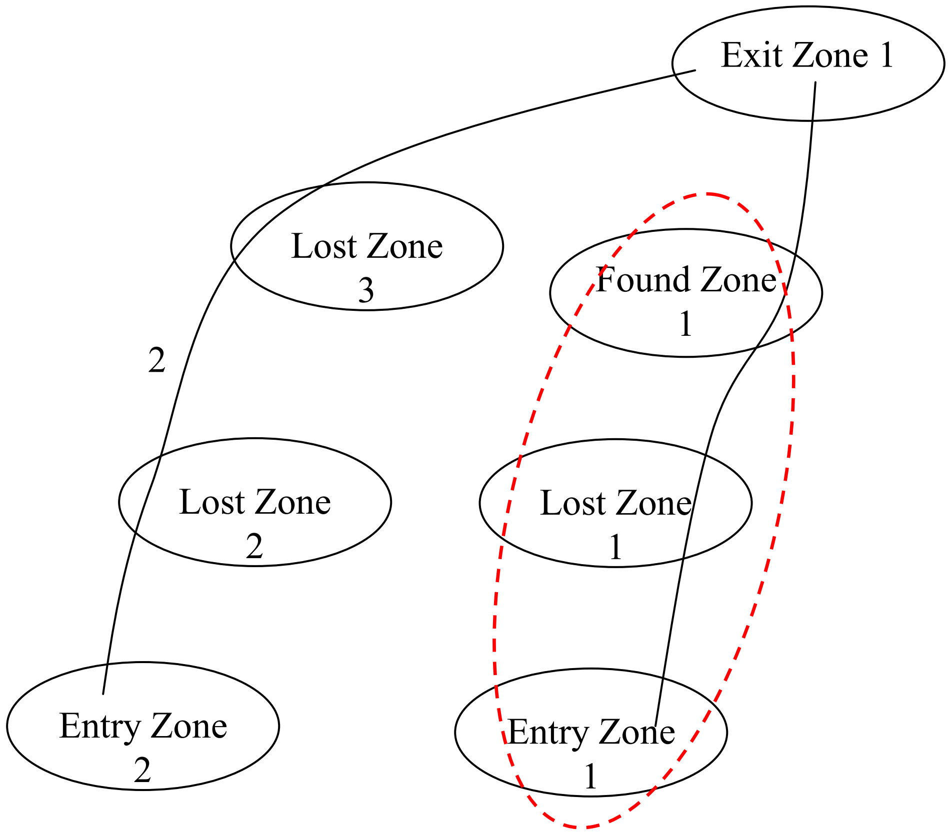


Offline training_
Computing the lost and found zones [3]

- Building zone triplets
  - Only complete trajectories that pass through: ‘entry zone’, ‘lost zone’ and ‘found zone’, are used to construct the zone triplets.
  - Zone triplets ranked (higher priority) by the number of trajectories passing through it.

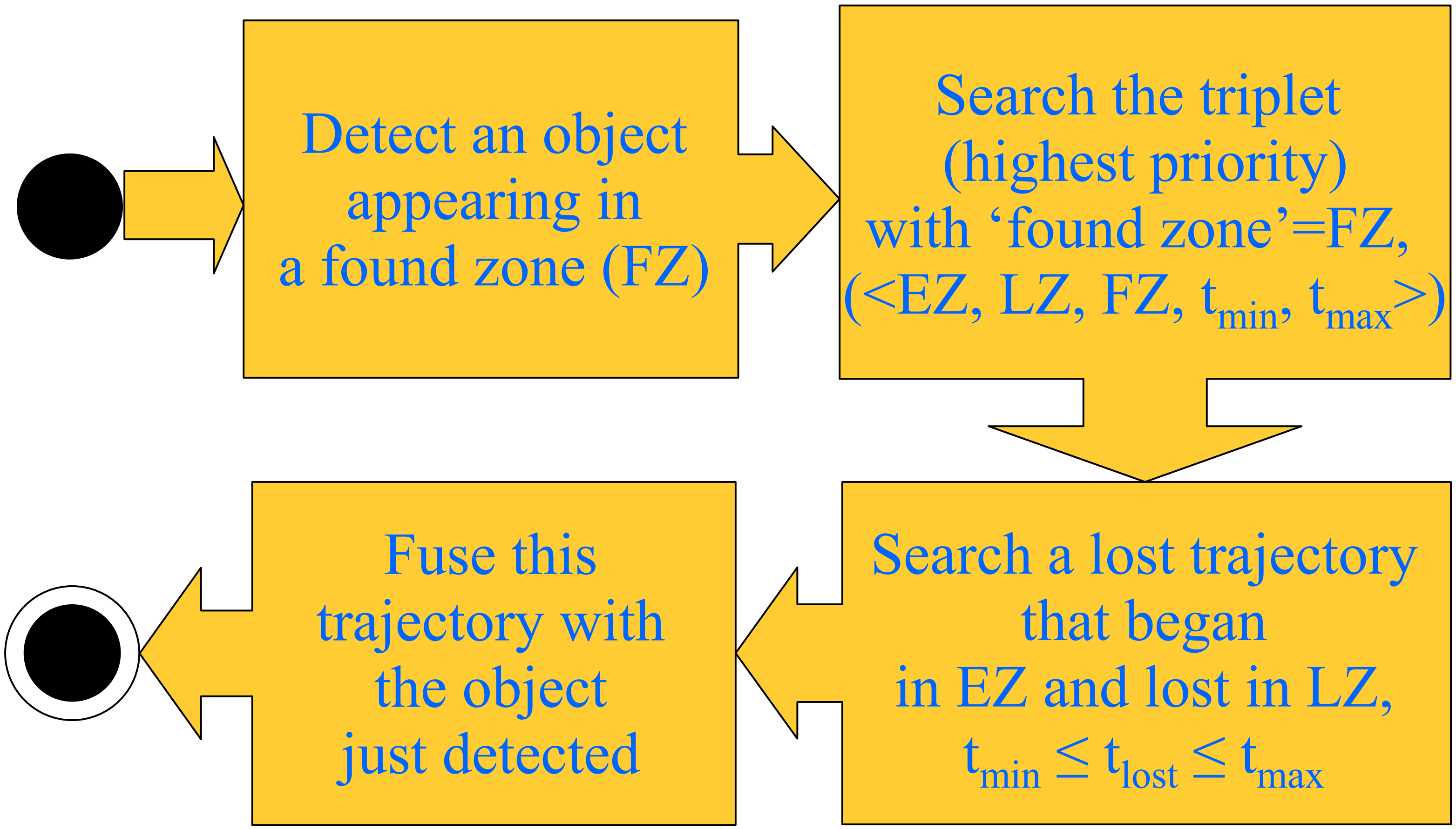


Offline training_
Computing the lost and found zones [4]

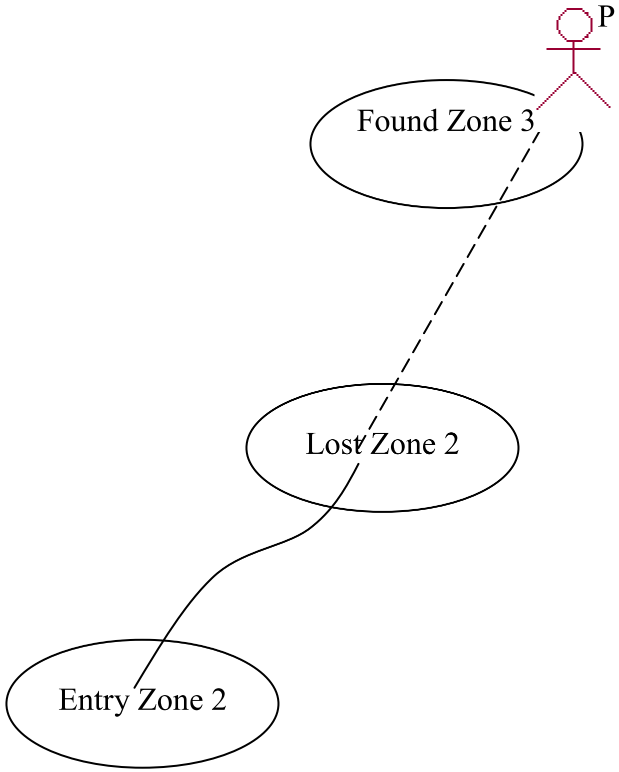


Creation of a zone triplet

Online testing_
Repair Lost Trajectories [1]

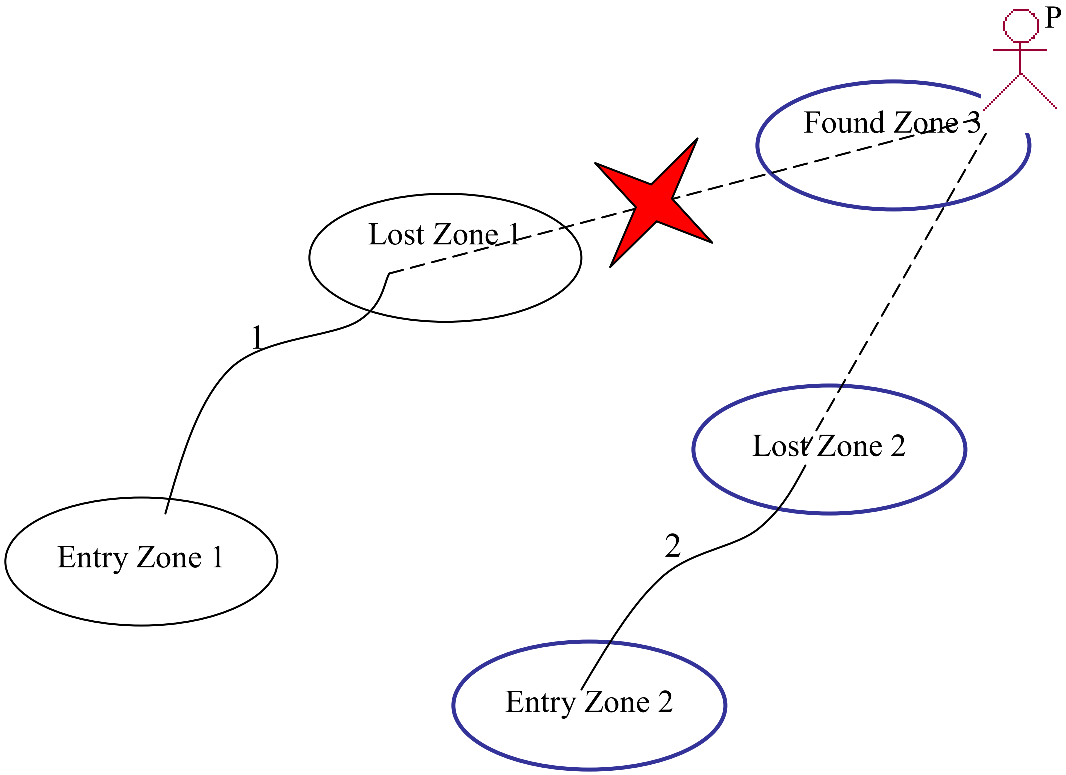


Online testing_
Repair Lost Trajectories [2]



Repairing algorithm

Online testing_
Repair Lost Trajectories [3]



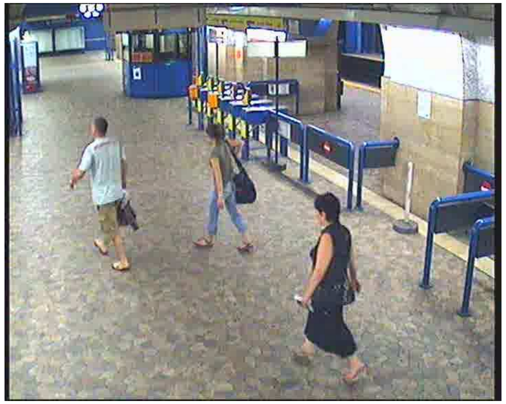
Repairing algorithm

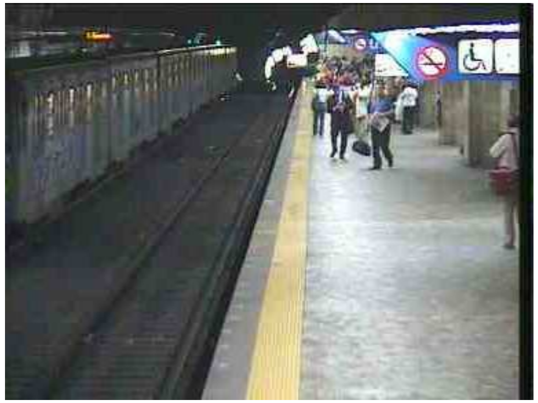
Experimentation [1]

- Video sequences of CARETAKER project

  (http://www.ist-caretaker.org/)

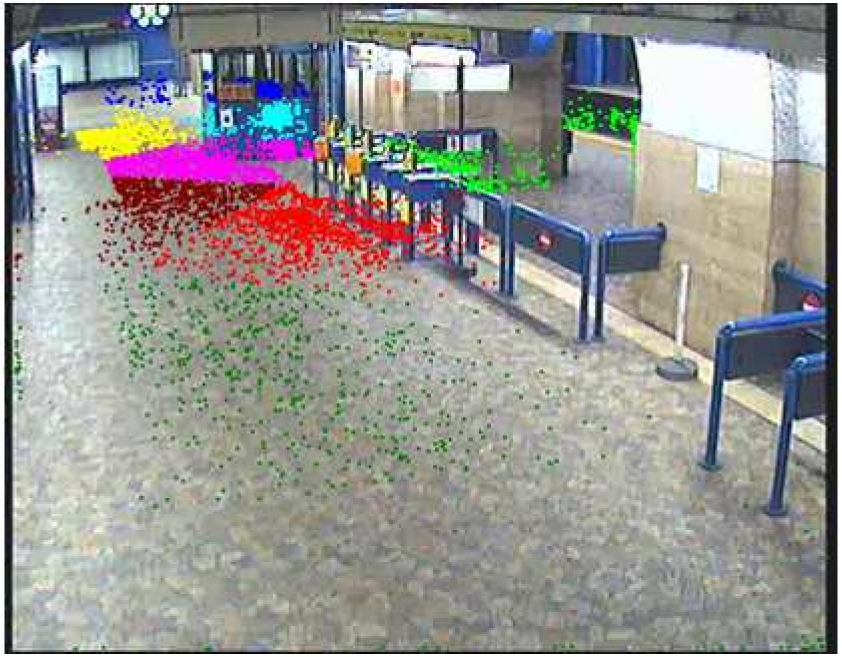
- Cameras are set up in Rome (Italia) subway stations





INRIA SOPHIA ANTIPOLIS

Experimentation [2]



8 learned lost zones

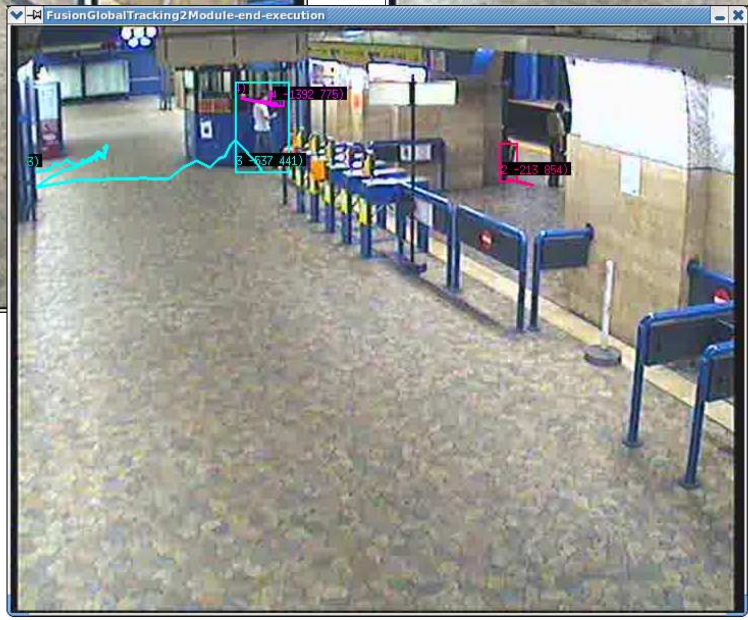
The yellow polygons show the outline of the learned lost zones. The red polygons show the outline of the entry zones, exit zones.

INRIA SOPHIA ANTIPOLIS

Experimentation [3]

t = 709 s

t = 711s without the algorithm



t = 711s with the algorithm

Experimentation [4]

t = 901 s

t = 903s without the algorithm

t = 903s with the algorithm

Experimentation [5]

Summary of results in two cases: with or without the proposed algorithm

	Without the algorithm		With the algorithm	
	Number	Percentage (%)	Number	Percentage (%)
Complete trajectories	758	9.0	795	9.9
Incomplete trajectories	3086	36.8	2778	34.5
Noise	4550	54.2	4481	55.6
Total	8394	100	8054	100

Conclusion and Future Work [1]

- Present a method in order to repair people trajectories in videos.
- Advantages
  - Require little a priori knowledge by exploiting the structure of the scene.
  - Compute trajectory confidence value to filter out noise and to evaluate the algorithm.
  - Define the notion ‘zone triplet’ => representation of general paths, compatible with complex scenes without clear paths, roads.
  - Good preliminary results

Conclusion and Future Work [2]

- Drawbacks
  - Depend on training videos and tracking quality
    - Need enough complete trajectories
  - Large number of short and incomplete trajectories
  - Can repair wrongly lost trajectories
  - Cannot handle crowded scene (people crossing each other)

Conclusion and Future Work [3]

- Future work
  - Using object appearance model for the repair process
  - The first zone of the zone triplet does not have to be a start zone, but could be anywhere
    - This start zone defined as a first zone was used to limit the space for searching trajectories to be fused and hence limits the repair process choices.

References

1. A. Almeida, J. Almeida, and R. Araujo, Real-time tracking of multiple moving objects using particle filters and probabilistic data association, *Automatika*, vol. 46, no. 1-2, pp. 39–48, 2005.
2. A. Avanzi, Francois Bremond, Christophe Tornieri and Monique Thonnat, Design and Assessment of an Intelligent Activity Monitoring Platform, *in EURASIP Journal on Applied Signal Processing*, special issue in "Advances in Intelligent Vision Systems: Methods and Applications", 2005:14, pp.2359-2374.
3. E. Brookner, John Wiley & Sons, Tracking and Kalman Filtering Made Easy, 1998.
4. Fernyhough, J H, Cohn, A G & Hogg, D C Generation of semantic regions from image sequences *in: Buxton, B & Cipolla, R (editors) ECCV'96*, pp.475-478. Springer-Verlag. 1996.
5. D. Makris, T. Ellis, Learning semantic scene models from observing activity in visual surveillance, *IEEE Transactions on Systems*, Man and Cybernetics, Part B 35 (3) (2005) 397–408.
6. S. Maskell, N. Gordon, M. Rollason and D. Salmond, Efficient Multitarget Tracking using Particle Filters, *Journal in Image and Vision Computing*, 21(10): 931-939, September 2003.
7. Raquel R. Pinho, João Manuel R. S. Tavares, Miguel V. Correia, An Efficient and Robust Tracking System using Kalman Filter, *VIPSI-2006 VENICE*, 2006.
8. Raquel R. Pinho1, João Manuel R. S. Tavares and Miguel V. Correia, A Movement Tracking Management Model with Kalman Filtering, Global Optimization Techniques and Mahalanobis Distance, *Lecture Series on Computer and Computational Sciences Volume 1*, 2005, pp. 1-3.

THANK YOU FOR YOUR ATTENTION

(Duc-Phu.Chau@sophia.inria.fr)