STAIRS 2014
U. Endriss and J. Leite (Eds.)

doi:10.3233/978-1-61499-421-3-121

Towards the Usage of Advanced Behavioral Simulations for Simultaneous Tracking and Activity Recognition

Arsène Fansi T. [a,b] and Vincent Thomas [a] and Olivier Buffet [a] and Fabien Flacher [b] and Alain Dutech [a,1]
[a] *INRIA / Université de Lorraine, Nancy, France*
[b] *Thales Services, Vélizy, France*

Abstract. Tracking and understanding moving pedestrian behaviors is of major concern for a growing number of applications. Classical approaches either consider the two problems separately or treat them simultaneously while relying on limited context-based graphical models. In this paper, we present an approach to tackle both the problems conjointly based on richer contextual information issued from agent-based behavioral simulators which aim to realistically reproduce human behaviors within complex environments. We focus on the special case of a single target and experimentally show that the proposed approach manages to track a single pedestrian with complex behavior even in case of long periods of occlusion.

1. Introduction

The ability of using sensor networks to track and understand the behavior of moving human beings is of great importance for a wide range of applications such as surveillance [1] and smart homes [2]. When sensors are cameras, this implies retrieving behavioral information from image analysis. This is not a trivial task to perform, even for humans, and interpretation errors are common. Moreover, when the considered environment is not fully under sensory coverage, one problem of interest is to determine what is the behavior of the tracked targets while being in a non-covered area.

Understanding a pedestrian's behavior is intimately coupled with tracking its location. Moving pedestrians are usually driven by an inner motivation in relation to the activity they are performing in the environment. Therefore, location and motivation are contextually dependent, and the knowledge of one may help estimate the other. However, there are different levels of granularity one can refer to when seeking to understand people's behavior. Imagine people walking through a given ticket machine for withdrawing cash, we may distinguish, in a hierarchical order, *atomic actions* (e.g., walking), *interaction* (e.g., approaching, queuing, avoiding) and *activities* (e.g., withdrawing cash) [2].

[1]Contact Information: {arsene.fansi-tchango, vincent.thomas, olivier.buffet, alain.dutech}@loria.fr, fabien.flacher@thalesgroup.com

[2]In the literature, the term "*activities*" is often used ambiguously to refer to atomic actions or interactions.

These different levels are clearly not independent and, the higher the level, the more the scene's context has to be taken into account.

Most of the existing works, when not considering the problem of tracking and activity recognition separately, rely on limited context-based graphical models for improving trajectory estimation from the estimated activity and vice-versa. We approve the idea of taking advantage of the scene information by performing both operations simultaneously. Moreover, we argue that considering tools providing richer contextual information is an asset to deal with high-level activity recognition even in more complex scenarios, a feature of particular interest for monitoring systems.

Nowadays, the use of simulators to generate contextual-based realistic human behaviors in indoor environments has become very popular in a wide range of application domains[3,4]. These simulators usually encapsulate a model of the considered building (together with the objects therein); and a description of authorized activities is provided, reflecting the prior knowledge about the context of the environment. Works from the situated artificial intelligence field have focused on designing reactive virtual agent control architectures based on sensorimotor loops [5,6] whose purpose is to define, at any time, the behavior a virtual agent will have according to his internal state and current environment context.

In this paper, we are interested in developing a system capable of inferring the activity of pedestrians in indoor non-crowded scenes based on their trajectory, even when they are not under sensor coverage. The proposed approach leverages richer contextual information from such a simulator, which in turn, is integrated within a particle filter for the analysis of people's behavior.

The remainder of this paper is structured as follows. Section 2 provides an overview of related work. Section 3 briefly introduces mathematical background of particle filters while Section 4 describes the implementation details of our system. Section 5 is dedicated to the experimental evaluation of the approach. Finally, Section 6 identifies some important research lines induced by the described work.

2. Related Work

Tracking pedestrians has been of major concern during the last decades. In complex scenes, occlusions often occur and interrupt tracks identified so far. Maintaining relevant information in such situations is of crucial importance in order to assemble complete tracks or understand the behavior of the concerned targets. Works in the literature [7,8,9] have focused on defining mathematical models based on a set of attractive and repulsive fields representing human motion features. Based on these models, methods [10,11,12] have been developed for better handling short occlusions in human motion prediction. However, they do not attempt to understand the behavior of the different individuals.

On the other hand, methods [13,14] have been recently proposed in the literature for action recognition in which authors mainly rely on image feature extraction and feature classification. However, these works do not benefit from advantages available in considering tracking people's trajectories.

Dealing conjointly with location tracking and behavior understanding presents an advantage as it is possible to exploit the relationship between location and behavior. In [15], Bruce et al. represent the state of the pedestrian by his final goal (in terms of destination point) together with his location and then use a particle filter for inferring the pedestrian goal with the help of a path planner. A limit of their method is that people's behaviors are assimilated to the environment physical points. Moreover, there is no causality modeling between these points.

Wilson et al. [16] formally introduced the simultaneous tracking and activity recognition (STAR) problem and consider the use of dynamic Bayesian networks (DBNs) to represent causal influences and causality through time among different variables representing their system state. However, the granularity of the modeled activities is limited to whether or not a target is moving. In [17], the authors investigate the use of relational DBNs to represent relationships among interacting targets in the STAR problem. They probabilistically model the dynamics of relations between different targets and formalize a dynamic model that takes into account such relationships. While result improvements are noticeable, the graphical model used fails to encapsulate relevant information regarding target-object interactions.

In this paper, we also address conjointly the problem of tracking and activity recognition. However, unlike previous approaches, we take into account richer contextual information. By contextual information, we refer to environmental information perceived by a pedestrian that may affect his behavior. In order to do this, we rely on process-oriented behavioral models for autonomous agents. The purpose of such advanced behavioral models is to determine, for each agent, the action he should undertake in the environment based on his internal state and current perceptions. Such models, as described in [4], are not only characterized by physical attributes (e.g., position, velocity), but integrate, within themselves, action selector mechanisms often coupled with cognitive control architectures [18,19,5,6] responsible for creating and executing navigational plans which may involve interactions with objects in the environment. The description of the different objects together with the interactions they can offer are provided to the simulator in order to easily match agent's current action into corresponding interactions in the scene. We then integrate such a simulator, as a predictive block, within a particle filter for people's behavior estimation and analysis.

The main contribution of this paper is two-fold: (i) by integrating advanced agent-based behavioral simulations, we add significant contextual information that may be useful in understanding target behavior from image sequences; (ii) as the results will show, considering richer contextual data makes it possible to handle long periods of occlusions, times during which the tracked agent is likely to perform several activities.

In the following sections, we will briefly introduce relevant background regarding particle filters and describe how we leverage autonomous agent based behavioral models for pedestrian behavior tracking. For the latter, we focus on the special case of a single target.

3. Particle Filtering

In the field of state estimation, the Bayesian filtering framework [20] provides a recursive way of computing the belief $Bel_t(\mathbf{x}_t)$ regarding the state $\mathbf{x}_t$ of a dynamical system at time t given the (potentially noisy) partial observation $\mathbf{z}_t$. It is assumed the availability of the prior belief Bel_0 about the system's initial state. A particle filter (PF) [21] is an approximation of the Bayesian filter in which the belief $Bel_t(\mathbf{x}_t)$ at time t is represented by a set of N_s weighted particles $\mathcal{S}_t = \{\mathbf{x}_t^i, w_t^i\}_{i=1}^{N_s}$ where $\mathbf{x}_t^i$ and w_t^i are respectively the state and the weight of the i^{th} particle. The particle set $\mathcal{S}_t$ is typically constructed from the previous set $\mathcal{S}_{t-1}$ and the current observation $\mathbf{z}_t$ as follows [21]:

- **prediction:** a sample $\mathbf{x}_{t|t-1}^i$ is generated from each sample $\mathbf{x}_{t-1}^i$ of the set $\mathcal{S}_{t-1}$ using a proposal density function $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t)$. Usually, $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t)$ is equals to $p(\mathbf{x}_t|\mathbf{x}_{t-1})$, the system's dynamics.
- **weight assignment:** each predicted sample $\mathbf{x}_{t|t-1}^i$ is assigned an importance weight w_t^i computed based on the the observation model $p(\mathbf{z}_t|\mathbf{x}_t)$ of the system as $w_t^k = w_{t-1}^k \cdot \frac{p(\mathbf{z}_t|\mathbf{x}_{t|t-1}^i) \cdot p(\mathbf{x}_{t|t-1}^i|\mathbf{x}_{t-1}^i)}{q(\mathbf{x}_{t|t-1}^i|\mathbf{x}_{t-1}^i, \mathbf{z}_t)}$. Once computed, the importance weights are normalized.
- **resampling:** it consists in deleting or duplicating particles according to their weights. This is usually done by generating a new set of particles $\{\mathbf{x}_t^j\}_{j=1}^{N_s}$ from an approximate discrete representation of $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ given by

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \sum_{i=1}^{N_s} w_t^i \delta(\mathbf{x}_t, \mathbf{x}_{t|t-1}^i) \tag{1}$$

  so that $p(\mathbf{x}_t^j = \mathbf{x}_{t|t-1}^i) = w_t^i$. $\delta(.,.)$ is the Dirac delta function. At the end, each resulting particle $\mathbf{x}_t^j$ is assigned a weight $w_t^j = 1/N_s$.

Usually, in order to encourage the state space exploration, the resampling phase is not performed at every time step, but only when the effective size $\widehat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_s}(w_t^i)^2}$ of the filter goes below a given threshold N_T. For more details about PFs, see [21].

4. Agent Based Behavior Tracking

The solution we proposed can be represented by Fig. 1. The simulator is assumed to handle virtual-agent microscopic navigational features and object-agent interactions (e.g., escalators, cash dispensers). It is then used within the prediction step of a particle filtering process to estimate the belief regarding both the aimed location and the activity of the pedestrian. Then, the noisy observation (location of the detected targets obtained after an image analysis process) received from the sensor network is used during the correction step of the filtering. We assume to be aware of areas covered by the sensors. Furthermore, for preserving coherence between the real world and the simulated one, the simulator is fed with changes occurring in the real world such as object states modification (e.g., escalator failures) or exogenous events (e.g., fire alerts). We assumed that the video analysis is performed upstream and it is not part of the discussion in this paper.

In what follows, we describe the different models as required by the Bayesian filter and discuss their implementation. Also, we assume that, although a target may interact with objects within the environment, he can not modify their internal state.

4.1. System Dynamics

Given a behavioral model, a state $\mathbf{x}_t$ of an agent contains attributes that are taken into account within his action-selector mechanism, that is all attributes that may play a role

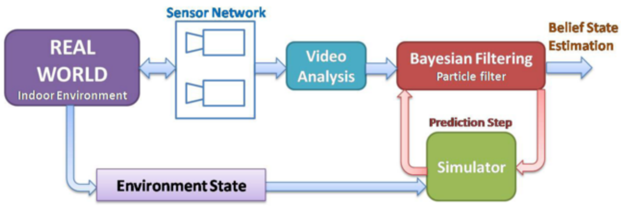

Figure 1. The process overview

in the selection of the actions to be performed by the agent. These attributes can be regrouped into two categories. The first category regroups spatial attributes (e.g., position, velocity) while the second one contains internal attributes representing for example motivations such as psychological and physiological traits (e.g., the thirst level) and resources (e.g., ticket, money) owned. Furthermore, a virtual agent is able to sense his surroundings and builds his proper knowledge of its world (e.g., the object states). This knowledge, combined with its internal variables, represent richer contextual information on which the behavioral model relies for defining self-explanatory agent trajectories; e.g., interrupting his initial plan for getting some drink when thirsty. Therefore, internal attributes may naturally evolve even when the individual is static according to his perceptions.

As the environment state $\mathbf{E}_t$ (including objects) is known and because, as assumed, an agent cannot modify the object states, we only need to consider the agent's dynamics. Since the simulator is in charge of navigational features, the agent's dynamics is fully encoded therein and may be represented as follows:

$$\mathbf{x}_{t+1} \sim f(\mathbf{x}_t, \mathbf{E}_t),$$

where f is the simulator's implemented stochastic function taking as input both the state $\mathbf{x}_t$ of the agent with the environment state $\mathbf{E}_t$ and modifying the agent's inner attributes.

4.2. *Observation Model*

The observation data $\mathbf{z}_t$ received from the sensors depends on whether the agent is within a covered area or not. However, assuming Gaussian noise with a covariance matrix Q_v, the agent may still be undetected even within covered areas, especially when he is close to the boundaries of both areas. The probability φ of such an event can be approximated as the portion of the Gaussian (represented by a circle of radius r_h) belonging to the non-covered areas (Fig. 2). That is

$$\hat{\varphi} = \frac{\sum \text{prob. of region's cells in non-covered area}}{\sum \text{prob. of all region's cells}}. \tag{2}$$

Finally, we have

$$p(\mathbf{z}_t|\mathbf{x}_t) = \begin{cases} 1 & \text{if } \mathbf{z}_t = \varnothing \text{ and } unc(\mathbf{x}_t), \\ \hat{\varphi} & \text{if } \mathbf{z}_t = \varnothing \text{ and } \neg unc(\mathbf{x}_t), \\ \mathcal{N}_{0;Q_v}(\mathbf{u}_t) & \text{if not,} \end{cases}$$

where $\mathbf{u}_t = \mathbf{z}_t - h(\mathbf{x}_t)$, h is a function extracting the location data from $\mathbf{x}_t$, and $unc(\mathbf{x}_t)$ indicates that the agent ($\mathbf{x}_t$) is within a non-covered area. $\varnothing$ means no-detected agent.

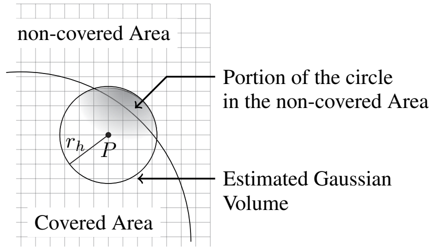

Figure 2. Approximation of φ: P is the agent position. The ratio is computed with respect to all cells in the considered circle.

4.3. Implementation in SE-Star

The simulator used is SE-Star, a Thales proprietary engine capable of modeling adaptive behaviors, navigations and interactions with objects. Each agent is characterized by a motivational tree based on a free flow hierarchy approach [22] containing a set of attributes on which the action selector mechanism relies for computing his current action.

However, SE-Star is a simulator with less random effects in behavior model dynamics and, running several simulations with a given agent state will lead to identical results in terms of behaviors exhibited. Such processes with little noise are not appropriate within a particle filter because of the sample impoverishment phenomenon [21].

Regularized particle filters (RPFs) have been introduced in [23] for preventing this phenomenon. The RPF's idea is to re-sample from a continuous approximation of the probability density function (pdf) $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ instead of its discrete approximation (see Equation 1), hence producing a new particle set with N_s different particles. The continuous approximation of the posterior pdf is computed as follows (see Fig. 3):

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \approx \hat{p}_\lambda(\mathbf{x}_t|\mathbf{z}_{1:t}) = \sum_{i=1}^{N_s} w_t^i.K_\lambda(\mathbf{x}_t - \mathbf{x}_t^i),$$

where $K_\lambda(\mathbf{x}) = \frac{1}{\lambda^{n_x}} K(\frac{\mathbf{x}}{\lambda})$ is the rescaled kernel density, $K(.)$ is a kernel function, λ is the bandwidth and n_x is the dimension of the state space. We use the Gaussian kernel and, according to [24], the corresponding optimal bandwidth is given by $\lambda_{opt} = \left(\frac{4}{(n_x+2)N_s}\right)^{\frac{1}{n_x+4}}$. Also, when the pdf is multi-modal, it is suggested [24] to have $\lambda = \lambda_{opt}/2$.

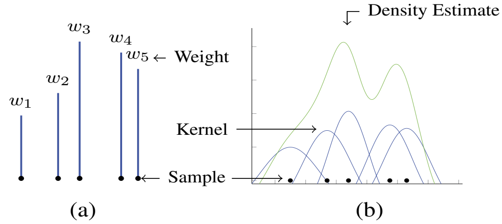

Figure 3. Regularization Process: (a) - Weighted empirical measure; (b) - Regularized measure

[2]Thales Group is a French multinational company that designs and builds electrical systems and provides services for the aerospace, defence, transportation and security markets.

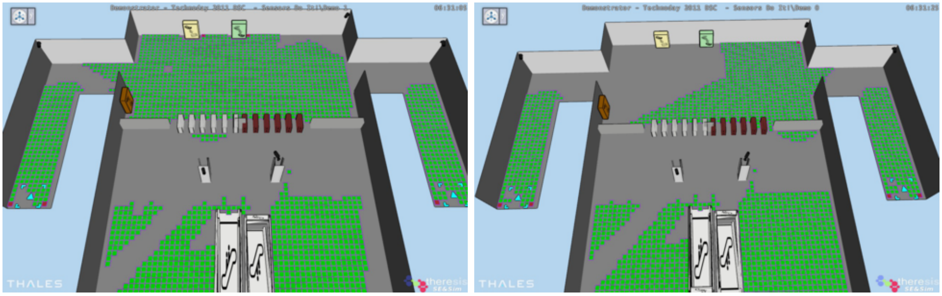

Figure 4. Scenario 1 (left) and 2 (right). Green squares represent areas covered by sensors.

5. Experimental Evaluation

We conducted experiments in a virtual environment representing a two-storey subway station (Fig. 4). The station contains an escalator, a train door, an ATM (in light green), a ticket machine (yellow), a beverage dispenser (brown), ticket barriers (white) and exit barriers (red) and an agent may interact with any of these objects in order to fulfill its objectives. Also, the station is equipped with a camera network set up to not completely cover the environment and configured to report noisy passenger location data.

We consider two scenarios. In the first one, the cameras are set to cover areas occupied by the ATM, the ticket machine and the beverage dispenser. In the second one, these objects are no longer covered by the sensors (Fig. 4) in order to assess the robustness of our approach. A passenger may initially own a certain amount of money and/or a valid ticket and, during his lifetime, may be motivated by three goals or actions: taking a train, drinking or leaving.

Tracking in such an environment is challenging since, depending on the resources owned, which are unknown, a passenger may undertake a sequence of interactions (subgoals) with objects in the scene based on his current motivation, thus affecting his trajectory. For example, a passenger willing to take a train may first buy a ticket if he does not have a valid one. However, buying a ticket will require, if he does not own enough money, to get some cash from the ATM. Furthermore, in non-covered areas, nothing prevents the passenger to switch between motivations and perform several interactions. It is our task to infer, solely from the observed trajectory, the corresponding behavior.

Such a passenger model has been designed in SE-Star in which the three motivations are represented by numerical attributes taking values in $[0, 2]$ and the simulator is in charge of their evolution. Also, the model includes two attributes representing respectively the amount of money and the number of tickets owned.

We run our algorithm with 2000 particles and set r_h to be $0.5m$. The noise standard deviation is set to 0.8, 0.8 and $0.05m$ for the x, y and z coordinates respectively. Initially, a passenger has 30% chances to own a number of tickets (an amount of money) chosen uniformly in the interval $[1, 3]$ ($[1, 5]$) and nothing otherwise. For other attributes, we assume a Gaussian distribution $\mathcal{N}(0.75; 0.5)$.

Next, we refer to behavior inferred from the filter the one exhibited by the majority of the particles within the filter in terms of goal (or subgoal). We consider the **similarity** indicator defined below as a criterion to evaluate the exactness between the passenger real behavior and the inferred one:

Table 1. Results: Similarity, Robustness and Mean Square Errors

	% Non Cov.	Goal Sim.(%)	Goal Rob.(%)	Subgoal Sim.(%)	Subgoal Rob.(%)	General MSE	Goal MSE	SubGoal MSE
Sc. 1	32.7	94.31 (± 6.76)	99.17 (± 0.72)	83.26 (± 7.82)	95.02 (± 6.62)	3.08 (± 2.04)	2.35 (± 1.46)	2.27 (± 1.48)
Sc. 2	67.7	90.25 (± 12.27)	99.46 (± 0.26)	72.41 (± 12.23)	92.20 (± 9.42)	4.29 (± 3.48)	4.78 (± 3.45)	4.49 (± 3.74)

$$Similarity = \frac{\sum_{t=0}^{T} D(g(t), g'(t))}{T},$$

where $g(t)$ and $g'(t)$ are the real and the inferred behavior (goal or subgoal) respectively; $D(.,.)$ returns 1 when the two parameters are equal and 0 otherwise.

Another criterion we consider is the **robustness** which corresponds to the ratio of time the filter contains a valid hypothesis regarding the passenger behavior:

$$Robustness = \frac{\sum_{t=0}^{T} P(g(t))}{T},$$

where $P(g)$ is an indicator of the behavior g's presence in the filter.

Moreover, we compute the mean squared error (MSE) of the location data estimated from the filter with respect to the the position of the passenger. This error is computed according to three such location estimates: general estimate (weighted mean of the location of all the particles), goal-based and subgoal-based estimates (weighted mean of the location of particles whose behavior — goal or subgoal — corresponds to the one exhibited by the filter). Finally, each experimentation has been carried out 10 times and results (mean of all the runs) are reported in Table 1.

For scenario 1, in which the passenger spends 1/3 of the time in non-covered areas, the algorithm has relatively high average scores with low standard deviations, regardless of the criterion. It appears that the filter goal-based similarity and robustness are respectively 94.3% and 99.17$\%$. When focusing on passenger subgoal, these values are respectively 83.26$\%$ and 95.02$\%$. This decrease with respect to goal-based values is explained by the fact that, given a motivation, there exists a variety of sequence of interactions, according to the agent internal attributes, for fulfilling it. Moreover, paths in the environment do not help discriminate between possible interactions (e.g., ATM and ticket machine as they are close to each other).

In scenario 2, a degradation of performance can be observed. This is due to the fact the passenger is undetected most of the time. However, despite these significant occlusions (2/3 of the time), the approach is still efficient and keeps good estimates of the target's behavior; the goal and subgoal based similarities being respectively 90.25 and 72.41. An example of the result for the subgoal estimation is depicted in Fig 5.

In both scenarios, the robustness criterion is high. This is of particular interest in case of long occlusions for subsequently recovering the target's behavior when re-observed.

6. Discussion and Future work

In this paper, we address the problem of pedestrian tracking and behavior understanding simultaneously. Unlike previous works in the literature, we take advantage from richer

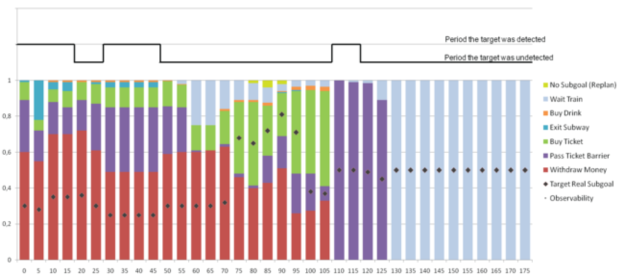

Figure 5. Estimation of the passenger's behavior over a discretized time slot (scenario 2): each bar represents the belief over subgoals (modeled as slices on the bar) by the filter. The black point on each bar represents the actual behavior of the tracked target. The line on top of the bars represents time frames during which the target were either observed or not. We can see that maintaining a valid hypothesis regarding the exact behavior (timesteps 75–105) helps recovering the target's behavior as soon as he is observed again (timestep 110).

contextual information by relying on advanced agent-based behavioral models designed in the field of situated artificial intelligence and corresponding simulators. We integrate within a particle filter such a virtual agent simulator as a predictive block for behavior tracking purposes. We focus on the single target case and evaluate our approach on a virtual environment.

The idea behind evaluating in virtual environments was to perform "witness" experiments in which the behavioral model of real humans is assimilated to a clearly circumscribed model that we perfectly master in such a way to avoid inexplicable phenomena. Indeed, the same behavioral model is used for the real world simulation and within the particle filter in order to verify that, in this extreme case, we actually get a very good prediction. The results detailed in Section 5 show encouraging filtering performances in terms of both behavior (goal/subgoal) and location estimations even in cases of long periods of occlusion.

It would be interesting, in a future work, to confront our system on real scenarios and evaluate the impact of approximating the mechanism of internal decision-making of real humans, which is unknown, by an artificially designed behavioral model. However, when considering the real context, the first step to undertake is the calibration of the real sensor network with the simulator used in the filter. Secondly, we may face the problem of characterizing the duration of real human-object interactions as it may differ from one individual to another given the same object. One solution consists in defining within the simulator, for a given object, a unique duration time representing the average of different interaction durations observed in the real world. Another solution is to include within the artificial behavioral model, an attribute representing the duration of the interaction a given agent is about to perform. The third issue is the realism of the behavioral model with respect to what is generally observed; hence the necessity to work together with experts in behavior analysis for capturing the essence of people's behavior and trajectories within the considered environment. As a very short term objective, we plan to deploy the solution in a Thales office buildings.

The proposed approach presents several advantages. Firstly, by relying on agent-based behavioral models, we clearly emphasize the fact that each individual has its own specificity in terms of reasoning capabilities and strategies. Usually, people do tracking based on simple and stereotyped behaviors (e.g., flow of trajectories). We instead consider finer individual behavior that makes it possible to explain outliers impossible to explain with stereotyped behaviors. A second advantage is the ability to handle, without any supplementary effort, exogenous events or changes that may occur within the environment (e.g., escalator failure, fire alerts). Indeed, such events, as soon as they are perceived by an individual, are automatically taken into account during the decision process of his behavioral model therefore adapting its behavior accordingly and updating the particles in consequence. Finally, there is no need to build any graphical models (e.g., DBNs) representing relationships between variables of the system state.

As a medium-term objective, we will investigate the multi-target case. This is a very challenging problem as, because of the camera network, the data association problem is added on top of the tracking problem. The data association problem consists in determining from which target a received observation comes. Furthermore, the interactions between targets add another layer of complexity.

References

[1] I. Haritaoglu, D. Harwood, and L. S. Davis, "W4: Real-time surveillance of people and their activities," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 809–830, 2000.

[2] J. Krumm, "Multi-camera multi-person tracking for easyliving," pp. 3–10, 2000.

[3] S. Stylianou, M. M. Fyrillas, and Y. Chrysanthou, "Scalable pedestrian simulation for virtual cities," in *Proc. of the Symposium on Virtual Reality Software and Technology*, pp. 65–72, 2004.

[4] W. Shao and D. Terzopoulos, "Autonomous pedestrians," *Graph. Models*, vol. 69, pp. 246–274, 2007.

[5] P. Maes, "Modeling adaptive autonomous agents," *Artificial Life*, vol. 1, pp. 135–162, 1994.

[6] J.-A. Meyer, "From natural to artificial life: Biomimetic mechanisms in animat designs," *Robotics and Autonomous Systems*, vol. 22, pp. 3–21, 1997.

[7] D. Helbing and P. Molnár, "Social force model for pedestrian dynamics," *Phys. Rev. E*, vol. 51, pp. 4282–4286, may 1995.

[8] R. C. Arkin, *Behavior-based Robotics*. Cambridge, MA, USA: MIT Press, 1st ed., 1998.

[9] C. Reynolds, "Steering Behaviors for Autonomous Characters," in *Game Developers Conference 1999*, pp. 763–782, 1999.

[10] S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *Proc. IEEE 12th Int. Conf. on Computer Vision*, pp. 261–268, 2009.

[11] M. Luber, J. Stork, G. Tipaldi, and K. Arras, "People tracking with human motion predictions from social forces," in *proc. of the IEEE Int. Conf. on Robotics and Automation*, pp. 464–469, 2010.

[12] B. Tastan and G. Sukthankar, "Leveraging human behavior models to predict paths in indoor environments," *Pervasive Mobile Computing*, vol. 7, pp. 319–330, jun 2011.

[13] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie, "Behavior recognition via sparse spatio-temporal features," in *Proc. of ICCCN '05*, pp. 65–72, 2005.

[14] J. Liu, J. Luo, and M. Shah, "Recognizing realistic actions from videos "in the wild"," 2009.

[15] A. Bruce and G. Gordon, "Better motion prediction for people-tracking," in *Proceedings of ICRA*, 2004.

[16] D. H. Wilson and C. Atkeson, "Simultaneous Tracking and Activity Recognition using many anonymous, binary sensors," in *Proc. of the 3rd Int. conf. on Pervasive Computing*, pp. 62–79, 2005.

[17] C. Manfredotti, D. J. Fleet, H. J. Hamilton, and S. Zilles, "Simultaneous tracking and activity recognition," in *Proc. of ICTAI'11*, (Washington, DC, USA), pp. 189–196, 2011.

[18] X. Tu and D. Terzopoulos, "Artificial fishes: Physics, locomotion, perception, behavior," in *Proc. of SIGGRAPH*, pp. 43–50, 1994.

[19] J. Funge, X. Tu, and D. Terzopoulos, "Cognitive modeling: Knowledge, reasoning and planning for intelligent characters," in *Proc. of SIGGRAPH*, pp. 29–38, 1999.

[20] S. Sarkka, *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

[21] M. S. Arulampalam, S. Maskell, and N. Gordon, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. on Sig. Processing*, vol. 50, pp. 174–188, 2002.

[22] T. Tyrell, "Defining the action selection problem," 1993.

[23] C. Musso, N. Oudjane, and F. L. Gland, "Improving Regularized Particle Filters," 2001.

[24] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, 1986.