Applying Bayes Markov Chains for the Detection of ATM Related Scenarios

Dejan Arsić, Atanas Lyutskanov, Moritz Kaiser, Björn Schuller and Gerhard Rigoll
Institute for Human Machine Communication
Technische Universität Münchnen
arsic@tum.de

Abstract

Video surveillance systems have been introduced in various fields of our daily life to enhance security and protect individuals and sensitive infrastructure. Up to now it has been usually utilized as a forensic tool for after the fact investigations and are commonly monitored by human operators. In order to assist these and to be able to react in time, a fully automated system is desired. In this work we will present a multi camera surveillance system, which is required to resolve heavy occlusions, to detect robberies at ATM machines. The resulting trajectories will be analyzed for so called Low Level Activities (LLA), such as walking, running and stationarity, applying simple but robust approaches. The results of the LLA analysis will subsequently be fed into a Bayesian Network, that is used as a stochastic model to model so called High Level Activities (HLA). Introducing state transitions between HLAs will allow a temporal modeling of a complex scene. This can be represented by a Markovian process.

1. Introduction

One of the major aspects of automated visual surveillance systems is to detect objects in the scene and track these over time. The most challenging problem herein is to segment people in complex scenes, where high object density leads to occlusions. To model individual behaviors these have to be resolved robustly. Tracking techniques based on a single view, such as the mean shift algorithm [10], are able to track objects robustly, but require an initialization of single objects prior to the group formation and the subsequent handling of merge and split events [16]. However, in some cases a single view seems not sufficient to detect and track objects due to severe occlusion, which as a fact requires the utilization of multiple camera views.

Camera networks are frequently applied to extend the limited field of view of a camera, performing tracking in each sensor separately and fusing this information [1]. In order to deal with dense crowds, the cameras should be mounted to view defined regions from different perspectives. Within these, corresponding objects now have to be located. Approaches based on geometrical information rely on geometrical constraints between views using calibrated data [25] or homography between uncalibrated views, which e.g. Khan [15] used to localize feet positions. This approach, though very simple and effective, localizes feet and consequently tends to segment persons into further parts. This can be avoided by applying multi layer homography, as proposed in [3], which is capable to create a 3D representation of the scene. The detected object locations can now be utilized as initialization for any tracking approach.

Having associated the single detections to trajectories, it is possible to analyse a person's behavior and additionally detect anomalies on-line. This step is basically the most important one, as it moves the system from passive CCTV, that is used for forensics, to an active system that enables security staff to react in time and even prevent crimes. In this work we will focus on the detection of robberies at ATM machines, as these seem to occur quite frequently in unsecure urban regions. Therefore a system, which analyzes an individual person's behavior on a low level activity basis in the first place and combines observations in a Bayesian Network will be introduced in this work. We will show, that this static representation will robustly detect Higher Level Activities, without the cost of collecting a large amount of data to train HMMs [18] or behavioral maps [6]. Despite the scenario's complexity and large inter class variance, some scenarios are though following a similar scheme, which can be modeled by a Markov chain architecture. This is achieved by the introduction of state transition between HLAs, allowing a detailed dynamic scene representation. Observing this we are able to detect scenarios with feeding expert knowledge into the network structure.

The performance of this approach will be demonstrated on the PROMETHEUS data set [17], which has been created for the comparison of tracking and behavior detection systems, and introduced new sensors such as thermal infrared, 3D cameras and also used audio. Ground truth is provided both for the person locations and associated events.

978-1-4244-5496-9/09/$25.00 ©2009 IEEE

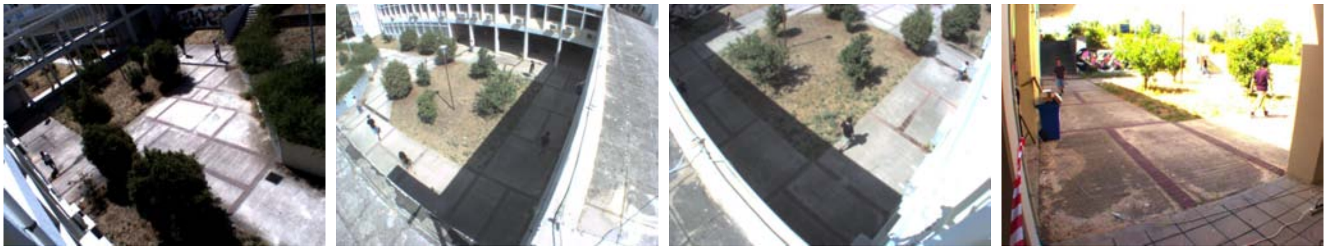


Figure 1. All four views of the PROMETHEUS outdoor scenario

2. The PROMETHEUS ATM Corpus

One of the integral parts of the PROMETHEUS corpus is the security related outdoor scenario. It has been recorded in an outdoor facility using three synchronized overview Firewire cameras with a resolution of 1076×768pixels. These were utilized to track persons along the paths and the lawn in the scene. The cameras were setup respecting the scene geometry, in order to resolve occlusions created by trees and bushes. Furthermore lenses with a short focal length have been installed, to enlarge the field of view. Additionally a detail camera with PAL resolution has been installed at the ATM, providing a more detailed view on the relevant region. This way even the persons limbs could be modeled. Furthermore a photonic mixture device, that creates a depth image of the scene, has been used in in front of the ATM, which can be used to resolve occlusions in dense environments.

As the recordings were conducted in a public place multiple people and groups could be observed in the video material. Eleven actors have been engaged to simulate both luggage [13] and ATM related events, which will be addressed in this work. Therefore actors were told to draw money at a simulated ATM machine and eventually cue in line behind a person operating the ATM. Throughout the three hours of video material the behavior of operating the ATM has been recorded twelve times, whereas only three robberies occurred. While an actor was drawing money, in some cases another actor has been instructed to rob the person at the ATM. Therefore the robber would approach the person, grab the money or hand bag and run away into a random direction. As the actors did not know, when they might be robbed the reaction was quite spontaneous and various reactions have been observable. Some were shouting and following the thief, others were just standing in front of the ATM and screaming for help. Screams have been recorded by a microphone array behind the ATM, although audio is not used in this part of the work.

In order to be able to evaluate the system's performance, the entire amount of one hour of video material has been manually annotated. Thereby the persons' position has been determined for every fifth frame in the sequence in world coordinates. Furthermore the timestamps of ATM incidents have been also annotated. The database is available for research purposes. For further details please contact the corresponding author.

3. Multiple Camera Person Tracking

In the first stage a synchronized image acquisition is needed, in order to compute the correspondences of moving objects in the corresponding views $C_1, C_2, \ldots, C_n$. Additionally the sensors should be set up keeping in mind that the observed region should be as large as possible and direct occlusions of the sensor should be avoided. Therefore a field of view looking down on the scenery from an elevated point would be preferable.

Subsequently a foreground segmentation is performed in all available smart sensors to detect changes from the empty background BG [15] :

$$FG_n(x,y,t) = I_n(x,y,t) - BG_n(x,y) \tag{1}$$

where the appropriate technique to update the background pixel, here based on Gaussian Mixture Models [26], is chosen for each sensor. It is advisable to set parameters, such as the update time, separately in all sensors to guarantee a high performance. Computational effort is reduced by masking the images with a predefined tracking area. Now the homography H_i between a pixel p_i in the view C_i and the corresponding location on the ground plane π can be determined. In all views the observations $x_1, x_2, \ldots, x_n$ can be made at the pixel positions $p_1, p_2, \ldots, p_n$. Let X resemble the event that a foreground pixel p_i has a piercing point within a foreground object with the probability $P(X|x_1, x_2, \ldots, x_n)$. With Bayes' law

$$P(X|x_1, x_2, \ldots, x_n) \propto P(x_1, x_2, \ldots, x_n|X)P(X) \tag{2}$$

the first term on the right side is the likelihood of making an observation $x_1, x_2, ..., x_n$ given an event X happens. Assuming conditional independence, the term can be rewritten to

$$P(x_1, \ldots, x_n|X) = P(x_1|X) \times \ldots \times P(x_n|X) \tag{3}$$

According to the homography constraint, a pixel within an object will be part of the foreground object in every view

$$P(x_i|X) \propto L(x_i) \tag{4}$$

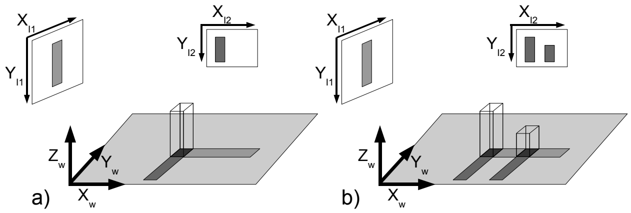


Figure 2. a) Planar homography for object detection.b) Resolving occlusions by adding further views.

where $L(x_i)$ is the probability of x_i belonging to the foreground. An object is then detected in the ground plane when

$$P(X|x_1, x_2, \ldots, x_n) \propto \prod_{i=1}^{n} L(x_i) \tag{5}$$

exceeds a threshold θ. In order to keep computational effort low it is feasible to transform only regions of interest. These are determined by thresholding the entire image, resulting in a binary image, before the transformation and the detection of blobs with a simple connected component analysis. This way only the binary blobs are transformed into the ground plane instead of probabilities. Therefore eq. 5 can be simplified to

$$P(X|x_1, x_2, \ldots, x_n) \propto \sum_{i=1}^{n} L(x_i) \tag{6}$$

without any influence on the performance. The value of theta θ is usually set dependent on the number n of camera sensors to $\theta = n - 1$, in order to provide some additional robustness in case one of the views accidentally fails. The thresholding on sensor level has a further advantage compared to the so called *soft threshold* [15], [7], where the entire probability map is transformed and probabilities are actually multiplied as in eq. 5. A small probability or even $x_i = 0$ would result in a small overall probability, whereas the thresholded sum is not affected that dramatically. Using the homography constraint hence solves the correspondence problem in the views $C_1, C_2, \ldots, C_n$, as illustrated in fig 2a) for a cubic object. In case the object is human, only the feet of the person touching the ground plane will be detected. The homography constraint additionally resolves occlusions, as can be seen in fig. 2a). Pixel regions located within the detected foreground areas, indicated in grey on white ground and representing the feet, will be transformed to a piercing point within the object volume. Foreground pixel not satisfying the homography constraint are located off the plane, and are being warped into background regions of other views. The piercing point is located outside the object volume. All outliers indicate regions with high uncertainty, as there is no depth information available. This limitation can now be used to detect occluded objects. As visualized in fig. 2b) the smaller cuboid is occluded by the large one in view C_1, as apparently foreground blobs are merged. The smaller object's bottom side is occluded by the larger object's body. In contrast both objects are visible in view C_2, resulting in two detected foreground regions. A second set of foreground pixel, located off the ground plane π, in view C_1 will now satisfy the homography constraint and localize the occluded object. This process allows the localization of feet positions, although they are entirely occluded, by creating a kind of see through effect.
The implemented algorithm can be described as following:

- Foreground objects ψ_{in} are detected in all n views and a binary map is created. Subsequently n object boundaries can be extracted utilizing connected components analysis in the binary image

- Object boundaries are then being transformed into a predefined reference view

$$\Psi_{in} = \mathbf{H}\psi_{in}. \tag{7}$$

  Though any of the views can be chosen, the most convenient one is a top view on the ground plane, visualizing spatial relationships between objects.

- Next the intersections of the polygons are computed. These can be calculated by a plane-sweep algorithm within the reference view. The binary represented regions B_n

$$B_n(x,y) = \left\{ \begin{array}{c} 1 \text{ if } P_n(x,y) \in \Psi_{in} \\ 0 \text{ else} \end{array} \right\} \tag{8}$$

  located within detected foreground, are now transformed into the ground plane. In a subsequent step these values are summed up to

$$B(x,y) = \sum_{i=1}^{n} B_i(x,y). \tag{9}$$

- The resulting map B(x,y) is subsequently thresholded with the previously defined parameter θ to encounter possible object regions

$$S(x,y) = \left\{ \begin{array}{c} 1 \text{ if } B(x,y) \geq \theta \\ 0 \text{ else} \end{array} \right. \tag{10}$$

  This is usually computed with $\theta = n - 1$ to obtain higher reliability in the tracking process.

- Finally coherent regions indicating feet positions are indexed applying a simple connected component analysis.

This procedure can be repeated for multiple heights besides the ground layer to create a 3D view of the scenery, which

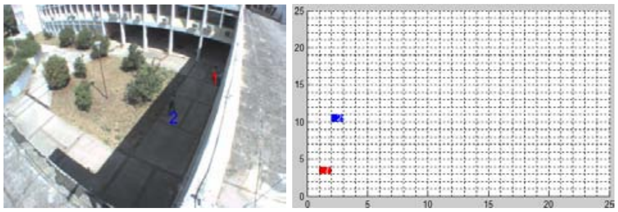


Figure 3. Exemplary labeling of persons present in the scene on the left. Occupancy of the ground floor on the right

will help rising the localization performance drastically [3]. Fig. 3 illustrates the localization results for two persons in the scene, which can be directly labeled either in a field of view or in a simplified occupancy map, that illustrates a top view of the scenery.

As seen in fig. 3, there are bushes and trees present in the scene, which occlude objects walking along the paths. In quite extreme combinations a person will only be visible in one or two camera views, and hence she will not be detected by the homography algorithm. Setting a lower threshold for required overlaps of blob transformations, is not an adequate solution, as it will have an effect on the entire scene and create multiple false positives. Therefore it has been decided to incorporate contextual knowledge into the scene. All obstacles are manually labeled, their homographies are computed and assigned a unique label to determine which camera is responsible for the homography.

4. Low-Level Trajectory Events

Human behavior is considered as very complex and a wide range of varieties can be observed for each individual behavior. Therefore it is frequently not possible to create one meaningful model for a complex activity. In contrast to a task like gesture recognition, which is quite limited in appearance, it has been suggested to decompose complex scenarios into common and simple to detect so called Low Level Activities (LLA) [4] or pre-defined indicators (PDIs) [8]. These can subsequently be further analyzed after being detected. Following we will shortly describe the employed LLAs and how these are robustly detected by simple means.

Stationary Object Detection:

For some scenarios, such as left luggage detection, objects not altering their spatial position have to be picked up in a video sequence. Due to noise in the video material or slight changes in the detector output, e.g. the median of a particle filter, the object location is jittering a little. A simple spatial threshold over time is usually not adequate, because the jitter might vary in intensity over time. Therefore the object position is averaged over the last N frames:

$$\overline{o_i} = \frac{1}{N} \sum_{t'=t-N}^{t} o_i(t') \qquad (11)$$

Subsequently the normalized variance in both $x-$ and $y-$ direction

$$\sigma_i(t) = \left| \frac{1}{N} \sum_{t'=t-N}^{t} (o_i(t') - \overline{o_i})^2 \right|_2 \qquad (12)$$

is computed [5] [2]. This step is required to smooth noise created by the sensors and errors during image processing. Stationarity can then be assumed for object with a lower variance than a predefined threshold θ

$$\text{stationarity} = \begin{cases} 1 \text{ if } var < \theta \\ 0 \text{moving else} \end{cases} \qquad (13)$$

Given only the location coordinates this method does not discriminate between pedestrians and other objects, enabling the stationarity detection for any given object in the scene.

Detection of Loitering Persons According to authorities a person would observe a scene for a while until the supposable right point of time appears prior to performing a threat. This is depending on external circumstances, which have to be met. Observations are frequently performed from a well-defined place in the scenery, where the person tries not to draw attention to himself, requiring steady movement in a crowded environment. Therefore it is important to monitor the visibility of pedestrians in sensitive areas. The PETS2007 challenge defines loitering as a subject being located in the field of view more than a predefined hard time threshold, here $\theta_{time} > 60\,s$ [13].

This kind of behavior can be easily solved with a rule based approach implemented into the person tracking modules [2]. While tracking an individual object the *age*, meaning the time an object is visible in the scene, can be determined by simply counting the frames an object track is maintained. Tracks older than θ_{time} will trigger an alarm. Analysis has been performed only on blob level, not discriminating between objects and pedestrians. The integration of a luggage piece detector as presented in [11] or a pedestrian detection system [19] could eliminate false positives. **Discriminating Between Walking and Running**

In the past various gait recognition systems [23],[9] based on machine learning techniques have been designed to recognize pedestrians from gait or discriminate between different kind of gait, such as walking and running. These are commonly trained with 2D data acquired from a predefined field of view, which cannot be granted in every real world scenario. retraining these algorithms for every possible system setup is a rather expensive task, as video material has to be collected and annotated. Considering the trajectories projected in a virtual top view a human operator would probably be analyzing the object's speed to discriminate between walking and running. This observation is utilized in this work. Defining walking as movement up to a maximum speed, here $v_{max} = 6\,km/h = 1.66\,m/s$, and faster movements as running simple thresholding operation can be performed

$$s(t) = \begin{cases} \text{walking if } v_i(t) < v_{max} \text{ and } \overline{stationary} \\ \text{running if } v_i(t) > v_{max} \end{cases} \tag{14}$$

The speed $v_i(t)$ can be easily computed with the covered distance

$$d = \sqrt{(x(t) - x(t-1))^2 + (y(t) - y(t-1))^2} \tag{15}$$

in meters and the frame rate of the captured video.

Once again jitter in the detection process is flattened by averaging the frame based results over time. Experience has shown that the summation of up to 25 frames is sufficient for this task. While the discrimination between walking and running relies solely on the covered distance, the direction of motion can be simply computed by the difference between two adjacent positions $\vec{v}_t = \vec{x}_t - \vec{x}_{x-1}$.

Detection of Splits and Mergers

According to Hu [20] so called splits and merges have to be detected in order to maintain IDs in the tracking task. Guler [14] tried to handle these as low level events describing more complex scenarios, such as people getting out of cars or forming crowds. A merger usually appears in case two previously independent objects o_1 and o_2 unite to a mostly bigger one

$$o_{12} = o_1 \cup o_2 \tag{16}$$

This observation is usually made if two objects come extremely close to each other or touch one another in 3D, whereas in 2D a partial occlusion might be the reason for a merger. In contrast two objects o_{11} and o_{12} can be created by one single splitting object o_1, which might be created by a previous merger.

While others analyze object texture and luminance [22], the herein applied rule based approach only relies on the object position and the region's size. Basically disappearing and appearing objects have to be recognized during the tracking process, to incorporate a split or merge:

- **Merge:**One object disappears but two objects can be mapped on one and the same object during tracking. In an optimal case both surfaces would intersect with the resulting bigger surface $o_1 \cap o_{12} \& o_1 \cap o_{12}$

- **Split:**Similar to the object split two objects at frame t are mapped to one object at time $t-1$, where the objects both intersect with the old splitting one $o_1 1 \cap o_1 \& o_1 2 \cap o_1$

Detection of Group Movements

As in various cases persons are interacting with each other it seems reasonable to model combined motions. This can be done according to the direction of movement, proximity of objects and velocity. As the direction of motion can be simply computed, it is possible to elongate the motion vector $\vec{v}$ and compute intersections with interesting objects or other motion vectors. Further the distance between object positions can be easily detected with $d_{ij} = \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2}$. Thereby most relevant LLAs can be detected applying simple heuristics, as already employed for left luggage detection [5]. Among the required activities following need to be detected:

- **Approaching a stationary object or person:** The mean motion vector is simply elongated and intersections with stationary persons or objects are computed. If an intersection is detected and maintained for a time $t > \theta$, the person is approaching a stationary object.

- **Two persons walking or standing next to each other:** The distance between all objects in the scene is computed continuously over time. In case the distance is constant over time, allowing some variance of course, or getting smaller over time and are heading into the same direction with the same speed, the objects are considered walking or standing next to each other.

- **A person following another one:** Two persons are heading in the same direction for a time $t > \theta$ for a pre-defined time.

- **Two persons approaching each other:** The distance of two persons is getting smaller over time and the elongated motion vectors are intersecting at any time.

Utilizing this simple rules it is possible to model all cases in a simple, yet effective fashion.

5. Bayesian Network Based Modeling of HLAs

Bayesian Networks (BN) have already been used to analyze behaviors in the past, as these are capable to model dependencies between variables [12, 4]. Such a network can be interpreted as directed acyclic graph, where the nodes represent the state variables X and the edges represent the conditional of nodes and their parent nodes. Thereby all state variables $X_1, \ldots, X_n$ are described by a previously detected LLA. A BN can be completely described in structure and conditional probabilities by its joint probability distribution. Let N denote the total of random variables, and the distribution can be calculated as

$$P(X_1, \ldots, X_N) = \prod_{i=1}^{I} P\left(X_i|\text{parents}(X_i)\right). \tag{17}$$

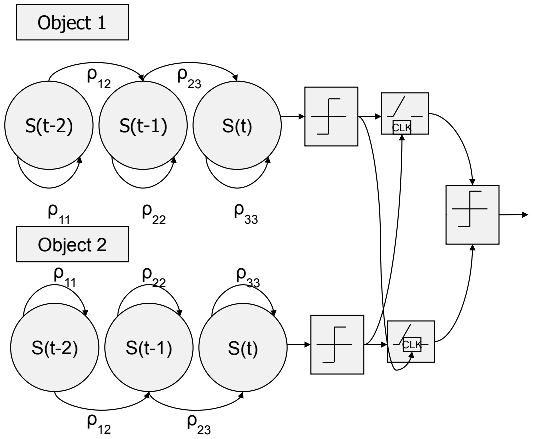


Figure 4. Structure of a Markov chain with state transitions for two persons, who are interfering after some time.

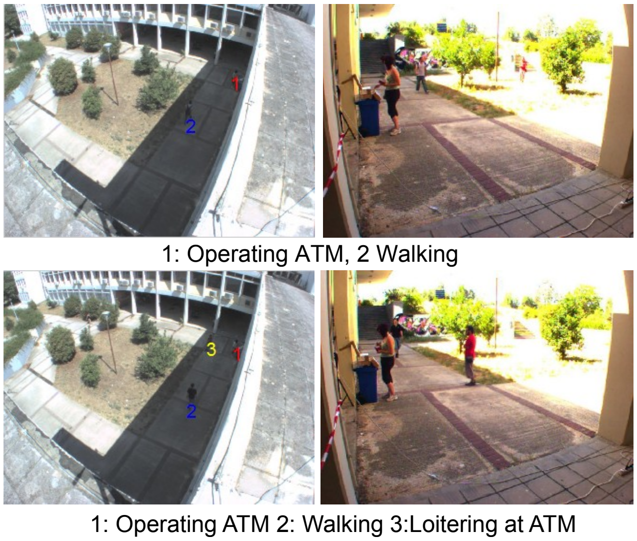
The used BN in this work is enhanced by the capability to handle soft evidences. The relationship between the LLAs can now be used to describe a High Level Activity (HLA). A HLA, as e.g. observing the ATM, can therefore be modeled by a wide range of indicators, which would be standing near the ATM, loitering near the ATM, or approaching the ATM, meaning that the distance to the ATM is becoming smaller. Utilizing the LLAs, context knowledge, object distances and velocity following HLA can be modeled by BNs:

- Operating an ATM
- Loitering in the region of an ATM
- Approaching an ATM while being operated
- Being approached while operating an ATM
- Leaving the ATM in normal pace
- Running Away from ATM
- Leaving Luggage

The major advantage of this approach is its simplicity and the ability to incorporate training material and expert knowledge. As data is usually rather sparse and data collection is quite expensive, this feature should be taken into account. This way the recognition of HLAs, which are easily recorded, can be trained, while others can be determined by a set of predefined probabilities.

6. Event Recognition With Dynamic Bayesian Networks

A complex scenario unfortunately cannot be described by relying on simple BNs, as these are not designed to model temporal relationships. Therefore it seems reasonable to create a dynamic model, in order to recieve a more complex scenario description. The most obvious solution would be to use the HLAs as states within a Markov chain, the probably simplest form of a Hidden Markov Model HMM [21]. A probability is given for every transition between two subsequent states, allowing for auto transitions, in case the actual state does not change. A complex scenario can hence be modeled as a sequence of observations, which have been recognized by the previously trained Bayesian networks. If a new HLA is detected, the most probable path through a set of Markov chains is computed, in order to recognize the given scenario.


1: Operating ATM, 2 Walking

1: Operating ATM 2: Walking 3:Loitering at ATM

Figure 5. Detection examples for operating an ATM machine, once with and once without cuing.

The process of drawing money at the ATM could therefore be represented by observing the ATM as the person had to wait in line, approaching the ATM and operating it and finally leaving the machine. An exemplary result for ”operating ATM” is provided in fig. 5. Such models can now be created for any given scenario.

Finally a model for an ATM robbery has to be created. One obvious implementation of a Markov chain would probably be that the person, that operated the ATM, leaves it in a hurry and runs away in order to follow the thief. Unfortunately there is no evidence why the person is actually running away. This can only be gained if other activities than the robbed person’s one are analyzed. Therefore a second Markov chain is evaluated in parallel. This is used to model the potential theft. Thereby the thief would observe the ATM, approach the person at the ATM, stand very close to her and even merge with her before walking/running away from the crime site. On the other hand this could of course also describe someone meeting another person at an ATM.

At this place an inference, as illustrated in fig. 4, between two concurring chains has been introduced. Thereby both outputs are fed into the other network and the resulting sum

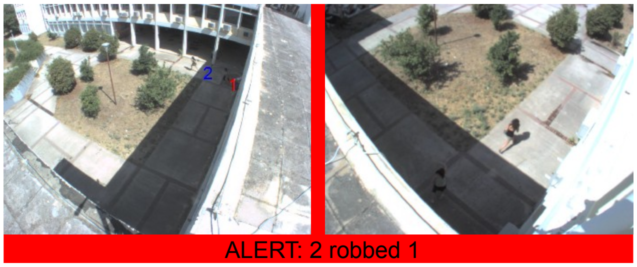


Figure 6. Recognition examples for a robbery at an ATM. Here ID 1 is robbing ID 2.

is analyzed by a transfer function. If a value larger than a predefined threshold θ is observed, an alert is produced. This method can of course be used to model other events that follow a sequential order with small adoptions.

7. Evaluation

As the performance of the homography tracking approach has been evaluated in previous works, this short evaluation focuses on the event detection abilities. The localization precision has once more be confirmed to be appx. $0.15m$, which is an acceptable value for the human class. Furthermore only few ID changes have been observed, due to simplicity of the tracking scenario. In order to evaluate the behavior detection module, events have been annotated manually, while only HLAs and few LLAs have been considered. Tab. 1 shows the activities of interest. In the first place some LLAs, here loitering, stationarity and running, were considered as meaningful LLAs for the scenario recognition task. These could be recognized flawlessly with little to none false positives. The HLAs queuing at ATM, operating ATM and the complex scenario robbing an ATM were basically the most important scenarios.

Tab. 1 illustrates the recognition results. All 17 person operates ATM scenarios have been detected with an average delay of $1.7s$. Only two false positive has occurred, as persons were walking by the ATM and were standing there for a prolonged time period. Loitering at the ATM has also been detected flawlessly, while creating only one false positive. The large amount of loiterings can be explained by persons waiting in line or just observing the scene although not being involved in the scenario. These can be basically considered as correct detections. Nevertheless the loitering and queuing persons could be discriminated quite well in the end as all 15 events have been recognized with only two false positives. As can be seen all four ATM robberies have been flawlessly detected without any false positives and a very short reaction time of 1.1 s. Besides the detection of the event itself the temporal alignment t has been of great interest, as a real application requires short reaction times. The delay of detection and incident has an average 1.1 s, which could be computed by the delay in frames, as cameras with $15fps$ are used. Furthermore two left luggage events have been recorded for test purposes, which have also been detected without error.

Event	[#]	det	fpos	Δt
Loitering	48	48	1	$0\,s$
Stationarity	3	3	0	$0\,s$
Sprint/Run	8	8	0	$1.4\,s$
Left Luggage	2	2	0	$1.2\,s$
Operate ATM	17	17	2	$1.7\,s$
Queuing at ATM	15	15	3	$2.1\,s$
Rob ATM	3	3	0	$1.1\,s$

Table 1. Evaluation of the behavior detection module. All ATM related events could be recognized flawlessly

As the examples in fig. 5 and 6 illustrates the tracking results are only visualized in one view. This is used as overview camera during tracking evaluation. Nevertheless it would be possible to transform them into any other view. For a more convenient visualization of the events, the FOV on the left hand side is chosen dynamically. In case the thief leaves the FOV of the detail camera, the system automatically switches to the best camera perspective, by simply analyzing the persons direction and position in the plane.

8. Conclusion and Outlook

We have presented an integrated framework for the detection of ATM related events in a multi camera surveillance system in this work. The tracking part has been conducted using multi layer homography, which has created reliable results in previous applications already. Nevertheless tracking performance can be further enhanced by creating a 3D model of the person using texture information and implementing a parallel tracking of texture and blob position [3]. Furthermore the introduction of other sensors, such as 3D cameras or thermal infrared, could provide a more reliable segmentation of the scene.

Further it has been demonstrated, that a complex behavior can be decomposed into multiple easy to detect LLAs, where especially heuristics has shown high reliability without the cost of an expensive training phase. The detected LLA are subsequently fed into a Dynamic Bayesian Network, allowing a stochastic model of behaviors. all ATM related events, which are dynamic processes, could be reliably detected. For future development it would be desired to analyze persons in further detail and for instance recognize even gestures [24], which will allow an exacter model creation.

References

[1] J. K. Aggarwal and Q. Cai. Human motion analysis: A review. *Journal on Computer Vision and Image Understand-*

ing, 73(3):428–440, 1999.

[2] D. Arsić, M. Hofmann, B. Schuller, and G. Rigoll. Multi-camera person tracking and left luggage detection applying homographic transformation. In *Proceedings Tenth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS 2007, Rio de Janeiro, Brazil*, Oct. 2007.

[3] D. Arsić, N. Lehment, E. Hristov, B. Hörnler, B. Schuller, and G. Rigoll. Applying multi layer homography for multi camera tracking. In *Proceeedings Second ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC2008, Stanford, CA, USA*, sep 2008.

[4] D. Arsić, F. Wallhoff, B. Schuller, and G. Rigoll. Video based online behavior detection using probabilistic multi-stream fusion. In *Proceedings IEEE International Conference on Image Processing (ICIP) 2005, Genoa, Italy*, pages 606–609, Sept. 2005.

[5] E. Auvinet, E. Grossmann, C. Rougier, M. Dahmane, and J. Meunier. Left-luggage detection using homographies and simple heuristics. In *Proceedings of the ninth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS 2006, IEEE, New York, NY, USA*, Oct. 2006.

[6] J. Berclaz, F. Fleuret, and P. Fua. Multi-camera tracking and atypical motion detection with behavioral maps. In *The 10th European Conference on Computer Vision, October 12-18, 2008, Marseille, France*, October 2008.

[7] A. Broadhurst, T. Drummond, and R. Cipolla. A probabilistic framework for space carving. In *Proceedings. Eighth IEEE International Conference on Computer Vision, ICCV 2001*, pages 388–393, 2001.

[8] N. L. Carter and J. M. Ferryman. The safee on-board threat detection system. In *International Conference on Computer Vision Systems*, pages 79–88, May 2008.

[9] D. Chen, H. M. Liao, and S. Shih. Continuous human action segmentation and recognition using a spatio-temporal probabilistic framework. In *ISM '06: Proceedings of the Eighth IEEE International Symposium on Multimedia*, pages 275–282, Washington, DC, USA, 2006. IEEE Computer Society.

[10] D. Comaniciu, V. Ramesh, and P. Meer. Real-time tracking of non-rigid objects using mean shift. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition, Hilto Head Island, SC, USA*, volume 2, pages 142–149, 2000.

[11] D. Damen and D. Hogg. Detecting carried objects in short video sequences. In *Proceedings of the 10th European Conference on Computer Vision, ECCV 2008, Marseille, France*, pages 154–167, 2008.

[12] F. V. F. V. Jensen. *Bayesian Networks and Decision Graphs*. Information Science and Statistics. Springer, July 2001.

[13] J. Ferryman and D. Tweed. An Overview of the PETS 2007 Dataset. In *Proceedings Tenth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS 2007, Rio de Janeiro, Brazil*, October 2007.

[14] S. Guler. Scene and content analysis from multiple video streams. In *AIPR '01: Proceedings of the 30th on Applied Imagery Pattern Recognition Workshop*, page 119, Washington, DC, USA, 2001. IEEE Computer Society.

[15] S. Khan and M. Shah. A multiview approach to tracking people in crowded scenes using a planar homography constraint. In *Proceedings of the 10th European Conference on Computer Vision, ECCV 2006, Graz, Austria*, pages 133–146, 2006.

[16] W. Niu, J. Long, D. Han, and Y.-F. Wang. Human activity detection and recognition for video surveillance. In *IEEE International Confenrence on Multimedia and Expo, Taipei, Taiwan*, pages 719–722, June 2004.

[17] S. Ntalampiras, D. Arsić, A. Störmer, T. Ganchev, I. Potamitis, and N. Fakotakis. Prometheus ddatabase: A multi-modal corpus for research on modeling and interpreting human behavior. In *Proceedings 16th IEEE International Conference on Digital Signal Processing, DSP2009, Santorini, Greece*, 2009.

[18] N. Oliver, B. Rosario, and A. Pentland. A bayesian computer vision system for modeling human interactions. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 22(8):831–843, 2000.

[19] C. Papageorgiou and T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000.

[20] A. G. A. Perera, C. Srinivas, A. Hoogs, G. Brooksby, and W. Hu. Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *CVPR '06: Proceedings of the 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 666–673, Washington, DC, USA, 2006. IEEE Computer Society.

[21] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 77, pages 257–286, 1989.

[22] S. Vigus, D. Bul, and C. Canagarajah. Video object tracking using region split and merge and a kalman filter tracking algorithm. In *Proceedings International Conference On Image Processing, ICIP2001, Thessaloniki, Greece*, volume x, pages 650–653, october 2001.

[23] L. Wang. Abnormal walking gait analysis using silhouette-masked flow histograms. In *Proceedings of the 18th International Conference on Pattern Recognition*, pages 473–476, Washington, DC, USA, 2006. IEEE Computer Society.

[24] C. Wu and H. Aghajan. Model-based human posture estimation for gesture analysis in an opportunistic fusion smart camera network. *Proceedings IEEE Conference on Advanced Video and Signal Based Surveillance, AVSS2007*, pages 453–458, Sept. 2007.

[25] Z. Yue, S. Z, and R. Chellappa. Robust two-camera tracking using homography. *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP2004, 17-21 May 2004, Montreal, Quebec, Canada*, 3:1–4, May 2004.

[26] Z. Zivković. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings 17th IEEE International Conference on Pattern Recognition, ICPR'04*, Washington, DC, USA, 2004.