Semi-Interactive Tracing of Persons in Real-life Surveillance Data

Michael J. Metternich
Intelligent Systems Lab Amsterdam
University of Amsterdam
Science Park 107
1098 XG Amsterdam
The Netherlands
m.j.metternich@uva.nl

Marcel Worring
Intelligent Systems Lab Amsterdam
University of Amsterdam
Science Park 107
1098 XG Amsterdam
The Netherlands
m.worring@uva.nl

ABSTRACT

To increase public safety, more and more surveillance cameras have been placed over the years. To deal with the resulting information overload many methods have been deployed, focusing either on real-time crime detection or post-incident investigation. In this paper we concentrate on post-incident investigation i.e. crime reconstruction using video data. For a complete crime reconstruction, the location of all persons of interest should be known before and during the incident. To do so, we follow persons within the field of view of a single camera (tracking) and between different cameras (tracing).

We present a semi-interactive approach to post-incident investigation. This method is specifically capable of tracking and tracing persons of interest. Our system supports the analytical reasoning process of the investigator with automatic analysis, visualization methods, and interaction processing. We show that the automatic tracing method significantly speeds up tracing of persons with clear visual characteristics. Tracing of persons without obvious characteristics is an inherently difficult task, but we show that intelligent use of interactive methods greatly improves the tracing performance of our system.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*Perceptual reasoning; Intensity, color, photometry, and thresholding; Representations, data structures, and transforms*

General Terms

Security, Performance

Keywords

Real-life surveillance, person matching, relevance feedback

MiFOR'10, October 29, 2010, Firenze, Italy.

1. INTRODUCTION

Over the last few decades, more and more surveillance cameras have been used to increase the security of public areas. Unfortunately, due to the resulting abundance of data, a lot of potentially useful information cannot be used. To deal with the information overload, many methods have been deployed, which can in general be divided into two categories: real-time crime detection and post-incident investigation. For real-time crime detection, automatic detection of suspicious behavior helps an observer focus on certain cameras of interest [14]. In this paper, we focus on post-incident investigation, which main objective is to create a complete reconstruction of an incident.

Commonly in police investigations, the investigator has the task to provide answers to the so called W4 questions: *Who* were present? *Where* were they seen? *When* were they seen? and *What* were they doing? For this, videos from all relevant surveillance cameras in the area of an event are secured. These videos are then combined with other forms of information such as witness reports, cell-phone information and geographical information. Traditionally, the W4 questions are answered by an investigator who manually searches through all this data. Such an approach is obviously prone to errors and time consuming. To aid the investigator, various interactive systems are designed to continuously provide all necessary information. A very important feature of such a system is the ability to follow persons within a cameras field-of-view (tracking) and between different cameras (tracing). In literature, this problem is addressed using either fully interactive approaches [11, 15] or fully automatic approaches [1, 6, 9].

To reconstruct an event using interactive techniques, the user must have an accurate visualization of the data. Janoos et al. [11] accomplish this by showing activity maps, which provide an overview of the general movement within cameras and easy access to any anomalies. Livnat et al. focus in [15] entirely on the correlations between events. They automatically detected events and represented these by dots on a circle. Correlations between events are then shown visualized using colored lines.

True automatic recognition of a person, for example from a database of wanted criminals, is possible to some extent in high-resolution images or under lab conditions. In practice this is infeasible however, due to the low resolution of real-life surveillance cameras. The best we can do in these cases is match a selected person with detected persons in

other cameras. Dollár et al. [5] showed that the most reliable method to detect persons in real-life surveillance data is Histograms of Oriented Gradients [4]. We therefore use this method as the basis for our automatic person detection system. Matching of detected persons can be done using various techniques; e.g. Alahi et al use a cascaded set of color histograms [1], and Farenza et al. use a combination of chromaticity, spatial and recurrence information [6]. Gray et al [8, 9] have assembled the VIPeR dataset for this specific purpose and compared various standard techniques in this setting. In previous work [17] we showed that a covariance matrix with a combination of information about spatial layout, color and texture outperforms these methods both in constrained conditions and in a real-life dataset. We will therefore use covariance matrices in this paper as well.

Though big steps have been taken in both manual and automatic approaches, only by combining both techniques a satisfactory result can be obtained. We therefore let our person tracing system learn from the feedback the user provides. This is called relevance feedback.

This paper is organized as follows: We first formally introduce (interactive) tracking and tracing in section 2. Then in section 3 we discuss the methods used to detect and match tracks. In section 4 we focus on the visualization while section 5 describes the Relevance Feedback methods used. Both the matching methods and the Relevance Feedback methods are evaluated in section 6.

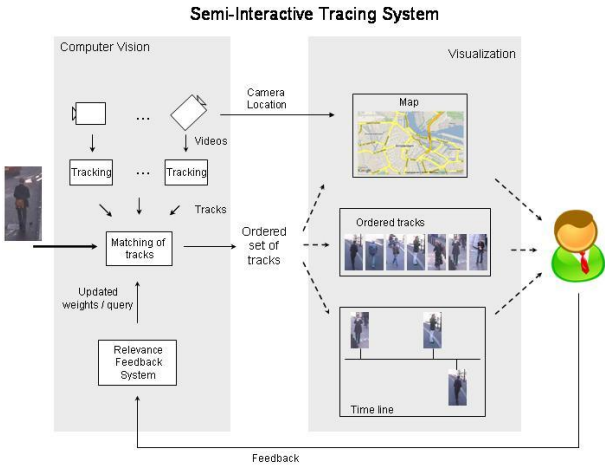


Figure 1: Overall semi-interactive tracing scheme. The user starts the process by selecting a single detection. The track this detection belongs to is matched to all other tracks from all cameras. The user can visualize these tracks using a time-line, map and / or ordered list of tracks. Based on this information the user provides feedback on all tracks shown. This feedback is used by the semi-interactive tracing system to improve tracing.

2. OVERVIEW OF THE SYSTEM

The overall scheme for tracing a person in real-life surveillance videos is shown in Figure 1. We first formally introduce all elements of the tracing scheme to obtain a consistent description throughout this paper.

The detection of a person in one frame f of the video v is represented by its region r. These regions form a track t when detections in subsequent frames are combined. The similarity between two tracks is given by $\Delta(\vec{f_1}, \vec{f_2})$ where $\vec{f_n}$ is the feature representation of track t_n. The resulting distances between a query track q and all other tracks are ordered to provide the user with the most probable matches first. For Relevance Feedback, the system returns the first n tracks to the user on which he or she provides feedback.

3. TRACKING AND MATCHING METHODS

Before tracing persons, we first need to detect and track persons within a single camera.

3.1 Tracking

For person detection we use the standard method introduced by Dalal and Triggs [4]. This method is based on the Histograms of Oriented Gradients which describes an image region using the dominant derivatives in the sub-regions of that region. For efficiency reasons it is only applied to certain Regions Of Interest. Since a person must have moved in order to enter the field of view of a camera, we focus on regions with movement. Such movement regions can be detected using background estimation [18] or motion field analysis [12]. In this paper we use the background estimation implementation provided by Zivkovic and Van der Heijden [22] for its ability to deal with complex and changing backgrounds.

Clearly, the true goal is not to find positions of persons in a single frame, but rather the complete tracks of those persons. The set of detections therefore needs to be combined to create tracks. Though many standard tracking methods exist [21], these methods assume a static frame-rate whereas real-life surveillance material often consists of time-lapse data. We therefore resort to a graph-based method based on hysteresis thresholding [13]. This method combines the best matching detections within a certain time frame and finds the best intermediate regions by lowering the demands on the detection threshold and subsequently uses A* to find the optimal tracks.

The result of the methods described in this section is a set of tracks, where each track consists of a set of subsequent detections.

3.2 Track Matching

To match tracks, each track is described using distinctive features and compared to other tracks using a distance measure. In constrained situations, calibrated cameras can be used to make sure that if the same person is seen in another camera with a similar direction towards the camera the feature description will indeed be similar. While in constrained situations pre-calibrated cameras are used [2] or the overlap in field of view between cameras is known [3] to calibrate the cameras afterwards, in real -life situations calibration cannot be used. The features used therefore need to be able to compensate for any differences in visual appearances between cameras. In previous work [17] we showed that for the matching of individual detections, an image region could best be described using a covariance matrix. This method combines descriptions of each individual pixel in an image region into one feature representation $\vec{f}$:

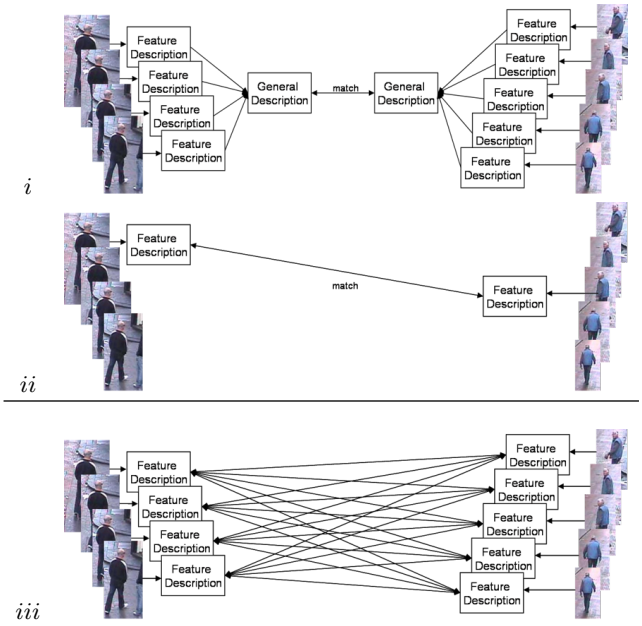


Figure 2: Comparison of two tracks using an average description (i), the largest detection (ii) or the minimal distance between all detections (iii).

$$f = \frac{1}{N-1}\sum_{n=1}^{N}(f_{pixel} - m)(f_{pixel} - m)^T$$

where N is the number of points in the region, m the mean vector of all feature vectors and f_{pixel} the feature vector used to describe a single pixel in the region. In [17] we also showed that the feature vector f_{pixel} could best be described using the location within the image region (spatial layout), the values of each color channel (color) and orientation and magnitude of the gradient (texture). The covariance matrix was originally proposed for tracing by Tuzel et al. [20]. The resulting covariance matrix is compared to other matrices using the distance measure proposed by Forstner and Moonen [7]:

$$\Delta(f_1, f_2) = \sqrt{\sum_i \ln \lambda_i(f_1, f_2)^2}$$

here, λ_i is the i^{th} generalized eigen vector of the covariance matrices f_1 and f_2.

To match complete tracks instead of single detections, a specific track comparison method is needed. Ideally this method is independent of the direction towards the camera, ignores false detections and does not care about the position of the person within a frame. A first approach to achieve this is to average the description of all detections in a track as this captures different views in one representation. Another way of comparing tracks is by focusing on the best detection. Since a person is best visible when he or she is closer to the camera, we represent the path using the feature description of the largest detection. Lastly, we compare tracks by letting all individual detections of the first track be matched to all individual detections of the second track. The smallest distance is then considered the true distance between the tracks. These matching methods are visualized in Figure 2.

Applying the image analysis methods discussed in this section to surveillance videos results in a set of tracks and a distance matrix between those tracks.

4. VISUALIZATION

While an automatic tracking and tracing system significantly eases the post-incident investigation, a good user interface is invaluable to answer the W4 questions. Figure 3 shows our interface which assists a user in answering these questions using a two screen setting.

This interface aids the user in answering the W4 questions as follows:

"Who?" Though complete identification of a person in low resolution images is infeasible, multiple occurrences of the same person are used to construct the complete trace of that person. After the reconstruction of an event, the user interface is capable of giving all visual information about any person in an accurate and concise manner.

"When" All tracks have inherent temporal information which can be used to alter the probability of two tracks showing the same person. To give a complete overview of the temporal correlation between tracks, the user interface incorporates a time-line.

"Where?" Similar to the temporal information, all tracks have a physical location where the track was recorded. This spatial information is shown using a map.

"What?" While in low resolution surveillance videos the current state-of-the-art is incapable of automatically identifying what action is performed, this information is invaluable for a complete overview of a video. Letting the user define what action is performed and show that information in a sensible manner is therefore an important part of the visualization.

As a concise track visualization we use a moving icon (Micon) since it is able to show all characteristics within the track without having to view the complete video. This method was first shown to handle video data in [16].

5. RELEVANCE FEEDBACK

So far, the interaction between a user and the tracing system we proposed in section 3.2 is restricted to navigating through the results. While this interaction might be sufficient in many situations and greatly speeds up the search process when compared to linear search, it could still be improved. An obvious improvement in this sense is to let the automatic tracing system learn from feedback the user provides, i.e. Relevance Feedback.

Two main directions of Relevance Feedback can be distinguished: optimizing the query or updating the weights of the feature dimensions. For query optimization we follow the well known Rocchio feedback approach [19]:

$$Q_m = \left(\alpha * \vec{Q_o}\right) + \left(\beta * \frac{\sum_{\vec{D_j} \in D_r} \vec{D_j}}{|D_r|}\right) - \left(\chi * \frac{\sum_{\vec{D_k} \in D_{nr}} \vec{D_k}}{|D_{nr}|}\right)$$

here, Q_m is the updated query and Q_o the original query. $\vec{D_j}$ is an element of the set of relevant tracks D_r and $\vec{D_k}$ an element from the set of non-relevant tracks D_{nr}. α, β and χ are weights which can be determined by the user beforehand.

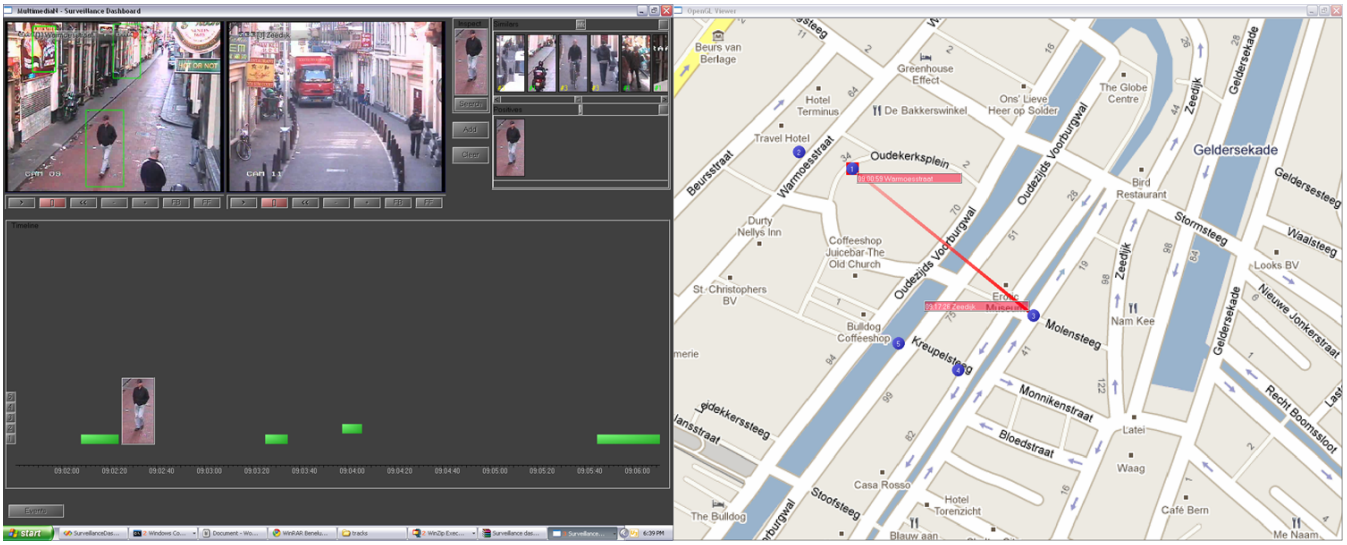


Figure 3: User interface of the proposed semi-interactive tracing system. The right screen is reserved for a map which provides the needed spatial information of any track. If more than one track is selected, the locations of those tracks are connected by a red line. Next to these locations the time of the detection is shown. This gives a direct overview of the path the person might have taken. Of the left screen, we use the bottom half for a time-line that shows all detected tracks and has the capability to zoom or pan at any moment. When zooming in, more information about the track is provided, giving overview by default and more detail on demand. The top half is dedicated to the visual characteristics of the person being traced, showing the original video containing the query track on the left. The right side shows an image of the person being traced together with a list of twenty tracks, ordered by visual similarity to the query track. The video of a potentially matching track is showed in the middle. Tracks can be selected using the original video or the time-line, its representation then becomes visually more apparent in all other components.

Feature re-weighting adds a weight to all feature dimensions, thereby gaining the ability to manage the influence of certain dimensions. The traditional method for re-weighting is to let the weights be inversely proportional to the variance over the feature dimension of all known relevant elements:

$$\vec{f_n} = [f_o^1, \ldots, f_o^k]^T [W^1, \ldots, W^k]$$

here, $\vec{f_n}$ is the new feature after weighing the original feature $\vec{f_o}$ with weights W. These weights are set using:

$$W^i = 1 - var(E^i_{relevant})$$

where $var(x^i)$ is the variance of dimension i over all elements in the set x and $E_{relevant}$ the set of tracks annotated by the user as being relevant.

For text retrieval and content based image retrieval, Relevance Feedback is often accompanied by active learning [10]. The idea behind this method is to let the system query those elements it beliefs it could learn the most from. Unfortunately this does not work for person tracing as the total number of tracks showing the same person is too small. Only querying the most difficult tracks will therefore lead to an unnecessary number of feedback loops.

The behavior and performance of the two relevance feedback paradigms presented in this section are thoroughly documented for the field of text retrieval and content based image retrieval. For tracing in real-life surveillance however, both the behavior and performance are unknown.

6. EXPERIMENTS

The different visualizations are difficult to evaluate in an objective manner without large scale user studies. In contrast, the automatic processing methods can be objectively analyzed. We conduct two experiments, comparing the matching techniques and Relevance Feedback methods respectively.

For both experiments we simulate user behavior and assume that if a user sees two matching tracks in the visualization he/she will identify them as such. Since we aim to create a complete reconstruction, our evaluation criterion is to minimize the interaction required to obtain a full reconstruction. We therefore use Recall in both experiments. In experiment one, for each matching technique the recall is set out against number of images seen. This evaluation method is known as a Cumulative Matching Curve [8]. To evaluate Relevance Feedback, the number of interaction steps is of bigger influence to the user than the number of images seen. For experiment two we therefore set Recall against the number of interaction steps.

6.1 Dataset

To test the different matching methods and relevance feedback methods we recorded a real-life dataset with the assistance of the Dutch police. This dataset consists of simultaneous recordings of five cameras without overlap in field-of-view, each lasting one hour. These recordings were made as part of the regular surveillance process for that area. A ground-truth is obtained by manually labeling the positions of nine persons who were asked to walk around in the area under surveillance. The dataset contains several sources of

Figure 4: Sample tracks after the detection and tracking methods of section 3 are applied to a real-life dataset of surveillance cameras in Amsterdam.

variation. Most notable are the changes in weather, camera angle, colors and texture of clothing and reflections in windows. Furthermore, the visual appearance of these nine persons varied greatly; some wearing distinctive colors where others were less characteristic.

Sample tracks of applying the detection and tracking methods described in section 3 are shown in Figure 4. The persons cooperating in the experiment were present in at least three and at most five cameras. Applying the detection and tracking methods described in section 3.1 results in a total of 2433 tracks. The nine participants are visible in sixty three of those tracks.

6.2 Results

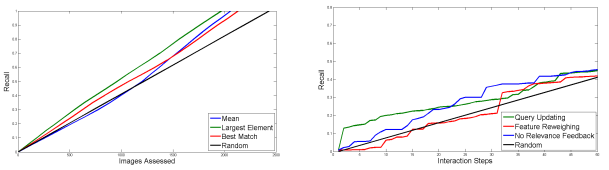
The average results of applying the three matching methods described in section 3.2 are given in Figure 5 (I). For the second experiment we simulated a human user as follows: For each element in the ground truth, twenty results are returned. The simulated user selects the relevant items in the list based on the predefined ground truth. Either query extension or feature weighing is then used to improve results. This iterative process is continued until all matching tracks are found. The results of both relevance feedback methods are shown in Figure 5 (II) together with the results of random ordering and not using any relevance feedback.

Obviously these results are greatly dependent on the visual characteristics of the person being traced. Figure 6 therefore shows the results of track matching for all persons individually. The tracks are matched using only the largest detection as this method performs best overall. In Figure 7 the person-dependent results for Relevance Feedback are given, using Query extension.

6.3 Discussion

In the first experiment we showed that all track matching methods had great difficulty matching the query track with its corresponding tracks. Using only the largest element as representative for a track gave the best results, but even this method outperforms random matching by only 20%. As expected the people with distinctive colors yield a much better result compared to the results of those that do not wear distinctive clothes.

When averaging the results over all persons of both relevance feedback methods, the results are worse when compared to not using relevance feedback. However, the findings from experiment one are contrary to experiment two. When using Relevance Feedback, a clear improvement in



I *II*

Figure 5: Influence of different matching methods (I) and Relevance Feedback methods (II) on track matching performance. In both figures the black line shows random performance. While the differences are small, largest element matching outperforms both minimal distance matching and mean matching. Query Updating is the best performing Relevance Feedback method.

performance can be observed for the group of persons wearing indistinctive clothing. These results indicate that a user of our semi-interactive tracing system should first identify a person as visually distinctive or indistinctive. Based on this classification either the standard matching method or Relevance Feedback is to be used.

7. CONCLUSION

In this paper we presented a system capable of tracing persons over multiple cameras. While all individual elements are well known, the combination of these methods is very powerful. We used state-of-the-art background extraction and detection methods to obtain initial detections. By combining these detections using an A* based optimization scheme we were able to obtain complete inner-camera tracks. These tracks were used to find multiple instances of the same person in different cameras. Different matching techniques were deployed which as expected showed that overall performance was best for persons traced with visual distinctive characteristics.

If the person traced did not have any obvious characteristics, tracing based solely on visual appearance was shown to be a tedious task. In these situations, either extra temporal or spatial information should be considered to further specify the search. Another approach is to improve the interaction between the user and the tracing system using relevance feedback. We showed that relevance feedback greatly improves tracing performance on persons without obvious visual characteristics.

8. ACKNOWLEDGMENTS

This work is partially sponsored by FES Camera 3D. The used real-life surveillance dataset was provided by the Netherlands Forensic Institute (NFI) in conjunction with the Dutch police, for which we thank Jurrien Bijhold.

9. REFERENCES

[1] A. Alahi, P. Vandergheynst, M. Bierlaire, and M. Kunt. Cascade of descriptors to detect and track objects across any network of cameras. In *CVIU*, 2010.

[2] J. Black and T. Ellis. Multi camera image tracking. In *IVC*, 2006.

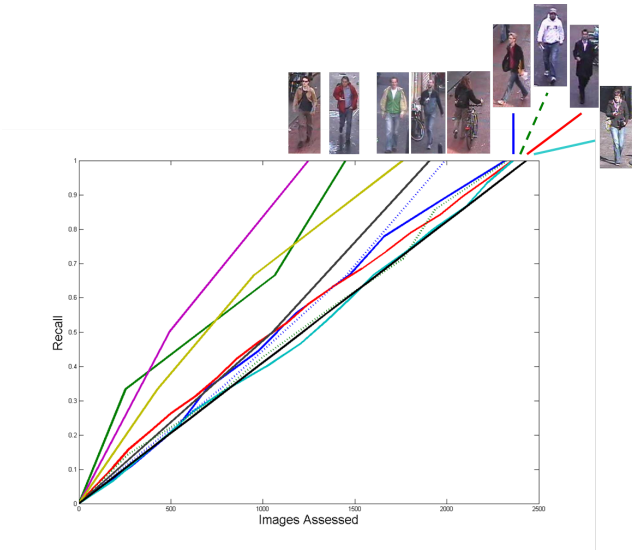


Figure 6: Results of person specific tracing, using largest detection as representation. For each person a representative image is added to show the influence of visual characteristics on tracing performance.

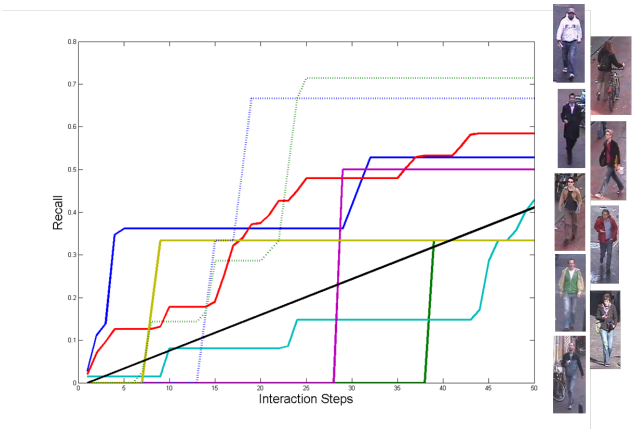


Figure 7: Person specific relevance feedback results, using Query updating. Note the large variation in results and large performance improvement on tracing of the most challenging persons.

[3] S. Calderara, A. Prati, and R. Cucchiara. Hecol: Homography and epipolar-based consistent labeling for outdoor park surveillance. In *CVIU*, 2008.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[5] P. Dóllar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, 2009.

[6] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *CVPR*, 2010.

[7] W. Forstner and B. Moonen. A metric for covariance matrices. In *Qua vadis geodesia*, 1999.

[8] D. Gray, S. Brennan, and H. Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *PETS*, 2007.

[9] D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV*, 2008.

[10] T. Huang, C. Dagli, S. Rajaram, E. Chang, M. Mandel, G. Poliner, and D. Ellis. Active learning for interactive multimedia retrieval. In *ICPR*, 2008.

[11] F. Janoos, S. Singh, O. Irfanoglu, R. Parent, and R. Machiraju. Activity analysis using spatio-temporal activity maps in surveillance applications. In *VAST*, 2007.

[12] A. Khashman. Modes: moving objects detection and extraction system. In *ACC*, 2008.

[13] P. Koppen and M. Worring. Multi-target tracking in time-lapse video forensics. In *MiFor*, 2009.

[14] G. Lavee, M. Rudzsky, E. Rivlin, and A. Borzin. Video event modeling and recognition in generalized stochastic petri nets. In *CirSysVideo*, 2010.

[15] Y. Livnat, J. Agutter, S. Moon, and S. Foresti. Visual correlation for situational awareness. In *INFOVIS*, 2005.

[16] W. E. Mackay and G. Davenport. Virtual video editing in interactive multimedia applications. In *Commun. ACM*, 1989.

[17] M. Metternich, M. Worring, and A. W. M. Smeulders. Color based tracing in real-life surveillance data. In *Springer Trans. on DHMS*, In Press, 2010.

[18] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam. Image change detection algorithms: A systematic survey. In *IEEE Trans. on Image Processing*, 2005.

[19] J. Rocchio. *Salton: the Smart Retrieval System: Experiments in Automatic Document Processing*, chapter Relevance Feedback in Information Retrieval, 1971.

[20] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. In *ECCV*, 2006.

[21] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. In *ACM Comput. Surv.*, 2006.

[22] Z. Zivkovic and F. van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. In *Pattern Recognition Letters*, 2006.