2012 IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance

Boosting Multi-Hypothesis Tracking by means of Instance-specific Models

Michael Pätzold and Rubén Heras Evangelio and Thomas Sikora
Communication System Group, Technische Universität Berlin
Einsteinufer 17, 10587 Berlin
paetzold, heras, sikora@nue.tu-berlin.de

Abstract

In this paper we present a visual person tracking-by-detection system based on on-line-learned instance-specific information along with the kinematic relation of measurements provided by a generic person-category detector. The proposed system is able to initialize tracks on individual persons and start learning their appearance even in crowded situations and does not require that a person enters the scene separately. For that purpose we integrate the process of learning instance-specific models into a standard MHT-framework. The capability of the system to eliminate detections-to-object association ambiguities occurring from missed detections or false ones is demonstrated by experiments for counting and tracking applications using very long video sequences on challenging outdoor scenarios.

1. Introduction

Tracking of multiple persons in a video sequence recorded by a static camera is a important low-level computer vision task and essential for automatic scene analysis. The trajectories gathered by this analysis can be utilized for economical purposes and security applications such as detecting abnormal behaviour, counting people entering public transports or assessing the influence of advertisement.

In general, the operation of multi-object-tracking-by-detection systems can be divided into two main parts: at first, a system searches in each new video frame possible object candidates by means of a beforehand learned generic model of the category of interest. Afterwards the measurements of the detector from multiple frames are combined to object trajectories by filtering spurious and missing measurements by a tracking module.

The development of generic models for the detection of the category "persons" is a challenging task, since different clothing and articulation of walking humans and additionally arbitrary lightning conditions lead to a variety of color and texture of their representations in an image. This problem of large intra class variation has been tackled in recent years and several reliable gradient based methods for the detection of separated persons in still images has been developed [6, 12, 16]. But when dealing with crowded scenarios, and therefore with a high degree of inter-object-occlusion, ambiguous gradient information may irritate detectors designed for the whole body. Wu and Nevatia tackled this problem by designing human part detectors based on gradient information and combining the detections in a joined likelihood model [17]. Pätzold *et al*. proposed to search only for the head-shoulder-region of a person and reject false positives by additional analysis of the motion of the region [13].

(a) (b)

Figure 1. **Proposed system at work: (a):** Circles represent the measurements of the generic person-category. By tracking multiple hypotheses of measurement-to-track associations new tracks are initialized. If a track is assessed as reliable instance-specific appearance models are built. **(b):** The appearance similarity for every learned model is evaluated (illustrated here in different colors) and the result is fed back to the MHT-framework aiming at improving the tracking quality.

Nevertheless, the detection rate, the false positive rate and the measurement accuracy of generic person-category detectors is not sufficient to infer the position of all existent objects in a single image directly. Therefore, applying a tracking filter to the noisy detector results is essential to infer reliably the number of objects and their positions.

978-0-7695-4797-8/12 $26.00 © 2012 IEEE
DOI 10.1109/AVSS.2012.18

In 1960 Kálmán proposed to use a recursive linear filter in order to obtain the statistically optimal state estimate for one object from a sequence of discrete, noisy data [10]. But when multiple objects move close to each other, there are more than one possible assignment of the set of available measurements to the set of objects. Assuming that at most one measurement can be assigned to one object, a solution for finding the global nearest neighbor (GNN) can be computed by linear programming. After completing the data association, all emerged tracks are updated by Kalman filtering with the associated measurement. Since the GNN-approach only takes the most likely association for the current frame into account, closely spaced targets and false measurements can lead to consecutive wrong associations and therefore, to track loss. This effect is alleviated by the Joint-Probabilistic-Data-Association-Filter (JPDAF). The JPDAF updates all tracks with all measurements of one time-step under consideration of the measurement origin uncertainty [1]. But JPDAF requires to know the number of objects and provides no solution to track initialization. The global optimal solution for multi object tracking should be found by propagating every possible data association hypothesis to the next time steps. Reid proposes an algorithm to build all possible hypotheses and provides a probabilistic formula in order to evaluate their probability [15]. The system proposed in this paper is based on a MHT-data-association system, because of its capability to detect and track objects even in crowded environments and under presence of clutter in time-critical applications by considering only information from previous frames.

Contrary to the before mentioned recursive approaches, there were published various methods which find an optimal solution for the data association problem by taking all measurements of all time-steps jointly into account [9, 18, 19]. Real-time performance of these methods can be accomplished by analyzing the scene using a sliding window technique [14].

In complex situations with partial or full object occlusion or highly maneuvering objects that do not obey the linear motion model the inherent state information (estimated position, velocity and their covariances) may not be sufficient to keep track. Wu and Nevatia propose to use the last available state information (color histogram, dynamic model and detector confidence) and initialize a mean-shift tracking during occlusion incidents [17]. More accurate instance-specific models can be build by integrating additional knowledge about the appearance of background or neighboring persons. By means of machine learning techniques it is possible to extract only the discriminative information [4, 11]. We also use these techniques to improve the tracking performance in challenging situations.

The main contributions of this paper is to describe how to integrate and manage person-specific information within the standard MHT-framework. By propagating multiple data association hypotheses MHT is able to initialize object trajectories even when people walk in crowded areas. These initial object trajectories are then used to provide reliable training data with correct labels to the instance-specific model learning algorithm. The benefit of the appearance information for tracking is twofold: we propose to incorporate the instance-specific information into the posterior probability computation of each hypothesis. Furthermore, we present a method to guide the tracker based on specific appearance information in the case of missed measurements from the generic person-category detector.

We structured the paper as follows: In the next section the standard MHT-theory and our tree-based implementation is introduced. In section three the integration of the instance-specific model into the MHT-framework is described. Section four shows experimental results and section five concludes the paper.

2. Standard MHT-Implementation

The proposed system is based on the standard Multi-Hypothesis-Tracking approach by Reid [15]. He proposes to propagate multiple hypotheses for the data association task from one time-step to another. For this purpose the MHT-algorithm creates a set of new hypotheses containing all possible combinations between each of the tracks and the set of measurements for each particular prior hypothesis. In order to evaluate the hypotheses Reid recursively defines a posterior probability of a hypothesis i at time k given a set of new measurements as

$$P_i^k = \frac{1}{c} P_D^{N_{DT}} (1-P_D)^{(N_{TGT}-N_{DT})} \beta_{FT}^{N_{FT}} \beta_{NT}^{N_{NT}} \times \left[\prod_{m=1}^{N_{DT}} \mathcal{N}(Z_m - H\hat{x}_j, P_j) \right] P_i^{k-1}, \quad (1)$$

where P_i^{k-1} is the prior hypothesis probability, P_D is the detection rate, β_{NT} and β_{FT} are the new target and false target density, N_{TGT},N_{DT}, N_{FT} and N_{NT} represent current hypothesis configuration parameters and c is a normalization constant. The likelihood to assign a measurement m to track j is modeled by the normal distribution $\mathcal{N}$ of its Kalman filter with its state $\hat{x}_j$ and covariance P_j. Thus, the measurements are assumed to be indistinguishable and the likelihood of assignment to a track only depends on their position. In order to prevent the set of hypotheses Y from growing exponentially over time, the unlikely ones are pruned at every time-step to a fixed maximal cardinality κ_{max}.

The storage of the previous tracking states of all current hypotheses is mandatory for global computation of the tracker performance, graphic data output and all extensions to the algorithm that require data from the past time-steps.

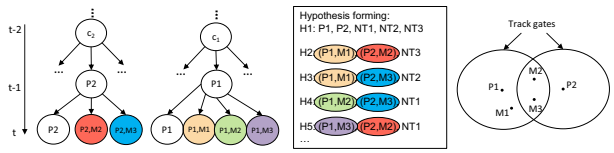

Figure 2. **Example for tree based track management:** Two tracks $P1$ and $P2$ with their track trees and the measurements $M1$, $M2$ and $M3$ are forming several new hypotheses at time-step t. The track tree nodes (here illustrated in different colors) store all information and the hypotheses are only pointing to the respective nodes.

In order to achieve an efficient storage of the state information, we implemented a tree-based track management as proposed in [3] and depicted in figure 2. Each track state for a time-step is stored in a particular node. The nodes of one track are linked over time resulting in an incrementation of the tree depth at every upcoming time-steps. Since at every time-step each single track spawns multiple new possible tracks (which are hold in the different hypothesis), the linked nodes represent a tree, the so called track tree and a track with its history is represented by a tree branch from a particular leaf node to the root node. All active tracks with a common starting measurement reference into the leafs of one track tree. By managing the data in this way the memory resources are used efficiently. Furthermore, by counting the references from the hypothesis tracks into the tree nodes it is possible to automatically delete nodes, on condition that there is no reference to this node anymore.

The generic person-category measurements used by our basic MHT-implementation are generated by a detector based on histograms of oriented gradients (HoG) [6]. We trained only the upper body region of a human, since that is usually the only body part a camera with common tilt angle is able to observe. In the next sections we give a description how instance-specific information is integrated into this MHT-framework and how it improves the tracking performance.

3. Integration of the Instance-Specific Model

3.1. Instance-specific Model Initialization

Our system initializes new tracks based on the described standard MHT-framework using measurements of the generic person-category detector. In order to benefit from instance-specific information we need to pick at every time-step suitable tracks from the set of hypotheses for which we learn an appearance model. It is obvious that learning a model for every track in each hypothesis is computational intractable and also an unprofitable process, because there exist plenty of duplicate or at least similar tracks in the set of hypotheses and we ideally should only learn a model per person. Therefore, we propose a method to identify tracks to learn based on the previously described tree-based track management. First, we merge duplicate tracks in the track-trees. Afterwards the number of nodes to train a model for is further reduced by collecting a all nodes which exist a defined number of time-steps back in history. Finally, we apply a filter to obtain a small set of tree-nodes, the so called *model nodes*, for which a specific model is trained.

We reduce the number of nodes pointing to an individual person by merging tracks referencing to different track tree nodes, but representing the same person. For this purpose we compare the states of two tracks j and k and merge them if the Mahalanobis distance

$$(\hat{x}_k - \hat{x}_j)'[P_j]^{-1}(\hat{x}_k - \hat{x}_j) \leq \sigma, \tag{2}$$

is below a threshold σ. This value has to be set, maintaining the distinctiveness of different hypotheses, while different track trees pointing to the same person are merged.

All current track information is obtained by collecting the leaf nodes of all track trees. This set of nodes is reduced by ascending the tree of each node by n_{LB} nodes (time-steps of looking back). This way many nodes are replaced by a common parent node, which had only minor modifications in the last n_{LB} time-steps. We can count for each of these parent nodes the number of tracks ϵ that share this node. This number represents directly the number of hypotheses that use this node, since every track that shares this node must be in different hypotheses according to the assumption that an object originates at most one measurement per time step. We only take these nodes as model nodes that are conform with the following constraint

$$\epsilon > \frac{\kappa_{max}}{2}. \tag{3}$$

This method ensures that recently created targets has to be established for several time steps before a model is built. And by only considering information from established tracks the labeling of this information is reliable, which is a major requirement for the training of instance-specific models using supervised learning methods.

3.2. Boosting the Instance-specific Model

The instance-specific model should not only model a specific person but also discriminate the person from surrounding persons and background as much as possible, since it is used for associating the detections to the tracks later on. Therefore, we collect the training data and label them similar to [4, 11]. We use the last measurements of a track as positive labeled input data. They are already stored in the respective track tree nodes and can be collected by ascending beginning at the model node to their parent nodes recursively. Furthermore, we take other neighboring tracks

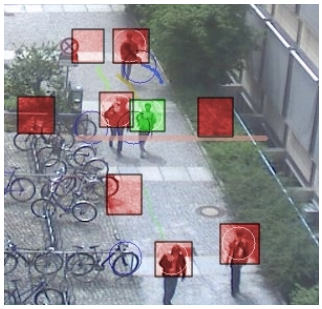

Figure 3. **Gathering of labeled data for training the instance-specific model:** Detector measurements of the particular person serve as positive labeled data samples (displayed as green patch). Background patches and patches of surrounding persons are collected and labeled negative (depicted as red patches).

and background patches of the current time-step as negative labeled input data as depicted in figure 3. For every data location a set of haar-like features and a normalized RGB-histogram is computed. We apply Adaptive Boosting [8] to the gathered labeled training data per model node. We use a set of decision stumps as feature pool $\mathbf{h}$ by testing each bin value of the histogram and the haar-like feature values with specific thresholds. After learning the instance-specific model c is a linear combination of decision stumps which can be evaluated at position x as follows

$$c(x) = \sum_{t=1}^{T} \alpha_t h_t(x), h_t \in \mathbf{h}. \tag{4}$$

Finally, the created model is stored in the model node, so that it is accessible by all child nodes. In figure 2 the instance-specific models c_1 and c_2 for $n_{LB} = 2$ are depicted. These models are used by all corresponding child nodes.

3.3. Data Association aided by Instance-Specific-Model

In Reid's standard MHT-approach the likelihood of the assignment of a measurement to a track depends only on the kinematics of the detector measurements, as seen in eq. 1. But the availability of instance-specific models enables to augment the likelihood with a term for object appearance similarity. The instance-specific model c_j for track j which can be found by ascending its track tree is evaluated at each measurement z_m and the posterior probability of a hypothesis is now computed as

$$\begin{aligned} P_i^k &= \frac{1}{c} P_D^{N_{DT}} (1-P_D)^{(N_{TGT}-N_{DT})} \beta_{FT}^{N_{FT}} \beta_{NT}^{N_{NT}} \\ &\times \left[\prod_{m=1}^{N_{DT}} \rho \mathcal{N}(z_m - H\hat{x_j}, P_j) + (1-\rho) \frac{c_j(z_m)}{\tau_{max}} \right] \\ &\times P_i^{k-1}, \end{aligned} \tag{5}$$

where τ_{max} is a normalization constant and ρ enables to control the influence of the kinematics and the instance-specific model.

3.4. Unassigned Track Guiding by PDA-Filtering

Due to object occlusions or background clutter the detector may miss some detections of a track. We propose to guide the tracker in these cases by using the instance-specific model knowledge as additional sensor input.

In order to use the instance-specific information the models need to be evaluated at each new frame. For that purpose we take all tracks of the most probable hypothesis and search for model nodes by ascending their track trees. Since one particular model c_j provides information to multiple tracks, an appropriate evaluation area has to be defined, so that the costly evaluation is only performed once. Therefore, we traverse the tree beginning from the model node to all leaf nodes and span a minimal bounding box Ω which includes all leaf positions. Afterwards, a probability map P_{cj} is build by evaluating the model at every pixel x within this area

$$P_{cj} = c_j(x) : \forall x \in \Omega. \tag{6}$$

The probability maps for multiple persons are depicted color-coded in figure 1(b). Potential object positions $Z^j = \{z_1^j, z_2^j, \cdots, z_k^j\}$ for a model c_j are computed by seeking the modes of the map P_{cj} by applying Non-Maxima-Suppression and Mean-shift [5].

The set Z^j is considered as measurements for each unassigned Kalman filter which uses the model c_j. We propose to use PDAF [1] to cope with measurement origin uncertainty and update the Kalman filter accordingly. We compute the association probability β_i^j for the potential position z_i^j of instance-specific model c_j as described in [1], where the likelihood ratio $\mathcal{L}_i^j$ that z_i^j arises from the object rather than from clutter is computed by

$$\mathcal{L}_i = \frac{\mathcal{N}(z_i^j - H\hat{x}, P(k|k-1)) P_D \frac{c_j(z_i^j)}{\tau_{max}}}{\lambda} \tag{7}$$

considering the detection rate P_D, the Poison clutter model density λ and also the normalized confidence $\frac{c_j(z_i^j)}{\tau_{max}}$ of the instance-specific model. The state estimate of the Kalman filter is then updated as

$$\hat{x}(k|k) = \hat{x}(k|k-1) + W(k) \sum_{i=1}^{m} \beta_i^j(k)(z_i^j - H\hat{x}(k|k-1)), \tag{8}$$

where $W(k)$ denotes the Kalman gain. The updated covariance is computed as

$$P(k|k) = \beta_0^j P(k|k-1) + (1-\beta_0^j) P^c(k|k) + \tilde{P}(k), \tag{9}$$

where $P^c(k|k)$ is the standard Kalman error covariance and $\tilde{P}(k)$ reflects the measurement origin uncertainty and is computed according to [1].

Finally, this additional information is also integrated into the posterior probability of hypothesis within the MHT-framework similar to a Sensor Type 2 as explained in detail in [15].

4. Experimental Results

In this section we present the results of the evaluation of our system applied to long-term sequences. We implemented the whole system including HoG-detector (the same as in [13]) and MHT-framework with instance-specific model generator in C++. It processes one video frame on a Intel Duo Core CPU (E8400) in 0.3 to 4 frames per second depending on the number of people in the scene. The parameters σ, τ_{max} and ρ of our system are set experimentally, but fixed for all sequences.

We use two video sequences recorded at our campus by a camera which was observing a courtyard of circa 5 meters width and 30 meters length which is highly frequented (see Figure 1(a)). In the first sequence 'Sparse Crowd' separated persons as well as small groups of maximal five people are passing the courtyard in changing lightning conditions. The second sequence 'Dense Crowd' shows a higher crowd density with up to 18 persons passing through the scene together. Additionally, we evaluated the performance of the system with the sequence 'S1 L1 13-57' of the public available PETS dataset [7].

One of our main applications of the system is the estimation of the number of people passing the scene. For that purpose we defined a fictive line and annotated each person crossing it and compared this data to the output of the proposed method and the standard MHT. The proposed method is able to count reliably the number of persons in a video stream as illustrated in figure 4. The tracking fails in few cases caused by permanent absence of detections. The proposed method clearly outperforms the standard MHT, which uses the same measurements provided by the person detector. This can be explained by the capability of our system to keep track of a person even if it is not detected for a longer interval. Also at the PETS-sequence, the proposed system outperforms the standard MHT-system. But, due to the lack of a sophisticated motion analysis (as applied in the approach of [13]) it misses some heavily occluded persons.

In order to evaluate the tracking performance in more detail we annotated each person trajectory of the most challenging part of the 'dense crowd'-sequence containing high crowd density and computed the CLEAR MOT metrics as presented in [2]. The Multiple Object Tracking Accuracy (MOTA) provides a measure for the object configuration errors made by the tracker (false positives, misses,

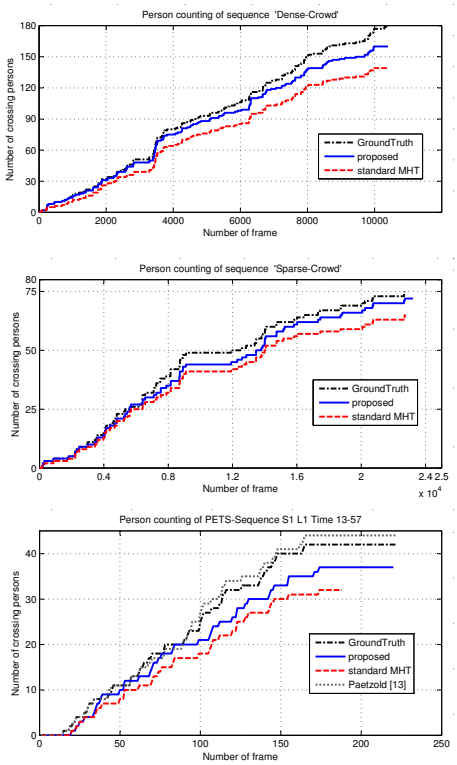

Figure 4. Number of persons counted by the proposed system on our dataset and sequence 'S1_L1-Time_13-57' from the public available PETS dataset.

mismatches). The precision of the tracker position estimates is measured by the Multiple Object Tracking Precision (MOTP), which is defined as the average total position error between annotated and tracked positions that are considered as configuration matches during the evaluation process. In order to illustrate the characteristics of the evaluated methods in detail both measures are computed for batches of 50 frames.

The first plot of figure 5 shows that after the entering of a large group of people at frame 3300 the MOTA of the standard MHT method decreases, while the proposed system's MOTA is still high. The standard MHT loses track of persons with very infrequent detector responses, while the proposed method is able to continue the tracking of these persons, as it incorporates the specific appearance information. But the more inaccurate position estimation of these objects causes a higher MOTP value for this situation compared to the standard MHT, as illustrated in the second plot of figure 5. Furthermore, a higher MOTA for the proposed method is observable for the entire sequence. This can be explained by a better handling of re-identification situations caused by occlusions and the lower probability of assigning false positive detections to existing tracks.

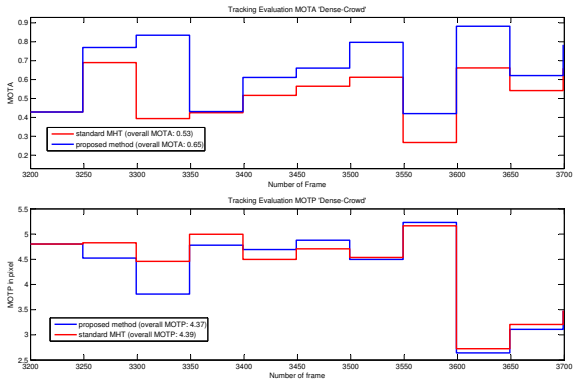

Figure 5. Evaluation of tracking performance for sequence 'Dense-Crowd' using the MOTA and MOTP measures.

5. Conclusions

We presented a system for tracking multiple person in an environment with high crowd density. Due to the use of a multi hypothesis tracker the system does not require separated persons in order to initialize new tracks. We presented a method to assess the confidence of the existence of a track and proposed to train instance-specific models at the moment when tracks become reliable. The trained models are evaluated efficiently and the data obtained thereby is integrated into the MHT-framework and thus, supports the tracking task. We showed in experiments that our system performs significantly better than the standard MTH-approach.

6. Acknowledgements

The research leading to these results has received funding from the European Community's FP7 under grant agreement number 261743 (NoE VideoSense).

References

[1] Y. Bar-Shalom, F. Daum, and J. Huang. The probabilistic data association filter. *Control Systems Magazine, IEEE*, 29(6):82 –100, 2009.

[2] K. Bernardin and R. Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *J. Image Video Process.*, 2008:1:1–1:10, Jan. 2008.

[3] S. Blackman and R. Popoli. *Design and analysis of modern tracking systems*. Artech House radar library. Artech House, 1999.

[4] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Robust tracking-by-detection using a detector confidence particle filter. In *Proc. IEEE 12th Int Computer Vision Conf*, pages 1515–1522, 2009.

[5] D. Comaniciu and P. Meer. Distribution free decomposition of multivariate data. *Pattern Analysis and Applications*, 2:22–30, 1998.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. International Conference on Computer Vision & Pattern Recognition (CVPR2005)*, volume 2, pages 886–893, INRIA Rhône-Alpes, ZIRST-655, av. de l'Europe, Montbonnot-38334, June 2005.

[7] J. Ferryman and A. Ellis. Pets2010: Dataset and challenge. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 143 –150, 292010-sept.1 2010.

[8] Y. Freund and R. Schapire. A short introduction to boosting. *J. Japan. Soc. for Artif. Intel.*, 14(5):771–780, 1999.

[9] C. Huang, B. Wu, and R. Nevatia. Robust object tracking by hierarchical association of detection responses. In *Proceedings of the 10th European Conference on Computer Vision: Part II*, ECCV '08, pages 788–801, Berlin, Heidelberg, 2008. Springer-Verlag.

[10] R. E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME–Journal of Basic Engineering*, 82(Series D):35–45, 1960.

[11] C.-H. Kuo, C. Huang, and R. Nevatia. Multi-target tracking by on-line learned discriminative appearance models. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 685 –692, june 2010.

[12] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *International Journal of Computer Vision*, 77(1-3):259–289, May 2008.

[13] M. Pätzold, R. H. Evangelio, and T. Sikora. Counting people in crowded environments by fusion of shape and motion information. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 157 –164, 29 2010-sept. 1 2010.

[14] M. Pätzold and T. Sikora. Real-time person counting by propagating networks flows. In *Advanced Video and Signal-Based Surveillance (AVSS), 2011 8th IEEE International Conference on*, pages 66 –70, 30 2011-sept. 2 2011.

[15] D. Reid. An algorithm for tracking multiple targets. *IEEE Trans. Automatic Control*, 24(6):843–854, 1979.

[16] E. Seemann, M. Fritz, and B. Schiele. Towards robust pedestrian detection in crowded image sequences. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1 –8, june 2007.

[17] B. Wu and R. Nevatia. Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors. *Int. J. Comput. Vision*, 75(2):247–266, 2007.

[18] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR 2008)*, pages 1–8, 2008.

[19] T. Zhao and R. Nevatia. Tracking multiple humans in crowded environment. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, pages 406–413, 2004.