The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XL-3, 2014
ISPRS Technical Commission III Symposium, 5 – 7 September 2014, Zurich, Switzerland

A DYNAMIC BAYES NETWORK FOR VISUAL PEDESTRIAN TRACKING

T. Klinger*, F. Rottensteiner, C. Heipke

Institute of Photogrammetry and GeoInformation, Leibniz Universität Hannover, Germany
(klinger, rottensteiner, heipke)@ipi.uni-hannover.de

Commission III, WG III/3

KEY WORDS: Classification, Reasoning, Tracking, On-line, Video

ABSTRACT:

Many tracking systems rely on independent single frame detections that are handled as observations in a recursive estimation framework. If these observations are imprecise the generated trajectory is prone to be updated towards a wrong position. In contrary to existing methods our novel approach suggests a Dynamic Bayes Network in which the state vector of a recursive Bayes filter, as well as the location of the tracked object in the image are modelled as unknowns. These unknowns are estimated in a probabilistic framework taking into account a dynamic model, prior scene information, and a state-of-the-art pedestrian detector and classifier. The classifier is based on the Random Forests-algorithm and is capable of being trained incrementally so that new training samples can be incorporated at runtime. This allows the classifier to adapt to the changing appearance of a target and to unlearn outdated features. The approach is evaluated on a publicly available dataset captured in a challenging outdoor scenario. Using the adaptive classifier, our system is able to keep track of pedestrians over long distances while at the same time supporting the localisation of the people. The results show that the derived trajectories achieve a geometric accuracy superior to the one achieved by modelling the image positions as observations.

1. INTRODUCTION

Pedestrian tracking is one of the most active research topics in image sequence analysis and computer vision. The aim of tracking is to establish correspondences between target locations over time and is hence useful for a semantic interpretation of a scene. Following Smeulders et al. (2013), visual object tracking can be categorised according to the way in which the pedestrian position in image space is acquired. *Matching-based* approaches used for instance in (Comaniciu et al., 2003), update the trajectory at every epoch. As a consequence in cases matching fails or returns ambiguous results the trajectory is easily attracted to other objects than the target. *Detection-based* approaches typically use classifiers to discriminate the regarded object class(es). Available approaches differ in the number of classes (binary versus multi-class) and in the way the training is conducted (on-line vs. off-line). Binary off-line trained classifiers differentiating one class from the background such as the HOG/SVM (Dalal and Triggs, 2005) and AdaBoost based approaches (Viola and Jones, 2001) can be trained with a large set of training data and hence perform well in many different scenarios. The outcomes of such systems are applicable to multi-object tracking, if the data association problem is solved, either explicitly, as in (Schindler et al., 2010), or implicitly, as in (Milan et al., 2014). While these classifiers work well for the underlying object class, they are prone to fail when the appearance of the individual pedestrians undergoes object- or scene specific changes. These changes can be taken into account by classifiers trained on-line, which can learn and update statistics about an object appearance, e.g. (Saffari et al., 2009), (Kalal et al., 2010). The adaptation of the appearance changes makes these approaches applicable to complex scenes with a wide range of depth, temporary occlusions, and changing lighting conditions. Ommer et al. (2009) discern different moving object classes present in typical outdoor scenarios using a multi-class SVM which is trained off-line. Distinguishing between various classes is expected to increase the per-class-accuracy because individual classes can be better separated from other similar object classes. To this end, Breitenstein et al. (2011) suggest an on-line adaptive multi-object tracking approach using a single boosted particle-filter for each tracked individual. In (Klinger and Muhle, 2012) an on-line approach based on on-line Random Forests (Saffari et al., 2009), in which each class represents one pedestrian, is suggested for multi-object tracking.

The term *detection* involves finding evidence for the presence of a pedestrian and a (at least coarse) localisation. Though there exists a lot of work related to the detection and tracking of pedestrians, only few papers address its geometric accuracy, e.g. (Dai and Hoiem, 2012). The position of a detected person is usually defined to be the location of some window around that person, which does not necessarily align well to the actual position of the person itself and thus only yields an approximate position. If image acquisition by multiple cameras is possible, a stereoscopic approach can be used to estimate the 3D position and size of pedestrians, which in turn supports the localisation in the image, see for instance (Eshel and Moses, 2010) and (Menze et al., 2013). For many realistic applications like motion analysis and interaction of people in sports, video surveillance and driver assistance systems, where one has to decide whether a pedestrian does actually enter a vehicle path or not, geometric accuracy is crucial. Most tracking approaches use variants of the recursive Bayes filter in order to find a compromise between image based measurements (i.e. automatic pedestrian detections) and a motion model, where the motion model implies the expected temporal dynamics of the objects, e.g. constant velocity and smooth motion. In such filter models, the state variables are modelled as unknowns and the image based measurements as observations. Approaches where the filter state is represented in factorised form are referred to as Dynamic Bayes Networks, see for instance (Dean and Kanazawa, 1989) and (Montemerlo et al., 2002).

In this paper we propose and investigate a Dynamic Bayes Network for pedestrian tracking which combines the results of detection, recursive filtering, prior scene knowledge, and a classifier with on-line training capability in a single probabilistic tracking-

*Corresponding author.

This contribution has been peer-reviewed.
doi:10.5194/isprsarchives-XL-3-145-2014

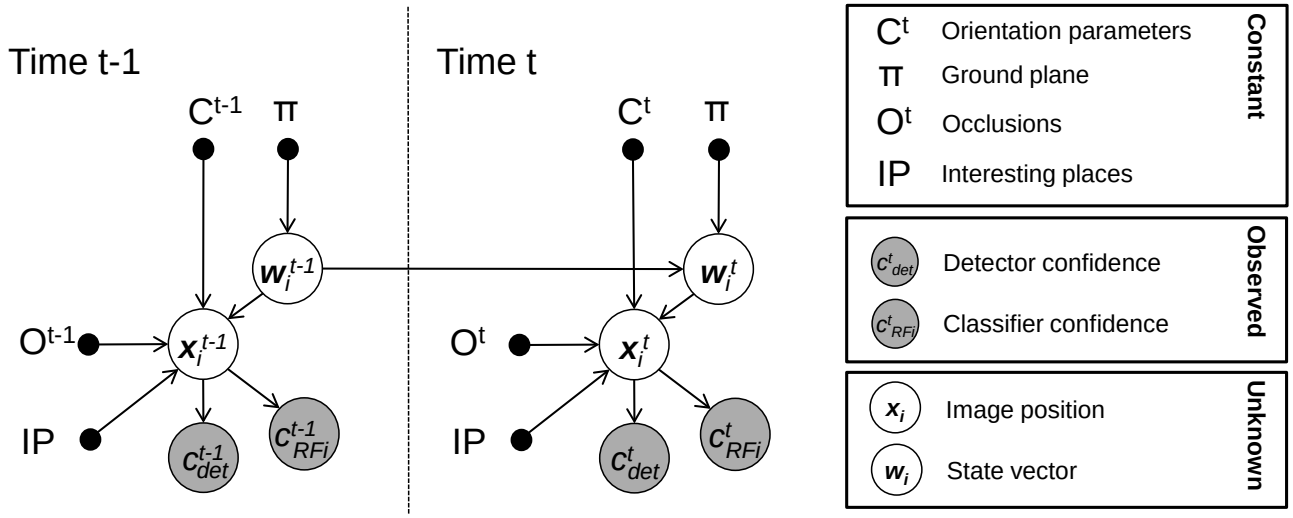


Figure 1: Dynamic Bayes Network proposed for pedestrian tracking. The nodes represent random variables, the edges conditional dependencies between them. The meaning of the variables is briefly explained on the right and in detail in the text.

by-detection framework using mono-view image sequences. By modelling the results of the pedestrian detection, i.e., the position of a person visible in the image, as a hidden variable, the system allows the detection to be corrected before it is incorporated into the recursive filter. In this way, the proposed method carries out the update step of the recursive filter with an improved detection result, leading to a more precise prediction of the state for the next iteration. In turn, the precise prediction supports the determination of the new image position which is important for both, the filter and the on-line classifier.

2. PROPOSED METHOD

In a standard Kalman Filter, the system state of a sequential process is supposed to be unknown and is directly combined with observations (in our context: the pedestrian position in image space). In the proposed method the position in the image is modelled as a hidden variable instead, which is connected to the detection and classification algorithms (see below). The basic building block of the proposed system is thus a Dynamic Bayes Network (referred to as DBN). Following the standard notation for graphical models (Bishop, 2006), the network structure of the proposed DBN is depicted in Figure 1. The small solid circles represent deterministic parameters and the larger circles random variables, where the grey nodes correspond to observed and the blank nodes to unknown parameters. One such graphical model is constructed for each tracked pedestrian. As indicated by the subscript i, the system state $\mathbf{w}_i^t$, the image position $\mathbf{x}_i^t$ and the results c_{RFi}^t of a classifier are modelled for each person individually, while all other variables are either valid for an entire image frame (if denoted by a superscript t indicating the time step), or for the entire sequence. The joint probability density function (pdf) of the involved variables can be factorised in accordance with the network structure:

$$\begin{aligned} &P(\mathbf{x}_i^t, \mathbf{w}_i^t, \mathbf{w}_i^{t-1}, c_{det}^t, c_{RFi}^t, C^t, O^t, IP, \pi) \\ &\propto P(\mathbf{x}_i^t|\mathbf{w}_i^t, C^t) P(\mathbf{w}_i^t|\mathbf{w}_i^{t-1}, \pi) \\ &P(c_{RFi}^t|\mathbf{x}_i^t) P(c_{det}^t|\mathbf{x}_i^t) P(\mathbf{x}_i^t|O^t) P(\mathbf{x}_i^t|IP). \end{aligned} \quad (1)$$

In the following the different variables considered in the approach are explained in detail. For the ease of readability the superscript t is omitted in the remainder of the paper where it is obvious.

2.1 Probabilities related to the image position

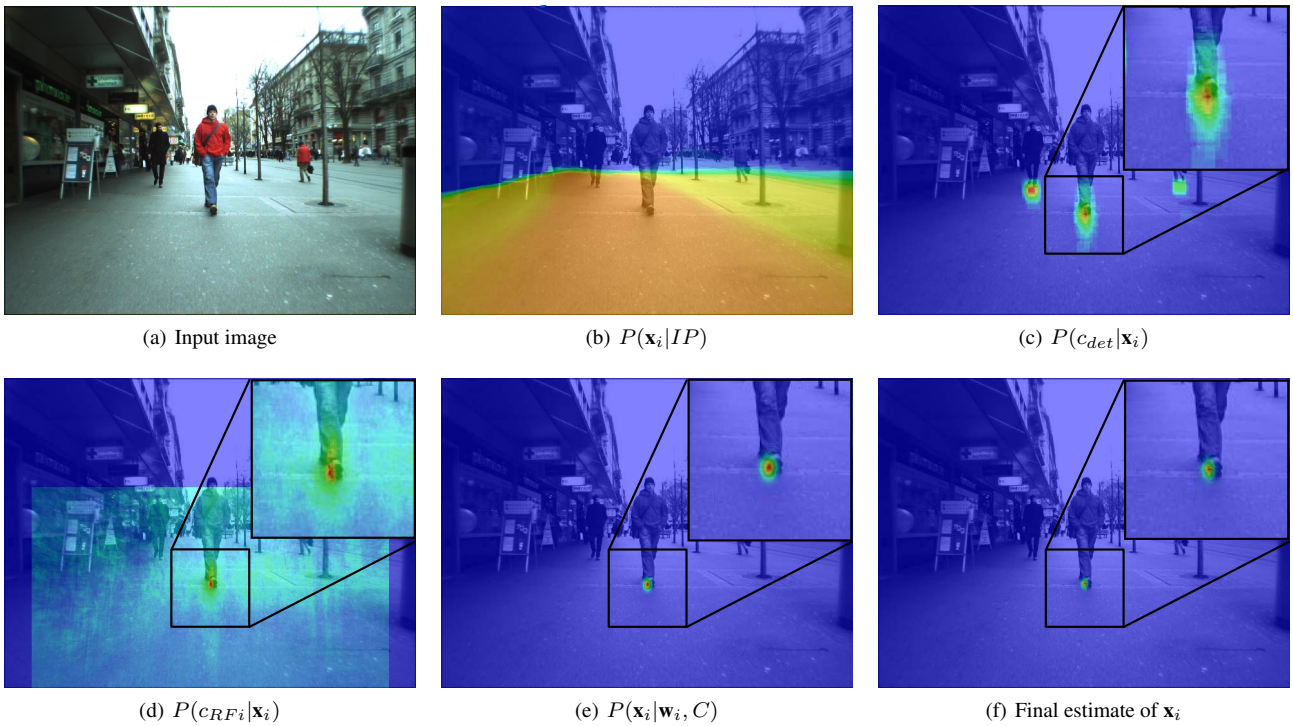
The image position $\mathbf{x}_i = [x_i, y_i]^T$ represents the position of person i in the image, where x_i and y_i are the column and row coordinates of the bottom centre point of the minimal spanning rectangle around the person which is related to the position of the feet (this point is referred to as reference point of the person in the following). In our model the variable $\mathbf{x}_i$ cannot be observed directly, so we model it as unknown and determine its optimal value by applying maximum-a-posteriori voting given the observed and the fixed entities of the system. The position in the image depends on the interior and exterior orientation parameters C of the camera (which we consider as given at each time step), on a binary variable O indicating if the object is occluded, on prior information IP about the scene, and on the position and velocity $\mathbf{w}_i$ of the pedestrian in world coordinates. Furthermore, the image position relates to the confidence of an on-line Random Forest classifier (c_{RFi}) and a pedestrian detector (c_{det}).

Model relating the state vector to the image position: $P(\mathbf{x}_i|\mathbf{w}_i, C)$ relates the (predicted) state $\mathbf{w}_i$ at time t to the corresponding image position, given also the orientation parameters C of the camera. The model is formulated as a Gaussian distribution $P(\mathbf{x}_i|\mathbf{w}_i, C) = N(f(\mathbf{w}_i), \Sigma_m)$ with a non-linear function $f(\mathbf{w}_i)$ of the state as mean and a covariance matrix Σ_m accounting for the uncertainty in the determination of $\mathbf{x}_i$. $f(\mathbf{w}_i)$ is related to the collinearity equations and computes the image position from the given point on the ground plane and the orientation parameters of the camera. We set the elements of Σ_m according to an assumed localisation uncertainty of $0.3m$ in world coordinates propagated to the image, which will be adapted to the actual uncertainty in the determination of $\mathbf{x}_i^t$ in future work.

Occlusion model: In order to model mutual occlusions between pedestrians in the scene we define a binary indicator O which describes whether a person is expected to be occluded or not, depending on its position in the image. Therefore we set

$$P(\mathbf{x}_i|O) = \begin{cases} 1, & \text{if } O(\mathbf{x}_i) = 0 \\ 0, & \text{if } O(\mathbf{x}_i) = 1 \end{cases}. \quad (2)$$

$O(\mathbf{x}_i)$ can be estimated for each position in the image by projecting $\mathbf{x}_i$ to the ground plane π (see below for a definition) and investigating the depth ordering of predicted pedestrian positions



(a) Input image (b) $P(\mathbf{x}_i|IP)$ (c) $P(c_{det}|\mathbf{x}_i)$

(d) $P(c_{RFi}|\mathbf{x}_i)$ (e) $P(\mathbf{x}_i|\mathbf{w}_i, C)$ (f) Final estimate of $\mathbf{x}_i$

Figure 2: Frame #2 of the test sequence and visualisations of the probabilities associated with the position of the nearest pedestrian to the camera in the image. Blue pixels indicate low, red pixels high probabilities.

relative to the camera position. We do not model the conditional dependencies between the state, the camera orientation and the occlusion explicitly, since we do not strive to optimise the occlusion and only take it as an indication to omit trajectory updates when an object is obviously situated behind others. Hence, the occlusion variable is modelled as given variable in our model. This model is a simplification that disregards the actual dependencies between the variable O and $\mathbf{w}_i$. We will involve a more sophisticated occlusion model in future work.

Interesting places: $P(\mathbf{x}_i|IP)$ is designed to emphasise regions in the image where pedestrians occur with higher frequency, thus the variable is called *Interesting Places*. We train a binary Random Forest classifier with x_i and y_i as features and class assignments according to true and false positive detections obtained by a HOG/SVM detector (Dalal and Triggs, 2005) in a training phase. The training samples are split into positive and negative samples by validation with reference data, using an intersection-over-union score threshold of 50%. In Figure 2(b) $P(\mathbf{x}_i|IP)$ generated by the Random Forest classifier is visualised for every possible position in the image shown in Figure 2(a). As can be seen, locations on the ground plane are favoured by the classifier and among those the locations on the side walk are assigned a higher value than those along the tram (on the right-hand side of the image). Since the tilt angle of the camera remains approximately constant throughout the sequence we use for our experiments (see Section 3) and the function is relatively smooth in the lower part of the image, $P(\mathbf{x}_i|IP)$ can be transferred well from the training to the actual experiment. The image in Figure 2(b) is hence assumed to be valid for the entire test sequence.

Detector confidence: $P(c_{det}|\mathbf{x}_i)$ is the probability density for *any* person to be present if $\mathbf{x}_i$ is its position in the image. We set the detector confidence proportional to the number of hits a HOG/SVM without internal threshold achieved in scale space, where we increment the number at each pixel within a square of 21 pixels side length centred on the reference point of each detected person. The value of 21 pixels accounts for the geometric uncertainty of the detector and is chosen heuristically. The detector confidence $P(c_{det}|\mathbf{x}_i)$ computed for the image shown in Figure 2(a) is depicted in Figure 2(c). Regions that are assigned as pedestrians by the HOG/SVM multiple times are favoured over those with fewer assignments. The corresponding pdf is designed to highlight the positions of any pedestrian visible in an image and is computed for each frame.

Classifier confidence: By $P(c_{RFi}|\mathbf{x}_i)$ we denote the pdf that $\mathbf{x}_i$ is the position of the ith person in the image. For that purpose an on-line Random Forest (Saffari et al., 2009) is trained, which considers one class for each person and an additional class for the background. To guarantee that the number of training samples is equal for every class, the classifier is trained anew with samples stored in a queue every time a new trajectory is initialised or terminated. Every time a trajectory is updated, we take positive training samples from an elliptic region with the new target position as reference point and a width-to-height ratio of 0.5. The height of the ellipse corresponds to the height of the pedestrian in $[m]$ (estimated given the bounding box height of the initial detection), transformed to a height in $pixels$ by a scale factor depending on the focal length and the distance of the predicted state to the camera. The width and height of the ellipse are stored for the evaluation and visualisation of the results (Section 3). Because the samples are rare from scratch, further positive training samples are taken from positions shifted by one pixel up, down, left and right. Negative samples (for the background class) are taken from positions translated by half of the size of the ellipse in the same directions. The feature vector is composed of the RGB values inside the ellipse.

Classification then delivers $P(c_{RFi}|\mathbf{x}_i) \propto \frac{n_i}{n_0}$, where n_i and n_0 are the relative frequencies of class i and the background class 0, respectively, assigned to the leaf nodes of all decision trees in the

Random Forest to which the sample $\mathbf{x}_i$ propagates. $P(c_{RFi}|\mathbf{x}_i)$ is evaluated for every $\mathbf{x}_i$ located within a square of 21 pixels side length around the predicted state of the ith trajectory. In Figure 2(d) an exemplary classifier confidence distribution is depicted. $P(c_{RFi}|\mathbf{x}_i)$ is shown for every potential position in the image (though we only use the smaller region of 21 by 21 pixels for computation). Note that at the positions, where persons different to the one closest to the camera (cf. Figure 2(a) and 2(c)) were found, $P(c_{RFi}|\mathbf{x}_i)$ is rather low.

2.2 Probabilities related to the state vector

The state vector $\mathbf{w}_i^t = [X_i^t, Y_i^t, \dot{X}_i^t, \dot{Y}_i^t]^T$ consists of the two-dimensional coordinates of the pedestrian on the ground plane and the 2D velocity components.

Temporal model: In our model the state vectors form a Markov chain over time. The state at time t depends on the state at time $t-1$ and the ground plane parameter π (see below). We describe the pdf for the state transition $P(\mathbf{w}_i^t|\mathbf{w}_i^{t-1}, \pi)$ as a Gaussian distribution with a linear function $\mu_+ = T\mathbf{w}_i^{t-1}$ of the preceding state as mean and the covariance Σ_+ (see Section 2.3) of the predicted state: $P(\mathbf{w}_i^t|\mathbf{w}_i^{t-1}, \pi) = N(\mu_+, \Sigma_+)$. T denotes the transition matrix and is defined as for the standard linear Kalman Filter with constant velocity assumptions (Kalman, 1960).

Ground plane: The ground plane π is defined in a Cartesian world coordinate system, where the X and Y axes point in the horizontal directions and Z is the vertical axis. π is the plane parallel to the X and Y axes of the coordinate system at a constant height below the camera, which is given in advance. We compute the position of a person in world coordinates as the intersection point of the image ray of the lowest visible point of a person (in our model given by x_i and y_i) and the ground plane.

2.3 Maximum a posteriori (MAP) estimation

For the computation of the posterior state $\mathbf{w}_i^t$ of our model an extended Kalman Filter model is used. As opposed to the traditional recursion between prediction and correction we apply an intermediate step for the computation of the image position variable $\mathbf{x}_i$ considered as hidden variable (see above), which then is used for the correction of the predicted state. The recursion hence consists of three steps:

i) **Prediction of the state vector**. The state vector is predicted in accordance with the temporal model, involving the uncertainty of the previous state Σ^{t-1} and the transition noise accounted for by Σ_p in the way that

$$P(\mathbf{w}_i^t|\mathbf{w}_i^{t-1}, \pi) = N(\mu_+, \Sigma_+) \tag{3}$$

with $\mu_+ = T\mathbf{w}_i^{t-1}$ and $\Sigma_+ = \Sigma_p + T\Sigma^{t-1}T^T$. We account for the transition noise by assigning a standard deviation of $\sigma_{Xi} = \sigma_{Yi} = \pm 0.3m$ and $\sigma_{\dot{X}i} = \sigma_{\dot{Y}i} = \pm 0.3\frac{m}{s}$ to the elements of Σ_p.

ii) **Estimation of the image position**. We estimate $\mathbf{x}_i^t$ by maximising the product of the probability terms relating the image position to the predicted state μ_+ and the observed and constant variables. The probability distributions involved in the estimation of $\mathbf{x}_i^t$ (except for $P(\mathbf{x}_i^t|O^t)$) are depicted in Figure 2.

$$P(\mathbf{x}_i^t|\mu_+, C^t)P(\mathbf{x}_i^t|O^t)P(\mathbf{x}_i^t|IP)P(c_{RF}^t|\mathbf{x}_i^t)P(c_{det}^t|\mathbf{x}_i^t) \rightarrow \mathbf{max}. \tag{4}$$

The value of $\mathbf{x}_i^t$ maximising the product in Equation 4 is used for the update step (see step iii)). There, the estimate of $\mathbf{x}_i^t$ is expected to follow a normal distribution, which we justify by the observation that the individual terms of Equation 4 are either equally distributed or resemble Gaussian distributions themselves (see also Figures 2(b) to 2(f)). The probability distribution related to the on-line Random Forest classifier usually peaks at the target's position and decreases radially and thus can be approximated by a Gaussian distribution as well (see Figure 2(d)). The probability distribution $P(\mathbf{x}_i^t|\mu_+, C^t)$ relating $\mathbf{x}_i^t$ to the predicted state (see Figure 2(e)) is used to support the estimation of $\mathbf{x}_i^t$, and also acts as a gating function by restricting the search space for the estimation of $\mathbf{x}_i^t$ to the 3σ-ellipse (projected into the image) given by the uncertainty about the predicted state Σ_+.

iii) **Update of the state vector**. The incorporation of the estimated image position into the recursive filter is conducted in accordance with the Kalman update equation.

$$E(\mathbf{w}_i^t) = \mu_i^t = \mu_+ + K(\tilde{\mathbf{w}}_i^t - \mu_+). \tag{5}$$

In Equation 5 K is the Kalman Gain matrix and $\tilde{\mathbf{w}}_i^t$ is the state computed from the projective transformation of the expected value of the image point $\mathbf{x}_i^t$ to the ground plane. Step iii) is conducted only if the product in Equation 4 exceeds a threshold. If this is not the case, the trajectory is only continued based on the prediction.

After steps i) to iii), the values for the unknown variables $\mathbf{x}_i^t$ and $\mathbf{w}_i^t$ maximising the joint pdf (see Equation 1) are determined. The estimate of the state vector is then used for the prediction step in the next recursion at the successive time step. From the estimated image position new training samples for the on-line Random Forest classifier are extracted as described in Section 2.1. During an occlusion the on-line Random Forest classifier is not updated.

2.4 Initialisation and termination

For the detection of new pedestrians we apply the strategy from (Klinger et al., 2014) and validate the outcomes of a HOG/SVM detector by two classifiers, one concerning the geometry of the search window $\mathbf{r} = [x_r, y_r, width_r, height_r]^T$ and the other concerning the confidence value given by the SVM (c_{SVM}). By classification, the probabilities $P(v|c_{SVM})$ and $P(v|\mathbf{r})$ for the classified position being either a person ($v = 1$) or background ($v = 0$) are obtained. A new trajectory is initialised if the decision rule in Equation 6 votes for a person and if according to the occlusion model (see Section 2.1) no trajectory of an existing target is predicted at the position of the search window.

$$v = \begin{cases} 1, & \text{if} \frac{P(v=1|\mathbf{r})P(v=1|c_{SVM})}{P(v=0|\mathbf{r})P(v=0|c_{SVM})} > 1, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

If the person is predicted to be occluded, the trajectory is updated only with respect to the temporal model (see Section 2.2). If the trajectory is occluded in more than 5 consecutive frames, or if a person leaves the image, the trajectory is terminated.

3. EXPERIMENTS AND RESULTS

Experiments are conducted on the *Bahnhof*-sequence of the ETHZ dataset (Ess et al., 2008) captured from a moving platform in a challenging outdoor scenario. For an automatic detection of pedestrians we apply the HOG/SVM-detector of OpenCV, which is trained with the INRIA person dataset (http://pascal.inrialpes.fr/data/human/). In our application only pedestrians with a minimum height of 96 pixels are considered. The HOG/SVM detector is configured without internal threshold, so that the results are as complete as possible. The bounding rectangles resulting from the HOG/SVM are decreased to account for the systematic margin of 16 pixels around people in the training

(a) Frame #502 (b) Frame #511 (c) Frame #526

(d) Frame #543 (e) Frame #554 (f) Frame #585

Figure 3: Exemplary tracking results achieved by the proposed system. Most pedestrians are tracked persistently throughout their presence in the images, though also early terminations of trajectories occur.

data. The detection results given by the HOG/SVM are evaluated at two different stages: first, for the identification and initialisation of new targets and second, for supporting the estimation of the image position (Equation 4). For the initialisation of new trajectories we apply the strategy described in Section 2.4. Every time a new position $\mathbf{x}_i^t$ of a pedestrian is determined, a bounding rectangle $T_i^t = [x_i^t, y_i^t, width_i^t, height_i^t]$ with $width_i^t$ and $height_i^t$ the width and height of the ellipse determined at the classification step (see Section 2.1), is assigned to the trajectory.

In Figure 3 six images taken from the test sequence with superimposed bounding rectangles T_i^t and trajectories are shown. Each tracked pedestrian is assigned a separate colour for the visualisation. As validated by visual inspection of the results, most pedestrians have been tracked by our system throughout their presence in the sequence. For a quantitative evaluation of the achieved tracking performance we build three different set-ups of tracking algorithms. In the first set-up tracking is conducted without recursive estimation or motion model so that the trajectory consists of the positions with the highest confidence achieved by the on-line Random Forest separately in each image (referred to as ORF). In the second set-up, the position with the highest confidence of the on-line classifier is introduced as an observation into an extended Kalman Filter (ORF&KF). The third set-up reflects the model proposed in this paper, modelling the image position as hidden variable (DBN).

We evaluate the tracking performance on the Bahnhof-sequence of 1000 images, split the data in two halves and apply cross-validation, using one half for learning the Interesting Places (Sec. 2.1) and the classifiers (Sec. 2.4), and the other half for testing. The Position Based Measure (PBM) (see, e.g. (Smeulders et al., 2013) for a reference) is computed as

$$PBM = \frac{1}{N_{TP}} \sum_i [1 - \frac{Distance(i)}{T_h(i)}], \qquad (7)$$

	PBM	Recall	Precision	ID-sw.
ORF	0.91	0.57	0.37	8
ORF&KF	0.93	0.52	0.56	7
DBN	0.94	0.59	0.55	3

Table 1: Results of the investigated set-ups using the on-line Random Forest separately (ORF), together with a standard Kalman Filter model (ORF&KF) and the proposed method (DBN).

with N_{TP} the number of true positive detections, $Distance(i)$ the $L1$-norm distance between the automatic detection (T_i) and a reference result GT_i. $T_h(i)$ is defined as $T_h(i) = (width(T_i) + height(T_i) + width(GT_i) + height(GT_i))/2$. Only automatic detection results with an overlap of at least 50% between T_i and GT_i are counted as correct. For the three set-ups the achieved PBM, the recall and precision rates, as well as the total number of ID-switches are given in Table 1.

The results demonstrate the benefit of using the proposed method. If tracking is conducted using only the on-line Random Forest classifier, the geometric accuracy in terms of the PBM with a score of 0.91 performs worst among the applied set-ups. The results improve when recursive estimation in form of Kalman filtering is applied. By estimating the state vector using the Kalman Filter, the position in the image is constrained by a motion model, which keeps the track close to a more plausible path (if the motion model is correct). Using this model we obtain a slightly improved geometric accuracy. If the position in the image is modelled as a hidden variable, as in our approach, the geometric accuracy is further improved to a PBM value of 0.94. The principal difference between the second and the third set-ups is the way in which the image position is modelled. Since the image position essentially contributes to the posterior state of the trajectory, a correct value for the image position is crucial. When modelling this position as a hidden variable, its accuracy can be improved by considering further information, here the detection and classification results,

occlusion model and prior information about the scene, before it is used for the update of the filter. Furthermore, the consideration of additional observations in our model decreases the risk of identity switches, compared to the trajectory estimation using the on-line classifier or the Kalman Filter only.

However, the recall and precision rates indicate that only every second decision concerning the presence or absence of a pedestrian is correct and that about every second pedestrian present in the scene is not obtained (evaluated in each frame). Also the geometric accuracy achieved with the proposed method, although best between the three investigated set-ups, can be further improved. We encounter two major causes of geometric inaccuracy in our system. First, the method relies on the initialisation which is taken from the HOG/SVM detector. If this image position is imprecise due to a misaligned initial detection, erroneous training samples are derived, which keeps the classifier attracted to regions with an offset from the actual position of the pedestrian. In this case the trajectory hardly converges towards the correct position during further tracking. When this happens, the automatically determined position might not overlap sufficiently with the reference data, so that the number of false positive detections increases, although the pedestrian is being correctly tracked, however with a lower accuracy. In this way also new initialisations can be prevented, increasing thus the number of false negatives as well. Second, in crowded scenarios interactions between pedestrians take place. If two pedestrians appear next to each other, the gap normally visible between them might vanish, giving rise to ambiguities both in the classification with the HOG/SVM and with the on-line classifier. A position can hence be inaccurate, if (at least) two pedestrians interact.

4. CONCLUSIONS

In this paper we proposed a probabilistic model designed for the task of visual pedestrian tracking. The pedestrian state (position and velocity) in world coordinates and the corresponding position in the image are modelled as hidden variables in a Dynamic Bayes Network. The network combines a dynamic model, prior scene information, a state-of-the-art pedestrian detector, and a classifier with on-line training capability in a single framework. The results show that the derived trajectories achieve a geometric accuracy superior to the one obtained by processsing each frame of the image sequence individually or by using a standard Kalman Filter. To overcome problems of our approach, we will focus on an improvement of the geometric accuracy particularly at the initialisation step of the tracking method in future work. To better dissolve ambiguities in the trajectory update, a trajectory optimisation can be conducted on a global level, considering also time steps further in the past. Also a better pedestrian detector, which detects body parts or pairs of pedestrians will be applied. More comprehensive experiments including the suggested improvements will be conducted in future work.

References

Bishop, C., 2006. Pattern recognition and machine learning. Vol. 4, springer New York.

Breitenstein, M., Reichlin, F., Leibe, B., Koller-Meier, E. and Van Gool, L., 2011. Online multi-person tracking-by-detection from a single, uncalibrated camera. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(9), pp. 1820–1833.

Comaniciu, D., Ramesh, V. and Meer, P., 2003. Kernel-based object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(5), pp. 564 – 577.

Dai, Q. and Hoiem, D., 2012. Learning to localize detected objects. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE, pp. 3322–3329.

Dalal, N. and Triggs, B., 2005. Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, Vol. 1, IEEE, pp. 886–893.

Dean, T. and Kanazawa, K., 1989. A model for reasoning about persistence and causation. Computational intelligence 5(2), pp. 142–150.

Eshel, R. and Moses, Y., 2010. Tracking in a Dense Crowd Using Multiple Cameras. International Journal of Computer Vision 88(1), pp. 129–143.

Ess, A., Leibe, B., Schindler, K., and van Gool, L., 2008. A mobile vision system for robust multi-person tracking. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08), IEEE Press, pp. 1–8.

Kalal, Z., Matas, J. and Mikolajczyk, K., 2010. Pn learning: Bootstrapping binary classifiers by structural constraints. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, pp. 49–56.

Kalman, R. E., 1960. A new approach to linear filtering and prediction problems. Journal of basic Engineering 82(1), pp. 35–45.

Klinger, T. and Muhle, D., 2012. Persistent object tracking with randomized forests. In: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XXXIX-B3, pp. 403-407.

Klinger, T., Rottensteiner, F. and Heipke, C., 2014. Pedestrian recognition and localisation in image sequences as bayesian inference. In: Kukelov Z, Heller J. (Eds.), Proceedings of the Computer Vision Winter Workshop 2014 , Czech Society for Cybernetics and Informatics, pp. 51-58.

Menze, M., Klinger, T., Muhle, D., Metzler, J. and Heipke, C., 2013. A stereoscopic approach for the association of people tracks in video surveillance systems. PFG Photogrammetrie, Fernerkundung, Geoinformation 2013(2), pp. 83–92.

Milan, A., Roth, S. and Schindler, K., 2014. Continuous energy minimization for multi-target tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 36(1), pp. 58–72.

Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B. et al., 2002. Fastslam: A factored solution to the simultaneous localization and mapping problem. In: AAAI/IAAI, pp. 593–598.

Ommer, B., Mader, T. and Buhmann, J. M., 2009. Seeing the objects behind the dots: Recognition in videos from a moving camera. International Journal of Computer Vision 83(1), pp. 57–71.

Saffari, A., Leistner, C., Santner, J., Godec, M. and Bischof, H., 2009. On-line random forests. In: Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, IEEE, pp. 1393–1400.

Schindler, K., Ess, A., Leibe, B. and Van Gool, L., 2010. Automatic detection and tracking of pedestrians from a moving stereo rig. ISPRS Journal of Photogrammetry and Remote Sensing 65(6), pp. 523–537.

Smeulders, A., Chu, D., Cucchiara, R., Calderara, S., Dehghan, A. and Shah, M., 2013. Visual tracking: An experimental survey. IEEE Transactions on Pattern Analysis and Machine Intelligence PP(99), pp. 1–1.

Viola, P. and Jones, M., 2001. Robust real-time object detection. International Journal of Computer Vision 57(2), pp. 137–154.