A modular tracking system for far infrared pedestrian recognition

E. Binelli, A. Broggi, A. Fascioli, S. Ghidoni and P. Grisleri
Dipartimento di Ingegneria dell'Informazione
Università di Parma
Parma, I-43100, Italy
email: {broggi, fascal, ghidoni, grisleri}@ce.unipr.it

T. Graf, M. Meinecke
Electronic Research
Volkswagen AG
Wolfsburg, D-38436, Germany
email:{thorsten.graf, marc-michael.meinecke}@volkswagen.de

***Abstract*— This paper describes a modular tracking system designed to improve the performance of a pedestrian detector. The tracking system consists of two modules, a *labeler* and a *predictor*. The former associates a tracking identifier to each pedestrian, keeping memory of the past history; this is achieved by merging the detector and predictor outputs combined with data about vehicle motion. The predictor, basically a Kalman filter, estimates the new pedestrian position by observing his previous movements. Its output helps the labeler to improve the match between the pedestrians detected in the new frame and those observed in the previous shots (feedback). If a pedestrian is occluded by some obstacle for a short while, the system continues tracking its movement using motion parameters. Moreover, it is able to reassign the same tracking ID in case the occlusion disappears in a short time. This behavior helps to correct temporary mis-recognitions that occur when the detector fails. The system has been tested using a quantitative performance evaluation tool, giving promising results.**

I. Introduction

Obstacle detection and classification in road scenes is one of the most challenging tasks in the artificial vision field. In recent years, several researches have been focused on the detection of pedestrians, because of the great impact on road safety.

Recent works on pedestrian detection using far infra-red (FIR) images achieved promising results, thanks to the specific appearance of human beings in the FIR domain: effective algorithms have been proposed [1], [2], [3], [4] to detect people in this domain. However, pedestrians can appear in a wide variety of clothings, postures, and sizes, therefore even sophisticated detectors turn out to fail in some cases. Detection can be affected by three major types of errors:

(i) pedestrians may be unrecognized in some frames due to noise or unusual dresses or postures,
(ii) complex scenarios featuring objects resembling the human shape may lead to false detections,
(iii) the pedestrian is detected but some parameters (e.g. size or distance) are erroneously estimated by the algorithm.

To improve the robustness of pedestrian recognition, a system should avoid missing critical targets and correctly measure the target parameters. It is of paramount importance to achieve high detection rates while keeping the false detection rate low, since false positives decrease the driver's trust in the system.

Vision based detection systems can exploit temporal correlation and enforce the processing of the current frame with the results of the previous frames. In fact, by taking into account the data obtained in the past history it is possible to strengthen the decisions on the current frame. If a pedestrian is detected in one single frame over a sequence of several frames it is probably a false positive; on the other hand, if a detection occurred continuously in a sequence and disappears abruptly in the current image, then the detector is probably producing a wrong result.

The tracking technique uses data about past detections (such as size and position of the target) and the detection output in the current scene, and predicts, with small errors, where a pedestrian will be in the future frame. This allows to follow targets even if in the next frame the detector fails the recognition. Moreover, the system is capable of filtering out spurious recognitions, reducing the number of false positives.

Measurements obtained using vision sensors are affected by intrinsic errors due to quantization. These errors depend on the distance of the observed object. Using past measurements, the precision of the estimated target distance can be increased.

This work presents a modular tracking system that enforces the pedestrian detector under development at the Artificial Vision and Intelligent System Laboratory of the University of Parma in collaboration with Volkswagen AG. The work is based on a previously developed system able to detect pedestrians in front of a moving vehicle and to measure their distance [5]. The proposed tracking system integrates the use of off-the-shelf additional sensors, mounted on current common vehicles, to obtain a better estimation of ego-motion parameters.

Section II presents an overview on the state of the art about tracking for pedestrian detectors. Section III describes the system structure. Results obtained comparing with the original detector performance are discussed in section IV. Section V contains some final considerations regarding the overall system behavior and the guidelines for future developments.

II. State of the art

Several studies presented in the literature describe systems for pedestrians tracking in video sequences taken by cameras mounted on a vehicle [3], [4]. Some of them use single or stereo[2] infrared cameras as source. Classic tracking techniques are used in some of these systems. In [6] a Kalman based tracker is used as a source to locate moving pedestrians. A mean-shift method is used to find the exact position of the target around the a posteriori position.

In order to reduce the high computational cost of the detection due to image analysis, some systems follow the target in each frame using the predictions given by tracking and perform a new detection only when needed or every fixed number of frames; motion estimation parameters are necessary to minimize Kalman filter errors. In [7] sensors available in off-the-shelf vehicles are integrated with a laserscanner and a FIR camera in order to enforce both ego-motion and obstacle distance estimation.

In this paper a new modular approach that combines Kalman filter prediction with past history analysis is presented. The contribute of this work is the introduction of a complete system integrating all the leading edge technologies in the field of pedestrians detection for vehicular applications.

A high performance detector produces a list of pedestrians in both image and 3D world coordinates. A prediction module that uses Kalman filter and the previous overall system output generates a second list with the expected new position and size for each pedestrian of the previous frame. Using the two lists and a convenient set of rules, each pedestrian is associated to a unique label or tracking identifier. Moreover, data acquired from sensors installed on off-the-shelf vehicles, such as odometers and inertial sensors, are used to correct the Kalman filter prediction error due to vehicle motion. This kind of ego motion correction, based on measurements from other sensors is more robust than any estimation made using vision.

III. System structure

The system is composed of three modules, as shown in figure 1. The pedestrian detector [8] is the module that analyzes data coming from the FIR camera mounted on the front bumper of the vehicle. It is based on a multi-resolution localization of warm symmetrical objects with specific size and aspect ratio; the multi-resolution approach allows to detect both close and far pedestrians. A match against a set of 3D models encoding human shape's morphological and thermal characteristics is used as a validation process to remove false positives. The detector output is a list of bounding boxes in image coordinates, each containing a pedestrian.

The labeler is the central module that performs associations and high-level tasks such as obstacle motion interpretation. It keeps track of bounding boxes found in a certain number M of past frames. Bounding boxes detected in the current frame are compared to those found in the last M frames and with predictions made in the past by the predictor.

The predictor is a module that implements techniques to estimate the future position of each target through the observation of the evolution of its position.

A. Labeler

The labeler gets from the detector a list of bounding boxes found in the current frame. For each new bounding box the labeler estimates the overlapping with all the boxes found in the previous frame: if the overlapping area is larger than a threshold, which decreases with the obstacle distance, an association is established between the two bounding boxes, with a specific probability.

The generation of associations is performed in the 2D image. The use of the 3D space has been avoided because the uncertainty of measures made by a monocular vision system increases with the distance. When the detector locates a pedestrian, the estimation of its baseline (the bounding box bottom) can shift by few pixels across different frames, and for far distances an offest of a few pixels corresponds to a few meters. If this erroneous detection was used as input for a 3D labeler, it would have been necessary to associate bounding boxes that are even a few meters far apart in the 3D world. There are obviously a number of cases in which this assumption leads to a wrong behavior. Labeler associations must rely on a robust distance estimation; this could be obtained for example by integrating a radar sensor in the system.

At this step a tracking label is assigned to each bounding box. If no association with the previous frame is found, the candidate is classified as a new entry, but it is not immediately fed to the output as a pedestrian. To do that it must remain in the tracker loop for a given number of frames[1]. A bipartite graph is generated to maintain associations, as shown in figure 2. Each class of the graph is the set of bounding boxes found in a frame by the detector. Arcs are used to represent associations. For each arc an association value is computed. Each new bounding box can be associated with at most all bounding boxes in the previous frame. This association is based on the assumption that a pedestrian does not move significantly in the image between two frames; this is particularly true at the frame rates reached by real time applications (at least 10 fps) if the vehicle goes up to 40 km/h along a straight way. The next step is the analysis of the associations already decided in order to preserve the same label for new occurrences of the same objects. To achieve this, a time window of several frames (defined as parameter during the initialization) describing the detector result and the corresponding associations between bounding boxes for each frame, is maintained in memory. An example of this trellis structure is depicted in figure 2, where a series of four frames and the relative associations between bounding boxes are defined. For each new bounding box that has to be produced in output (i.e. having a lifetime of at least one cycle) an association tree is generated. The trellis is visited

[1]One in the current implementation.

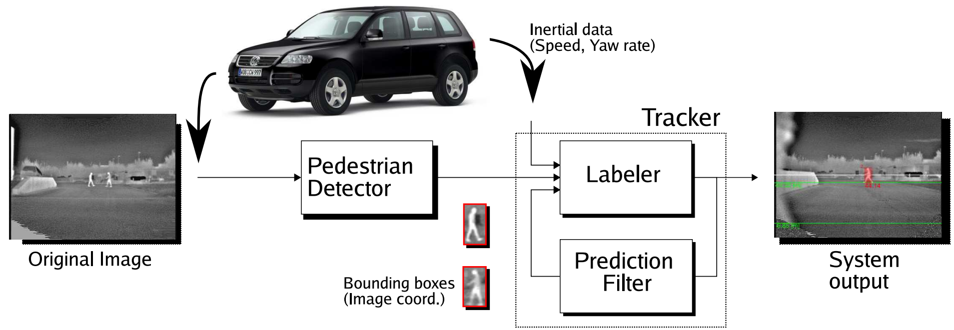


Fig. 1. The system structure. The labeler takes as input the pedestrian detection output, inertial data and the prediction about future position of past results.

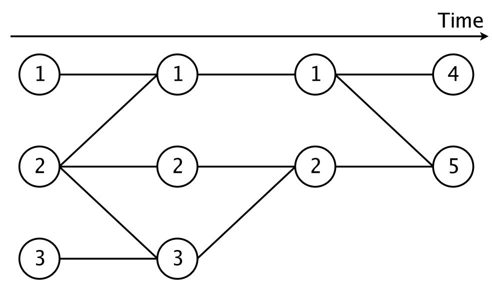


Fig. 2. Trellis maintained inside the labeler. Each node represents a bounding box with its associated label. Each column represents bounding boxes found in a frame. The rightmost column is the last detected frame.

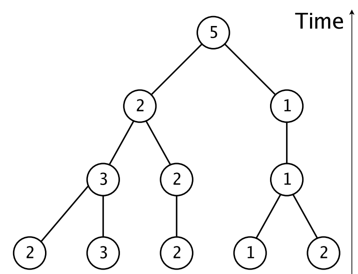


Fig. 3. Example of binary tree obtained from trellis during the assignment of node '5' label.

backwards to build the tree. The tree related to the bounding box labeled with number '5' in the trellis of figure 2 is shown in figure 3.

The tracking label for the bounding box in the root node is generated as follows: a temporary new label is assigned to the root node. Then, a set of possible labels is populated visiting the whole tree. In this set, the label that will substitute that of the root node is selected as the one with the highest association index A_L, if it overcomes a threshold. This index is computed as follows:

$$A_L = \sum_{i=1}^{M-1} \sum_{j=0}^{N_{L,i}} p_j(L) e^{-i} \tag{1}$$

where $N_{L,i}$ is the number of bounding boxes of the i-th level of the tree with label L, and $p_j(L)$ is the corresponding association probability.

As an example, the bounding box temporary labeled as '5' can assume the label '1', '2' or '3'. For each node of the tree a value, that depends on the association probability between the two considered bounding boxes, is computed. This value decreases exponentially as a function of the difference between the current frame timestamp and that containing the considered bounding box. The frame timestamp, related to all bounding boxes in a certain frame, is maintained in the trellis. The computed value associated to each tree node depends also on lost detection life-time which is defined below. The probability of the label associated to the node can finally be updated. The label that assumes the larger value will be assigned to the bounding box under examination.

To obtain a reliable output and filter out spurious recognitions, the labeler assumes a pedestrian has been found only when it has been detected in two consecutive frames. It also tries to avoid false negatives: when the detector fails a recognition, there is a pedestrian present in the past that is not associated with any bounding box in the current frame. When this situation happens, the labeler inserts a bounding box in the position given by the prediction filter for the current frame, and classifies it as a lost detection, as explained in the following.

1) Lost detections: The detection phase is sometimes affected by discontinuities: in urban traffic scenes, for instance, pedestrians can be occluded for a few frames by other objects, such as cars or other people. Furthermore, the detector itself sometimes fails to recognize a pedestrian in a single frame. To avoid a discontinuous recognition, the labeler considers not only bounding boxes found in the current frame by the detector, but also those that were detected in the past, labeling them as *lost detections*.

A lost detection is a pedestrian generated in the loop composed by the labeler and predictor; its presence is not confirmed by image analysis, and its position is determined by a prediction: for these reasons, it should be somehow distinguished from a pedestrian actually found in the image. Lost detection life-time is the time spent by such a bounding box in the loop. Probably, a lost detection is re-associated, at the following cycle, to a bounding box found by the detector. If this happens, it becomes a pedestrian, that will have the same label of the lost detection, like depicted in figure 4.

In the opposite case, it survives in the loop, without confirmation from the detector, for a time that depends on his previous detection probability. This probability decreases, as the corresponding pedestrian is not detected in the following images, under a decreasing exponential law. When the detection probability becomes too low, the lost detection vanishes, because the corresponding pedestrian is not detected anymore, and this means it really disappeared from the scene. Furthermore, it would be very unreliable to determine the position of a pedestrian using many predictions not confirmed by any detection.

2) Vehicle motion estimation: The most limiting constraint of tracking methods for vehicular applications concerns the vehicle movements. In fact, when the vehicle moves, particularly along a bend, an object may appear in two consecutive frames in completely different positions, so that bounding boxes referring to the same pedestrian can have no overlapping. The prediction filter cannot produce right estimations, because the apparent movement in the image plane is irregular and far unpredictable. To cope with this problem, additional information is supplied as input to the labeler for every processed frame: velocity and yaw-rate. These data come from sensors commonly available on current production vehicles, and are gathered through the CAN bus. By knowing car movements, and then also camera movements, it is possible to estimate how the scene will change in the next frame; this way the labeler can associate bounding boxes correctly, even if they are in different parts of the image.

The vehicle trajectory can be locally seen as an arc, whose radius is infinite if the route is straight; this movement is the composition of a translation and a rotation occurring at the same time. For simplicity sake, the two movements are considered separately. As a starting point, knowing the velocity, yaw-rate and time T elapsed between the current frame and the previous one, the space driven by the car and its rotation can be calculated.

To evaluate the effect of rotation, the rotation angle (*rot*) is related to camera parameters by dividing it by the camera aperture angle α; then, to compute how this ratio affects object movements in the image, it is multiplied by the image width in pixel, IW:

$$d = \frac{rot}{\alpha} IW. \tag{2}$$

The number d then represents the movement of all objects in the image plane, expressed in pixels, regardless of their distance; positive values correspond to translations to the right, because the yaw-rate is positive when the vehicle turns left. Such a calculation contains some approximations, but it is very fast to be performed, and an error of few pixels does not significantly affect the performance of the labeler.

The second movement analyzed is the translation. To take into account the optical flow, each bounding box in the image plane is mapped into 3D coordinates, and then it is translated according to the distance covered by the vehicle. The translated bounding box is finally remapped in 2D image plane coordinates. The 2D to 3D mapping process computes the distance of a pedestrian from the camera analyzing the base of each bounding box. Therefore, for the optical flow computation to work properly, all bounding boxes' bottom limits should be accurately positioned, otherwise the distance from the camera is not correctly evaluated. This causes a wrong 3D mapping and, finally, erroneous associations.

The main contribution to the apparent movements in the image is due to the rotation, therefore the translation is neglected; the perspective effect due to translation, in fact, does not significantly affect associations.

B. Motion Prediction

The system should be able to continue the tracking even if the detector fails the recognition for few frames; thus, some predictions of each pedestrian's future movements should be made. Prediction filters perform this task by means of observation of the past movements: at each frame, the input of the filter is the position of each pedestrian found by the detector, while its output is the predicted position in the next frame; of course, the output becomes accurate when a few frames have been analyzed. The predictor works in the 2D image plane domain, and its target is the central point of the bottom of each bounding box. Also a three dimensional approach was attempted, but it gave worse results. This is because the distance of a pedestrian from the camera is calculated observing the position of the lower limit of the bounding box: at distances of 35 m, a pixel corresponds to 5 m and the resulting longitudinal coordinate is very inaccurate, and not suitable to be tracked.

There are two well-known approaches to this problem: the α-β-γ filter and the Kalman filter; both were applied, since the system is modular, and it is very easy to substitute one with another. Their behavior automatically adjusts to the acquisition period, because calculated velocity and acceleration are those between two subsequent frames.

1) α-β-γ filter: This filter represents the simplest solution. It is a one-step ahead predictor that expresses the expected position x_p of a pedestrian in terms of the corresponding smoothed position x_s, velocity v_s and acceleration a_s, like follows, [9]:

$$x_p(k+1) = x_s(k) + T v_s(k) + \frac{1}{2} T^2 a_s(k), \tag{3}$$

where k denotes a generic discrete time instant, and T is the period between two subsequent frames. There is also a similar equation for the predicted velocity v_p:

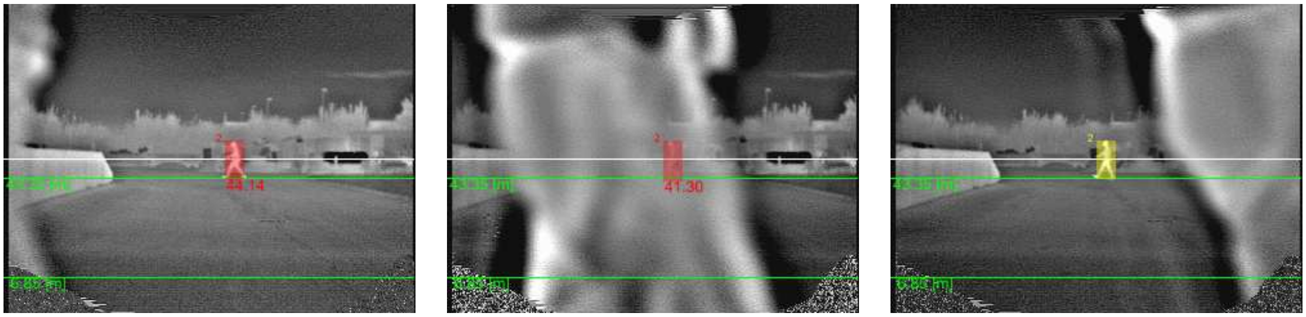
$$v_p(k+1) = v_s(k) + T a_s(k). \tag{4}$$

The smoothed parameters are calculated as weighted sum of estimates and evaluation error $(x_o(k) - x_p(k))$, called innovation:

$$x_s(k) = x_p(k) + \alpha\,(x_o(k) - x_p(k)) \tag{5}$$

$$v_s(k) = v_p(k) + \frac{\beta}{T}\,(x_o(k) - x_p(k)) \tag{6}$$

$$a_s(k) = a_s(k-1) + \frac{\gamma}{2T^2}\,(x_o(k) - x_p(k)), \tag{7}$$



(a) (b) (c)

Fig. 4. Example of lost detection: in (a) the pedestrian is visible and a correct recognition occurs. In (b) the camera is occluded and the labeler generates a lost detection. In (c) the pedestrian reappears and it takes the same label it had in (a).

where x_o is the observed (measured) position. The weighting factors, α, β and γ, have effects on the filter reactions to discontinuities of the target position, and should be chosen according to stability conditions, [10]. The approach of this kind of predictor is one-dimensional, so two filters should be used to track each pedestrian in the image plane.

2) Kalman filter: This prediction technique is more sophisticated, and needs a model of the observed system, defined by some vectors and matrices, [11]. The state vector used is composed by six variables: position, velocity and acceleration in the horizontal and vertical directions of the image plane, therefore it is a constant acceleration filter. The state vector is:

$$x = \begin{bmatrix} x & v_x & a_x & y & v_y & a_y \end{bmatrix}^T, \tag{8}$$

where M^T is the transpose of the matrix M. The filter works in two steps called prediction and correction: the former tries to evaluate the future state of the system, while the latter corrects the prediction using the measurement of some state variables. Kalman filter also takes into account the noise of the system, dynamically calculating the Kalman gain, a parameter used in the computation of the future state. The gain acts as a weight between measurement and past prediction. The prediction phase calculates the future state and the a priori estimate error covariance, given by the following equations, respectively:

$$\hat{x}^-(k+1) = Ax(k) \tag{9}$$
$$\hat{P}^-(k+1) = AP(k)A^T + Q, \tag{10}$$

where P is the a posteriori estimate error covariance and Q is the process noise covariance. In the correction step, three parameters are evaluated: Kalman gain $K(k)$, a posteriori state estimate $\bar{x}(k)$ and a posteriori error covariance:

$$K(k) = P^-(k)H^T\left(HP^-(k)H^T + R\right)^{-1} \tag{11}$$
$$\hat{x}(k) = \hat{x}^-(k) + K(k)\left(z(k) - H\hat{x}^-(k)\right) \tag{12}$$
$$P(k) = (I - K(k)H)P^-(k), \tag{13}$$

in which R is the measurement noise covariance, I is the identity matrix, $z(k)$ is the vector of measurements and H is a matrix such that:

$$z(k) = Hx(k) + \nu(k), \tag{14}$$

where ν is the measurement noise. By applying these equations, the Kalman filter is able to predict the most probable future state vector, that is future position, velocity and acceleration.

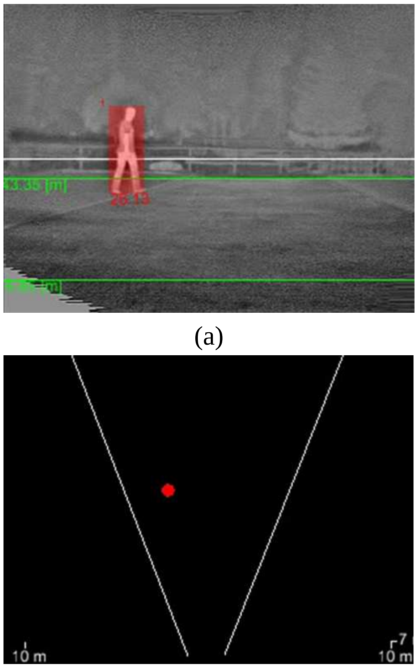
IV. Results

An example of the output provided by the tracking system is displayed in figure 5. Both a perspective view and bird-eye view of the scene are given.

The perspective view shows a shaded bounding box overlapped to each detected pedestrian. The distance and tracking label are also displayed. The tracker assigns a probability value to each pedestrian. This value is used to control the pedestrian alpha channel in the visualization of results: the more the detection probability is low, the more the bounding box looks transparent. Conversely, the distance of pedestrians is used to control the visualization color. Distant pedestrians are displayed in yellow and near pedestrians in red, and the color changes accordingly to a law integrated into the display function.

The bird-eye view shows a map of the scene in front of the vehicle as seen from above, like displayed in figure 5(b). For each pedestrian a marker is displayed in correspondence to its position.

The system has been tested on several sequences acquired in different conditions of weather, scene, and environment. Tests have shown that the considered approach behaves well both when the vehicle is still and moving, even along a bend. This means that the ego-motion analysis performed using CAN data gives good results, since the detector alone, on the same sequence, failed most of times. Problems still remain when a pedestrian enters the scene running. In this case, even when the vehicle is still, the recognition fails because correct associations cannot be made since the Kalman filter cannot be appropriately trained. Moreover, camera pitch decreases the overlapping between bounding boxes in subsequent frames, this reducing the reliability of associations.



(a)

(b)

Fig. 5. The perspective (a) and bird-eye (b) views.

Quantitative results have been assessed using a performance evaluation tool discussed in [12]. The integration of the tracking system improves the detection rate (+4%).

Time performance have been evaluated as well. The complete system (detector plus tracker) runs on a 2.8 GHz 1 GB RAM Pentium 4 in an average time of about 80 ms. The introduction of the tracker did not significantly affect the processing time.

V. Conclusions and further developments

In this paper a new modular approach to the task of tracking pedestrians in image sequences has been presented. The tracking system is capable of improving the performance of a pedestrian detector combining Kalman filter prediction with past history analysis. This behavior helps to correct temporary miss-recognitions that occur when the detector fails. Moreover, occasional false positives are reduced thanks to the histeresys necessary to obtain a tracking identifier. The contribute of this work is the introduction of a complete system integrating all the leading edge technologies in the field of pedestrian detection for vehicular applications.

The system can be further improved by adding the capability of understanding whether a pedestrian is in a dangerous situation or not. This is possible by relating its position and motion to the vehicle's trajectory, and analyzing if a collision is probable. All necessary data are already available to the system, since the prediction filter has knowledge of each pedestrian's movements, and a precise estimation of the vehicle's future position can be drawn from CAN data.

The association mechanism described in this paper exploits information on the yaw-rate of the vehicle to efficiently track pedestrians while the vehicle is driving along a bend. The effect of rotation is the most evident one, but another perspective effect still exists when the vehicle is in motion along a straight way. Thus, the labeler can be further improved by adding the capability to analyze the effect of perspective when the vehicle moves straight. In fact, even if the labeler works correctly also without perspective motion analysis, this would lead to a more robust pedestrian tracking. Another improvement regarding the labeler concerns the capability of keeping the correct labels when two or more pedestrians are occluded. For example, if the trajectories of two pedestrians in the scene cross each other, the labeler is not capable of keeping both labels during the crossing, and at least one pedestrian is given a new label. This happens because the future position of each pedestrian is predicted one step ahead: at each frame, the filter predicts the position only in the next one; on the other hand, a long term prediction could correctly associate pedestrians even after their trajectories crossed each other. Improvements will also involve some code optimization, which can speed up the handling of the structures used to keep memory of the past.

Acknowledgment

This work was funded by Volkswagen AG.

References

[1] H. Nanda and L. Davis, "Probabilistic Template Based Pedestrian Detection in Infrared Videos," in *Procs. IEEE Intelligent Vehicles Symposium 2002*, Paris, France, June 2002.

[2] X. Liu and K. Fujimura, "Pedestrian Detection using Stereo Night Vision," *IEEE Trans. on Vehicular Technology*, vol. 53, no. 6, pp. 1657–1665, Nov. 2004, iSSN 0018-9545.

[3] Y. Fang, K. Yamada, Y. Ninomiya, B. K. P. Horn, and I. Masaki, "A Shape-independent Method for Pedestrian Detection with Far-infrared Images," *IEEE Trans. on Vehicular Technology*, vol. 53, no. 6, pp. 1679–1697, Nov. 2004, iSSN 0018-9545.

[4] A. Broggi, M. Bertozzi, , R. Chapuis, F. C. A. Fascioli, and A. Tibaldi, "Pedestrian Localization and Tracking System with Kalman Filtering," in *Procs. IEEE Intelligent Vehicles Symposium 2004*, Parma, Italy, June 2004, pp. 584–589.

[5] M. Bertozzi, A. Broggi, A. Fascioli, T. Graf, and M.-M. Meinecke, "Pedestrian Detection for Driver Assistance Using Multiresolution Infrared Vision," *IEEE Trans. on Vehicular Technology*, vol. 53, no. 6, pp. 1666–1678, Nov. 2004, iSSN 0018-9545.

[6] F. Xu and K. Fujimura, "Pedestrian Detection and Tracking with Night Vision," in *Procs. IEEE Intelligent Vehicles Symposium 2002*, Paris, France, June 2002.

[7] U. Scheunert, H. Cramer, B. Fardi, and G. Wanielik, "Multi Sensor based Tracking of Pedestrians: a Survey of Suitable Movement Models," in *Procs. IEEE Intelligent Vehicles Symposium 2004*, Parma, Italy, June 2004, pp. 774–778.

[8] A. Broggi, A. Fascioli, M. Carletti, T. Graf, and M.-M. Meinecke, "A Multi-resolution Approach for Infrared Vision-based Pedestrian Detection," in *Procs. IEEE Intelligent Vehicles Symposium 2004*, Parma, Italy, June 2004, pp. 7–12.

[9] D. Tenne and T. Singh, "Analysis of alpha-beta-gamma Filters," in *IEEE International Conference on Control Applications*, Kohala Coast-Island of Hawai'i, Hawai'i, USA, Aug. 1999.

[10] Y. Kosuge and M. Ito, "A Necessary and Sufficient Condition for the stability of an alpha-beta-gamma Filter," in *SICE 2001*, Nagoya, July 2001.

[11] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," Department of Computer Science, University of North Carolina at Chape Hill, Tech. Rep. NC 27599-3175, Apr. 2004.

[12] M. Bertozzi, A. Broggi, P. Grisleri, A. Tibaldi, and M. D. Rose, "A Tool for Vision based Pedestrian Detection Performance Evaluation," in *Procs. IEEE Intelligent Vehicles Symposium 2004*, Parma, Italy, June 2004, pp. 784–789.