Aalborg Universitet

Trajectory Analysis and Prediction for Improved Pedestrian Safety

Integrated Framework and Evaluations

Møgelmose, Andreas; Trivedi, Mohan M.; Moeslund, Thomas B.

Trajectory Analysis and Prediction for improved Pedestrian Safety: Integrated Framework and Evaluations

Andreas Møgelmose[1,2], Mohan M. Trivedi[2], and Thomas B. Moeslund[1]

Abstract— This paper presents a monocular and purely vision based pedestrian trajectory tracking and prediction framework with integrated map-based hazard inference. In Advanced Driver Assistance systems research, a lot of effort has been put into pedestrian detection over the last decade, and several pedestrian detection systems are indeed showing impressive results. Considerably less effort has been put into processing the detections further. We present a tracking system for pedestrians, which based on detection bounding boxes tracks pedestrians and is able to predict their positions in the near future.

The tracking system is combined with a module which, based on the car's GPS position acquires a map and uses the road information in the map to know where the car can drive. Then the system warns the driver about pedestrians at risk, by combining the information about hazardous areas for pedestrians with a probabilistic position prediction for all observed pedestrians.

I. INTRODUCTION

As technology advances, Advanced Driver Assistance Systems (ADAS) becomes more and more commonplace in today's cars. ADASs can range from parking assistance to safety systems such as lane departure warning, and all the way to autonomous driving in stop-and-go traffic. Our focus is on safety systems related to pedestrians. In 2009 there were 4,000 deaths and 60,000 injuries from pedestrian-vehicle collisions in the US alone [1]. Since a pedestrian is much more vulnerable than people in cars, even slow speed accidents can prove deadly.

There has been a wealth of good work done on pedestrian detection [2]. There has been comparatively little research on what to do with these detections. This paper is concerned with how to use pedestrian detections in a driver assistance context. While driving, some observations are very important: pedestrians in front of the car, traffic signals, other road users. Some are less important: pedestrians on the sidewalk, "no parking" signs when the driver is not trying to park, billboards along the road. Sometimes too much information is ignored [3], leading to accidents, e.g. from overlooking a pedestrian. This is where predictive decision support ADASs can help [4]–[6].

An ADAS which aims to help in preventing pedestrian-vehicle accidents must prioritize pedestrians and only inform the driver about those who are in immediate risk of being hit. Since pedestrians often move around, some predictive ability is desired for the risk assessment.

The system presented in this paper is a monocular pedestrian tracking system. It has two parts:

1) From input of detection bounding boxes in a monocular view, it tracks pedestrians in a top-down map-like view in the area in front of the car. Using the tracks, the motion of the pedestrians in the immediate future can be predicted.
2) Based on the position and orientation of the vehicle, obtained from a highly sensitive GPS module, it retrieves a map of the area and uses the information about road locations from this to infer dangerous areas for tracked pedestrians.

The remainder of this paper is structured as follows: In section II, a brief overview of related work is given. Section II gives an overview of the system, and further details are explained in section IV and V. Section VI shows system output, and finally section VII rounds off the paper.

II. RELATED STUDIES

Plenty of work has been done on pedestrian detection in the past decade. The classic approaches are Haar-cascades [7] and HOG-SVM [8]. These two works form the foundation for much of the more recent work, from a combination of the two methods in [9], to the deformable parts model championed by Felzenszwalb et. al. [10], and Integral Channel Features [11], [12]/Aggregated Channel Features [13] by Dollár et. al. For a comprehensive overview of pedestrian detection methods, see [14].

What all these methods have in common is that they find the pedestrians, but do no further analysis. Recently, pedestrian intent prediction has gained traction, championed by the group of prof. Dariu Gavrila. There are two basic approaches: Tracking pedestrians, or looking at pedestrian orientation and local motion features, such as optical flow, of the pedestrian.

The papers [15]–[17] look at pedestrian orientation using different kinds of classifiers on static monocular pedestrian images. [18]–[20] do the same, but based on RGB-D data, and [20] even determines the orientation of the head and the torso separately. In [21], local motion featues dubbed MCHOG are used to predict whether or not the pedestrian is about to take a step. The input data, however, is not coming from a car perspective, but a stationary multi-camera setup mounted at an intersection. A similar task is carried out in the very interesting [22], which uses optical flow and stereo data. This time on data from a real car, though in rather simplistic scenarios.

[1]Visual Analysis of People Lab, Aalborg University, Denmark `[am, tbm]@create.aau.dk`
[2]Laboratory for Intelligent and Safe Automobiles, UC San Diego, United States `mtrivedi@ucsd.edu`
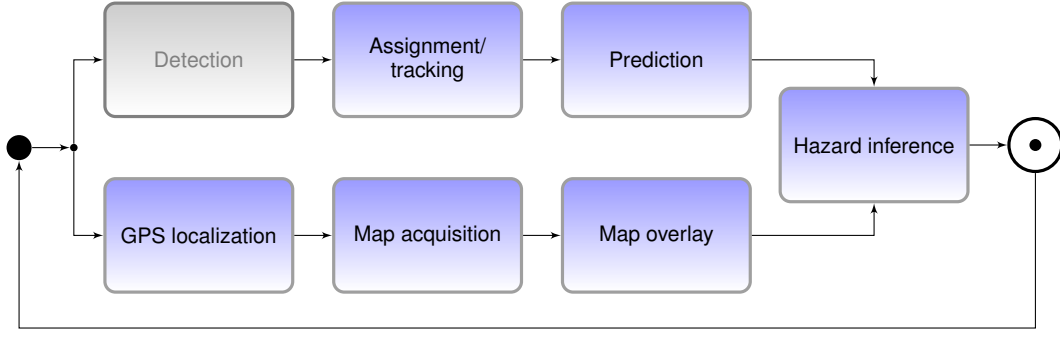
Fig. 1. System flow. Note that *Detection* is grayed out, since it is outside of the scope of this work.

Tracking of moving pedestrians is done from a surveillance perspective in [23], [24], and from a car perspective on stereo data in [25], [26] using Interacting Multiple Model Kalman Filters and SLDS tracking, respectively. Finally, long term path prediction from a stationary camera is done in [27].

The work presented in this paper also belongs to the class of tracking-based prediction systems. Its main differences to the state of the art are:

1) The system works on a monocular camera from a car perspective, where most others use RGB-D data.
2) Maps are integrated in the system and used to infer hazardous areas.
3) The analyzed scenes are complex, natural, and unconstrained with real pedestrians.
4) Particle filters are used for tracking.

III. SYSTEM OVERVIEW

The structure of the system presented here is shown in fig. 1. Two parallel processes run. In the upper row: Pedestrian bounding boxes are supplied from some kind of detector. Detection itself is outside the scope of this work, and throughout the project, hand-annotated bounding boxes have been used in place of a detector. Each detectionis assigned to a track. A new track is initiated if no existing track fits. Using the dynamics captured by the particle filter tracker, the pedestrian's position can be predicted into the near future. The lower row shows the mapping part. The position and orientation of the ego-vehicle is determined with a GPS and an electronic compass. A corresponding map is retrieved from OpenStreetMap and rotated to fit. A top view of the car's surroundings is generated via Inverse Perspective Mapping (IPM), and the tracks and street map are superimposed to this. By using pre-acquired mapping, we do not need to rely on road segmentation in the input images. Instead we know where the car will drive in the future.

In hazard inference, the projected position of any pedestrian is compared to the road position, and if the pedestrian enters with sufficient certainty, the driver can be warned. The UI for warning the driver is not covered in this paper.

IV. TRAJECTORY GENERATION AND TRACKING

The trajectory generation consists of two tasks: Assignment and tracking. In assignment all detections are assigned

Fig. 2. Example of assignment between pedestrians. Two boxes are shown per person: the current bounding box and the previous bounding box from the pedestrian's track. Some boxes are very close to each other, and thus hard to distinguish on the picture.

to an existing track, or a new track is created for them. In tracking, each track is updated. Assignment takes places in the native camera coordinate system, whereas tracking is done on the top-down map of the vehicle generated using IPM. Fig. 6 shows both views. The input image is a 1280x960 RGB image captured with a networked PointGrey camera.

A. Assignment

Assignment is done with the Munkres algorithm between bounding boxes in the current input image and the previous bounding box for each track. A cost matrix is populated with the cost for associating a bounding box with any given previous bounding box. The cost is the Euclidean distance between the box centers plus the size change of the box (a bounding box is expected to be roughly the same size in two consecutive frames). Since boxes move and change size in bigger increments when pedestrians are close to the camera, the cost is weighted by the inverse of the box size, so large costs are lowered when the bounding box is large:

$$D(a,b) = \left(\sqrt{|a_x - b_x|^2 + |a_y - b_y|^2} + \sqrt{|a_w - b_w|^2 + |a_h - b_h|^2} \right) \cdot \frac{1}{a_w + a_h} \quad (1)$$

where $D(a,b)$ is the distance between boxes a and b, and subscripts x, y, w, h means center x-coordinate, center y-coordinate, width, and height, respectively.

n bounding boxes gives an $n \times n$ matrix. The Munkres algorithm allows for unequal numbers of input and output
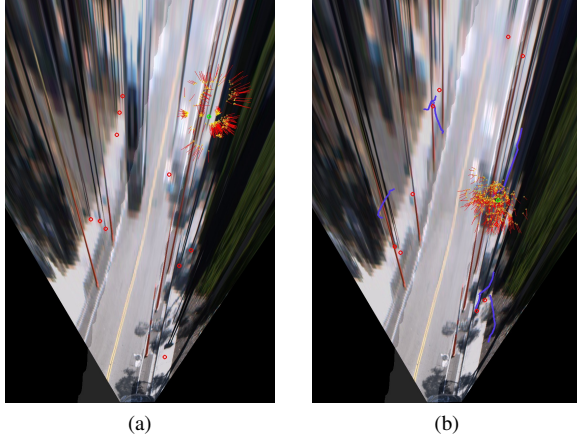
Fig. 3. All 1000 particles for a single track (the green observation) plotted with position and orientation. (a) shows the distribution before convergence and (b) shows the distribution after.
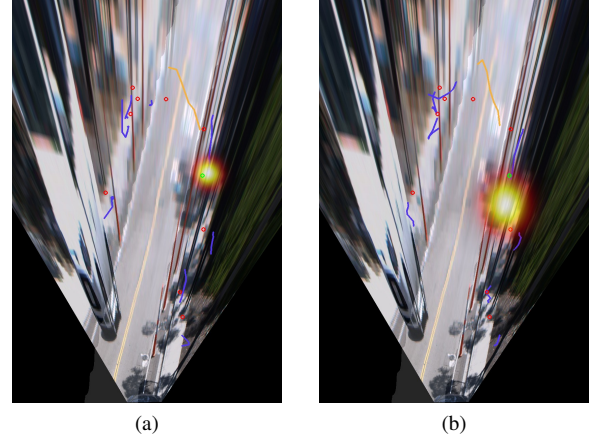

Fig. 4. Example position predictions of the green observation (a) one time step ahead and (b) five steps ahead. The one-step prediction is behind the actual observation because of measurement-noise, which makes the observation unreliable and the filter smooths the movement out. This is exacerbated by the inverse perspective mapping, which is very sensitive to even small changes in bounding box position, especially at a distance. As the prediction is done further into the future, the uncertainty rises, which can be seen by the expanding heatmap.
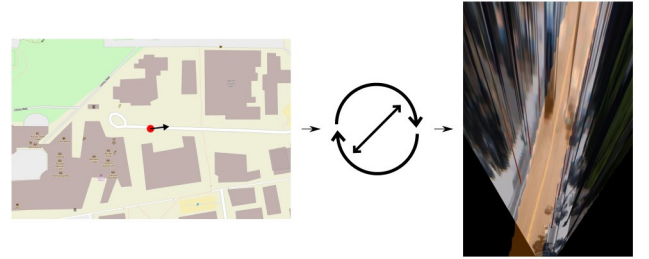

Fig. 5. The road in front of the car is extracted from OpenStreetMap. It is then rotated and scaled appropriately and overlaid on the IPM-generated map.

by padding the cost matrix with "infinity" until it is square. This, however, finds a global optimum, and will often lead to all boxes shifting. Imagine a case where one track ends, and another begins simultaneously. There will still be an equal number of boxes on the in- and output side, so all boxes will be reassigned, when in reality one box should have been assigned to nothing and one should have prompted a new track. To accommodate these scenarios, we add another n columns of close-costs. The close-cost is simply a threshold over which we decide that it is better to close a track than to reassign it. As a result, in the $n \times n$ case, the cost matrix will now be $n \times 2n$, with the left half containing reassignment costs, and the entire right half containing indentical close-costs in all entries. After the assignment in done, any unassigned detection will be assigned a new track. An assignment example is shown in fig. 2.

B. Tracking

In this system tracking is done by a particle filter for each track. Each track has 1000 particles that are modeled using the unicycle model: a particle has a certain speed and orientation. Each update is done by applying a certain amount of Gaussian noise to each of these parameters. Each measurement is also applied Gaussian measurement noise. The particles are weighted by distance to the measurement using a bivariate Gaussian:

$$P(p, m) = \exp \left(\frac{(p_x - m_x)^2}{2 \cdot v_x} - \frac{(p_y - m_y)^2}{2 \cdot v_y} \right) \quad (2)$$

where $P(p, m)$ is the probability of a particle p given the measurement m and v is the variance in two dimensions. Example pictures of the particle filter tracking are shown in fig. 3

V. BEHAVIOR PREDICTION AND HAZARD INFERENCE

The prediction is carried out by the particle filter. When the prediction has been computed for the desired prediction horizon, a bivariate Gaussian is fitted over the particles.

Particle filters support multimodal hypotheses, but they are not relevant with a relatively direct measurement setup as in this system. Thus fitting a single bivariate Gaussian does not lead to a significant loss of information. The fitted Gaussian is then used to describe the probability of positions the pedestrian might be at in the near future. Fig. 4a visualizes the probability map one step ahead and fig. 4b shows the prediction 5 steps ahead.

A central step in the hazard inference process is using a map to determine where the road is. A map is obtained from OpenStreetMap, which can be installed on a server locally in the car. Then the map is rotated according to the car's orientation and scaled appropriately. Fig. 5 illustrates this process.

To estimate whether a pedestrian is about to enter a hazardous zone, a combination of the predicted position – as visualized by the heatmaps in fig. 4 – and the known road area is used. Each pixel of the heatmap has a value, depending on the probability of the pedestrian being there. The values of the pixels overlapping with the road are summed, and if the sum is above some threshold, a warning
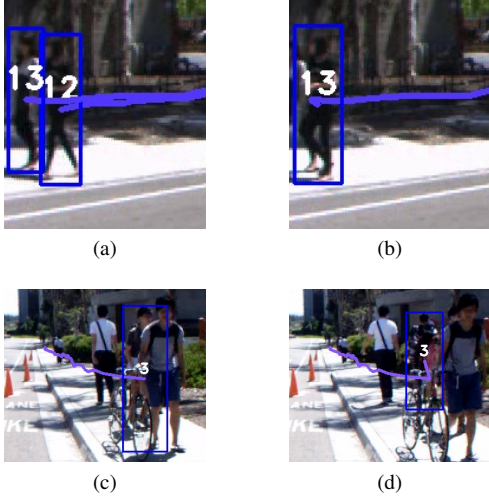
Fig. 7. Two of the just four wrong assignments in the 1022 detections of the test sequence. Each row is an example with the before and after frame shown. See text for further discussion.

Fig. 8. Example of how the tracking (orange lines) is smoothing the original input (blue lines).

is emitted:

$$W(p) = \sum^{(x,y)\in H} H(x,y) \cdot R(x,y) \qquad (3)$$

where $W(p)$ is the warning-value (subject to a threshold) of particle p, H is the position probability map of p and R is the mask of the road, expressed as 1 where road is present and 0 otherwise.

VI. EVALUATION AND DISCUSSION

Figure 6 shows output examples of the full system. The left side show the native view with detection bounding boxes, while the right side shows the map view with circles for pedestrians. Heatmaps on the map view show the predicted position of the pedestrians, and the color of both bounding boxes and circles show whether the pedestrian is about to enter a hazardous area.

Assignment: The system consists of several blocks, and in this section we discuss the performance of each. The assignment part performs very well. On a test sequence with 1022 individual detections over 106 frames, only 4 wrong assignments happen. Two of these instances are shown in fig. 7. In the top row, the person of track 13 is exiting the frame. In 7b, after 13 having exited, the person previously assigned to track 12 steps to the exact position that 13 had before, and the algorithm determines that the lowest global cost is achieved by closing track 12. This issue might be solved by having a stronger prediction input to the assignment.

In 7c and 7d, the previous track 3 is occluded by a person in front of him at the exact same time as a pedestrian which was previously hidden emerges in the same line-of-sight of the camera.

Tracking: Tracking has two jobs. Smooth out the trajectory of the tracked pedestrians and allow for prediction of their position. The smoothing is especially necessary in a monocular setup, such as ours, where the distance to pedestrians is detemined via IPM.If a bounding box position differs by even
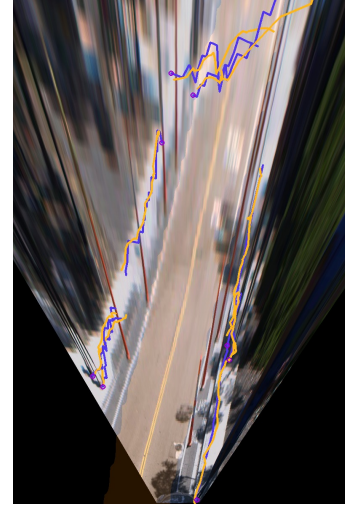
just a pixel in the y-axis, the estimated position on the map can jump several meters. The tracking should compensate for this effect.Setting up exact metrics for this is difficult at best, when the ground truth position of the pedestrians is not known.

An example of this is shown in fig. 8. Here, the original input is shown in blue, and the particle filter output in orange. The smoothing can be adjusted via the system noise until a satisfying combination of smoothness and reaction time is reached. From a visual inspection it is clear that the orange tracks provide a much better approximation to the real world than the very jagged input data, even at relatively long distances. This is especially clear in the tracks to the left.

Hazard inference: Hazard inference which combines the predictive power of the tracking with the map based road localization. Examples are shown in fig. 9. 9a shows a successful - but easy - prediction, where a crossing cyclist is in the middle of the road and is not predicted to leave the road in the next 2 seconds. 9b shows the situation a few frames later, as the cyclist is just about to leave the road, and is thus predicted to be out of danger soon. 9c shows another use for the system. Here, there is no prediction of impending hazards, but there is clearly a large concentration of pedestrian activity in the lower right corner. This knowledge in itself might be useful in a driver assistance context. Finally, 9d shows a faulty prediction. In this case a pedestrian steps onto the bike lane to overtake slower pedestrians on the sidewalk. The system predicts her to enter the roadway, but in fact she never does. Technically, this makes it a false prediction, but it could be argued that it is beneficial to warn the driver about this still, since she is in very close proximity to the road.

VII. CONCLUDING REMARKS

This work presented a pedestrian intent prediction system for use in driver assistance. It uses monocular a monocular
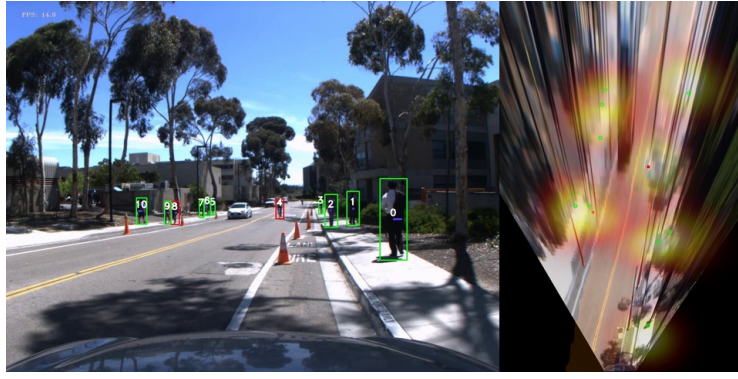
Fig. 6. Hazard warnings at different times. Green boxes/circles are pedestrians who are not about to enter the road. Red indicates pedestrians who either are on the road already, or are about to enter it. (a) is very early in the sequence, so the tracks are still uncertain, and the heatmaps are large. The remaining examples are taken later, when the tracks are more reliable. (c) has a faulty warning at track 12, due to a very short track with great uncertainty (the previous tracks were cut when a car passed). It also has a missing warning all the way in the background at track 14. This is due to that pedestrian being so far away that he is outside of the map, and thus not considered for the risk analysis.

view of the road, in which detected pedestrians are mapped to a top view computed using Inverse Perspective Mapping. For tracking, pedestrians are assigned to a track based on their bounding box in the native camera view, and tracking takes place in the map view using particle filters. To determine which areas in view are potential hazardous zones for pedestrian, external mapping is used. A map of the nearby area is acquired from OpenStreetMap and superimposed onto the IPM view. Using the knowledge of road positions in the map, hazardous areas in the camera view are obtained. Based on the overlap of the trackers' predictions for all observed

pedestrians with the road area, the driver can be informed about wayward pedestrians.

In the future, ego-motion compensation should be added to the system, so it works for moving vehicles. Furthermore, the orientation of the pedestrians – based on appearance, not dynamics – should be included in the tracking measurements, since pedestrians are capable of very rapid orientation changes, which are hard to capture in a purely dynamics-based system such as this. It is also possible that local motion cues from e.g. the pedestrians' legs can be used in improving performance.
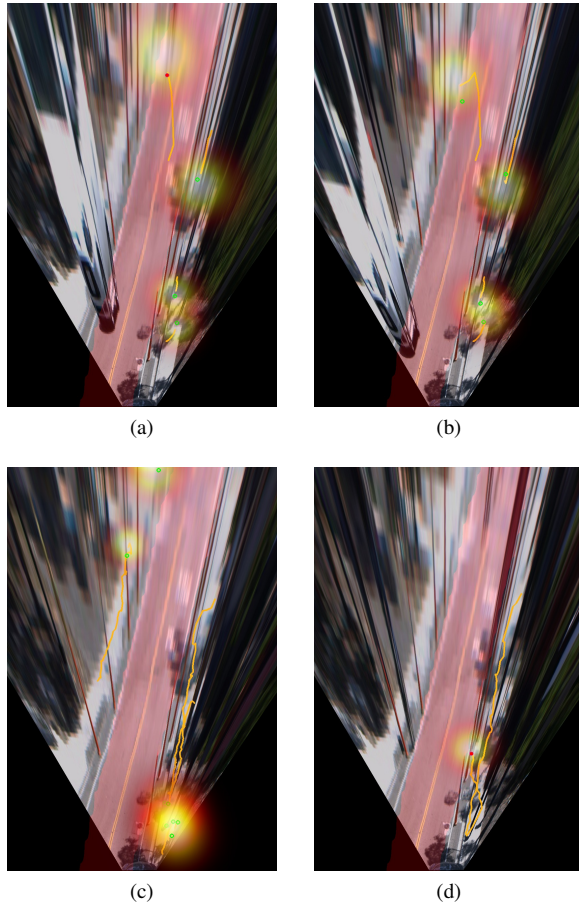
Fig. 9. Hazard inference in 4 situations. The hazardous area - the road - is marked with a translucent red color. For clarity, only a few tracks are shown in each. (a) and (b) are just a few frames apart, and the hazard indicator for the subject crossing the road goes from red to green when he is predicted to leave the road. (c) shows an example of heavy pedestrian activity, and (d) shows a mistaken prediction.

REFERENCES

[1] J. Cinnamon, N. Schuurman, and S. M. Hameed, "Pedestrian injury and human behaviour: observing road-rule violations at high-incident intersections," *PloS one*, vol. 6, no. 6, p. e21063, 2011.

[2] T. Gandhi and M. M. Trivedi, "Pedestrian protection systems: Issues, survey, and challenges," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 8, no. 3, pp. 413–430, 2007.

[3] A. Tawari, A. Møgelmose, S. Martin, T. B. Moeslund, and M. M. Trivedi, "Attention Estimation by Simultaneous Analysis of Viewer and View," in *Intelligent Transportation Systems (ITSC), 2014 17th International IEEE Conference on*, Oct 2014.

[4] A. Doshi and M. Trivedi, "Tactical Driver Behavior Prediction and Intent Inference: A Review," in *14th IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2011.

[5] A. Doshi, B. Morris, and M. M. Trivedi, "On-road prediction of driver's intent with multimodal sensory cues," *IEEE Pervasive Computing*, vol. 10, no. 3, pp. 22–34, 2011.

[6] E. Ohn-Bar, A. Tawari, S. Martin, and M. M. Trivedi, "On surveillance for safety critical events: In-vehicle video networks for predictive driver assistance systems," *Computer Vision and Image Understanding*, vol. 134, no. 0, pp. 130–140, 2015, image Understanding for Real-world Distributed Video Networks.

[7] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pp. I–511–I–518 vol.1, 2001.

[8] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," in *CVPR*, 2005.

[9] A. Prioletti, A. Mogelmose, P. Grisleri, M. M. Trivedi, A. Broggi, and T. B. Moeslund, "Part-Based Pedestrian Detection and Feature-Based Tracking for Driver Assistance: Real-Time, Robust Algorithms, and Evaluation." *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1346–1359, 2013.

[10] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, "Object Detection with Discriminatively Trained Part-Based Models." *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, 2010.

[11] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral Channel Features," in *BMVC*, vol. 2, no. 3, 2009, p. 5.

[12] P. Dollár, S. Belongie, and P. Perona, "The Fastest Pedestrian Detector in the West," in *BMVC*, vol. 2, no. 3. Citeseer, 2010, p. 7.

[13] P. Dollar, R. Appel, S. Belongie, and P. Perona, "Fast Feature Pyramids for Object Detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 8, pp. 1532–1545, Aug 2014.

[14] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian Detection: An Evaluation of the State of the Art," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 743–761, 2012.

[15] M. Enzweiler and D. Gavrila, "Integrated pedestrian classification and orientation estimation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, June 2010, pp. 982–989.

[16] T. Gandhi and M. Trivedi, "Image based estimation of pedestrian orientation for improving path prediction," in *Intelligent Vehicles Symposium, 2008 IEEE*, June 2008, pp. 506–511.

[17] J. Tao and R. Klette, "Integrated Pedestrian and Direction Classification Using a Random Decision Forest," in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, December 2013.

[18] M. C. Liem and D. M. Gavrila, "Person appearance modeling and orientation estimation using Spherical Harmonics," *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 0, pp. 1–6, 2013.

[19] W. Liu, Y. Zhang, S. Tang, J. Tang, R. Hong, and J. Li, "Accurate Estimation of Human Body Orientation From RGB-D Sensors," *Cybernetics, IEEE Transactions on*, vol. 43, no. 5, pp. 1442–1452, Oct 2013.

[20] F. Flohr, M. Dumitru-Guzu, J. F. P. Kooij, and D. M. Gavrila, "Joint probabilistic pedestrian head and body orientation estimation," in *Intelligent Vehicles Symposium (IV), 2014 IEEE*, june 2014, pp. 617–622.

[21] S. Köhler, M. Goldhammer, S. Bauer, S. Zecha, K. Doll, U. Brunsmann, and K. Dietmayer, "Stationary Detection of the Pedestrian's Intention at Intersections," *Intelligent Transportation Systems Magazine, IEEE*, vol. 5, no. 4, pp. 87–99, winter 2013.

[22] C. Keller and D. Gavrila, "Will the Pedestrian Cross? A Study on Pedestrian Path Prediction," *Intelligent Transportation Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–13, 2013.

[23] B. T. Morris and M. M. Trivedi, "Learning and classification of trajectories in dynamic scenes: A general framework for live video analysis," in *Advanced Video and Signal Based Surveillance, 2008. AVSS'08. IEEE Fifth International Conference on*. IEEE, 2008, pp. 154–161.

[24] A. Bera, N. Galoppo, D. Sharlet, A. Lake, and D. Manocha, "Adapt: real-time adaptive pedestrian tracking for crowded scenes," in *Proceedings of Conference on Robotics and Automation, Hong Kong*, 2014.

[25] N. Schneider and D. M. Gavrila, "Pedestrian Path Prediction with Recursive Bayesian Filters: A Comparative Study," in *Pattern Recognition*, ser. Lecture Notes in Computer Science, J. Weickert, M. Hein, and B. Schiele, Eds. Springer Berlin Heidelberg, 2013, vol. 8142, pp. 174–183.

[26] J. F. P. Kooij, N. Schneider, and D. M. Gavrila, "Analysis of pedestrian dynamics from a vehicle perspective," in *Intelligent Vehicles Symposium (IV), 2014 IEEE*, june 2014, pp. 1445–1450.

[27] K. M. Kitani, B. D. Ziebart, J. A. Bagnell, and M. Hebert, "Activity forecasting," in *Computer Vision–ECCV 2012*. Springer, 2012, pp. 201–214.