Improving Object Tracking by Adapting Detectors

Lu Zhang Laurens van der Maaten
Vision Lab, Delft University of Technology, The Netherlands
Email: {lu.zhang, l.j.p.vandermaaten}@tudelft.nl

***Abstract*—The goal of model-based object trackers is to automatically detect and track specific objects, such as cars or pedestrians. To solve this problem, many modern trackers train a detector on a collection of annotated object images and use the trained detector in a tracking-by-detection framework. A major limitation of such an approach is that a single, generic detector is used to track specific objects; the additional information on the visual appearance of the particular object under consideration that is available after the initial detection is ignored. This paper proposes an approach that addresses this limitation by adapting the appearance model for each particular object using online learning techniques. We demonstrate the effectiveness of the approach in a state-of-the-art object detector based on deformable template models, the parameters of which are adapted online using an online structured SVM. We further improve the performance of the resulting model-based trackers by online learning a prior distribution over the size of objects. The experimental evaluation of our tracker demonstrates its effectiveness in pedestrian tracking.**

I. Introduction

Model-based object tracking is a seminal problem in computer vision with a wide range of applications. For instance, in applications such as automatic surveillance and driving assistance, it is essential to detect and track objects like faces, pedestrians and cars. Most modern model-based trackers train a detector on a collection of annotated object images (*e.g.*, faces [1], pedestrians [2], [3], [4], [5], cars [6], and rigid objects [7]) and use the trained detector in a tracking-by-detection framework. The advantage of these trackers is that they are very robust since their object appearance model learned to cope with large variances of data (*e.g.*, with different view-point and partial occlusion). However, a major limitation of existing tracking-by-detection approaches is that a single, generic detector is used to track specific objects, whereas additional information on the visual appearance of the specific object under consideration is ignored.

In this paper, we present an online-learning approach that adapts the appearance model of the detector to the specific object that is tracked. Our approach combines on recent advances from model-free tracking [8], [9], [10], [11], [12], [13] with state-of-the-art object detection frameworks (*e.g.*, [6]). Specifically, we develop an online-learning algorithm that addresses the limitations of model-based trackers by updating the parameters of a deformable part model in an online structured SVM framework. The online updates adapt the appearance model to more accurately describe the target object under consideration. We coin the resulting tracker the online deformable-part-based model (ODPM) tracker. We further improve the performance of the ODPM tracker by online learning a prior distribution over the size of objects.

This paper proposes an online deformable-part-based model (ODPM) that addresses the limitation of model-based trackers by adapting the appearance model for each particular object using online learning techniques. We demonstrate the effectiveness of the approach in a state-of-the-art object detector based on deformable-part-based models (DPM) [6], the parameters of which are adapted online using an online structured SVM [13]. We further improve the performance of the resulting model-based trackers by online learning a prior distribution over the locations and the size of objects, thereby substantially reducing the number of false positives that our detector generates [14]. We demonstrate the effectiveness of our tracker empirically in a pedestrian-tracking task [15].

II. Related Work

The work described in this paper combines ideas from object detection and model-free tracking. We briefly introduce related work in both these areas below.

Despite the recent popularity of convolutional networks for object detection [16], [17], [18], [19], many popular detectors still rely on hand-crafted gradient features such as histograms of oriented gradients (HOG) features [20]. For instance, many deformable part models [6] and grammar models [21] generally rely on a combination of linear models and HOG features. The most prominent approach in this area [6] uses a star-structured part-based model defined by a "root" filter (analogous to the Dalal-Triggs filter) plus a set of part filters and deformation models. The score of one of star models at a particular position and scale within an image is the score of the root filter at the given location plus the sum over parts of the maximum, over placements of that part, of the part filter score at its location minus a deformation cost measuring the deviation of the part from its ideal location relative to the root. Both root and part filter scores are defined by the dot product between a filter (a set of weights) and a subwindow of a feature pyramid computed from the input image. Similar deformable part models have been used, among others, in pedestrian detection and tracking [22], [23] and in human body pose estimation [24], [25], [26].

In model-free tracking, most recent approaches employ an online-learning approach in order to learn better models of the visual appearance of the target object and to adapt that model to appearance changes over time. In general, these approaches assume that the track in a frame can be used as a positive training example for the appearance models. Many prior studies in model-free tracking focus on exploring different feature representations for the target object, including feature representations based on points [27], [28], [29], contours [30], [31], [32], integral histograms [33], subspace learning [34], sparse representations [35], and local binary patterns [12]. Recent work also focuses on developing new learning approaches

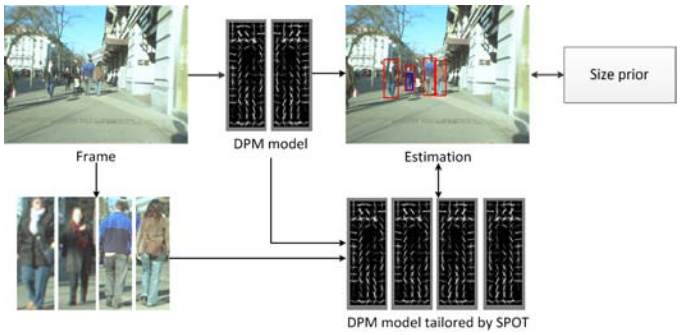

Fig. 1. Flowchart of the ODPM algorithm.

to better distinguish the target object from the background. In particular, previous studies have investigated approaches based on boosting [9], [36], random forests [12], multiple instance learning [8], and structured output learning to predict object transformations [10]. Recent work also focuses on learning motion priors to obtain better trajectory association; for instance, [15] learns a context-conditioned motion prior on sport players (such as basketball and hockey players).

Some recent studies propose to perform model-free tracking using deformable part models that are trained online [13]. The appearance model is similar to that of [6] in this work, but the difference is that its parameters are trained in an online structured SVM framework [37]. Our work is motivated by the success of [13], which paves the way for adapting state-of-the-art object detectors [6] to the specific objects that are being tracked via online learning.

III. Methodology

The basis of our model-based tracker is formed by a deformable part model (DPM) that employs histograms-of-oriented gradient (HOG) features and that is trained using a (latent) structured SVM [6]. We adapt the parameters of this DPM to the particular object under consideration via online learning. A general overview of the resulting tracker is shown in Fig. 1. Each element of the tracker described in detail in the following subsections.

A. Deformable Part-Based Model

Deformable part-based models (DPMs) consist of two main submodels: (1) a model for the visual appearance of each part and (2) a geometric model that captures spatial relationships between the parts. The parameters of DPMs are generally learned using maximum likelihood estimation. The basis of our tracker is the DPM introduced in [6], which models the visual appearance of the parts using linear models trained on HOG features. Specifically, the model employs a global root "part" that models the entire object using coarse HOG features, and a number of smaller parts for which HOG features are measured on a finer scale. The geometric model comprises a star-structure model that represents the spatial relationships between each part and the root part. As the visual appearance of objects might be different depending on the viewpoint, an object category is represented by a mixture of DPMs, *e.g.*, a typical pedestrian detector is a mixture of two DPMs (one for a left side view and one for a right side view).

In the DPM, each part $i \in V$ (with V representing the set of all parts) is indicated by a box $B_i = \{\mathbf{x}_i, w_i, h_i\}$ with center location $\mathbf{x}_i = (x_i, y_i)$, width w_i, and height h_i. We denote the HOG features extracted from a box B_i in image $\mathbf{I}$ by $\phi(\mathbf{I}; B_i)$. Subsequently, we define a star graph $G = (V, E)$ over all parts and root $i \in V$ with edges $j \in E$ between the parts and root part. The edges in the graph can be viewed as springs that represent spatial constraints between the parts and the root part. The *score* of a *configuration* $C = \{B_1, \ldots, B_{|V|}\}$ of each model as the sum of two terms: (1) an appearance score that sums the similarities between the observed image features and the classifier weights for all parts, and (2) a deformation penalty that measures how much a configuration compresses or stretches the springs between the parts and root. Specifically, the score of a configuration C is defined as:

$$s(C; \mathbf{I}, \Theta) = \sum_{i \in V} \mathbf{w}_i^{\mathrm{T}} \phi\left(\mathbf{I}; B_i\right) - \sum_{j \in E} \lambda_j \phi_d\left(dx_j, dy_j\right), \quad (1)$$

where the deformation cost $\phi_d\left(dx_j, dy_j\right)$ for the j-th part is defined to be the squared distance between its actual position and its anchor position relative to the root, the parameters $\mathbf{w}_i$ represent linear weights on the HOG features for part i, λ_j is the tradeoff parameter between appearance and deformation, and the set of all parameters is denoted by $\Theta = \{\mathbf{w}_1, \ldots, \mathbf{w}_{|V|}, \lambda_1, \ldots, \lambda_{|E|}\}$. The parameters Θ is learned using a latent structured support vector machine, as described in [6].

Given the parameters of the model, finding the most likely object configuration (for a single component of the mixture model) amounts to maximizing Eqn. (1) over all possible configurations C. This maximization is intractable in general because it requires searching over exponentially many configurations, but for star-structured graphs G, a combination of dynamic programming and min-convolutions can be used to perform the maximization in linear time [6]. The final most likely object configuration is to maximize the score over all mixture component models M, since each component of the model is conditionally independent (given the mixture component assignment).

B. Online adaptation

The main contribution of this paper is an algorithm that online updates the parameters of the DPM in order to better model the visual appearance of the object that is being tracked. After observing an image $\mathbf{I}$, we detect or track the object under consideration by finding the most likely object configuration C^* by maximizing Eqn. (1). The key difference between detection and tracking is that for tracking, we incorporate a Gaussian location prior centered around the previous location of the target object. (In both detection and tracking, we also use additional size priors that are learned online — further details of these priors are given in the next subsection.)

Subsequently, we assume that we have correctly localized the target object. This implies that we can use the optimal object configuration C^* as a positive example in online learning. As before, we employ a structured SVM formulation in order to perform the parameter update. In other words, we

would like to update the parameters of the DPM in such a way that the score $s(C^*; \mathbf{I}, \Theta)$ for the optimal configuration C^* is larger than any other configuration $\hat{C}$ by at least a margin $\Delta(C, \hat{C})$. Herein, Δ is a task-loss function that equals zero iff $\hat{C} = C$. The task loss is assumed to be non-negative, $\forall \hat{C} \neq C : \Delta(C, \hat{C}) > 0$, and upper bounded by a constant Q, $\exists Q \forall C : \max_{\hat{C}} \Delta(C, \hat{C}) < Q$. To adapt our model in such a way that it assigns a higher score to the positive locations and lower scores to other locations, we update the model parameters Θ by taking a gradient step in the direction of the structured SVM loss function [38]. Herein, the structured SVM loss ℓ is defined as the maximum violation of the task loss by configuration $\hat{C}$ (as described above):

$$\ell(\Theta; \mathbf{I}, C) = \max_{\hat{C}} \left[s(\hat{C}; \mathbf{I}, \Theta) - s(C; \mathbf{I}, \Theta) + \Delta(C, \hat{C}) \right]. \quad (2)$$

The structured SVM loss function does not contain quadratic terms, but it is the maximum of a set of affine functions. As a result, the structured SVM loss in Eqn. (2) is a convex function in the parameters Θ.

We assume that the graph structure for the target object does not change much over time, which is why we only adapt the appearance parameters $\mathbf{w}_1, \ldots, \mathbf{w}_{|V|}$ of the DPM. Specifically, we use a passive-aggressive update of these parameters [39]. The passive-aggressive algorithm sets the step size in such a way as to substantially decrease the loss, while ensuring that the parameter update is not too large. In particular, it uses the following parameter update:

$$\mathbf{w} \leftarrow \mathbf{w} - \frac{\ell(\Theta; \mathbf{I}, C)}{\|\nabla_{\mathbf{w}} \ell(\Theta; \mathbf{I}, C)\|^2 + \frac{1}{2K}} \nabla_{\mathbf{w}} \ell(\Theta; \mathbf{I}, C), \quad (3)$$

where $K \in (0, +\infty)$ is a hyperparameter that controls the "aggressiveness" of the parameter update. In our experiment, we choose $K = 1$. Since each object detector may consists of several component models, we only update the component model with highest score each time. By using such a simple online learning algorithm, the appearance of the whole person and more detailed sub-parts of person is learned during tracking. The updated model for a certain person will be more discriminative and more adaptive to this person's visual appearance.

We build and update an separate model for each detection by tailoring the corresponding DPMs as described above. To prevent our tracker from drifting, we use some checks that aim to confirm that every detection is correct. Specifically, we only perform a parameter update if the detections of both the base detector and the adapted detector agree on a particular bounding box. Herein, we assume that both detectors agree when the overlap of the two bounding boxes they produce is larger than 50%. Moreover, we stop tracking an object when the base detector cannot detect the target object anymore for more than five frames.

C. Size Prior

Whilst the online learning of our DPMs reduces the number of errors during tracking, it does not eliminate false positives in the initial detection of target objects. Since our approach may adapt the DPM parameters to better model such a false positive, it is essential to remove false positive detections as soon as possible. Generally, false positive detections appear at locations which have a similar visual appearance as the target object. However, the size of false positive detections is often not consistent with the size of other target objects in nearby locations. Moreover, false positives also occur quite often at locations that are unlikely *a* prior. For instance, a bottle on the table may sometimes be recognized as a pedestrian. We propose a simple approach that eliminates false positives by noting that the bottle has a unlikely size and location when considering its environment. Specifically, we learn a prior distribution over the sizes of target objects in online manner.

We initialize the prior to be a uniform distribution over sizes at all locations, so we accept all detections in the first frames of a video. Because the size of objects generally increases when objects are closer to the camera, we opt to not make the prior distribution over sizes dependent on the x-coordinate of the object location, but only on the y-location. This assumes that objects that are located on the same horizontal line generally do have a similar size. We use a Gaussian distribution $p(s|y, \mu_y, \sigma_y^2)$ to model the prior over object size s for each y-location. The variance σ_y^2 of $p(s)$ is set to a predefined fixed value. The mean μ_y is initially set according to the average detection size of the same horizontal line as observed in the first t frames, and it is updated with new detections overtime using online averaging. Since we do not generally have observed sufficient detections at every y-location, we use linear interpolation based on the available detections to set all values of μ_y.

IV. Experiment

We perform experiments with our online DPM tracker in which we compare its performance with that of a DPM that is not updated during tracking. In both cases, the initial detection is made using a Felzenszwalb object detector [6] that was trained on the pedestrian class of the Pascal VOC 2007 challenge. The ODPM tracker is initialized using the parameters of that off-the-shelf detector; after this initialization, the ODPM parameters are updated using the procedure described above. Following [40], we measure two characteristics of our tracker: (1) the *miss rate* that is defined as the rate of objects that are missed by the tracker, where we define a miss when there is no detection that has more than 50% overlap with a ground-truth bounding box; and (2) the *false positive rate*, where a false positive is an detection that has less than 50% overlap with a ground-truth bounding box. To compute these two measures, we solve a simple assignment problem between the ground-truth annotations and the detections by the (O)DPM tracker.

To evaluate the performance of the ODPM track, we run experiments on five videos from ETH pedestrian database [41]: *Sunny day*, *Bahnhof*, *Jelmoli*, *Crossing*, and *Linthescher*. The results of these experiment are shown in Fig. 2. Specifically, the figure presents the miss rate as a function of the false positive rate DPM and ODPM on all five videos (lower curves indicate better performance). The results presented in the figure show that ODPM outperforms the baseline DPM detector on average, which supports our hypothesis that tailoring the appearance model of an object detector to a particular object of interest may indeed improve tracker performance. For videos like *Sunny day* and *Crossing*, there are less pedestrian-like patches in the background and most pedestrians are big enough

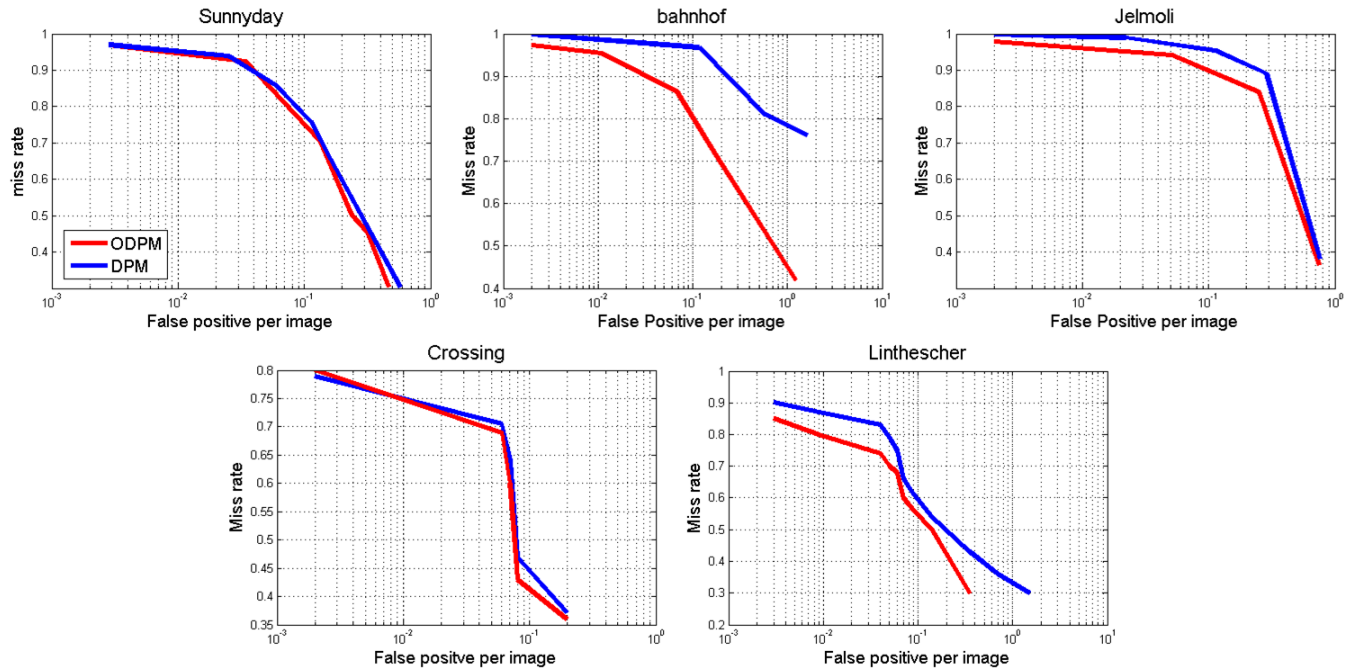

Fig. 2. Miss rate as a function of the false positive rate of DPM and ODPM on all five videos (lower curves indicate better performance). Figure best viewed in color.

to detect for almost every frame, as a result of which we may be observing a ceiling effect: the baseline DPM tracker already has a very good performance, as a result of which the improvement of the ODPM tracker on these videos is insignificant. By contrast, on challenging videos like *Bahnhof* and *Linthescher*, in which the size and appearance variation of pedestrians are very large (due to occlusion, cluttered background, *etc.*), the baseline DPM tracker produces a substantial amount of false positive detections and false negatives. On these challenging videos, the ODPM tracker substantially outperforms the baseline because it adapts the appearance models to individual pedestrians, as a result of which it handles low illumination and partial occlusion problems better. In addition, the ODPM tracker benefits from the size prior that we learn online, which facilitates the removal of false positives with erroneous sizes.

Fig. 3 shows some of the tracking results of ODPM (in red) and DPM (in blue) on the *Sunny day*, *Bahnhof*, *Jelmoli*, *Crossing*, and *Linthescher* videos. Please note that in some visualizations, some blue rectangles from the images are partially covered by red rectangles due to the order of plotting the rectangles. The figure illustrates the strong performance of the ODPM tracker, in particular, its ability to remove false positives based on the size prior (such as detections in the sky or that are too big). The figure also illustrates why ODPM tracking does not improve over DPM tracking for simple videos like *Sunny day*: DPM already does a good job on these videos. A disadvantage of the proposed ODPM is that we need to compute the detection score of all the object-specific models in every scale and location of the HOG pyramid, which is time-consuming. To obtain an efficient tracker nonetheless, one may use ODPM in combination with an efficient searching method such as selective search [42], feature ignoring tracking [43], or cascaded models based on Bing features [44]. When such approaches are employed, we expect that the ODPM tracker can still operate in real-time.

V. Conclusion

We have proposed an online adaptation mechanism for deformable part-based models in order to improve their performance when used for tracking. Specifically, our algorithm tailors the generic object appearance model of off-the-shelf object detectors to specific instances of that object. We have shown empirically that tracking by tailoring object detectors online decrease the miss rate of these detectors, in particular, when we learn object size priors in an online manner to reduce the number of false positive detections. In future work, we aim to investigate specific model-based object tracking problems in domains such as activity recognition and consumer science.

Acknowledgment

This work was supported by the EU-AAL / ZonMW project SALIG++, and by the China Scholarship Council (CSC). The authors thank Pedro F. Felzenszwalb for publicly sharing his implementation of deformable part-based models.

References

[1] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, pp. 511–518.

[2] M. Isard and J. Maccormick, "Bramble: A Bayesian multi-blob tracker," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2001, pp. 34–41.

[3] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1820–1833, 2011.

[4] S. Pellegrini, A. Ess, K. Schindler, and L. van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *Proc. IEEE Intl Conf. Computer Vision*, 2009, pp. 261–268.

Fig. 3. Pedestrian tracks obtained using proposed ODPM (in red) and DPM (in blue) on the *Sunny day*, *Bahnhof*, *Jelmoli*, *Crossing*, and *Linthescher* videos from the ETH pedestrian data set. Figure best viewed in color.

[5] Z. Wu, A. Thangali, S. Sclaroff, and M. Betke, "Coupling detection and data association for multiple object tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012, pp. 1948–1955.

[6] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part based models," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[7] V. Lepetit and P. Fua, "Keypoint recognition using randomized trees," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1456–1479, 2006.

[8] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE TPAMI*, vol. 33, no. 8, pp. 1619–1632, 2011.

[9] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *BMVC*, 2006, pp. 47–56.

[10] S. Hare, A. Saffari, and P. H. Torr, "Struck: Structured output tracking with kernels," in *Proc. IEEE Conf. ICCV*, 2011, pp. 263–270.

[11] T. B. Dinh, N. Vo, and G. Medioni, "Context tracker: Exploring supporters and distracters in unconstrained environments," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011, pp. 1177–1184.

[12] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-n learning: Bootstrapping binary classifiers by structural constraints," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2010, pp. 49–56.

[13] L. Zhang and L. van der Maaten, "Structure preserving object tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013.

[14] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool, "Pedestrian detection at 100 frames per second," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2012, pp. 2903–2910.

[15] J. Liu, P. Carr, R. T. Collins, and Y. Liu, "Tracking sports players with context-conditioned motion models," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013, pp. 1830–1837.

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *arXiv preprint arXiv:1311.2524*, 2013.

[17] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *BMVC*, 2010.

[18] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan, "Fast human detection using a cascade of histograms of oriented gradients," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, 2006, pp. 1491–1498.

[19] C. Wojek and B. Schiele, "A performance evaluation of single and multi-feature people detection," in *Pattern Recognition*. Springer, 2008, pp. 82–91.

[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[21] R. B. Girshick, P. F. Felzenszwalb, and D. A. Mcallester, "Object detection with grammar models," in *Advances in Neural Information Processing Systems*, 2011, pp. 442–450.

[22] S. Tang, M. Andriluka, A. Milan, K. Schindler, S. Roth, and B. Schiele, "learning people detectors for tracking in crowded scenes," in *Proc. IEEE Conf. ICCV*, 2013.

[23] J. Yan, X. Zhang, Z. Lei, S. Liao, and S. Z. Li, "Robust multi-resolution pedestrian detection in traffic scenes," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2013, pp. 3033–3040.

[24] M. Andriluka and S. R. an Bernt Schiele, "Pictorial structures revisited: People detection and articulated pose estimation," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 1014–1021.

[25] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Pose search: Retrieving people using their pose," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.

[26] Y. Yang and D. Ramanan, "Articulated pose estimation using flexible mixtures of parts," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2011, pp. 1385–1392.

[27] B. D. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. International joint conference on Artificial intelligence*, vol. 81, 1981, pp. 674 –679.

[28] P. Sand and S. Teller, "Particle video: Long-range motion estimation using point trajectories," *Intl J. Computer Vision*, vol. 80, no. 1, pp. 72–91, 2008.

[29] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1994.

[30] C. Bibby and I. Reid, "Real-time tracking of multiple occluding objects using level sets," in *proc. European Conference on Computer Vision*, 2010.

[31] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *IJCV*, vol. 29, no. 1, pp. 5–28, 1998.

[32] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1998.

[33] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2006, pp. 798–805.

[34] D. Ross, J. Lim, R.-S. Lin, and M.-H. Yuang, "Incremental learning for robust visual tracking," *Intl J. Computer Vision*, vol. 77, no. 1, pp. 125–141, 2008.

[35] X. Mei and H. Ling, "Robust visual tracking and vehicle classification via sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, pp. 2259–2272, 2011.

[36] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *proc. European Conference on Computer Vision*, 2008.

[37] S. Branson, P. Perona, and S. Belongie, "Strong supervison from weak annotation: Interactive training of deformable part models," in *Proc. IEEE Int. Conf. Computer Vision*, 2011, pp. 1832–1839.

[38] I. Tsochantaridis, T. Joachims, T. Hofmann, and Y. Altun, "Large margin methods for structured and interdependent output variables," *Journal of Machine Learning Research*, vol. 6, pp. 1453–1484, 2005.

[39] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *Journal of Machine Learning Research*, vol. 7, no. Mar, pp. 551–585, 2006.

[40] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *PAMI*, vol. 99, 2011.

[41] A. Ess, B. Leibe, K. Schindler, and L. van Gool, "A mobile vision system for robust multi-person tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2008.

[42] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[43] *Speeding Up Tracking by Ignoring Features*, 2014.

[44] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. H. S. Torr, "BING: Binarized normed gradients for objectness estimation at 300fps," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014.