Information feedback loop for improved pedestrian detection in an autonomous perception system

Martin Dimitrievski Peter Veelaert Wilfried Philips

IMEC-IPI-Ghent University
Sint-Pietersnieuwstraat 41
B-9000 Gent, Belgium

***Abstract*—Environmental perception systems for autonomous vehicles are often built using heterogeneous technologies that operate in a sequential manner. In the task of object tracking in particular, where the classical detector-tracker interaction is a serial process, it is viable to break the design rule by introducing information loops. This is especially feasible in a tracker that operates in a prediction-update cycle. Tracking predictions can steer object detection towards regions where an object is anticipated and, in turn, tracking updates can be improved by incorporating reinforced detections. In this paper we propose a novel detector-tracker feedback loop for information exchange based on the spatio-temporal similarity of detections and tracklets. We reinforce pedestrian detections that have weak confidence scores by matching their bounding boxes to estimated tracklets with high tracking confidence. The proposed system has several compelling advantages: based on a positive feedback principle it extracts the maximum detection and tracking information, while operating transparently and with minimal computational load. In a controlled ablation study we evaluate our feedback mechanism using the KITTI object tracking dataset. We show that our system gains significant performance increase over the baseline in both frame-by-frame detection and tracking quality.**

***Index Terms*—pedestrian detection, object detection, deep learning, tracking, multi-object tracking, feedback loop, autonomous vehicles, environmental perception**

I. Introduction

Environmental perception systems in autonomous vehicles are tasked with the challenging problem of traffic situational awareness. Understanding the environment is necessary so that the vehicle can reliably identify objects and make informed predictions and actions. Contemporary autonomous research platforms consist of heterogeneous sensor arrays, all of which operate in different modalities, at different sensitivity levels while covering only parts of vehicle surrounding. In this context, different computer vision algorithms have to be designed to co-operate using available data and also achieve temporal synchronization. On the downside, sensors are less than perfect and computer algorithms often have practical limitations. For example, cameras don't work well at nighttime, or in dazzling sunlight. LiDAR has trouble with rain, fog, and dust, because the laser bounces off the particles in the atmosphere. Radar can be confused by small but highly reflective metal objects, like a soda can in the street. Ultrasound sensors operate with a very limited range and resolution. Even systems that combine data from all sensors can struggle with images of humans on billboards, reflections off shop windows or photo realistic advertisements printed on other vehicles.

Standard perception systems rely on spatio-tempral object tracking and most often employ the principle of tracking by detection. First, an object detector trained off-line detects candidate targets in the image or LiDAR/Radar data. Trajectories are then estimated by connecting detected objects within a temporal window through a certain optimization algorithm. Due to the aforementioned sensor imperfections, object detection is usually not temporally consistent, so employing a tracker can correct for these temporal artifacts. Tracking-by-detection methods build upon detection and tracking as two distinct processes, which can sometimes lead to unsatisfactory perception results. Moreover, due to the diversity and complex occlusions of objects, the ability to detect all relevant traffic users (recall) often requires setting a very low detection threshold. Achieving high recall rates usually has the negative consequence of creating more false positives and decreases the efficiency of the later tracking algorithm.

To mitigate the effects of limited object detection performance we propose a novel information sharing technique by utilizing a feedback mechanism between the object detector and tracker. To this end, our method adds a loop in the perception system by reinforcing detections using tracking estimates. More specifically, our system consists of an object detector that operates in the image plane, which then feeds regions of interest to a 2D/3D object tracker. The object detector operates at a near 100% recall rate producing up to 10^3 candidate objects with reduced precision. However, our method feeds back the spatio-temporal information from the tracker to increase the precision of candidate objects that closely match tracked objects. The reinforced detection scores are then fed back into the tracker. Our proposed information feedback mechanism is designed to be agnostic of the design of object detector or tracker. The only requirement is that the object detector operates at a high recall rate and that the tracker works in an standard estimation-update cycle.

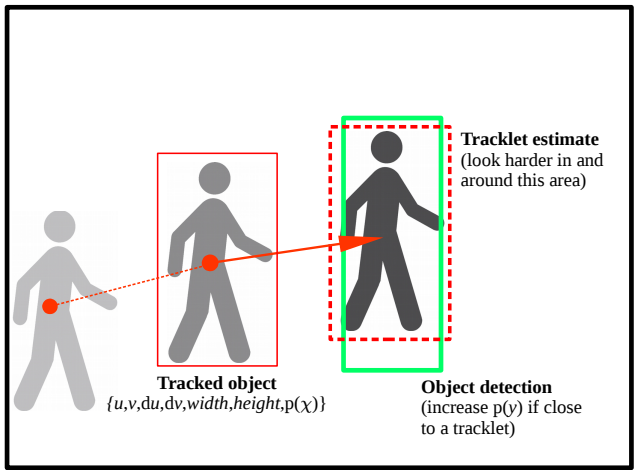

Figure 1. A pedestrian detection (green) and tracking (red) system where the position, size and velocity of a person is sequentially tracked. In the current frame, an estimate of the state is given (dashed line). We use the discrepancy between the estimated and detected position of a pedestrian to adjust the detection score of the candidate ROI.

In the following section we give a brief overview of several information feedback mechanisms from the literature. Then, in section §III we discuss the design of a generic multi-object detection and tracking pipeline where we formulate our novel feedback loop mechanism. Next, in section §IV we perform rigorous ablation experiments which single out the performance gains coming from our feedback loop mechanism over the baseline, and finally in section §V we conclude with some remarks on the fail cases of the system and how they can be remedied.

II. Related work

Various tracking by detection and detection by tracking feedback methods have already been considered in the literature. Tian et al. [1] perform multi-category multi-object tracking in traffic surveillance videos. They define two distinct situations where an image region is considered as a detection result. At initialization the region is detected when it is predicted from a tracklet and estimated as a foreground region at the same time even if it is not classified into object by the detector. During sequential tracking, the probability of detection is a product of both former terms and also the object detector. The downside of this approach is in the simplistic binary mathematical apparatus as well as that it relies heavily on background subtraction and is thus only applicable in static scenes. Furthermore, the paper is focused on measuring absolute tracking performance and lacks an ablation study to quantify the relative gain from the feedback loop.

Another approach by Li et al. [2] proposes a detection and tracker mutual feedback where detection is done by a Gaussian mixture model (GMM) of principal component analysis (PCA) features. Tracking is performed by computing the Bhattacharyya distance of the detected object and the tracklet which predicts the position of the tracking object based on expectation maximization (EM) Kalman filter. The feedback loop consists of detection of future candidate objects based on estimated tracklet positions and computed differences in the PDFs of the target and candidate regions. These authors claim that their scheme decreases the accumulation error and improves object detection and tracking performance. One drawback of this approach is the estimation of object motion by simple intensity difference between consecutive frames which can easily fail in presence of occlusions. This paper also lacks an extensive evaluation in order to accurate pinpoint the gains obtained by using a detector-tracker feedback loop.

Balntas et al. [3] propose a novel single object tracking method by online learning. Their model considers all detections, including false positives, provided by a classifier. The input detector is a high recall fern based classifier that returns a large set of candidate regions using the sliding window approach. Candidates, which the authors call pointers, are fused to form an estimated object position using a voting scheme in the Hough space. Voting is performed both in the spatial and in the temporal domain by using Euclidean and Hamming distance metrics of stored pointers. The maximum in the voting space is detected and target detections that overlap with this maximum are considered as valid. Experimental evaluation concludes a significant increase in precision and recall rates over the baseline detector and tracker, however this approach is limited in a sense that it is only applicable in single object tracking problems. Additionally, the method requires memory to store pointers from previous frames which hinders performance.

Ingersoll et al. [4] also investigate the tracker sensor feedback in stationary object detection from an UAV platform. Tracking information is sent back through a loop to inform the detector, which is a GMM for ROI estimation. They do so by a so called conservative scheme for updating the background model where they set an adaptive threshold for the minimum blob area, i.e. the extent of their target. At each step, a Kalman filter is updated with every ROI (blob) and keeps track of the object position and size. The feedback loop consists of setting the minimum blob area threshold in the GMM detector equal to three variances below the mean. By exploiting the feedback loop these authors report a significant improvement over their baseline which is measured by higher MOTA and MOTP tracking scores and lower false positive rate in detection. One serious limitation of this approach is that it assumes a static camera and heavily relies on the GMM foreground background detector for generating candidate objects.

An approach that adapts the appearance model for each particular object using on-line learning techniques is proposed in [5]. Authors 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 on-line using a structured SVM. They further improve the performance of the model-based tracker by on-line learning a prior distribution over the size of objects. Parameter updates are performed only if the base detector and the updated detector agree on the particular bounding

box for which the Intersection over Union (IoU) of 50% is used. Evaluation on the ETH pedestrian database [6] shows that the adapted detector outperform the baseline on some of the tested sequences. The biggest issue with this technique is that it tightly couples the design of the detector with the information propagated back from the tracker. The same concept is therefore difficult to re-implement in a different system environment.

Lastly, a method that exploits the sequential nature of videos to improve the quality of proposals based on the available information on previous frames determined by detector outputs is proposed in [7]. This method is actually independent of tracking as it re-ranks object proposals based on the overlap with detections and detector scores obtained by a state-of-the-art CNN approach, [8], in the previous frame. The authors propose a score re-weighting scheme based on the IoU measurement between ROIs in the current and the previous frame. Newly computed detection scores are a linear combination of the current score and the IoU times a normalization constant. This paper contains an ablation study where the performance of the proposed feedback loop is evaluated against detection of objects in the YouTube Objects dataset [9]. The downside of this method is the rather simplistic model of the feedback loop which doesn't exploit motion information of objects. Additionally, it lacks performance analysis for the class pedestrian.

In this paper we propose a detection-tracking feedback loop which improves the object perception by exploiting spatio-temporal correlation of pedestrian positions in autonomous driving settings. Our system operates on a continuous depth and video stream for detection and prediction of positions of other road users. We optimally exploit the estimated object positions from the tracker and achieve better frame by frame object detection. This improved detection rate, in turn, increases the tracking performance in a closed loop. The novelty of our proposed method lies in the principle of confidence boosting of ROIs that happen to lie near locations where we expect to track an object, figure 1. Additionally, we propose a design that is agnostic of both object detector type and object tracker in a way that it only relies on generic bounding box positions and estimated motion vectors. Finally, to the best of our knowledge, this is the first application and evaluation of such a feedback loop in a highly challenging autonomous driving environment. Thus, our method is able to reinforce heterogeneous sensing technologies without inferring any significant complexities or lag on the system.

III. Proposed method

A. General considerations

Multi-object tracking (MOT) is an umbrella term for methods covering multitude of applications. A large part of the research is done for the field of video surveillance where cameras are mostly static. Having this assumption, objects of interest can be easily detected as foreground (FG) blobs using a background (BG) model of the environment. Tracking of such blobs is usually done using a standard kinematic model

Algorithm 1 Proposed tracking by detection with feedback loop

At each time step t:

1) Apply an object detector to the frame I_t:
   $f(I_t) = S_y : \{\mathbf{y}_i\}_{i=1...n}$
   $\mathbf{y}_i = (u, v, width, height, label, s)_i$;
2) **Apply gating according to equation** (1)
3) Estimate the state of old tracklets
   a) *Kalman filter, Particle filter*
   b) *Optical flow vectors*
4) **Boost weak detections**
   foreach detection $\mathbf{y}_i \in \{\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_n\}$:
   if $s_i \leq \tau_1$: //weak detection
   foreach tracklet $\mathbf{k}_j \in \{\mathbf{k}_1, \mathbf{k}_2, ..., \mathbf{k}_m\}$:
   if $\chi_j > 0.9 \wedge J(\mathbf{k}_j, \mathbf{y}_i) > 0.8$
   $\hat{s}_i = s_i + (1 - s_i)\, exp\left(-\frac{[J(\mathbf{k}_j, \mathbf{y}_i)-1]^2}{\sigma^2}\right)$
   endif
   endfor
   endif
   endfor
5) Apply gating, ROI $\hat{s}_i > \tau$
6) Perform matching of reinforced ROIs and tracklets
   a) *Hungarian algorithm*
7) Update the state with matched data

of the object category on hand. As discussed in the overview, there are several techniques of how the FG/BG segmentation can be guided by the tracking process. Information can thus easily leak back from the tracker into the detector. However, these feedback techniques are intrinsically coupled to the system design and are difficult to port to newer detection and tracking technologies.

In autonomous driving, tracking objects from a moving camera is much more difficult since the static background assumption becomes invalid. Object detection must be performed by scanning every image position for possible object occurrences (objectness). A recent and highly efficient object detector is the Aggregated Channel Features detector (ACF) by Dollar et. al [10] which uses multi-resolution feature pyramids and a cascaded classifier to quickly scan the image plane for pedestrians. The design of the ACF cascaded classifier allows the detector to focus more attention on pedestrian-looking regions whilst quickly rejecting areas with clutter. Detected pedestrians are represented as regions within rectangular bounding boxes, each with a corresponding confidence value. However, as can be seen in the leader board of the KITTI object detection dataset, [11], the performance of ACF and similar approaches in autonomous driving scenarios is somewhat limited. Sliding window approaches in general are not able to recall 100% of the pedestrians due to performance limitations. Additionally, object detection on a frame-by-frame basis is not able to detect 100% of the objects in the presence of occlusions. A typical multi-object tracker such as the MDP [12], which utilizes ACF object detector as input

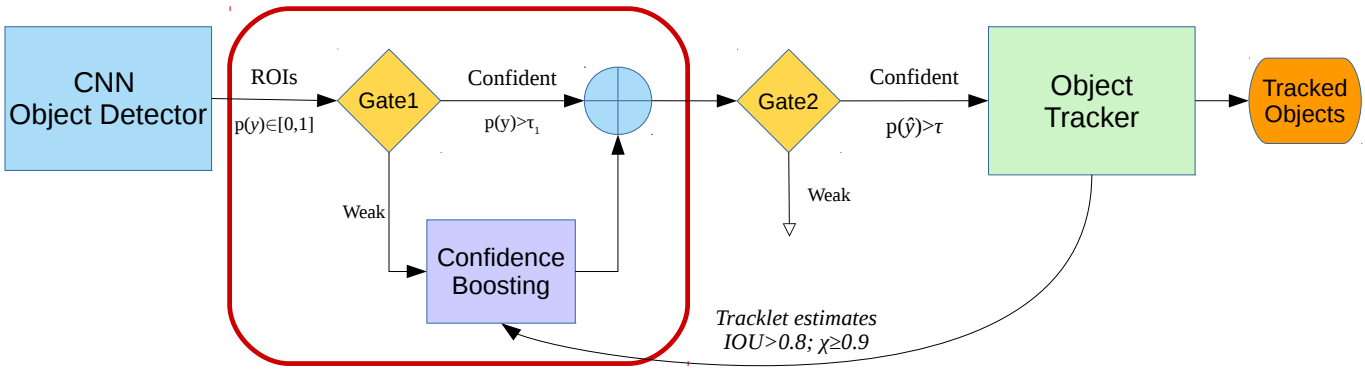

Figure 2. System diagram of our proposed feedback mechanism (in red). We separate initial ROIs based on the detection confidence. ROIs with weak detection confidence are matched against expected pedestrian positions from the tracker. Any matching ROIs get their detection confidence boosted accordingly while the rest are later discarded by a second gate.

must then try to estimate the position of missed pedestrians using various temporal consistency mechanisms.

More recently, the proliferation of high performance GPU computing paved the way for Convolutional Neural Network (CNN) based detectors by utilizing simple, yet efficient deep learning algorithms for training. These detectors can be trained to achieve 100% recall rates with varying degrees of precision. However, even using powerful GPU devices these CNNs tend to have slow execution times, which makes tracking infeasible. The advent of Region Proposal Networks (RPNs), [8], solves this problem by designing a pre-processing CNN that produces region proposals which are later classified as objects. In a typical camera frame there are around $10^3 \sim 10^4$ region proposals which usually cover close to 100% of all objects in the scene.

The task of the object tracker then is to select how, and which of these regions to track. In order not to overwhelm the tracker, MOT methods customarily employ gating to accept only highly confident ROIs. A kinematic and appearance based model then deals with any missed detections by exploiting their spatio-temporal and appearance based correlation from previous frames.

Measuring the performance of MOT methods reveals the differences of how each one handles missed detections, unpredictable motion, background motion, occlusions, etc. Most importantly, the better the input object detections are, the better tracking becomes. It thus becomes imperative to design an object detector with 100% recall rate and as high as possible precision. In the following sub-section we introduce our feedback loop mechanism, exemplified by the items 3 and 4 in algorithm 1.

B. Detection by feedback from tracking

Contemporary object detectors based on region proposal CNNs such as Regionlets [13], Faster R-CNN [8], SubCNN [14], YOLO9000 [15], etc. already perform at close to 100% recall on standard pedestrian detection benchmarks. One issue is that this is done at a great cost of precision where more than 10^3 object proposals can classified as pedestrians with low detection scores. It is therefore difficult to set the optimal gating threshold balancing between precision and recall. Given an image region $\mathbf{x}$, a typical detector output $\mathbf{y}$ is an object proposal defined by a bounding box with image plane parameters:

$$f(\mathbf{x}) = \mathbf{y}_i : \{u, v, width, height, label, s\},$$

where the score s is a classifier metric that represents certainty that the ROI belongs to a specific class ($label$). Conversely, a typical tracker output $\mathbf{k}$ is a tracked object represented by a bounding box, vectors of motion, appearance model and metrics for the tracking certainty:

$$\mathbf{k} : \{u, v, width, height, \dot{u}, \dot{v}, \text{appearance}, label, \chi\}.$$

The tracker usually cycles between an estimation and an update step, the former computes the most probable state of each object given the past states and measurements, while the later integrates the new data into the estimate to create an update of the estimate.

Our method interfaces with the detector-tracker system at the point where object detection and a tracking estimation is already performed and before the final update of tracking states is made. We use the set of tracking estimates $S_k : \{\mathbf{k}_j\}_{j=1\ldots m}$ to adjust confidence scores $s_{\mathbf{y}_i}$ (or s_i), i.e. we force the detector to look closer into regions where we expect to find tracked objects. Since we use an off-the-shelf object detection algorithm, we are not motivated to fine tune the inner workings of the detection and classification. Thus we focus on adjusting the confidence scores of some of the detection candidates. On figure 2 we present a schematic depiction of the proposed feedback loop (in red). Formally, at time t we employ gating $g(\mathbf{y}_i)$ to the set of initial object detections $S_y : \{\mathbf{y}_i\}_{i=1\ldots n}$ such that $S_y \rightarrow \{S_{strong} \vee S_{weak}\}$:

$$g(\mathbf{y}_i) \rightarrow \begin{cases} S_{strong} \ni \mathbf{y}_i, & s_i > \tau_1 \\ S_{weak} \ni \mathbf{y}_i, & s_i \leq \tau_1 \end{cases} \quad (1)$$

where τ_1 is the gating threshold manually adjusted so that it splits approximately 20% of the detections into the set S_{strong} and the rest in S_{weak}. Weak detections are then

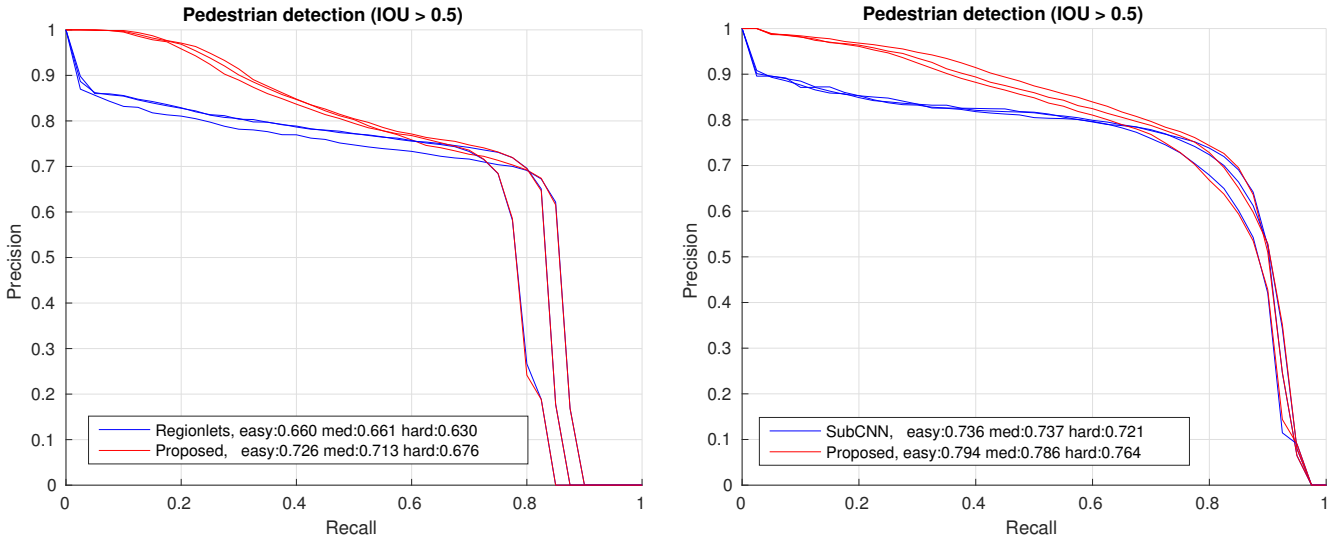

Figure 3. Evaluation of detection precision of the baseline (blue) against the same method using our proposed feedback loop (red) at $IoU > 0.5$. Left: Regionlets [13] and right: SubCNN [14].

matched against tracklet estimates $\mathbf{k}_j$ using the Jaccard index, i.e. Intersection over Union:

$$J(\mathbf{k}_j, \mathbf{y}_i) = \frac{|\mathbf{k}_j \cap \mathbf{y}_i|}{|\mathbf{k}_j \cup \mathbf{y}_i|}, \tag{2}$$

where the intersection and union operations are computed over the image bounding boxes of both detections and tracklets. We exploit this "closeness" information for detections that happen to fall close to expected pedestrian positions, $J \geq 0.8$, in a way that we adjust weak detection confidence values $s_i \,|\mathbf{y}_i \in S_{weak}$ in the following manner:

$$\hat{s}_i|\mathbf{k}_j = \begin{cases} s_i + (1 - s_i)\, exp\left(-\frac{[J(\mathbf{k}_j, \mathbf{y}_i)-1]^2}{\sigma^2}\right) & ;\chi_j > 0.9 \\ s_i & ;\chi_j \leq 0.9 \end{cases} \tag{3}$$

where σ controls the spread of the effect of the "closeness" between object and tracklet, while χ allows confidence boosting based only on accurately tracked objects. When the IoU is close to 1 and tracking confidence χ is above 0.9, the boosting of detection scores is maximal and as the IoU decreases the effect of the boosting diminishes. The motivation behind this mechanism lies in the temporal stability of observing pedestrians in video sequences. Once we are certain that we are tracking a pedestrian, $\chi > 0.9$, then we can be sure that it will be detected at or near the expected location by the tracker. If for some reason (camera noise, jitter, occlusion, shadows, etc.) the object detector is not very certain anymore, we can reinforce the score by factoring in how close it is to an expected pedestrian. Finally, our method concatenates the boosted and originally confident object detections into a list that is passed to the standard detection-tracking architecture. A second gating is then applied which removes any remaining false positives and the update step of the tracking is performed transparently.

IV. Experimental results

In order to evaluate the effectiveness of the proposed feedback loop mechanism we performed a series of experiments using prerecorded video sequences in real traffic environments, namely the object tracking benchmark of the KITTI [16] dataset. This benchmark includes 21 sequences with annotated ground truth of tracked objects and additional 29 sequences without available ground truth which are used for independent evaluation. For brevity, in our experiments we focus on the sequences $\{13, 15, 16, 17, 19\}$ of the training set since they contain most of the pedestrians while the rest are mainly sequences containing cars and other motorized vehicles. There are a total of 10312 instances from 143 unique pedestrians within the 2129 frames with a duration of the set of 3.5 minutes. The chosen sequences represent scenarios where the ego vehicle is driving through urban and campus environments with adequate amount of pedestrians. They provide traffic situations with a spectrum of difficulty such as moving camera, occlusions, difficult lighting, object interaction, etc. Using this dataset we can test for object detection precision and recall, various tracking performance metrics, but also evaluate the robustness of our feedback mechanism. To this end, we optimize the proposed solution using a single set of hyper-parameters which are applied while processing each and every frame of the selected dataset.

Average Precision				
Method	Easy	Medium	Hard	Improvement
Regionlets	0.660	0.661	0.630	
Regionlets*	0.726	0.713	0.676	**8.40%**
SubCNN	0.736	0.737	0.721	
SubCNN*	*0.794*	0.786	0.764	**6.83%**

Table I
AVERAGE PRECISION MEASURED AT 40 UNIFORMLY SPACED RECALL VALUES FOR THE BASELINE AND **PROPOSED*** METHOD.

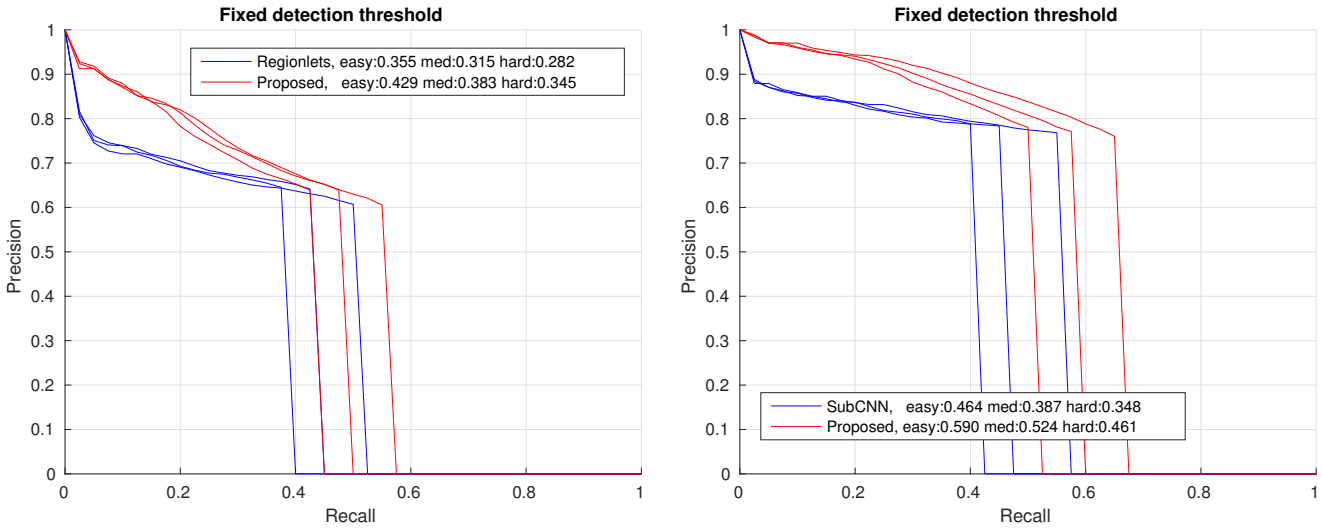

Figure 4. Evaluation of recall rates at a fixed detection threshold, baseline (blue) against our proposed feedback loop (red) at $IoU > 0.7$. Left: Regionlets [13] and right: SubCNN [14]..

We evaluate on a state-of-the-art pedestrian detection and tracking system built from the Regionlets [13] and SubCNN [14] object detectors and our own 2D-3D multi-object tracker based on the MDP [12] method. The input frame is processed with a CNN to detect ROIs with labels and detection scores which are then gated and passed to the tracker which tracks pedestrians based on position, appearance and motion. In all experiments, the baseline method uses a sequential processing pipeline, whereas the proposed method additionally incorporates our feedback loop mechanism to adjust the detection scores. Up to 10^3 candidate bounding boxes are generated by the detector which are then subject to our proposed confidence boosting algorithm, equation (3), and later passed to the object tracker.

The effectiveness of the proposed method is done by evaluation of both object detection and object tracking through ablation experiments where we measure precision, recall and Multiple Object Tracking Accuracy (MOTA). Precision and recall are detection specific metrics of how many detections are relevant and how many of the relevant detections are selected. MOTA combines false positives (FP), False Negatives (FN) and tracking identity switches IDS to indicate overall performance of the tracker. Formally MOTA is the ratio:

$$MOTA = 1 - \frac{\sum_t (FN_t + FP_t + IDS_t)}{\sum_t GT_t}, \qquad (4)$$

	Recall			
Method	Easy	Medium	Hard	Improvement
Regionlets	0.32	0.40	0.35	
Regionlets*	0.45	0.45	0.40	**20.93%**
SubCNN	0.37	0.35	0.32	
SubCNN*	0.57	0.50	0.45	**45.23%**

Table II
RECALL RATES OF BOTH BASELINES AND **PROPOSED*** METHODS USING A FIXED DETECTION THRESHOLD AT 65% FOR [13] AND 80% PRECISION FOR [14].

where t is the time step (frame index) and GT is number of Ground Truth objects. Value of MOTA can also be negative if the number of errors exceeds the number of actual objects. Most trackers in the literature are compared primarily using this metric since it represents a good balance between tracking precision, recall and temporal stability.

Firstly, we test the raw pedestrian detection performance of our two baseline detectors against the performance of the same methods with our proposed feedback loop. On figure 3 we report results for two experiments, one using Regionlets as a baseline (left plot) and the other using SubCNN as a baseline (right plot). Pedestrians are split into three categories $\{easy, medium, hard\}$ depending on their occlusion level and distance to the camera. We show that, in all three cases, pedestrian detection is significantly improved when using our proposed feedback loop. The precision rates, summarized in table I, of the Regionlets detector are improved by 8.4% on average while the SubCNN, which is originally the better performing detector, is further improved by 6.8%.

Next, we evaluated the impact on sensitivity that our feedback loop has on pedestrian detection. For this experiment we measured the recall rate at a fixed detection threshold. We used a threshold that coincides with 65% and 80% precision for the Regionlets and SubCNN detectors respectively. This choice was made by finding the critical point in the precision/recall curve where detection performance starts to deteriorate, as seen on the vertical axis in figure 3. In order to demonstrate the sensitivity more clearly, we also increased the bounding box matching threshold to $IoU > 0.7$. A higher matching threshold provides a more strict test that measures the absolute accuracy of the position of proposed object positions. Therefore, it stresses the quality of the tracking information which is fed back into the detection score. We measured that, on average, our feedback loop boosts the performance of both detectors by increases recall rates by 20.9% and 45.2% respectively. The results are

Method	Tracking performance metrics										
	MOTA	MOTP	MODA	MODP	F1	FAR	MT	PT	ML	IDS	FRAG
Regionlets (@65% prec.)	0.501	**0.786**	0.503	0.609	0.693	**0.076**	0.132	**0.650**	0.216	**13**	307
Regionlets* (@65% prec.)	**0.521**	0.785	**0.523**	**0.616**	**0.676**	0.078	**0.167**	0.622	**0.209**	19	**266**
SubCNN (@80% prec.)	0.512	**0.828**	0.514	0.655	0.682	0.034	0.202	**0.580**	0.216	**16**	325
SubCNN* (@80% prec.)	**0.531**	0.827	**0.533**	**0.658**	**0.699**	0.034	**0.216**	0.566	0.216	27	**291**

Table III
TRACKING PERFORMANCE EVALUATION, BOLD INDICATES BETTER RESULTS.

summarized in table II and shown on figure 4, where the left plot shows the output of the Regionlets detector and on the right is the output for the SubCNN detector. The cutoff points in the plots indicate where detections are below the chosen thresholds, however, there is clearly visible increase in the recall rates. This result shows that by using the same detection threshold, our boosted detector can detect more of the pedestrians present in the scene without producing additional false positives.

Lastly, we compare the tracking performance gains when turning on our feedback loop in the system. First, we performed tracking using the raw pedestrian detections of Regionlets and SubCNN while setting all tracker hyper-parameters the same value for each run of the system. Then, in both object detectors, we turned on our proposed feedback loop and fed the boosted object detections back into the tracker. We compare the measured MOTA scores of both baseline detector-tracker pairs to the MOTA of the trackers that use our feedback loop. On average, MOTA scores improved by 3.9% when applying the feedback loop on the Regionlets object detections, and 3.5% when applying to the SubCNN detections. These results are summarized in table III.

V. CONCLUSION

In this paper we show that tracking by detection in an autonomous vehicle environment can greatly benefit from adding an information feedback loop between the tracker and detector. By exploiting tracking estimates of pedestrian positions we showed that frame-by-frame detection can be greatly improved. Our pedestrian tracker estimates contain a non-trivial amount of information that we are able to leak back into the system and steer the object detector into regions of high probability of containing a pedestrian. We proposed a simple, yet effective mechanism for re-weighting object detection confidence scores that lie at or near positions of expected pedestrians. Using the IoU, as a basis for measuring closeness between detections and tracklet estimates, we are able to proportionally increase weak detections in areas of high likelihood for detecting a pedestrian. Experiments show that our detector-tracker system is more precise and at the same time has a greater pedestrian recall rate. Thus, pedestrians can be detected with more confidence at a fixed recall rate, or more pedestrians can be detected at the same precision level.

On the tracker side, adding the feedback loop shows that tracking performance also improves by a non-trivial amount. We observed an increase in overall tracking accuracy, explained by higher MOTA scores, but also higher tracking quality, i.e. longer tracked trajectories and less track fragmentations. All of these improvements upon the baseline system come at a minimal computational penalty. The computational burden of the added feedback loop block using our Quasar [17] GPU implementation is around $785\mu s$ per KITTI frame processed with Regionlets object detector and $729\mu s$ for the SubCNN detector.

We note that one of the most important assumptions is that our object detector operates at close to 100% recall rate. This way, it is theoretically possible to reinforce the weakly detected objects. In cases when perfect recall is not possible, our feedback loop will provide less than optimal results in a sense that completely missed objects cannot directly be recovered using the feedback information. Such cases indeed exist in reality and we suspect that they are later handled, to some degree, by the temporal mechanisms of the object tracker. Nonetheless, these effects shouldn't be ignored and will be the subject of our further study. One possible solution is to interface with the chosen object detector by steering it's model parameters using the feedback loop information, however, in doing so we will violate the transparency of the method and make it less general.

REFERENCES

[1] S. Tian, F. Yuan, and G.-S. Xia, "Multi-object tracking with inter-feedback between detection and tracking," *Neurocomput.*, vol. 171, pp. 768–780, Jan. 2016.

[2] X. Li and S. Bian, "Robust object detection and tracking using a space-temporal mutual feedback scheme," in *2008 IEEE International Symposium on Knowledge Acquisition and Modeling Workshop*, pp. 212–216, Dec 2008.

[3] V. Balntas, L. Tang, and K. Mikolajczyk, "Improving object tracking with voting from false positive detections," in *2014 22nd International Conference on Pattern Recognition*, pp. 1928–1933, Aug 2014.

[4] K. Ingersoll, P. C. Niedfeldt, and R. W. Beard, "Multiple target tracking and stationary object detection in video with recursive-ransac and tracker-sensor feedback," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1320–1329, June 2015.

[5] L. Zhang and L. van der Maaten, "Improving object tracking by adapting detectors," 08 2014.

[6] A. Ess, B. Leibe, K. Schindler, and L. V. Gool, "A mobile vision system for robust multi-person tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, June 2008.

[7] L. Galteri, L. Seidenari, M. Bertini, and A. D. Bimbo, "Spatio-temporal closed-loop object detection," *IEEE Transactions on Image Processing*, vol. 26, pp. 1253–1263, March 2017.

[8] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015.

[9] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning object class detectors from weakly annotated video," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3282–3289, June 2012.

[10] P. Dollár, R. Appel, S. Belongie, and P. Perona, "Fast feature pyramids for object detection," *PAMI*, 2014.

[11] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[12] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4705–4713, Dec 2015.

[13] X. Wang, M. Yang, S. Zhu, and Y. Lin, "Regionlets for generic object detection," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[14] Y. Xiang, W. Choi, Y. Lin, and S. Savarese, "Subcategory-aware convolutional neural networks for object proposals and detection," in *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 924–933, March 2017.

[15] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," *CoRR*, vol. abs/1612.08242, 2016.

[16] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, June 2012.

[17] J. De Vylder and B. Goossens, "Quasar: A programming framework for rapid prototyping," p. 1, NVIDIA, 2016.