Video Object Detection and Tracking based on Angle Consistency between Motion and Flow

Toshiki Seo[1,*], Tsubasa Hirakawa[1], Takayoshi Yamashita[1], Hironobu Fujiyoshi[1]

***Abstract*— Detect and Track (D&T) extracts a foreground region by using a feature map and region proposal network (RPN) and estimates an object class by using fully connected layers. A correlation layer, which is a hidden layer that obtains displacement between adjacent frames, estimates the movement and size of an object between the adjacent frames. Then, object class and regression are estimated by the feature maps obtained from the correlation layer and RPN. Finally, D&T estimates the moving direction and movement of a bounding box from the detection results obtained from the correlation layer and adjacent frames. Although D&T can achieve accurate object detection and tracking, the object detection and movement estimation of the correlation layer relies on the detection results of the RPN. Therefore, the correlation layer does not acquire local and global pixel changes in video frames and has to estimate the moving direction only from the similarity of detected regions. As a result, the estimation of the moving direction tends to fail. In this work, we propose a method to improve the moving direction estimation by performing the estimation in such a way as to maintain the consistency between the estimated direction and optical flow. Experimental results show that the proposed method can successfully estimate the moving direction and thereby improves both the detection and the tracking accuracy.**

I. INTRODUCTION

Object detection is widely used for driving support systems and for surveillance camera analysis. Various approaches such as a histogram of oriented (HoG) feature and a support vector machine (SVM) [1], template matching-based detection [2], and key-point matching-based detection [3] have been proposed. In recent years, object detection methods based on a convolutional neural network (CNN) have been widely investigated [4], [5], [6]. Most of the CNN-based detection methods deal with a single frame that trains a network with a loss value calculated from the positional relationship and the size of the target objects. However, when applying such detection methods to a video sequence, occlusion and changes of object size between successive video frames reduce the detection accuracy.

Detection methods that use adjacent frames [7], [8] to detect occluded objects have shown promise because they can consider the movement of objects and the relationship between objects that appear in video frames. *Detect to Track and Track to Detect* (D&T) [8] is one such detection method. D&T first extracts foreground regions by using feature maps obtained from a feature extractor and a region proposal network (RPN) [9] and estimates the object class by using fully connected layers. Then, object movement and size between adjacent video frames are estimated by a *correlation layer* that acquires the displacement between the adjacent frames. The feature map obtained from the correlation layer is used by the RPN to estimate object class and bounding box regression. These results are then input to a region of interest (RoI) tracking module. By using the detection results obtained from both adjacent frames and the correlation layer, D&T can estimate the movement of the bounding boxes and select the correct one, which leads to improved detection accuracy.

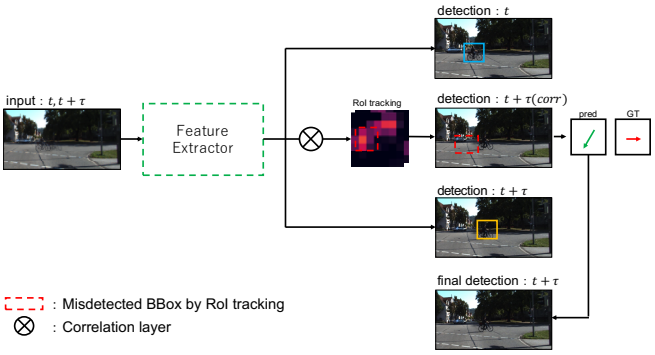

Fig. 1. An example of failure to detect object in D&T. D&T applies an RPN for a feature map obtained from a correlation layer and obtains a vector representing how much the object moves from frame t to $t+\tau$. Estimating an incorrect vector leads to an incorrect detection result.

However, the object detection and direction estimation of the correlation layer does not acquire local and global pixel changes over video frames, which means that the moving direction is estimated on the basis of only the similarity of detected regions. An example of a mis-detection by an RoI tracking module is shown in Fig. 1. If the detection results obtained from the candidate region of the RPN are incorrect, the estimated moving direction is not appropriate and an incorrect detection result is provided.

In this paper, we propose a method to appropriately estimate the moving direction and thereby accurately detect objects. The proposed method consists of two key components. First, we construct a multi-task learning network consisting of D&T and optical flow estimation. To estimate optical flow, we adopt FlowNet [10], specifically, FlowNetC, as it also contains a correlation layer. Since D&T and FlowNetC both have a correlation layer, the proposed method can train the similarity of a vector representing object movement direction and optical flow obtained from FlowNetC. Second, we define a novel loss function that maintains consistency between optical flow and estimated moving direction and utilize the estimated optical flow to determine object movement

[1] Chubu University, Kasugai, 487-8501, Japan.

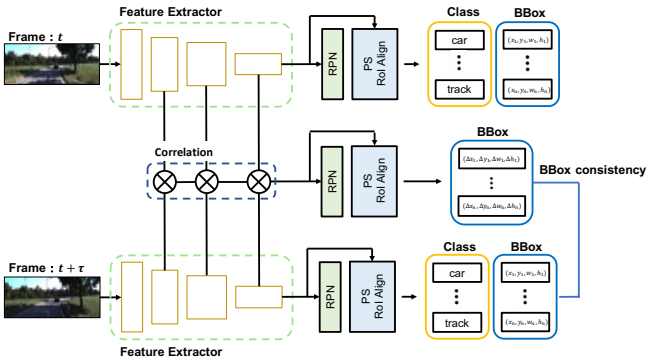

Fig. 2. The network structure of D&T.

direction. Through experiments, we discuss how our method affects the detection results and tracking results.

Our contributions are as follows:

- Focusing on the fact that D&T and FlowNetC both have a correlation layer and output a similar feature map, we define a unified network as a multi-task learning framework of D&T and FlowNetC.
- The proposed method outputs an optical flow related to the moving direction obtained from the RPN. We utilize this optical flow to compute the moving direction and achieve accurate object detection that considers appropriate moving direction.

II. RELATED WORK

A. Object detection for video frames

Detect to Track and Track to Detect (D&T) [8] is a method to detect and track objects from video sequences. Figure 2 shows the network structure of D&T. It extends a region-based fully convolutional network (R-FCN) [11] and estimates the size and movement of an object from correlated feature maps of object candidate regions obtained from an RPN [9] in the R-FCN. Then, objects are detected and tracked by the mutual relationship between each object.

D&T consists of three modules: a module to detect objects from a single image, an RoI tracking module, a module to decide class score. The first module detect objects from a single image by R-FCN. The RoI tracking module, which consists of a correlation layer, estimates the moving direction of an object and decides the object candidate region. The third module decides the class score on the basis of the object detection results from the intersection over union (IoU) correlation layer and R-FCN. When selecting a bounding box from IoU, D&T achieves higher detection accuracy than R-FCN because the estimated class score is added to candidate regions whose IoU is higher than 0.5. During the training, a loss function is introduced for tracking results L_{tra} in addition to the loss functions for classification L_{cls} and bounding box L_{loc} used in Fast R-CNN.

The moving vector of D&T is estimated by the relationship between the candidate regions of frame t and $t+\tau$. Therefore, if RPN outputs incorrect detection results D&T might not obtain the appropriate moving vector. Our method solves this problem by introducing optical flow estimation to accurately estimate the moving vector, which improves the detection and tracking accuracies.

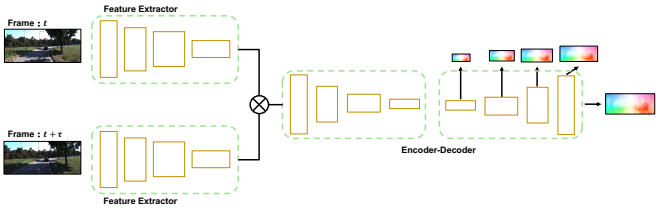

Fig. 3. The network structure of FlowNetC.

B. Optical flow

One of the best optical flow methods is currently FlowNet [10], which is based on a fully convolutional neural network architecture. FlowNet has two architectures: FlowNetS and FlowNetC. FlowNetS estimates optical flow by convolution and deconvolution for concatenated video frames, while FlowNetC inputs adjacent video frames into a feature extractor separately and then applies deconvolution to the displacement obtained from the correlation layer.

In this work, we focus on the network structure of FlowNetC, which is shown in Fig. 3. At the correlation layer of FlowNetC, we apply several convolutions for input video frames and obtain feature maps $f_1(x_1)$ and $f_2(x_2)$. Then, we can obtain the displacement of the maps to estimate object movement by concatenating feature maps $c(x_1, x_2)$ with movement width o, as

$$c(x_1, x_2) = \sum_{o \in [-k,k] \times [-k,k]} \langle f_1(x_1 + o), f_2(x_2 + o) \rangle, \quad (1)$$

where k is the focused region with a focus on o. The size of the correlated feature map is aligned by padding and stride.

III. PROPOSED METHOD

The proposed method consists of two key components. First, we construct the network structure as a multi-task learning framework by focusing on the correlation layer that is used in both D&T and FlowNetC. These output similar feature maps. Second, we define a loss function L_{rad} using optical flow and acquire the frame direction in adjacent video frames. This enables the proposed network to decide the appropriate direction of a moving vector. Figure 4 shows the proposed network structure. In this network, we first detect objects from video frames by R-FCN. Then, we input feature maps obtained from the convolutional layers into the correlation layer and generate a correlated feature map. The correlated feature map is input to the RPN module and the PS RoI align module to estimate the size and movement of the bounding boxes and to the FlowNetC-based module to

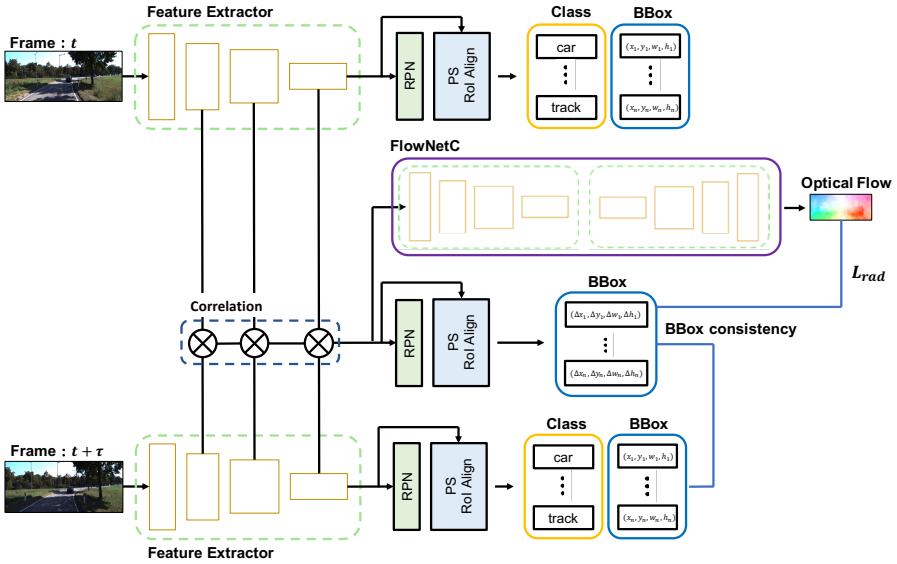

Fig. 4. The network structure of the proposed method. This network consists of an object detection module, a correlation layer, and a module to estimate optical flow and object movement. Because the feature extractor is trained by a multi-task learning framework of optical flow, we can remove the FlowNetC architecture during inference.

estimate optical flow. This network can be trained in an end-to-end manner. Hereafter, we introduce the details of each module and process.

A. Multi-task learning framework

D&T and FlowNetC have a common process that estimates the displacement for two input images. These methods are also similar in that they both use a correlation layer. Therefore, it seems likely that the accuracies on both tasks could be simultaneously improved by multi-tasking. In multi-task learning, we can obtain better intermediate feature representations that are related to both tasks [12]. As shown in Fig. 4, we integrate the correlation layer and can therefore output the results of D&T and optical flow at the same time.

The structure of the RPN used for estimating the foreground region is defined as shown in Tab. I. We adapted the ordinary RPN structure in Faster R-CNN by adding a convolutional layer and a batch normalization layer [13] in each layer to make the training more stable and efficient.

B. Loss function based on the angle error

Herein, we define the loss function to train the proposed network. The proposed loss function is based on those of D&T and FlowNetC.

The loss function for D&T, L_{dt}, introduces a loss function for object tracking L_{tra} in addition to the losses for classification L_{cls} (i.e., cross-entropy loss) and for bounding boxes L_{loc} (i.e., smoothL1 loss). L_{tra} computes the loss for the bounding box regression between the ground truth of frame t and the predicted box of frame $t+\tau$. Let $\{p_i\}_{i=1}^{N}$, $\{b_i\}_{i=1}^{N}$, and $\{\Delta_i^{t+\tau}\}_{i=1}^{N_{tra}}$ be confidence, offset, and moving vector of the RoI tracking module, respectively. The loss for D&T L_{dt} is defined as

$$
\begin{aligned}
L_{dt} =& L(\{p_i\},\{b_i\},\{\Delta_i\}) \\
=& \frac{1}{N}\sum_{i=1}^{N} L_{cls}(p_i, c^*) \\
&+ \lambda\frac{1}{N_{fg}}\sum_{i=1}^{N}[c_i^* > 0]L_{loc}(b_i, b_i^*) \\
&+ \lambda\frac{1}{N_{tra}}\sum_{i=1}^{N_{tra}} L_{tra}(\Delta_i^{t+\tau}, \Delta_i^{*,t+\tau}),
\end{aligned}
\tag{2}
$$

where N, N_{tra}, and N_{fg} are the numbers of RoI predictions, ground truth of candidate region by RoI tracking, and foreground RoIs. where N, N_{tra}, and N_{fg} are the numbers of RoI predictions, the ground truth of candidate region by RoI tracking, and foreground RoIs. An indicator function $[c_i^* > 0]$ represents foreground for 1 and background for 0, where $\lambda(=1)$ is the tradeoff parameter. Here, we denote the bounding box as $B^t = (B_x^t, B_y^t, B_w^t, B_h^t)$ at frame t. The elements of moving vector $\Delta^{*,t+\tau} = \{\Delta_x^{*,t+\tau}, \Delta_y^{*,t+\tau}, \Delta_w^{*,t+\tau}\Delta_h^{*,t+\tau}\}$ is defined as

$$
\Delta_x^{*,t+\tau} = \frac{B_x^{t+\tau} - B_x^t}{B_w^t} \tag{3}
$$

$$
\Delta_y^{*,t+\tau} = \frac{B_y^{t+\tau} - B_y^t}{B_h^t} \tag{4}
$$

$$
\Delta_w^{*,t+\tau} = \log\frac{B_w^{t+\tau}}{B_w^t} \tag{5}
$$

$$
\Delta_h^{*,t+\tau} = \log\frac{B_h^{t+\tau}}{B_h^t}. \tag{6}
$$

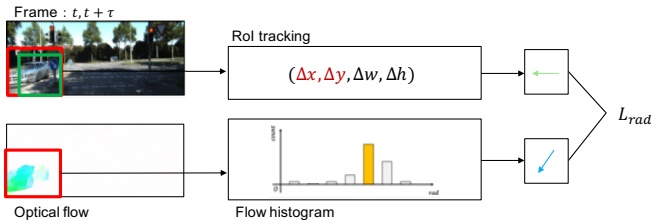

Fig. 5. Outline for angle consistency loss L_{rad}. This loss function keeps a consistency with motion and flow histgram.

The loss for optical flow L_{flow} is defined by

$$L_{flow} = \frac{1}{N_{deconv}} \sum_{i=1}^{N_{deconv}} MSE(flow_{gt_i}, flow_{pred_i}), \quad (7)$$

where N_{deconv} is the number of deconvolutional layers and MSE is mean squared error. $flow_{gt_i}$ and $flow_{pred_i}$ are the ground truth and output flow, respectively.

Furthermore, to keep angle consistency between the movement estimated from D&T (u, v) and the ground truth optical flow in the candidate bounding boxes (u', v'), we add a loss function L_{rad}, as

$$L_{rad} = \min_{u,v} \cos^{-1} \frac{(u, v) \cdot (u', v')}{|(u, v)||(u', v')|} \quad (8)$$

This function is defined as a minimization problem of angle error based on the inner product of optical flow and the moving vector obtained from RoI tracking. Figure 5 shows the outline for the computation of angle error L_{rad}. The optical flow in the object candidate region is converted into a gradient histogram. We set the mode of the histogram as the ground truth label. By introducing this loss function, we can make the moving direction of the candidate region close to the optical flow.

Consequently, the loss function used in the proposed network L is defined as

$$L = L_{dt} + L_{flow} + L_{rad} \quad (9)$$

Note that we make these loss elements converge in the order of L_{dt}, L_{flow}, and L_{rad} for stable and successful training.

IV. EXPERIMENTS

A. Datasets

In these experiments, we assume the method is applied for an autonomous driving application. Virtual KITTI [14] and KITTI [15] are the datasets used.

1) Virtual KITTI: The Virtual KITTI dataset contains synthetic videos taken under several conditions with different weather (e.g., rain, fog, overcast) and times of day (e.g., sunset, noon). Moreover, several ground truths for 2D and 3D multi-object tracking, optical flow, and depth are contained In this experiment, we use 3,210 samples for training and 1,014 for evaluation.

TABLE I
DETAILED STRUCTURE OF RPN.

Layer	Detail
1st Conv.	kernel: 1024 × 1 × 1 activation func.: ReLU stride: 1
1st BN	size: 1024
2nd Conv.	kernel: 1024 × 1 × 1 activation func.: ReLU stride: 1
2nd BN	size: 1024

2) KITTI: The KITTI dataset contains images recorded by an on-board vehicle camera in Karlsruhe, Germany. This dataset includes stereo images, optical flow, odometry, 3D object detection, and tracking annotations. At most, 15 cars and 30 pedestrians appear in a single image. In this experiment, we evaluate the proposed method on an 7-class object detection and tracking task with 5,796 samples for training and 1,191 for evaluation from *object tracking evaluation 2012* data.

B. Baselines

As conventional methods, we use R-FCN and D&T and compare their accuracies with that of the proposed method. In addition, we use the following proposed models:

D&T+f: The proposed network is constructed as a multi-task learning of D&T and FlowNetC. During the training we use only loss functions of D&T L_{dt} and FlowNetC L_{flow} (shown in Eqs. (2) and (7)).

D&T+f+L_{rad}: The proposed network is trained using the proposed loss function L_{all} (shown in Eq. (9)).

D&T+f+O_{eval}: This method trains the proposed network as with D&T+f+L_{rad}. During the inference, we decide the moving direction by using optical flow in a candidate region (as shown in Fig. 5).

We used ResNet101 [16] as the feature extractor for all methods. Also, as described in the previous section, we use the RPN defined in Tab. I.

C. Evaluation metrics

As an evaluation metric, we use mean average precision (mAP) for the object detection task. Also, we use object racking rate $Track_{acc}$ for the object tracking task. Let Box_t be the number of bounding boxes in frame t. $Track_{acc}$ is defined by

$$Track_{acc} = \sum_{i=0}^{t} \frac{Box^c_{(t)} \cap Box^c_{(t+1)}}{gtBox^c_{(t)} \cap gtBox^c_{t+1}}. \quad (10)$$

This means how many the same objects are detected over adjacent frames t and $t+1$.

D. Results

Table II shows the evaluation results for the Virtual KITTI and KITTI datasets. Note that the IoU of mAP is set to 0.5. The proposed method achieved the highest accuracies on both datasets. The mAP for Track and Van

TABLE II

THE EVALUATION RESULTS ON VIRTUAL KITTI AND KITTI DATASETS.

	Virtual KITTI		KITTI								
	All	$Track_{acc}$	Van	Cyclist	Pedestrian	Car	Misc	Truck	Tram	All	$Track_{acc}$
D (R-FCN)	84.4	68.3	80.4	73.4	74.8	79.2	71.4	**90.2**	79.1	79.8	71.1
D&T	86.4	71.2	80.4	77.3	77.3	82.5	75.7	90.0	86.4	82.1	76.2
D&T+f	86.6	82.9	80.7	77.9	78.7	**83.4**	**77.2**	**90.2**	89.8	83.9	81.0
D&T+f+L_{rad}	**87.3**	83.4	**80.8**	**78.9**	79.1	83.3	**77.2**	90.0	**90.0**	**84.4**	**82.4**
D&T+f+O_{eval}	87.2	**83.5**	**80.8**	78.4	**79.2**	83.2	**77.2**	90.0	89.9	84.3	82.2

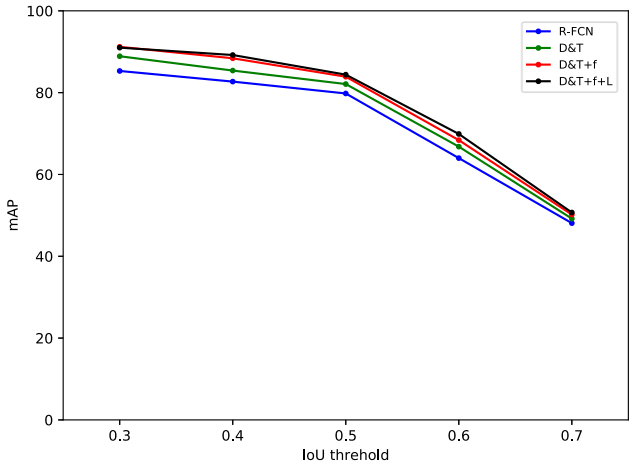

Fig. 6. mAP over different IoU thresholds on KITTI dataset. The proposed method outperforms the converntional methods for every thresholds.

was barely changed because these objects do not move much in the evaluation data. In contrast, the mAPs of moving objects, i.e., Cyclist, Pedestrian, and Car, were improved by the proposed method. Moreover, regarding $Track_{acc}$, our method improved the score because of the multi-task learning framework.

Figure 6 shows the mAP over different IoU thresholds on the KITTI dataset. The proposed method outperforms the conventional methods for every thresholds. Detection results on KITTI dataset is shown in Fig. 7. In the results of D&T, tracking is interrupted and the detection is failed in frame t. Meanwhile, the proposed method successfully detect these objects without such tracking interruption. We also show the estimated optical flow and the moving vectors of the proposed method in Fig. 7. For the D&T, we show the moving vectors when detection is failed between frames $t-1$ and t. Focusing on the moving vectors, the vector of the proposed method can obtain moving direction corresponding to the post frame. These results show that the proposed method and the loss function L_{rad} improve the detection and tracking results.

V. CONCLUSIONS

In this paper, we proposed a method to estimate moving direction and detect objects accurately in video sequences. Focusing on the common part of D&T and FlowNetC, namely, the correlation layer, we constructed a unified network structure as a multi-task learning framework. A loss function is defined for the moving angle considering optical flow, which improves the accuracies of object detection and tracking. Experimental results show that the proposed method with training that maintains consistency of moving direction improved the detection performance in successive video frames. Our future work will include reducing the computational cost of the network and extension to three-dimensional object detection.

REFERENCES

[1] Y. Pang, Y. Yuan, X. Li, and J. Pan, "Efficient hog human detection," *Signal Processing*, vol. 91, no. 4, pp. 773–781, 2011.

[2] C.-W. Park and M. Park, "Fast template-based face detection algorithm using quadtree template," *Journal of Applied Sciences*, vol. 6, no. 4, pp. 795–799, 2006.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, vol. 1. IEEE, 2005, pp. 886–893.

[4] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[5] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[7] J. Mitch, "Image frame detection," May 9 1995, uS Patent 5,414,779.

[8] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Detect to track and track to detect," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3038–3046.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[10] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox, "Flownet: Learning optical flow with convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2758–2766.

[11] J. Dai, Y. Li, K. He, and J. Sun, "R-fcn: Object detection via region-based fully convolutional networks," in *Advances in neural information processing systems*, 2016, pp. 379–387.

[12] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[13] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[14] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4340–4349.

[15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

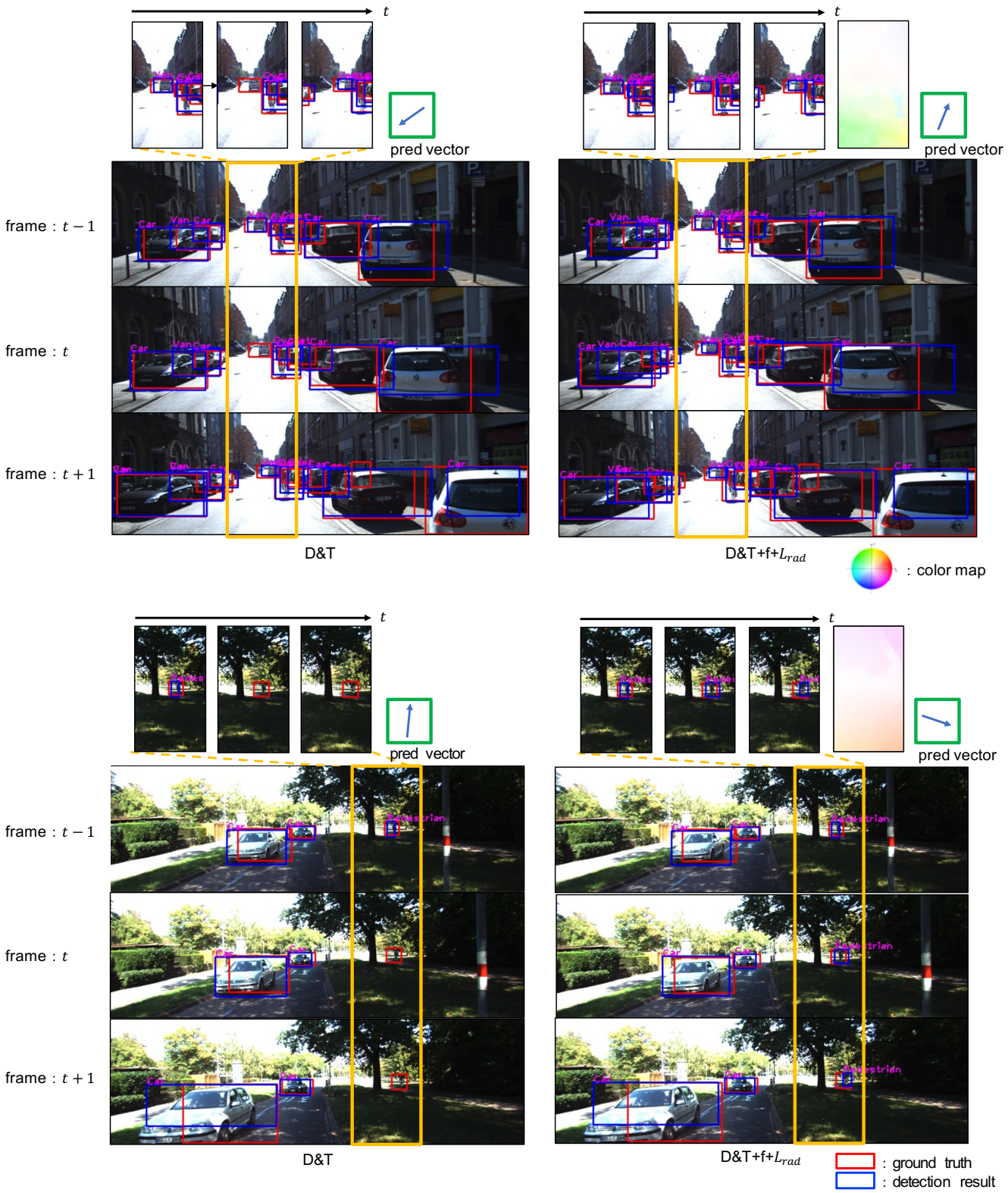

Fig. 7. Detection results on D&T and the propoosed method (D&T+f+L_{rad}). The red box indicates ground truth and blue box indicates the detected results. Orange box shows the detection results which are improved by the proposed method. Also, we show the moving vectors (pred vector) of un-detected bounding box for the conventional method and the proposed method and optical flow at frame t.