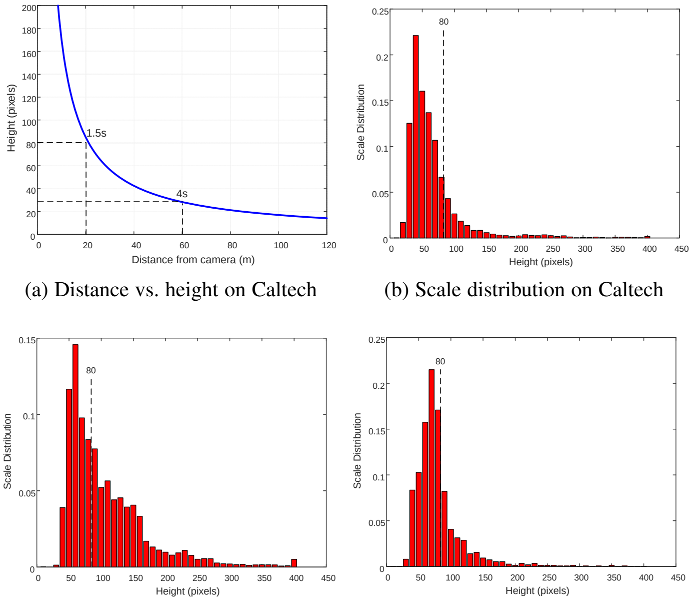
Too Far to See? Not Really — Pedestrian Detection with Scale-aware Localization Policy

1 Data statistics on public benchmark

Due to the effect of spatial scales on pedestrian detection, we investigate the scale distribution of pedestrian instances on the public benchmark. Followed by P. Dollar [5], we group pedestrian instances by their image size (height in pixels) into two scales: near scale(80 or more pixels) and far scale(under 80 pixels). This division is motivated by the distribution of sizes in the data set, human performance, and automotive system requirements. Note that below 30 pixels, annotators have difficulty identifying pedestrians reliably, we focus on pedestrian instances above 30 pixels in height.

For safety automotive systems, most pedestrians are observed at the far scale(between 30-80 pixels in height) and detection must occur in this scale as well because there leave sufficient time to alert the driver. As shown in Fig. 1, with the vehicle traveling at an urban speed of 15 m per second, an 80 pixel person is just 1.5 s away, while a 30 pixel person is 4 s away. Moreover, a large proportion of the pedestrians lie in the far scale, approximately 81.6% for the Caltech dataset, 48.4% for ETH dataset and 73.8% for TUD-Brussels dataset as shown in Fig. 1. However, most current detectors are designed for the near scale and perform poorly even at the far scale. Thus, there is an important mismatch in current research efforts and the requirements of real systems. Using higher resolution cameras would help; nevertheless, given the good human performance and lower cost, we believe that accurate detection in the far scale is an important and reasonable goal.



(a) Distance vs. height on Caltech

(b) Scale distribution on Caltech

(c) Scale distribution on ETH

(d) Scale distribution on TUD-Brussels

Fig. 1: (a) Pedestrian pixel height as a function of distance. Assuming an urban speed of 55 km per hour, an 80 pixel person is just 1.5 s away, while a 30 pixel person is 4 s away. (b) (c) (d) Scale distribution of pedestrian pixel heights. And most proportion of observed pedestrians are at the far scale.

TABLE 1: Comparison of different features under IoU=0.5 on the Caltech benchmark.

RoI features	Time/image	Near-scale	Far-scale
ResNet-50-C1	0.38	78.98%	27.50%
ResNet-50-C2	0.38	89.17%	32.62%
ResNet-50-C3	0.38	90.45%	57.91%
ResNet-50-C4	0.38	95.54%	53.50%
ResNet-50-C5	0.38	92.68%	44.16%
ResNet-50-C3, C4, C5	0.42	97.13%	68.74%

2 Our approach

2.1 Multi-layer feature representation of our proposed method

The original implementation of Faster R-CNN with ResNets extracted features from the final convolutional layer of the 4-th stage, which is denoted as ResNet-50-C4 by a common choice used in [7] because of the backbone with ResNet-50. In this work, following the common caption, we compared the multi-layer feature representation from ResNet-50-C1 to ResNet-50-C5 in the initial proposal selection. Fig. 2 shows the network architecture of multi-layer feature representation for ResNet-50. We can see that the residual network has two kinds of shortcut connections. One dotted line shortcuts performs identity mapping, with extra zero entries padded for increasing dimensions. Another solid line shortcuts can be used to match dimensions (done by 11 convolutions) when the input and output are of the same dimensions.

We study the recall rate of pedestrian proposals over different convolutional layer (from ResNet-50-C1 to ResNet-50-C5) of ResNet [7] using the faster RCNN framework [10]. It

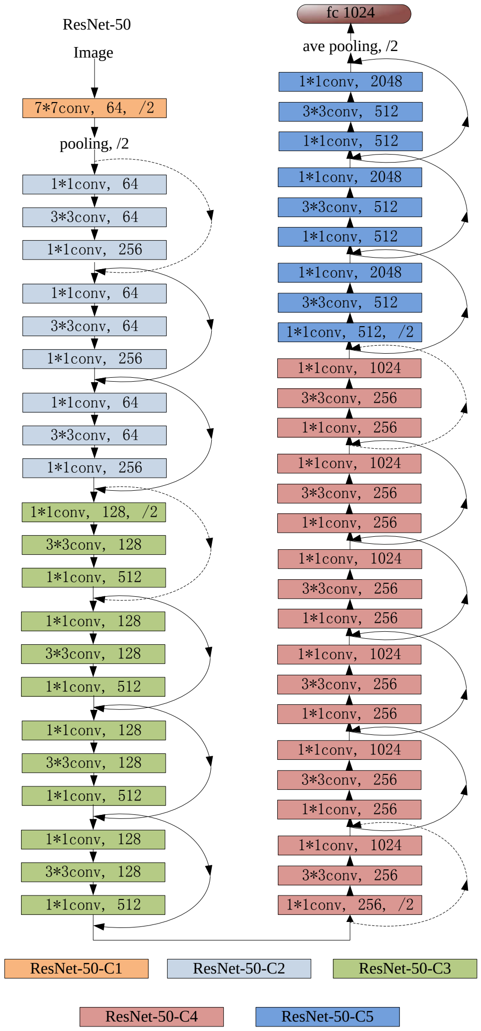


Fig. 2: Residual network architecture[7]: a residual network with 50 parameter layers (called ResNet-50). The dotted shortcuts increase dimensions.

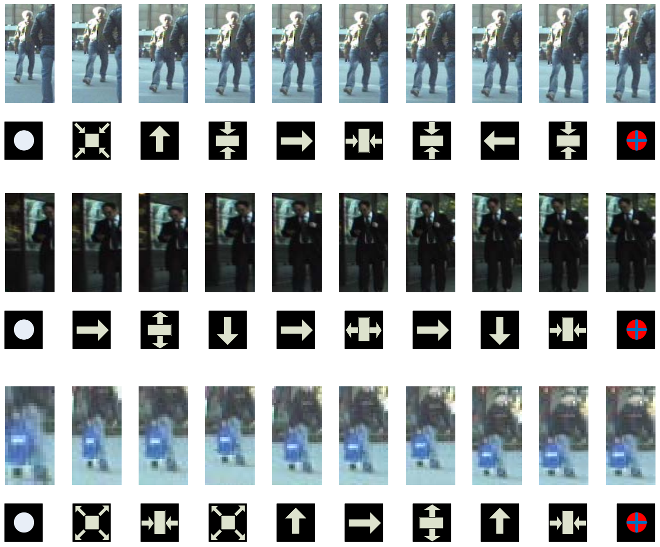


Fig. 3: Example sequences observed by the agent and the actions selected to focus pedestrian instances. Regions are warped in the region of interesting pooling layer in the ResNet-50 as they are fed to RNN. Actions keep the object in the center of the box.

is observed that the feature maps of the higher convolutional layers can encode the semantic information of targets and such representations are robust to significant appearance variations. ResNet-50-C1 and ResNet-50-C2 show poor recall rate 78.98% and 89.17% respectively, which can be explained by the weaker representation of the shallower layers. ResNet-50-C3, ResNet-50-C4, and ResNet- 50-C5 alone yields good results, showing the effects of higher convolutional features. For near-scale pedestrian instances, the higher convolutional layers (e.g., ResNet-50-C5) perform better than in lower convolutional layers (e.g.,ResNet-50-C3) for object proposals generation task as shown in Fig.7(a) of the main text. Especially, ResNet-50-C4 gets 95.54% recall for generating near-scale proposals in a single convolutional layer, outperforming ResNet-50-C3 by 5.09% and ResNet-50-C2 by 6.37% with IoU = 0.5 in Table 1. However, due to the coarseness of the higher convolutional feature maps, ResNet-50-C4 is reduced by 4.41% compared to ResNet-50-C3 for pedestrian instances under far-scale in Fig.7(b). As Table 1 shows the recall rate is considerably improved to 97.13% and 68.74% for near- and far-scale respectively, when jointly considering the multi-layer representation of ResNet-50-C3, ResNet-50-C4 and ResNet-50-C5 features.

2.2 Our Active Detector Model

The object localization of pedestrian detection as the Markov decision process (MDP) of a goal-directed agent interacting with a visual environment. Our formulation treats a single image as the environment, the goal of the agent is to transform a bbox to a tight box by a series of actions, which combine contextual regions around each object proposal for a target object to improve localization accuracy. Compared with [2], allows an agent to adaptively choose the feature maps catering for current object sizes from the ResNet-50 in the localization task. The agent has a localization state with information of the currently visible region and past history actions, and receives positive and negative rewards for each localization decision made during the training phase. In our localization policy,The reward at the last time step was 1 if the agent classified correctly and 0 otherwise. The rewards for all other timesteps were 0. The state of the environment would correspond to the true contents of the image, the environmental action would correspond to the classification decision. The agent can also affect the true state of the environment by executing actions. Since the environment is only partially observed the agent needs to integrate information over time in order to determine how to act most effectively.

The sequences of attended regions by our context-aware localization policy are shown in Fig. 3, as well as the actions selected in each step. Notice that regions are warped in the Region-of-Interest (RoI) pooling layer from ResNet-50 as they are fed to the Markov decision process of our context-aware localization policy. The actions chosen attempt to approach the ground-truth label by maximizing the conditional probability of the true label given the localizations from the image.

3 Visualized Experimental Results

In order to observation the detection performance of our proposed approach, we visualize the experimental results of pedestrian detection on the public benchmarks. The detection performance of our proposed approach is evaluated to the state-of-the-art methods on Caltech [4], ETH [3] and TUD-Brussels [12] datasets, including SpatialPooling [9], LDCF [8], TA-CNN [11], MS-CNN [1], and SA-FastRCNN [6].

Fig 4 shows the pedestrian detection results of our proposed method, and four state-of-the-art methods including MS-CNN [1] and SA-FastRCNN [6]. A detected image sliding window is represented as true positive by the green dotted bounding box when the overlapping area between the detected window and the ground truth (green solid bounding box) exceeds 50%, or false positive by the red dotted bounding box if otherwise. As shown in Fig. 4, the green dotted boxes demonstrate the detection results for pedestrian detection. It is observed that our proposed method has less detection errors compared to state-of-the-art methods such as MS-CNN [1] and SA-FastRCNN [6]. Moreover, our scale-aware localization policy adaptively chooses the convolutional features with different resolutions for various scale pedestrians, the small-size pedestrian instances also can be detected, where the red dotted bounding box represents the positive pedestrians which are not marked by the ground truth as shown in Fig. 4.

Similar observation to what we have observed for the ETH benchmark occurs in Fig. 5. We can see that the state-of-the-art method TA-CNN [11] have the reasonable detection results, while also having more missings. And while SpatialPooling [9] can recall the ground truth of pedestrian instances, it has more detection errors. This can be verified by another TUD-Brussels benchmark as shown in Fig. 6. One can observe that our approach can successfully detect most of the pedestrian instances, especially for the far-scale pedestrian instances.

References

[1] Z. Cai, Q. Fan, R. Feris, , and N. Vasconcelos. A unified multi-scale deep convolutional neural network for fast object detection. In *ECCV*, 2016.

[2] J. Caicedo and S. Lazebnik. Active object localization with deep reinforcement learning. In *ICCV*, 2015.

[3] P. Dollar, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *BMVC*, 2010.

[4] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: A benchmark. In *CVPR*, 2009.

[5] P. Dollar, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: An evaluation of the state of the art. *IEEE TPAMI*, 34(4):743–61, 2012.

[6] R. Girshick. Fast R-CNN. In *ICCV*, 2015.

[7] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[8] W. Nam, P. Dollar, and J. Han. Local decorrelation for improved pedestrian detection. In *NIPS*, 2014.

[9] S. Paisitkriangkrai, C. Shen, and A. van den Hengel. Strengthening the effectiveness of pedestrian detection. In *ECCV*, 2014.

[10] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[11] Y. Tian, P. Luo, X. Wang, and X. Tang. Pedestrian detection aided by deep learning semantic tasks. In *CVPR*, 2015.

[12] C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *CVPR*, 2009.

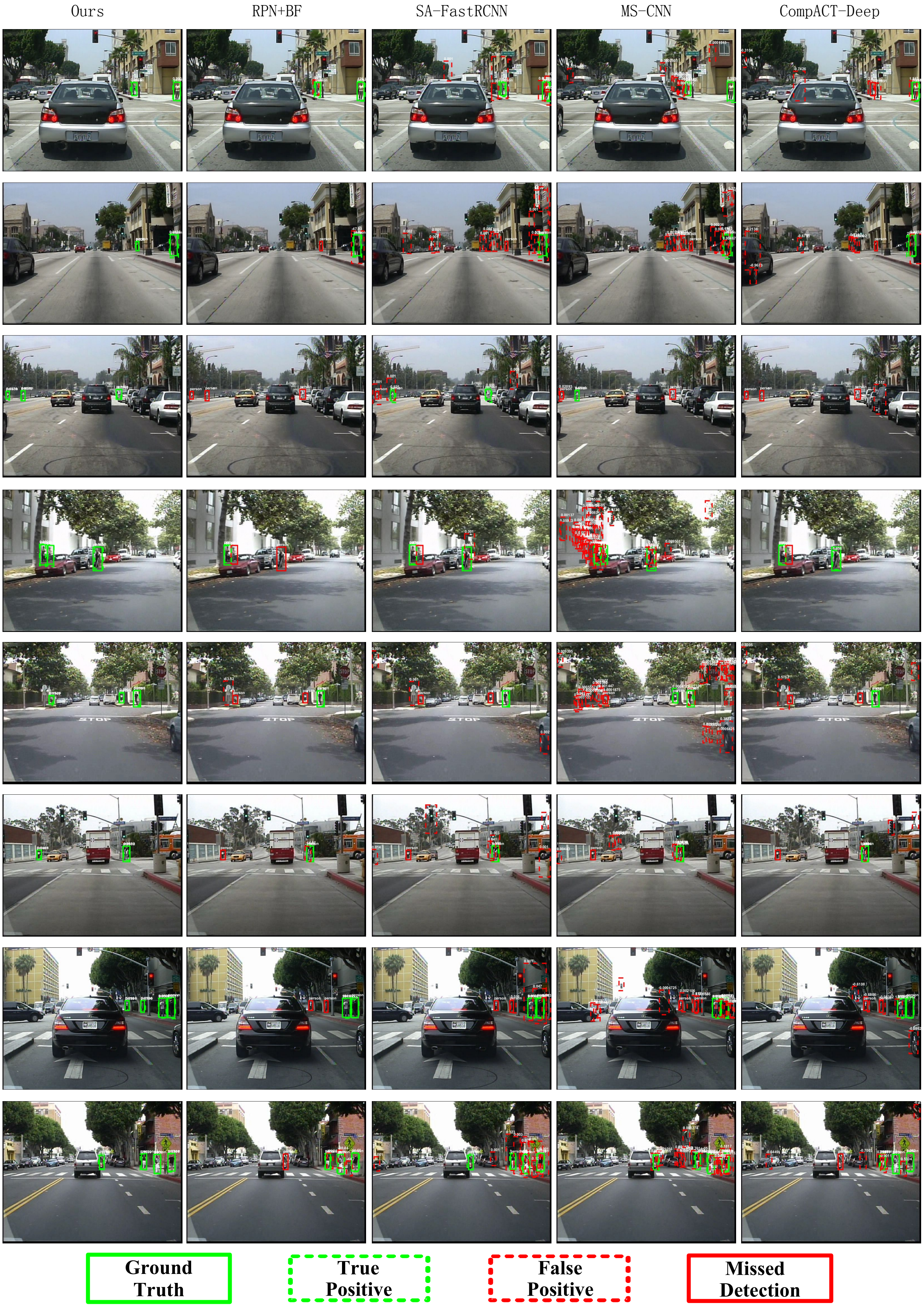


Fig. 4: Visual comparison of our detection results vs. those of the start-of-the-arts on the Caltech benchmark.

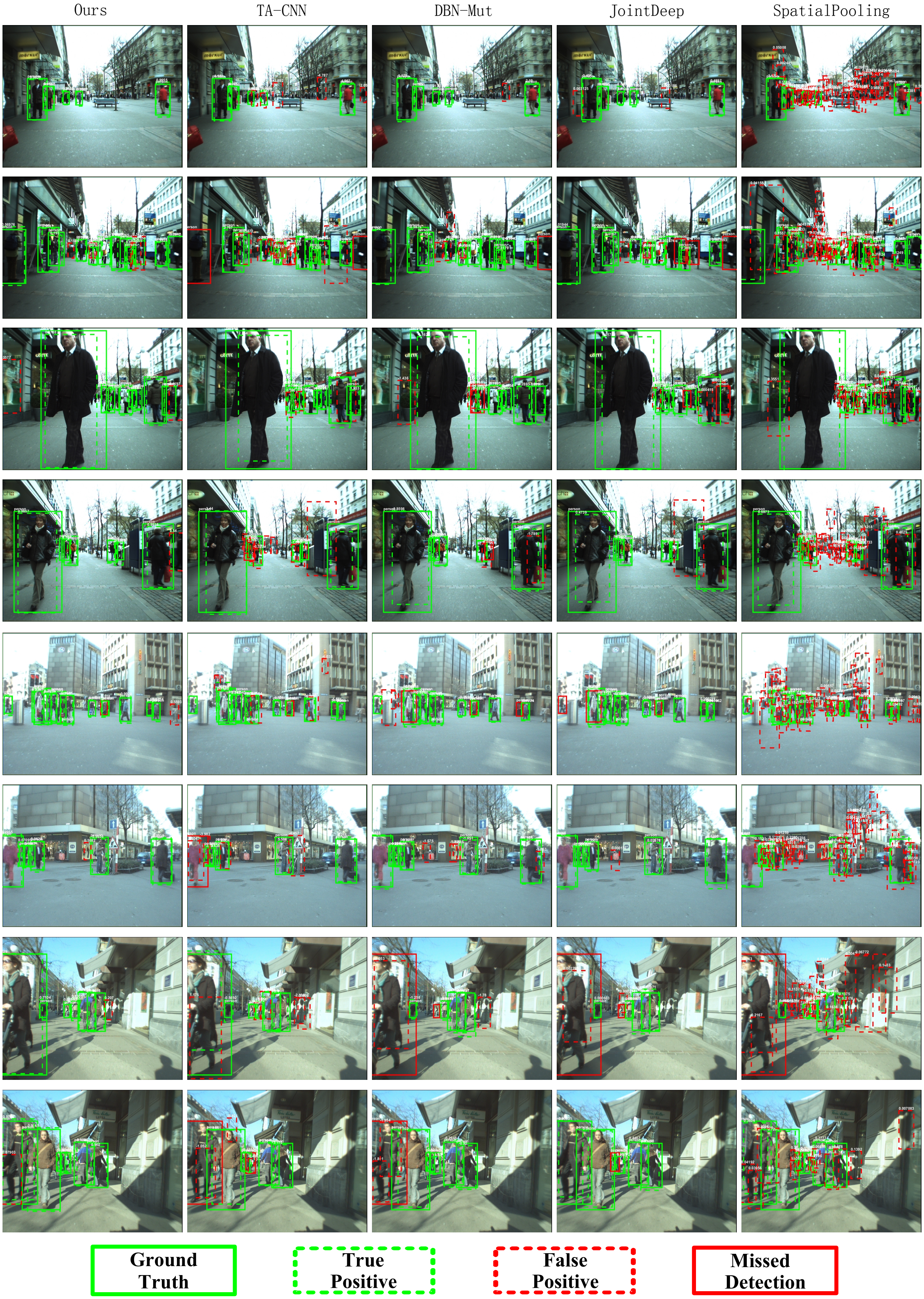


Fig. 5: Visual comparison of our detection results vs. those of the start-of-the-arts on the ETH benchmark.

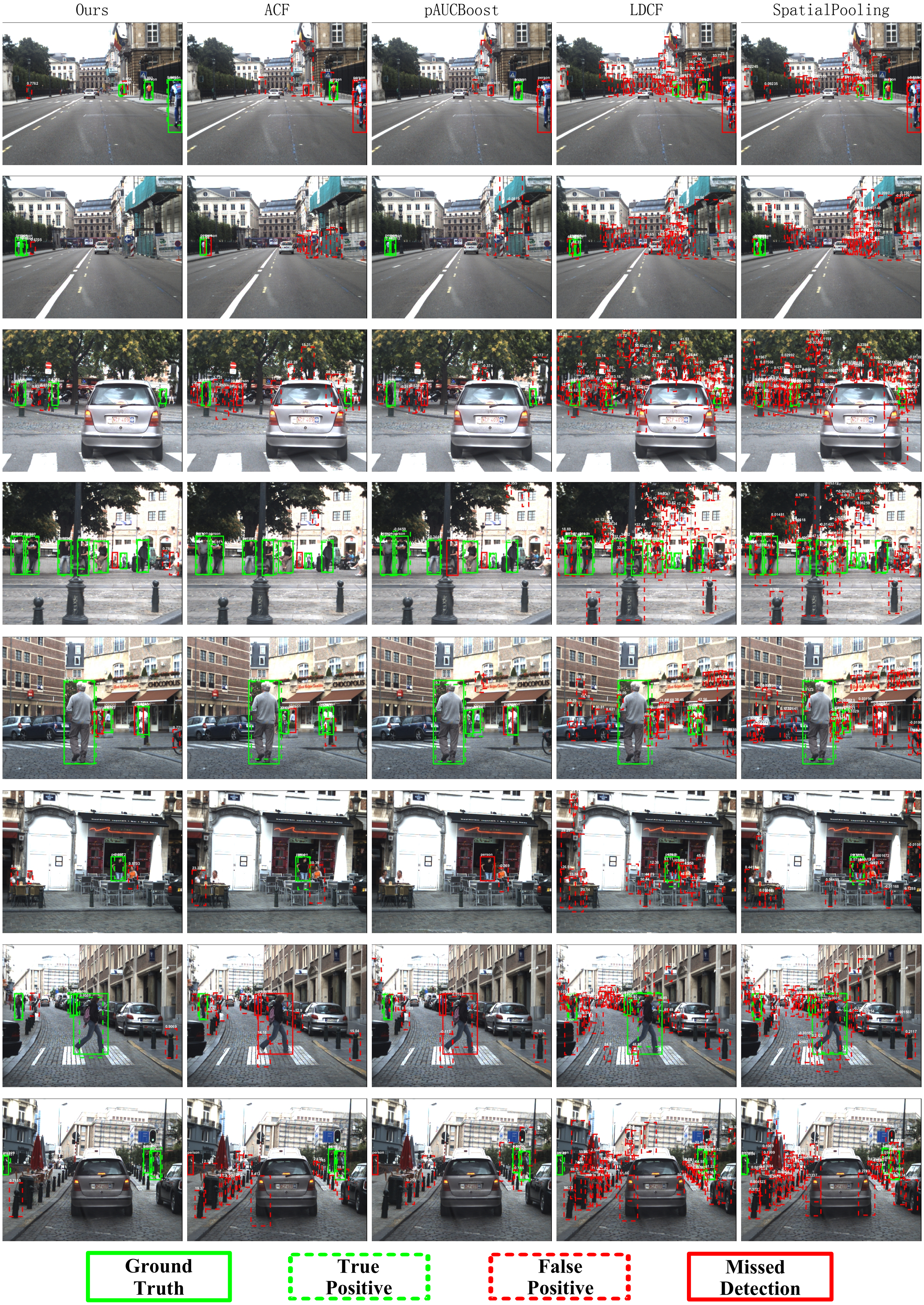


Fig. 6: Visual comparison of our detection results vs. those of the start-of-the-arts on the TUD-Brussels benchmark.