Online Domain Adaptation for Multi-Object Tracking

Adrien Gaidon
adrien.gaidon@xrce.xerox.com
Eleonora Vig
eleonora.vig@xrce.xerox.com

Computer Vision Group
Xerox Research Centre Europe
Meylan, France

Abstract

Automatically detecting, labeling, and tracking objects in videos depends first and foremost on accurate category-level object detectors. These might, however, not always be available in practice, as acquiring high-quality large scale labeled training datasets is either too costly or impractical for all possible real-world application scenarios. A scalable solution consists in re-using object detectors pre-trained on generic datasets. This work is the first to investigate the problem of on-line domain adaptation of object detectors for causal multi-object tracking (MOT). We propose to alleviate the dataset bias by adapting detectors from category to instances, and back: (i) we jointly learn all target models by adapting them from the pre-trained one, and (ii) we also adapt the pre-trained model on-line. We introduce an on-line multi-task learning algorithm to efficiently share parameters and reduce drift, while gradually improving recall. Our approach is applicable to any linear object detector, and we evaluate both cheap "mini-Fisher Vectors" and expensive "off-the-shelf" ConvNet features. We quantitatively measure the benefit of our domain adaptation strategy on the KITTI tracking benchmark and on a new dataset (PASCAL-to-KITTI) we introduce to study the domain mismatch problem in MOT.

1 Introduction

Tracking-by-detection (TBD), the dominant paradigm for object tracking in monocular video streams, relies on the observation that an accurate appearance model is enough to reliably track an object in a video. State-of-the-art Multi-Object Tracking (MOT) algorithms [5, 7, 15, 18, 32, 35], which aim at automatically detecting and tracking objects of a known category, rely on the recent progress on object detection. Most MOT approaches, indeed, consist in finding the best way to associate detections to form tracks. They, therefore, directly rely on object detection performance. However, a high-quality detector might not always be available in practice. In particular, acquiring high-quality large scale labeled training datasets required to train modern detectors is either too costly or impractical for all possible real-world application scenarios.

In this paper, we investigate a scalable solution to this data acquisition issue: re-using object detectors pre-trained on generic datasets. We propose to alleviate the ensuing *dataset bias* problem [41] for causal MOT via *on-line domain adaptation of object detectors from category to instances, and back*. Previous works (*cf.* Section 2) investigated detector adaptation *or* on-line learning of appearance models, but not both *jointly*. Our approach can be interpreted as a generalization, where we show that doing the joint adaptation is key, and

Pages 3.1-3.13
DOI: https://dx.doi.org/10.5244/C.29.3

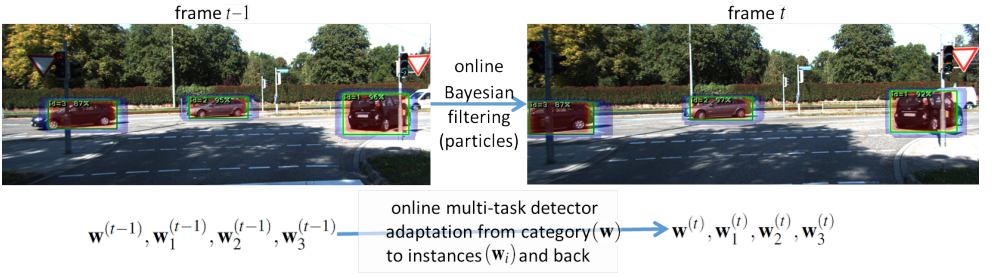

Figure 1: Online domain adaptation for MOT via Bayesian filtering coupled with multi-task adaptation of all detectors jointly.

doing no adaptation at all significantly degrades performance because of dataset bias. We propose a *convex multi-task learning objective* to *jointly adapt on-line* (i) all trackers from the pre-trained generic detector (*category-to-instance*), and (ii) the pre-trained category-level model from the trackers (*instances-to-category*). Our multi-task formulation enforces parameter sharing between all models to reduce model drift and robustly handle false alarms, while allowing for continuous domain adaptation to gradually decrease missed detections. We integrate our domain adaptation strategy in a novel motion model combining learned deterministic models with standard Bayesian filtering (*cf.* figure 1) inspired by the popular Bootstrap filter of Isard & Blake [22]. In particular, we leverage several techniques not widely used for MOT yet: (i) recent improvements in object detection based on generic candidate proposals [42, 52], (ii) large-displacement optical flow estimation [46], (iii) the Fisher Vector representation [33, 36], and (iv) ConvNet features for object detection [16]. In addition, we use a Sequential Monte Carlo (SMC) algorithm [8] to approximate the filtering distribution of our Markovian motion model of the latent target locations.

Section 2 reviews the related work. Section 3 describes our on-line multi-task learning of the trackers and domain adaptation of the category-level detector. Our motion model is described in Section 4. Finally, in Section 5, we report quantitative experimental results on the challenging KITTI tracking benchmark [14][1] and on a new *PASCAL-to-KITTI* dataset we introduce for the evaluation of domain adaptation in MOT.

2 Related Work

Following recent works [18, 29, 47], MOT approaches can be divided into three main categories: (i) Association-Based Tracking (ABT), (ii) Category-Free Tracking (CFT) and (iii) Category-to-Instance Tracking (CIT).

ABT approaches consist in building object tracks by associating detections precomputed over the whole video sequence. Recent works include the network flow approach of Pirsiavash *et al.* [35] (DP_MCF), global energy minimization [32] (CEM), two-granularity tracking [12], Hungarian matching [15], and the hybrid stochastic / deterministic approach of Collins and Carr [7]. These approaches rely heavily on the quality of the pre-trained detector, as tracks are formed only from pre-determined detections. Furthermore, they are generally applied off-line and are not always applicable to the streaming scenario.

CFT approaches, *e.g.,* [19, 21, 27, 30], can be considered as an extension of the category-free single target approaches to the MOT setting. In the single target case, the initial target

[1] http://www.cvlibs.net/datasets/kitti/eval_tracking.php

bounding box is given as input, and a specialized tracker is learned on-line, *e.g.,* via the Track-Learn-Detect approach [24]. The MOT extension consists in learning different trackers independently for each target automatically initialized by a generic pre-trained detector, while also handling the inter-target interactions. A strength of CFT methods is that they can track any type of object, as long as their location can be automatically initiated.

CIT approaches are similar to CFT ones in that they learn independent instance-specific trackers from automatic detections, but the target-specific models correspond to specializations of the generic category-level model. This requires the pre-trained detector and the target-specific trackers to have the same parametric form (*i.e.* same features and classifier) that work well both at the category and instance levels. This idea was recently introduced by Hall and Perona [18] to track pedestrians and faces by intersecting detections from a generic boosted cascade with a target-specific fine-tuned version of the cascade.

Our method is labeled **ODAMOT**, for "Online Domain Adaption for Multi-Object Tracking" (*cf.* figure 1), as it combines category-to-instance tracker adaptation with a novel (i) multi-task learning formulation (Section 3.2) and (ii) algorithm for on-line domain adaptation of the generic detector (Section 3.3). To the best of our knowledge, our approach is the first MOT approach to perform domain adaptation of the generic category-level detector.

Related to our work, Breitenstein *et al.* [5] track automatically detected pedestrians using a boosted classifier on low-level features to learn target-specific appearance models. Another related approach [28] uses a multi-task objective to learn jointly a generic object model and trackers. It, however, does not use a pre-trained detector, but initializes targets by hand for each video, assuming that instances form a crowd of slow-moving near duplicates. Other related works [40, 43, 44] include approaches for domain adaptation from generic to specific scene detectors for similar scenarios, although they do not learn trackers. Some other works [13, 37, 38] do not address MOT but nonetheless perform detector adaptation specifically for videos via other means. For instance, [13] puts forth a procedure to self-learn object detectors for unlabeled video streams by making use of a similar multi-task learning formulation. On the other hand, [38] relies on unsupervised multiple instance learning to collect online samples for incremental learning. Finally, adaptive tracking methods often adopt selective update strategies to avoid drift, for instance by integrating unlabeled data in the model in a semi-supervised manner [17].

3 Online adaptation from Category to Instances, and back

3.1 Generic object detection

Object proposals. Current state-of-the-art object detectors (*e.g.,* [6, 16, 42]) avoid exhaustive sliding window searches. Instead, they use a limited set of category-agnostic object location proposals, generated using general properties of objects (*e.g.,* contours), and overlapping most of the objects visible in an image. Although prevalent in detection, object proposals have not found their way into multi-object tracking yet. Nevertheless, the advantages of employing object proposals in MOT are apparent. Since proposals are category- and target-agnostic, we can reuse feature computations across all detectors (for any target and category). The speed-up is all the more apparent when many targets (of possibly different categories) must be tracked. In addition, object proposals seem well-suited for domain adaptation. Since object proposal methods rely on generic properties of objects, such as edge and contour density, they are, indeed, inherently agnostic to the data source. We here adopt the

Edge Boxes of Zitnick and Dollar [52], as they yield a good efficiency / accuracy trade-off (*cf.* [20] for an extensive review and evaluation of existing proposal methods). We extract around 4000 object proposals per frame.

Visual features. To represent candidate proposals, we explore the two most common image representations in current state-of-the-art object detectors with proposals: Fisher Vectors (FV) [6] and features from pre-trained Convolutional Networks [16]. In addition to being good representations for object detection, they are efficient for both image classification [25, 36] and retrieval [2, 23]. This highlights their potential for both category-level and instance-level appearance modeling. Our FV implementation follows [6]. We differ, however, by using only a *single* Gaussian FV (which we call "mini-FV"), a way to drastically reduce FV dimensionality (to 2176 in our case), while maintaining acceptable performance, as shown for retrieval by [34]. Regarding the ConvNet features, we follow R-CNN [16], except that we replace the standard AlexNet FC7 features with the smaller 1024-dimensional features from the penultimate layer of the more memory-efficient GoogLeNet convolutional network [39]. Higher-dimensional representations generally yield higher recognition performance, but at a prohibitive cost in terms of both speed and memory. The problem is further exacerbated in MOT, where per-target signatures need to be persistently stored for re-identification. We found in our experiments that the aforementioned features offer a good efficiency and accuracy trade-off, making them suitable for MOT. To the best of our knowledge, our method is the first application of FV or ConvNet features for MOT.

Linear object detector. We rank object proposals with a category-specific linear classifier parameterized by a vector $\mathbf{w} \in \mathbb{R}^d$. This classifier returns the probability that a candidate window $\mathbf{x}$, represented by a feature vector $\phi_t(\mathbf{x}) \in \mathbb{R}^d$, contains an object of the category of interest in frame $\mathbf{z}_t$ at time t by $P(\mathbf{x}|\mathbf{z}_t;\mathbf{w}) = \left(1+e^{-\mathbf{w}^T\phi_t(\mathbf{x})}\right)^{-1}$. In our experiments, we estimate the model $\mathbf{w}$ via logistic regression, a regularized empirical risk minimization algorithm based on the logistic loss $\ell_t(\mathbf{x},y,\mathbf{w}) = \log\left(1+\exp\left(-y\mathbf{w}^T\phi_t(\mathbf{x})\right)\right)$, as this gives calibrated probabilities and has useful properties for on-line optimization [3].

3.2 Adaptation from category to instances: multi-task tracking

Tracker warm-starting. The first category-to-instance adaptation happens at the creation of a new track. In addition to initializing the target location from a top detection, in frame t_0, we *warm-start* the optimization of the target-specific appearance model $\mathbf{w}_i^{(t_0)}$ from the category-level one $\mathbf{w}^{(t_0)}$. Warm-starting allows to start the target optimization close to an already good solution, as it was used to detect the initial target location. This yields two positive effects: faster convergence and stronger regularization. Therefore, warm-starting effectively mitigates the lack of training data due to the causal nature of our tracker, where we learn models from a single frame at a time. Note that warm-starting is often not possible in common MOT approaches, which generally rely on incompatible features and classifiers (*e.g.,* HOG+SVM and boosted cascades on low-level features [5]).

Multi-task regularization. Our second adaptation from category to instances consists in *updating all target models jointly* using multi-task learning. This allows all targets to share features, reflecting the fact that they belong to the same object category. Let N_t be the number of object instances tracked at time t. Each target $i = 1,\cdots,N_t$ has a location $\hat{\mathbf{x}}_i^{(t)}$ predicted by its associated motion model in frame t (*cf.* Section 4), and a learned appearance model $\mathbf{w}_i^{(t-1)}$. The goal is to update this appearance model $\mathbf{w}_i^{(t-1)} \rightarrow \mathbf{w}_i^{(t)}$ with the new

data from frame t by using the predicted location. Let $\{\mathbf{x}_{i,k}, k = 1, \cdots, n_i\}$ be the n_i training samples of object i in frame t, where $\hat{\mathbf{x}}_i^{(t)}$ is considered as positive, and negative windows are sampled according to the common "no teleportation and no cloning" assumption on each target individually [24]. Let $\mathbf{W}^{(t)} = \{\mathbf{w}_1^{(t)}, \ldots, \mathbf{w}_{N_t}^{(t)}\}$ be the stacked target models, and $(\mathbf{X}^{(t)}, \mathbf{y}^{(t)})$ be the training samples and labels mined for all targets in frame t. Updating all appearance models jointly amounts to minimizing the following regularized empirical risk:

$$\mathbf{W}^{(t)} = \arg\min_{\mathbf{W}} L_t(\mathbf{X}^{(t)}, \mathbf{y}^{(t)}, \mathbf{W}) + \lambda \Omega_t(\mathbf{W}) \tag{1}$$

where the loss L_t and multi-task regularization term Ω_t are defined as:

$$L_t(\mathbf{X}^{(t)}, \mathbf{y}^{(t)}, \mathbf{W}) = \frac{1}{N_t} \sum_{i=1}^{N_t} \frac{1}{n_i} \sum_{k=1}^{n_i} \ell_t(\mathbf{x}_{i,k}\ ,\ y_{i,k}\ ,\ \mathbf{w}_i)\ , \quad \Omega_t(\mathbf{W}) = \frac{1}{2N_t} \sum_{i=1}^{N_t} \|\mathbf{w}_i - \bar{\mathbf{w}}^{(t-1)}\|_2^2, \tag{2}$$

where $\bar{\mathbf{w}}^{(t-1)}$ is the (running) mean of all previous instance models, which comprises all past values of the models of currently tracked or now lost targets (this is equivalent to summing all pairwise comparisons between target-specific models). This formulation is closely related to the mean-regularized multi-task learning formulation of Evgeniou and Pontil [10], with the difference that it is designed for on-line learning in streaming scenarios.

3.3 Online adaptation from instances back to category

Our joint multi-task adaptation of the target-specific models allows to track more reliably while limiting model drift and false alarms thanks to feature sharing and joint regularization. In addition, we hypothesize that maintaining and adapting the generic pre-trained category-level detector should allow to lower the miss-rate by continuously specializing the global appearance model to the specific video stream, which might be non-stationary and significantly different than the off-line pre-training data. In fact, one can observe that our regularization term (Eq. 2) already provides a theoretical justification to using the running average $\bar{\mathbf{w}}^{(t)}$ as a single category-level detector. Indeed, once the detectors $\mathbf{w}_i$ are updated in frame t, a new scene-adapted detector is readily available as:

$$\bar{\mathbf{w}}^{(t)} = \frac{1}{\bar{N}_{t-1} + N_t} \left(\bar{N}_{t-1} \bar{\mathbf{w}}^{(t-1)} + \sum_{i=1}^{N_t} \mathbf{w}_i^{(t)} \right)\ , \quad \text{where } \bar{N}_{t-1} = \sum_{j=1}^{t-1} N_j. \tag{3}$$

As we use linear classifiers, this multi-task learning is akin to a "fusion" of exemplar-based models (*e.g.,* Exemplar-SVMs [30]). A major improvement is that our models are learned *jointly* and *adapt continuously* to both the data stream and other exemplars. This adaptation allows to limit drift of the category model. There is, indeed, an "inertia" in the update due to the warm-starting of the trackers from the generic model. Furthermore, as the adapted model corresponds to a (potentially long) running average, the contribution of false alarms to the model should be limited, as false alarms are more likely to be tracked for less time thanks to our multi-task penalization. We optimize the learning objective in Eq. 1 using Stochastic Gradient Descent (SGD) with constant learning rate of 10^{-5}.

4 Causal Multi-Object Tracking-by-Detection

In this section, we describe our causal (*i.e.* on-line) MOT framework to track a variable number of objects belonging to a category known in advance (*e.g.,* cars) in a monocular

video stream coming from a fixed or moving camera. Algorithm 1 provides a high-level pseudo-code description of ODAMOT.

4.1 Bayesian motion model

Let $\mathbf{z}_t$ be the random variable representing our observation, a frame of the video stream at time t. Let $\mathbf{x}_t = (x_t, y_t, w_t, h_t)^T$ be the random variable representing the latent location (a bounding box) of the object i in frame $\mathbf{z}_t$. We model the evolution of the object's location using a dynamical system specific to target i characterized by the following Bayesian model. The initial distribution is $\mathbf{x}_{t_0} \sim \mathcal{N}(\hat{\mathbf{x}}_{t_0}, \Sigma_{t_0})$, where $\hat{\mathbf{x}}_{t_0}$ is the target's initial location in a frame $t_0 < t$, which corresponds to a detection of the generic detector $\mathbf{w}^{(t_0)}$ in frame t_0, and Σ_{t_0} is the initial covariance modeling the uncertainty on this initial location.

Our Markovian transition model is $\mathbf{x}_t = \mathbf{x}_{t-1} + \mathbf{v}_{t-1}(\mathbf{x}_{t-1})\Delta t + \varepsilon_t$, where ε_t is Gaussian noise and $\mathbf{v}_{t-1}(\mathbf{x}_{t-1})$ is the instantaneous target velocity estimated by median filtering in the dense large-displacement optical flow field computed using the deep flow algorithm [16]. Note that this differs from the standard constant velocity assumption [5], which is not suitable for fast moving objects and moving cameras.

Our observation model is characterized by: $P(\mathbf{z}_t|\mathbf{x}_t) \propto P(\mathbf{x}_t|\mathbf{z}_t; \mathbf{w}_i^{(t-1)}) \cdot P(\mathbf{x}_t|\mathbf{z}_t; \mathbf{w}^{(t-1)})$. We define the likelihood to be proportional to the probability that the window has both the appearance of target i, modeled by the target-specific appearance model $\mathbf{w}_i^{(t-1)}$, and of the category, modeled by the category-level appearance model $\mathbf{w}^{(t-1)}$, assuming uniform priors over the frames $\mathbf{z}_t$ and locations $\mathbf{x}_t$. The appearance models $\mathbf{w}_i^{(t-1)}$ and $\mathbf{w}^{(t-1)}$ are obtained at the previous time step $t-1$, as described in the previous Section 3.

Algorithm 1 Pseudo-code overview of our approach. Refer to the main text for details.

Input: generic detector $\mathbf{w}$, video stream
Output: adapted detector $\bar{\mathbf{w}}^{(t_{\text{end}})}$, tracks list $\mathcal{W}$
Initialization: $\bar{\mathbf{w}}^{(t_0)} = \mathbf{w}$, $\mathcal{W} = \emptyset$
while video stream is not finished **do**
  for each target i in $\mathcal{W}$ **do**
    Update i's location with a Bayesian motion model (*cf.* Sec. 4.1)
  end for
  Detect new targets not in $\mathcal{W}$ with $\bar{\mathbf{w}}^{(t-1)}$ in frame t and add them to $\mathcal{W}$
  Merge overlapping tracks in $\mathcal{W}$
  for each target i in $\mathcal{W}$ **do**
    if i is a new target **then**
      Learn initial detector $\mathbf{w}_i^{(t-1)}$ warm-started from $\bar{\mathbf{w}}^{(t-1)}$ (*cf.* Sec. 3.2)
    end if
    Run the detector $\mathbf{w}_i^{(t-1)}$
    if object i is lost **then**
      Remove i from $\mathcal{W}$
    else
      Get $\{(\mathbf{x}_{i,k}^{(t)}, y_{i,k}^{(t)}),\ k = 1 : n_i\}$
      Update detector $\mathbf{w}_i^{(t)}$ (*cf.* Sec. 3.2)
    end if
  end for
  Update generic detector $\bar{\mathbf{w}}^{(t)}$ (Eq. 3)
end while

4.2 Sequential Monte Carlo approximation of the filtering distribution

In order to use this model, we need to recursively estimate the filtering distribution $P(\mathbf{x}_t|\mathbf{z}_{1:t})$. Following the standard practice, we approximate the filtering distribution using Sequential Monte Carlo sampling. We use Sequential Importance Sampling [8] to compute our filtering distribution approximation recursively over time using N particles $\mathbf{x}_{t-1}^{(p)}$, $p = 1, \cdots, N$. In practice, we found that $N = 100$ particles provided a good trade-off between exploration, exploitation, and computational efficiency. We use $\sigma_0 = 5\%$ as fixed initial relative noise variance, and scale it by the inverse of the number of successful updates for the target.

To predict precisely the location of an object at each time instant from our estimate of the filtering distribution, we use the expectation of the latent variable $\mathbf{x}_t$, as it can be easily estimated as the weighted average of the particles: $\hat{\mathbf{x}}_t = \sum_{p=1}^{N} w_t^{(p)} \mathbf{x}_t^{(p)}$. We observed that using the expectation yields good results, as the distribution tends to have a limited variance due to the specialization of the per-target appearance models.

4.3 Inter-target reasoning

Our *identification* strategy to MOT differs from standard global data association methods, as it relies on the detector(s) to limit ID switches and fragmentation. We also rely mostly on appearance to handle occlusions, as this sort of invariance is a goal of object detection. However, as our detectors might suffer from dataset bias, we apply further post-processing to deal with occlusions. In particular, we temporarily lose a target, *i.e.* make no location prediction, and try to reinitialize its location in subsequent frames using its specialized detector. If the reinitialization fails consecutively for more than T frames, we terminate the target. Note that later re-identification can be performed by trying to reinitialize at bigger regular time intervals. We also assume that two overlapping tracks correspond to the same target if the location predictions intersect by more than 30% for more than T consecutive frames. In this case, the tracker with the lower score is terminated. In our experiments, we used the very short $T = 3$ interval in order to deal with potentially fast moving objects (cars) filmed from a fast moving platform (a car-mounted camera). Note that our main contribution (online joint domain adaptation of all appearance models) is orthogonal to the numerous occlusion reasoning and data association improvements to MOT (*e.g.,* [48]), which could be combined with our method for improved performance.

5 Experiments

We evaluate our MOT algorithm on the challenging KITTI car tracking benchmark [14]. As this challenge discourages multiple submissions on its evaluation server, we evaluate only the best detector we can train on related training data using state-of-the-art ConvNet features. We then perform an ablative analysis and quantitatively demonstrate the benefit of our domain adaptation strategy on the new PASCAL-to-KITTI (P2K) dataset, which we describe below. In our experiments, we follow the KITTI evaluation protocol by using the CLEAR MOT metrics [4] and code[2] – including MOT Accuracy (MOTA), MOT Precision (MOTP), Fragmentation (FRG), and ID Switches (IDS) – complemented by the Mostly Tracked (MT) and Mostly Lost (ML) ratios, Precision (Prec), Recall (Rec), and False Alarm Rate (FAR).

5.1 KITTI tracking benchmark

The KITTI object tracking benchmark [14][3] consists of 21 training and 29 test videos recorded using cameras mounted on a moving vehicle. This is a challenging dataset designed to investigate how computer vision algorithms perform on real-world data typically found in robotics and autonomous driving applications. We train an R-CNN-like car detector on the 21 training videos for which ground truth tracks are available (*cf.* Section 3.1 for more details). As in [16], for increased performance, we perform domain-specific fine-tuning of the network on the KITTI training set prior to training the detector.

[2] http://kitti.is.tue.mpg.de/kitti/devkit_tracking.zip
[3] http://www.cvlibs.net/datasets/kitti/eval_tracking.php

method	MOTA↑	MOTP↑	MT↑	ML↓	Rec.↑	Prec.↑	FAR↓	IDS↓	FRG↓
DP_MCF† [35]	36.62%	78.49%	11.13%	39.18%	46.19%	96.64%	5.03%	2738	3240
HM [26]	42.22%	78.42%	7.77%	41.92%	43.83%	96.80%	4.54%	**12**	577
MCF† [51]	44.28%	78.32%	10.98%	39.94%	45.87%	**97.03%**	**4.40%**	23	590
TBD† [15]	52.44%	78.47%	13.87%	34.30%	55.28%	95.51%	8.16%	33	538
DCO† [1]	35.17%	74.50%	10.67%	33.69%	50.74%	77.56%	46.13%	223	622
CEM† [32]	48.23%	77.26%	14.48%	33.99%	54.52%	90.47%	18.04%	125	398
RMOT [48]	49.87%	75.33%	15.24%	33.54%	56.39%	90.16%	19.35%	51	**385**
DCO_X*† [31]	**62.76%**	**78.96%**	26.22%	15.40%	77.08%	86.47%	39.17%	326	984
RMOT* [48]	60.46%	75.57%	**26.98%**	**11.13%**	**79.19%**	82.68%	54.02%	216	742
ODAMOT	57.06%	75.45%	16.77%	18.75%	64.76%	92.04%	17.93%	404	1304

Table 1: KITTI Car tracking benchmark results. Metrics with ↑ (resp. ↓) should be increasing (resp. decreasing). Methods with * use Regionlets [45]. Those with † are offline, the others online.

Results. Table 1 summarizes the tracking accuracy of our method (**ODAMOT**) and other state-of-the-art approaches on the 29 test sequences whose ground truth annotations are not public. We compare against all the results on this benchmark where the methodology has been described in the literature. Our algorithm ranks third in terms of MOTA, which summarizes multiple aspects of tracking performance. An explanation for the performance gap lies in the adoption of more sophisticated inter-target and occlusion reasoning by competing methods [31, 48]. RMOT [48], for instance, performs data association and leverages motion context in addition to Bayesian filtering. Indeed, the rather simple inter-target reasoning of ODAMOT explains the high number of ID switches and fragmentations, which are detrimental to performance.

5.2 PASCAL-to-KITTI: domain adaptation in MOT

PASCAL-to-KITTI (P2K) dataset. Domain adaptation of appearance models for MOT has remained largely unaddressed until now. To allow the systematic study of this problem, we assembled a new MOT dataset called *PASCAL-to-KITTI* (P2K). The training set (the *source domain*) consists of the training images of the standard Pascal VOC 2007 detection challenge [9]. As this dataset is general-purpose, it is reasonable to expect it to yield pre-trained appearance models that are likely to transfer to more specific tasks or domains, at least to a certain extent. The test set (the *target domain*) consists of the 21 training videos of the KITTI tracking challenge. Fig. 2 highlights some striking differences between source and target domains and illustrates the difficulty of transfer.

Detector pre-training. The pre-training of the detector is performed off-line via batch logistic regression (using liblinear [11]) with hard negative mining as described in [6]. Our mini-FV model yields 40% Average Precision (AP) for car detection on the Pascal test set, which is 18% below the results of [6] for a fraction of the cost. Our R-CNN-like detector achieves 60% AP on the Pascal test images, which is on par with the results reported by [16] (58,9% AP for R-CNN fc_7). On three validation videos of the KITTI training set this detector gives 42% AP, which hints at the domain gap between Pascal and KITTI.

Baselines. We compare **ODAMOT** to the related MOT algorithms from Section 2: off-line Association Based Tracking (ABT) type methods (DP_MCF [35], and G_TBD [15], for which code is available), and our implementation of an on-line Category-Free Tracker (CFT) and an on-line Category-to-Instance Tracker (CIT). CFT corresponds to the TLD approach of [24], and differs from ODAMOT in that it does not warm-start the target models from a pre-trained detector, performs no multi-task regularization (target models are independent), and no online adaptation of the pre-trained detector. CIT is inspired by [18]. It is similar to

Figure 2: Images from the Pascal VOC 2007 (top) and KITTI Tracking (bottom) benchmarks. Note the striking differences in visual appearance between the two datasets.

method	MOTA↑	MOTP↑	MT↑	ML↓	Rec.↑	Prec.↑	FAR↓	IDS↓	FRG↓
DP_MCF† [35]	1.9%	74.0%	0.0%	98.6%	2.1%	**92.9%**	**0.5%**	**6**	**25**
G_TBD† [15]	8.4%	71.2%	0.2%	86.9%	11.1%	81.0%	8.1%	13	174
CFT [24]	16.6%	**74.9%**	**1.1%**	71.8%	19.2%	88.0%	8.1%	68	254
CIT [18]	18.2%	73.9%	**1.1%**	67.3%	21.8%	86.1%	10.9%	40	193
ODAMOT	**19.7%**	74.5%	**1.1%**	**64.6%**	**23.5%**	86.4%	11.5%	55	232
DP_MCF† [35]	12.0%	68.5%	0.1%	80.2%	14.6%	**85.5%**	**7.7%**	**84**	**327**
G_TBD† [15]	17.5%	68.0%	0.9%	59.2%	30.0%	71.3%	37.6%	115	528
CFT [24]	17.6%	66.7%	1.8%	45.7%	33.5%	69.1%	47.2%	238	592
CIT [18]	22.8%	68.5%	**1.9%**	**43.4%**	33.9%	76.5%	32.6%	380	809
ODAMOT	**23.6%**	**68.7%**	1.8%	43.6%	**34.2%**	77.5%	31.1%	376	784

Table 2: MOT results on the P2K domain adaptation dataset. The upper block contains results for the "mini-Fisher Vector" detector, while the lower block shows results for the more powerful R-CNN-like detector. Methods with † are offline, the others are online.

CFT, except that the trackers are warm-started from the pre-trained category-level detector.

Results. As shown in Table 2, ODAMOT outperforms all related methods that rely on the same general-purpose detectors trained on Pascal VOC 2007. As expected, unrelated training data strongly degrades MOT performance. Nevertheless, our results show that domain adaptation partly mitigates this problem. By improving recall and maintaining high precision, ODAMOT allows to track more targets than the related CFT and CIT online methods, which do not perform the *joint* adaptation of category and instance models. This multi-task online adaptation allows to gradually discover and track more targets while limiting model drift, although at the cost of moderately increased identity switches and track fragmentation. On the other hand, off-line ABT-type methods (DP_MCF [35] and G_TBD [15]) suffer greatly from the low quality of the pre-trained detector, especially when using "mini-FV" (upper block of Table 2). As expected, more powerful state-of-the-art ConvNet features improve all results (from 19.7% to 23.6% MOTA for ODAMOT) but surprisingly not substantially. This confirms the difficulty of domain transfer, in particular due to the overfitting tendency of deep nets, which is problematic when faced with dataset bias. Note that this might be partly alleviated by using features from earlier layers, which might transfer better [49]. Our results also hint at the transferability potential of the weaker mini-FV features, where ODAMOT improves more significantly the MOT performance *w.r.t.* the baselines.

Failure cases. Our method tends to suffer from two main problems. The first is tied to the failure modes of the detector (missed detections and false alarms), and is common to all TBD methods. Although our adaptation improves, the multi-task objective tends to favor conservative updates to prevent drift, similarly to self-paced learning approaches like [40].

Second, our tracks contain many ID switches and are generally fragmented. This hints at a lack of specialization of the appearance models, which could be addressed by designing features that can better represent instances. Another solution to this issue would consist in complementing our method with advanced inter-target and occlusion reasoning, *e.g.,* [43].

6 Conclusion

We address the question of how to re-use object detectors pre-trained on general-purpose datasets for causal multi-object tracking, when strongly related training data is not available. To overcome the dataset bias present in these generic detectors, we propose the joint online adaptation of category- and target-level detectors. Our multi-task adaptation from category-to-instances and back allows to improve overall MOT accuracy by increasing recall while maintaining high precision and limiting model drift in challenging real-world videos.

References

[1] A. Andriyenko, K. Schindler, and S. Roth. Discrete-continuous optimization for multi-target tracking. In *CVPR*, 2012. 8

[2] A. Babenko, A. Slesarev, A. Chigorin, and V. Lempitsky. Neural codes for image retrieval. In *ECCV*, 2014. 4

[3] F. Bach and E. Moulines. Non-strongly-convex smooth stochastic approximation with convergence rate O (1 / n). In *NIPS*, 2013. 4

[4] K. Bernardin and R. Stiefelhagen. Evaluating Multiple Object Tracking Performance: The CLEAR MOT Metrics. *EURASIP Journal on Image and Video Processing*, 2008. 7

[5] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online Multi-Person Tracking-by-Detection from a Single, Uncalibrated Camera. *PAMI*, 2011. 1, 3, 4, 6

[6] R.G. Cinbis, J. Verbeek, and C. Schmid. Segmentation driven object detection with Fisher vectors. In *ICCV*, 2013. 3, 4, 8

[7] R.T. Collins and P. Carr. Hybrid Stochastic/Deterministic Optimization for Tracking Sports Players and Pedestrians. In *ECCV*, 2014. 1, 2

[8] A. Doucet, S. Godsill, and C. Andrieu. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing*, 2000. 2, 6

[9] M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 8

[10] T. Evgeniou and M. Pontil. Regularized multi–task learning. In *SIGKDD*, 2004. 5

[11] R.E. Fan, K.W. Chang, C.J. Hsieh, X.R. Wang, and C.J. Lin. Liblinear: A library for large linear classification. *JMLR*, 2008. 8

[12] K. Fragkiadaki, W. Zhang, G. Zhang, and J. Shi. Two-granularity tracking: mediating trajectory and detection graphs for tracking under occlusions. In *ECCV*, 2012. 2

[13] A. Gaidon, G. Zen, and J.A. Rodriguez-Serrano. Self-Learning Camera: Autonomous adaptation of object detectors to unlabeled video streams. Technical report, 2014. arXiv:1406.4296. 3

[14] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *CVPR*, 2012. 2, 7

[15] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun. 3D Traffic Scene Understanding from Movable Platforms. *PAMI*, 2014. 1, 2, 8, 9

[16] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2, 3, 4, 7, 8

[17] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *ECCV*, 2008. 3

[18] D. Hall and P. Perona. Online, Real-Time Tracking Using a Category-to-Individual Detector. In *ECCV*, 2014. 1, 2, 3, 8, 9

[19] S. Hare, A. Saffari, and P.H.S. Torr. Struck : Structured Output Tracking with Kernels. In *ICCV*, 2011. 2

[20] J. Hosang, R. Benenson, and B. Schiele. How good are detection proposals, really? In *BMVC*, 2014. 4

[21] Y. Hua, K. Alahari, and C. Schmid. Occlusion and motion reasoning for long-term tracking. In *ECCV*, 2014. 2

[22] M. Isard and A. Blake. Condensation-conditional density propagation for visual tracking. *IJCV*, 1998. 2

[23] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local image descriptors into compact codes. *PAMI*, 2012. 4

[24] Z. Kalal, K. Mikolajczyk, and J. Matas. Tracking-Learning-Detection. *PAMI*, 2011. 3, 5, 8, 9

[25] A. Krizhevsky, I. Sutskever, and G.E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS*, 2012. 4

[26] H.W. Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 8

[27] X. Li, W. Hu, C. Shen, Z. Zhang, and A. Dick. A Survey of Appearance Models in Visual Object Tracking. *ACM Transactions on Intelligent Systems and Technology*, 2013. 2

[28] W. Luo, T.K. Kim, B. Stenger, X. Zhao, and R. Cipolla. Bi-label Propagation for Generic Multiple Object Tracking. In *CVPR*, 2014. 3

[29] W. Luo, X. Zhao, and T.K. Kim. Multiple object tracking: A review. *arXiv preprint arXiv:1409.7618*, 2014. 2

[30] T. Malisiewicz, A. Gupta, and A.A. Efros. Ensemble of Exemplar-SVMs for object detection and beyond. In *ICCV*, 2011. 5

[31] A. Milan, K. Schindler, and S. Roth. Detection- and trajectory-level exclusion in multiple object tracking. In *CVPR*, 2013. 8

[32] A. Milan, S. Roth, and K. Schindler. Continuous Energy Minimization for Multi-Target Tracking. *PAMI*, 2014. 1, 2, 8

[33] F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, 2007. 2

[34] F. Perronnin, Y. Liu, J. Sánchez, and H. Poirier. Large-scale image retrieval with compressed fisher vectors. In *CVPR*, 2010. 4

[35] H. Pirsiavash, D. Ramanan, and C. Fowlkes. Globally-optimal greedy algorithms for tracking a variable number of objects. In *CVPR*, 2011. 1, 2, 8, 9

[36] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek. Image Classification with the Fisher Vector: Theory and Practice. *IJCV*, 2013. 2, 4

[37] P. Sharma and R. Nevatia. Efficient detector adaptation for object detection in a video. In *CVPR*, 2013. 3

[38] P. Sharma, C. Huang, and R. Nevatia. Unsupervised incremental learning for improved object detection in a video. In *CVPR*, 2012. 3

[39] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 4

[40] K. Tang, V. Ramanathan, L. Fei-Fei, and D. Koller. Shifting weights: Adapting object detectors from image to video. In *NIPS*, 2012. 3, 9

[41] A. Torralba and A.A. Efros. Unbiased look at dataset bias. In *CVPR*, 2011. 1

[42] K.E.A. van de Sande, J.R.R. Uijlings, T. Gevers, and A.W.M. Smeulders. Segmentation as selective search for object recognition. In *ICCV*, 2011. 2, 3

[43] M. Wang, W. Li, and X. Wang. Transferring a generic pedestrian detector towards specific scenes. In *CVPR*, 2012. 3

[44] X. Wang, M. Wang, and W. Li. Scene-specific pedestrian detection for static video surveillance. *PAMI*, 2013. 3

[45] X. Wang, M. Yang, S. Zhu, and Y. Lin. Regionlets for generic object detection. In *CVPR*, 2013. 8

[46] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. DeepFlow: Large displacement optical flow with deep matching. In *ICCV*, 2013. 2, 6

[47] B. Yang and R. Nevatia. Online learned discriminative part-based appearance models for multi-human tracking. In *ECCV*, 2012. 2

[48] J.H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon. Bayesian multi-object tracking using motion context from multiple objects. In *WACV*, 2015. 7, 8, 10

[49] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *NIPS*, 2014. 9

[50] L. Zhang and L. van der Maaten. Preserving structure in model-free tracking. *PAMI*, 2014. 2

[51] L. Zhang, Y. Li, and R. Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, 2008. 8

[52] C.L. Zitnick and P. Dollar. Edge Boxes: Locating Object Proposals from Edges. In *ECCV*, 2014. 2, 4