Temporal Coherence for Active Learning in Videos

Javad Zolfaghari Bengar[1,2] Abel Gonzalez-Garcia[1] Gabriel Villalonga[1,2]
Bogdan Raducanu[1,2] Hamed H. Aghdam[1] Mikhail Mozerov[1,2] Antonio M. López[1,2]
Joost van de Weijer[1,2]
Computer Vision Center (CVC)[1], Univ. Autònoma of Barcelona (UAB)[2]
{jzolfaghari,agonzalez,gvillalonga,bogdan,haghdam,mozerov,antonio,joost}@cvc.uab.es

Abstract

Autonomous driving systems require huge amounts of data to train. Manual annotation of this data is time-consuming and prohibitively expensive since it involves human resources. Therefore, active learning emerged as an alternative to ease this effort and to make data annotation more manageable. In this paper, we introduce a novel active learning approach for object detection in videos by exploiting temporal coherence. Our active learning criterion is based on the estimated number of errors in terms of false positives and false negatives. The detections obtained by the object detector are used to define the nodes of a graph and tracked forward and backward to temporally link the nodes. Minimizing an energy function defined on this graphical model provides estimates of both false positives and false negatives. Additionally, we introduce a synthetic video dataset, called SYNTHIA-AL, specially designed to evaluate active learning for video object detection in road scenes. Finally, we show that our approach outperforms active learning baselines tested on two datasets.

1. Introduction

For autonomous driving systems, the quality of object detection is of key importance. Its progress in recent years has been notable, partially due to the presence of large datasets [15, 61]. However, pushing detectors to further improve and finally be close to flawless, requires the collection of ever larger labeled datasets, which is both time and labor expensive. Active learning methods [49] tackle this problem by reducing the required annotation effort. The key idea behind active learning is that a machine learning model can achieve a satisfactory performance with a subset of the training samples if it is allowed to choose which samples to label. This contrasts with passive learning, where the data to be labeled is taken at random without taking into account the potential benefit of annotating each sample.

Active learning has been mainly investigated for the image classification task [24, 34, 14, 46, 35, 55, 8]. Only few works have investigated active learning for object detection, even though the problem of active learning is more pertinent for object detection than for image classification since the labelling effort also includes the more expensive annotation of the bounding box [29]. For instance, in [59, 53] the object detector is learned interactively in an incremental manner using a simple margin approach to select the most uncertain images. In [44], the active learning approach is based on a 'query-by-committee' strategy.

In this work we focus on active learning for object detection in videos. To the best of our knowledge, we are the first to consider this scenario. Object detection in videos has become of great interest ever since the introduction of the large-scale video object detection challenge ImageNet-VID [45]. The task has proven highly challenging due to phenomena such as detector flicker [43, 23], i.e. the drastic effects in the predicted outputs given by small changes in the images. This has spawn a multitude of video-specific approaches [26, 27, 63, 64, 54] that require comprehensive video annotation. However, exhaustively annotating all object instances in every frame is extremely costly. Possibly because of this, recent datasets for autonomous driving [61, 40] only offer a small subset of frames with object ground-truth annotations.

Video data has the inherent property of *temporal coherence*, i.e. nearby frames are expected to contain the same instances in nearby locations. This property can be exploited to identify frames in which the detector might have wrongly detected objects (there is no support in nearby frames) or frames in which the detector failed to detect an object (there is evidence of the object in the surrounding frames). These frames are expected to be more beneficial to annotate than others, leading to potentially more accurate models when used for training.

In this paper, we confirm that annotating those frames that contain detection errors leads to higher accuracy given a limited annotation budget. We consider two types of errors, false positives and false negatives, and show the ef-

fect of selecting either type. This exploratory experiment suggests a potentially powerful approach for active learning. Motivated by this, we develop a novel method to estimate detection errors in videos by exploiting the temporal coherence in the videos. We track detections forward and backward and define a graph on the detections that are temporally linked. Minimization of an energy function defined on this graphical model provides us with the detection of false positives and false negatives. These we subsequently use to select the frames to be annotated. In summary, the contributions of this paper are:

- We propose a new method for active learning in videos which exploits the temporal coherence.
- We propose a new synthetic dataset specially designed for active learning in road scene videos.
- Our proposed method outperforms several baseline methods both on synthetic and real video data.

2. Related Work

Active learning for object detection. A critical aspect for an active learner is represented by the strategy used to query the next sample to be labeled. Four main query frameworks exist, which rely mostly on heuristics: informativeness [58, 13, 17, 4], representativeness [46, 48], hybrid [22, 57], and performance-based [47, 16, 12, 56]. Among all these, informativeness-based approaches are the most successful ones. A comprehensive survey of these frameworks and a detailed discussion can be found in [49]. Active learning has been successfully applied to a series of traditional computer vision tasks, such as image classification [28, 24, 14] (including medical image classification [46] and scene classification [35]), visual question answering (VQA) [37], image retrieval [62], remote sensing [8], action localization [19], and regression [11, 25].

With a strong emphasis on image classification, active learning for object detection has received less attention than expected due to the difficulty to aggregate several object hypothesis at frame level. Recently, [60] employed a loss module to learn the loss of a target model and select the images based on their output loss. However, in hybrid tasks such as object detection learning the loss is challenging. In [44], the active learning approach is based on a 'query-by-committee' strategy. A committee of classifiers is formed by the last convolutional layer of the base network together with the extra convolutional layers of the SSD architecture [39]. The disagreement between them for each candidate bounding box in an image is used as query strategy. In [53], the authors propose a system that learns object detectors on-the-fly, by refining its models via crowd-sourced annotations of web images. As active learning criterion, they use a simple margin approach which selects the most uncertain images which should be annotated. A similar idea is reported in [59], where an object detector is learned interactively, in an incremental manner. The system selects the images most likely to require user input based on an estimated annotation cost computed in terms of false positive and false negative detections. Other approaches to reduce the annotation cost for object detection are based on domain adaptation [20] or transfer learning [52].

In the current work, we introduce a novel active learning approach for object detection in videos, which exploits the temporal coherence of the found detections. The query strategy is based on the number of false positives and false negatives detections identified using a graphical model.

Temporal coherence in video object detection. Several video object detection approaches [18, 26, 27, 38, 63, 64, 54] have attempted to use temporal information to enhance single-image object detectors [41] for multi-class video object detection. There are two main types of approaches. First, temporal information can be used to refine the detections output by the detector as a post-processing step. For example, Seq-NMS [18] re-scores detections using highly overlapping detections from surrounding frames. Some approaches [26, 27] are based on the concept of *tubelet*, i.e. spatio-temporal bounding boxes that span consecutive frames. T-CNN [27] uses tubelets, generated by tracking high confidence detections across frames, to re-score detections and recover false negatives. The second type of approaches introduces temporal coherence while learning the features used by the model in an end-to-end manner. FGFA [63] uses optical flow to estimate the motion between frames, which is employed to learn features that aggregate information from surrounding frames, while [64] uses it for efficiency reasons, extracting features only for selected frames and propagating them to nearby frames. Contrary to the pixel-level approaches, Motion-Aware network [54] introduces instance-level feature aggregation by estimating the movement of proposals across frames and combining them. All these approaches use temporal information to improve object detection in videos, whereas we exploit it to select sets of samples in the context of active learning.

3. Active Learning for Video Object Detection

We describe here the general process of active learning applied to video object detection. Given a large pool of unlabeled data $\mathcal{D}_U$ (video frames) and an annotation budget b, the goal of active learning is to select a subset of b samples to be annotated as to maximize the performance of an object detection model (e.g. Faster R-CNN [41]). Active learning methods generally proceed sequentially by splitting the budget in several *cycles*. Here we consider the batch-mode variant [49], which annotates multiple samples per cycle, since this is the only feasible option for CNN training. At

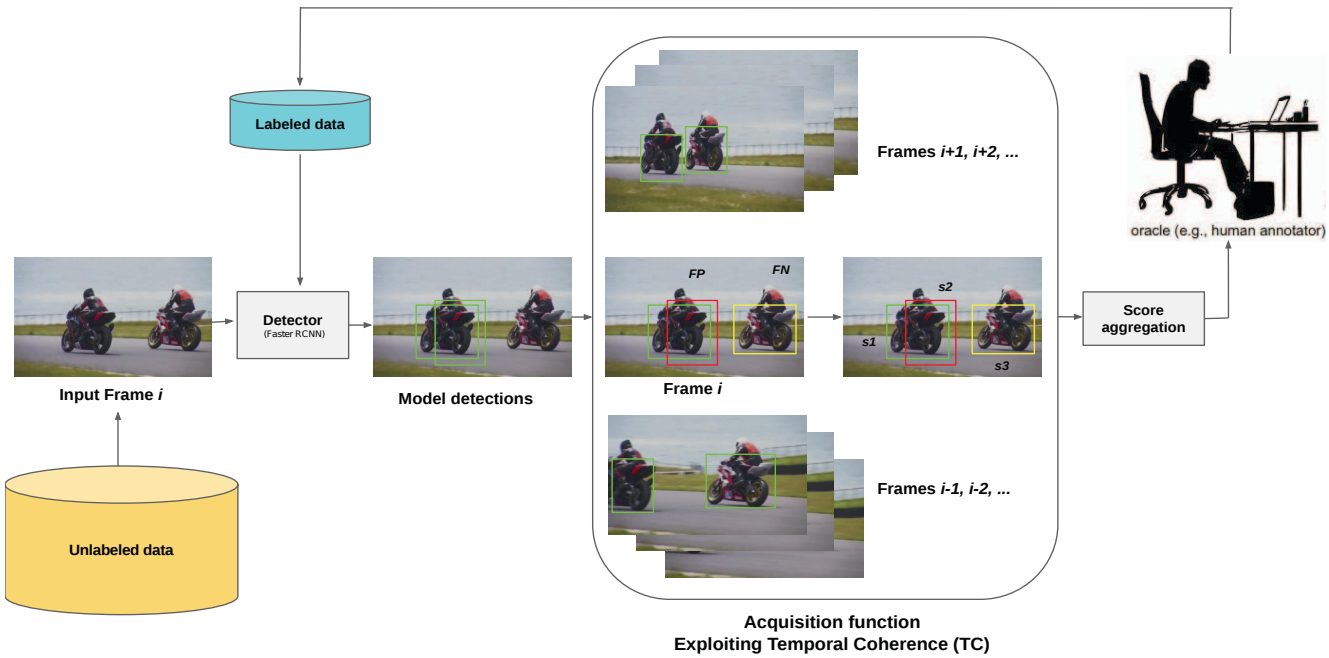

Figure 1. **Overview of our active learning framework exploiting temporal coherence.** The detector outputs detections (green) for each frame in the unlabeled data. Considering the relationships between the detections of neighboring frames (both forward and backward), our temporal coherence acquisition function predicts false positive (red) and false negative (yellow) errors. Based on these predictions, each frame is given an aggregated score and ranked for selection. Finally the frames with top scores are annotated and added to the labeled data.

the beginning of each cycle, the model is trained on the labeled set of samples $\mathcal{D}_L$[1]. After training, the model is used to select a new set of samples to be annotated at the end of the cycle via an *acquisition function*. The selected samples are added to the labeled set $\mathcal{D}_L$ for the next cycle and the process is repeated until the annotation budget b is spent. Fig. 1 presents the active learning framework with our temporal coherence acquisition function, described in sec. 3.2. Note how each sample corresponds to an entire frame and thus all objects in the frame are annotated simultaneously.

The acquisition function is the most crucial component and the main difference between active learning methods in the literature. In general, an acquisition function φ receives a sample x and outputs a score $\varphi(x)$ indicating how valuable x is for training the current model. More sophisticated acquisition functions may consider additional data such as the samples already selected for the current batch, the previously labeled samples $\mathcal{D}_L$, or the unlabeled pool $\mathcal{D}_U$ (see [49] for details). In the remainder of this section, we introduce our two proposed acquisition functions for video object detection in road scenes. Sec. 3.1 presents an exploratory function that approximates a performance upper bound. Sec. 3.2 describes our main contribution: a practical acquisition function based on temporal coherence and specialized for video object detection.

[1]Most methods start with a small initial labeled set selected at random.

3.1. Oracle-based acquisition

The underlying assumption of active learning is that some data samples provide more valuable information than others, so that when labeled and used for training, they improve the model performance by decreasing the number of errors. A suitable acquisition function would select those samples in which the network commits the greatest number of errors so they can be remedied. Assuming perfect generalization from training to test data, such function would be an upper bound for all active learning methods.[2] Motivated by this and in order to study the potential of active learning for video object detection, we propose here an *oracle-based* acquisition function to implement this desirable behavior.

Our oracle-based active selection uses ground-truth information to quantify the number of errors in a given image, and selects those images that have the greatest number of errors. Note this is not a useful active learning function in practice, as we would not have access to the ground-truth annotations in a real scenario. We consider two types of errors that directly affect the usually employed object detection metric of Average Precision (AP) [9, 36]: False Positives (FP) and False Negatives (FP). Let us consider a detection as *correct* if it overlaps a ground-truth bounding box more than 0.5, using the Intersection-over-Union (IoU)

[2]In practice, a decrease in errors in the training set may not necessarily lead to better performance in a separate test set, making this acquisition function an *approximation* to the upper bound.

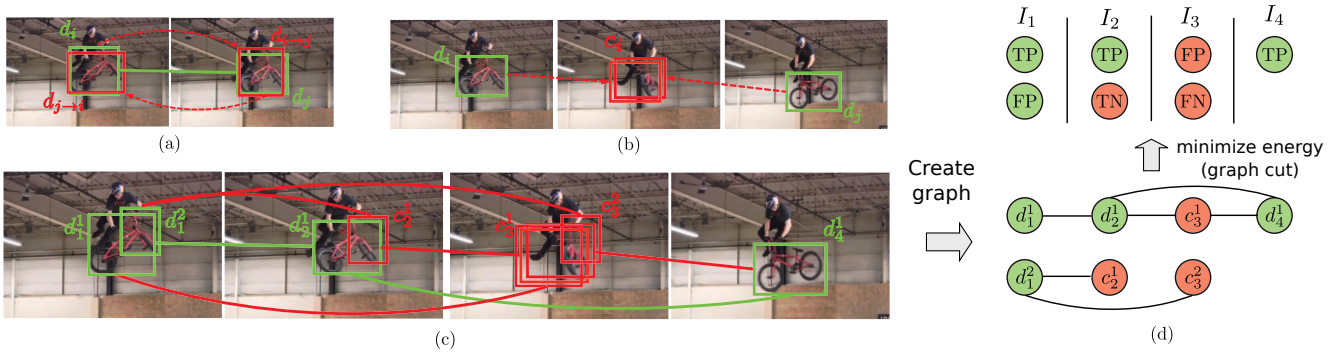

Figure 2. **Error estimation using temporal coherence.** (a) Detections (green) across different frames are linked depending on the overlap with their corresponding tracks (red). (b) Candidate detections (red) are obtained by clustering tracked detections that do not overlap any local detection. (c) Example of detections, candidates, and their links for four consecutive frames. (d) Nodes of the generated graph using detections and candidates corresponding to figure (c). Once the graph is created, we minimize its energy via graph-cut to obtain and estimation of the errors in terms of FP and FN. In this example, we only track up to two surrounding frames, but in practice we use three.

measure for overlap [9]. FPs are detections that are not correct (i.e. have little or no overlap with any ground-truth) or are duplicated, while FNs are those ground-truth instances that have not been detected. We consider two different acquisition functions, one which considers the number of FPs in a frame and the other which considers the number of FNs in a frame[3]. Since the acquisition scores of these functions are integer numbers, it is frequent to have ties between images. We disambiguate between ties by random selection.

3.2. Temporal coherence for error estimation

Video data has the inherent property of *temporal coherence*, i.e. nearby frames are expected to contain the same instances in nearby locations. Based on this, we propose a method to estimate the errors of a video object detector by exploiting the expected temporal coherence, and then use the estimates with the oracle-based acquisition function proposed in sec. 3.1, but using estimations as oracle.

Let us consider a video v composed of a sequence of L frames $\{I_1, ..., I_L\}$. An object detector outputs a set of detections $D_i = \{d_i^0, ..., d_i^K\}$ for each frame I_i[4]. Temporal coherence induces a bijective mapping between sets of detections in nearby frames when corrected for minor localization changes. In order to correct such changes we employ an object tracker, of which details follow later. Formally, given a detection d_i^k in frame I_i, the tracker estimates the location of the contents of this region in frame I_j, which we refer to as $d_{i\to j}^k$. The tracking can be performed in the direction of time ($i < j$) or in the reverse direction. The set of all tracked detections $D_{i\to j} = \{d_{i\to j}\}$ can be thought of as weak detections obtained via temporal coherence using another frame's detections, rather than being directly predicted by the object detector based on the frame's content. We can now link detections of the same class across frames based on their tracked detections. More concretely, we link detection d_i^k in frame I_i with detection d_j^l in I_j if $\text{IoU}(d_i^k, d_{j\to i}^l) > \theta$ or $\text{IoU}(d_j^l, d_{i\to j}^k) > \theta$ (Fig. 2a). That is, if any of the tracked detections (forward or backward) overlaps the other detection in the corresponding frame. Note how there might be tracked detections that are not matched with any local detection (Fig. 2b). Such tracked detections could indicate the presence of an instance in that frame that has been missed by the detector. We cluster groups of unmatched tracked detections in the same frame based on their overlap. We term these groups as detection *candidates* and use the notation c_i^k for the k-th candidate of frame I_i.

Each detection d_i can either be a True Positive (TP) if it correctly localizes an object instance in the image, or a FP if it erroneously predicts the presence of a particular object. On the other hand, a detection candidate c_i can be a True Negative (TN) if no object instance is present in its location, or a FN if it corresponds to a missed detection. We now estimate the type of every detection and detection candidate by formalizing our approach as a graphical model.

Graphical model. Let us express all detections and candidates as a set of binary random variables $\mathcal{V} = \{v_1, ..., v_N\}$, where $v_n = d$ if it corresponds to a detection d_i^k and $v_n = c$ for a candidate c_i^k. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with vertices $\mathcal{V}$ and edges $\mathcal{E}$ between connected detections across different frames (via the links previously introduced) and candidates connected with their originating detections (see Fig. 2). Each v_n can take one of four possible labels: TP, FP, TN, or FN. We consider the following energy function on label assignment $\mathcal{L}$:

$$E(\mathcal{L}) = \sum_{v\in\mathcal{V}} \phi_v(l_v) + \sum_{v_1,v_2\in\mathcal{C}} \psi_{v_1,v_2}(l_{v_1}, l_{v_2}), \quad (1)$$

[3]We experimented with combining both FP and FN in the acquisition function but found this to not improve results.

[4]Here we consider object detectors that process each frame independently, such as Faster R-CNN [41].

where $\phi_v(l_v)$ is the unary cost of assigning label l_v to v and $\psi_{v_1,v_2}(l_{v_1}, l_{v_2})$ is the pairwise cost of assigning the label pair (l_{v_1}, l_{v_2}) to a pair of connected variables $(v_1, v_2) \in \mathcal{E}$. We define the unary cost for detection variables as

$$\phi_{v=d}(l_v) = \begin{cases} 0 & \text{if } l_v = \text{TP} \\ \infty & \text{if } l_v = \text{TN} \\ 1 & \text{if } l_v = \text{FP} \\ \infty & \text{if } l_v = \text{FN} \end{cases} \quad (2)$$

This indicates that in principle we trust the outputs of the detector and that assigning a contradicting label should incur some cost. By definition, detections are 'positives' and thus assigning a 'negative' label is strongly discouraged. Analogously, the unary cost for candidate variables is

$$\phi_{v=c}(l_v) = \begin{cases} \infty & \text{if } l_v = \text{TP} \\ 0 & \text{if } l_v = \text{TN} \\ \infty & \text{if } l_v = \text{FP} \\ 1 & \text{if } l_v = \text{FN} \end{cases} \quad (3)$$

In this case, candidates can only be negatives as they are not part of the original outputs of the detector and hence cannot be positives.

We specify the pairwise cost using the following matrix

$$\psi_{v_1,v_2}(l_{v_1}, l_{v_2}) = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}, \quad (4)$$

where the considered label assignment order is l_v = (TP, FP, TN, FN). This indicates that TP should be connected with other TP or FN, whereas FP are preferably connected with other FP or with TN. Intuitively, the pairwise cost enforces temporal coherence between the detections and the candidates, propagating the correctness to connected variables and collaboratively determining the errors.

We optimize the energy function in (1) via graph cut [30], which finds the globally optimal solution by solving the dual max-flow problem. In fact, the problem can be reduced to a binary labelling problem, considering only two possible labels (True or False) with different meanings depending on the type of input variable, i.e. positives for detections and negatives for candidates. We use the graph-cut implementation in the Python library PyMaxflow [2].

Acquisition function. Once all variables in $\mathcal{V}$ have been assigned their optimal labels, we record the estimated number of FPs and FNs contained in each frame. We revert now to the oracle-based acquisition function described in sec. 3.1, but using error estimates instead of actual errors, which makes the function is useful in practice as it does not require any ground-truth information. We refer to this acquisition function as Temporal Coherence (TC). Experimental results show similar performance when considering only FP, only FN, or both FP and FN. Therefore, we use only the number of FP for the acquisition function of TC.

Subset Name	Seq.	Frames	Area	Conditions	P(Pe/Cy/Ca/Wh)
Default	150	74K	C,H	S,W,F,R	30/20/35/0
Town	36	17K	T	S,W,F,R	30/20/35/0
Night	6	3K	C,H	N	0/0/35/0
Wheelchair	5	2K	C,T	S	20/20/0/100
Test (no WC)	85	40K	C,H,T	S,F,R,N	30/20/35/0
Test (WC)	12	5K	C,T	S	20/20/0/100

Table 1. **SYNTHIA-AL data distribution.** Seq. indicates the number of videos. Environment conditions are Fall (F), Winter (W), Spring (S), Rain (R), and Night (N). Areas are City (C), Town (T), and Highway (H). The spawning probabilities are given for pedestrians (Pe), cyclists (Cy), cars (Ca), and wheelchairs (Wh).

Object tracker. In order to temporally link detections and construct connections between graph nodes, we considered two types of object trackers, namely Optical Flow (PWC-NET) [51] and SiamFC tracker [1]. To utilize optical flow for the purpose of object tracking, we first compute a dense 2D real-valued vector map of the motions between all pairs of consecutive frames in the dataset. Then, we translate the box coordinates using the motion vector corresponding to the box center to obtain the tracked box in the next or previous frame. As an alternative to track detections we employ SiamFC [1], a state of the art Siamese-based object tracker. The bottleneck of this tracking method in the context of active learning is that, despite its efficiency, it imposes a huge computational burden when tracking detections every cycle, given the vast amount of detections. On the contrary, optical flow is only computed once at the beginning and can be used throughout all cycles with a negligible overhead.

4. Synthetic Dataset

Most active learning methods [13, 48, 49] are evaluated on simple image classification datasets such as MNIST [32] or CIFAR [31]. Approaches specific for object detection [3, 44, 53, 60] mainly use PASCAL VOC [9], covering various scene types. In the context of autonomous driving, only [44] uses a dataset depicting road scenes, KITTI [15]. Similarly to several other image datasets for autonomous driving [6, 61], KITTI is manually curated to mostly contain relevant knowledge usable to train object detection models. This process is performed by human annotators that select interesting data samples containing cars, pedestrians, etc. The goal of active learning, however, is automatizing this process, making existing datasets not suitable for a proper evaluation. Ideally, a good dataset for evaluating active learning contains a more raw version of the data, in which the image distribution is unbalanced towards the uninteresting (e.g. empty road scenes) and highly redundant. Such dataset would better represent the type of data collected in a real setting, for example, video captured from a driving car.

For this reason, and following recent trends [42, 50], we have created a new synthetic dataset to evaluate active learning for object detection in road scenes. In particular, we modified the SYNTHIA environment [42] to generate the

SYNTHIA-AL dataset[5] using Unity Pro game engine. The aim is having an unbalanced foreground/background distribution, simulating the real collection scenario of a driving car. Moreover, a set of object classes and conditions should be predominantly present, while other classes and conditions must appear less frequent.

The data is generated by driving a car in a virtual world consisting of three different areas, namely town, city, and highway. These areas are populated with a variety of pedestrians, cars, cyclists, and wheelchairs, except for the highway which is limited to cars. These dynamic objects are arbitrarily spawned at predefined positions with a given probability and follow randomly predefined paths without leaving each area. Several environmental conditions can be set: season (winter, fall, spring), day time (day or night), and weather (clear or rainy). By default, we always use spring and clear during the day, and only change one condition at a time. Objects with no lights can be hard to visualize during the night, so we only use cars for the night condition. Figure 3 shows examples of images in the dataset.

Table 1 provides the specification of the dataset. The video sequences are captured at 25 fps with a random length between 10 and 30 seconds. We have generated one subset with the default parameters and three smaller subsets with altered conditions. The first subset consists of 150 sequences, which amounts to 75% of all the data, with the default settings, i.e. containing cars, pedestrians, and cyclists, under different daily conditions, but only in the city and highway areas. The second subset contains 36 sequences (20% of the dataset) captured in the town area instead. The night condition only represents 3% of the whole data (6 sequences) and it is fully contained in the third subset. Finally, we have added wheelchairs and removed cars in the fourth subset, which represents the 2% of the dataset with only 5 sequences. The test set contains 85 sequences with balanced distributions on areas and conditions (except winter) on the three main classes plus another 12 sequences including wheelchairs. All images are automatically annotated with 2D bounding boxes and class labels for every object that can be reasonably seen (more than 50 pixels).

5. Experimental Setup

5.1. Active learning procedure

All considered active learning methods follow the same procedure and employ the same state-of-the-art object detector based on Faster R-CNN [41]. We start with the model pre-trained on COCO [36], which contains 80K images from 80 different object categories. The initial labeled set $\mathcal{D}_L$ consists of 2% of train dataset that is selected randomly once for all the methods. At each cycle, we fine-tune the latest model of the previous cycle, as we have experimentally observed that this leads to faster convergence than fine-tuning the initial model or from scratch as in [5]. We have also seen that in order not to get stuck in local minima, the learning rate should be high enough. Once the new model is fine-tuned, we use it with the corresponding acquisition function to select b/C frames, which are then labeled and added to $\mathcal{D}_L$. We continue for C cycles until budget b is completely exhausted. In all experiments, the budget per cycle is 2% of the dataset.

Evaluation. For each cycle, we evaluate the model trained with the updated labeled set for that cycle on the test set. Detections are processed using Non-Maxima Supression [10] and thresholded by score, rejecting all detections below 0.5. We use AP averaged over all classes using a detection threshold of IoU> 0.5.

Implementation details. We used Tensorflow's Object Detection API [21] as the base code to develop our experiments. We trained all models with the momentum optimizer with value 0.9 and the initial learning rates 0.02 and 0.001 for SYNTHIA-AL and ImageNet-VID [45] datasets, respectively. We train for 10 epochs and reduce the learning rate by a factor of 5 once after 5 epochs and again at 7 epochs for SYNTHIA-AL. In the case of ImageNet-VID we reduce the learning rate at epochs 3 and 5, training a total of 6 epochs. For efficiency reasons, we resize all images to fixed height of 300 pixels and preserve the aspect ratio. We use a batch size of 12 for all the experiments. Finally, to obtain more stable results we repeat the experiments 3 times and report the mean and standard deviation in our results.

5.2. Baselines

Random. Random sampling selects an arbitrary subset of frames from all unlabeled frames. Given the extreme imbalance inherent to video data due to varying video length, uniform random sampling selects most frames from the longer videos while under-representing shorter videos, which damages the performance. Moreover, video data is redundant due to the high similarity between nearby frames, which makes annotating the surrounding frames of an already annotated frame wasteful. For these reasons, we also consider an improved random sampling procedure that includes temporal representativeness, which prevents selecting the k neighbors in both directions of already labeled frames. In the experiments, we set the k to 3 for ImageNet-VID dataset and 1 for SYNTHIA-AL dataset for all the methods. This criterion naturally increases the diversity of the selected batches at each cycle by limiting the similarity between data samples. We call this baseline *Random+R*.

Uncertainty. We consider three other baselines based on uncertainty measures used in recent active learning approaches for object detection [3, 44]. *Least confidence* [33, 44] considers the score of the most probable class and selects those samples that have the lowest score on it. *Entropy* [7] is an information theory measure that captures the

[5] Available at http://www.synthia-dataset.net

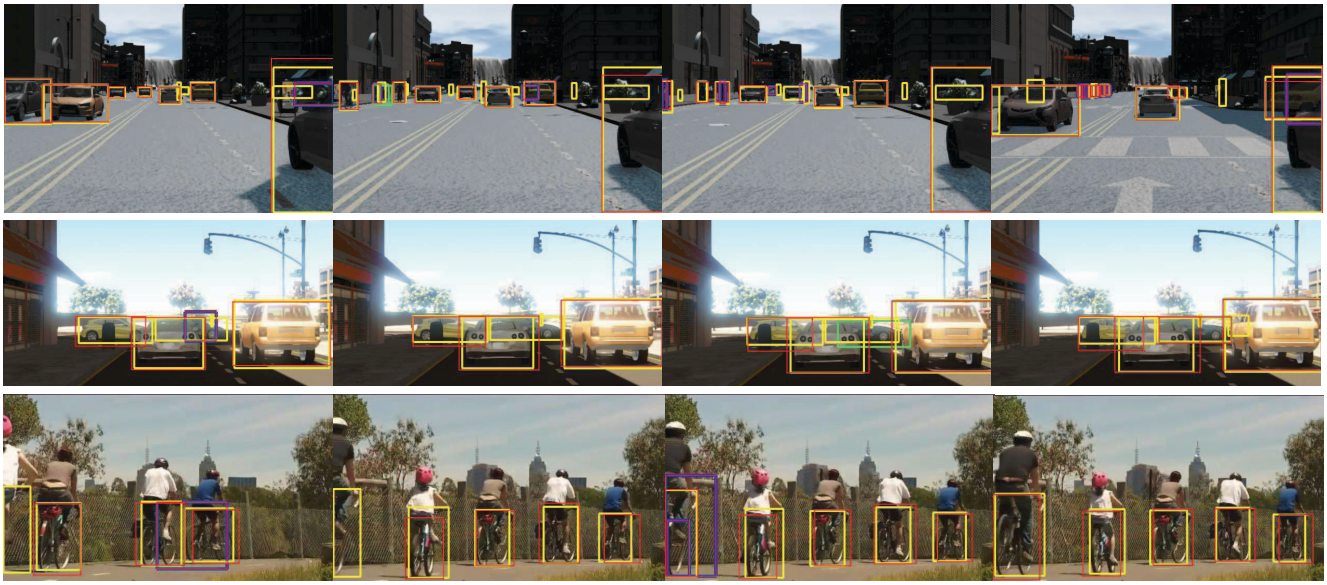

Figure 3. **Examples of errors detected by our temporal coherence approach on SYNTHIA-AL (top, middle) and ImageNet-VID [45] (bottom)**. We show ground-truth boxes in yellow and output detections in red. After solving our graphical model based on temporal coherence, some of the detections are considered as false positives (purple), while other boxes are added as false negatives (green).

average amount of information contained in the predictive distribution, attaining its maximum value when all classes are equally probable. In both cases, we use the average score of all detections in the image to obtain a single score per image. *Margin sampling* [49, 3] uses the difference between the two classes with the highest scores as a measure of proximity to the decision boundary. Following [3], we sum all margin sampling scores of individual detections to aggregate them into an overall image score.

5.3. Datasets

Besides our SYNTHIA-AL dataset (sec. 4), we also perform experiments on a real-image dataset, ImageNet-VID [45], which is commonly used as video object detection benchmark. Since the focus of this paper is video object detection in road scenes, we select 3 classes that are likely to be encountered in the context of autonomous driving, namely: car, bike, and motorcycle. Selecting all videos that contain these classes amounts to 795 videos in the training set and 87 videos in the validation set, which we use for test. The length of the videos varies between a few frames to over 1000. We have cleaned this dataset by manually discarding all those frames that had missing annotations, which amounts to 20K frames in the training set and 5K frames in the validation set. The final dataset contains 129K frames for training and 14K frames for validation.

6. Results

We present active learning results using performance (mAP) curves as a function of the number of selected samples, as usually reported in the literature [13, 48]. This allows us assess the benefit of each active learning method for different total number of samples used to train the model. For each method, we plot the average performance for all runs with vertical bars to represent the standard deviation.

We first validate the ability of our graphical model (sec. 3.2) to estimate detection errors using temporal coherence. Fig. 3 presents some resulting predictions on both datasets. We can see how many FP (purple) are correctly detected, including those corresponding to double detections (top row, rightmost column). Moreover, FN (green) are discovered due to the forward and backward tracking of surrounding detections (middle row, third column).

6.1. SYNTHIA-AL

Fig. 4 presents all quantitative results on our SYNTHIA-AL dataset. We start by evaluating the difference between the two random baselines: uniform and our enhanced Random+R baseline (Fig. 4a). We can observe how the addition of representativeness is clearly beneficial for active learning in video object detection. In the remainder of the paper, we always include temporal representativeness and per-video sampling for all evaluated methods.

Next, we evaluate the effect of the two types of trackers considered in our temporal coherence method, SiamFC [1] and Optical Flow [51], within the active learning cycles. Fig. 4b presents the quantitative evaluation of temporal coherence with either tracker. The results show that there is no improvement gained by using the more sophisticated SiamFC tracker compared to Optical Flow. Furthermore, Optical Flow can significantly speed up the active learning process. In this case, the motion vectors are computed once at the beginning of the process, whereas SiamFC needs to perform expensive computations at every cycle.

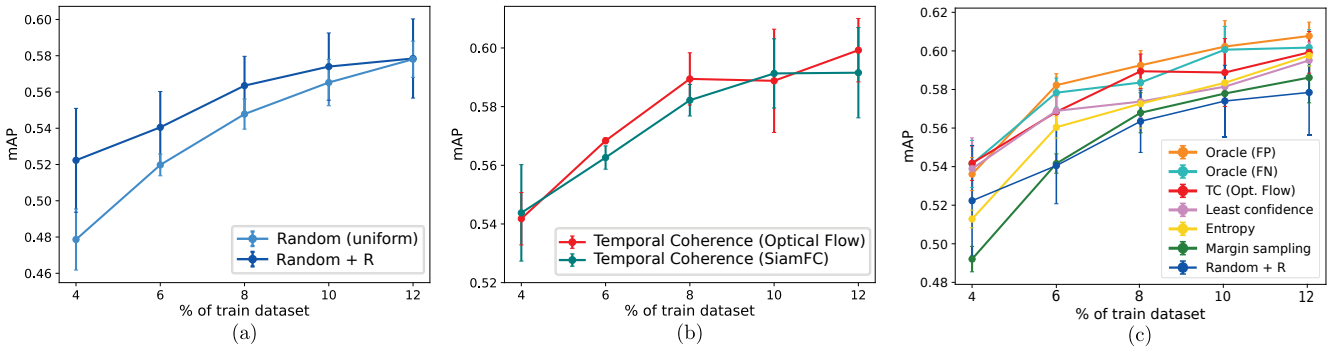

Figure 4. **Results on SYNTHIA-AL.** (a) Random baselines with and without representativeness. (b) Our Temporal Coherence using either Optical Flow or SiamFC. (c) Baselines, oracle-based acquisition, and Temporal Coherence. All curves are the average of 3 runs.

Methods	SYNTHIA-AL		ImageNet-VID	
	mAP	Rel.	mAP	Rel.
All data	0.628	100%	0.839	100%
Random+R	0.578	92.0%	0.821	97.8%
Least Confidence	0.595	94.7%	0.818	97.4%
Margin sampling	0.586	93.3%	0.820	97.7%
Entropy	0.597	95.0%	0.821	97.8%
Oracle (FP)	0.607	96.6%	-	-
Oracle (FN)	0.601	95.7%	-	-
Temporal Coherence (SiamFC)	0.591	94.1%	-	-
Temporal Coherence (Opt. Flow)	**0.599**	**95.3%**	**0.830**	**98.9%**

Table 2. **Active learning results.** The first row shows the performance (mAP) obtained when using the entirety of the dataset. All other rows show the performance of all methods using 12% of all data in SYNTHIA-AL and 10% of ImageNet-VID [45], both in absolute performance and relative to using all data.

Finally, we compare Temporal Coherence (TC) with all baselines. To explore an upper bound for TC, we also consider the oracle-based methods of section 3.1, selecting those frames with the highest number of FP or FN based on ground-truth information. These methods are designated by Oracle (FP) and Oracle (FN), respectively. The results in Fig. 4c show that our TC method outperforms all three uncertainty based methods and the random baseline. The narrow gap between our TC method and the oracle-based methods implies that FP and FN predictions of the graphical model are effective estimates of the actual errors that the model can learn from. Moreover, TC enables us to achieve more than 95% of performance of the model trained on entire dataset by annotating only 12% of the data. Table 2 shows the effectiveness of active learning methods in videos by using a small portion of datasets.

6.2. ImageNet-VID

To evaluate our temporal coherence method on a dataset of real images, we perform experiments on ImageNet-VID [45]. Fig. 5 compares TC with Optical Flow against uncertainty and random baselines. The results illustrate that TC is superior to all the baselines for all cycles. Additionally, Table 2 shows that TC manages to attain almost the full performance of a model trained with the entire dataset by using only 10% of the data, which is a significant reduction in the annotation effort.

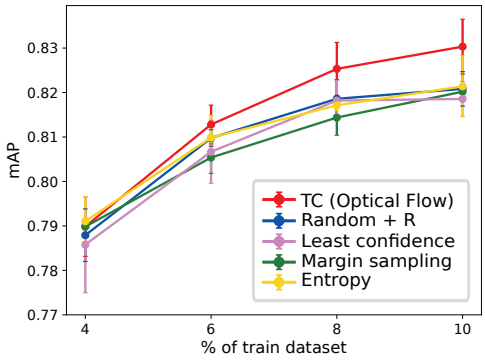

Figure 5. **Results on ImageNet-VID [45].** Average of 3 runs.

7. Conclusions

In this paper, we introduced a novel active learning approach for object detection in videos which exploits the temporal coherence. Our approach is formulated in terms of an energy minimization function of a graphical model built on tracked object detections. Additionally, we introduced a new synthetic dataset specially designed to evaluate active learning for object detection in the context of autonomous driving. Experimental results conducted on two datasets showed that our approach outperformed major active learning baselines. A drawback of temporal coherence based active learning is that it is computationally more demanding than the baselines. We plan to minimize the computational overhead of our system in future research.

Acknowledgements. The authors thank Audi Electronics Venture GmbH for their support during the development of this work, the Generalitat de Catalunya CERCA Program and its ACCIO agency, Unity for the support in the synthetic dataset generation, the EU Project CybSpeed MSCA-RISE-2017-777720 and CYTED Network (Ref. 518RT0559). Antonio thanks the financial support by ICREA under the ICREA Academia Program, and the Spanish project TIN2017-88709-R (MINECO/AEI/FEDER, UE).

References

[1] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object tracking. In *ECCV*, pages 850–865, 2016. 5, 7

[2] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Trans. on PAMI*, 26(9):1124–1137, 2004. 5

[3] C.-A. Brust, C. Käding, and J. Denzler. Active learning for deep object detection. In *VISAPP*, 2019. 5, 6, 7

[4] W. Cai, Y. Zhang, S. Zhou, W. Wang, C. Ding, and X. Gu. Active learning for support vector machines with maximum model change. In *Machine Learning and Knowledge Discovery in Databases*, pages 211–226. Springer, 2014. 2

[5] K. Chitta, J. M. Alvarez, and A. Lesnikowski. Large-scale visual active learning with deep probabilistic ensembles. *arXiv preprint arXiv:1811.03575*, 2018. 6

[6] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *CVPR*, pages 3213–3223, 2016. 5

[7] I. Dagan and S. P. Engelson. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings 1995*, pages 150–157. Elsevier, 1995. 6

[8] C. Deng, X. Liu, C. Li, and D. Tao. Active multi-kernel domain adaptation for hyperspectral image classification. *Pattern Recognition*, 77:306–315, 2018. 1, 2

[9] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 3, 4, 5

[10] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. on PAMI*, 32(9):1627–1645, 2010. 6

[11] A. Freytag, E. Rodner, and J. Denzler. Selecting influential examples: Active learning with expected model output changes. In *ECCV*, pages 562–577, 2014. 2

[12] W. Fu, M. Wang, S. Hao, and X. Wu. Scalable active learning by approximated error reduction. In *KDD*, pages 1396–1405, 2018. 2

[13] Y. Gal, R. Islam, and Z. Ghahramani. Deep bayesian active learning with image data. In *ICML*, pages 1183–1192, 2017. 2, 5, 7

[14] E. Gavves, T. E. J. Mensink, T. Tommasi, and T. Snoek, C. G. M.and Tuytelaars. Active transfer learning with zero-shot priors: Reusing past datasets for future tasks. In *ICCV*, pages 1–9, 2015. 1, 2

[15] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *IJRR*, 32(11):1231–1237, 2013. 1, 5

[16] Q. Gu, T. Z. Zhang, C. Ding, and J. Han. Selective labeling via error bound minimization. In *NIPS*, pages 1–9, 2012. 2

[17] Y. Guo. Active instance sampling via matrix partition. In *NIPS*, pages 1–9, 2010. 2

[18] W. Han, P. Khorrami, T. L. Paine, P. Ramachandran, M. Babaeizadeh, H. Shi, J. Li, S. Yan, and T. S. Huang. Seq-nms for video object detection. *arXiv preprint arXiv:1602.08465*, 2016. 2

[19] F. C. Heilbron, J.-Y. Lee, H. Jin, and B. Ghanem. What do i annotate next? an empirical study of active learning for action localization. In *ECCV*, pages 212–229, 2018. 2

[20] J. Hoffman, S. Guadarrama, E. Tzeng, J. Donahue, R. B. Girshick, T. Darrell, and K. Saenko. Lsda: Large scale detection through adaptation. In *NIPS*, pages 1–9, 2014. 2

[21] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *CVPR*, pages 7310–7311, 2017. 6

[22] S.-J. Huang, R. Jin, and Z.-H. Zhou. Active learning by querying informative and representative examples. *IEEE Trans. on PAMI*, 10(36):1936–1949, 2014. 2

[23] S. Jin, A. RoyChowdhury, H. Jiang, A. Singh, A. Prasad, D. Chakraborty, and E. Learned-Miller. Unsupervised hard example mining from videos for improved object detection. In *ECCV*, pages 307–324, 2018. 1

[24] A. J. Joshi, F. Porikli, and N. P. Papanikolopoulos. Scalable active learning for multiclass image classification. *IEEE Trans. on PAMI*, 34(11):2259–2273, 2012. 1, 2

[25] C. Käding, E. Rodner, A. Freytag, O. Mothes, B. Barz, and J. Denzler. Active learning for regression tasks with expected model output changes. In *BMVC*, pages 1–15, 2018. 2

[26] K. Kang, H. Li, T. Xiao, W. Ouyang, J. Yan, X. Liu, and X. Wang. Object detection in videos with tubelet proposal networks. In *CVPR*, pages 727–735, 2017. 1, 2

[27] K. Kang, H. Li, J. Yan, X. Zeng, B. Yang, T. Xiao, C. Zhang, Z. Wang, R. Wang, X. Wang, et al. T-cnn: Tubelets with convolutional neural networks for object detection from videos. *IEEE TCSVT*, 28(10):2896–2907, 2018. 1, 2

[28] A. Kapoor, K. Grauman, R. Urtasun, and T. Darrell. Active learning with gaussian processes for object categorization. In *CVPR*, pages 1–8, 2007. 2

[29] V. Karasev, A. Ravichandran, and S. Soatto. Active frame, location, and detector selection for automated and manual video annotation. In *CVPR*, pages 2131–2138, 2014. 1

[30] V. Kolmogorov and R. Zabih. What energy functions can be minimizedvia graph cuts? *IEEE Trans. on PAMI*, 26(2):147–159, 2004. 5

[31] A. Krizhevsky and G. Hinton. Learning multiple layers of features from tiny images. *Technical report, University of Toronto*, 2009. 5

[32] Y. LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998. 5

[33] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *SIGIR*, pages 3–12, 1994. 6

[34] X. Li and Y. Guo. Adaptive active learning for image classification. In *cvpr*, pages 860–866, 2013. 1

[35] X. Li and Y. Guo. Multi-level adaptive active learning for scene classification. In *ECCV*, pages 234–249, 2014. 1, 2

[36] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pages 740–755, 2014. 3, 6

[37] X. Lin and D. Parikh. Active learning for visual question answering: An empirical study. *arXiv preprint arXiv:1711.01732*, 2017. 2

[38] M. Liu and M. Zhu. Mobile video object detection with temporally-aware feature maps. In *CVPR*, pages 5686–5695, 2018. 2

[39] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016. 2

[40] V. Madhavan and T. Darrell. The bdd-nexar collective: A large-scale, crowsourced, dataset of driving scenes. Master's thesis, EECS Department, University of California, Berkeley, May 2017. 1

[41] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015. 2, 4, 6

[42] G. Ros, L. Sellart, J. Materzyska, D. Vázquez, and A. López. The SYNTHIA dataset: a large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, pages 3234–3243, 2016. 5

[43] A. Rosenfeld, R. Zemel, and J. K. Tsotsos. The elephant in the room. *arXiv preprint arXiv:1808.03305*, 2018. 1

[44] S. Roy, A. Unmesh, and V. P. Namboodiri. Deep active learning for object detection. In *BMVC*, pages 1–12, 2018. 1, 2, 5, 6

[45] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 1, 6, 7, 8

[46] P. Saito, C. Suzuki, J. Gomes, P. de Rezende, and A. Falcão. Robust active learning for the diagnosis of parasites. *Pattern Recognition*, 48(11):3572–3583, 2015. 1, 2

[47] A. I. Schein and L. H. Ungar. Active learning for logistic regression: an evaluation. *Machine Learning*, 68(3):235–265, 2007. 2

[48] O. Sener and S. Savarese. Active learning for convolutional neural networks: A core-set approach. In *ICLR*, pages 1–13, 2018. 2, 5, 7

[49] B. Settles. Active learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2009. 1, 2, 3, 5, 7

[50] Stephan R. Richter, Z. Hayder, and V. Koltun. Playing for benchmarks. In *ICCV*, pages 2213–2222, 2017. 5

[51] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. June 2018. 5, 7

[52] A. Vezhnevets, J. M. Buhmann, and V. Ferrari. Active learning for semantic segmentation with expected change. In *CVPR*, pages 3162–3169, 2012. 2

[53] S. Vijayanarasimhan and K. Grauman. Large-scale live active learning: Training object detectors with crawled data and crowds. *IJCV*, 108(1–2):97–114, 2014. 1, 2, 5

[54] S. Wang, Y. Zhou, J. Yan, and Z. Deng. Fully motion-aware network for video object detection. In *ECCV*, pages 542–557, 2018. 1, 2

[55] Y. Yan, F. Nie, W. Li, C. Gao, Y. Yang, and D. Xu. Image classification by cross-media active learning with privileged information. *IEEE Trans. on Multimedia*, 18(12):2494–2502, 2016. 1

[56] Y. Yang and M. Loog. A benchmark and comparison of active learning for logistic regression. *Pattern Recognition*, 83:401–415, 2018. 2

[57] Y. Yang and M. Loog. A variance maximization criterion for active learning. *Pattern Recognition*, 78:358–370, 2018. 2

[58] Y. Yang, Z. Ma, F. Nie, X. Chang, and A. G. Hauptmann. Multi-class active learning by uncertainty sampling with diversity maximization. *IJCV*, 113(2):113–127, 2015. 2

[59] A. Yao, J. G. Gall, C. Leistner, and L. Van Gool. Interactive object detection. In *CVPR*, pages 3242–3249, 2012. 1, 2

[60] D. Yoo and I. S. Kweon. Learning loss for active learning. In *CVPR*, pages 93–102, 2019. 2, 5

[61] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv preprint arXiv:1805.04687*, 2018. 1, 5

[62] D. Zhang, F. Wang, Z. Shi, and C. Zhang. Interactive localized content based image retrieval with multiple-instance active learning. *Pattern Recognition*, 43(2):478–484, 2010. 2

[63] X. Zhu, Y. Wang, J. Dai, L. Yuan, and Y. Wei. Flow-guided feature aggregation for video object detection. In *ICCV*, pages 408–417, 2017. 1, 2

[64] X. Zhu, Y. Xiong, J. Dai, L. Yuan, and Y. Wei. Deep feature flow for video recognition. In *CVPR*, pages 2349–2358, 2017. 1, 2