T-CNN: Tubelets with Convolutional Neural Networks for Object Detection from Videos

Kai Kang, Hongsheng Li, Junjie Yan, Xingyu Zeng, Bin Yang, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, and Wanli Ouyang

Abstract—The state-of-the-art performance for object detection has been significantly improved over the past two years. Besides the introduction of powerful deep neural networks such as GoogleNet [1] and VGG [2], novel object detection frameworks such as R-CNN [3] and its successors, Fast R-CNN [4] and Faster R-CNN [5], play an essential role in improving the state-of-the-art. Despite their effectiveness on still images, those frameworks are not specifically designed for object detection from videos. Temporal and contextual information of videos are not fully investigated and utilized. In this work, we propose a deep learning framework that incorporates temporal and contextual information from tubelets obtained in videos, which dramatically improves the baseline performance of existing still-image detection frameworks when they are applied to videos. It is called T-CNN, i.e. tubelets with convolutional neueral networks. The proposed framework won the recently introduced object-detection-from-video (VID) task with provided data in the ImageNet Large-Scale Visual Recognition Challenge 2015 (ILSVRC2015). Code is publicly available at https://github.com/myfavouritekk/T-CNN.

1 Introduction

In the last two years, the performance of object detection has been significantly improved with the success of novel deep convolutional neural networks (CNN) [1], [2], [6], [7] and object detection frameworks [3], [4], [5], [8]. The state-of-the-art frameworks for object detection such as R-CNN [3] and its successors [4], [5] extract deep convolutional features from region proposals and classify the proposals into different classes. DeepID-Net [8] improved R-CNN by introducing box pre-training, cascade on region proposals, deformation layers and context representations. Recently, ImageNet introduces a new challenge for object detection from videos (VID), which brings object detection into the video domain. In this challenge, an object detection system is required to automatically annotate every object in 30 classes with its bounding box and class label in each frame of the videos, while test videos have no extra information pre-assigned, such as user tags. VID has a broad range of applications on video analysis.

Despite their effectiveness on still images, these still-image object detection frameworks are not specifically designed for videos. One key element of videos is temporal information, because locations and appearances of objects in videos should be temporally consistent, i.e. the detection results should not have dramatic changes over time in terms of both bounding box locations and detection confidences. However, if still-image object detection frameworks are directly applied to videos, the detection confidences of an object show dramatic changes between adjacent frames and large long-term temporal variations, as shown by an example in Fig. 1 (a).

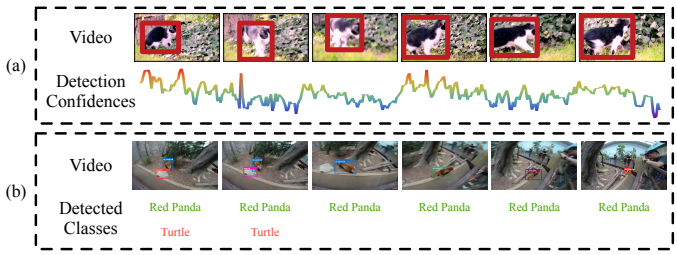

Fig. 1. Limitations of still-image detectors on videos. (a) Detections from still-image detectors contain large temporal fluctuations, because they do not incorporate temporal consistency and constraints. (b) Still-image detectors may generate false positives solely based the information on single frames, while these false positives can be distinguished considering the context information of the whole video.

One intuition to improve temporal consistency is to propagate detection results to neighbor frames to reduce sudden changes of detection results. If an object exists in a certain frame, the adjacent frames are likely to contain the same object at neighboring locations with similar confidence. In other words, detection results can be propagated to adjacent frames according to motion information so as to reduce missed detections. The resulted duplicate boxes can be easily removed by non-maximum suppression (NMS).

Another intuition to improve temporal consistency is to impose long-term constraints on the detection results. As shown in Fig. 1 (a), the detection scores of a sequence of bounding boxes of an object have large fluctuations over time. These box sequences, or tubelets, can be generated by tracking and spatio-temporal object proposal algorithms [9]. A tubelet can be treated as a unit to apply the long-term constraint. Low detection confidence on some positive bounding boxes may result from moving blur, bad poses, or lack of enough training samples under particular poses. Therefore, if most bounding boxes of a tubelet have high confidence detection scores, the low confidence scores at certain frames should be increased to enforce its long-term consistency.

Besides temporal information, contextual information is also a key element of videos compared with still images. Although image context information has been investigated [8] and incorporated into still-image detection frameworks, a video, as a collection of hundreds of images, has much richer contextual information. As shown in Fig. 1 (b), a small amount of frames in a video may have high confidence false positives on some background objects. Contextual information within a single frame is sometimes not enough to distinguish these false positives. However, considering the majority of high-confidence detection results within a video clip, the false positives can be treated as outliers and then their detection confidences can be suppressed.

The contribution of this works is three-folded. 1) We propose a deep learning framework that extends popular still-image detection frameworks (R-CNN and Faster R-CNN) to solve the problem of general object detection in videos

- *Kai Kang and Hongsheng Li share co-first authorship.*
- *Wanli Ouyang is the corresponding author.*
  Email: wlouyang@ee.cuhk.edu.hk
- *Kai Kang, Hongsheng Li, Xingyu Zeng, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, and Wanli Ouyang are with The Chinese University of Hong Kong.*
- *Junjie Yan and Bin Yang are with the SenseTime Group Limited.*

by incorporating temporal and contextual information from tubelets. It is called T-CNN, i.e. tubelets with convolution neural network. 2) Temporal information is effectively incorporated into the proposed detection framework by locally propagating detection results across adjacent frames as well as globally revising detection confidences along tubelets generated from tracking algorithms. 3) Contextual information is utilized to suppress detection scores of low-confidence classes based on all detection results within a video clip. This framework is responsible for winning the VID task with provided data and achieving the second place with external data in ImageNet Large-Scale Visual Recognition Challenge 2015 (ILSVRC2015). Code is available at https://github.com/myfavouritekk/T-CNN.

2 Related Work

Object detection from still images. State-of-the-art methods for detecting objects of general classes are mainly based on deep CNNs [1], [3], [4], [5], [8], [10], [11], [12], [13], [14], [15], [16], [17]. Girshick et al. [3] proposed a multi-stage pipeline called Regions with Convolutional Neural Networks (R-CNN) for training deep CNN to classify region proposals for object detection. It decomposed the detection problem into several stages including bounding-box proposal, CNN pre-training, CNN fine-tuning, SVM training, and bounding box regression. Such framework showed good performance and was widely adopted in other works. Szegedy et al. [1] proposed the GoogLeNet with a 22-layer structure and "inception" modules to replace the CNN in the R-CNN, which won the ILSVRC 2014 object detection task. Ouyang et al. [8] proposed a deformation constrained pooling layer and a box pre-training strategy, which achieved an accuracy of 50.3% on the ILSVRC 2014 test set. To accelerate the training of the R-CNN pipeline, Fast R-CNN [4] was proposed, where each image patch was no longer wrapped to a fixed size before being fed into CNN. Instead, the corresponding features were cropped from the output feature maps of the last convolutional layer. In the Faster R-CNN pipeline [5], the region proposals were generated by a Region Proposal Network (RPN), and the overall framework can thus be trained in an end-to-end manner. All these pipelines were for object detection from still images. When they are directly applied to videos in a frame-by-frame manner, they might miss some positive samples because objects might not be of their best poses at certain frames of videos.

Object localization in videos. There have also been works on object localization and co-localization [18], [19], [20], [21]. Although such a task seems to be similar, the VID task we focus on is actually much more challenging. There are crucial differences between the two problems. **1) Goal:** The (co)locolization problem assumes that each video contains only *one* known (weakly supervised setting) or unknown (unsupervised setting) class and only requires localizing *one* of the objects in each test frame. In VID, however, each video frame contains unknown numbers of objects instances and classes. The VID task is closer to real-world applications. **2) Metrics:** Localization metric (CorLoc [22]) is usually used for evaluation in (co)locolization, while mean average precision (mean AP) is used for evaluation on the VID task. With the above differences, we think that the VID task is more difficult and closer to real-world scenarios. The previous works on object (co)localization in videos cannot be directly applied to VID.

Image classification. The performance of image classification has been significantly improved during the past few years thanks to the large scale datasets [23] and novel deep neural networks [1], [2], [6], [24]. The models for object detection are commonly pre-trained on the ImageNet 1000-class classification task. Batch normalization layer was proposed in [7] to reduce the statistical variations among mini batches and accelerate the training process. Simonyan et al. proposed a 19-layer neural network with very small 3×3 convolution kernels in [2], which was proved effective in other related tasks such as detection [4], [5], action recognition [25], and semantic segmentation [26].

Visual tracking. Object tracking has been studied for decades [27], [28], [29]. Recently, deep CNNs have been used for object tracking and achieved impressive tracking accuracy. Wang et al. [30] proposed to create an object-specific tracker by online selecting the most influential features from an ImageNet pre-trained CNN, which outperforms state-of-the-art trackers by a large margin. Nam et al. [31] trained a multi-domain CNN for learning generic representations for tracking objects. When tracking a new target, a new network is created by combining the shared layers in the pre-trained CNN with a new binary classification layer, which is online updated. Tracking is apparently different from VID, since it assumes the initial localization of an object in the first frame and it does not require predicting class labels.

3 Methods

In this section, we first introduce the VID task setting (Section 3.1) and our overall framework (Section 3.2). Then each major component will be introduced in more details. Section 3.3 describes the settings of our still-image detectors. Section 3.4 introduces how to utilize multi-context information to suppress false positive detections and utilize motion information to reduce false negatives. Global tubelet re-scoring is introduced in Section 3.5.

3.1 VID task setting

The ImageNet object detection from video (VID) task is similar to the object detection task (DET) in still images. It contains 30 classes to be detected, which are a subset of 200 classes of the DET task. All classes are fully labeled in all the frames of each video clip. For each video clip, algorithms need to produce a set of annotations (f_i, c_i, s_i, b_i) of frame index f_i, class label c_i, confidence score s_i and bounding box b_i. The evaluation protocol for the VID task is the same as the DET task, i.e. we use the conventional mean average precision (mean AP) on all classes as the evaluation metric.

3.2 Framework overview

The proposed framework is shown in Fig. 2. It consists of four main components: 1) still-image detection, 2) multi-context suppression and motion-guided propagation, 3) temporal tubelet re-scoring, and 4) model combination.

Still-image object detection. Our still-image object detectors adopt the DeepID-Net [8] and CRAFT [32] frameworks

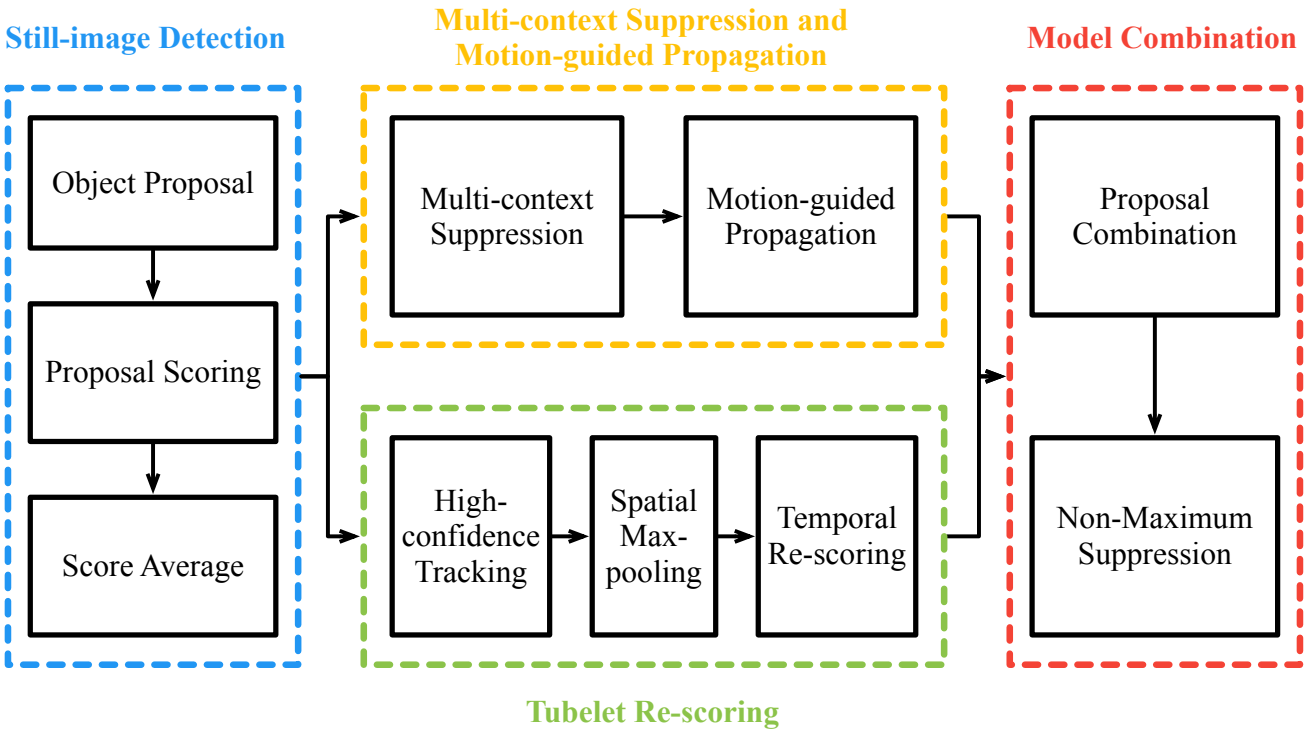

Fig. 2. Our proposed T-CNN framework. The framework mainly consists of four components. 1) The still-image object detection component generates object region proposals in all the frames in a video clip and assigns each region proposal an initial detection score. 2) The multi-context suppression incorporates context information to suppress false positives and motion-guided propagation component utilizes motion information to propagate detection results to adjacent frames to reduce false negatives. 3) The tubelet re-scoring components utilizes tracking to obtain long bounding box sequences to enforce long-term temporal consistency of their detection scores. 4) The model combination component combines different groups of proposals and different models to generate the final results.

and are trained with both ImageNet detection (DET) and video (VID) training datasets in ILSVRC 2015. DeepID-Net [8] is an extension of R-CNN [3] and CRAFT is an extension of Faster R-CNN [5]. Both of the two frameworks contain the steps of object region proposal and region proposal scoring. The major difference is that in CRAFT (also Faster R-CNN), the proposal generation and classification are combined into a single end-to-end network. Still-image object detectors are applied to individual frames. The results from the two still-image object detection frameworks are treated separately for the remaining components in the proposed T-CNN framework.

Multi-context suppression. This process first sorts all still-image detection scores within a video in descending orders. The classes with highly ranked detection scores are treated as high-confidence classes and the rest as low-confidence ones. The detection scores of low-confidence classes are suppressed to reduce false positives.

Motion-guided Propagation. In still-image object detection, some objects may be missed in certain frames while detected in adjacent frames. Motion-guided propagation uses motion information such as optical flows to locally propagate detection results to adjacent frames to reduce false negatives.

Temporal tubelet re-scoring. Starting from high-confidence detections by still-image detectors, we first run tracking algorithms to obtain sequences of bounding boxes, which we call tubelets. Tubelets are then classified into positive and negative samples according to the statistics of their detection scores. Positive scores are mapped to a higher range while negative ones to a lower range, thus increasing the score margins.

Model combination. For each of the two groups of proposals from DeepID-Net and CRAFT, their detection results from both tubelet re-scoring and the motion-guided propagation are each min-max mapped to $[0, 1]$ and combined by an NMS process with an IOU overlap 0.5 to obtain the final results.

3.3 Still-image object detectors

Our still-image object detectors are adopted from DeepID-Net [8] and CRAFT [32]. The two detectors have different region proposal methods, pre-trained models and training strategies.

3.3.1 DeepID-Net

Object region proposals. For DeepID-Net, the object region proposals are obtained by selective search (SS) [33] and Edge Boxes (EB) [34] with a cascaded selection process that eliminates easy false positive boxes using an ImageNet pre-trained AlexNet [24] model.

All proposal boxes are then labeled with 200 ImageNet detection class scores by the pre-trained AlexNet. The boxes whose maximum prediction scores of all 200 classes are lower than a threshold are regarded as easy negative samples and are eliminated. The process removes around 94% of all proposal boxes while obtains a recall around 90%.

Pre-trained models. ILSVRC 2015 has two tracks for each task. 1) For the **provided data** track, one can use data and annotations from all ILSVRC 2015 datasets including classification and localization (CLS), DET, VID and Places2. 2) For the **external data** track, one can use additional data and annotations.

For the provided data track, we pretrained VGG [2] and GoogLeNet [1] with batch normalization (BN) [7] using the CLS 1000-class data, while for the external data track, we used the ImageNet 3000-class data. Pre-training is done at the object-level annotation as in [8] instead of image-level annotation in R-CNN [3].

Model finetuning and SVM training. Since the classes in VID are a subset of DET classes, the DET pretained networks and SVM can be directly applied to the VID task, with correct class index mapping. However, due to the mismatch of the DET and VID data distributions and the unique statistics in videos, the DET-trained models may not be optimal for the VID task. Therefore, we finetuned the networks and re-trained the 30 SVMs with combination of DET and VID data. Different combination configurations are investigated and a $2:1$ DET to VID data ratio achieves the best performance (see Section 4.2).

Score average. Multiple CNN and SVM models are trained separately for the DeepID-Net framework, their results are averaged to generate the detection scores. Such score averaging process is conducted in a greedy searching manner. The best single model is first chosen. Then for each of the remaining models, its detection scores are averaged with those of the chosen model, and the model with best performance is chosen as the second chosen model. The process repeats until no significant improvement is observed.

3.3.2 CRAFT

CRAFT is an extension of Faster R-CNN. It contains the Region Proposal Network (RPN) stream to generated object proposals and the Fast-RCNN stream which further assigns a class (including background) score to each proposal.

Object region proposals. In this framework, we use the enhanced version of Faster-RCNN by cascade RPN and cascade Fast-RCNN. In our cascaded version of RPN, the proposals generated by the RPN are further fed into a object/background Fast-RCNN. We find that it leads to a 93% recall rate with about 100 proposals per image. In our cascade version of the Fast-RCNN, we further use a class-wise softmax loss as the cascaded step. It is utilized for hard negative mining and leads to about 2% improvement in mean AP.

Pretrained models. Similar to the DeepID-Net setting, the pretrained models are the VGG and GoogLeNet with batch normalization. We only use the VGG in the RPN step and use both models in the later Fast-RCNN classification step.

Score average. The same greedy searching is conducted for model averaging as the DeepID-Net framework.

3.4 Multi-context suppression (MCS) and motion-guided propagation (MGP)

Multi-context suppression (MCS). One limitation of directly applying still-image object detectors to videos is that they ignore the context information within a video clip. The detection results in each frame of a video should be strongly correlated and we can use such property to suppress false positive detections. We observed that although video snippets in the VID dataset may contain arbitrary number of classes, statistically each video usually contains only a few classes and co-existing classes have correlations. Statistics of all detections within a video can therefore help distinguish false positives.

Fig. 3. Multi-context suppression. For each video, the classes with top detection confidences are regarded as the high-confidence classes (green arrows) and others are regarded as low-confidence ones (red arrows). The detection scores of high-confidence classes are kept the same, while those of low-confidence ones are decreased to suppress false positives.

For example in Fig. 3, in some frames from a video clip, some false positive detections have very large detection scores. Only using the context information within these frames cannot distinguish them from the positive samples. However, considering the detection results on other frames, we can easily determine that the majority of high-confidence detections are other classes and these positive detections are outliers.

For each frame, we have about a few hundred region proposals, each of which has detection scores of 30 classes. For each video clip, we rank all detection scores on all boxes in a descending order. The classes of detection scores beyond a threshold are regarded as high-confidence classes and the rest as low-confidence classes. The detection scores of the high-confidence classes are kept the same, while those of the low-confidence classes are suppressed by subtracting a certain value. The threshold and subtracted value are greedily searched on the validation set.

Motion-guided propagation (MGP). The multi-context suppression process can significantly reduce false positive detections, but cannot recover false negatives. The false negatives are typically caused by several reasons. 1) There are no region proposals covering enough areas of the objects; 2) Due to bad pose or motion blur of an object, its detection scores are low.

These false negatives can be recovered by adding more detections from adjacent frames, because the adjacent frames are highly correlated, the detection results should also have high correlations both in spatial locations and detection scores. For example, if an object is still or moves at a low speed, it should appear at similar locations in adjacent frames. This inspires us to propagate boxes and their scores of each frame to its adjacent frame to augment detections and reduce false negatives.

We propose a motion-guided approach to propagate

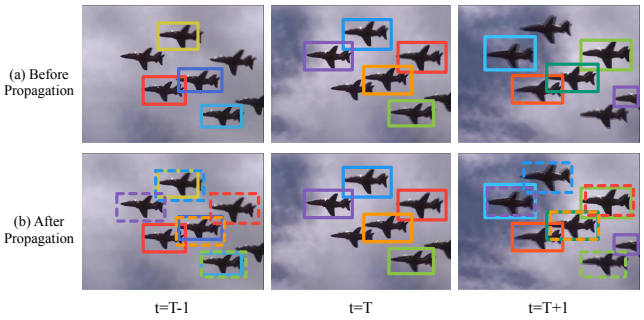

Fig. 4. Motion-guided propagation. Before the propagation, some frames may contain false negatives (e.g. some airplanes are missing in (a)). Motion-guided propagation is to propagate detections to adjacent frames (e.g. from $t = T$ to $t = T - 1$ and $t = T + 1$) according to the mean optical flow vector of each detection bounding box. After propagation, fewer false negatives exist in (b).

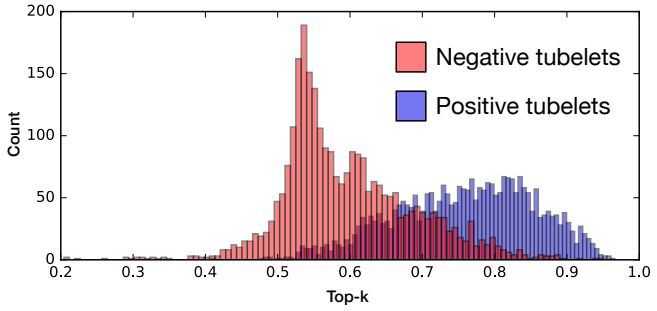

Fig. 5. Tubelet classification. Tubelets obtained from tracking can be classified into positive and negative samples using statistics (e.g. top-k, mean, median) of the detection scores on the tubelets. Based on the statistics on the training set, a 1-D Bayesian classifier is trained to classify the tubelets for re-scoring.

detection bounding boxes according to the motion information. For each region proposal, we calculate the mean optical flow vector within the bounding box of the region proposal and propagate the box coordinate with same detection score to adjacent frames according the mean flow vectors. An illustration example is shown in Fig. 4.

3.5 Tubelet re-scoring

MGP generates short dense tubelets at every detection by our still-image detectors. It significantly reduces false negatives but only incorporates short-term temporal constraints and consistency to the final detection results. To enforce long-term temporal consistency of the results, we also need tubelets that span long periods of time. Therefore, we use tracking algorithms to generate long tubelets and associate still-image object detections around tubelets.

As shown in Fig. 2, the tubelet re-scoring includes three sub-steps: 1) high confidence tracking, 2) spatial max-pooling, and 3) tubelet classification.

High-confidence tracking. For each object class in a video clip, we track high-confidence detection proposals bidirectionally over the temporal dimensions. The tracker we choose is from [30], which in our experiments shows robust performance to different object poses and scale changes. The starting bounding boxes of tracking are called "anchors", which are determined as the most confident detections. Starting from an anchor, we track biredictionally to obtain a complete tubelet. As the tracking is conducted along the temporal dimension, the tracked box may drift to background or other objects, or may not adapt to the scale and pose changes of the target object. Therefore, we stop the tracking early when the tracking confidence is below a threshold (probability of 0.1 in our experiments) to reduce false positive tubelets. After obtaining a tubelet, a new anchor is selected from the remaining detections to start a new track. Usually, high-confidence detections tend to cluster both spatially and temporally, and therefore directly tracking the next most confident detection tends to result in tubelets with large mutual overlaps on the same object. To reduce the redundancy and cover as many objects as possible, we perform a suppression process similar to NMS. Detections that have overlaps with the existing tracks beyond a certain threshold (IOU, i.e. Intersection of Union, 0.3 in our experiment) will not be chosen as new anchors. The tracking-suppression process performs iteratively until confidence values of all remaining detections are lower than a threshold. For each video clip, such tracking process is performed for each of the 30 VID classes.

Spatial max-pooling. After tracking, for each class, we have tubelets with high-confidence anchors. A naive approach is to classify each bounding box on the tubelets using still-image object detectors. Since the boxes from tracked tubelets and those from still-image object detectors have different statistics, when a still-image object detector is applied to a bounding box obtained from tracking, the detection score many not be accurate. In addition, the tracked box locations may not be optimal due to the tracking failures. Therefore, the still-image detection scores on the tracked tubelets may not be reliable.

However, the detections spatially close to the tubelets can provide helpful information. The spatial max-pooling process is to replace tubelet box proposals with detections of higher confidence by the still-image object detector.

For each tubelet box, we first obtain the detections from still-image object detectors that have overlaps with the box beyond a threshold (IOU 0.5 in our setting). Then only the detection with the maximum detection score is kept and used to replace the tracked bounding box. This process is to simulate the conventional NMS process in object detection. If the tubelet box is indeed a positive box but with low detection score, this process can raise its detection score. The higher the overlap threshold, the more trust on the tubelet box. In an extreme case when IOU = 1 is chosen as the threshold, we fully rely on the tubelet boxes while their surrounding boxes from still-image object detectors are not considered.

Tubelet classification and rescoring. High-confidence tracking and spatial max-pooling generate long sparse tubelets that become candidates for temporal rescoring. The main idea of temporal rescoring is to classify tubelets into positive and negative samples and map the detection scores into different ranges to increase the score margins.

Since the input only contains the original detection scores, the features for tubelet classification should also be simple. We tried different statistics of tubelet detection scores such as mean, median and top-k (i.e. the kth largest detection score from a tubelet). A Bayesian classifier is

TABLE 1
Performances of the still-image object detector DeepID-Net single model by using different finetuning data configurations on the initial validation set. The baseline DeepID-Net of only using the DET training data has the mean AP 49.8.

DET:VID Ratio	1:0	3:1	2:1	1:1	1:3
Mean AP / %	49.8	56.9	**58.2**	57.6	57.1

TABLE 2
Performances of the still-image object detector CRAFT single model by using different finetuning data configurations on the final validation set.

DET:VID Ratio	0:1	2:1
Mean AP / %	61.5	63.9

trained to classify the tubelets based on the statistics as shown in Fig. 5, and in our experiment, the top-k feature works best.

After classification, the detection scores of positive samples are min-max mapped to $[0.5, 1]$, while negatives to $[0, 0.5]$. Thus, the tubelet detection scores are globally changed so that the margin between positive and negative tubelets is increased.

4 Experiments

4.1 Dataset

1) The training set contains 3862 fully-annotated video snippets ranging from 6 frames to 5492 frames per snippet. 2) The validation set contains 555 fully-annotated video snippets ranging from 11 frames to 2898 frame per snippet. 3) The test set contains 937 snippets and the ground truth annotation are not publicly released.

Since the official test server is primarily used for competition and has usage limitations, we primarily report the performances on the validation set as a common convention for object detection tasks. In the end, test results from top-ranked teams participated in ILSVRC 2015 are reported.

4.2 Parameter Settings

Data configuration. We investigated the ratio of training data combination from the DET and VID training sets, and its influence on the still-image object detector DeepID-Net. The best data configuration is then used for both DeepID-Net and CRAFT. Because the VID training set has many more fully annotated frames than the DET training set, we kept all the DET images and sampled the training frames in VID for different combination ratios in order to training the still-image object detectors.

We investigated several training data configurations by finetuning a GoogLeNet with BN layers. From the Table 1 and 2, we can see that the ratio of $2:1$ between DET and VID data has the best performance on the still-image detector DeepID-Net and CRAFT single models, therefore, we finetuned all of our models using this data configuration.

In addition to model finetuning, we also investigated the data configurations for training the SVMs in DeepID-Net. The performances are shown in Table 3, which show that using positive and negative samples from both DET and VID data leads to the best performance.

TABLE 3
Performances of different data configurations on the validation set for training SVMs in DeepID-Net. Baseline (the first column) only uses DET positive and negative samples and the result is a mean AP of 49.8.

DET Positive	✓	✓	✗	✗	✗	✓
VID Positive	✗	✓	✓	✓	✓	✓
DET Negative	✓	✓	✓	✓	✗	✓
VID Negative	✗	✗	✗	✓	✓	✓
mean AP / %	49.8	47.1	35.8	51.6	52.3	**53.7**

TABLE 4
Performances on the validation set by different temporal window sizes of MGP.

Methods	Temporal window size				
	1 (baseline)	3	5	7	9
Duplicate	70.7	71.7	72.1	71.5	-
Motion-guided		71.7	**73.0**	**73.0**	72.6

Because of the redundancy among video frames, we also sampled the video frames by a factor of 2 during testing and applied the still-image detectors to the remaining frames. The MCS, MGP and re-scoring steps in Section 3.4 and 3.5 are then conducted. The detection boxes on the unsampled frames are generated by interpolation and MGP. We did not observe significant performance differences with frame sampling on the validation set.

To conclude, we sampled VID frames to half the amount of DET images and combined the samples to finetune the CNN models in both DeepID-Net and CRAFT. Positive and negative samples from both DET and VID images are used to train SVMs in DeepID-Net.

Hyperparameter settings. For motion-guided propagations, as described in Section 3.4, Table 4 shows the performances of different propagation window sizes. Compared to directly duplicating boxes to adjacent frames without changing their locations according to optical flow vectors, MGP has better performances with the same propagation durations, which proves that MGP generates detections with more accurate locations. 7 frames (3 frames forward and 3 backward) are empirically set as the window size.

In multi-context suppression, classes in the top 0.0003 of all the bounding boxes in a video are regarded as high-confidence classes and the detection scores for both frameworks are subtracted by 0.4. Those hyperparameters are greedily searched in the validation set.

Network configurations. The models in DeepID-Net and CRAFT are mainly based on GoogLeNet with batch-normalization layers and VGG models. The techniques of multi-scale [35] and multi-region [13] are used to further increase the number of models for score averaging in the still-image object detection shown in Fig. 2. The performance of a baseline DeepID-Net trained on ImageNet DET task can be increased from 49.8 to 70.7 with all the above-mentioned techniques (data configuration for finetuning, multi-scale, multi-region, score average, etc.).

4.3 Results

Qualitative results. Some qualitative results of our proposed framework are shown in Fig. 6. From the figure, we

TABLE 5
Performances of individual components, frameworks and our overall system.

Data	Model	Still-image	MCS+MGP +Rescoring	Model Combination	Test Set	Rank in ILSVRC2015	#win
Provided	CRAFT [32]	67.7	73.6	73.8	67.8	#1	28/30
	DeepID-net [8]	65.8	72.5				
Additional	CRAFT [32]	69.5	75.0	77.0	69.7	#2	11/30
	DeepID-net [8]	70.7	75.4				

TABLE 6
Performaces of our submitted models on the validation set.

Method	airplane	antelope	bear	bicycle	bird	bus	car	cattle	dog	d_cat	elephant	fox	g_panda	hamster	horse	lion
Provided	83.70	85.70	84.40	74.50	73.80	75.70	57.10	58.70	72.30	69.20	80.20	83.40	80.50	93.10	84.20	67.80
Additional	85.90	86.90	87.80	77.90	74.70	77.50	59.00	70.90	74.40	79.60	80.40	83.90	82.40	95.80	87.80	64.10

Method	lizard	monkey	motorcycle	rabbit	r_panda	sheep	snake	squirrel	tiger	train	turtle	watercraft	whale	zebra	mean AP	#win
Provided	80.30	54.80	80.60	63.70	85.70	60.50	72.90	52.70	89.70	81.30	73.70	69.50	33.50	90.20	**73.80**	28/30
Additional	82.90	57.20	81.60	77.50	79.70	68.00	77.70	58.30	90.10	85.30	75.90	71.20	43.20	91.70	**77.00**	11/30

can see the following characteristics of our proposed framework. 1) The bounding boxes are very tight to the objects, which results from the high-quality bonding box proposals combined from Selective Search, Edge Boxes and Region-proposal Networks. 2) The detections are consistent across adjacent frames without obvious false negatives thanks to the motion-guided propagation and tracking. 3) There are no obvious false positives even though the scenes may be complex (e.g. cases in the third row), because the multi-context information is used to suppress their scores.

Quantitative results The component analysis of our framework on both provided-data and additional-data tracks are shown in Table 5. The results are obtained from the validation set. From the table, we can see that the still-image object detectors obtain about $65-70\%$ mean AP. Adding temporal and contextual information through MCS, MGP and tubelet re-scoring significantly improves the results by up to 6.7 percents. The final model combination process further improves the performance.

Overall, our framework ranks 1st on the provided-data track in ILSVRC2015 winning 28 classes out of 30 and 2nd on the additonal-data track winning 11 classes. The detailed AP lists of the submitted models on the validation set are shown in Table 6. The final results of our team and other top-ranked teams on the test data are shown in Table 7.

TABLE 7
Performance comparison with other teams on ILSVRC2015 VID test set with provided data (sorted by mean AP, the best model is chosen for each team).

Rank	Team name	mean AP	#win
1	**CUVideo (Ours)**	**67.82**	**28**
2	ITLab VID - Inha	51.50	0
3	UIUC-IFP [36]	48.72	0
4	Trimps-Soushen	46.12	0
5	1-HKUST	42.11	0
6	HiVision	37.52	0
7	RUC_BDAI	35.97	2

5 Conclusion

In this work, we propose a deep learning framework that incorporates temporal and contextual information into object detection in videos. This framework achieved the state of the art performance on the ImageNet object detection from video task and won the corresponding VID challenge with provided data in ILSVRC2015. The component analysis is investigated and discussed in details. Code is publicly available.

The VID task is still new and under-explored. Our proposed framework is based on the popular still-image object detection frameworks and adds important components specifically designed for videos. We believe that the knowledge of these components can be further incorporated in to end-to-end systems and is our future research direction.

References

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

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Int'l Conf. Learning Representations*, 2014.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *CVPR*, 2014.

[4] R. Girshick, "Fast r-cnn," *ICCV*, 2015.

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

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

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

[8] W. Ouyang, X. Wang, X. Zeng, S. Qiu, P. Luo, Y. Tian, H. Li, S. Yang, Z. Wang, C.-C. Loy *et al.*, "DeepID-net: Deformable deep convolutional neural networks for object detection," *CVPR*, 2015.

[9] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid, "Spatio-temporal Object Detection Proposals," *ECCV*, 2014.

[10] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229*, 2013.

[11] D. Erhan, C. Szegedy, A. Toshev, and D. Anguelov, "Scalable object detection using deep neural networks," 2014.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *arXiv preprint arXiv:1506.02640*, 2015.

[13] S. Gidaris and N. Komodakis, "Object detection via a multi-region and semantic segmentation-aware cnn model," in *ICCV*, 2015.

[14] R. Girshick, F. Iandola, T. Darrell, and J. Malik, "Deformable part models are convolutional neural networks," in *CVPR*, 2015.

[15] D. Mrowca, M. Rohrbach, J. Hoffman, R. Hu, K. Saenko, and T. Darrell, "Spatial semantic regularisation for large scale object detection," in *ICCV*, 2015.

[16] S. Bell, C. L. Zitnick, K. Bala, and R. Girshick, "Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks," *arXiv preprint arXiv:1512.04143*, 2015.

[17] J. Yan, Y. Yu, X. Zhu, Z. Lei, and S. Z. Li, "Object detection by labeling superpixels," in *CVPR*, 2015.

[18] A. Prest, C. Leistner, J. Civera, C. Schmid, and V. Ferrari, "Learning object class detectors from weakly annotated video," *CVPR*, 2012.

[19] A. Papazoglou and V. Ferrari, "Fast Object Segmentation in Unconstrained Video," *ICCV*, 2013.

[20] A. Joulin, K. Tang, and L. Fei-Fei, "Efficient Image and Video Co-localization with Frank-Wolfe Algorithm," *ECCV*, 2014.

[21] S. Kwak, M. Cho, I. Laptev, J. Ponce, and C. Schmid, "Unsupervised Object Discovery and Tracking in Video Collections," *ICCV*, 2015.

[22] T. Deselaers, B. Alexe, and V. Ferrari, "Localizing Objects While Learning Their Appearance," *ECCV*, 2010.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," *CVPR*, 2009.

[24] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *NIPS*, 2012.

[25] K. Simonyan and A. Zisserman, "Two-Stream Convolutional Networks for Action Recognition in Videos," in *Conference on Neural Information Processing Systems*, 2014.

[26] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[27] H. Possegger, T. Mauthner, P. M. Roth, and H. Bischof, "Occlusion geodesics for online multi-object tracking," *CVPR*, 2014.

[28] Y. Li, J. Zhu, and S. C. Hoi, "Reliable patch trackers: Robust visual tracking by exploiting reliable patches," *CVPR*, 2015.

[29] Z. Hong, Z. Chen, C. Wang, X. Mei, D. Prokhorov, and D. Tao, "Multi-store tracker (muster): a cognitive psychology inspired approach to object tracking," *CVPR*, 2015.

[30] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," *ICCV*, 2015.

[31] H. Nam and B. Han, "Learning multi-domain convolutional neural networks for visual tracking," *arXiv:1510.07945*, 2015.

[32] B. Yang, J. Yan, Z. Lei, and S. Z. Li, "Craft objects from images," *CVPR*, 2016.

[33] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," *IJCV*, 2013.

[34] C. L. Zitnick and P. Dollar, "Edge Boxes: Locating Object Proposals from Edges," *ECCV*, 2014.

[35] X. Zeng, W. Ouyang, and X. Wang, "Window-object relationship guided representation learning for generic object detections," *arXiv preprint arXiv:1512.02736*, 2015.

[36] 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.

Fig. 6. Qualitative results. The bounding boxes are tight to objects because of the combination of different region proposals. The detection results are consistent across adjacent frames thanks to motion-guided propagation and tracking. The false positives are much eliminated by multi-context suppression. (Different colors are used to mark bounding boxes in the same frame and do not represent tracking results)