2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

TSSD: Temporal Single-Shot Detector Based on Attention and LSTM

Xingyu Chen, Zhengxing Wu, and Junzhi Yu

***Abstract*— Temporal object detection has attracted significant attention, but most popular methods can not leverage the rich temporal information in video or robotic vision. Although many different algorithms have been developed for video detection task, real-time online approaches are frequently deficient. In this paper, based on attention mechanism and convolutional long short-term memory (ConvLSTM), we propose a temporal single-shot detector (TSSD) for robotic vision. Distinct from previous methods, we aim to temporally integrate pyramidal feature hierarchy using ConvLSTM, and design a novel structure including a high-level ConvLSTM unit as well as a low-level one (HL-LSTM) for multi-scale feature maps. Moreover, we develop a creative temporal analysis unit, namely, ConvLSTM-based attention and attention-based ConvLSTM (A&CL), in which the ConvLSTM-based attention is specially tailored for background suppression and scale suppression while the attention-based ConvLSTM temporally integrates attention-aware features. Finally, our method is evaluated on ImageNet VID dataset. Extensive comparisons on detection performance confirm the superiority of the proposed approach, and the developed TSSD achieves a considerably enhanced accuracy vs. speed trade-off, i.e., 64.8% mAP vs. 27 FPS.**

I. Introduction

With rapid development of computer vision, the robot is competent in visually perceiving environments gradually. For example, Nguyen et al. designed an affordances detection method to help a robot plan grasp [1]. On the other hand, object detection is one of the important vision tasks. However, recent works have largely focused on detecting in static images, so they are not suited to temporally concordant visual tasks. Thus, it is essential to develop an approach to integrate spatial features with temporal information for robot's intelligent perception.

Taking advantage of convolutional neural network (CNN), existing detection methods can be divided into two categories, i.e., two-stage and one-stage detectors. The former is represented by RCNN family [2]–[5] and RFCN [6], all of which detect objects based on region proposal. On the other hand, regression and classification are computed simultaneously in one-stage pipelines, e.g., YOLO [7], SSD [8], RetinaNet [9], etc. In particular, making use of convolution features more effectively, SSD is one of the first methods that adopt the pyramidal feature hierarchy for detection. Considering the two-stage detectors have better detection accuracy and their region proposal part can be generalized to process consecutive frames, researchers tend to apply two-stage detection methods to video detection task. However, one-stage approaches have advantageous inference speed. Therefore, it is necessary to study temporal performance of one-stage detectors to take into account accuracy vs. speed trade-off for robotic applications.

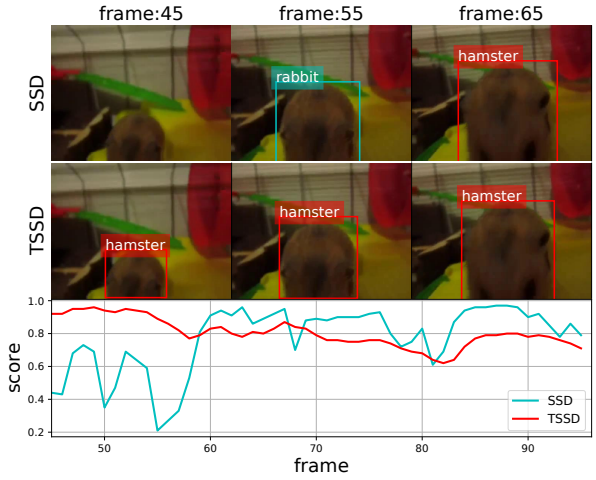

Fig. 1. Toy example. This is a changing video snippet containing a hamster. With large temporal fluctuations in terms of detection score, SSD's results contain the false positive and false negative, whereas the performance of TSSD is more stable and accurate.

Recurrent neural network (RNN) has achieved great success in sequence processing tasks, and typically, long short-term memory (LSTM) is proposed for longer sequence learning [10]. Recently, Shi et al. developed convolutional LSTM (ConvLSTM) to associate LSTM with spatial structure [11]. However, the total amount of convolution features for detection is very huge, especially when pyramidal feature hierarchy is adopted. Thus, a temporal model for multi-scale feature maps is becoming urgently necessary. Moreover, since background takes up most of an image, only a small part of visual features devote themselves to detecting targets. Thus, the feature selection is a pivotal step. Fortunately, attention mechanism is an exciting idea which imitates human's cognitive patterns, promoting CNN concern something essential. For example, Mnih et al. proposed a recurrent attention model to find the most suitable local feature for classification [12]. Nevertheless, attention model for image-sequence detection has not yet been widely studied.

In this paper, aiming at detecting objects in consecutive vision, we propose a temporal detection model based on SSD, namely, temporal single-shot detector (TSSD), whose

This work was supported by the National Natural Science Foundation of China (nos. 61633004, 61633020, 61603388, 61633017, and 61725305), and by the Beijing Natural Science Foundation (no. 4161002).

X. Chen, Z. Wu, and J. Yu are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China and University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: {chenxingyu2015, zhengxing.wu, junzhi.yu}@ia.ac.cn).

toy example is illustrated in Fig. 1. To integrate visual features through time, ConvLSTM is employed for temporal information. Due to the pyramidal feature hierarchy for multi-scale detection, SSD always generates a large body of visual features for static regression and classification, so a ConvLSTM is hard to integrate these multi-scale feature maps. Then, if ConvLSTMs were utilized for each feature map, the dramatically increasing parameters could heavily raise model complexity. Thus, we design a new structure including a high-level ConvLSTM unit as well as a low-level one (HL-LSTM) for multi-scale features. Furthermore, according to above analyses, a more crucial problem is that only a small part of visual features are related to targets. Thereby, we propose a creative module to integrate spatiotemporal information, namely, ConvLSTM-based attention and attention-based ConvLSTM (A&CL), in which overfull useless information (i.e., multi-scale background features) is prevented from being fed to ConvLSTM. Finally, facilitated by HL-LSTM and A&CL, the TSSD achieves considerable accuracy vs. speed trade-off. The contributions made in this paper are summarized as follows:

- To temporally integrate pyramidal feature hierarchy, we design an HL-LSTM structure to effectively propagate multi-scale visual features through time.
- We propose an A&CL module as a temporal analysis unit, in which redundant information is reduced.
- We achieve a considerably improved result on ImageNet VID dataset, i.e., a mean average precision (mAP) of 64.8%, and an average inference speed of 27 frames pre second (FPS).

II. Related Work

At the beginning, static detection and post-proposing methods were combined to counteract video detection task [13]–[15]. They statically detected in each video frame, then comprehensively delt with multi-frame results. Kang et al. developed TCNN based on tubelet (i.e., temporally propagative bounding boxes) using still-image object detection, multi-context suppression, motion guided propagation, and temporal tubelet re-scoring [13], [14]. Taking inspiration from non-maximum suppression (NMS), Han et al. proposed SeqNMS to suppress temporally discontinuous bounding boxes [15]. However, due to complex post-processing, the time efficiency decreases, and such methods did not improve the performance of detector.

Faster RCNN used region proposal network (RPN) for coarse localization [4], so some approaches for videos tried to enhance RPN with temporal information [16]–[18]. Galteri et al. designed a closed-loop RPN to fuse current object proposal with previous detection results. This method effectively reduced the number of invalid region, but it could also make the proposed regions excessively concentrated. Kang et al. developed tubelet proposal networks (TPN) to propose tubelets rather than bounding boxes. Then, an encoder-decoder LSTM is used for classification. Such methods were extended from two-stage detectors, so they still suffered from low time efficiency.

Object tracking is able to localize targets in a video with the prior knowledge of the initial position. Feichtenhofer et al. combined RFCN detector with correlation-filter-based tracker to detect objects in videos, called D&T [19]. Thanks to the tracking method, D&T achieved high detection accuracy, but obviously, the RFCN in D&T is not capable of temporal analysis. Moreover, correlation filters could hardly work in real time, especially for numerous objects.

Object detector and RNN have been applied comprehensively in recent years [20], [21]. Ning et al. proposed ROLO for tracking based on YOLO and LSTM. The YOLO was responsible for static detection, and the visual features and positions of high-score objects were fed to LSTM for temporally modeling. Lu et al. employed SSD for static detection, and similarly, temporal relations of high-score objects were modeled using association LSTM. Although they were unified in such methods, RNN merely worked as a post-processing for detection results.

III. Approach

A. Architecture

Extending form SSD with VGG-16 [22] as the backbone, we build a temporal single-shot detector for robotic vision. Fully connected layers $fc6, fc7$ in original VGG-16 are converted to convolutional layers, namely, $Conv6, Conv7$. The network predicts bounding boxes and corresponding classification scores, followed by NMS for final detection. Additionally, as illustrated in Fig. 2, the novel HL-LSTM and A&CL are designed for temporal information.

1) HL-LSTM: There are six-scale pyramidal features in SSD model, whose sizes are $38 \times 38 \times 512$, $19 \times 19 \times 512$, $10 \times 10 \times 512$, $5 \times 5 \times 256$, $3 \times 3 \times 256$, and $1 \times 1 \times 256$. Creatively, we group the multi-scale feature maps according to the order of different convolutional layers, i.e., high-level and low-level features (shown in gold and red in Fig. 2). Further, we use the same two structures to integrate them temporally, called HL-LSTM. We aim to address two problems with the HL-LSTM: i) Avoiding redundant parameters. For example, the original SSD contains 2.6 M parameters, and SSD with HL-LSTM has 4.9 M parameters. However, if six ConvLSTMs are employed, the parameter size increases to 15.5 M; ii) Simplifying training process. As reported in [21], the highest- and lowest-level feature maps make relatively less contribution to detection. That is, there are a small amount of data for oversized or tiny-size objects. Thus, if six-scale ConvLSTMs were employed, the highest- and lowest-level ConvLSTM would be hard to train.

2) A&CL: In object detection task, most features are related to background, and additionally, feature maps in different scales contribute to detection in different degrees. Therefore, it is inefficient when a ConvLSTM handles background or aforementioned small-contributed multi-scale feature maps. In this paper, we propose A&CL for background suppression and scale suppression, including ConvLSTM-based attention and attention-based ConvLSTM. Attention module selects object-related features for ConvLSTM, and in turn, the ConvLSTM provides attention module with temporal information

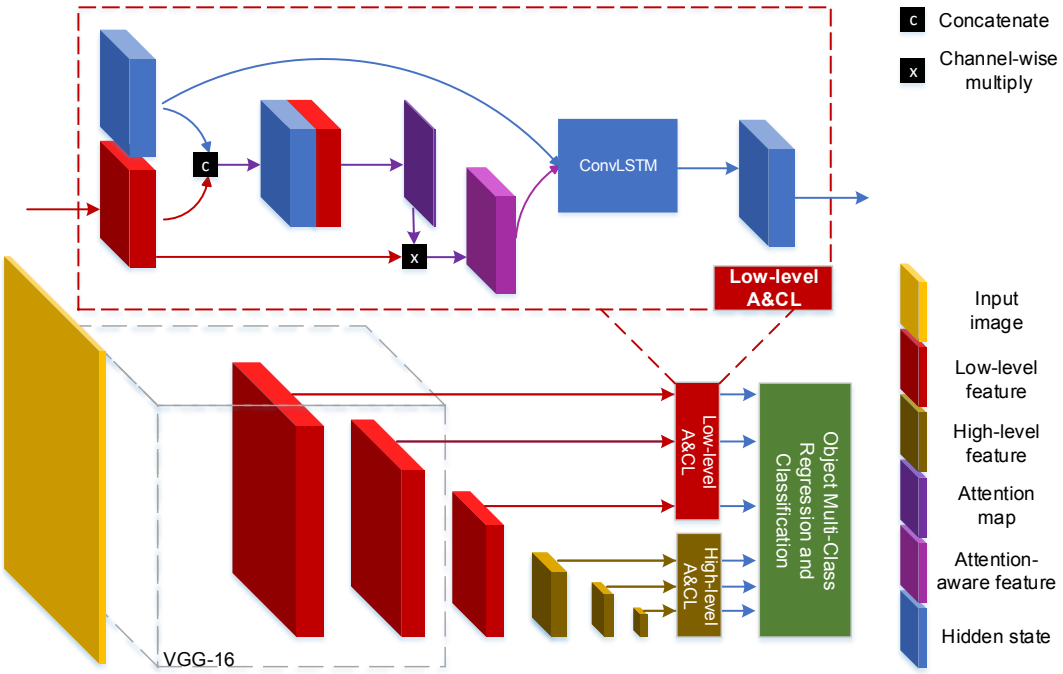

Fig. 2. The proposed TSSD architecture. For better visualization, we only display the layers used for detection. The high-level features share a temporal analysis unit and low-level features do so, namely, HL-LSTM. An attention module and ConvLSTM jointly integrate the temporal features in A&CL. Finally, the hidden state will be used for multi-box regression and classification.

to improve attention precision. As shown in Fig. 2, one of the low-level feature maps serves as the input of low-level A&CL. Instead of computing attention map using the current feature, we firstly concatenate the input with previous information, followed by the produce of a temporal attention map. Subsequently, ConvLSTM integrates current attention-aware feature with previous information more effectively, and the current hidden state will be used for multi-box regression and classification. As a temporal analysis unit, A&CL can be formulated as follows:

$$
\begin{aligned}
a_t &= \sigma(W_a * [x, h_{t-1}]) \\
i_t &= \sigma(W_i * [a_t \circ x, h_{t-1}] + b_i) \\
f_t &= \sigma(W_f * [a_t \circ x, h_{t-1}] + b_t) \\
o_t &= \sigma(W_o * [a_t \circ x, h_{t-1}] + b_o) \\
c_t &= \tanh(W_c * [a_t \circ x, h_{t-1}] + b_c) \\
s_t &= (f_t \odot s_{t-1}) + (i_t \odot c_t) \\
h_t &= o_t \odot \tanh(s_t),
\end{aligned}
\tag{1}
$$

where $*$ denotes convolution operation; $[\cdot, \cdot]$ is concatenation; $\odot$ is element-wise multiplication; and $\circ$ represents that a one-channel map multiplies with each channel in a multi-channel feature map. At time step t, a_t, h_t, i_t, f_t, o_t, c_t, s_t are attention map, hidden state, input gate, forget gate, output gate, LSTM's new information, and memory, respectively. σ represents sigmoid activation function.

In detail, referring Fig. 3, the A&CL is designed with CNN and RNN. Current feature map (x) and previous hidden state (h) serve as the input of ConvLSTM-based attention. After three-layer convolution, a one-channel temporal attention map (a) is generated, which contains pixel-wise positions for object-related features. For feature selection, each channel in current feature map multiplies this attention map pixel-by-pixel, and the attention-aware feature ($a \circ x$) can be obtained.

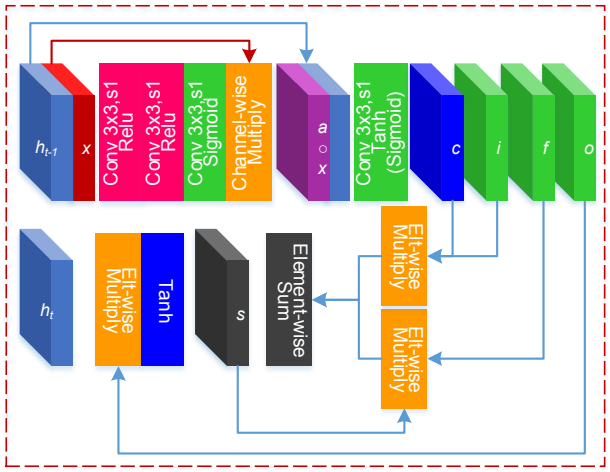

Fig. 3. Implementation detail of A&CL, including ConvLSTM-based attention and attention-based ConvLSTM. The symbol system follows (1).

The attention-aware feature and previous hidden state are concatenated as the input of attention-based ConvLSTM. Different from traditional LSTM, gates (i, f, o) and incoming information (c) are computed with convolution operation. Subsequently, controlled by the gates, the temporal memory (s) is updated, and new hidden state is generated for multi-box regression and classification.

B. Training

1) Basic setting: At first, we train an SSD model following [8]. Then, the TSSD is trained based on well-trained SSD. In particular, the ConvLSTM is trained with RMSProp optimizer while the rest of TSSD is trained using SGD optimizer. The initial learning rate is 10^{-4} for the first 40 epochs, and we use 10^{-5} for another 20 epochs.

Moreover, the TSSD should be trained with a frame sequence, but it should not be trained frame by frame for better generalization. Instead, we only choose seq_len frames in a video for back propagation in an iteration. The seq_len frames should be chosen uniformly based on the start frame (sf) and skip (sp),

$$\begin{aligned} sp &= R[1, v/seq\_len] \\ sf &= R[1, v - seq\_len * sp + 1], \end{aligned} \tag{2}$$

where v is the frame number in a video, and $R[min, max]$ represents the operation of selecting an integer randomly between min and max. Finally, the uniform seq_len frames are chosen with sf as the start frame and sp as the skip. In this paper, $seq\_len = 8$.

2) Objective: We design a multi-term objective to train the TSSD, including a localization loss $\mathcal{L}_{loc}$, a confidence loss $\mathcal{L}_{conf}$, and an attention loss $\mathcal{L}_{att}$,

$$\mathcal{L} = \frac{1}{N}(\alpha\mathcal{L}_{loc} + \beta\mathcal{L}_{conf}) + \gamma\mathcal{L}_{att}, \tag{3}$$

where N is the number of positive boxes, and $\mathcal{L}_{loc}$ and $\mathcal{L}_{conf}$ are inherited from SSD [8].

We also supervise the generation of attention maps using cross entropy. At first, we construct the ground truth attention map A_g, in which elements locating within ground truth boxes equal to 1 and others are assigned to 0. Then, six-scale attention maps A_{p_s} is unified to the same resolution as the input image through bilinear upsampling, followed by the produce of $A^{up}_{p_s}$. Hence, $\mathcal{L}_{att}$ can be given as

$$\mathcal{L}_{att} = \sum_{s=1}^{6} \mu(-A^{up}_{p_s}\log(A_g) - (1 - A^{up}_{p_s})\log(1 - A_g)), \tag{4}$$

where μ averages all elements of a matrix. The hyper parameters $\alpha = 1, \beta = 1, \gamma = 0.5$ are selected based on the performance of validation set.

3) Inference: At inference phase, the netowrk regresses and classifies objects frame by frame with HL-LSTM and A&CL, and outputs confident object candidates (confident scores > 0.01). Subsequently, these candidates are processed by NMS with 0.45 jaccard overlap pre class and retain top 200 high confident objects as the final detections.

IV. Experiment

A. Dataset

We evaluate the TSSD on the ImageNet dataset for object detection from video (VID) [23], which requires algorithms detect 30-class targets in consecutive frames. There are 4000 videos in the training set, containing $1,181,113$ frames. On the other hand, the validation set compasses 555 videos, including $176,126$ frames. We measure performance as mAP over the 30 classes on the validation set following [4], [8].

In addition, ImageNet DET dataset is employed as training assistance. The 30 categories in VID dataset are a subset of the 200 categories in the DET dataset. Therefore, following [13], [14], [18], [19], we sample at most $2,000$ images per class from DET (only using the data from the 30 VID classes), and select 10 frames in each VID video for SSD training. Then, the TSSD is trained using the whole VID training set.

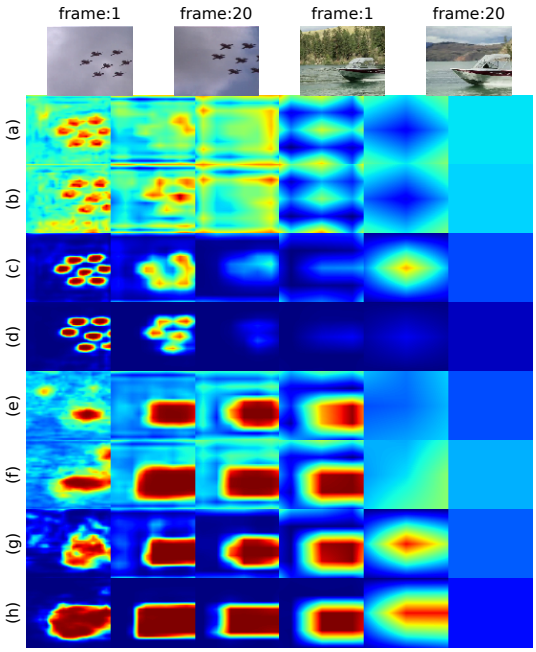

Fig. 4. Multi-scale attention maps. There are two video snippets containing airplanes or watercraft. (a),(b),(e),(f) are generated by traditional module, whereas (c),(d),(g),(h) are produced by ConvLSTM-based attention module. (a)–(d) are attention maps for airplanes; (e)–(h) are attention maps for watercraft. In above 4 pairs maps, the former is for the first frame while the latter is with respect to the 20th frames. Each line is multi-scale attention maps, and higher-level maps are displayed on the righter.

B. Run Time Performance

Our method is implemented under the PyTorch [1] framework. The training and experiments are carried out on a workstation with an Intel 2.20 GHz Xeon(R) E5-2630 CPU, a NVIDIA TITAN-Xp GPU, and 64 GB of RAM. As a result, our proposed TSSD reaches 27 FPS on VID validation set, so it is capable of real-work applications.

C. Results

1) Attention Results: As shown in Fig. 4, the comparison of ConvLSTM-based attention and traditional attention module are presented. Note that the traditional attention module only uses current feature map as the input. In presented heat maps, crimson means a higher probability of being a target, whereas mazarine indicates ignorable pixel position in feature maps. Moreover, multi-scale attention maps are

[1] https://pytorch.org

TABLE I

AP LIST ON IMAGENET VID VALIDATION SET BY THE PROPOSED METHOD AND COMPARED METHODS.

Method	airplane	antelope	bear	bicycle	bird	bus	car	cattle	dog	d.cat	elephant
SSD	82.01	72.67	71.62	60.19	65.54	68.77	56.86	59.79	47.69	63.88	72.48
SSD+Attention	81.74	75.69	72.24	58.71	62.46	67.40	57.38	65.64	48.41	63.92	69.94
SSD+ConvLSTM	79.86	75.06	68.75	62.60	63.38	69.08	59.78	58.34	48.96	63.66	69.97
TSSD	82.16	76.03	68.88	61.57	66.26	70.04	59.39	67.07	49.18	63.29	71.55
Method	fox	g.panda	hamster	horse	lion	lizard	monkey	m.bike	rabbit	r.panda	sheep
SSD	77.47	79.04	89.04	61.53	26.43	61.34	41.78	73.58	49.20	20.96	58.99
SSD+Attention	74.57	78.78	89.34	62.81	21.80	57.21	40.70	75.69	55.39	35.57	53.67
SSD+ConvLSTM	80.61	78.03	90.12	62.53	28.17	62.15	41.25	75.69	54.33	44.90	56.19
TSSD	80.85	80.71	90.18	63.36	30.21	64.61	41.50	75.81	56.00	39.85	57.26
Method	snake	squirrel	tiger	train	turtle	w.craft	whale	zebra	FPS	mAP(%)	
SSD	47.95	47.11	80.71	76.98	69.07	61.61	63.54	83.34	∼45	63.04	
SSD+Attention	51.90	45.62	79.28	76.92	69.71	62.30	57.16	82.29	∼32	63.14	
SSD+ConvLSTM	46.41	45.95	81.18	76.03	70.33	62.56	58.65	83.96	∼38	63.95	
TSSD	49.34	46.41	82.45	77.68	71.49	62.02	54.58	83.20	∼27	**64.76**	

generated in TSSD, the righter maps response higher-level features. For the ease of observation, the multi-scale attention maps have been unified to the same resolution as the input image through bilinear upsampling.

As shown in Fig. 4(a), (b), (e), (f), the original attention method is not able to handle these two scenes. That is, although the targets are focused roughly, the background and small-contributed multi-scale feature maps are not suppressed effectively. On the contrary, as illustrated in Fig. 4(c), (d), (g), (h), ConvLSTM-based attention performs better. The proposed attention method not only localizes the targets more accurately, but also suppresses the background more efficiently. Further, our method is effective for scale suppression (see Fig. 4(d)). In addition, the performance of proposed approach improves along with the accumulation of temporal information. For example, in Fig. 4(g), (h), the lowest-level attention map can hardly find the watercraft in the first frame, but it is focused without overmuch background in the 20th frame. Moreover, if attention maps for the first frame are compared, a conclusion can be drawn that ConvLSTM-based attention is better even though the temporal information has not generated owing to more effective training.

2) Results on ImageNet VID Dataset: We evaluate our method with VID validation dataset using mAP. At first, SSD is employed as the baseline. Then, we employ ConvLSTM following HL-LSTM and denote the result as "SSD+ConvLSTM". Subsequently, attention module is adopted alone, where the attention-aware feature is utilized for regression and classification, followed by the result being called "SSD+Attention". Finally, the result of proposed TSSD including HL-LSTM and A&CL are presented.

As shown in Table I, SSD achieves 63.04% mAP, and if the attention module or ConvLSTMs is added, the mAP for detection will increase (i.e., 63.14% and 63.95%). Further, if A&CL works, a better performance is obtained by the TSSD, i.e., 64.76% mAP. Moreover, AP for particular classes increase dramatically. For example, AP for "lion" raises by about 4 points. Since SSD's AP for "lion" is quite low, we deem this phenomenon is caused by imbalanced data. That is, the amount of training data for "lion" is relatively small, so it is likely to be assigned to other categories with similar features by the static detector. If the temporal information is considered, this phenomenon is relieved to some extent. On the other hand, a small part of classes lose AP. For example, the AP for "whale" decreases by about 9%. Since whales successively emerge and submerge from the water, temporal information may mislead the detector about their appearance and disappearance. Some typical temporal detection results are shown by our supplemental video [2].

We also compare the TSSD against several prior and contemporary approaches. As shown in Table II, their components and performances have been summarized. Most methods are based on two-stage detector with RPN. In addition, few approaches successfully adopt attention or LSTM for temporal coherence, especially ConvLSTM. On the other hand, tracking employed in TCN [13], TCNN [14], and D&T [19] is a good idea for high recall rate, but it affects time efficiency and model complexity. In terms of perfoemance, few methods can run in real time, and additionally, the TPN uses previous and further frames for current frame detection, so it is not an online detector. Thus, the real-time online TSSD is able to temporally detect targets for robotic applications.

D. Discussion

The TSSD achieves a great improvement in terms of temporal inference speed, but its mAP is modest when compared to state-of-the-art approaches. The main reason is two-fold: 1) the two-stage methods with RPN are more precise than one-stage detector, and 2) tracking employed in TCNN [14] and D&T [19] is in favor of recall rate. For high computational efficiency, aforementioned two advantageous processes can not been employed in the TSSD, but in our opinion, adding loss function in an inter-frame manner to model object association could further enhance the accuracy.

Furthermore, besides object detection, object identification is also imperative for robotic application. Therefore, tracking by detection should be further studied, and the TSSD is well suited to serve as the detection component owing to its real-time online characteristic. Then, the detection results should

[2] https://youtu.be/A6Z8A6NF6nc

TABLE II

COMPARISON OF THE TSSD AND SEVERAL PRIOR AND CONTEMPORARY APPROACHES.

Method	Components						Performances		
	One stage	Two stage	Optical flow	Tracking	Attention	LSTM	Real-time	Online	mAP
Closed-loop [16]		✓						✓	50.0
Seq-NMS [15]								✓	52.2
TCN [13]		✓		✓				✓	47.5
T-CNN [14]		✓	✓	✓				✓	61.5
TPN [18]		✓				✓			68.4
D&T [19]		✓		✓				✓	79.8
TSSD(propsed)	✓				✓	✓	✓	✓	64.8

be associated through time, but current real-time multi-object tracking methods (e.g., SORT [24]) can hardly leverage within-class-similar features in detector for identification. However, in our opinion, multi-scale attention maps in the TSSD are suitable for this task, because they are individual-aware and computationally inexpensive.

V. CONCLUSION

This paper has aimed at temporally detecting objects in real time for consecutive vision, and a creative TSSD approach has been proposed. Differing from existing video detection methods, the TSSD is a temporal one-stage detector, and it can perform well in terms of both detection accuracy and speed. To efficiently integrate pyramidal feature hierarchy, an HL-LSTM is proposed, in which high-level and low-level features share their respective ConvLSTM units. For background suppression and scale suppression, attention mechanism is employed to reduce information redundancies. Thereby, we design A&CL as a temporal analysis unit, where ConvLSTM-based attention is responsible for selecting object-related features for attention-based ConvLSTM. As a result, the TSSD achieves considerably enhanced accuracy vs. speed trade-off on ImageNet VID dataset. Furthermore, owing to its real-time online characteristic, the TSSD is well-suited to robot's intelligent perception.

In the future, objects in a video will be identified, and the TSSD will be used for robotic visual navigation under dynamic environments.

REFERENCES

[1] A. Nguyen, D. Kanoulas, D. G. Caldwell, and N. G. Tsagarakis, "Object-based affordances detection with convolutional neural networks and dense conditional random fields," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Vancouver, BC, Canada, Sep. 2017, pp. 24–28.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, Columbus, USA, Jun. 2014, pp. 580–587.

[3] R. Girshick. "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis.*, Santiago, Chile, Dec. 2015, pp. 1440–1448.

[4] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Info. Proc. Syst.*, Montreal, Canada, Dec. 2015, pp. 91–99.

[5] K. He, G. Gkioxari, P. Dollar, and R. Girshick. "Mask R-CNN," Venice, Italy, Oct. 2017, pp. 2961–2969.

[6] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," in *Proc. Adv. Neural Info. Proc. Syst.*, Barcelona, Spain, Dec. 2016, pp. 379–387.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, Las Vegas, USA, Jun. 2016, pp. 779–788.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, Netherlands, Oct. 2016, pp. 21–37.

[9] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, Oct. 2017, pp. 2980–2988.

[10] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[11] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Info. Proc. Syst.*, Montreal, Canada, Dec. 2015, pp. 802–810.

[12] V. Mnih, N. Heess, and A. Graves, "Recurrent models of visual attention," in *Proc. Adv. Neural Info. Proc. Syst.*, Montreal, Canada, Dec. 2014, pp. 2204–2212.

[13] K. Kang, W. Ouyang, H. Li, and X. Wang, "Object detection from video tubelets with convolutional neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, Las Vegas, USA, Jun. 2016, pp. 817–825.

[14] K. Kang, H. Li, J. Yan, X. Zeng, B. Yang, T. Xiao, C. Zhang, Z. Wang, R. Wang, X. Wang, and W. Ouyang, "T-CNN: Tubelets with convolutional neural networks for object detection from videos," *IEEE Trans. Circuits Syst. Video Technol.*, DOI:10.1109/TCSVT.2017.2736553.

[15] 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:1602.08465*, 2016.

[16] L. Galteri, L. Seidenari, M. Bertini, and A. D. Bimbo, "Spatio-temporal closed-loop object detection," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1253–1263, 2017.

[17] S. Tripathi, S. Belongie, Y. Hwang, and T. Nguyen, "Detecting temporally consistent objects in videos through object class label propagation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, New York, the US, Mar. 2016, pp. 1–9.

[18] K. Kang H. Li, T. Xiao, W. Ouyang, J. Yan, X. Liu, and X. Wang, "Object detection in videos with tubelet proposal networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, Hawaii, USA, Jul, 2017, pp. 727–735.

[19] C. Feichtenhofer, A. Pinz, A. Zisserman, "Detect to track and track to detect," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognition*, Venice, Italy, Oct. 2017, pp. 3038–3046.

[20] G. Ning, Z. Zhang, C. Huang, X. Ren, H. Wang, C. Cai, and Z. He, "Spatially supervised recurrent convolutional neural networks for visual object tracking," in *Proc. IEEE Int. Symp. Circuits and Syst.*, Baltimore, USA, May. 2017, pp. 1–4.

[21] Y. Lu, C. Lu, and C. K. Tang, "Online video object detection using association LSTM, in *Proc. IEEE Int. Conf. Comput. Vis.*, Venice, Italy, Oct. 2017, pp. 2344–2352.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv:1409.1556*, 2014.

[23] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and F. Li, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[24] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process.*, Phoneix, USA, Sept. 2016, pp. 3464–3468.