ACTION VISUAL OBJECT TRACKING USING DEEP LEARNING

1. Dr.I.Satyanarayana,Principal,Sri Indu Institute of Engineering&Technology(SIIET), Sheriguda, Ibrahimpatnam,Hydarabad
2. D. Rajeshwari,assistant professor,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad
3.Sakshi Jadhav,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad
4.Yadavalli Santosh Reddy,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad
5.Ravula Deekshith Patel,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad
6 .C Abhilash Yadav,Student,CSE,SIIET,Sheriguda,Ibrahimpatnam,Hydarabad

ABSTRACT:

In this paper, we propose an efficient visual tracker, which directly captures a bounding box containing the target object in a video by means of sequential actions learned using deep neural networks. The proposed deep neural network to control tracking actions is pretrained using various training video sequences and fine-tuned during actual tracking for online adaptation to a change of target and background. The pretraining is done by utilizing deep reinforcement learning (RL) as well as supervised learning. The use of RL enables even partially labeled data to be successfully utilized for semisupervised learning. Through the evaluation of the object tracking benchmark data set, the proposed tracker is validated to achieve a competitive performance at three times the speed of existing deep network- based trackers. The fast version of the proposed method, which operates in real time on graphics processing unit, outperforms the state-of-the-art real-time trackers with an accuracy improvement of more than 8%

INTRODUCTION

The aim of visual object tracking is to find a bounding box tightly containing the target moving object in every frame of a video, which is one of thefundamental problems in the computer vision field. In recent decades, there have been many advances in visual tracking algorithms, but there are still many challenging issues arising from diverse tracking obstacles, such as motion blur, occlusion, illumination change, and background clutter. In particular, conventional tracking methods using low-level hand-crafted features encounter the above-mentioned
tracking obstacles because of their insufficient feature representation. YOLO method comes under this category. In this, we won't select the interested regions from the image. Instead, we predict the classes and bounding boxes of the whole image at a single run of the algorithm and

detect multiple objects using a single neural network. YOLO algorithm is fast as compared to

other classification algorithms. In real time our algorithm process 45 frames per second. YOLO algorithm makes localization errors but predicts less false positives in the background. Recently, tracking methods using convolutional neural networks (CNNs) have been proposed for robust tracking and vastly improved tracking performance with the help of rich feature representation by deep hidden layers.The initial works utilize pretrained CNNs, which are

trained on a large-scale classification data set, such as ImageNet. However, a CNN pretrained on a classification data set is not sufficient to solve the problem of adaptation to object shape

deformation and illumination changes in tracking, because the deep CNN is not appropriate for online adaptation. An existing state-of-the-art algorithm by Nam and Han adopts a tracking-bydetection approach using CNNs trained with plenty of video data sets. In addition, it improves the ability to distinguish between the target and the background using subnetworks that learn the discriminative features of the target and the background via online adaptation. This CNNbased tracking by-detection approach achieves a breakthrough in tracking performance, but it suffers from an inefficient exhaustive search strategy that explores the region of interest and selects the best candidate by referring to the scores obtained by the network. To overcome this exhaustive search problem with the tracking-by-detection methods, we introduce an actiondriven tracking mechanism that actively pursues the target movementconsidering the context change of the image within the bounding box.In addition, there is a critical problem with constructing training data for a deep CNN-based tracker. Deep CNN-based trackers require a large amount of training data in order to learn convolutional features from scratch. Even though there are plenty of video sequences, it is extremely expensive to annotate the target position in every frame for the construction of training data. If we can train a deep CNN-based tracker using a partially labelled video sequence, a variety of videos can be utilized with less effort for the training. However, the existing deep CNN-based trackers adopt a supervised learning (SL) scheme and so have difficulties with utilizing the partially labelled video sequences. In this paper, in contrast to the existing deep CNN-based trackers using SL, we try to develop a reinforcement learning (RL) scheme to utilize the partially labelled video sequences effectively for training our action-driven deep tracker. The proposed action-driven deep tracker pursues a change of target by repetitive

actions controlled by an action-decision network (ADNet) consisting of deep CNN architecture. We cast the visual tracking problem as selecting the sequential actions and suggest a training method using RL. The basic concept of the proposed visual tracking is

shown in Fig. 1. The ADNet is designed to generate actions to find the location and the size of the target object in a new frame. The ADNet learns the policy that selects the optimal actions to track the target from the state of its current position. In the ADNet, the policy network is designed with a CNN, in which the input is an image patch cropped at the position of the previous state and the output is the probability distribution of actions, including translation and scale changes. This actionselecting process has fewer searching steps than sliding window or candidate sampling approaches. In addition, since our method can precisely localize the target by selecting actions,

postprocessing, such as bounding box regression, is not necessary. We also propose a learning algorithm with deep RL to train the ADNet. The whole training framework is composed of an

SL stage and an RL stage. In the SL stage, we train our network to select actions to track the position of the target using samples extracted from training videos. In this step, the network learns to track general objects without sequential information. In the RL stage, the trained network in the SL stage is used as an initial network. We obtain training samples for RL by performing tracking simulation on training sequences. The network is trained with deep RL based on a policy gradient, using the rewards obtained during the tracking simulation. Even in the case where training frames are partially labelled [semisupervised (SS) case], the proposed framework successfully learns the unlabelled frames by assigning the rewards according to the results of tracking simulation. A preliminary version of this paper was presented as a spotlight in CVPR2017 , which is attached to a supplementary material for this paper. The prior work initially suggests the tracking mechanism by selecting sequential actions and the network architecture. In this paper, we supplement the related works by adding action-driven methods adopted in computer vision. In the text, additional derivations and algorithms have been added to explain the training scheme of RL and online adaptation. While the preliminary version evaluated the proposed tracker in general situations, the extended work includes evaluations with the various attributes of tracking scenes. In addition, the additional experiments and the analysis have been added to rigorously validate the proposed actiondriven approach. We

investigate the effect of the movement size of the action on tracking performance and speed. We analyze the action dynamic factors in the fc6 layer to examine the impact of the past actions.

For self-evaluation, two additional variants of the ADNet have been conducted and discussed:

1) the policy gradient method is replaced with a value function-based method and 2) the reward function is replaced with a continuous one. The main contributions of this paper are summarized as follows. The action-driven deep tracker is

proposed for the first time to dynamically track the target object by pursuing actions instead of tracking-by-detection scheme. We cast the visual

tracking problem as a Markov decision process (MDP) and design a deep network architecture to realize the decision process.

3) Deep RL algorithm is developed to train the ADNet with partially labeled data in the SS setting.

4) 4) The proposed deep tracker can control the trade-off between tracking performance and computational complexity by simply changing the meta parameter in tracking

5) The proposed tracker achieves a state-of-the-art performance with much more efficient searching complexity than the existing deep network-based trackers using a tracking-by-detection strategy. Also, the fast version of the proposed method operates in real time on graphics processing units (GPUs), outperforming the state-of-the-art real-time trackers.

LITERATUTE SURVEY

You Only Look Once: Unified, Real-Time Object Detection, by Joseph Redmon. Their prior work is on detecting objects using a regression algorithm. To get high accuracy and good predictions they have proposed YOLO algorithm in this paper [1]. Understanding of Object Detection Based on CNN Family andYOLO, by Juan Du. In this paper, they generally

explained about the object detection families like CNN, R-CNN and compared their efficiency and introduced YOLO algorithm to increase the efficiency [2]. Learning to Localize Objects with Structured Output Regression, by Matthew B. Blaschko. This paper is about Object

Localization. In this, they used the Bounding box method for localization of the objects to overcome the drawbacks of the sliding window method . Adopting Tile convolution neural network and recursive mode of same network helps in finding objects aiding applications for Driver assistance systems (DAS). Approach includes unsupervised training to help learn and modulate weights based on wide range of training data. Obstacle validation algorithms are included to reduce the count of valid detections [1]. Concepts like Optical flow and Histogram of magnitudes is used to analyze motion of objects, which are not evident to bare eyes. Detection of normal and abnormal events is achieved by classification

and localization helping campus environment to differentiate between normal and abnormal events [2]. Features are extracted using pretrained network; classified results are differentiated using SVM. Approach helps in guiding the route for ITS [3]. Many approaches like feature

extraction based on color and gradients fail to give spatial positioning in the image. The

challenges are overcome by employing Analysis of principal components by PCANet [4]

pipeline of image undistortion, image registration, classification and detections based on coordinates and velocities. Approach uses detectors like FAST, FREAK descriptors and followed by classification of Squeeze Net [5]. The workflow of candidate target generation, extracting features from candidate targets, the ground truth boxes around objects assist in tracking. The objects are classified using VGGNet [6]. CNN was designed to classify images, was repurposed to perform the object detection. The approach treats object detection as a relapse for object class to bounding objects detected. Series of gradual improvements has been witnessed from RCNN, Fast RCNN and faster RCNN then finally to YOLO. Instead of assessing image repetitively as in CNN, image is scanned once for all, thereby increasing the processing of frames per second (fps). YOLO is trained based on loss occurred unlike the

traditional Classification approach [7]. Paper describes about video analytics part for road traffic. One of main application area apart from vehicle detection and tracking is vehicle

counting. One of the novel algorithm called Single Shot Detector (SSD) is employed. Algorithm handles features like Binary large objects. It gives better results in applications like classification of objects. Object tracking employs concepts like background subtraction and virtual coil method. In terms of precision SSD outperforms YOLO versions. Swiftness and precision are always tradeoffs while selecting the right algorithm for object detection with the speed of 58fps performance metric for accuracy exceeds 85% [8], paper explains about upgradation to YOLO was made in the paper. Gradual updating has been witnessed throughout series of YOLO versions namely YOLOv1, YOLOv2, YOLOv3. YOLOv3 is state of the art technology. Upgradation such as thinner bounding boxes without affecting adjacent pixels. YOLOv3's implementation on COCO dataset shows mAP as good as SSD. YOLOv3 gives

three times faster results. YOLOv3 promises in detecting smaller objects [9]. With increase in vehicle density in urban region, Single object tracking will no longer cater for the need. Multi object tracking is achieved by employing kernelized correlation filter (KCF). Many KCF are run in parallel. KCF is best suited when images have occlusions. KCF when combined with background subtraction yield reliable results on the urban traffic [10] [12] [14]. Deep Networks require more computer power and time, more data, better performance of Neural Nets. The success of any algorithm lies in parameter tuning. Algorithms are application specific. Finetuning of state of the art Neural Nets decreases training time while increasing accuracy. Results are dependent on dataset used, algorithm and network employed. There have been tremendous improvements in deep learning and reinforcement learning

techniques. Automating learning and intelligence to the full extent remains a challenge. The amalgamation of Reinforcement Learning and Deep Learning has brought breakthroughs in

games and robotics in the past decade. Deep Reinforcement Learning (DRL) involves training the agent with raw input and learning via interaction with the environment. Motivated by recent

successes of DRL, we have explored its adaptability to different domains and application areas. This paper also presents a comprehensive survey of the work done in recent years and simulation tools used for DRL. The current focus of researchers is on recording the experience in a better way, and refining the policy for futuristic moves. It is found that even after obtaining good results in Atari, Go, Robotics, multi-agent scenarios, there are challenges such as generalization, satisfying multiple objectives, divergence, learning robust policy. Furthermore,

the complex environment and multiple agents are throwing new challenges, which is an open area of research.

This is due to the selected reward in the study, where it utilised the oriented angle of the road with the car speed. More generally, Yun et al. [10,11] proposed an action-decision network

(ADNet) to track objects (not necessarily for road tracking purposes), where a pre-trained Convolutional Neural Network (CNN) was firstly employed followed by the reinforcement

learning. Neural networks Accuracy Karaduman and Eren [16] CNN 67.42% Bojarski et al. [17] CNN 74.24% George and Routray [18] CNN 83.33% Yun et al. [10,11] ADNet 83.33% Mnih et al.

[19] DQN 88.64% This work DRL-RT 93.94% Table 5: A comparison between the DRL-RT method and other suggested networks errors of obtaining precise outputs. This is also due to the architecture of this network, which was designed for classifying eye gaze

directions. The ADNet used in [10,11] attained the same accuracy of 83.33%. The essential problem lies in the rewards used there, which were basically designed for recognising moved objects, as the rewards are updated in the stop action. Yun et al. [32] use an offline convolutional deep neural network (ADNet) for object detection and an action-driven method for temporal tracking. In a method presented in [33], authors proposed a learning-based tracking method which uses deep appearance to learn a discriminative appearance model from large training datasets for a reliable association between tracklets and detections. To the best knowledge of the authors, the five mentioned detectors have not previously been compared in online and offline trackers. The online tracker is far different from two-step

trackers which first detect objects in images and then associate the detected objects using another

classifier. In contrast with [34,35], the online tracker does not have any offline phase.

Neural networks are a group of algorithms designed to recognize patterns, modeled loosely after the human brain. They view sensory data by means of a form of raw input device perception, marking or clustering. Detect faces, identify people in pictures, recognize facial expressions. Identify objects in pictures or videos. The identification of similarities is clustering and grouping. Deep learning may create associations between, say, pixels in a picture and a person's name with classification. Neural Networks is currently one of the most common

algorithms for machine learning. The fact that neural networks outperform other algorithms in accuracy and speed has been clearly proved over time. With different variants such as CNN (Convolutional neural network), RNN (Recurrent Neural Networks), Deep Learning, etc. Deeplearning networks are differentiated by their depth from the more common singlehidden-layer

neural networks; that is, the number of node layers that information can move through in a pattern recognition multi-stage system. Most researchers use techniques of deep learning to extract qualified deep characteristics. In many challenging tasks, which historically rely on hand-crafted features such as location, monitoring, identification, human crowd detection, selfstabilization, obstacle and crash avoidance, perception of forest or mountain trails, and object

tracking, they have exhibited magnificent results. With the rise of autonomous vehicles, smart video surveillance, facial recognition and numerous people-counting applications, the demand for quick and accurate object detection systems is increasing. Such systems require not only the

identification and classification of each object in an image, but also the location of each object by drawing around the correct bounding box. This makes identification of objects a much harder

task than their conventional counterpart in computer vision, the recognition of images. Object Detection is modeled as a classification problem where at all possible locations we take gaps of fixed sizes from the input object to feed these patches into an image classifier. Every window is fed to the classifier which determines the object's class in the window. Therefore, we know the category and location of the image objects. CNNs consist of neurons with learning weights and biases such as neural networks. Each neuron receives multipleinputs, takes over a weighted sum, passes it through an activation function, and provides an output response. These are often used to recognize patterns like edges (vertical / horizontal), shapes, colours, and textures in object detection. Example of a CNN architecture: [INPUT — CONV — RELU — POOL — FC]. There are quite a few algorithms for object detection which have been

developed over the years. Inspired by the ground-breaking image classification results obtained by CNN and the success of selective search in regional proposal for hand-crafted apps, Girshick et al. were among the first to explore CNN for generic object detection and developed Region-based

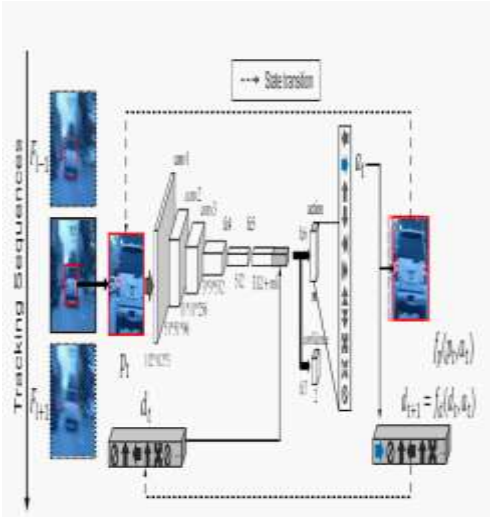
Convolutional Neural Networks(R-CNN) , which combines AlexNet with regional proposal system selective search. Since R-CNN's proposal, many improved models have been proposed,

including Fast R-CNN, which jointly optimizes classification and bounding box regression tasks, Faster R-CNN, which allows an additional subnetwork to produce local proposals, and YOLO, which performs object detection by means of a fixed grid regression. All bring different degrees of improvements in detection efficiency over the primary R-CNN and make object recognition more feasible in real-time and accuracy . YOLO is one of the fastest algorithms out there to detect objects. Although it is no longer the most accurate algorithm for object detection, when you need real-time detection without losing too much precision, it is a very good choice. YOLO uses a single convolutional network to predict several bounding boxes and category probabilities for these boxes at the same time . For computer vision, object tracking is an

important field. Object trackers can be categorized into TBD (Tracking by Detection) and DFT (Detection-Free Tracking) and online and offline trackers can also be separated by whether potential frames are used. This includes the method of tracking an object through a series of frames that could be a human, a ball or a vehicle. This begins with identifying all possible detections in a frame in object tracking and assigning them an ID. For the following images,

the current object ID is attempted to be carried forward. If the object moves away from the image, the ID will be removed. If a new object appears, a fresh ID will begin. This is a challenging task because objects may look similar, forcing the template to change IDs, an object may become occluded as when an item or entity is concealed behind something, or some objects may disappear and reappear later

Architecture:



EXISTING SYSTEM

Many problems in computer vision were saturating on their accuracy before a decade.

However, with the rise of deep learning techniques, the accuracy of these problems drastically improved.

One of the major problem was that of image classification, which is defined as

predicting the class of the image.

There is no surveillance surveillance object detection in existing system by using Opencv.

PROPOSED SYSTEM

Dense Optical flow: These algorithms help estimate the motion vector of every pixel in a video frame. Sparse optical flow: These algorithms, like the Kanade-Lucas-Tomashi (KLT) feature tracker, track the location of a few feature points in an image. Kalman Filtering: A very popular signal processing algorithm used to predict the location of a moving surveillance object based on prior motion information.

One of the early applications of this algorithm was missile guidance! Also as mentioned here, “the on-board computer that guided.

 ADVANTAGES OF PROPOSED SYSTEM

Here we can detect the surveillance object for uploaded video file.

METHODOLOGY

6.1 Algorithm

Convolutional Neural Networks (CNN)

CNN is widely used neural network architecture for computer vision related tasks. Advantage of CNN is that it automatically performs feature extraction on images i.e. important features are detected by the network itself.

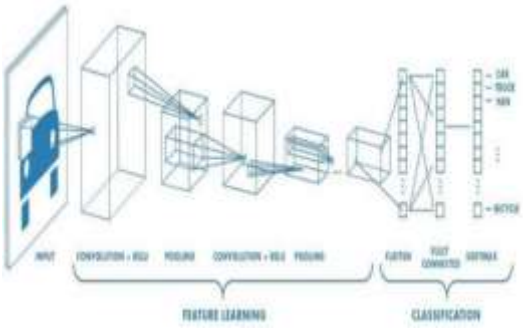


Fig. Overview of CNN Architecture.

CNN is made up of three important components called Convolutional Layer, Pooling layer, fully connected Layer as shown in Fig. 3. Considering a gray scale image of size 32*32 would have 1024 nodes in multi-layer approach. This process of flattening pixels loses

spatial positions of the image. Spatial relationship between picture elements is retained by learning internal feature representation using small squares of input data. Convolutional layer: Convolutional Layer encompasses filters and feature maps. Filters are processors of a particular layer. These filters are distinct from one another. They take pixel value as input and gives out feature Map. Feature map is output of one filter layer. Filter is traversed all along the image, moving one pixel at a time. Activation of few neurons takes place resulting in a feature map. Pooling layer: Pooling layer is employed to reduce

dimensionality. Pooling layers are

included after one or two convolutional layer to generalize features learnt from previous feature

maps. This helps in reducing chances of over fitting from training process. Fully connected layer: Fully connected layer is used at the end to assign the feature to class probability after extracting and consolidating features from Convolutional Layer and pooling later respectively. These layers use linear activation functions or softmax activation function. A CNN is designed and is trained on Urban Vehicle dataset, which is an Indigenous dataset for traffic surveillance applications. Specifications of Urban Vehicle Dataset or on road vehicle dataset [11] is tabulated in Table I. And number of images considered for each class is tabulated in Table II.

In total there are 64339 images belonging to 4 classes that are taken under different times of the day and capturing conditions, including NIR images. The images are classified based on utility and size of vehicles. The auto class has images of three wheelers; Heavy class includes buses, trucks, Freight carriers; Light class has cars, SUVs and sedans and two wheelers include motorcycles and bicycles. Most of images have only one object

belonging to its respective class.

Some of the sample images of dataset is shown in Figure below

FLOW CHART:

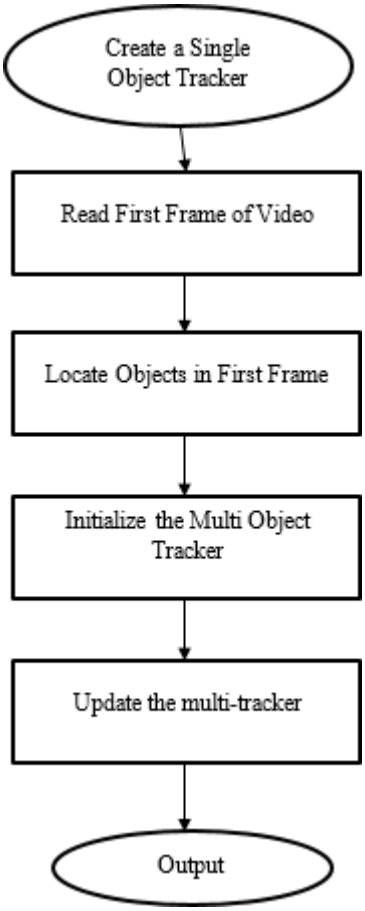


Fig.Flow Chart of Multiple Object Tracking

RESULT:

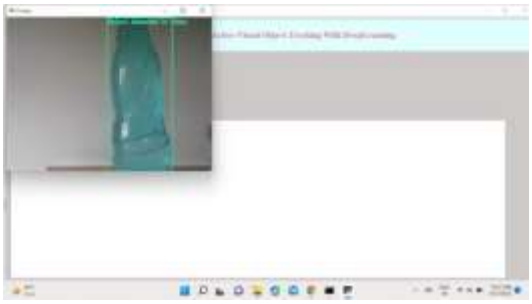
Double click on 'run.bat' file to get below screen

Now click on 'Browse System Videos' button to upload videos from system.

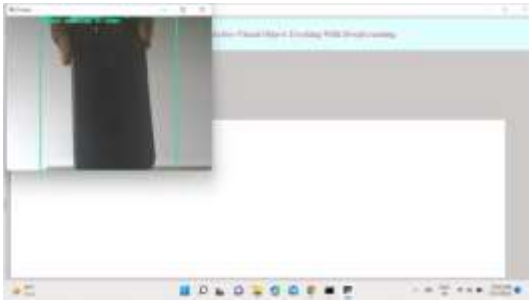
In above screen I am uploading one video, after upload will get below screen.

In above video we can see application start tracking objects from video and mark them with bounding boxes. Similarly we can upload any video and track objects from video

In above screen now click on another button called 'Start Webcam Video Tracking' to connect application to web cam and start streaming. After connecting to webcam will get below screen



In above screen we can see objects is getting tracked from webcam also. In above screen it track computer mouse from web cam video



CONCLUSION

In this paper, we have proposed a novel action-driven method using deep convolutional networks for visual tracking. The proposed tracker is controlled by an ADNet, which pursues the target object by sequential actions iteratively. The action-driven tracking strategy makes a significant contribution to the reduction of computation complexity in tracking. In addition, RL makes it possible to use partially labeled data, which could greatly contribute to the building of training data with a little effort. According to the evaluation results, the proposed tracker achieves the state-of-the-art performance in 3 frames/s, which is three times faster than the existing deep network-based trackers using a tracking-by-detection strategy. Furthermore, the fast version of the proposed tracker achieves a real-time speed (15 frames/s) by adjusting the meta parameters of the ADNet, with an accuracy that outperforms state- of-the-art realtime trackers. An accurate and efficient surveillance surveillance object detection system has been developed which achieves comparable metrics with the existing state-of-the-art system. This project uses recent techniques in the field of computer vision and deep learning.

.

REFERENCES

[1] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proc. Brit. Mach. Vis. Conf.*, 2006, vol. 1. no. 5, p. 6.

[2] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 234–247.

[3] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1619–1632, Aug. 2011.

[4] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2012.

[5] D. S. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2544–2550.

[1] M. Danelljan, G. Häger, F. Khan, and M. Felsberg, "Accurate scale estimation for robust

[2] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2015.

[3] 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," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 749–758.

[4] J. Choi, H. Jin Chang, J. Jeong, Y. Demiris, and J. Y. Choi, "Visual tracking using attention-modulated disintegration and integra- tion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 4321–4330.

[5] N. Wang, S. Li, A. Gupta, and D.-Y. Yeung. (2015). "Transferring rich feature hierarchies for robust visual tracking." [Online]. Available: https://arxiv.org/abs/1501.04587

[6] S. Hong, T. You, S. Kwak, and B. Han. (2015). "Online tracking by learning discriminative saliency map with convolutional neural net- work." [Online]. Available: https://arxiv.org/abs/1502.06796

[7] H. Nam and B. Han. (2015). "Learning multi-domain convolu- tional neural networks for visual tracking." [Online]. Available: https://arxiv.org/abs/1510.07945

[8] O. Russakovsky *et al.*, "ImageNet large scale visual recognition chal- lenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.

[9] Y. Wu, J. Lim, and M. H. Yang, "Object tracking benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1834–1848, Sep. 2015.

[10] M. Kristan, *et al.*, "The visual object tracking VOT2014 challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshops*, Berlin, Germany, Mar. 2014, pp. 191–217.

[11] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. (2014). "Return of the devil in the details: Delving deep into convolutional nets." [Online]. Available: https://arxiv.org/abs/1405.3531

[12] R. Tao, E. Gavves, and A. W. Smeulders. (2016). "Siamese instance search for tracking." [Online]. Available: https://arxiv. org/abs/1605.05863

[13] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, 1992.

[14] S. Yun, J. Choi, Y. Yoo, K. Yun, and J. Young Choi, "Action-decision networks for visual tracking with deep reinforcement learning," in *Proc. IEEE Conf. Comput. Vis. Pattern*

Recognit., Jun. 2017, pp. 1349–1358.

[15] H. Yang, L. Shao, F. Zheng, L. Wang, and Z. Song, "Recent advances and trends in visual tracking: A review," *Neurocomputing*, vol. 74, no. 18, pp. 3823–3831, Nov. 2011.

[16] A. W. M. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara,

A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1442–1468, Jul. 2014.

[17] M. Kristan *et al.*, "The visual object tracking vot2015 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2015, pp. 1–23.

[18] J. Choi, H. J. Chang, S. Yun, T. Fischer, Y. Demiris, and J. Y. Choi, "Attentional correlation filter network for adaptive visual tracking," in *Proc.*

IEEE Conf. Comput. Vis. Pattern Recognit., Jul. 2017, pp. 4828–4837.

[19] M. Danelljan, G. Häger, F. S. Khan, and M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2015, pp. 58–66.

[20] M. Danelljan, A. Robinson, F. Shahbaz Khan, and M. Felsberg, "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. ECCV*, 2016, pp. 472–488.

[21] N. Wang and D.-Y. Yeung, "Learning a deep compact image representa- tion for visual tracking," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 809–817.

[22] H. Li, Y. Li, and F. Porikli, "Robust online visual tracking with a single convolutional neural network," in *Proc. Asian Conf. Comput. Vis.*, 2014, pp. 194–209.

[23] H. Li, Y. Li, and F. Porikli, "DeepTrack: Learning discriminative feature representations online for robust visual tracking," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1834–1848, Apr. 2016.

[24] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 3119–3127.