Implementation of Fully-Convolutional Siamese Networks for Object Tracking

Kapil wanaskar
San José State University
Computer Engineering Department
San Jose, CA 95112
Email: kapil.wanaskar@sjsu.edu

***Abstract*—In this study, we address the challenge of tracking objects in video footage, a task traditionally dependent on models learned online from video data alone [1]. We propose a solution that involves a novel, fully-convolutional Siamese network, trained end-to-end on the extensive ILSVRC15 dataset, to enhance object detection capabilities in video [1]. This method surpasses the limitations of previous techniques that required online Stochastic Gradient Descent for network weight adjustments, which hindered system speed [1]. Our approach, utilizing the Siamese network, demonstrates exceptional performance in multiple tracking benchmarks, operating at frame rates that exceed real-time expectations, despite its straightforward design [1].**

***Index Terms*—object tracking, Siamese network, similarity learning, deep learning.**

I. Introduction

The objective of this paper is to advance the field of video-based object tracking, where the object is initially identified by a single rectangle in the first frame [1]. Historically, this has involved learning the object's appearance online using the video as the primary data source, a method utilized by TLD [2], Struck [3], and KCF [4]. However, this approach is limited due to the simplicity of the models it can generate [1]. Deep convolutional networks offer a promising solution but face challenges due to the lack of supervised data and the need for real-time processing [1].

Our innovative approach involves training a deep convolutional network in an offline phase for a broader similarity learning task, followed by its application in real-time tracking [1]. This strategy, leveraging a Siamese network architecture, allows for efficient processing and is fully-convolutional relative to the search image, featuring a bilinear layer for effective cross-correlation [1]. The research demonstrates the model's ability to generalize across different video domains, moving from the ImageNet Video domain to the ALOV/OTB/VOT domains[1,11,12], a significant step given the restrictions imposed by the VOT committee on training and testing in the same domain [1]. This method marks a departure from traditional online learning models, illustrating the potential of deep learning in object tracking under the constraints of limited large annotated datasets like the ILSVRC dataset[1,10].

II. Related Work

In the realm of object tracking, the potential of Recurrent Neural Networks (RNNs) has been explored in recent studies. Gan et al. utilized RNNs to predict the target's absolute position in each frame [25], while Kahou et al. adopted a differentiable attention mechanism within RNNs for tracking [26]. Despite their innovative approaches, these methods have not yet achieved competitive results in modern benchmarks, indicating a fertile area for future exploration. An interesting parallel exists between these methods and our Siamese network approach, as the latter can be viewed as an unrolled RNN trained on short sequences[25,26].

Denil et al. explored object tracking through a particle filter that employs a learned distance metric for comparing appearances [27]. Their approach differs significantly from ours, focusing on fixations within the object's bounding box, utilizing Restricted Boltzmann Machines (RBMs) to learn the distance metric [27]. This method, while innovative, has been primarily demonstrated in controlled settings such as MNIST digit sequences and specific tracking scenarios like face and person tracking [27].

In terms of leveraging deep convolutional networks (conv-nets), several studies have investigated the feasibility of fine-tuning pre-trained network parameters for specific videos. SO-DLT [7] and MDNet [9] have shown promise in this regard, but their real-time operation is hindered by the computational load of forward and backward passes on numerous examples. Another approach is the application of traditional shallow methods using features from pre-trained conv-nets, as seen in DeepSRDCF [6], Ma et al. [5], and FCNT [8]. While these methods have achieved strong results, they struggle with frame-rate operation due to the high dimensionality of conv-net representations[5,6,8].

Concurrently with our research, other authors have also utilized conv-nets for object tracking by learning functions from image pairs. Held et al. introduced GOTURN [28], training a conv-net to locate objects in subsequent images. Chen et al. developed a network (YCNN) mapping an exemplar and search region to a response map [29]. Both approaches, however, lack intrinsic invariance to the translation of the second image, necessitating extensive dataset augmentation[28,29]. Tao et al. proposed SINT, a Siamese network for identifying candidate image locations, but their architecture was not fully-convolutional concerning the search image [30]. Despite optimizations like RoI pooling, their system's speed was still not real-time [30].

A critical aspect of these studies is the training data domain. Competitive methods like MDNet [9], SINT [30], and GOTURN [28] have used training data from the ALOV/OTB/VOT domain, which is now prohibited in the VOT challenge to prevent overfitting[28,30,9]. Our contribution is significant in demonstrating that a conv-net can be effectively trained for object tracking without relying on the same distribution of videos as the testing set, thus addressing concerns of overfitting to benchmark scenes and objects.

III. Project Goals and Methodology

The objective of this project is to refine object tracking by leveraging similarity learning, focusing on a unique function f(z, x). This function will assess the resemblance between an exemplar image z and a candidate image x, providing high scores for identical objects and low scores otherwise. Our strategy involves examining all possible locations in a new image to locate the object, using its initial appearance as a benchmark[13,14,15,16]. The function f, instrumental in this process, will be derived from a set of videos with labeled object trajectories.

Deep convolutional networks (conv-nets), acclaimed for their success in computer vision, will serve as the backbone for function f. We aim to utilize Siamese architectures for similarity learning within these conv-nets[17,18,19]. In these networks, both inputs undergo a consistent transformation , followed by the integration of their outputs using a function g, conceptualized as f(z, x) = g((z), (x)). This structure allows to act as an embedding when g is a basic distance or similarity metric. Prior applications of deep Siamese conv-nets include diverse tasks like face verification, keypoint descriptor learning, and one-shot character recognition[18,20,14,19,21,22].

A. Convolutional Approaches and Image Analysis

Our approach during tracking employs a centered search image at the target's previous position. The network's efficiency is amplified by processing multiple image scales in a single pass and employing cross-correlation for feature map combination. This method not only streamlines testing but also enhances the training process.

B. Training Techniques and Network Optimization

For training, we adopt a discriminative model, focusing on positive and negative image pairs and utilizing logistic loss. Training pairs are curated from annotated videos, with images centered on the target. The symmetric nature of our network, indicated by f(z, x) = f(x, z), allows for flexibility in handling exemplar images, although we currently standardize their sizes for batch processing efficiency.

Fig. 1. Training pairs extracted from the same video: exemplar and search image.

C. Leveraging ImageNet Video Dataset

The expansive ImageNet Video dataset, part of the 2015 ILSVRC, provides a rich resource for our training needs [10]. Its vastness and variety offer a significant advantage over traditional datasets like VOT [12], ALOV [1], and OTB [11]. Training with ImageNet Video also mitigates the risk of overfitting to specific video domains prevalent in standard benchmarks.

D. Practical Implementation and Dataset Utilization

1) Dataset Selection and Preparation: Training involves using standardized exemplar and search images, scaled for consistency. The entire ImageNet Video dataset, encompassing over two million labeled bounding boxes, forms the basis of our training data [10].

2) Architectural Choices: The network architecture, inspired by Krizhevsky et al. [16], omits padding to preserve the fully-convolutional nature. The layout includes max pooling and ReLU non-linearities, complemented by batch normalization for each layer [24].

3) Simplified Tracking Algorithm: The project employs a straightforward tracking algorithm, intentionally avoiding complex model updates or additional tracking cues. This simplicity, however, doesn't detract from its effectiveness, demonstrating the robustness of our offline-learned similarity metric. During live tracking, the algorithm imposes basic temporal constraints to enhance accuracy and reliability.

Layer	Support	Chan. map	Stride	Activation size for exemplar	Activation size for search	chans.
				127×127	255×255	$\times 3$
conv1	11×11	96×3	2	59×59	123×123	$\times 96$
pool1	3×3		2	29×29	61×61	$\times 96$
conv2	5×5	256×48	1	25×25	57×57	$\times 256$
pool2	3×3		2	12×12	28×28	$\times 256$
conv3	3×3	384×256	1	10×10	26×26	$\times 192$
conv4	3×3	384×192	1	8×8	24×24	$\times 192$
conv5	3×3	256×192	1	6×6	22×22	$\times 128$

TABLE I
ARCHITECTURE OF CONVOLUTIONAL EMBEDDING FUNCTION

IV. Experimentation and Implementation Strategy

Our project's experimentation and implementation plan is structured to comprehensively assess and refine our object tracking methodology. This plan is divided into four key sections: system architecture, advantages, disadvantages, and limitations.

A. System Architecture

Our system's architecture is centered around a Siamese network, trained to identify objects in videos. The training phase utilizes Stochastic Gradient Descent (SGD) with parameters initially set according to a Gaussian distribution and adjusted via the improved Xavier method[31,32]. Each training epoch comprises 50,000 sampled pairs, and mini-batches of size 8 are used for gradient estimation. For tracking, the online phase is designed to be minimalistic. The initial object's embedding is computed once and then convoluted across subsequent frames. Scale variations are accounted for by searching across five scales, and bicubic interpolation is used for enhancing the precision of the score map.

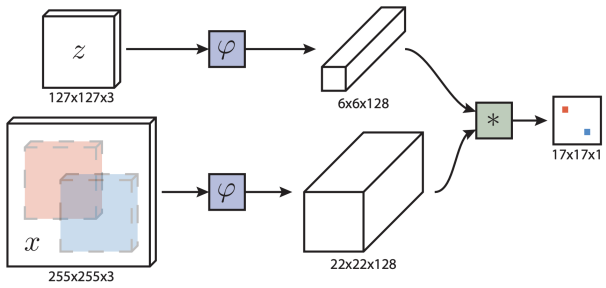

Fig. 2. Fully-convolutional Siamese architecture.

This architecture is fully convolutional relative to the search image x. It generates a scalar score map whose dimensions are contingent on the search image's size. This design allows the computation of similarity for all shifted sub-windows within the search image in a single operation. The score map features red and blue pixels, indicating the similarity levels of corresponding sub-windows. This map is best interpreted in color.

B. Advantages

The system showcases several advantages, particularly in its ability to track objects in real-time with high accuracy. Our simplistic approach, while streamlined, proves to be effective in various benchmarks. The success plots (Fig. 3, 4, 5) and accuracy-robustness plots (Fig. 6) demonstrate our method's superiority over other state-of-the-art real-time trackers, particularly in the VOT-15 benchmark as illustrated by the expected average overlap rankings (Fig. 7).

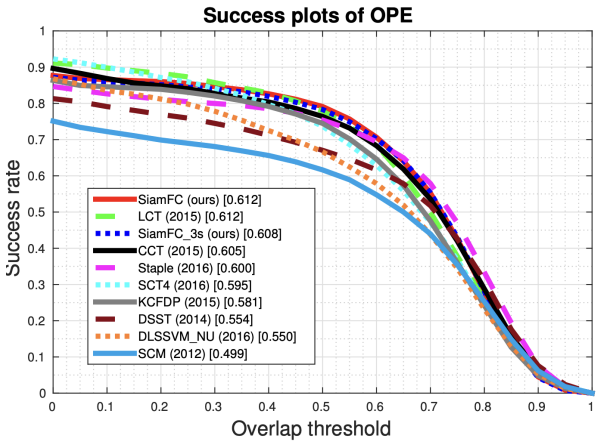

Fig. 3. Success plots for OPE (one pass evaluation).

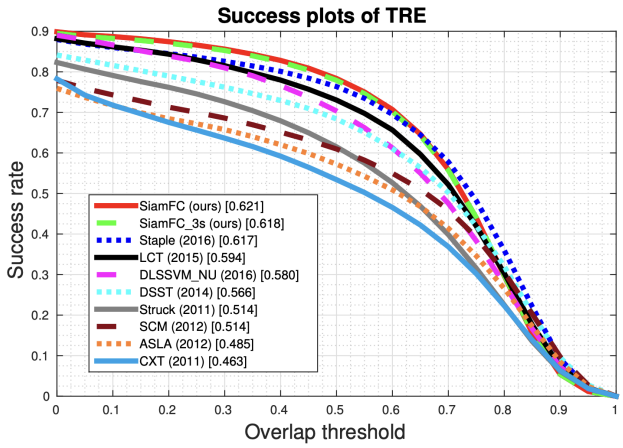

Fig. 4. Success plots for TRE (temporal robustness evaluation).

The success plots (Fig. 3) and accuracy-robustness plots (Fig. 4) demonstrate our method's superiority over other state-of-the-art real-time trackers, particularly in the VOT-15 benchmark as illustrated by the expected average overlap rankings (Fig. 5).

Accuracy-robustness plots (Fig. 6) demonstrate our method's superiority over other state-of-the-art real-time trackers, particularly in the VOT-15 benchmark as illustrated by the expected average overlap rankings (Fig. 7).

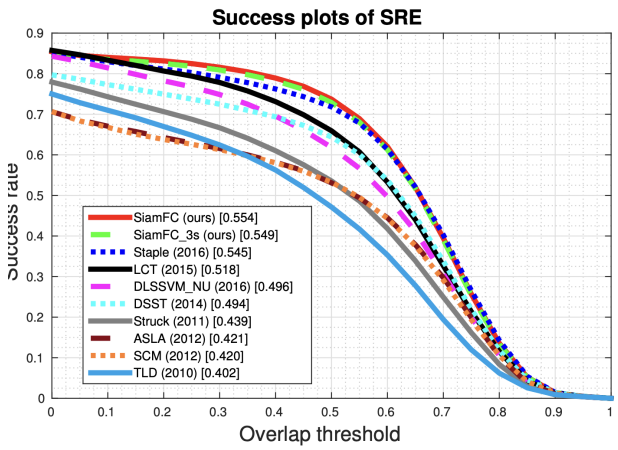

Fig. 5. Success plots for SRE (spatial robustness evaluation).

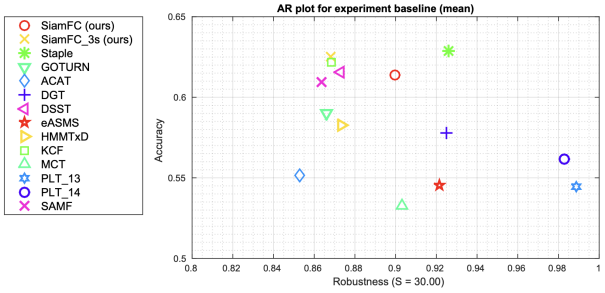

Fig. 6. Accuracy-robustness plots.

C. Disadvantages

Despite its strengths, our system has some drawbacks. The simplistic nature of the online tracking phase, although efficient, lacks advanced features such as model updates, bounding-box regression, and fine-tuning. This could potentially limit the system's adaptability in more complex tracking scenarios.

D. Limitations

One of the main limitations of our approach is its dependence on the size and diversity of the training dataset. The performance, as indicated by the VOT-15 results (Fig. 5 and Table 3), improves with the dataset's size. However, the current dataset, despite its large number of supervised bounding boxes, offers limited video variety, potentially constraining the system's ability to generalize across diverse tracking environments.

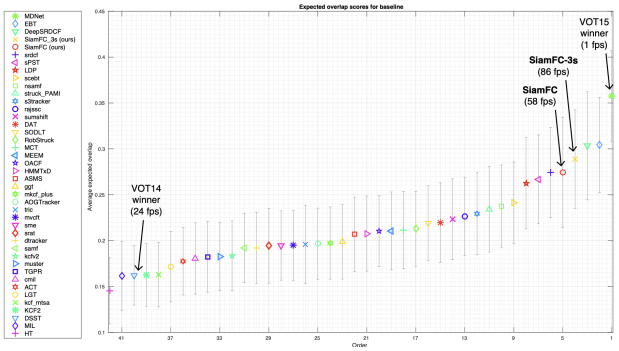

Fig. 7. Expected average overlap rankings.

V. Conclusion

Our conclusion synthesizes the outcomes of this study, highlighting the major findings, acknowledging the limitations, and proposing directions for future research.

Tracker	accuracy	# failures	overlap	speed (fps)
MDNet [9]	0.5620	46	0.3575	1
EBT [41]	0.4481	49	0.3042	5
DeepSRDCF [6]	0.5350	60	0.3033	< 1 *
SiamFC-3s (ours)	0.5335	84	0.2889	**86**
SiamFC (ours)	0.5240	87	0.2743	58
SRDCF [42]	0.5260	71	0.2743	5
sPST [43]	0.5230	85	0.2668	2
LDP [12]	0.4688	78	0.2625	4 *
SC-EBT [44]	0.5171	103	0.2412	–
NSAMF [45]	0.5027	87	0.2376	5 *
StruckMK [3]	0.4442	90	0.2341	2
S3Tracker [46]	0.5031	100	0.2292	14 *
RAJSSC [12]	0.5301	105	0.2262	2 *
SumShift [46]	0.4888	97	0.2233	17 *
DAT [47]	0.4705	113	0.2195	15
SO-DLT [7]	0.5233	108	0.2190	5

TABLE II
RAW SCORES, OVERLAP AND REPORTED SPEED

A. Major Findings

This research marks a significant deviation from the conventional online learning approach typically employed in object tracking. We have demonstrated the effectiveness of Siamese fully-convolutional deep networks in tracking applications, showcasing their efficiency in utilizing available data. This efficiency is evident not only in spatial searches during test-time but also in the training phase, where each sub-window serves as a valuable sample with minimal additional cost. Our experiments have confirmed that deep embeddings offer a rich source of features for online trackers, allowing even simple test-time strategies to yield strong performance. Fig. 6 presents snapshots of our simple tracker in action, illustrating the practical application and effectiveness of our approach.

Dataset (%)	# videos	# objects	accuracy	# failures	expected avg. overlap
2	88	60k	0.484	183	0.168
4	177	110k	0.501	160	0.192
8	353	190k	0.484	142	0.193
16	707	330k	0.522	132	0.219
32	1413	650k	0.521	117	0.234
100	4417	2m	**0.524**	**87**	**0.274**

TABLE III
TRACKER'S PERFORMANCE ON USING INCREASING PORTIONS.

B. Limitations of the Study

While our approach has shown promising results, it's important to acknowledge its limitations. The primary

limitation lies in the dependency on the size and diversity of the training dataset. A more varied and extensive dataset could potentially enhance the system's performance and generalizability. Additionally, the simplicity of our online tracking strategy, though efficient, might not be sufficient for more complex and dynamic tracking scenarios.

Fig. 8. Snapshots of the tracker.

C. Recommendations for Future Research

Future research should aim to explore the integration of our offline learning approach with more sophisticated online tracking methods. This could involve the development of advanced algorithms that combine the strength of deep embeddings with adaptive online learning techniques. Further exploration into expanding and diversifying the training dataset would also be beneficial, as it would likely improve the system's ability to generalize across a wider range of tracking scenarios. This research paves the way for more comprehensive and versatile object tracking solutions, and we anticipate significant advancements in this field in the near future.

REFERENCES

[1] A.W.M. Smeulders, D.M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, M. Shah, "Visual tracking: An experimental survey," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1442–1468, 2014.

[2] Z. Kalal, K. Mikolajczyk, J. Matas, "Tracking-learning-detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[3] S. Hare, A. Saffari, P.H.S. Torr, "Struck: Structured output tracking with kernels," in *International Conference on Computer Vision (ICCV)*, IEEE, 2011.

[4] J.F. Henriques, R. Caseiro, P. Martins, J. Batista, "High-speed tracking with kernelized correlation filters," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[5] C. Ma, J.B. Huang, X. Yang, M.H. Yang, "Hierarchical convolutional features for visual tracking," in *International Conference on Computer Vision (ICCV)*, 2015.

[6] M. Danelljan, G. Hager, F. Khan, M. Felsberg, "Convolutional features for correlation filter based visual tracking," in *ICCV 2015 Workshop*, 2015, pp. 58–66.

[7] N. Wang, S. Li, A. Gupta, D.Y. Yeung, "Transferring rich feature hierarchies for robust visual tracking," arXiv CoRR, 2015.

[8] L. Wang, W. Ouyang, X. Wang, H. Lu, "Visual tracking with fully convolutional networks," in *International Conference on Computer Vision (ICCV)*, 2015.

[9] H. Nam, B. Han, "Learning multi-domain convolutional neural networks for visual tracking," arXiv CoRR, 2015.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," in *International Journal of Computer Vision (IJCV)*, 2015.

[11] Y. Wu, J. Lim, M.H. Yang, "Online object tracking: A benchmark," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

[12] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, L. Cehovin, G. Fernandez, T. Vojir, G. Hager, G. Nebehay, R. Pflugfelder, "The Visual Object Tracking VOT2015 Challenge results," in *ICCV 2015 Workshop*, pp. 1–23, 2015.

[13] A. Razavian, H. Azizpour, J. Sullivan, S. Carlsson, "CNN features off-the-shelf: An astounding baseline for recognition," in *CVPR 2014 Workshop*.

[14] O.M. Parkhi, A. Vedaldi, A. Zisserman, "Deep face recognition," in *British Machine Vision Conference (BMVC)*, 2015.

[15] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, T. Brox, "FlowNet: Learning optical flow with convolutional networks," in *International Conference on Computer Vision (ICCV)*, 2015.

[16] A. Krizhevsky, I. Sutskever, G.E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, 2012.

[17] J. Bromley, J.W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, R. Shah, "Signature verification using a 'Siamese' time delay neural network," in *International Journal of Pattern Recognition and Artificial Intelligence*, 1993.

[18] Y. Taigman, M. Yang, M. Ranzato, L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1701–1708, 2014.

[19] S. Zagoruyko, N. Komodakis, "Learning to compare image patches via convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[20] F. Schroff, D. Kalenichenko, J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815–823, 2015.

[21] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *International Conference on Computer Vision (ICCV)*, pp. 118–126, 2015.

[22] G. Koch, R. Zemel, R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML 2015 Deep Learning Workshop*, 2015.

[23] W. Luo, A.G. Schwing, R. Urtasun, "Efficient deep learning for stereo matching," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5695–5703, 2016.

[24] S. Ioffe, C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International Conference on Machine Learning (ICML)*, pp. 448–456, 2015.

[25] Q. Gan, Q. Guo, Z. Zhang, K. Cho, "First step toward model-free, anonymous object tracking with recurrent neural networks," *arXiv CoRR*, 2015.

[26] S.E. Kahou, V. Michalski, R. Memisevic, "RATM: Recurrent Attentive Tracking Model," *arXiv CoRR*, 2015.

[27] M. Denil, L. Bazzani, H. Larochelle, N. de Freitas, "Learning where to attend with deep architectures for image tracking," *Neural Computation*, 2012.

[28] D. Held, S. Thrun, S. Savarese, "Learning to track at 100 fps with deep regression networks," *arXiv CoRR*, 2016.
[29] K. Chen, W. Tao, "Once for all: A two-flow convolutional neural network for visual tracking," *arXiv CoRR*, 2016.
[30] R. Tao, E. Gavves, A.W.M. Smeulders, "Siamese instance search for tracking," *arXiv CoRR*, 2016.