Proposing an Object Tracking Pipeline Inspired by the Visual System

Manuel Alvarez-Rios[1], Pulkit Grover [2] , Yorie Nakahira [2]

[1] Department of Computer Science, University of Puerto Rico Rio Piedras Campus

[2] Department of Electrical & Computer Engineering , Carnegie Mellon University

Objectives

- Compare object tracking by the human visual system against typical computer models
- Use insights from human visual system tracking to inspire novel algorithms

Motivation:

Multi Object Tracking (MOT) is a computer vision task that aims to analyze videos in order to identify and track objects belonging to one or more categories, such as pedestrians, cars, animals and inanimate objects, without any prior knowledge about the appearance and number of targets [11] (Fig.1)

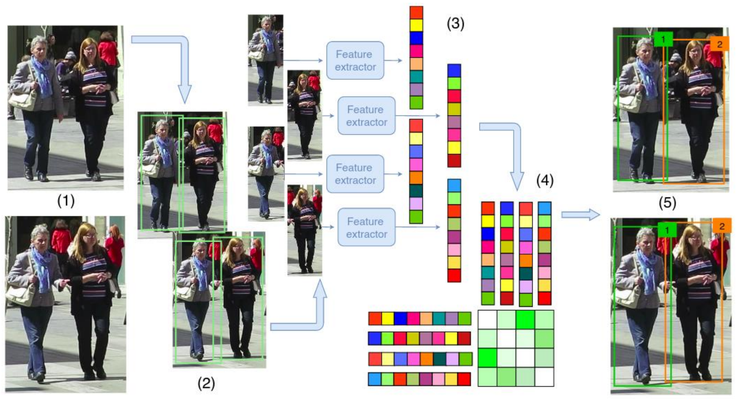


Figure 1. Usual workflow of a MOT algorithm:

1. Input frames
2. object detector is run to obtain the bounding boxes of the objects
3. For every detected object, different features are computed
4. An affinity computation step calculates the probability of two objects belonging to the same target
5. An association step assigns a numerical ID to each object

In step 2 an object detector is typically used. The essence of object detection is to locate and classify objects, which uses rectangular bounding boxes to locate the detected objects and classify the categories of the objects. (Fig 2) Careful use of deep learning in the detection step can considerably improve the performance of a tracking algorithm. [9,10]

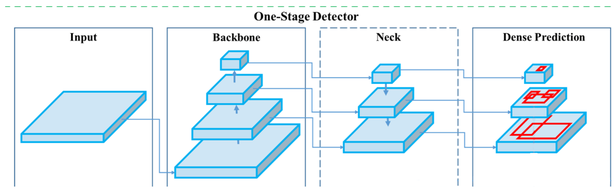


Figure 2. A simplification of the typical architecture of one stage deep learning object detector. Starting with an input image or video frame, this image gets passed to the backbone, a CNN for feature extraction. Additionally, sometimes they include the "neck", operations for enhancing the selection of features. After this, the feature vector is passed to the "head" or "Dense Predictor" that takes care of the detection and classification.

The backbone for object detection is typically a Feed-forward convolutional neural networks (CNNs). CNNs are currently state-of-the-art for most computer vision-related tasks. Further, they are quantitatively accurate models of temporally-averaged responses of neurons in the primate brain's visual system. [2,8]

CNNs cannot intuitively encode temporal or motion features due to their feed forward nature. That's why usually Kalman filter or LSTM are used for extracting motion features in tracking. We hypothesize that using a recurrences and feedback in in CNNs architectures could improve tracking by enabling feature reusability and extraction of motion features like an LSTM.

Related work

Biological visual systems have two architectural features not shared with typical Feed-Forward CNNs: local recurrence within cortical areas, and long-range feedback from downstream areas to upstream areas[1-8], it is hypothesized that these architectures play a role in object recognition and scene understanding [1,2]. Notably [8], presented the convolutional recurrent neural network (ConvRNN) (Fig. 3) implementing both architectures in CNNs and showing performance gains.

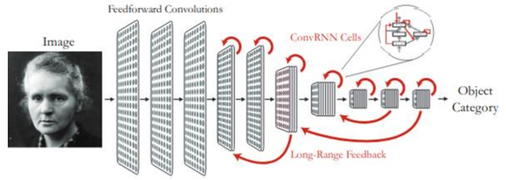


Figure 3. Usual workflow of a MOT algorithm:
The architecture of ConvRNN it has feedforward connections, long-range feedback connections, and self recurrence (ConvRNN Cells)

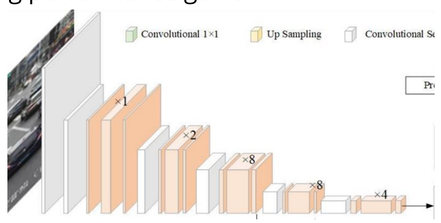
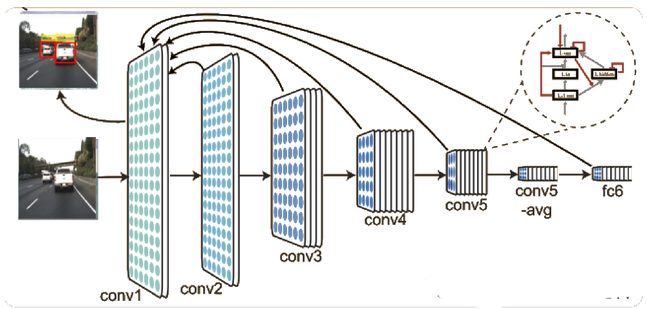


Figure 4. Yolo v3 Backbone : Darknet 53 has 53 convolutional layers . Some of the same transformation are perform upwards of 8 times .

- ConvRNNs have less parameters than a similar performing CNN: Recurrence “extends” a feedforward computation, reflecting the fact that an unrolled recurrent network is equivalent to a deeper feedforward network that conserves on neurons by repeating transformations several times e.g. Figure 4.
- ConvRNNs proved useful in scene understanding because they naturally integrate high- and low-level visual information, both of which are critical to understand scene structure. Long-range feedback in a ConvRNN plays a role analogous to the up-sampling layers typically used in the neck of object detectors , while locally recurrent cells are like “skip-connections” in that they combine features of the same spatial resolution.

Our Proposed Architectures:

- Detection Backbone: The first Architectures focuses on utilizing the ConvCNNs as the backbone for the object detector. Since it has been shown to perform better on certain task
- Unifying detection and tracking: Our feature extraction component is a ConvRNN based on a 5-layer CNN, each layer augmented with a task optimized recurrent unit. Additionally, long-range feedback connections are introduced from all layers to the first convolutional layer. This structure could take advantage of changing in features over time by extracting motion and visual features at multiple scales after differing numbers of passes.



References

1. Bear, D. M., Fan, C., Mrowca, D., Li, Y., Alter, S., Nayebi, A., ... Yamins, D. L. K. (2020). Learning Physical Graph Representations from Visual Scenes. Retrieved from http://arxiv.org/abs/2006.12373
2. Kar, K., Kubilius, J., Schmidt, K., Issa, E. B., & DiCarlo, J. J. (2019). Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. Nature Neuroscience, 22(6), 974–983. https://doi.org/10.1038/s41593-019-0392-5
3. Cao, C., Huang, Y., Yang, Y., Wang, L., Wang, Z., & Tan, T. (2019). Feedback Convolutional Neural Network for Visual Localization and Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(7), 1627–1640. https://doi.org/10.1109/TPAMI.2018.2843329
4. Wang, T., Yamaguchi, K., & Ordonez, V. (2018). Feedback-Prop: Convolutional Neural Network Inference under Partial Evidence. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 898–907. https://doi.org/10.1109/CVPR.2018.00100
5. Li, X., Jie, Z., Feng, J., Liu, C., & Yan, S. (2017). Learning with Rethinking: Recurrently Improving Convolutional Neural Networks through Feedback. Retrieved from http://arxiv.org/abs/1708.04483
6. Jarvers, C., & Neumann, H. (2019). Incorporating Feedback in Convolutional Neural Networks. 2019 Conference on Cognitive Computational Neuroscience. https://doi.org/10.32470/CCN.2019.1191-0
7. Fu, C., Wu, X., Dong, J., & He, R. (2018). Global Perception Feedback Convolutional Neural Networks. In Y. Wang, S. Wang, Y. Liu, J. Yang, X. Yuan, R. He, & H. B.-L. Duh (Eds.), IGTA 2017: Advances in Image and Graphics Technologies (pp. 65–73). https://doi.org/10.1007/978-981-10-7389-2_7
8. Nayebi, A., Bear, D., Kubilius, J., Kar, K., Ganguli, S., Sussillo, D., ... Yamins, D. L. K. (2018). Task-Driven Convolutional Recurrent Models of the Visual System. Advances in Neural Information Processing Systems 31, 5290–5301.
9. Bochkovskiy, A., Wang, C., & Liao, H. M. (2020). YOLOv4: Optimal Speed and Accuracy of Object Detection. Retrieved from http://arxiv.org/abs/2004.10934
10. Xiao, Y., Tian, Z., Yu, J., Zhang, Y., Liu, S., Du, S., & Lan, X. (2020). A review of object detection based on deep learning. Multimedia Tools and Applications. https://doi.org/10.1007/s11042-020-08976-6
11. Ciaparrone, G., Sánchez, F. L., Tabik, S., Troiano, L., Tagliaferri, R., & Herrera, F. (2019). Deep Learning in Video Multi-Object Tracking: A Survey. https://doi.org/10.1016/j.neucom.2019.11.023