Real Time Generic Object Detection and Tracking

Problem Statement

The objective is to detect and track the generic object in real time. In real life, therefore, we require rich information about the surrounding. We need to understand how the objects are moving with respect to the camera. It would also help to recognize the interaction between objects. For example, in case of the self-driving car the knowledge about the interaction between the pedestrians will help to predict the pedestrian behavior accurately. This prediction will eventually help the self-driving car to make intelligent choices on a crowded road.

Background

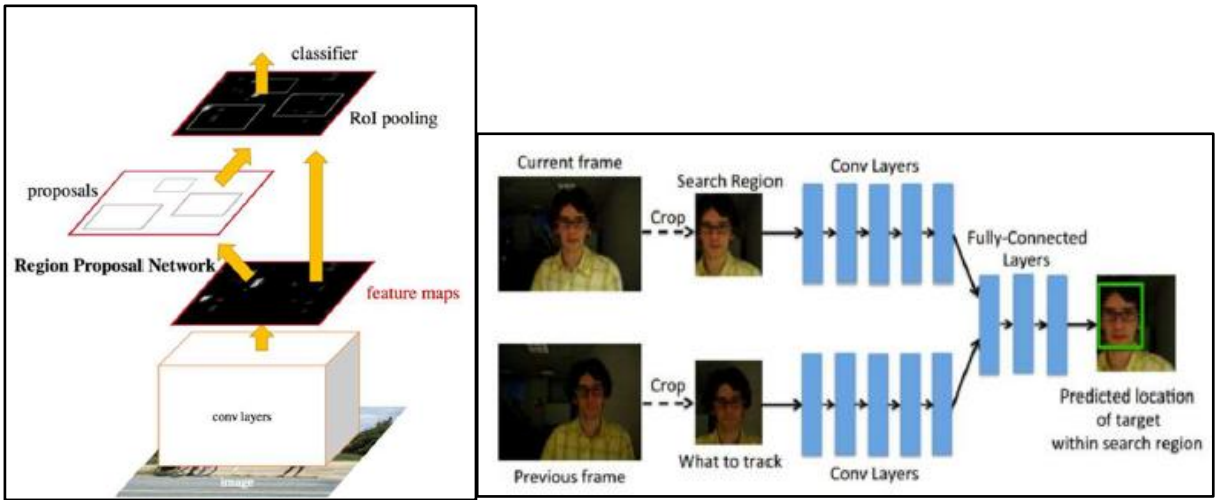
Traditionally, computer vision approaches primarily used Joint Probabilistic Data Association (JPDA) filters and Multiple Hypothesis Tracking (MHT). Most of the approaches are not suitable for real-time applications such as autonomous navigation as related problems are intractable. These approaches rely on understanding various features or cues from the images. These features include point features, color, intensity, optical flow, gradient, pixel-comparison, region covariance matrix, depth etc. Using these features, the model measures similarity and differences between observations. These approaches are also known as observational models. Other type of models tries to build appearances of the objects or a global model. For example, most of these model use momentum as a feature.

We can capture these requirements using a simple model. At the least, we should be able to identify the objects in the video. To get a better understanding, we can track these objects by establishing object correspondence between frames. We can extend this model further and estimate the object depth from the camera. Such rich data can then be processed to provide additional insights such as the closet object to the camera. This tracker is significantly faster than previous methods that use neural networks for tracking, which are typically very slow to run and not practical for real-time applications. The tracker mentioned in this paper uses a simple feed-forward network with no online training required. The tracker learns a generic relationship between object motion and appearance and can be used to track novel objects that do not appear in the training set.

Methodology

- Feed frames of a video into a neural network,
- The network successively outputs the location of the tracked object in each frame.
- We train the tracker entirely with video sequences and images. We use the MOT dataset to train the tracker. The tracker was based on previously available tensor flow code
- Through our training procedure, our tracker learns a generic relationship between appearance and motion that can be used to track novel objects at test time with no online training required.
- For the first part of the problem i.e. Detection, pre-trained Faster-RCNN is used. This model combines CNN to propose the region of interest and a region-based (R) CNN module that detects the presence of the object in these regions.
- For the object correspondence tracking, GOTURN (Generic Object Tracking Using Regression Networks) is used to track a single object in a video.
- GOTURN, trained on offline videos, uses images at time ‘t’ and ‘t-1’ of an online test video, crop them, and feed them individually in different CNNs.
- The output of these CNNs is then fed to another neural network that tries to establish an object correspondence by trying to look for similar features nearby the original object.
- Train our tracking model by utilizing pairs of subsequent frames with labels. Each video in MOT has over 200 frames and the overall training dataset includes 200k pairs of frames that

could be used for training. We note that the MOT dataset is focused exclusively on tracking people



(a) Faster R-CNN **(b) GOTURN**

http://cs231n.stanford.edu/reports/2017/pdfs/630.pdf

Experimental Design

Dataset:

The dataset contains videos from various scenarios: both static and moving cameras, low and high image resolutions, varying weather conditions and times of the day, viewpoints, pedestrian scale, density, and more

Evaluation Measures:

Precision measures how well the objects are localized i.e. the misalignment between predicted and the ground truth bounding box. While accuracy evaluates how many distinct errors such as missed targets (FN), ghost tracks (FP), or identity switches (IDSW) are made. We evaluate the performance of our tracker primarily using various scenes of the MOT benchmark video database

Software & Hardware Requirements:

Python based Computer Vision and Deep Learning libraries will be exploited for the development and experimentation of the project. Tools such as Anaconda Navigator, Python, and libraries such as Tensorflow, and Keras will be utilized for this process.

References

[1] Yi Wu, Jongwoo Lim, and Ming-Hsuan Yang, “Online object tracking: A benchmark,” in CVPR, 2013, pp. 2411–2418.

[2] Matej Kristan, Roman Pflugfelder, Aleˇs Leonardis, Jiri Matas, Luka Cˇ ehovin, GeorgNebehay, Toma´sˇ Voj´ırˇ, and Gustavo et al.Fern´andez, “The visual object tracking vot2014challenge results,”in ECCVW, 2015, pp. 191–217.

[3] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas, “Tracking-learning-detection,” PAMI, vol. 34, no. 7, pp. 1409–1422, 2012.

[4] Sam Hare, Amir Saffari, and Philip HS Torr, “Struck: Structured output tracking with kernels,” in ICCV. IEEE, 2011, pp.263–270.

[5] Kaihua Zhang, Lei Zhang, and Ming-Hsuan Yang, “Real-time compressive tracking,” in ECCV. Springer, 2012, pp. 864–877.
[6] Jo˜ao F Henriques, Rui Caseiro, Pedro Martins, and Jorge Batista, “High-speed tracking with kernelized correlation filters,”PAMI, vol. 37, no. 3, pp. 583–596, 2015.
[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, “Imagenet classification with deep convolutional neural networks,”in NIPS, 2012, pp. 1097–1105.
[8] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in ICLR, 2015.
[9] Hanxi Li, Yi Li, and Fatih Porikli, “Robust online visual tracking with a single convolutional neural network,” in ACCV.Springer, 2014, pp. 194–209.
[10] Hao Guan, Xiangyang Xue, and An Zhiyong, “Online video tracking using collaborative convolutional networks,” in ICME. IEEE, 2016, pp. 1–6.
[11] Martin Danelljan, Gustav Hager, Fahad Shahbaz Khan, and Michael Felsberg, “Convolutional features for correlation filter based visual tracking,” in ICCVW, 2015, pp. 58–66.