Generalized Appearance Features for Object Tracking

Andrew Liu (BS EECS 2018)
Dennis Tan (BS EECS 2018)
May 8, 2017

***Abstract*—Object tracking has become an important field in computer vision and robotics. In particular, models learning to move require some tracking protocol for navigating its environment and preventing collisions. This is a difficult tasks if an objects appearance changes while we are tracking. For a self-driving car, changes in lighting and perspective shouldnt disrupt tracking of pedestrians and other cars. One practical area for invariant object appearance vectors are in re-identification tasks, primarily through person re-identification networks [1][2][3]. Given two pictures of an object, our goal is to determine whether they share an identity. We accomplish this through the use of Siamese CNN network assigning pair-wise scores to images on whether they are the same person or not. The Siamese CNN learns a feature representation that encodes a generalized idea which is robust to changes in orientations, direction, lighting, and views. Combined with Faster-RCNN object proposals, we have a way to match between frame tracks and apply a Kalman filter for missing proposals in the case of occlusion.**

***Index Terms*—appearance features, object tracking, deep-ranking,**

I. Introduction

We want to solve the object tracking problem: given a sequence of video frames, determine a sequence of bounding boxes (tracklet) that tracks an object moving through each video frame. This task is difficult because the objects appearance may change in orientation, change in illumination, become partially occluded, or undergo other transformations that make it hard to continue tracking. Video object tracking is an important problem that has many applications such as self-driving cars, video surveillance, and as part of a larger artificial intelligence system (such as a robot).

Finding ways to encode images as generalized appearance vectors is different from traditional vision problems like classification and proposals. Imagenet is useful for training a classifier that can score a generic descriptor for every category. However re-identification and generalized appearance vectors are a distinctly different challenge by differentiating between different objects of the same category. Therefore an appearance feature will need to learn a representation that captures distinct features from an object.

A good appearance features should capture distinctive features of an object which can be used when re-identifying the same object. This includes robustness to perspective changes, lighting, and other manipulations which could potentially cause a model to incorrectly associate an object. One particular challenge is formulating an effective loss function to optimize. Beyond positive and negative labels, the model has to learn how to encode images to appearance features as well as how to decode appearance features to its prediction an object's label.

Finally we want to combine methods for proposing tracklets with our new affinity measurement with generalized appearance features. We outline the following contributions in the paper:

1) Develop a new method for encoding generalized appearance feature vectors.
2) Decode an identity prediction score given two appearance features.
3) Demonstrate a tracking algorithm built with the new model.

A. Previous Work

There's been a lot of work on re-identification tasks and appearance features. Fengwei Yu et. al created a framework for people tracking using appearance features. They trained a fully connected network over Pedestrian dataset and extracted appearance features at the last convolutional layer. These appearance features were combined with a matching algorithm and Kalman filter for estimated tracking [3]. Unfortunately their appearance feature encoding was specifically trained on people and is therefore potentially unreliable for more complex objects like cars.

Shi-Zhe Chen et al. proposed a deep-ranking person re-identification alternative to Fengwei Yu's work. They proposed training a model to rank pairs of images and learning to rank similar images from an assortment of images [1]. Therefore images that ranked similarly are more likely to come from the same image. Chen et al.'s method varies from POI: Multiple Object Tracking because they trained an encoder on groups of objects as opposed to stand-alone images. This allows the network to be flexible with how it distribute information over its appearance feature.

Chopra et. al showed that a discriminative training scheme could be used to train a model to predict similarity on faces. They used a Siamese network, two input CNN which share weights and back-propogation rules, to train a find similarity measure for faces [6]. Our goal with using a discriminative training scheme is to help our model learn a similarity prediction scheme based on appearance features as an alternative to Yu et. al's cosine similarity.

II. Data

Due to the accessibility of labeled images of people identity, most appearance feature papers focused on methods for

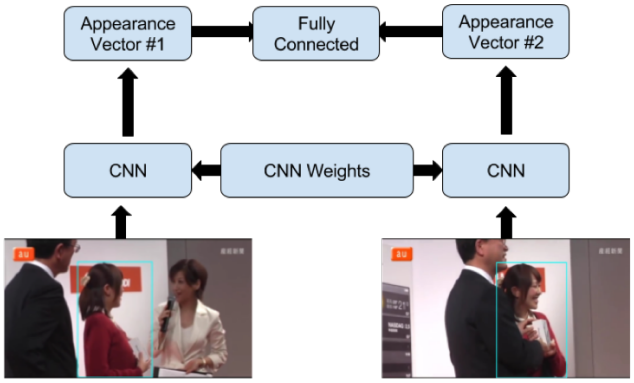


Fig. 1. Siamese Network training with a discriminative fully connected over annotated bounding boxes from YT-BB.

encoding distinctive features of people. These do not translate well to generating features for other objects like cars. A person appearance feature encodes information like shirt pattern and color which would not translate at all for cars.

On February 6, 2017, Google released the Youtube-Bounding Box dataset; an annotated video bounding box dataset over 23 categories [5]. This dataset was critical to making generalized appearance features work since it provided a way for training models over a multiple categories while providing an indicator on each object's identity. We downloaded 6 GB of downsampled annotated frames worth of images with a total of 17,000 annotated image sequences. This represents a fraction of the available data set. If we had more time, it could be possible to use the entire data-set with more parameters to get an even better feature encoding.

Due to the different categories, the expectation is that a feature vector encoding will be forced to learn unique characteristics that are useful for re-identifying any objects. In addition, each video also has many manipulations performed on the annotated boxes. Previous appearance feature works trained on people that have the same pose, scale, and are entirely visible. Youtube-BB enables us to incorporate adversarial manipulations in the worst case to get robust appearance features.

III. Training

Our training strategy consisted of randomly sampling k of 17,000 image sequences and within each sequence, randomly sampling n frames to form a group of images (a sequence typically had anywhere betwen ten to fourty annotated frames). Images that come from the sequence are assumed to share the same identity since YT-BB is designed to track a single object in a single sequence. Within the group we pairwise combine every image including with itself, assigning a positive label if the objects come from the same image sequence and a negative label if they come from different sequences. This forms a batch size of $(nk)^2$ pairs of images.

Our encoding model is a Siamese network which shares convolutional weights and back-propagation on its two inputs. The outputs of the Siamese net are appearance features. The Siamese net consisted of five convolutional structure similar to AlexNet[4].

Fig. 2. Positive labels come from the same video sequence, negative labels come from different video sequences.

On top of the Siamese net, we have a fully-connected network which predicts positive or negative label given two appearance features. This is effectively a decoder network which estimates how similar the appearance features. Ultimately this is a better proxy for similarity than cosine similarity since the occurrence of a certain index in an appearance vector may be positive correlated with other indices. The loss function is a softmax cross entropy over the probability that they are the same or different objects.

Training time typically took 6-8 hours and 11,000 epochs before converging to a stable loss. The Siamese-net and fully connected saw about 1.7 million (not necessarily unique) images. The training was done on one Nvidia Titan GPU.

IV. Tracking Algorithm

With a method for predicting the identity of bounding boxes, we want to apply it for a tracking algorithm. Yu et. al demonstrated a framework for tracking which achieved relatively high performance on tracking people. Given an appearance feature, their work used cosine similarity to assign scores between the previous frame's tracklet and the current frame's proposals. When an object fails to detect, they used a Kalman filter to estimate it's trajectory until it is detected again [3].

We seek to emulate a similar structure replacing a few key components.

1) Initialize an object proposal mechanism to get bounding boxes.
2) Pass proposals into Siamese networks to estimate the affinity between current frame's proposals and their previous frame's tracklets.
3) Based on the Siamese's affinity score, append bounding boxes to tracklets.

Fig. 3. A tracklet constructed using Siamese's affinity scoring. Despite having many proposals, the Siamese network consistently scores the top left man with each other.

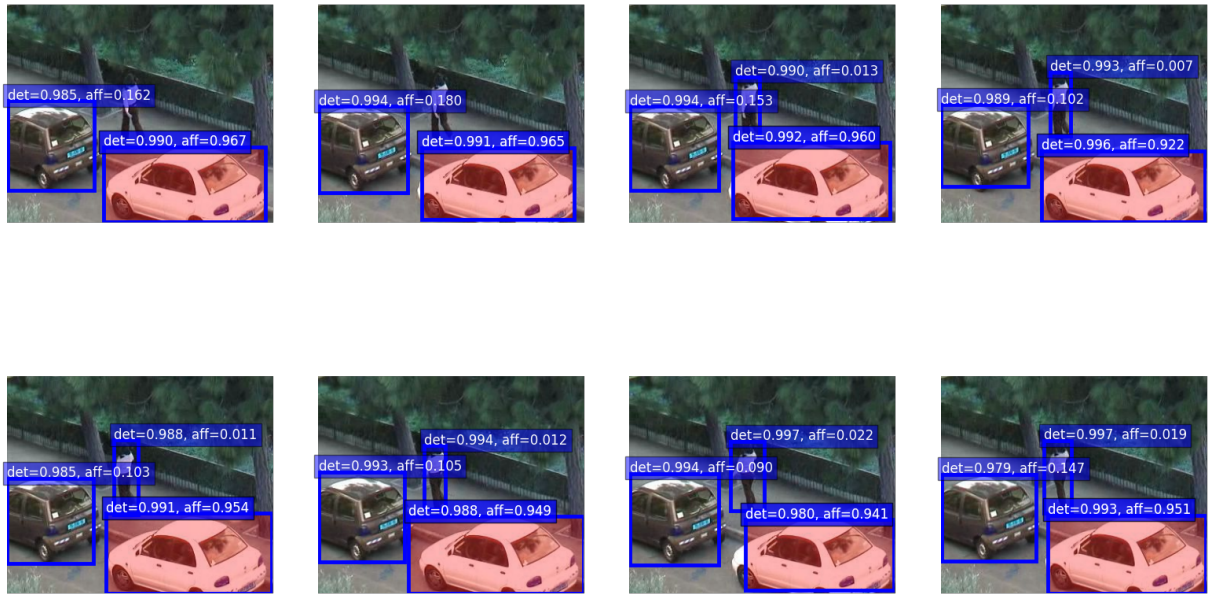


Fig. 4. Multiple car proposals from VOT Challenge, yet the Siamese network consistently correctly assigns the highest affinity to the white car.

4) Use Kalman Filter to estimate trajectory in the absence of proposals.

At the moment we have only implemented single object tracking, but multiple object tracking can be implemented under this framework if we do a ranking-like procedure as outlined in Shi-Zhe Chen et al. In spite of multiple tracklets, we have a way for distinguishing current object proposals and past tracklets.

For implementation, everything was done in Python. The Siamese-net was constructed and trained purely in Tensorflow. For object proposals we imported pre-trained weights for a Faster-RCNN network that was cross-implemented between Tensorflow and Caffe [7].

V. Results

In order to verify the correctness of the Siamese network, we asked the network to distinguish on pairs of images whether they were the same object or not from YT-BB. The network correctly predicted 74.9% of the time whether the inputted images were the same object. While this seems low, the actual

problem is extremely difficult since the object's perspective is constantly changing and may compare close ups of it to far away images.

The initial results shown above were obtained from a Youtube video and VOT Benchmark. One downside with the Youtube-BB dataset is that they sample videos every 1 second. This means that there's a lot of movement between each sequenced frame. VOT is sampled with significantly higher frequency which makes the Siamese's job easier since most object don't undergo significant change.

For the most part, our results seem primarily gated by the object proposal mechanism. If Faster-RCNN fails to return one proposal, our algorithm will wander due to the Kalman until a proposal is found again. For examples we found, affinity was consistently a good measure for whether the objects were the same or not. One good thing about this is that the results seem to generalize beyond just the training data set. Our training set for the Siamese consisted of only YT-BB annotated videos. But our results seem to work well on VOT.

The VOT Benchmark is a way to measure single object tracking in the presence of occlusion and noise. We specifically worked on VOT2013 since our Faster-RCNN could only produce rectangular bounding boxes. Due to limited time and insufficient experience working with VOT2013 integration, our testing suite took a very long time to run so we were only able to collect data over the first available sequence which tracks a person riding a bicycle. We compared against 6 other trackers and our results look very promising.

VOT Benchmark is evaluated based on percentage overlap with the ground truth annotations. Each tracker proposes a bounding box in a frame in a streaming fashion (not allowed to go back and update predictions). The evaluation pipeline compares the proposed box with the intersection of the ground truth.

Over the one sequence we benchmarked against, our Siamese tracker placed second behind SCTT in terms of accuracy around 63.2%. Our Siamese tracker lacked a little bit in terms of robustness, but that's because our Kalman filter was not sufficiently tuned for the video sequence it saw.

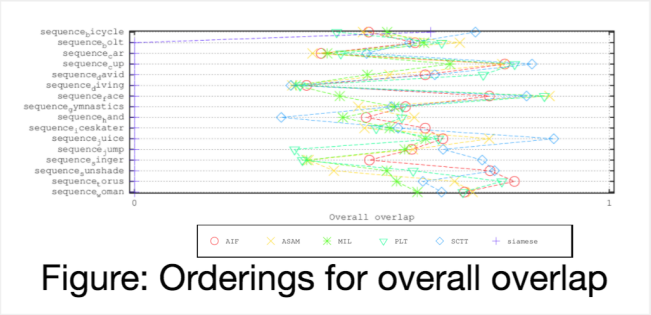
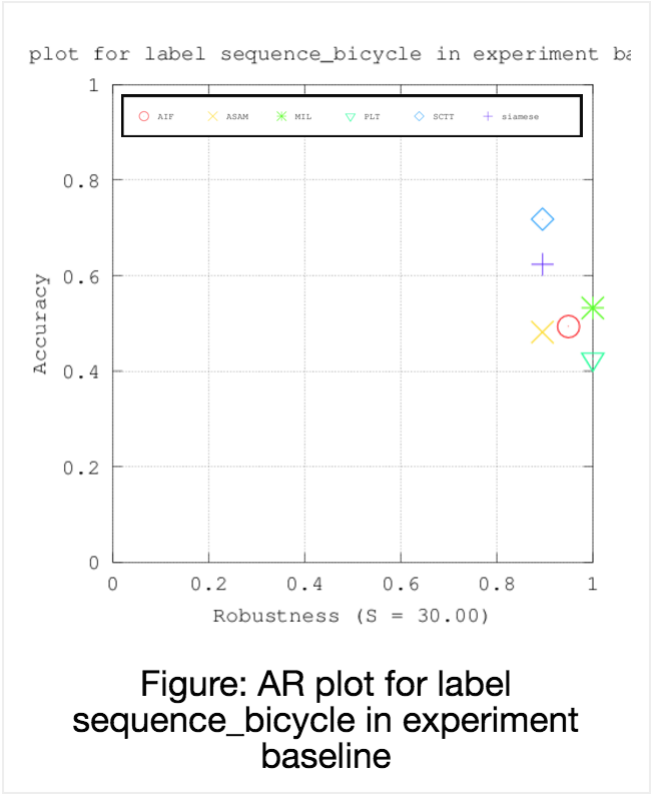


Fig. 5. Comparison for accuracy with all other trackers.



VI. Discussion

Our contributions are relatively minimal due to the incompleteness of our benchmark. We're still running the benchmark to conclusion but it won't make it into the report unfortunately.

The Siamese network proof of concept for generalized appearance features look exciting and promising direction. The benchmark information we collected performed within expectation compared to other trackers.

There's still some heavy lifting left to do with optimizing the Kalman filtering algorithm but our primary contribution in the Siamese Network shows good result for both similarity ranking and producing appearance features.

VII. Conclusion

The Siamese network implementation for generalized appearance features appear to be a good choice. In the results, the Siamese network computes affinities with relatively good frequencies. Youtube-BB empowers us to explore this space by offering millions of annotated and implicitly identity labeled objects.

The other parts of our tracking algorithm could use some more work, unfortunately we ran out of time and had other obligations as undergraduates to study for finals.

VIII. References

1) Shi-Zhe Chen, Chun-Chao Guo, and Jian-Huang Lai, Deep Ranking for Person Re-identification via Joint Representation Learning arXiv:1505.06821, 2017
2) Lianyang Ma, Xiaokang Yang, and Dacheng Tao, Person Re-Identification Over Camera Networks Using Multi-Task Distance Metric Learning, IEEE Transactions on Image Processing, 2014
3) Fengwei Yu, Wenbo Li, Quanquan Li, Yu Liu, Xiaohua Shi, and Junjie Yan, POI: Multiple Object Tracking with

High Performance Detection and Appearance Feature, ECCV 2016 Workshops, 2016
4) Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks", NIPS 2012
5) Esteban Real, Jonathon Shlens, Stefano Mazzocchi, Xin Pan, Vincent Vanhoucke, YouTube-BoundingBoxes: A Large High-Precision Human-Annotated Data Set for Object Detection in Video, Computer Vision and Pattern Recognition, 2017
6) Sumit Chopra, Raia Hadsell, Yann Lecunn, "Learning a Similarity Metric Discriminatively, with Application to Face Verification"
7) https://github.com/smallcorgi/Faster-RCNN_TF
8) https://github.com/Eschew/294-131