

**PhD**

## **Multi-object Tracking with Neural Gating Using Bilinear LSTM\_eccv18**

2019-12-09 9:29:47 PM

tracklet–detection similarity measure to decide if the new detection belongs to the same object

Missing detections are represented by an all zero vector input which might provide a useful trick to deal with falls negatives

Extension of multiple hypothesis tracking with deep learning

Main contribution as the so-called bilinear LSTM that replaces an addition operation within the internal cells by a multiplication due to its supposed equivalence with recursive least-squares regression that was used to solve the graph optimization in the earlier MHT paper

2019-12-15 11:40:45 AM

MHT works by generating multiple track proposals that are stored in a tree structure

- each node represents a detection and its child nodes represent detections in the same proposal from subsequent frames
- all detections from the same frame are at the same level of the tree
- multiple paths from the root to each of the leaves correspond to a different track proposals

Some sort of the graph optimization method called maximum weighted independent search is used to find the best proposals

gating refers to the selection of only a few detections In each frame to update the track tree instead of using all of them

neural gating is performed by using an LSTM that takes a sequence of detections corresponding to attract let as input and predicts binary output for each one of the detections

Bounding box coordinates are used as motion input and corresponding patches after being passed through a CNN are used as the appearance input to the Standard and multiplicative LSTM respectively

## PhD

outputs of the two LSTM are concatenated after L2 normalization for the final prediction

gating Is performed in each frame so that there are multiple binary labels for each sequence

Some dubious intuitions about how the various internal gates in the LSTM correspond to the previous and current templates and the rules used to update them

Some trickery is used in generating the training ground truth by inserting a different object in each tracklet with a corresponding negative label while all other frames have positive labels

There is, nonetheless, quite a bit of heuristical mess in the detection to GT association as usual

GT box jittering is used to simulate the imperfection of detections which doesn't seem to work all that well