Multiple object tracking and segmentation with R-CNN networks

Michał Daniłowicz

Embedded Vision Group, Computer Vision Laboratory,

AGH University of Science and Technology, Krakow, Poland

danilowi@agh.edu.pl

Abstract: In this work, a multiple object tracking system was implemented using two deep neural networks. The Mask R-CNN (Regions with Convolutional Neural Network features) detector was used in the tracking-by-detection approach. The detection association problem was addressed by training an additional neural network to compute an object similarity metric and using the Hungarian algorithm to find the optimal assignment. The association network was trained on the KITTI-MOTS dataset. Using a methodology compatible with the MOTS Challenge (Multiple Object Tracking and Segmentation), the implemented tracking system was evaluated and obtained the 13th place among the best methods (September 2020). The computing performance of the solution was tested on a mobile GTX 1660Ti GPU. The system achieved a processing speed exceeding 10 frames per second for 1242 x 375 pixel images, so it can be used in certain real-time applications.

System overview

The tracker uses the Mask R-CNN detector [3], which provides localisation, segmentation, and classification of objects among 80 classes. The input image features from the detector's Resnet backbone are cropped and resized for every detection to achieve scale and shape invariance (roi-pooling operation). These features are fed into a fully connected network to generate embeddings that encode the objects' visual appearance. The embeddings are then used to match detections in consecutive video frames to obtain the objects' trajectories.

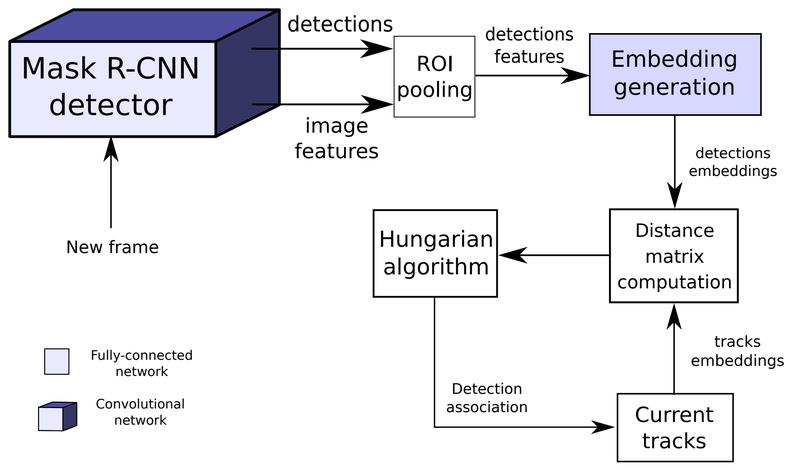

Figure 1: Scheme of the implemented tracking-by-detection system.

Detection matching

The embedding generation network is inspired by the FaceNet [4] framework for face recognition. The network maps the detection feature tensor to $\mathbb{R}^{128}$ embeddings space, where the Euclidean distance represents the visual similarity of detections (**Figure 2**). Low distance implies a higher likelihood of two given detections to represent the same tracked object.

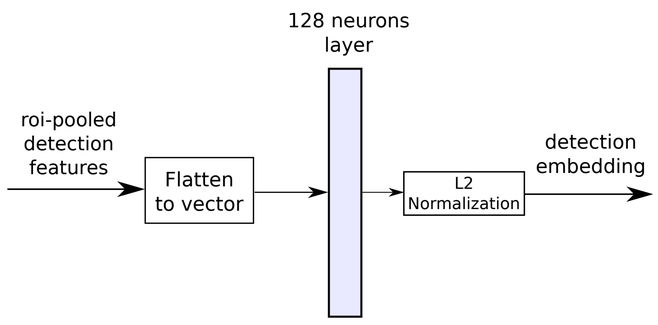

Figure 2: Detection embedding generation.

For every frame, the distance matrix is computed. It represents the similarities between new detections and all currently tracked objects. The Hungarian algorithm is then used to solve the assignment problem, while only accepting detection-track pairs with a distance smaller than the threshold α. Unassigned detections are added as new tracks and tracks undetected for β frames are removed.

Hard-triplet training

For association training, the KITTI MOTS dataset [1] was used. It contains video sequences, where every frame is labeled with objects' bounding boxes and their id number that matches them to a track. The embedding network is trained using the batch hard triplet loss (**Eq.1**).

$$L = \frac{1}{|D|}\sum_{d\in D}\max(\max_{\substack{e\in D:\\ id(e)=id(d)}}(||v_e - v_d||) - \min_{\substack{e\in D:\\ id(e)\neq id(d)}}(||v_e - v_d||) + \gamma, 0) \quad (1)$$

where: D denotes batch of detections from subsequent video frames, v_i is an embedding vector representing detection i. In other words, for every detection e in the batch, the loss equation samples the hardest positive (the furthest detection in the embedding vector space that represents the same track) and the hardest negative (the closest detection that belongs to other tracks).

Detector fine-tuning

The tracker was used for the perception system based on a drone equipped with a camera flying above the paired autonomous vehicle. The experiments shown that the pretrained Mask R-CNN detector was not able to effectively recognise cars from videos recorded by a drone (bird's eye view at altitudes 30 m and more). Therefore, the model was fine-tuned using the UAVDT benchmark dataset [2] containing photos from the drone's perspective, on which car detections were marked. The training set ground truth data contained bounding boxes, so only the ResNet, RPN (Region Proposal Network) and classifier subnetworks of the detector were trained. For the segmentation subnetwork, the original weights taken from the model learned on the COCO set were used. Fine-tuning yielded a significant improvement in detection performance from the drone's perspective.

System evaluation

The system was evaluated using the KITTI MOTS benchmark. The $sMOTSA$ score defined in (**Eq.2**) was used as the performance metric.

$$sMOTSA = \frac{\widetilde{TP} - |FP| - |IDS|}{|G|} \quad (2)$$

where: $\widetilde{TP}$ is the accumulated mask overlaps (intersection over union) of true positives, $|FP|$ is the number of false positives, $|IDS|$ is the number of identity switches among tracks in the sequence and $|G|$ denotes the total number of ground truth masks in the evaluation sequence. The tracker achieved a $sMOTSA$ score of $\mathbf{73.7}$, placing it on 13th place among the best methods in the KITTI MOTS challenge (September 2020). Qualitative results are shown in (**Figure 3**).

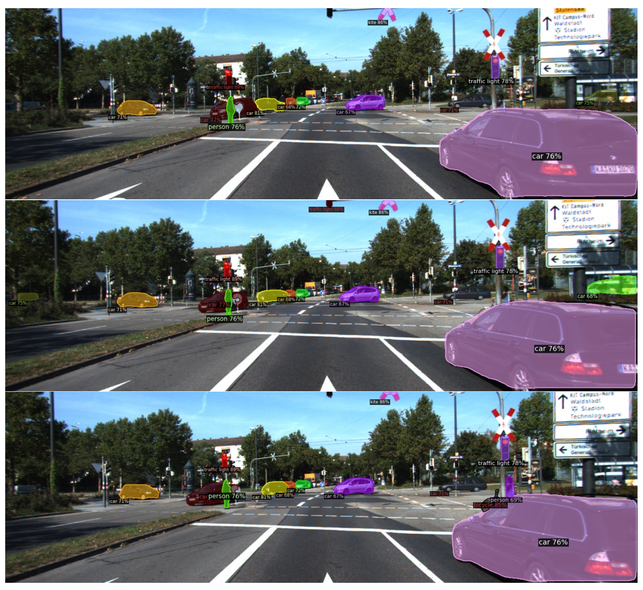

Figure 3: Tracker visualisation on subsequent frames of a test sequence.

Summary & Forthcoming Research

The tracker was implemented in Pytorch using CUDA acceleration. It achieved competitive results among state-of-the-art methods while working at 10.1 fps on GTX 1660Ti GPU. A detector fine-tuning procedure was developed to support the tracking of new object classes. The embedding matching system supports object re-identification of lost objects with low memory cost (single $\mathbb{R}^{128}$ tensor per object). Future research will be focused on: (1) implementation in embedded, low power devices, like FPGA or eGPU; (2) improving tracking performance by utilizing information about the object's movement and binary mask in the detection matching procedure; (3) optimizing the processing speed of neural network computation (quantization, model simplification); (4) training data augmentation and optimization of training hyperparameters.

References

[1] http://www.cvlibs.net/datasets/kitti/eval_mots.php.

[2] Dawei Du, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. The unmanned aerial vehicle benchmark: Object detection and tracking. *CoRR*, abs/1804.00518, 2018.

[3] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2980–2988, 2017.

[4] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015.