V for Vehicular Video Vision

John Chuter

PREDICTING

Motivation: Autonomous driving needs realtime analysis of a driving environment. One approach is image analysis, the AI for which has moved swiftly. These algorithms perform semantic image segmentation of a driving environment, on objects relevant to a driver (e.g. car, pedestrian). This project tests one of the most recent developments, Mask RCNN (2017), fine-tuned on a dataset for the 2018 CVPR WAD Video Segmentation Challenge.
I built a **Mask R-CNN implementation** that trains on Colaboratory, pulling data from Google Drive, with

different training parameters.   results = little to no

measurable improvement over baseline COCO. Inputs: images of driving scenes; outputs: the same driving scene, with "mask" applied pixel per pixel

DATASET

CVPR 2018 WAD Video DATASET; COCO Dataset:

- **source**: colored video images from car cameras, of vehicle driving environments
- **description:** these environments are labeled with 36 classes, pixel by pixel, where an output mask is generated for the input image
- While the COCO dataset has 80+ categories, and the CVPR 30+, I defined **6 categories** of interest: car, motorcycle, bicycle, pedestrian/person, truck, and bus
- This dataset was comprised of 720p annotated images of 30+ classes from Berkeley Deep Drive and Apolloscape. 92 GB of these images were used for a **training set**, with an additional 4GB for a **dev set** and a final private test set.
- Each pixel in the image contains information about object instance, and class; i.e. int(PixelValue/1000) is the labelled class, and PixelValue % 1000 is the instance id.

Mask R-CNN

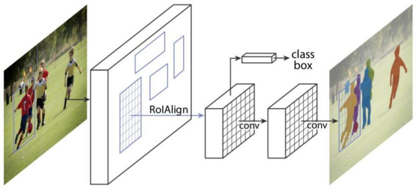


Figure 1. The **Mask R-CNN** framework for instance segmentation.

Mask RCNN is an extension of Faster RCNN that works in two stages, with a total of four parts.
Stage 1 identifies regions of interest, in two parts. The image is fed into an FPN + ResNet "backbone", which in image-to-vec fashion outputs a feature map. A Region Proposal Network (RPN) then scans over this feature map, convolutionally evaluating multiple anchors simultaneously and identifying the Regions of Interest, with the anchor and a simple foreground or background evaluation.
Stage 2 analyzes the regions considered foreground, generating masks for objects, and in parallel classifying objects to which the masks can be applied. First we take the regions identified by the RPN, pool those with ROI pooling to a consistent size, then classify those regions more deeply into specific categories, with refinements to the corresponding bounding boxes. The masks generated for objects are then applied to the classified regions.

DISCUSSION

-data management
-compute options
-models; implementations and choices
-teams, personal failures

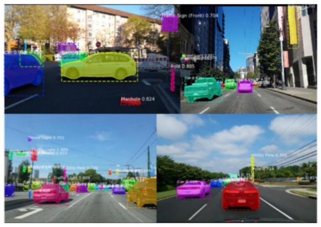
FUTURE

-implement the Data class
-confirm every step with other people
-play with other implementations

IMPLEMENTATION

Per Matterport:

- ResNet101 + FPN for CNN "backbone"
- L = L_class + L_box + L_mask
- pre-trained from MS COCO dataset
- fine-tuned head layers with WAD
- Image dimensions 1024x1024



Mask R-CNN on COCO

Configurable Hyperparameters

-learning rate = .001 - .01
-learning momentum: .9 - .8
-Mask Threshold; 32x32, 64x62, 128x128

defaults:
- STEPS_PER_EPOCH = 1000
- VALIDATION_STEPS = 50
- BACKBONE = "resnet101"
- WEIGHT DECAY = .0001

REFERENCES: https://arxiv.org/pdf/1703.06870.pdf , https://github.com/matterport/Mask_RCNN, http://cs230.stanford.edu/files_winter_2018/projects/6908784.pdf ,