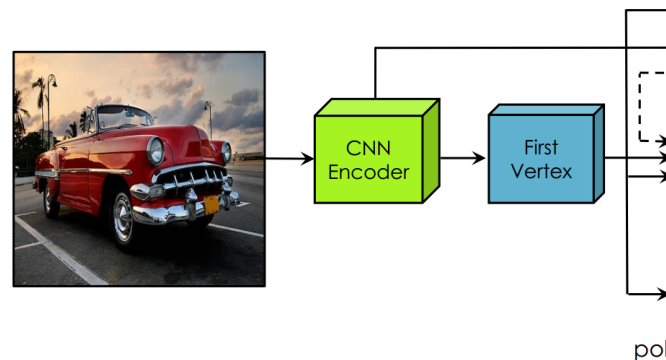

PolygonRNN++

This is the official inference code for Polygon-RNN++ (CVPR-2018). For technical details, please refer to:

An official pytorch reimplementation with training/tool code is available here

Efficient Interactive Annotation of Segmentation Datasets with Polygon-RNN++

David Acuna*, Huan Ling*, Amlan Kar*, Sanja Fidler (* denotes equal contribution)
CVPR 2018



[\[Paper\]](#) [\[Video\]](#) [\[Project Page\]](#) [\[Demo\]](#) [\[Training/Tool Code\]](#)

Usage

1. Clone the repository

```
1 git clone https://github.com/davidjesusacu/polyrnn && cd polyrnn
```

2. Install dependencies

(Note: Using a GPU (and tensorflow-gpu) is recommended. The model will run on a CPU, albeit slowly.)

```
1 virtualenv env
2 source env/bin/activate
3 pip install -r requirements.txt
```

3. Download the pre-trained models and graphs (448 MB)
(These models were trained on the Cityscapes Dataset)

```
1 ./models/download_and_unpack.sh
```

4. Run demo_inference.sh

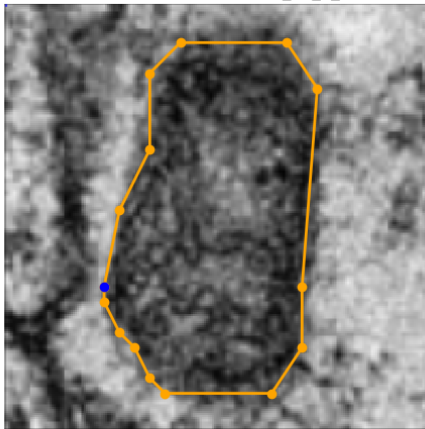
```
1 ./src/demo_inference.sh
```

PolygonRNN++ : frankfurt_000000_000294_42

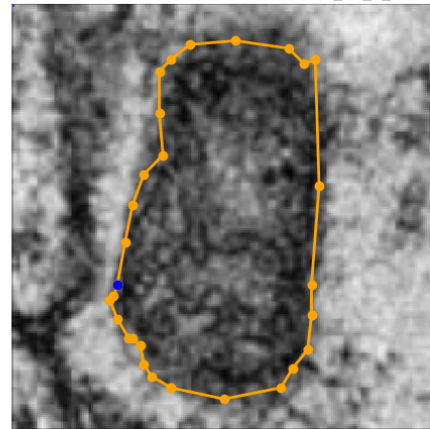


This should produce results in the output/ folder that look like

PolygonRNN++ : medical_00_5_20



PolygonRNN++ + GGNN : medical_00_5_20



Walkthrough

Checkout the ipython notebook that provides a simple walkthrough demonstrating how to run our model on sample input image crops

If you use this code, please cite:

```
1 @inproceedings{AcunaCVPR18,
2   title={Efficient Interactive Annotation of Segmentation Datasets with
      Polygon-RNN++},
3   author={David Acuna and Huan Ling and Amlan Kar and Sanja Fidler},
4   booktitle={CVPR},
5   year={2018}
6 }
7
8 @inproceedings{CastrejonCVPR17,
9   title = {Annotating Object Instances with a Polygon-RNN},
10  author = {Lluís Castrejon and Kaustav Kundu and Raquel Urtasun and
      Sanja Fidler},
11  booktitle = {CVPR},
12  year = {2017}
13 }
```