

Capstone Proposal

Image Segmentation on Vehicles

Machine Learning Engineer Nanodegree

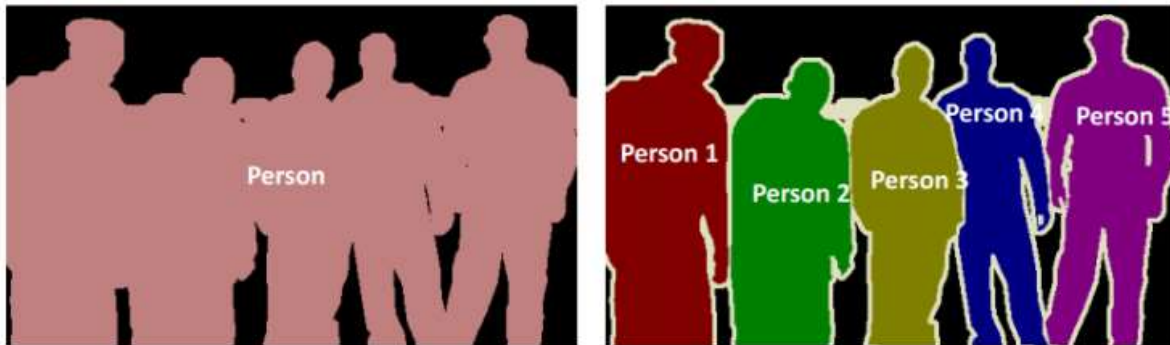
Udacity

Introduction

Domain: Computer Vision / Image segmentation

For this project, the dataset provided by Carvana wants to achieve segmentation, distinguishing the car from the image background. Semantic segmentation is the task of assigning a class to every pixel in a given image. We try to achieve this in order to differentiate various class of objects in our image.

However, we don't go into the details of differentiating each and every object in the segmentation like we do in Instance Segmentation



Semantic Segmentation

Instance Segmentation

Traditional approaches to analyze these images in supervised manner include Scale-invariant feature transform (SIFT), Bag-of-visual-words (BOV), Histogram of Oriented Gradients (HOG), etc. whereas for unsupervised approach, various clustering algorithms were used.

The application of CNNs and deep learning have greatly alleviated such detailed feature analysis with improved accuracy which is to be used in the project. Some of the major algorithms included with AWS are:

- Fully-Convolutional Network (FCN) algorithm
- Pyramid Scene Parsing (PSP) algorithm
- DeepLab V3 algorithm by Google

Motivation: Having worked on an object detection and caption generation model, I wanted to do a project on image segmentation using deep learning.

Problem Statement

The need for image segmentation is that even though Carvana takes high quality photos, bright reflections and cars with similar colors as the background cause automation errors, which requires a skilled photo editor to change. The objective is to develop an algorithm that automatically removes the photo studio background instead of relying on manual editing. This will allow Carvana to superimpose cars on a variety of backgrounds.



Implementing the Sagemaker provided model along with a possible deep learning model is the end goal of the project to achieve accurate segmentation of the image.

Detailed description of the company's motivation: <https://www.kaggle.com/c/carvana-image-masking-challenge/overview/description>

Datasets and Inputs

The dataset provided is quite large since it contains images. For present evaluation, to monitor our metrics, we'll utilize the training images and annotations provided as images, split them into: train/validation/test, and then evaluate the results.

The images provided are in '*jpeg*' while annotations are in '*gif*' format. Reformatting these images for our model would be the first step.

Dataset: <https://www.kaggle.com/c/carvana-image-masking-challenge/data>

Benchmark Model

The provided dataset has a simplistic and same greyish background as seen in the example picture below:

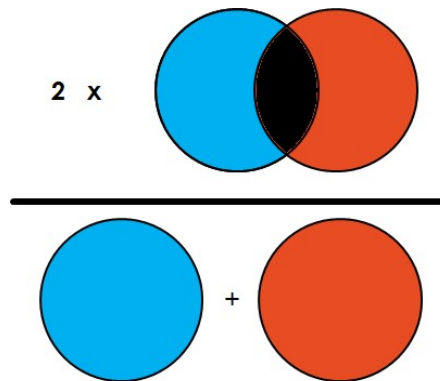


However, the actual application of the model is to use it with images in complex background/diverse scenes. Owing to that, we use the Pyramid Scene Parsing (PSP) Network which is designed to handle such scenarios instead of the default. Another option would be to use a Fully Connected Network (FCNs), but it requires several convolution steps and architecture iterations like selecting FCN-8, FCN-16,

FCN-32, etc. Lastly, the DeepLab V3 model is also a possible architecture to be looked at with improvements over the original DeepLab segmentation model by the application of Pyramid Pooling achieving similar accuracy as the PSP Net in various datasets.

Evaluation Metrics

The competition used the Dice Coefficient (F1 Score) as its evaluation metrics which is commonly used in segmentation of 2 classes.



$$\text{Dice Coefficient} = \frac{2 * \text{Area of Overlap}}{\text{Total number of pixels in both images}} = \frac{2TP}{2TP + FP + FN}$$

For model evaluation, we'll be using the Dice Coefficient. The default models on Sagemaker is monitored on another similar metric called the mean Intersection-Over-Union (mIOU).

$$\text{mIOU} = \frac{TP}{TP + FP + FN}$$

These metrics values slightly differ, but both the metrics converge similarly

Solution Statement

The objective of this project would be to achieve the best possible score on the project. Presently, AWS SageMaker provides default 'semantic-segmentation' solutions for prediction. Based on the study of model architectures, first attempt would be to use the PSP network. If the F1 scores achieved aren't satisfactory, the next step would be to optimize the hyperparameters. Lastly, a custom PyTorch model can be used if the solutions still do not converge .

As discussed previously, for model prediction and evaluation in the competition, we only utilize the train dataset since it contains about 9000 images (405 MB) which will be split into train, dev, test set with 0.8, 0.15 and 0.05 fraction of this data split respectively.

For segmentation training, we'll utilize **"ml.p2.xlarge"** instance for training and **"ml.c4.xlarge"** instance for evaluation as compute instances required by the segmentation tasks. The test dataset is 20 times larger (about 8 GB) and is hence not used for evaluation.

Project Design

The outline of the project is:

1. Load images for exploratory data analysis
2. Pre-process the dataset of train images and annotations to be split into train, validation and test sets and upload to S3
3. Creating an estimator with the desired hyperparameters for the training job and fitting the model weights
4. Deploying the model for predictions
5. Lastly, we'd like to make predictions on the test dataset and evaluate the Dice coefficient for our images and visualize the segmentation predictions.

References:

L. Chen, G. Papandreou, F. Schroff, H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation", Dec 2017 (arXiv:1706.05587)

E. Shelhamer, J. Long, T. Darrell, "Fully Convolutional Networks for Semantic Segmentation", May 2016 (arXiv:1605.06211)

H. Zhao, J. Shi, X. Qi, X. Wang, J. Jia, "Pyramid Scene Parsing Network", Dec 2017 (arXiv:1612.01105)

Metric image:

<https://towardsdatascience.com/metrics-to-evaluate-your-semantic-segmentation-model-6bcb99639aa2>

Algorithm basics: <https://www.analyticsvidhya.com/blog/2019/02/tutorial-semantic-segmentation-google-deeplab/>

Problem: <https://www.kaggle.com/c/carvana-image-masking-challenge/overview>