Image segmentation of car parts with Deloitte consulting

Farkas Attila Jakab (s230234), Konrad Kolos Drexler (s232457), Peter Vilmos Malak (s223264), Bence Kern (s201758)
DTU Compute, Technical University of Denmark

Introduction

The main project idea came from Deloitte, they wanted to develop an image segmentation model that is able to identify and distinguish car parts and use it in an insurance setting.

Deloitte.

References

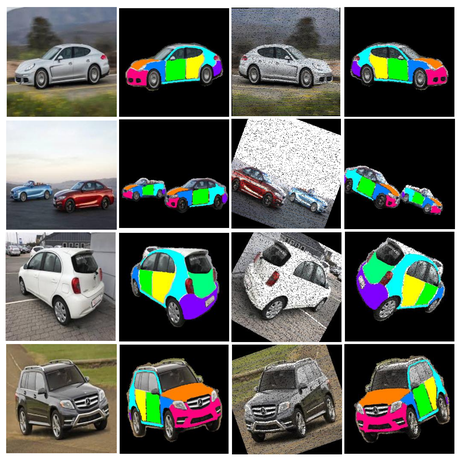
[1] Olaf Ronneberger: U-Net: Convolutional Networks for Biomedical Image Segmentation, https://lmb.informatik.uni-freiburg.de/people/ronneber/u-net/
[2] Semantic Shapes segmentation (Seth Adams): Github Link
[3] Brain Tumor Segmentation using UNET (Idiot Programmer): Github Link
[4] Image Segmentation Videos (Computer Vision Engineer): Youtube Video Link
[5] Further Reading: https://paperswithcode.com/task/image-segmentation

Data

Images:

- **CAD:** images of model cars with black and orange background

- **Real**: photos of real cars



Arrays:

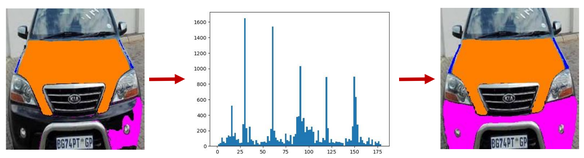
- 3 channels contain the image data (0-255)(height, width, RGB)
- 1 channel contains the class value

"color",	"description",	"class value",	"color values"
"orange",	"hood",	10	(250, 149, 10)
"dark green",	"front door",	20	(19, 98, 19)
"yellow",	"rear door",	30	(249, 249, 10)
"cyan",	"frame",	40	(10, 248, 250)
"purple",	"rear quater panel",	50	(149, 7, 149)
"light green",	"trunk lid",	60	(5, 249, 9)
"blue",	"fender",	70	(20, 19, 249)
"pink",	"bumper",	80	(249, 9, 250)
"no color",	"rest of car",	90	NA

Data Augmentation:

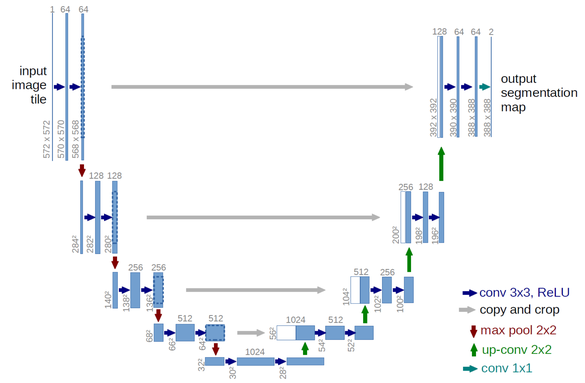
- Rotation
- Horizontal flip
- Dropout (noise)
- Gaussian blur
- Sharpening

Improving the segmented data:



Improving the background:

Model



U-Net architecture: robust and efficient way to provide localization and classification

Training

Hyperparameter optimization:

- **Optimizer:** RMSprop | Adam | SGD
- **Learning rate:** 1e-2 | 1e-3 | 1e-4
- **Batch size:** 16 | 32 | 64
- **Loss function:** Categorical cross entropy loss | DiceBCE loss | Dice loss

The best model:

- **Optimizer:** Adam
- **Learning rate:** 1e-3
- **Batch size:** 32
- **Loss function:** DiceBCE loss

Results

	Full Dice	Car Dice	Parts Dice		Full Dice	Car Dice	Parts Dice
	0.9553	0.8423	0.8107		0.9315	0.8596	0.7574
	0.9469	0.8823	0.9099		0.9411	0.8641	0.8226
	0.9476	0.8659	0.8579		0.9264	0.8753	0.7845
	0.9243	0.8511	0.7680		0.6041	0.5526	0.3274
	0.9058	0.7854	0.7054		0.8796	0.5208	0.1476