Affordable Self Driving Cars and Robots with Semantic Segmentation

Gaurav Bansal, Jeff Chen, Evan Darke

{gauravbs, jc1, edarke}@stanford.edu - December, 2017

Motivation

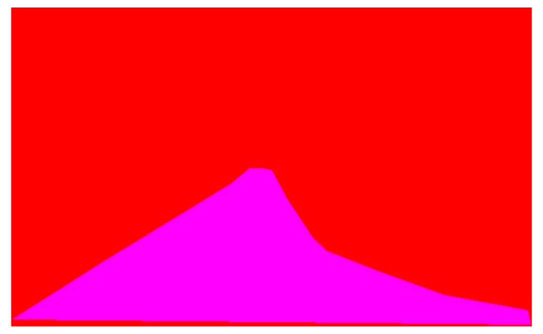
Semantic segmentation is emerging as a powerful tool for autonomous driving perception as it could simultaneously detect dynamic objects and free road spaces. It also has the potential to remove the dependence on LiDAR and High-Definition (HD) maps which are expensive to build and license.

There are applications in other domains: Robotics, Drones etc.

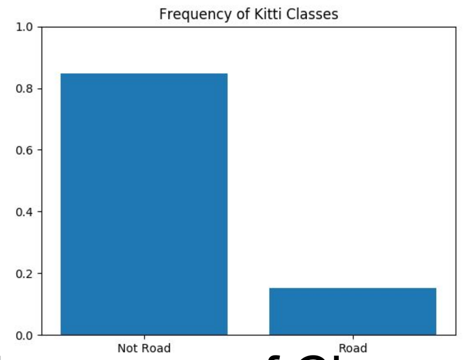
Datasets

KITTI: Popular dataset for Autonomous Driving. We performed initial experiments on KITTI (289 labeled images of 160*576 pixels with two classes - Road and not Road) to gain insights on semantic segmentation.

Training set Image



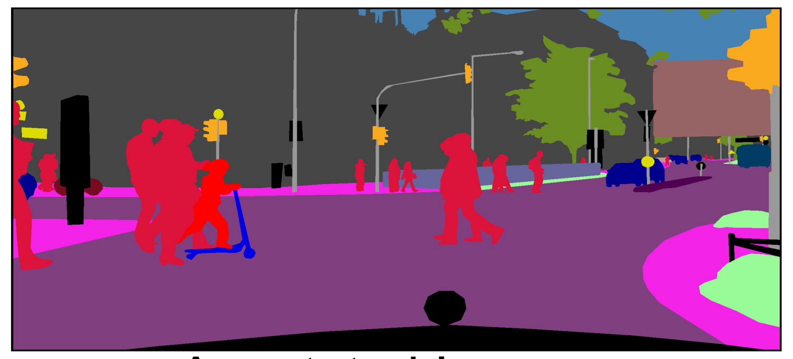
Annotated Image



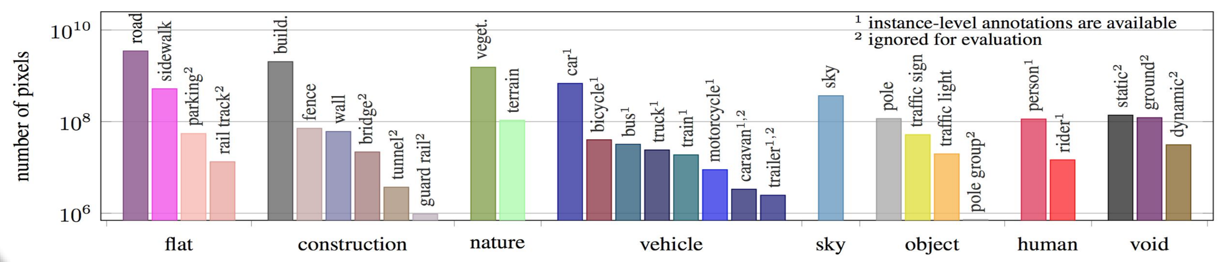
Frequency of Classes

Cityscapes: Large-scale database which focuses on semantic understanding of urban street scenes. 3475 fine annotated images of 1024*2048 pixels with 8 categories and 30 classes.

Training set Image



Annotated Image



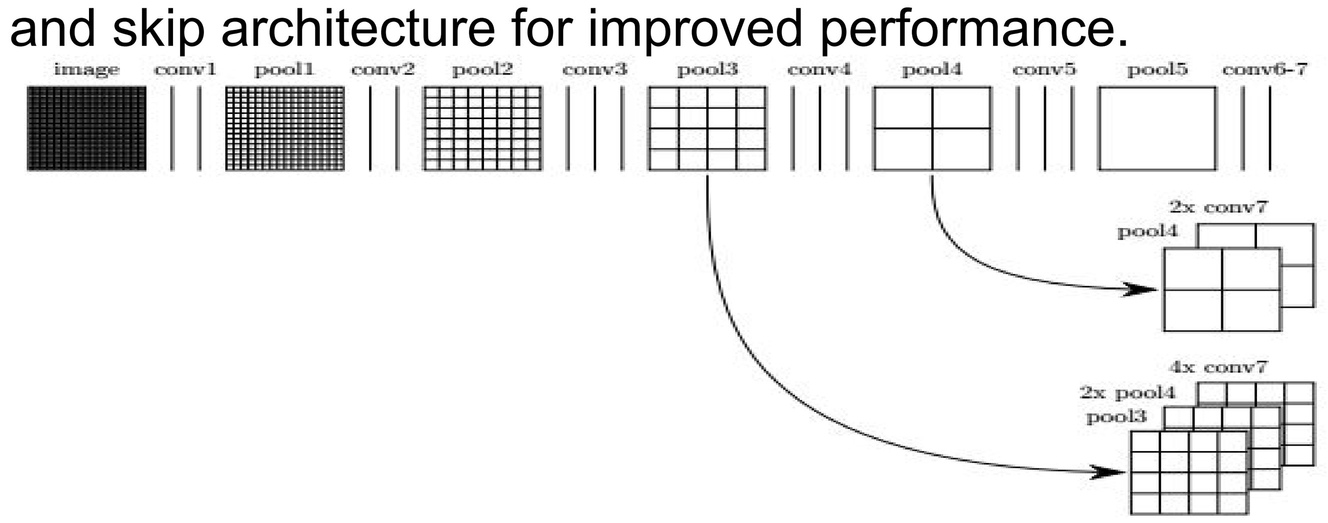
Number of finely annotated pixels per class and their associated categories

Methodology

We report the intersection over union (IoU) averaged over each class. IoU is more robust than per pixel accuracy when there is a large class imbalance. For example, always predicting 'Not Road' on KITTI would yield an accuracy of 85% but an IoU of only 42.5%

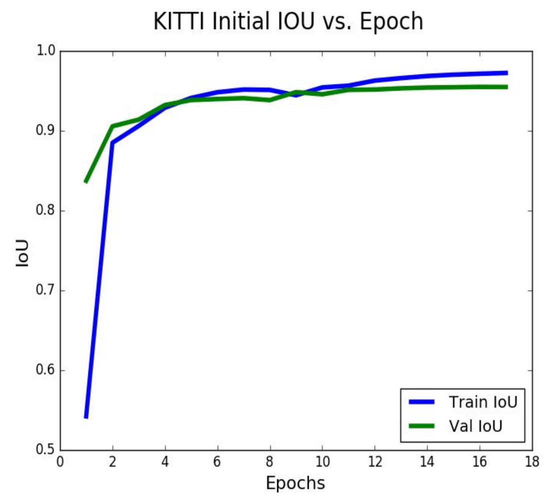
$$IoU = \frac{TP}{TP + FP + FN}$$

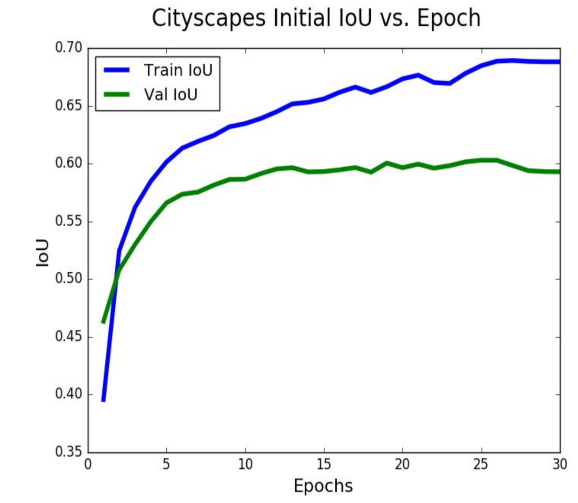
Fully-Convolutional Network: Seminal paper on Deep Learning for Semantic Segmentation. Based on VGG-16 Convolutional neural network architecture. Employs techniques of transposed convolution and skip architecture for improved performance.



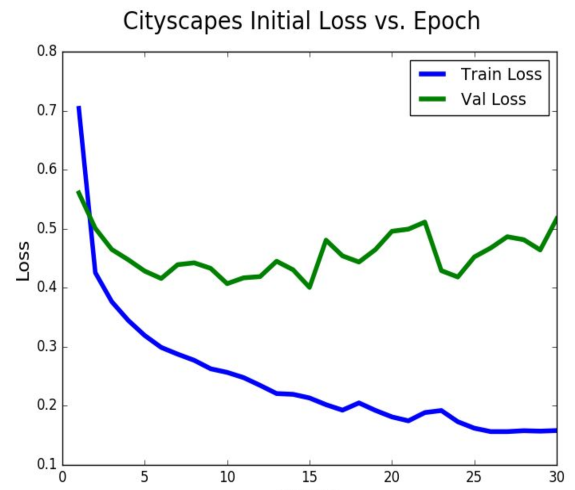
Initial Results and Experiments

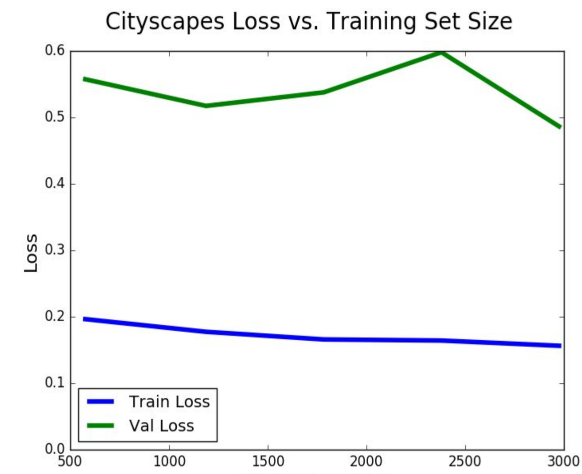
Performed hyperparameter tuning: Learning Rate (.0001), Batch size (4), Optimization algorithm (Adam), Dropout keep probability (0.8). Resized Cityscape images to 1/8th in each dimension (limited by compute)





Obtained 96% Val IoU on KITTI. Cityscapes achieved 60% mean Val IoU and suffers from overfitting, analyzed below:

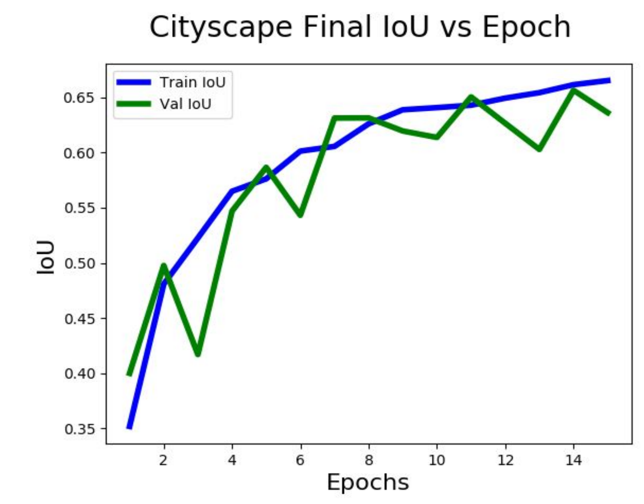


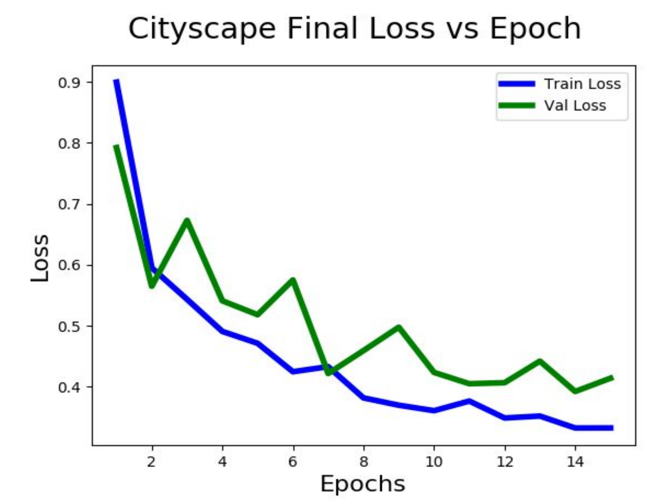


Overfitting: validation loss starts to increase while training loss is still decreasing. Additional training data could help. Experiments attempted: Regularization, Dropout, Data Augmentation, and more data.

Final Results and Analysis

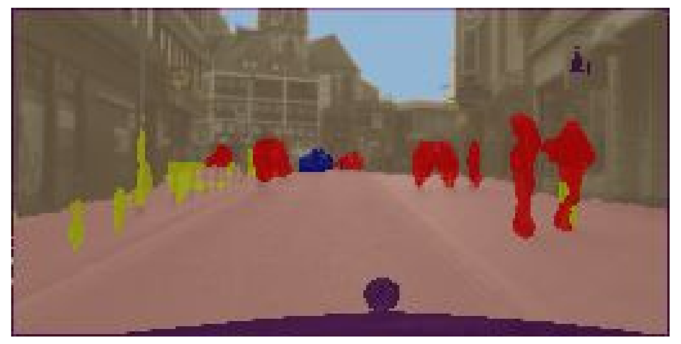
Data Augmentation was useful in reducing overfitting, we were able to achieve a final Val IoU score of 66%. Many other experiments did not significantly improve performance: L2 Regularization, Dropout, simpler architectures, and image flipping.





Left Figure shows well segmented image with cars. Right Figure shows humans and signs perform much worse with humans connected together and fuzzy delineation of sign posts:

Good Segmentation



Poor Segmentation

Confusion matrix shows objects (poles and traffic signs) and humans are segmented much worse than other categories. Ablative analysis shows L4 skip layer is most important.

	void	flat	const	object	nature	sky	human	vehicle
void	**69%**	14%	11%	1%	3%	1%	1%	1%
flat	0%	**97%**	1%	0%	1%	0%	0%	1%
const	0%	1%	**91%**	1%	5%	2%	0%	1%
object	1%	4%	45%	**24%**	21%	1%	1%	2%
nature	0%	1%	6%	0%	**92%**	0%	0%	0%
sky	0%	0%	3%	0%	4%	**93%**	0%	0%
human	1%	5%	30%	3%	5%	0%	**49%**	7%
vehicle	0%	3%	9%	0%	3%	0%	2%	**82%**

Network	**Val Loss**	**Val IoU**
FCN-8	0.42284	0.594875
Remove L3-Skip	0.454842	0.569899
FCN-16	0.437881	0.568810
Remove L4-Skip	0.542143	0.509061
FCN-32	0.530987	0.488239
Layer 4-out	0.428264	0.571850
Layer 3-out	0.432834	0.573281

Next Steps

- Analyze and improve performance of objects and humans
- Deeper architectures (ResNet 50, Google DeepLabv3) to reduce underfitting
- Analyze faster inference-time networks such as SqueezeNet