A Review and Validation of Novel Object Detection Models for Autonomous Agents

Ketan Anand
School of Electrical and Computer Engineering
Georgia Institute of Technology
Atlanta, U.S.A
kanand@gatech.edu

Kanupriya Anand
Department of Information Technology
R. V. College of Engineering
Bangalore, India
kanupriya_anand@outlook.com

***Abstract*—Semantic segmentation is an upcoming and important part of research in the field of Computer Vision and Deep learning. It involves the task of identifying different objects in an image and assigning classes to them. This particular exercise is very useful for autonomous driving applications in which identification of the objects in the vehicle's environment is of paramount importance. To that extent, we choose KITTI autonomous driving data-set as our research interest and focus on reviewing the four semantic segmentation models from the KITTI Dataset leaderboard, based on their IoU (Intersection-over-Union) score and suggest improvements to one of the models to improve its score. We also present an analysis to explain each model's superiority over another in terms of performance and suggest changes in the network architectures, loss functions, and optimizers to further improve each model's IoU score.**

***Index Terms*—Semantic Segmentation, Depth Estimation, Novel Softmax Loss, Intersection-over-Union, Convolutional Neural Network, Autonomous Vehicles**

I. Introduction

Semantic segmentation refers to the task of identifying different objects in a given image, identifying which class they belong to, and clustering these objects based on this class. This is a pixel-level classification task as each pixel in the image needs to be assigned to a particular class and thus group pixels based on some shared similarity.

CNNs are a subcategory of deep learning algorithms that find use in many image-processing applications these days. This is because their convolutional layers process and reduce the high dimensionality of images and extract features from them without losing information. Most state-of-the-art semantic segmentation models today have a CNN backbone.

Semantic segmentation has multiple applications in different fields. It is used in video surveillance to identify intruders; recognition tasks such as face detection, fingerprint detection and iris detection; and in medical imaging to plan surgeries, measure tissue volume and locate tumors and other anomalies. It also has significant applications related to autonomous driving such as pedestrian detection and brake light detection.

II. Methodology

This paper focuses on applying semantic segmentation for autonomous driving and thus we were choosing between CityScapes [8], KITTI [5], Vistas, DUS, and CamVid datasets.

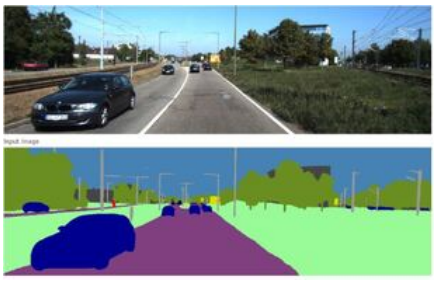

Fig. 1. Sample output of a semantically segmented image

For the scope of this paper, we limited our work to the KITTI image segmentation dataset .

Our primary target metric to evaluate model accuracy is **IoU (Intersection-over-Union)**. IoU quantifies the degree of overlap between two pixels/classes/boxes. In the case of object detection and segmentation, IoU evaluates the overlap of the Ground Truth and Prediction region. Ground truth is defined as the labeled/annotated images usually provided with the training data set used as a reference to evaluate our prediction region(pixels in this case). IoU is defined as follows:

$$IoU = \frac{TP}{TP + FP + FN} \tag{1}$$

where:

True Positive (TP): The intersection of the pixels of the ground truth and the predicted output (a.k.a. segmentation mask)

False Positive (FP): These are the predicted pixels that are outside of the ground truth.

False Negative (FN): Number of pixels that failed to be predicted by our model which is indeed part of ground truth.

We looked at the KITTI image segmentation leader-board and chose the following four models which can be classified according to figure 2. The following sections go over these models, their architecture, and performance.

A. *SGDepth*

Overview: The paper proposes a self-supervised semantically-guided depth estimation (SGDepth) method

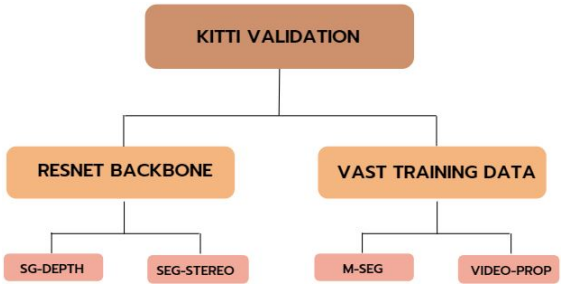

Fig. 2. Classification of the four models selected

to deal with moving dynamic-class (DC) objects, such as moving cars and pedestrians, which violate the static-world assumptions typically made during the training of such models. It proposed an approach where segmentation and depth estimation are performed in two different domains and the feedback from the segmentation helps to improve the sharpening of the depth boundaries and thereby increasing the performance of the model. However, the focus of our paper remains on the semantic segmentation part, so we will be reviewing that part only which is in the segmentation domain.

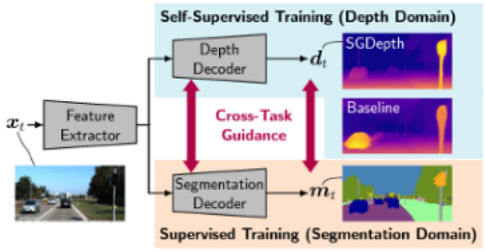

Fig. 3. Overview of SGDepth's framework

Image Segmentation Loss function: The model is trained by imposing a weighted cross-entropy loss between the posterior probabilities of the network y_t and the ground truth labels $\overline{y_t}$ with class weights w_s. Finally, again averaging over all pixels, the loss function for the image's posterior probabilities $y_{t,s} \in I^{HXW}$ of class s is defined as

$$J_t^{\text{ce}} = -\left\langle \sum_{s \in \mathcal{S}} w_s \overline{y}_{t,s} \odot \log\left(\boldsymbol{y}_{t,s}\right) \right\rangle \tag{2}$$

where $\log(\cdot)$ is applied each element of $\overline{y_t}$ and $\odot$ is element wise multiplication between two matrices.

Network architecture: The general architecture of both depth and image segmentation is based on the U-Net architecture, i.e. an encoder-decoder network, with skip connections, enabling us to represent both deep abstract features as well as local information and using pre-trained ResNet18 as its encoder (trained on ImageNet [6]). But our focus, the segmentation part has the same architecture as the depth estimation except for the last layer having S feature maps, whose elements are converted to class probabilities by a SoftMax function. To illustrate how U-net architecture refers to the image 4.

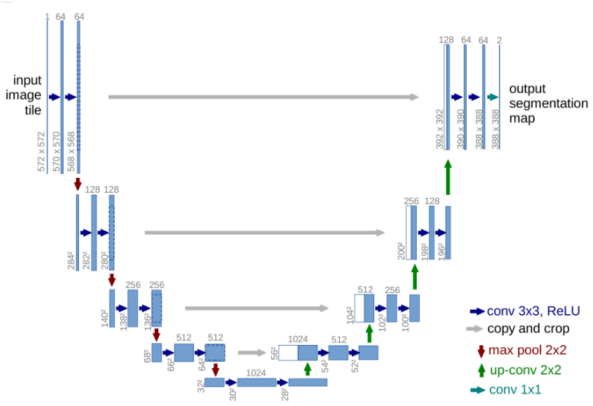

Fig. 4. Architecture of a U-Net

The input image is repeatedly convolved and then down-sampled using Max pool operations which helps in extracting information from the images. Now the image is up-sampled by up convolution to increase the resolution. At the same time, in order to localize, high-resolution features from the contracting path are combined with the upsampled output.

Training: For training the semantic segmentation the model utilized the Cityscapes dataset while at the same time they used different subsets of the KITTI dataset for training the depth estimation. The paper trained their models for 40 epochs with the Adam optimizer and batch sizes of 12. The learning rate is set to 10^{-4} and reduced to 10^{-5} after 30 epochs.

Results: The paper has multiple variants of depth estimation stated which help in improving the IoU metric of our segmentation due to the increased boundary predicting capability arising from it.But for the scope of our review we limited ourselves to the image segmentation part. The Model trained with segmentation only gives us a IoU of **43.1**.

Fig. 5. SGDepth Segmentation Only Output

B. SegStereo

Overview: SegStereo is a state-of-the-art semantic segmentation model that is used to understand objects encountered on the road by autonomous vehicles. SegStereo leverages semantic segmentation to improve disparity prediction, a feature that is useful in object detection and depth estimation. Semantic segmentation involves assigning each pixel in an image a label (such as a car, tree, road, etc.). When introduced into a disparity regression pipeline, an improvement in predicting disparity was a scene which in turn improves scene understanding in urban settings. The architecture implemented by SegStereo

sits on a ResNet backbone and introduces a novel semantic softmax loss to improve disparity prediction. [3]

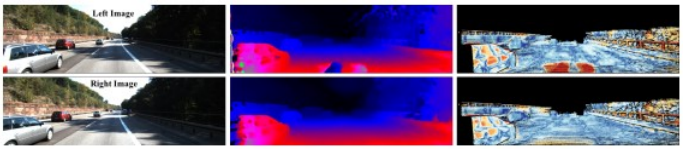

Fig. 6. Disparity Map Generation from Stereo RGB Image Pairs

Network Architecture: Seg Stereo is built on a ResNet-50 backbone but instead of computing disparity between raw pixels, the shallow part of ResNet-50 is used to extract image features. This process is called robust to local context information encoding. It is in this context that semantic clues improve disparity estimation. Semantic features from the right map are warped and reconstructed to get the left map. Pixel-wise classification is then carried out on the left semantic map with the objective of improving loss function L_{seg}. The learning rate used is a "poly" learning policy that adapts according to the distance from minima (including poly learning iterative equation).

A segmentation network is utilized to compute semantic features for left and right images respectively, sharing shallow layer representation with disparity network. The left transformed disparity features, the correlated features and the left semantic features are concatenated as hybrid feature representation. Here, semantic cues are preliminarily introduced to the disparity network as Semantic Feature Embedding.

Training: Evaluation of Seg Stereo is performed on CityScapes and KITTI. First, the model was trained, validated, and tested on CityScapes using 2975, 500, and 1525 images. Parameter fine-tuning was subsequently performed on the KITTI dataset using 200 training images and 200 more to test.

$$\mathcal{L}_r = \frac{1}{N_\nu}\Sigma_{i,j\in\nu}\|\mathcal{D}_{i,j} - \hat{\mathcal{D}}_{i,j}\| \tag{3}$$

When training the disparity network, the semantic loss L_{seg} is propagated back to the disparity branch through a semantic convolutional classifier and feature warping layer. Along with basic photometric loss L_p or regression loss L_r, semantic loss Lseg imposes extra object-aware constraints to guide disparity training. The experiments prove that semantic loss regularization can effectively resolve the local disparity ambiguities, especially in the unsupervised learning period.

Results: The SegStereo results as stated on the KITTI benchmark are as follows: The model IoU (might want to explain IoU at the beginning) is 59.10 and its disparity prediction error is 2.25 pixels. The ground-truth disparity maps are directly applied to train our SegStereo model. As KITTI's stereo dataset is too small, the model is pre-trained on the CityScapes dataset. Although the disparity maps computed by the SGM algorithm contain errors and holes, they are useful for our model to get reasonable accuracy. The maximum iteration is set to 90K. Different from unsupervised training, here the disparity regression loss L_r plays the major role.

C. *MSeg*

Overview: There are multiple datasets and benchmarks for semantic segmentation such as Cityscapes, COCO, Mapillary, and ADE20K but they correspond to different domains. For instance, Mapillary has data captured while driving outdoors, whereas ADE20K has data captured from hikes. In order to build a model that performs well in general and is able to accurately perform semantic segmentation on a given image without knowing anything about the conditions where it was captured, we would need to train a model using data from each of these individual datasets.

However, the issue with naively training using a combination of these datasets is that the accuracy of the model is low because of the difference in their taxonomies due to a difference in categorization and annotation across different domains to which these datasets belong. In order to improve the accuracy, the authors developed a universal taxonomy through merging and splitting classes from each of the component datasets and finally annotated and relabelled the dataset [4].

Network Architecture: The experiments in the paper use an HRNetV2-W48 [7] architecture as the backbone, where W48 describes the high-res convolution width. They use an SGD optimizer with momentum 0.9 and 10^{-4} weight decay.

Most models typically have a low-res subnetwork with high to low convolutions in series and then form a high-res recovery subnetwork by connecting low to high convolutions in series. HRNet or high-resolution networks [7] maintain a high resolution throughout the computation by starting with a high res convolution stream and adding high to low-convolution streams in parallel and adding multi-resolution fusions to exchange info across streams.

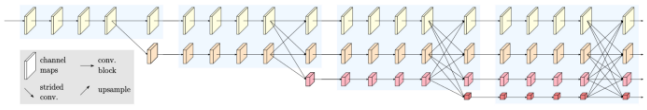

Fig. 7. Architecture of a HRNet

HRNet's high res representations result in higher precision since high res is maintained throughout and not recovered from low resolution. Further, the fusion between high and low res streams helps both streams boost each other and become semantically stronger.

Training: Image sizes vary across datasets so they resized all images to 713px x 713px while the shorter side is 1080px and preserving aspect ratios. Prior to this resizing, they upsampled images without at least 1000p resolution. They chose a minibatch size of 35 and equally split the minibatch across each training dataset. The model was trained until a million crops from each dataset were seen.

Results: Models have good accuracy when tested on the same dataset, but MSeg models match or outperform the individually trained models across almost all datasets. Models

trained on MSeg achieve better accuracy than the baseline, and relabeling the data further boosts the accuracy. IoU class metric on the KITTI website for MSeg is listed as 62.64.

Train/Test	COCO	ADE20K	Mapillary	IDD	BDD	Cityscapes	SUN	*h. mean*
COCO	**52.7**	19.1	28.4	31.1	44.9	46.9	29.6	32.4
ADE20K	14.6	**45.6**	24.2	26.8	40.7	44.3	36.0	28.7
Mapillary	7.0	6.2	**53.0**	50.6	59.3	71.9	0.3	1.7
IDD	3.2	3.0	24.6	**64.9**	42.4	48.0	0.4	2.3
BDD	3.8	4.2	23.2	32.3	63.4	58.1	0.3	1.6
Cityscapes	3.4	3.1	22.1	30.1	44.1	77.5	0.2	1.2
SUN RGBD	3.4	7.0	1.1	1.0	2.2	2.6	43.0	2.1
MSeg-w/o relabeling	50.4	**45.4**	**53.1**	**65.1**	66.5	**79.5**	**49.9**	**56.6**
MSeg	50.7	**45.7**	**53.1**	**65.3**	**68.5**	**80.4**	**50.3**	**57.1**

Fig. 8. Comparision of models trained on MSeg compared to models trained on other datasets

D. *VideoProp-LabelRelax*

In this work, video prediction models are utilized to efficiently create more training samples (image-label pairs) as shown in Figure x. Given a sequence of video frames having labels for only a subset of the frames in the sequence, we exploit the prediction models' ability to predict future frames in order to also predict future labels (new labels for unlabelled frames). [1]

The model is leveraged via Joint image-label Propagation (JP): Creation of a new training sample by pairing a propagated label with the corresponding propagated image. This approach is separately applied for multiple future steps to scale up the training data set. The approach for training data synthesis from sparsely annotated video frame sequences is when given an input video, $\boldsymbol{I} \in \boldsymbol{R}^{n \times W \times H}$ and semantic labels $\boldsymbol{L} \in \boldsymbol{R}^{m \times W \times H}$, where $\boldsymbol{m} \leq \boldsymbol{n}$, we synthesize $\boldsymbol{k} \times \boldsymbol{m}$ new training samples (image-label pairs) using video prediction models, where k is the length of propagation applied to each input image label pair $(\boldsymbol{I_i}, \boldsymbol{L_i})$.

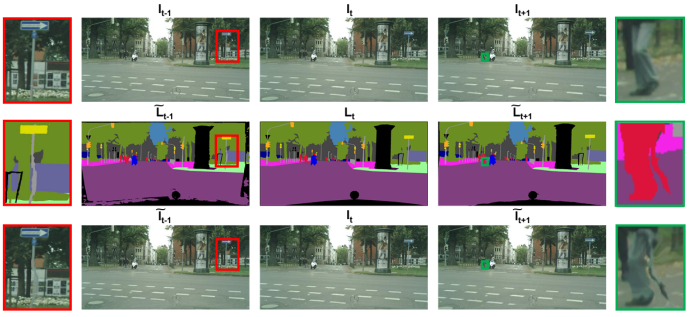

Fig. 9. Joint Image-Label Propagation

Training: For semantic segmentation, an SGD optimizer is used and a polynomial learning rate policy was applied, where the initial learning rate is multiplied by (1 - epoch max epoch)power. The initial learning rate is set to 0.002 and power to 1.0. Momentum and weight decay are set to 0.9 and 0.0001 respectively. Synchronized batch normalization was used (batch statistics synchronized across each GPU) with a batch size of 16 distributed over 8 V100 GPUs. The number of training epochs is set to 180 for Cityscapes, 120 for Camvid, and 90 for KITTI. The network architecture is based on DeepLabV3Plus with an output stride equal to 8. For the network backbone, ResNeXt50 was used for the ablation studies and WideResNet38 for the final test submissions.

Experiments: The proposed model was evaluated on three widely adopted semantic segmentation datasets, including Cityscapes, CamVid, and KITTI. For all three datasets, the standard mean Intersection over Union (mIoU) metric was used to report segmentation accuracy.

III. Experiments & Results

After analyzing each model, we decided to first reproduce the results to understand each model in detail. As pointed out in the introduction, our review came up with the following naive classification for the chosen 4 models. Common architecture backbone and numerically vast data sets. For the sake of tweaking the models and understanding the effects we went ahead and experimented with SegStereo and the following are the reasons why.

1) The difference between SGDepth and SegStereo IoU was too great for any reasonable improvements which can be made given the time frame.
2) Mseg and VideoProp have huge training data sets (in the order of millions of images). We were bound by time frame mostly. The reference time frame for training was given as 2-3 weeks.
3) Hence SegStereo was found to be the one that could be modified and studied in the scope of this project.

We trained the model with 4200 (2x the baseline) images from KITTI and CITYScapes datasets on a laptop with AMD Ryzen 5 Processor and NVIDIA GeForce RTX 3050 GPU (4GB GDDR6) to get the training results.

As reviewed we replaced the softmax loss function with a cross-entropy to see its effect on pixel classification. Doing so increased the disparity prediction error to 2.5 and caused black spots in prediction images.

Another modification was to introduce L2 regularization to the disparity prediction loss function instead of the proposed L1 regularized error improving the IoU to 61.23 from 59.1

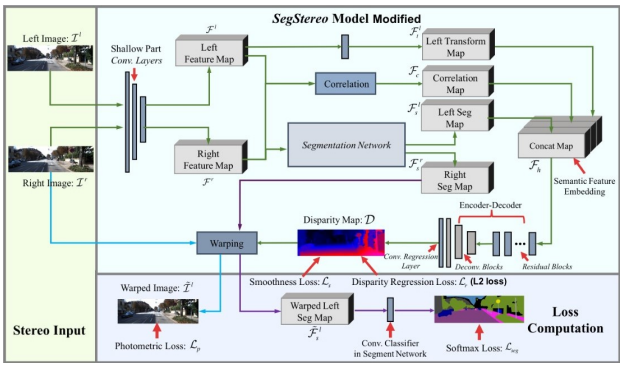

Fig. 10. Modified SegStereo architecture

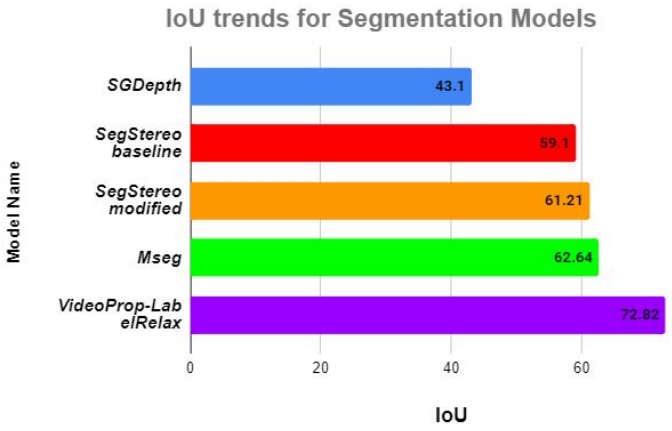

Fig. 11. Comparison of results

Based on papers and our experiments, we tried reasoning why each model's IoU score improved for each model. We found that better results are a combination of changes in factors such as the loss function and the choice of the optimizer and network architecture. While the first was demonstrated in our project experiment, it is left to be explored what kind of effect would be had if the same experiment were to be carried out on architectures such as DeepLab and WRP, which are placed on the top in the KITTI leaderboard currently. Our analysis of these diverse papers and models inspires future research questions that aim to improve the uncertainty reasoning associated with data augmentation through soft label relaxation through learned kernels, and also delve into unsolved problems related to combining propagated labels with manual annotations or coarse labels.

IV. Conclusion

Since SegStereo and SGDepth have similar network architecture, the improved IoU is attributed to the new semantic loss function which helped improve disparity loss too. We think using better datasets for MSeg and VideoProp resulted in better IoU scores. DeepLabV3Plus especially performs better due to atrous convolution which performs better than traditional deconvolutional layers. This removes the downsampling operator from the last layer of the DCNN and instead up-samples the filters in subsequent convolutional layers resulting in feature maps computed at a higher sampling rate

Our experiments show that datasets yield better results due to the model being trained on a larger set of features and having better feature vectors. We plan on carrying this work forward by exploring complex models (WRP and VideoProp) on more elaborate datasets. Modifying and developing the architecture, optimizer, and loss function lead to better performance on computer vision tasks.

Future possibilities include using the strongest statistical elements from state-of-the-art semantic models in the domain of autonomous driving to develop new, more accurate vision models, that have the potential of being published and placed on the KITTI leaderboard to make progress in the domain of autonomy.

References

[1] B. Yi Zhu, Karan Sapra, Fitsum A. Reda, Kevin J. Shih, Shawn Newsam, Andrew Tao, Bryan Catanzaro, 'Improving Semantic Segmentation via Video Propagation and Label Relaxation'. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[2] G. Yang, H. Zhao, J. Shi, Z. Deng and J. Jia, 'SegStereo: Exploiting Semantic Information for Disparity Estimation'. European Conference on Computer Vision (ECCV), 2018.

[3] M. Klingner, J. Termöhlen, J. Mikolajczyk and T. Fingscheidt, 'SegDepth: Self-Supervised Monocular Depth Estimation: Solving the Dynamic Object Problem by Semantic Guidance'. European Conference on Computer Vision (ECCV), 2020.

[4] J. Lambert, Z. Liu, O. Sener, J. Hays, and V. Koltun, 'MSeg: A Composite Dataset for Multi-domain Semantic Segmentation'. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[5] H. Alhaija, S. Mustikovela, L. Mescheder, A. Geiger, and C. Rother, 'Augmented Reality Meets Computer Vision: Efficient Data Generation for Urban Driving Scenes', International Journal of Computer Vision (IJCV), 2018.

[6] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 'ImageNet Large Scale Visual Recognition Challenge', International Journal of Computer Vision (IJCV), 2015.

[7] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, 'Deep High-Resolution Representation Learning for Visual Recognition', IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[8] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B., 'The Cityscapes Dataset for Semantic Urban Scene Understanding'. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.