TransUNet for Cross-Domain Semantic Segmentation of Urban Scenery

Wei Yuen Teh
Wau Labs Sdn Bhd
Subang Jaya, Malaysia
tehweiyuen@waulabs.com

Ian K. T. Tan
School of Mathematical and Computer Sciences
Heriot-Watt University Malaysia
Putrajaya, Malaysia
i.tan@hw.ac.uk

***Abstract*—TransUNet is a hybrid architecture that combines a transformer-based encoder with a CNN-based UNet. Originally introduced for semantic segmentation of medical images, we show in our work that TransUNet can be successfully applied to urban scenery datasets commonly used for developing autonomous driving systems. We also explore the performance characteristics of training on multi-domain data from the real world and a simulator, and show that using simulated images to augment a live dataset can improve segmentation performance. Code will be made available at https://github.com/weiyuen.**

***Index Terms*—Semantic Segmentation, Domain Adaptation, Vision Transformer, Urban Scenery, Autonomous Driving**

I. Introduction

For decades, autonomous driving has been a prominent goal for the AI community. While this goal seemed unattainable not too long ago, the deep learning breakthrough over the last decade has brought us closer than ever and led to renewed optimism in the field. While multiple challenges still remain to be solved to achieve fully autonomous vehicles, deep learning has revolutionized the field of computer vision, and resulted in the creation of models capable of performing tasks such as semantic segmentation at performance levels that would have been unthinkable of prior to deep learning.

However, as inductive systems, deep learning models require large amounts of data to perform well. This is especially true for autonomous driving models, where mistakes can be lethal, and as a result robust performance is expected even on out of distribution data. This means building datasets that are sufficiently large to include long-tail occurrences, which can be costly.

An emerging solution to this problem is to use simulated data to augment live training data. As part of the Smoky Mountain Data Challenge 2021 [1], we were provided with such a dataset, consisting of a mixture of live images sourced from the Cityscapes dataset [2] and simulated images obtained from the CARLA [3] driving simulator.

II. Related Work

A. Transformers for Semantic Segmentation

Vision Transformers (ViTs) were first introduced in 2020 by Dosovitskiy et al. [4]. Their work allowed for the transformer architecture [5] to work on visual inputs by representing input images as a sequence of patches. ViTs have since been used on a variety of other computer vision tasks, a field where Convolutional Neural Networks (CNNs) have traditionally dominated.

A variety of transformer-based semantic segmentation architectures have been proposed in 2021. Zheng et al. [6] proposed SETR (Segmentation Transformer), a ViT-like encoder-decoder architecture for semantic segmentation. SETR did not utilize any convolutions or downsampling, and achieved competitive results with leading CNN-based architectures. Variations of pure transformer segmentation networks have also been proposed, such as the Segmenter [7] and TrSeg [8].

Next, Chen et al. [9] proposed TransUNet, a hybrid architecture that combines the CNN-based U-Net with a ViT-based encoder. The classic U-Net [10] architecture contains skip connections between the encoder and decoder segments of the network, allowing for the recovery of fine details during reconstruction. However, while CNNs perform well at detecting local features, they are unable to reliably encode long-range dependencies. The addition of a transformer encoder alleviates this issue due to its use of self-attention. The authors showed that a transformer-UNet hybrid was able to outperform pure transformer-based architectures due to the U-Net's ability to recover low-level details. Our work in this paper will be based on the TransUNet architecture, and while the model was originally developed for medical image segmentation, we will show that it performs strongly on urban scenery datasets as well.

Finally, the Wide-Context Network (WiCoNet) in [11] builds on TransUNet by incorporating multiple views of the input image. Their work involved high-resolution remote sensing images, and the model received as input a downsampled global view of the image, as well as a cropped local view. While images from urban scenery datasets are often significantly lower in resolution, a potential direction for future work would be to determine if the method provides any benefits in this area.

III. Methodology

A. Dataset

The Smoky Mountains Data Challenge 2021 Challenge 3 [1] dataset consists of 5600 images, along with a segmentation map for each image. Simulated images from the CARLA simulator make up 4900 images in the dataset, while the

Function	Argument
Fliplr	0.5
MultiplyBrightness	(0.5, 1.5)
MultiplySaturation	(0.5, 1.5)
ChangeColorTemperature	(3200, 22000)
MultiplyHue	(0.7, 1.3)
LogContrast	(0.7, 1.3)
Sharpen	(0.0, 0.3)
GaussianBlur	(0.0, 0.3)

TABLE I: *imgaug* functions along with the arguments used.

remaining 700 images come from the Cityscapes dataset (representing real world data). The images generated from CARLA span 7 different categories of weather and lighting conditions. Note that this results in a 7:1 ratio of simulated to live data in the dataset.

Since the number of classes and label values differed between the provided segmentation maps for CARLA and Cityscape images, we first had to map all label values to a standardized format provided by the challenge sponsors, which resulted in a total of 15 classes.

To act as a regularizer and to improve model performance, we perform image augmentation during training, with parameters shown in Table I. All augmentations were implemented using the *imgaug* library. We also resized all images to 224x224 in order to suit the pre-trained model's requirements.

B. TransUNet

In their original paper, Chen et al. [9] test multiple variants of their TransUNet architecture, varying parameters such as the number of skip connections between the encoder and decoder, and the patch size of the transformer encoder.

For our experiments, we use the TransUNet variant that achieved the best results in [9]. This variant uses a ResNet-50 [12] CNN encoder with 3 skip connections, and a vision transformer with a patch size of 16. Both the CNN and vision transformer are pre-trained on ImageNet21k [13].

Our experiments utilize code made available by the original authors on GitHub [14]. While we have adapted and streamlined their code in various places to suit our dataset and studies, the semantics remain identical.

We use SGD as the optimizer and the average of the categorical cross-entropy loss and DICE loss [15] as our loss function. Unless otherwise specified, all models were trained for 200 epochs at a learning rate of 0.015 using a batch size of 16.

C. Evaluation Metrics

Throughout our experiments, we use the weighted mean IoU (wMIoU) as the primary metric to evaluate our models. This metric is commonly used to evaluate semantic segmentation models and represents the mean of the intersection over union across all classes, weighted by the size of the class. However, we also include the mean IoU (mIoU), as well as the individual IoUs for each class in our results to help paint a more representative picture of each model's performance.

IV. Experiments & Results

To study the effects of training on different combinations of simulated and live data on model performance, we partition our training/validation/test sets in three different ways as shown in Table II, and discuss the results of each in the subsections that follow.

A. Training on Simulated & Live Images

Table III shows the quantitative results obtained from training on a mixture of simulated and live images. We show the mIoU and wMIoU across the whole test set (Table IIIa), as well as the disaggregated results for the simulated and live images in the test set (Tables IIIb & IIIc). Unfortunately, due to the unique nature and size of the dataset provided for this challenge, direct comparison of quantitative results with existing methods is not possible.

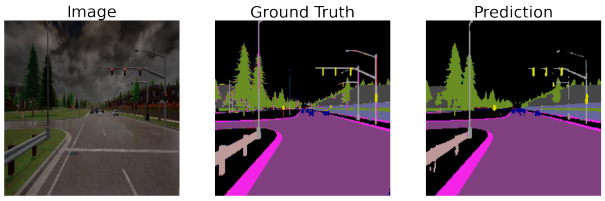

Fig. 1: Output sample of a simulated image (0.908 wMIoU).

Fig. 2: Output sample of a live image (0.750 wMIoU).

At first glance, the quantitative results show vastly better performance on simulated images (Table IIIb) relative to live images (Table IIIc). However, while it is clear that the model performs better on simulated images, we believe that the practical results are closer than the quantitative figures indicate, for reasons we will hence describe.

Qualitatively, we note that the ground truth labels on the live images are often more inconsistent than their simulated counterparts, likely due to the increased ambiguity of objects in live scenes. There are often multiple valid labels for a given object, and we note that quite often, the model's missed predictions are arguably valid.

To help illustrate this, Figures 1 & 2 show a simulated and live sample from the test set respectively. These images were chosen as their wMIoU values are close to the mean wMIoU values of their respective disaggregated test sets. Note that a significant cause of the live image's lower score is the 'misclassification' in the upper-right corner, as well as in the gap between the vegetation. The model classifies these regions as buildings (gray), whereas the ground truth labels them as part of the 'Other' class (black).

Section	Train (85%)	Validation (5%)	Test (10%)
IV-A	Sim + Live	Sim + Live	Sim + Live
IV-B	Sim	Sim	Live
IV-C	Live	Live	Live

TABLE II: Training, validation, and test set partitions.
(Sim=CARLA, Live=Cityscapes)

Class	Value	Class	Value	Class	Value
Building	0.872	Wall	0.778	Truck	0.042
Fence	0.692	Road	0.977	Bus	0.082
Pole	0.427	Traffic Light	0.408	Train	0.433
Sidewalk	0.873	Person	0.531	Bicycle	0.485
Vegetation	0.781	Car	0.845	Other	0.860
		mIoU:	**0.606**		
		wMIoU:	**0.884**		

(a) Results on full test set.

Class	Value	Class	Value	Class	Value
Building	0.897	Wall	0.815	Truck	0.002
Fence	0.731	Road	0.983	Bus	0.000
Pole	0.450	Traffic Light	0.473	Train	0.000
Sidewalk	0.906	Person	0.000	Bicycle	0.002
Vegetation	0.779	Car	0.841	Other	0.879
		mIoU:	**0.647**		
		wMIoU:	**0.900**		

(b) Disaggregated results for simulated images.

Class	Value	Class	Value	Class	Value
Building	0.770	Wall	0.111	Truck	0.051
Fence	0.267	Road	0.916	Bus	0.424
Pole	0.207	Traffic Light	0.319	Train	0.785
Sidewalk	0.622	Person	0.508	Bicycle	0.422
Vegetation	0.774	Car	0.846	Other	0.604
		mIoU	**0.508**		
		wMIoU	**0.756**		

(c) Disaggregated results for live images.

TABLE III: Results from training on simulated and live images.

A close inspection of the image indicates that the upper-right region is a pedestrian bridge, and that the gap between the vegetation is a wall of some form, showing that the model's predictions are very sensible (and arguably more valid than ground truth). Empirically, misclassifications like these happen much more often in live images than in simulated ones (see also Figure 4 from section IV-C), which likely exaggerates the gap seen in the quantitative results.

As a side note, since the challenge's withheld test set includes a combination of both simulated and live images, this subsection's training method is used for our submission, as we found it to produce the best performing model overall.

B. Training on Simulated Images Only

To determine the extent of the model's ability to generalize from the simulated domain to the live domain, we next train the model on simulated images only, and perform evaluation on live images. Results are shown in Table IV, along with results from a sample image in Figure 3.

These results show that some learning and generalization across domains does occur when training on purely simulated data, particularly in the larger classes such as 'Building', 'Vegetation', and 'Road', and that the high-level structure of scenes is usually captured well. However, the significantly worse quantitative and qualitative performance overall shows

Fig. 3: Output sample of a live image (0.471 wMIoU).

that the addition of even a small number of live images in the training set goes a long way to improving performance.

C. Training on Live Images Only

Finally, to study if the addition of simulated images affects performance on live images, we train and test a model on only the 700 live images in the provided dataset. Results are shown in Table V, with a sample output shown in Figure 4.

By comparing the results here to the disaggregated results of live images in Section IV-A, we see that the removal of simulated images from the training set results in a minor reduction to wMIoU on live images (0.756 to 0.748). mIoU sees a more significant reduction (0.508 to 0.430), largely driven by large decreases to the IoUs of the bus and train classes. However, these two classes constitute a miniscule

Class	Value	Class	Value	Class	Value
Building	0.549	Wall	0.099	Truck	0.000
Fence	0.022	Road	0.730	Bus	0.000
Pole	0.074	Traffic Light	0.035	Train	0.000
Sidewalk	0.397	Person	0.000	Bicycle	0.000
Vegetation	0.583	Car	0.282	Other	0.256
		mIoU:	**0.233**		
		wMIoU:	**0.500**		

TABLE IV: Results from training on simulated images and testing on live images.

Class	Value	Class	Value	Class	Value
Building	0.74	Wall	0.323	Truck	0.005
Fence	0.095	Road	0.918	Bus	0.002
Pole	0.163	Traffic Light	0.239	Train	0.278
Sidewalk	0.648	Person	0.387	Bicycle	0.441
Vegetation	0.820	Car	0.801	Other	0.588
		mIoU:	**0.430**		
		wMIoU:	**0.748**		

TABLE V: Results from training and testing on live images.

Fig. 4: Output sample of a live image (0.660 wMIoU).

portion of the overall dataset, meaning that the reduction in mIoU is not statistically significant, and that the wMIoU is more representative of the difference in performance.

Our results show that the use of simulated images to augment live images offers slight benefits to performance for this particular dataset. We posit that a simple way to increase these benefits would be to increase the alignment between ground truth labels for simulated and live images. In this instance, all vehicles in the simulated images were labelled as 'Car', whereas they were also subclassed into 'Truck' and 'Bus' in the live images. Apart from improving generalization between domains, we hypothesize that improved label alignment will also partially alleviate the ambiguity issues discussed in Section IV-A.

V. Conclusion

Through our experiments, we have shown that the TransUNet architecture can be successfully applied on an urban scenery dataset to perform semantic segmentation. In addition, we have shown that while some learning and generalization from the simulated to real domain does occur (Section IV-B), the best results are obtained from training on a combination of simulated and real data (Section IV-A), which validates the use of simulated images for training when limited live images are available. We have also discussed issues related to label ambiguity, and suggested potential directions for future work, namely pertaining to label consistency/alignment.

References

[1] "Smoky mountains data challenge: Challenges 2021." [Online]. Available: https://smc-datachallenge.ornl.gov/data-challenges-2021/

[2] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[3] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," 2017.

[4] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2021.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. [Online]. Available: http://arxiv.org/abs/1706.03762

[6] S. Zheng, J. Lu, H. Zhao, X. Zhu, Z. Luo, Y. Wang, Y. Fu, J. Feng, T. Xiang, P. H. Torr, and L. Zhang, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 6881–6890.

[7] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, "Segmenter: Transformer for semantic segmentation," 2021.

[8] Y. Jin, D. Han, and H. Ko, "Trseg: Transformer for semantic segmentation," *Pattern Recognition Letters*, vol. 148, p. 29–35, 2021.

[9] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, and Y. Zhou, "Transunet: Transformers make strong encoders for medical image segmentation," 2021.

[10] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[11] L. Ding, D. Lin, S. Lin, J. Zhang, X. Cui, Y. Wang, H. Tang, and L. Bruzzone, "Looking outside the window: Wide-context transformer for the semantic segmentation of high-resolution remote sensing images," 2021.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[13] T. Ridnik, E. B. Baruch, A. Noy, and L. Zelnik-Manor, "Imagenet-21k pretraining for the masses," *CoRR*, vol. abs/2104.10972, 2021. [Online]. Available: https://arxiv.org/abs/2104.10972

[14] Beckschen, "Beckschen/transunet: This repository includes the official project of transunet, presented in our paper: Transunet: Transformers make strong encoders for medical image segmentation." [Online]. Available: https://github.com/Beckschen/TransUNet

[15] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations," *CoRR*, vol. abs/1707.03237, 2017. [Online]. Available: http://arxiv.org/abs/1707.03237