Supplementary Material

1 Discussions

1.1 Why does SynthCP work better?

Current approaches, such as MSP, TCP and MCDropout, mainly focus on improving failure detection with self-estimated statistics. However, deep networks tend to yield high confidence prediction [3, 4], thus self-estimated statistics are not trustable. The approaches that leverage extra data [5] or alternating training strategies [3] can alleviate this problem. We propose to solve this problem from another prospective - **analyzing the performance of deep discriminative models by generative models**, the reverse procedure that models the conditional data distribution prior $P(x|y)$. SynthCP models $P(x|y)$ with a cGAN, which is proved to be beneficial to both failure and OOD detection of segmentation models.

1.2 Extra computational cost.

There are two steps that requires extra computation besides the original segmentation network (M, latency T) in SynthCP - GAN reconstruction (G) and the comparison function computation for failure detection/anomaly segmentation. Since M and G are mutually inverse procedures, the inference time should be in the same magnitude. Compared to M or G, the inference time of the failure detection network and distance computation for anomalies are insignificant. So the overall extra computational cost for our framework is the T. Compared to other approaches, MSP based approaches [4, 5] are the most efficient. VAE alarm [7] and the AE-based approach [1] both need a separate network, basically have the same latency as ours. Dropout based approaches [2, 6] require multiple sampling of a segmentation network, which typically consumes more than time of $10T$.

1.3 Detecting failures of unseen models

We provide additional results for testing the generalizability of our failure detection system. We train the model on Deeplab-v2 predictions and test on FCN-8 predictions. Then we do the opposite. As shown in Table 1, our results in comparable to Table 1 in the main paper. This illustrate our failure detection framework is generalizable, making it possible to build a general failure detection system without additional training on an unseen segmentation algorithm.

Table 1. Results for cross-approach direct testing. The first row means we train on ”Deeplab-v2” results and detect failures on ”FCN-8” results and the second row means the opposite. The results show that our approach has good generalizability and illustrate our potential to build a general failure detection system without additional training on a separate segmentation algorithm.

		image-level			
train	test	MAE↓	STD↓	P.C.↑	S.C.↑
Deeplab-v2	FCN-8	12.91	11.41	65.34	63.60
FCN-8	Deeplab-v2	14.73	13.05	60.79	61.01

		pixel-level			
train	test	AP-Err↑	AP-Suc↑	AUC↑	FPR95↓
Deeplab-v2	FCN-8	53.12	99.10	92.29	23.51
FCN-8	Deeplab-v2	49.34	99.32	92.85	22.06

1.4 Adding image style encoder

Since the generator G does not guarantee the same style between x and $\hat{x}$ which increases the difficulty of the comparison module, we try to mitigate the effect by using an image encoder version of SPADE [8]. We hope the encoder can encode the style and generate images condition on segmentation map with the same style. However, the performance is not satisfactory (AUPR-Error experiences a subtle drop from 55.53 to 55.22). We hypothesize that the style encoder may also encode content (semantic) information and “cheat” to synthesize image without the segmentation mask, thus make the generator “less conditional”.

1.5 GAN type

We assume a stronger GAN would yield better synthesis, and thus choose the state-of-the-art SPADE model [8] for all the main experiments. We also tried a weaker generator - pix2pixHD [9]. It turns out that the synthesis quality is far from satisfactory when the generator takes the prediction $\hat{y}$ as input. Some examples are shown in Fig 1. Under the same settings (FCN8 and pixel-level failure detection), the AUPR of pix2pixHD model is only 51.31, which is close to the baseline “direct prediction” (AUPR: 52.17). We thus conclude that a stronger generator benefits our failure detection scheme.

2 More visualization on Cityscapes and StreetHazards

We show more visualizations on Cityscapes failure detection (Fig 2) and StreetHazards anomaly segmentation (Fig 3).

3 Per-class IoU

We provide additional results for failure detection on the Citysapces dataset. In the main paper, we only provide results for the average of each class. Here,

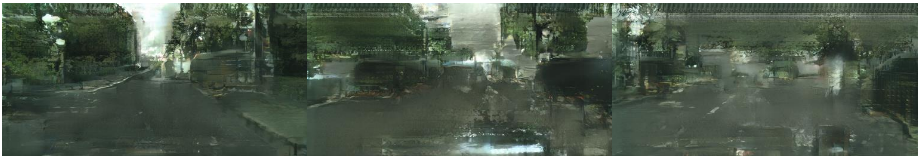

Fig. 1. Some examples of the synthesized images using Pix2pixHD [9]. These synthesized images are not informative enough for failure detection, if using our proposed comparison module.

detailed per-class results for all the four metrics (MAE, STD, PC and SC) are shown in Table 2, 3, 4, 5.

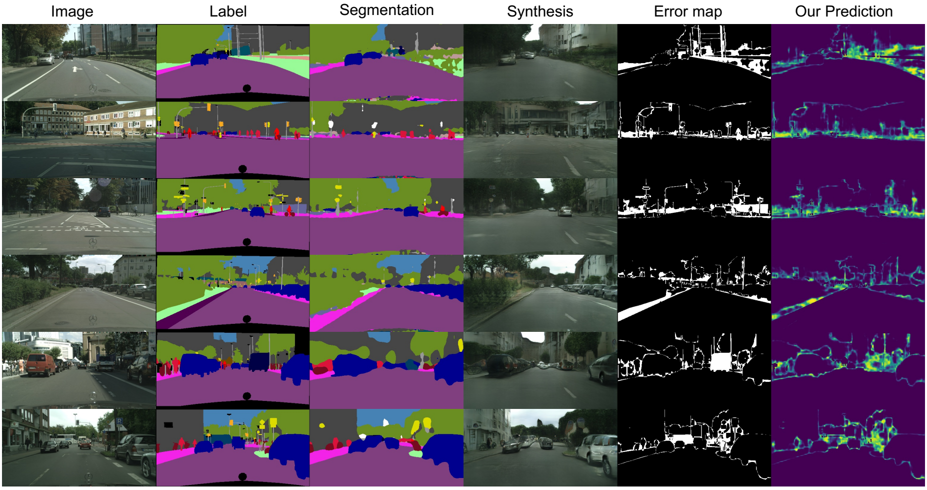

Fig. 2. Visualization on the Cityscapes dataset for pixel-level error map prediction . For each example from left to right (top), we show the original image, ground-truth label map, segmentation prediction, synthesized image conditioned on the segmentation prediction, (ground-truth) errors in the segmentation prediction and our pixel-level error prediction.

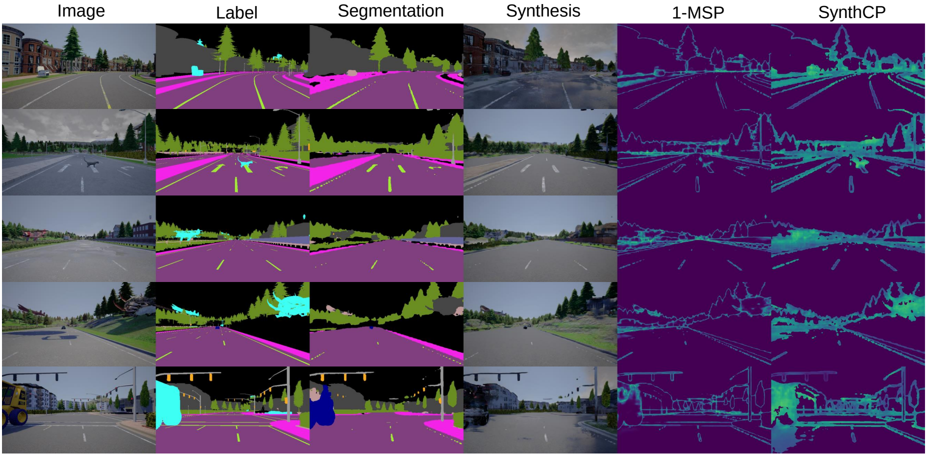

Fig. 3. Visualizations on the StreetHazards dataset. For each example, from left to right, we show the original image, ground-truth label map, segmentation prediction, synthesized image conditioned on segmentation prediction, MSP anomaly segmentation prediction and our anomaly segmentation prediction.

Table 2. Per-class performance of image-level IoU prediction in MAE.

	road	sidewalk	building	wall	fence	pole	t-light	t-sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bicycle	mean
									FCN-8											
MCDropout [2]	3.27	19.97	12.39	17.84	18.97	10.49	14.85	20.37	10.90	25.51	15.92	20.65	15.75	17.22	14.99	31.89	22.05	12.27	22.95	17.28
VAE alarm [7]	3.07	18.96	11.58	18.55	19.19	10.36	15.49	20.29	9.15	24.73	17.35	17.47	16.34	16.11	9.40	33.21	13.93	12.01	22.17	16.28
Direct Prediction	3.19	13.88	8.00	14.74	14.82	7.76	11.14	14.78	8.08	19.36	10.99	15.65	11.46	13.29	15.45	24.97	17.79	9.86	16.56	13.25
Ours separate	2.50	14.51	7.60	12.65	12.39	6.93	10.09	12.60	7.66	17.74	10.21	13.41	8.98	12.92	14.09	18.45	16.76	5.82	14.76	11.58
Ours joint	4.54	14.14	6.67	12.35	13.46	7.03	9.90	14.17	7.40	16.71	11.68	14.15	14.32	14.92	19.98	18.78	15.09	10.84	15.02	12.69
									Deeplab											
MCDropout [2]	3.29	18.95	11.37	21.22	20.39	12.74	18.23	21.02	9.63	26.54	13.95	22.98	19.55	15.78	26.50	35.86	25.85	19.30	23.82	19.31
VAE alarm [7]	2.94	19.17	11.94	18.54	19.13	10.34	15.49	20.28	8.77	24.83	17.40	17.68	16.63	14.90	17.18	33.17	16.61	11.52	22.27	16.78
Direct Prediction	2.58	14.56	9.74	18.64	16.85	9.51	14.67	15.97	6.86	20.72	9.10	15.48	14.99	11.34	21.51	23.30	17.66	14.19	16.86	14.45
Ours separate	3.76	16.52	7.00	14.74	11.31	9.69	10.71	15.32	7.29	19.21	8.99	15.22	11.49	12.51	19.62	23.73	16.11	17.17	18.05	13.60
Ours joint	2.41	13.36	6.75	17.83	14.98	9.46	11.49	14.89	9.03	19.48	8.92	15.28	12.62	12.71	19.77	21.24	19.51	14.73	15.47	13.68

Table 3. Per-class performance of image-level IoU prediction in STD.

	road	sidewalk	building	wall	fence	pole	t-light	t-sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bicycle	mean
FCN-8																				
MCDropout [2]	5.86	13.69	11.67	15.68	13.97	8.18	12.19	11.99	11.63	12.76	19.45	11.57	11.36	16.33	19.70	16.14	16.82	10.73	13.57	13.33
VAE alarm [7]	5.88	14.28	11.20	14.45	12.77	7.21	11.31	11.69	10.43	13.36	17.69	10.10	9.97	14.16	12.16	11.96	14.28	11.00	11.98	11.89
Direct Prediction	5.42	11.67	8.09	16.49	11.63	5.91	10.32	12.88	9.10	13.09	11.70	10.72	11.66	12.47	16.64	11.35	23.76	10.82	13.55	11.96
Ours separate	5.55	11.77	8.50	14.59	10.19	5.90	7.79	9.89	9.10	13.38	14.00	9.31	8.72	13.30	21.34	13.88	21.83	8.53	10.94	11.50
Ours joint	5.18	11.43	7.89	12.60	11.33	6.20	8.92	11.04	9.77	12.54	12.15	10.74	14.16	12.64	16.16	15.26	12.84	12.19	11.50	11.29
Deeplab																				
MCDropout [2]	5.61	14.27	11.27	15.20	13.20	9.52	11.29	13.02	8.68	14.38	17.57	12.18	10.11	15.48	19.73	17.23	12.63	9.78	13.21	12.86
VAE alarm [7]	5.77	14.27	11.81	14.57	12.82	7.22	11.28	11.71	9.98	13.42	17.70	10.56	9.82	12.89	15.66	12.14	17.79	10.86	11.82	12.22
Direct Prediction	5.19	11.52	9.79	13.29	10.86	8.57	10.05	11.91	7.35	14.08	13.40	11.26	12.18	12.11	16.96	17.97	18.32	14.56	12.43	12.20
Ours separate	7.10	12.61	8.61	12.79	11.07	8.46	8.41	11.31	6.66	13.93	13.41	10.67	11.17	14.91	24.41	14.74	15.81	13.08	14.89	12.32
Ours joint	5.44	11.58	8.07	11.30	14.05	8.49	8.25	11.82	7.28	13.35	13.48	11.24	11.92	11.49	14.50	16.59	17.09	12.73	11.70	11.60

Table 4. Per-class performance of image-level IoU prediction in P.C..

	road	sidewalk	building	wall	fence	pole	t-light	t-sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bicycle	mean
FCN-8																				
MCDropout [2]	7.50	2.62	6.45	4.02	-10.87	4.20	10.75	0.66	9.71	4.10	2.76	1.31	8.86	4.07	-3.94	12.87	-14.72	5.16	13.27	3.62
VAE alarm [7]	0.27	16.24	28.42	10.81	8.88	0.32	1.64	1.99	47.71	13.80	6.23	49.33	13.32	42.26	77.06	3.93	69.00	2.43	8.83	21.18
Direct Prediction	40.38	65.85	74.55	42.59	58.21	67.32	61.21	65.34	64.13	57.29	77.78	67.71	62.04	63.85	35.95	74.65	19.62	45.67	64.32	58.34
Ours separate	40.35	63.47	73.64	66.37	72.86	69.88	77.81	73.36	66.33	65.07	76.70	72.32	78.49	62.75	16.76	78.03	22.84	80.16	70.80	64.63
Ours joint	42.77	66.31	79.29	66.83	65.16	73.69	72.93	65.38	64.73	67.88	76.63	68.78	54.81	57.91	5.94	73.12	71.11	42.28	72.38	62.52
Deeplab																				
MCDropout [2]	7.11	1.64	0.98	2.42	-5.26	1.50	2.79	4.69	5.50	8.73	2.30	6.89	1.16	-0.66	-9.24	18.24	28.20	0.72	8.79	4.55
VAE alarm [7]	21.43	11.48	4.00	7.55	10.00	5.27	4.23	1.66	53.97	10.56	1.91	46.03	0.48	55.40	26.63	1.10	46.92	23.52	8.32	17.92
Direct Prediction	40.90	64.37	71.48	51.06	61.12	64.61	60.40	60.96	63.63	57.11	71.00	68.66	55.33	65.49	53.96	72.34	59.54	48.86	67.04	60.94
Ours separate	17.51	56.29	71.92	66.51	76.32	63.28	77.23	65.44	66.78	63.21	71.50	71.77	71.29	55.09	51.29	68.95	72.58	36.01	64.67	62.51
Ours joint	34.53	66.57	76.31	61.09	55.27	66.64	75.87	66.57	60.43	63.86	73.46	69.24	63.60	65.26	66.30	74.53	54.14	52.16	71.16	64.05

Table 5. Per-class performance of image-level IoU prediction in S.C..

	road	sidewalk	building	wall	fence	pole	t-light	t-sign	vegetation	terrain	sky	person	rider	car	truck	bus	train	motorbike	bicycle	mean
FCN-8																				
MCDropout [2]	10.24	0.48	-0.64	8.21	-10.56	4.99	10.02	-0.33	11.36	1.21	7.04	1.88	11.85	5.47	1.54	11.89	12.35	11.35	15.14	5.97
VAE alarm [7]	-8.42	17.82	26.40	9.28	8.37	-0.73	-1.41	2.24	49.44	22.37	3.18	48.57	13.04	43.15	56.59	6.49	36.43	3.57	10.58	18.26
Direct Prediction	58.66	70.24	78.59	33.46	45.25	67.74	57.10	68.32	63.61	55.97	80.39	68.81	58.75	73.27	34.24	74.74	39.21	39.25	67.45	59.74
Ours separate	61.97	66.85	78.45	52.07	68.66	70.43	74.79	75.78	66.71	65.37	84.59	72.97	73.81	74.30	14.15	74.68	35.92	60.14	75.40	65.63
Ours joint	59.24	68.23	82.43	49.42	61.63	74.49	64.20	66.05	68.03	65.64	78.53	70.37	52.94	64.23	-2.42	67.42	66.96	32.79	73.11	61.23
Deeplab																				
MCDropout [2]	5.13	-3.46	-4.03	-13.06	-1.71	0.46	2.00	5.17	4.37	8.44	1.12	5.55	-2.72	0.70	-16.27	11.41	15.50	0.94	6.54	1.37
VAE alarm [7]	27.11	22.43	-0.88	5.96	10.08	5.88	2.57	2.49	58.39	18.30	1.71	45.26	-0.36	54.80	22.82	0.56	68.35	19.51	8.04	19.63
Direct Prediction	51.13	68.68	78.80	42.82	51.26	66.93	58.96	63.34	65.81	57.08	77.31	69.88	49.87	75.88	54.20	69.00	62.87	46.06	68.28	62.01
Ours separate	38.56	64.52	77.00	50.09	69.71	64.53	78.80	66.49	67.61	60.87	80.43	72.78	69.53	69.78	40.80	72.46	65.44	29.60	65.71	63.41
Ours joint	54.99	71.31	80.62	54.73	46.82	69.91	77.96	69.50	58.17	65.60	77.58	68.93	64.71	76.25	57.96	76.88	51.17	46.47	73.47	65.42

References

1. Baur, C., Wiestler, B., Albarqouni, S., Navab, N.: Deep autoencoding models for unsupervised anomaly segmentation in brain mr images. In: International MICCAI Brainlesion Workshop. pp. 161–169. Springer (2018)
2. Gal, Y., Ghahramani, Z.: Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: international conference on machine learning. pp. 1050–1059 (2016)
3. Hein, M., Andriushchenko, M., Bitterwolf, J.: Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 41–50 (2019)
4. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. ICLR (2017)
5. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep anomaly detection with outlier exposure. ICLR (2019)
6. Kwon, Y., Won, J.H., Kim, B.J., Paik, M.C.: Uncertainty quantification using bayesian neural networks in classification: Application to biomedical image segmentation. Computational Statistics & Data Analysis **142**, 106816 (2020)
7. Liu, F., Xia, Y., Yang, D., Yuille, A.L., Xu, D.: An alarm system for segmentation algorithm based on shape model. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 10652–10661 (2019)
8. Park, T., Liu, M.Y., Wang, T.C., Zhu, J.Y.: Semantic image synthesis with spatially-adaptive normalization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2337–2346 (2019)
9. Wang, T.C., Liu, M.Y., Zhu, J.Y., Tao, A., Kautz, J., Catanzaro, B.: High-resolution image synthesis and semantic manipulation with conditional gans. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 8798–8807 (2018)