Supplementary Material for Synthetic Data for Defect Segmentation over Complex Metal Surfaces

Juraj Fulir[*] Lovro Bosnar[*†] Hans Hagen[†] Petra Gospodnetić[*]
[*]Fraunhofer ITWM, Fraunhofer-Platz 1, 67663 Kaiserslautern
{juraj.fulir,lovro.bosnar,petra.gospodnetić}@itwm.fraunhofer.de

[†]RPTU Kaiserslautern-Landau, Postfach 3049, 67663 Kaiserslautern
hagen@informatik.rptu.de

A. Comparison between RealClutch and SynthClutch

In this section we present the comparison of the two datasets using images with corresponding viewpoints (Fig. 1). The differences in surface reflectance is mainly due to per-texture domain randomization of synthetic object instances. To compensate for the illumination intensity difference, we pre-process synthetic images using the exposure value -1.

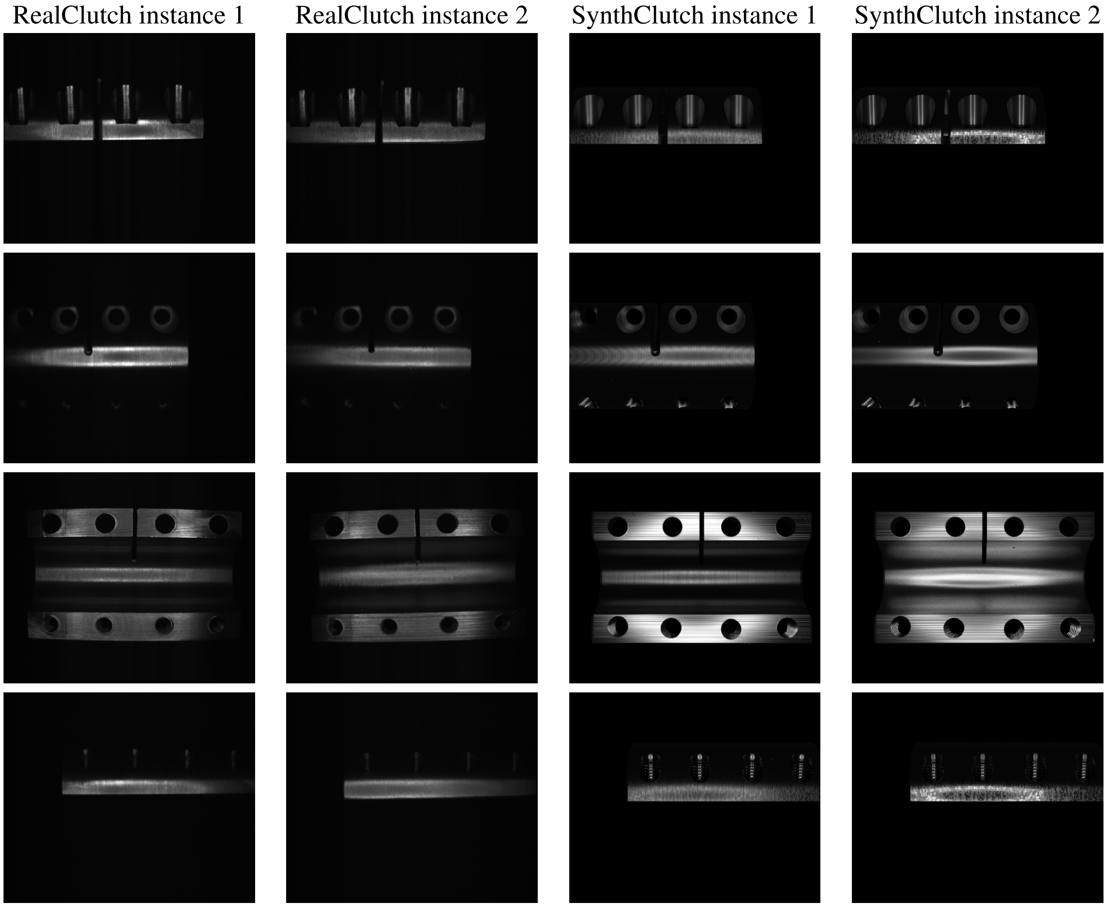

Figure 1. Example images of different object instances in the dual dataset, observed across corresponding viewpoints.

B. Hyperparameter analysis

B.1. Effectiveness of intensity biased random crops

In Tab. 1 we report the best results from grid search over different random cropping mechanisms for fully-convolutional network (FCN) [5], DeepLabV3 (DLv3) [2] amd U-Net [6] architectures. The reported training epochs are measured until validation loss convergence was detected, not the epoch of the best model, to measure the efficiency of optimization process. We were always guided by results on the validation set, however also report the method's effect on generalization to the test set.

Cropping mechanism	FCN			DLv3			U-Net		
	$F1_V$ [%]	$F1_T$ [%]	E	$F1_V$ [%]	$F1_T$ [%]	E	$F1_V$ [%]	$F1_T$ [%]	E
Random (uniform)	49.4	25.5	275	0.1	0.0	170	46.2	26.6	365
Intensity bias (5)	44.5	23.5	305	2.4	0.1	200	50.2	27.1	375
Intensity bias (10)	54.5	28.4	325	54.4	25.8	450	55.5	31.3	375
Intensity bias (15)	53.3	31.1	440	50.4	25.8	385	54.1	33.7	465
Intensity bias (20)	56.8	30.4	385	56.9	27.1	340	55.2	33.7	315

Table 1. Validation F1 score ($F1_V$), test F1 score ($F1_T$) and training epochs (E) of models on RealClutch trained using different cropping methods.

In Tab. 1 we observe that after at the intensity threshold 10 the gains mostly level-out. Further increase in the threshold would reveal significantly less area of the object and was thus avoided. DeepLabV3 shows to be very sensitive to optimization hyperparameters and requires more costly hyperparameter search. However, we observed that the intensity bias after a certain threshold value increased training stability and final model performance.

B.2. Crop size

We consider training with crop sizes up to size 256 due to memory limits. In Tab. 2 we observed a consistent decrease in model performance as we decrease crop size. This is related to the defect appearance ambiguity in lower crop sizes, as the glints produced by defects appear similar to glints produced by correct geometry. Additional contextual information can help resolve this problem by helping to localize the image patch on the object and ignore glints from common geometrical features, most commonly seen from the screw threads as shown in Fig. 2.

Crop size (square)	FCN		DLv3		U-Net	
	$F1_V$ [%]	$F1_T$ [%]	$F1_V$ [%]	$F1_T$ [%]	$F1_V$ [%]	$F1_T$ [%]
256	54.5	28.4	54.4	25.8	55.5	31.3
192	48.1	28.8	11.0	8.4	45.1	27.6
128	1.8	0.0	1.3	0.0	42.7	20.8

Table 2. Validation F1 score ($F1_V$) and test F1 score ($F1_T$) of models on RealClutch trained using different crop sizes.

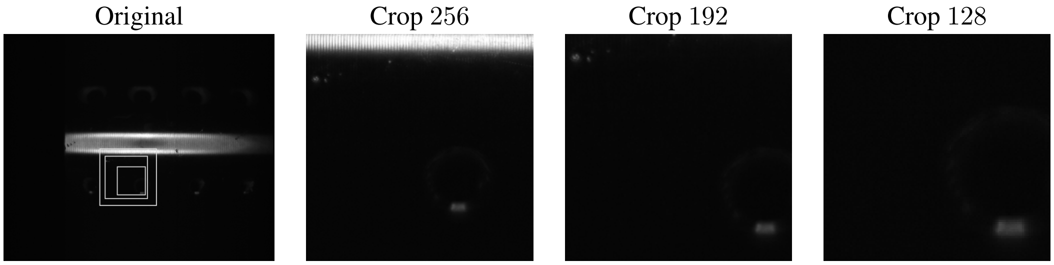

Figure 2. Smaller crop sizes lose the contextual information that is useful for disambiguation between the defects and geometrical features of the object. Notice the glint similarity between the screw thread (lower right) and defect on curved the surface (upper left).

C. Results of pre-training on source datasets

In this section we present the results of models reported in table 2 of the main paper, measured on the validation set of the respective source dataset it was trained on. Note that the same models were used for fine-tuning, thus the values in those experiments are same as these.

Source dataset	FCN			DLv3			U-Net		
	P [%]	R [%]	F1 [%]	P [%]	R [%]	F1 [%]	P [%]	R [%]	F1 [%]
RealClutch	55.3	53.7	54.5	53.1	55.8	54.4	58.4	52.9	55.5
DAGM [8]	81.6	73.9	77.6	82.4	74.2	78.1	78.8	69.2	73.7
KSDD2 [1]	71.3	67.5	69.4	70.3	69.7	70.1	69.9	69.0	69.5
Severstal Steel [7]	72.4	73.2	72.8	72.7	73.8	73.2	71.2	71.8	71.5
MTD [4]	71.0	61.7	66.0	65.6	56.5	60.7	65.8	57.5	61.4
CSEM-MISD [3]	35.1	24.3	28.7	41.0	23.5	29.9	42.6	22.0	29.0
SynthClutch	57.1	50.2	53.4	59.5	48.2	53.3	71.5	50.1	58.9
RealClutch (EX)	61.2	55.1	58.0	54.9	52.4	53.6	53.7	53.1	53.4
SynthClutch (EX)	57.7	50.8	54.0	62.2	51.8	56.5	69.0	50.4	58.3

Table 3. Precision (P), recall (R) and F1 score (F1) results of models trained on different source datasets with or without using fine-tuning (FT) and exposure stacking (EX), evaluated on the respective source validation split.

D. Predictions of best models

In this section we present the predictions and plots of metrics for reported models pre-trained on SynthClutch and fine-tuned on RealClutch. The following images represent images examples of different 3 viewpoints for different architectures. In color coding they depict the true-positives (green), false-positives (red) and false-negatives (blue).

The exposure stacking seems to be working inconsistently. In some cases the the poorly illuminated regions receive more complete predictions and in some difficult cases predictions are completely removed. This might be a problem with domain differences between the real and synthetic data, as in the real environment there exist inter-reflections between the object and the manipulator, adding intensity in unexpected regions, which gets amplified by increase in exposure. This can be fixed by randomizing an environment texture which will regularize the model to learn more robust features. However the exact type of environment texture could affect the visibility and this study is left as future work.

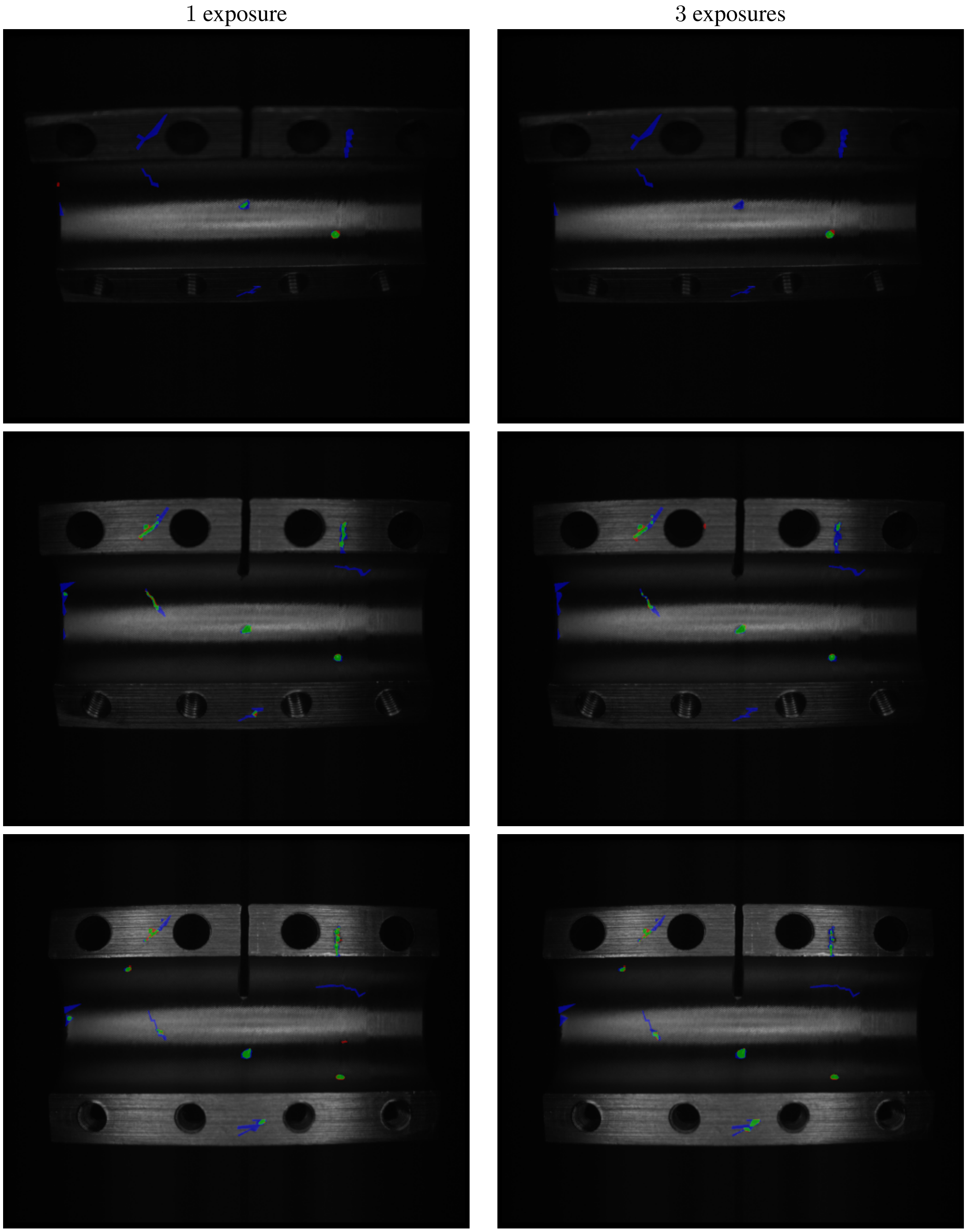

Figure 3. Example predictions of best FCN pre-trained on SynthClutch and fine-tunned on RealClutch.

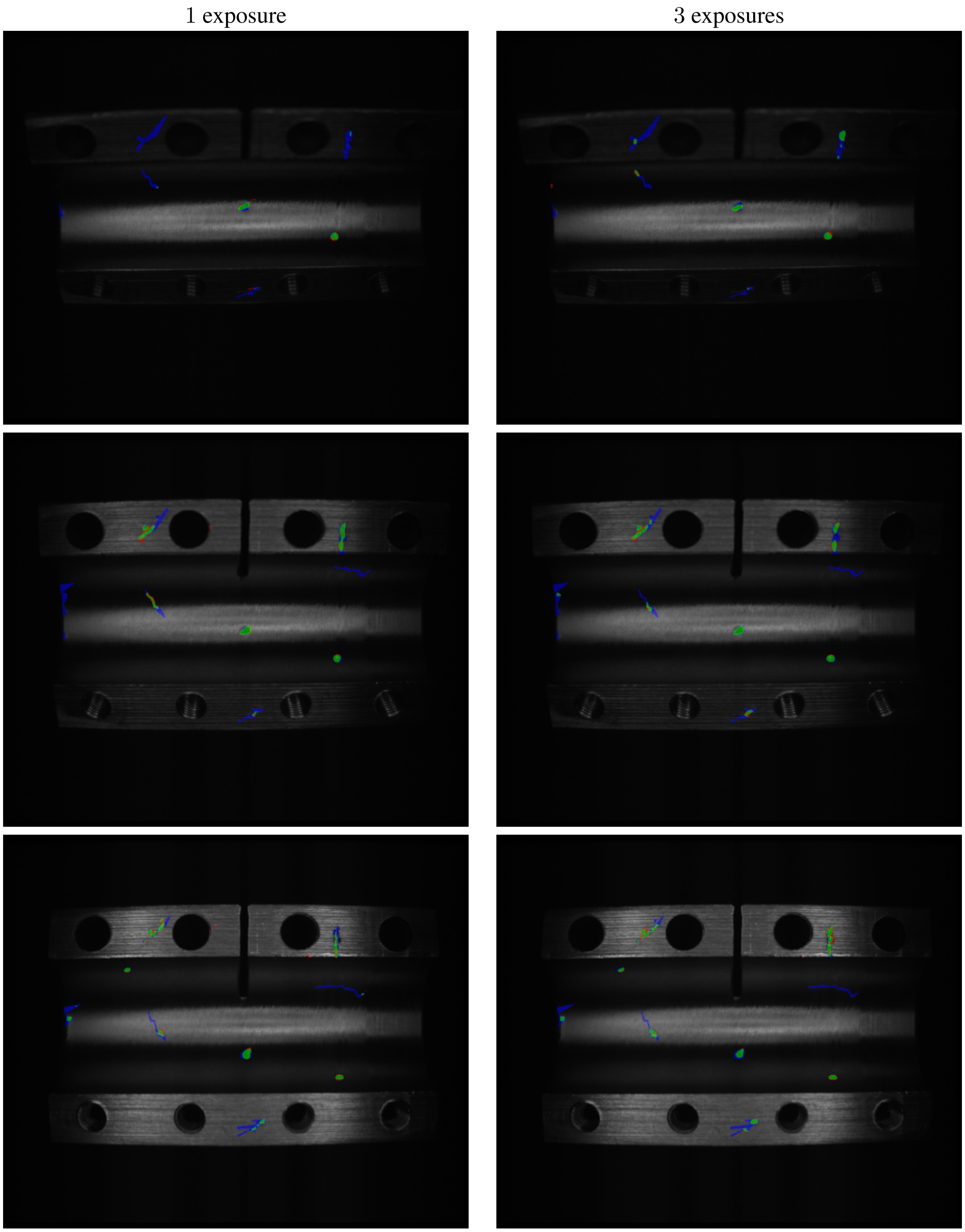

Figure 4. Example predictions of best DeepLabV3 pre-trained on SynthClutch and fine-tunned on RealClutch.

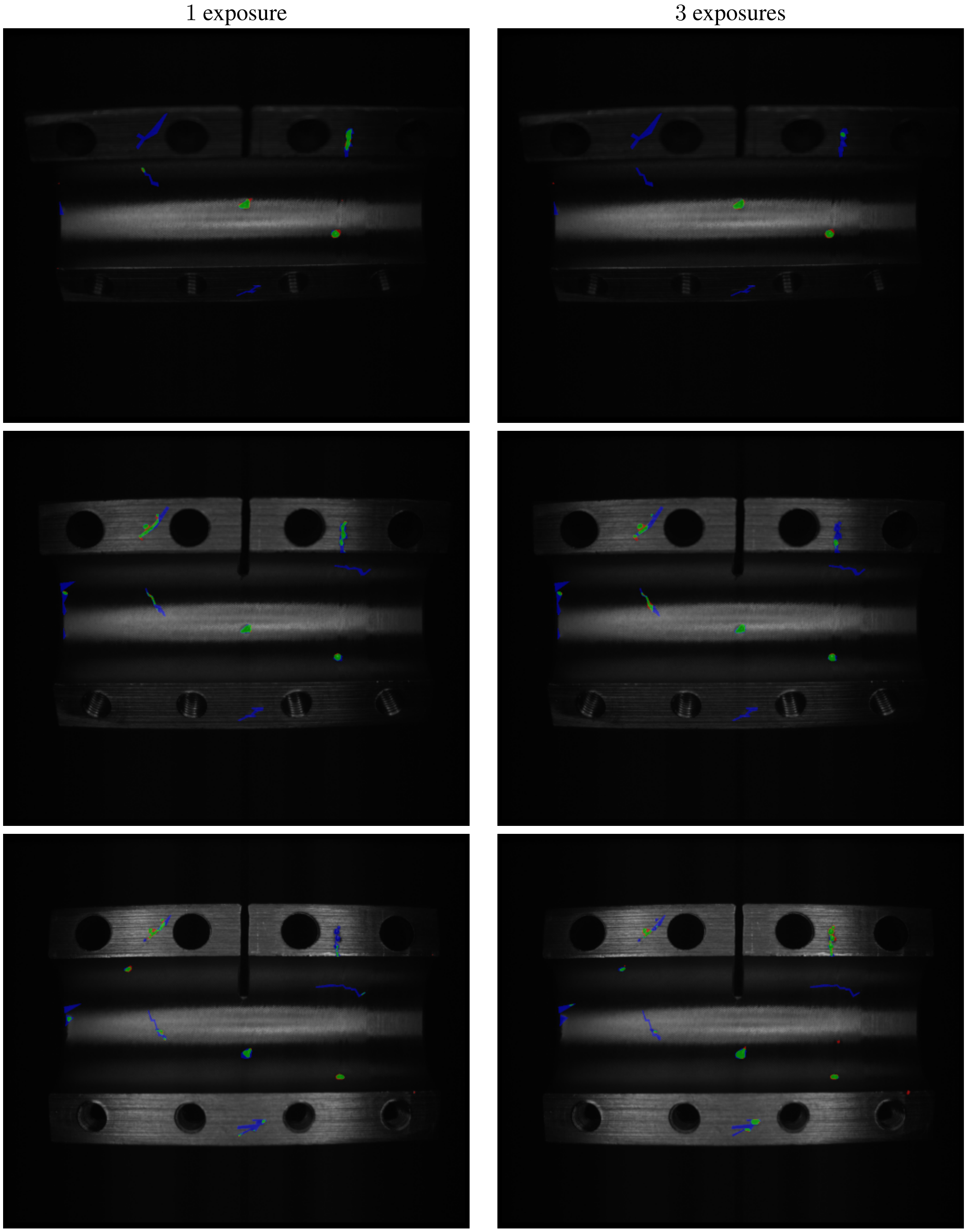

Figure 5. APRF plot of best U-Net pre-trained on SynthClutch and fine-tunned on RealClutch.

We also report the RealClutch test set plots of accuracy (A), precision (P), recall (R), F1 and F2 score dependent on the prediction binarization threshold, which we dub the *APRF plots*. The dotted line (TH) is the threshold value selected on the validation set of RealClutch, used to report the results in table 2 of the main paper.

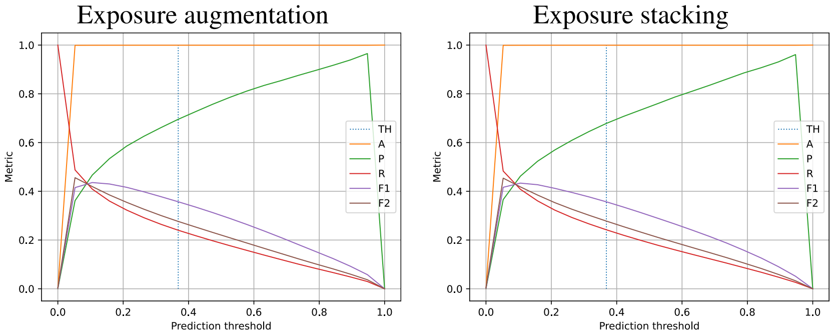

Figure 6. APRF plot of best FCN pre-trained on SynthClutch and fine-tunned on RealClutch.

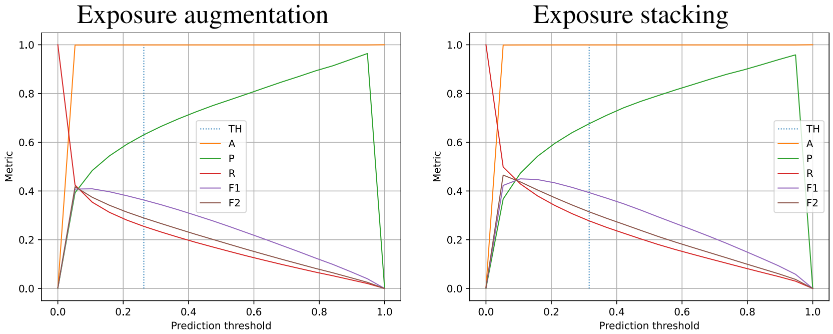

Figure 7. APRF plot of best DeepLabV3 pre-trained on SynthClutch and fine-tunned on RealClutch.

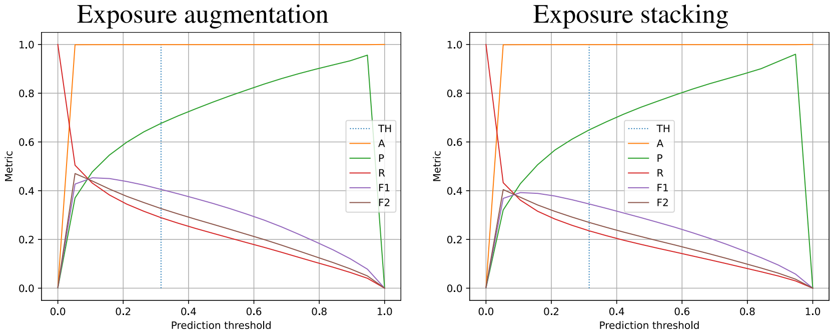

Figure 8. APRF plot of best U-Net pre-trained on SynthClutch and fine-tunned on RealClutch.

References

[1] Jakob Božič, Domen Tabernik, and Danijel Skočaj. Mixed supervision for surface-defect detection: from weakly to fully supervised learning. *Computers in Industry*, 2021. 3

[2] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv e-prints*, page arXiv:1706.05587, June 2017. 2

[3] David Honzátko, Engin Türetken, Siavash A. Bigdeli, L. Andrea Dunbar, and Pascal Fua. Defect segmentation for multi-illumination quality control systems. *Machine Vision and Applications*, 32(6):118, Sep 2021. 3

[4] Yibin Huang, Congying Qiu, Yue Guo, Xiaonan Wang, and Kui Yuan. Surface defect saliency of magnetic tile. In *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, pages 612–617, 2018. 3

[5] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3431–3440, Los Alamitos, CA, USA, jun 2015. IEEE Computer Society. 2

[6] O. Ronneberger, P.Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]). 2

[7] PAO Severstal. Severstal: Steel defect detection. https://www.kaggle.com/c/severstal-steel-defect-detection, 2019. 3

[8] Matthias Wieler, Tobias Hahn, and Fred. A. Hamprecht. Weakly supervised learning for industrial optical inspection [dataset]. https://hci.iwr.uni-heidelberg.de/content/weakly-supervised-learning-industrial-optical-inspection, 2007. 3