Diversity-Measurable Anomaly Detection: Supplementary Materials

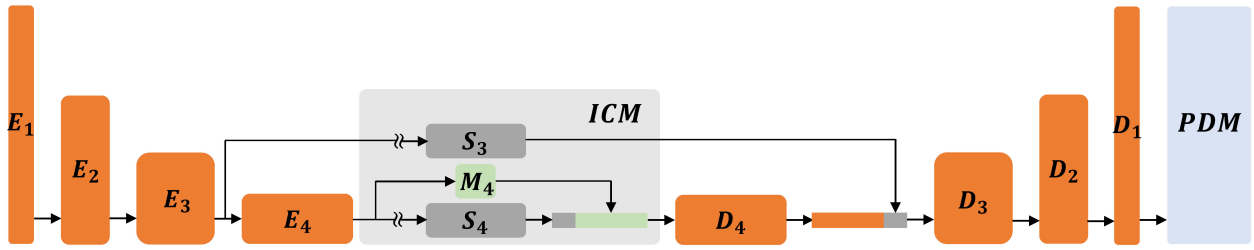

Figure 9. Detailed autoencoder architecture for PDM version of diversity-measurable anomaly detection.

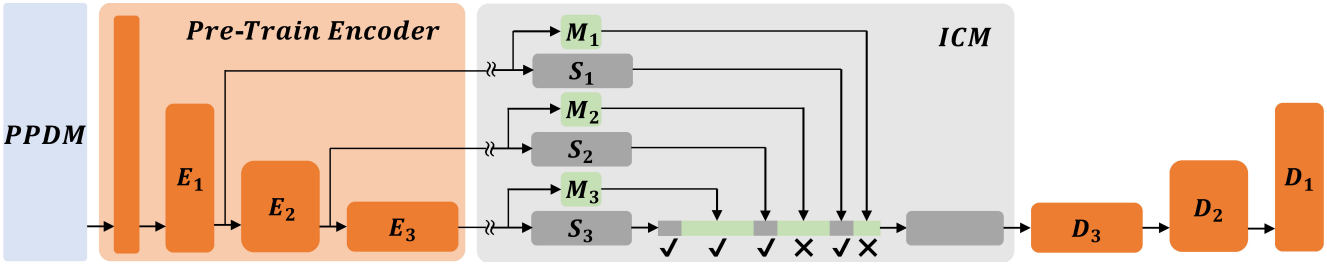

Figure 10. Detailed autoencoder architecture for PPDM version of diversity-measurable anomaly detection.

A. Detailed architectures

The detailed autoencoder architecture for anomaly detection in surveillance videos is shown in Fig. 9. The encoder contains four stages of different scales, where the Convolution blocks in each stage are: $Conv2d(c, c')$-$BN(c')$-$ReLU$-$Conv2d(c', c')$-$BN(c')$-$ReLU$ in turn and reduced skip-connections are applied in the last two stages. The latent feature dimensions are set to 128, 256, 512, 512 and 1024 for MNIST [6], Ped2 [11], Avenue [9], ShanghaiTech [10] and MVTec [2] respectively (the number of skip-connection dimensions is discussed in Appendix B). Besides, $Conv2d$ function is replaced by $CoordConv2d$ [8] in the decoder.

The backbone of PDM is composed of convolution blocks that same as the decoder and ResBlocks. The 1^{st} head includes two stride-2 ConvLayers, while the 2^{nd} does not. Then we generate a 2d-coordinate matrix with values between $[-1, 1]$ in both x and y directions, add it to the estimation and crop the output to values between $[-1, 1]$.

The autoencoder architecture for the PPDM version of DMAD framework is shown in Fig. 10. ICM receives three-layers of outputs from WideResNet [17] and performs reduction operation respectively, where the number of output channels is 64. Then stride-2 ConvLayer is used to reduce feature size to match the last layer. Finally, we concatenate the three 64-channel outputs with the quantized result and apply a residual bottleneck as in RD [4].

The autoencoder architecture used in toy experiment follows the implementation of VQ-VAE [14]. Please refer to the code for detailed settings.

B. Skip-connection channels and multi-level memory items

The reduced skip-connection channels is set to 32, 32, 64 and 64 for Ped2 [11], ShanghaiTech [10], Avenue [9] and MVTec [2] respectively. This value depends on how rich the detail is, *e.g.* we use more channels in Avenue [9] among surveillance videos, because it is collected by close shot and contains the highest resolution. And we follow the settings of RD [4] to obtain the output of WideResNet [17] directly for defect localization. The skip-connections are used for information compression in this architecture. A more detailed study on the number of skip-connection channels and multi-level memory items is shown in Tab. 6.

Table 6. Anomaly detection results of class "Screw" on MVTec [2] with different settings on skip-connection and multi-level memory. S_i and M_i mean the number of skip-connections and memory items in the i-th stage respectively. Hyperparameter α is set using grid-search.

Channels / Memory items						$AUC\%$
Stage-1		Stage-2		Stage-3		
S_1	M_1	S_2	M_2	S_3	M_3	
64	200	64	200	64	200	97.9
64	200	64	200	64	100	98.2
64	400	64	400	64	100	98.4
256	-	64	400	64	100	98.2
256	-	512	-	1024	200	97.8
256	-	512	-	64	200	98.7
64	-	64	-	64	100	98.3
64	-	64	-	64	200	**100.**
64	-	64	-	64	400	99.6

C. Details in foreground-background selection

Under normal circumstances, we jointly optimize x_{bg} and $f_m(\cdot)$ with reconstruction loss, where the background template is a learnable tensor $x_{bg} \in R^{C\times H\times W}$ and we estimate it by minimizing:

$$L_{bg} = \|x - x_{bg}\|_2. \tag{24}$$

Note that different camera angles, instance scales and light conditions may affect the convergence of the training process.

C.1. Training on sparse instances

The temporal and spatial sparsity is a serious problem in anomaly detection. Benefiting from learning a background template x_{bg}, we apply spatial weights by forcing $f_m(\cdot)$ to focus on the instances instead of the background in ShanghaiTech [10].

C.2. Training on different scenes

Estimating the foreground-background binary mask with input x directly allow for weighted mixture of background template x_{bg} and foreground reconstruction to improve prediction quality on the edge. However, different scenes and foreground scales in ShanghaiTech [10] make the training of x_{bg} and $f_m(\cdot)$ hard to convergence and lack of training data for each scene further increases the difficulty. We address these problems by initializing x_{bg} with traditional method (*e.g.* GMM) and applying an extra constraint on the binary mask $m = f_m(x)$:

$$L_{mask} = -\frac{1}{HW}\sum mlog(m) + max(0, \frac{1}{HW}\sum m - \epsilon), \tag{25}$$

where ϵ is a small margin, the first term encourages $f_m(\cdot)$ to conduct deterministic prediction and the second one makes the model tend to reconstruct the background based on x_{bg} instead of the reconstruction from memory items. The output mask of $f_m(x)$ and the final outputs of the reconstruction are shown in Fig. 11 qualitatively.

Note that although the algorithm improve the performance on different scenes, the constraint may introduce noise near the foreground, especially on the edges, which makes the results of fixed-view videos worse. So for fixed-view videos, we do not impose additional constraint to make the convergence easier.

Table 7. Ablation study of memory quantity on Ped2 [11].

$\sharp$ *Mem.*	5	20	100	200	500	1000
$AUC\%$	**99.7**	99.5	99.6	**99.7**	99.6	99.6

Figure 11. Qualitative results on ShanghaiTech [10]. The second row shows the binary masks output from $f_m(\cdot)$. The third row shows the reconstruction outputs, where the abnormal instances (as indicated by the red boxes) are not reconstructed well.

D. More Ablation study and explanation

D.1. Memory quantity

Tab. 7 demonstrates memory quantity is not a trade-off factor. We make further explanation through Fig. 3 in the main paper: tuning memory quantity without combination of memory items can only generate dataset-dependent nor-

Table 8. Ablation study of A_{rec}-*only* and A_{df}-*only* on 4 datasets [2, 9–11].

Task	Ped2	Avenue	Shanghai	MVTec
A_{rec}/A_{df}	99.4/88.9	91.7/89.2	77.7/77.8	99.2/75.2

mal pattern, so the boundary between normal and abnormal is in the long uncertain measurement (gray area in the last row). Our main contribution is pushing the unmeasurable reconstruction error to "*Abnormal Info.*" (gray area in the penultimate line) and compensating the insufficiency of normal diversity caused by memory reduction with diversity-measurable modeling, so that anomaly scores on the boundary are distinguishable and the memory sensitivity becomes lower.

D.2. Quantity of coarse-to-fine deformations

Tab. 4 in the main paper demonstrates that K is not a trade-off factor. Since the control grid generated by deformation module with size of 2^K is much larger than the target size (*e.g.* pedestrians and their limbs) on condition of $K > 2$, the redundant estimation heads cannot further decrease L_{rec}. So that L_{df} constrains the deformation and further increase of K will not bring performance gain.

D.3. Scoring function weight

Tab. 8 shows the results on settings of A_{rec}-*only* and A_{df}-*only*. We find using separate scoring function sometimes outperforms SOTA, though much worse than the combination. The value of hyperparameter α depends on whether the data includes significant geometrical variations. For pedestrian images with limited deformation, α is relatively big (0.2). For normal workpieces in MVTec with messy deformation as show in Fig. 12, the discrimination of A_{df} approaches saturation with increasing diversity, so that smaller α (0.05) can make use of more discriminative A_{rec}. If we set $\alpha = 0.2$ in all cases, PPDM also gets ***99.3%*** on MVTec [2] and outperforms SOTA.

D.4. Constraint weight

γ_1 is used to balance veracity and diversity of quantized embedding. It is not sensitive in our experiments. We set γ_1 to 1 for simplicity, just like VQ-VAE [14].

γ_2 is used to control the ability of diversity modeling (strength and smoothness). In order to meet *Cond.1* in Sec. 3.1, γ_2 needs to be small to allow for strong deformation in diverse normal patterns. Whereas PPDM may lead to over-deformation reconstruction on images (*e.g.* in MVTec), so we use larger γ_2 to alleviate shortcut learning while meeting *Cond.1* for this case. Actually, the performance only drops a little if we set $\gamma_2 = 1$ for all cases (Ped2 [11]: ***99.7%***→***99.5%***).

Table 9. One-class novelty (semantic shift) detection with $AUC(\%)$ on MNIST [6] and Fashion-MNIST [15]. Hyperparameter is set using grid-search.

Task	RD [4]	Ours
MNIST	99.3	**99.5**
Fashion-MNIST	95.0	**96.3**

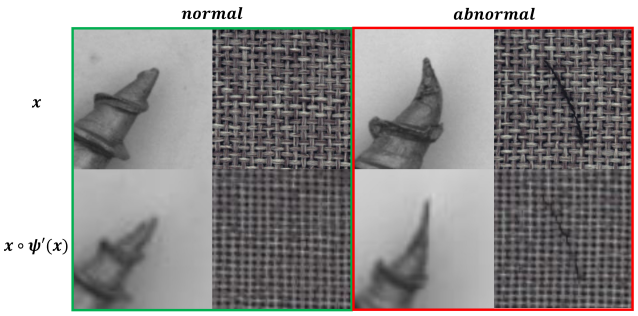

Figure 12. Qualitative results on MVTec [2]. PPDM tries to transform the original inputs to the reference with less details and degree of anomalies. The deformation together with reconstruction error determine the final anomaly score.

γ_3 is used to control the reconstruction granularity and alleviate shortcut learning as discussed in Sec. 4.5.

E. Semantic shift detection results

We verify DMAD (PPDM) in one-class novelty (semantic shift) detection on MNIST [6] and Fashion-MNIST [15] datasets. The results in Tab. 9 show DMAD outperforms the SOTA [4] in this task. Actually, DMAD works well on both semantic shift and covariate shift as long as they mainly involve geometrical diversity.

F. Detailed results on MVTec

We show more detailed results on MVTec [2] in Tab. 10 and Tab. 11 with hyperparameter α set by grid-search. PPDM can tolerate and eliminate diversity from normal images, while detect real damages which have more significant deformation and reconstruction error. As shown in Fig. 12, PPDM tries to construct reference for screw and carpet which need more deformation near the defects.

Table 10. Detailed detection results on MVTec [2].

Class\Methods	GN [1]	PSVDD [16]	DAAD [5]	CutPaste [7]	PaDiM [3]	PatchCore [12]	DRAEM [18]	RD [4]	Ours
Carpet	69.9	92.9	86.6	93.9	99.8	98.7	97.0	98.9	**100.**
Grid	70.8	94.6	95.7	**100.**	96.7	98.2	99.9	**100.**	**100.**
Leather	84.2	90.9	86.2	**100.**	**100.**	**100.**	**100.**	**100.**	**100.**
Tile	79.4	97.8	88.2	94.6	98.1	98.7	99.6	99.3	**100.**
Wood	83.4	96.5	98.2	99.1	99.2	99.2	99.1	99.2	**100.**
$\text{Avg}_{Textures}$	77.5	94.5	91.0	97.5	98.8	99.0	99.1	99.5	**100.**
Bottle	89.2	98.6	97.6	98.2	99.9	**100.**	99.2	**100.**	**100.**
Cable	75.7	90.3	84.4	81.2	92.7	**99.5**	91.8	95.0	99.1
Capsule	73.2	76.7	76.7	98.2	91.3	98.1	98.5	96.3	**98.9**
Hazelnut	78.5	92.0	92.1	98.3	92.0	**100.**	**100.**	99.9	**100.**
Metal Nut	70.0	94.0	75.8	99.9	98.7	**100.**	98.7	**100.**	**100.**
Pill	74.3	86.1	90.0	94.9	93.3	96.6	**98.9**	96.6	97.3
Screw	74.6	81.3	98.7	88.7	85.8	98.1	93.9	97.0	**100.**
Toothbrush	65.3	**100.**	99.2	99.4	96.1	**100.**	**100.**	99.5	**100.**
Transistor	79.2	91.5	87.6	96.1	97.4	**100.**	93.1	96.7	98.7
Zipper	74.5	97.9	85.9	99.9	90.3	99.4	**100.**	98.5	99.6
$\text{Avg}_{Objects}$	75.5	90.8	88.8	95.5	93.8	99.2	97.4	98.0	**99.4**
Avg_{All}	76.2	92.1	89.5	96.1	95.5	99.1	98.0	98.5	**99.6**

Table 11. Detailed localization results on MVTec [2].

Class\Methods	PSVDD [16]	CutPaste [7]	PaDiM [3]	PatchCore [12]	DRAEM [18]	TMAE [13]	RD [4]	Ours
Carpet	92.6	98.3	**99.1**	99.0	95.5	98.5	98.9	**99.1**
Grid	96.2	97.5	97.3	98.7	**99.7**	97.5	99.3	99.2
Leather	97.4	**99.5**	99.2	99.3	98.6	98.1	99.4	**99.5**
Tile	91.4	90.5	94.1	95.6	**99.2**	82.5	95.6	96.0
Wood	90.8	95.5	94.9	95.0	**96.4**	92.6	95.3	95.5
$\text{Avg}_{Textures}$	93.7	96.3	96.9	97.6	**97.9**	93.8	97.7	**97.9**
Bottle	98.1	97.6	98.3	98.6	**99.1**	93.4	98.7	98.9
Cable	96.8	90.0	96.7	**98.4**	94.7	92.9	97.4	98.1
Capsule	95.8	97.4	98.5	**98.8**	94.3	87.4	98.7	98.3
Hazelnut	97.5	97.3	98.2	98.7	**99.7**	98.5	98.9	99.1
Metal Nut	98.0	93.1	97.2	98.4	**99.5**	91.8	97.3	97.7
Pill	95.1	95.7	95.7	97.4	97.6	89.9	98.2	**98.7**
Screw	95.7	96.7	98.5	99.4	97.6	97.6	**99.6**	**99.6**
Toothbrush	98.1	98.1	98.8	98.7	98.1	98.1	99.1	**99.4**
Transistor	97.0	93.0	**97.5**	96.3	90.9	92.7	92.5	95.4
Zipper	95.1	**99.3**	98.5	98.8	98.8	97.8	98.2	98.3
$\text{Avg}_{Objects}$	96.7	95.8	97.8	**98.4**	97.0	94.0	97.9	**98.4**
Avg_{All}	95.7	96.0	97.5	98.1	97.3	93.9	97.8	**98.2**

References

[1] Samet Akcay, Amir Atapour Abarghouei, and Toby P. Breckon. Ganomaly: Semi-supervised anomaly detection via adversarial training. In *Computer Vision - ACCV 2018 - 14th Asian Conference on Computer Vision*, pages 622–637, 2018. 4

[2] Paul Bergmann, Michael Fauser, David Sattlegger, and

Carsten Steger. Mvtec AD - A comprehensive real-world dataset for unsupervised anomaly detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9592–9600, 2019. 1, 2, 3, 4

[3] Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: A patch distribution modeling framework for anomaly detection and localization. In *In International Conference on Pattern Recognition*, pages 475–489, 2020. 4

[4] Hanqiu Deng and Xingyu Li. Anomaly detection via reverse distillation from one-class embedding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9737–9746, 2022. 1, 3, 4

[5] Jinlei Hou, Yingying Zhang, Qiaoyong Zhong, Di Xie, Shiliang Pu, and Hong Zhou. Divide-and-assemble: Learning block-wise memory for unsupervised anomaly detection. In *2021 IEEE/CVF International Conference on Computer Vision*, pages 8771–8780, 2021. 4

[6] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, pages 2278–2324, 1998. 1, 3

[7] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 9664–9674, 2021. 4

[8] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018*, pages 9628–9639, 2018. 1

[9] Cewu Lu, Jianping Shi, and Jiaya Jia. Abnormal event detection at 150 FPS in MATLAB. In *IEEE International Conference on Computer Vision*, pages 2720–2727, 2013. 1, 3

[10] Weixin Luo, Wen Liu, and Shenghua Gao. A revisit of sparse coding based anomaly detection in stacked RNN framework. In *IEEE International Conference on Computer Vision*, pages 341–349, 2017. 1, 2, 3

[11] Vijay Mahadevan, Weixin Li, Viral Bhalodia, and Nuno Vasconcelos. Anomaly detection in crowded scenes. In *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition*, pages 1975–1981, 2010. 1, 2, 3

[12] Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter V. Gehler. Towards total recall in industrial anomaly detection. *arXiv preprint arXiv:2106.08265*, 2021. 4

[13] Daniel Stanley Tan, Yi-Chun Chen, Trista Pei-Chun Chen, and Wei-Chao Chen. Trustmae: A noise-resilient defect classification framework using memory-augmented auto-encoders with trust regions. In *IEEE Winter Conference on Applications of Computer Vision*, pages 276–285, 2021. 4

[14] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017*, pages 6306–6315, 2017. 1, 3

[15] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 3

[16] Jihun Yi and Sungroh Yoon. Patch SVDD: patch-level SVDD for anomaly detection and segmentation. In *Computer Vision - ACCV 2020 - 15th Asian Conference on Computer Vision*, pages 375–390, 2020. 4

[17] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *Proceedings of the British Machine Vision Conference 2016*, 2016. 1

[18] Vitjan Zavrtanik, Matej Kristan, and Danijel Skocaj. Dræm - A discriminatively trained reconstruction embedding for surface anomaly detection. In *2021 IEEE/CVF International Conference on Computer Vision*, pages 8310–8319, 2021. 4