

Table 3. Anomaly Detection Performance on MVTec AD [5] as measured in PRO [%] [5, 10].

Method	AE_{SSIM} [5]	Student [6]	SPADE [10]	PaDiM [14]	PatchCore-25%	PatchCore-10%	PatchCore-1%
PRO \uparrow	69.4	85.7	91.7	92.1	93.4	93.5	93.1
Error \downarrow	30.6	14.3	8.3	7.9	6.6	6.5	6.9

Table 4. PatchCore-1% with higher resolution/larger backbones/ensembles. The coreset subsampling allows for computationally expensive setups while still retaining fast inference.

Metric \rightarrow	AUROC	pwAUROC	PRO
DenseN-201 & RNext-101 & WRN-101 (2+3), Imagesize 320			
Score \uparrow	99.6	98.2	94.9
Error \downarrow	0.4	1.8	5.6
WRN-101 (2+3), Imagesize 280			
Score \uparrow	99.4	98.2	94.4
Error \downarrow	0.6	1.8	5.6
WRN-101 (1+2+3), Imagesize 280			
Score \uparrow	99.2	98.4	95.0
Error \downarrow	0.8	1.6	5.0

Table 5. Mean inference time per image on MVTec AD. Scores are (image AUROC, pixel AUROC, PRO metric).

Method	PatchCore-100%	PatchCore-10%	PatchCore-1%
Scores	(99.1, 98.0, 93.3)	(99.0, 98.1, 93.5)	(99.0, 98.0, 93.1)
Time (s)	0.6	0.22	0.17
Method	PatchCore-100% + IVFPQ	SPADE	PaDiM
Scores	(98.0, 97.9, 93.0)	(85.3, 96.6, 91.5)	(95.4, 97.3, 91.8)
Time (s)	0.2	0.66	0.19

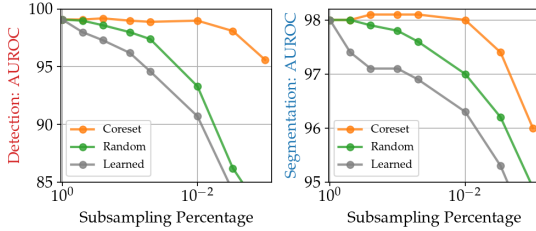


Figure 5. Performance retention for different subsamplers, results for PRO score in the supplementary.

4.4.1 Locally aware patch-features and hierarchies

We investigate the importance of locally aware patch-features (§3.3) by evaluating changes in anomaly detection performance over different neighbourhood sizes in Eq. 1. Results in the top half of Figure 4 show a clear optimum between locality and global context for patch-based anomaly predictions, thus motivating the neighbourhood size $p = 3$. More global context can also be achieved by moving down the network hierarchy (see e.g. [10, 14]), however at the cost of reduced resolution and heavier ImageNet class bias (§3.1). Indexing the first three WideResNet50-blocks with 1 - 3, Fig. 4 (bottom) again highlights an optimum between highly localized predictions, more global context and ImageNet bias. As can be seen, features from hierarchy level 2

can already achieve state-of-the-art performance, but benefit from additional feature maps taken from subsequent hierarchy levels (2+3, which is chosen as the default setting).

4.4.2 Importance of Coreset subsampling

Figure 5 compares different memory bank \mathcal{M} subsampling methods: Greedy coreset selection, random subsampling and learning of a set of basis proxies corresponding to the subsampling target percentage p_{target} . For the latter, we sample proxies $p_i \in \mathcal{P} \subset \mathbb{R}^d$ with $|\mathcal{P}| = p_{\text{target}} \cdot |\mathcal{M}|$, which are then tasked to minimize a basis reconstruction objective

$$\mathcal{L}_{\text{rec}}(m_i) = \left\| m_i - \sum_{p_k \in \mathcal{P}} \frac{e^{\|m_i - p_k\|_2}}{\sum_{p_j \in \mathcal{P}} e^{\|m_i - p_j\|_2}} p_k \right\|_2^2, \quad (8)$$

to find N proxies that best describe the memory bank data \mathcal{M} . In Figure 5 we compare the three settings and find that coreset-based subsampling performs better than the other possible choices. The performance of no subsampling is comparable to a coreset-reduced memory bank that is two orders of magnitudes smaller in size. We also find subsampled memory banks to contain much less redundancy. We recorded the percentage of memory bank samples that are used at test time for non-subsampled and coreset-subsampled memory banks. While initially only less than 30% of memory bank samples are used, coreset subsampling (to 1%) increases this factor to nearly 95%. For certain subsampling intervals (between around 50% and 10%), we even find joint performance over anomaly detection and localization to partly increase as compared to non-subsampled PatchCore. Finally, reducing the memory bank size \mathcal{M} by means of increased striding (see Eq. 3) shows worse performance due to the decrease in resolution context, with stride $s = 2$ giving an image anomaly detection AUROC of 97.6%, and stride $s = 3$ an AUROC of 96.8%.

4.5. Low-shot Anomaly Detection

Having access to limited nominal data is a relevant setting for real-world inspection. Therefore in addition to reporting results on the full MVTec AD, we also study the performance with fewer training examples. We vary the amount of training samples from 1 (corresponding to 0.4% of the total nominal training data) to 50 (21%), and compare to reimplementations of SPADE [10] and PaDiM [14] using the same backbone (WideResNet50). Results are summarized in Figure 6, with detailed results available in Supp.