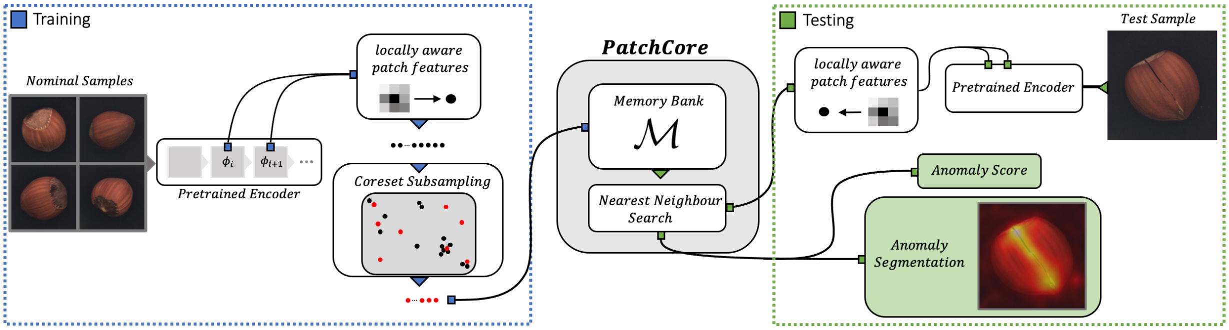
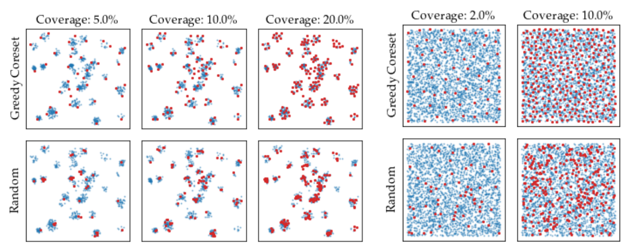


Figure 2. Overview of *PatchCore*. Nominal samples are broken down into a memory bank of neighbourhood-aware patch-level features. For reduced redundancy and inference time, this memory bank is downsampled via greedy coreset subsampling. At test time, images are classified as anomalies if at least one patch is anomalous, and pixel-level anomaly segmentation is generated by scoring each patch-feature.



(a) (b)

Figure 3. Comparison: coreset (top) vs. random subsampling (bottom) (red) for 2D data (iblue) sampled from (a) multimodal and (b) uniform distributions. Visually, coreset subsampling better approximates the spatial support, random subsampling misses clusters in the multi-modal case and is less uniform in (b).

multiple feature hierarchies to offer some benefit. However, to retain the generality of used features as well as the spatial resolution, *PatchCore* uses only two intermediate feature hierarchies j and $j+1$. This is achieved simply by computing $\mathcal{P}_{s,p}(\phi_{i,j+1})$ and aggregating each element with its corresponding patch feature at the lowest hierarchy level used (i.e., at the highest resolution), which we achieve by bilinearly rescaling $\mathcal{P}_{s,p}(\phi_{i,j+1})$ such that $|\mathcal{P}_{s,p}(\phi_{i,j+1})|$ and $|\mathcal{P}_{s,p}(\phi_{i,j})|$ match.

Finally, for all nominal training samples $x_i \in \mathcal{X}_N$, the *PatchCore* memory bank $\mathcal{M}$ is then simply defined as

$$\mathcal{M} = \bigcup_{x_i \in \mathcal{X}_N} \mathcal{P}_{s,p}(\phi_j(x_i)). \tag{4}$$

3.2. Coreset-reduced patch-feature memory bank

For increasing sizes of $\mathcal{X}_N$, $\mathcal{M}$ becomes exceedingly large and with it both the inference time to evaluate novel test data and required storage. This issue has already been noted in SPADE [10] for anomaly segmentation, which makes use of both low- and high-level feature maps. Due to computational limitations, SPADE requires a preselection stage of feature maps for pixel-level anomaly detection based on a weaker image-level anomaly detection mechanism reliant on full-image, deep feature representations, i.e., global averaging of the last feature map. This results in low-resolution, ImageNet-biased representations computed from full images which may negatively impact detection and localization performance.

These issues can be addressed by making $\mathcal{M}$ meaningfully searchable for larger image sizes and counts, allowing for patch-based comparison beneficial to both anomaly detection and segmentation. This requires that the nominal feature coverage encoded in $\mathcal{M}$ is retained. Unfortunately, random subsampling, especially by several magnitudes, will lose significant information available in $\mathcal{M}$ encoded in the coverage of nominal features (see also experiments done in §4.4.2). In this work we use a coreset subsampling mechanism to reduce $\mathcal{M}$, which we find reduces inference time while retaining performance.

Conceptually, coreset selection aims to find a subset $\mathcal{S} \subset \mathcal{A}$ such that problem solutions over $\mathcal{A}$ can be most closely and especially more quickly approximated by those computed over $\mathcal{S}$ [1]. Depending on the specific problem, the coreset of interest varies. Because *PatchCore* uses nearest neighbour computations (next Section), we use a *minimax facility location* coreset selection, see e.g., [48] and [49], to ensure approximately similar coverage of the $\mathcal{M}$-coreset $\mathcal{M}_C$ in patch-level feature space as compared to the original memory bank $\mathcal{M}$

$$\mathcal{M}_C^* = \underset{\mathcal{M}_C \subset \mathcal{M}}{\arg\min} \max_{m \in \mathcal{M}} \min_{n \in \mathcal{M}_C} \|m - n\|_2 . \tag{5}$$

The exact computation of $\mathcal{M}_C^*$ is NP-Hard [54], we use the iterative greedy approximation suggested in [48]. To further reduce coreset selection time, we follow [49], making use of the Johnson-Lindenstrauss theorem [11] to reduce dimensionalities of elements $m \in \mathcal{M}$ through random linear projections $\psi : \mathbb{R}^d \to \mathbb{R}^{d^*}$ with $d^* < d$. The memory bank reduction is summarized in Algorithm 1. For notation, we