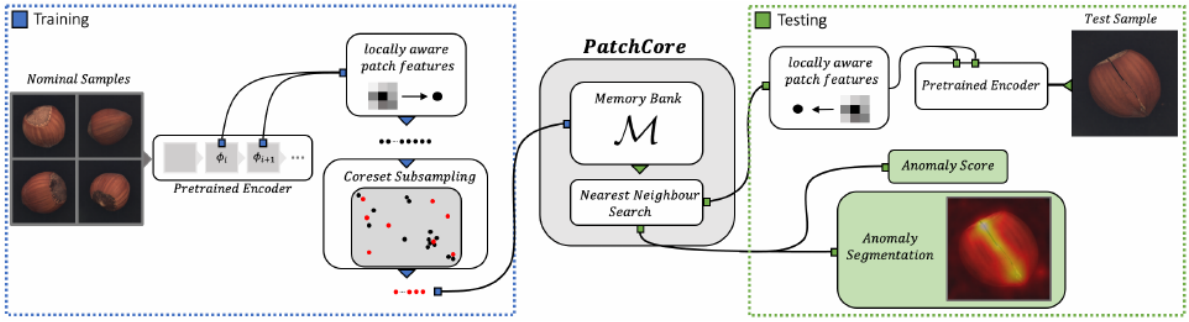

図2. PatchCoreの概要。名義サンプルは、近傍意識型パッチレベル特徴量のメモリバンクに分解されます。冗長性と推論時間の削減のため、このメモリバンクは貪欲なコアセットサブサンプリングによりダウンサンプリングされる。テスト時、少なくとも1つのパッチが異常であれば画像は異常と分類され、各パッチ特徴のスコアリングによりピクセルレベル異常セグメンテーションが生成される。

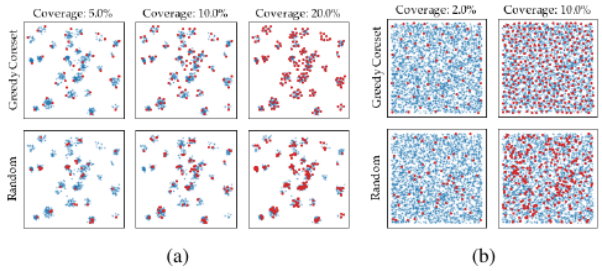

図3. 比較：コアセット（上）対ランダムサブサンプリング（下）（赤）2次元データ（iblue）を（a）多峰分布と（b）一様分布からサンプリングした場合。視覚的に、コアセットサブサンプリングは空間的サポートをより正確に近似し、ランダムサブサンプリングは多峰分布の場合にクラスターを捕捉できず、（b）では一様性が低い。

複数の特徴階層を提供することで一部の効果をもたらします。しかし、使用される特徴の汎用性と空間解像度を維持するため、PatchCoreは中間特徴階層jとj + 1のみを使用します。これは、$P_{s,p}(\phi_{i,j+1})$を計算し、各要素を最も低い階層レベル（つまり最高解像度）で使用される対応するパッチ特徴量と集約することで実現されます。これには、$P_{s,p}(\phi_{i,j+1})$を双線形補間により再スケーリングし、|P_{s,p}$(\phi_{i,j+1})$|と|$P_{s,p}(\phi_{i,j})$|が一致するようにします。

最後に、すべての標準トレーニングサンプル $x_i \in X_N$ に対し、PatchCoreメモリバンク M は単純に次のように定義される

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

3.2. コアセット削減パッチ特徴量メモリバンク

X_N のサイズが増加するにつれ、Mは極めて大規模になり、新規テストデータの評価に必要な推論時間とストレージ容量も増加します。この問題は、低レベルと高レベルの特徴マップを併用する異常セグメンテーションにおけるSPADE [10]でも既に指摘されています。

計算上の制約により、SPADEは画像レベル異常検出メカニズム（全画像の深層特徴表現に基づくグローバル平均化）に依存する弱いメカニズムに基づくピクセルレベル異常検出のための特徴マップの事前選択段階を必要とします。これにより、全画像から計算された低解像度でImageNetバイアスのかかった表現が生成され、検出と局所化性能に悪影響を及ぼす可能性があります。

これらの問題は、Mを大きな画像サイズとカウントに対して意味のある検索可能にすることで解決できます。これにより、異常検出とセグメンテーションの両方に有益なパッチベースの比較が可能になります。これには、Mにエンコードされた名目上の特徴のカバー範囲を維持する必要があります。残念ながら、特に数桁の規模でランダムサブサンプリングを行うと、Mにエンコードされた名目上の特徴のカバー範囲に含まれる重要な情報が失われます（§4.4.2の実験も参照）。本研究では、Mを削減するためにコアセットサブサンプリングメカニズムを使用し、推論時間を短縮しつつ性能を維持することを確認しました。

概念的に、コアセット選択は、集合 A における問題の解を、集合 S における解によって最も正確かつ特に迅速に近似できるような部分集合 $S \subset A$ を求めることを目的とします [1]。問題の具体的内容に応じて、関心のあるコアセットは異なります。PatchCoreは最近傍計算（次節）を使用するため、M_Cのパッチレベル特徴空間におけるカバー範囲が元のメモリバンクMとほぼ同様になるよう、ミニマックス施設位置コアセット選択（例：[48]および[49]）を採用しています。

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

M_C^* の正確な計算は NP-Hard である [54] ため、[48] で提案された反復的貪欲近似法を用いる。コアセット選択時間をさらに短縮するため、[49] に従い、ジョンソン・リンドェンシュトラウス定理 [11] を利用して、要素 $m \in M$ の次元を、$d^* < d$ となるランダムな線形写像 $\phi : R^d \to R^{d^*}$ を通じて次元削減します。メモリバンクの削減はアルゴリズム 1 に要約されています。表記については、

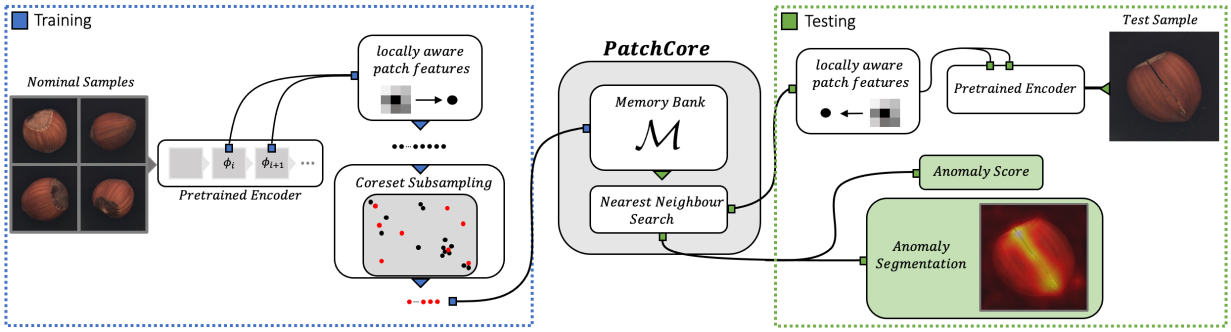

Figure 2. Overview of *PatchCore*. Nominal samples are broken down into a memory bank of neighbourhood-aware patch-level features. For reduced redundancy and inference time, this memory bank is downsampled via greedy coreset subsampling. At test time, images are classified as anomalies if at least one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

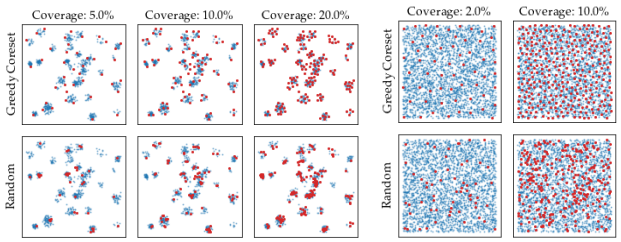

Figure 3. Comparison: coreset (top) vs. random subsampling (bottom) (red) for 2D data (iblue) sampled from (a) multimodal and (b) uniform distributions. Visually, coreset subsampling better approximates the spatial support, random subsampling misses clusters in the multi-modal case and is less uniform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

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

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