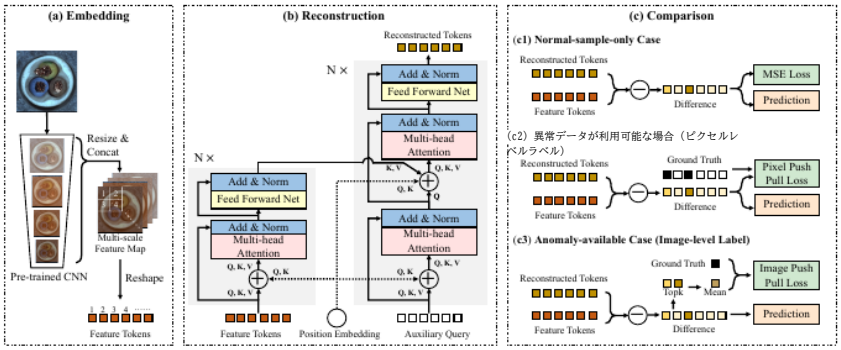

図2：当手法の概要。(a) 埋め込み：事前訓練されたCNNバックボーンを適用してマルチスケール特徴を抽出します。(b) 再構築：補助的な学習可能なクエリ埋め込みを用いて、特徴トークンを再構築するためにトランスフォーマーを利用します。(c) 比較：当手法は、正常サンプルのみの場合と異常データが利用可能な場合の両方に適用可能です。異常スコアマップは、抽出された特徴と再構築された特徴の差分から得られます。

層1から層5までの特徴は同じサイズにリサイズされ、結合されてマルチスケール特徴マップ、f $\in R^{C\times H\times W}$ を形成します。ここで、層は同じ特徴サイズを持つステージの組み合わせとして定義されます。マルチスケール特徴マップを採用するのは、異なる層の特徴マップが異なる受容野レベルを持ち、異なる異常に対して敏感だからです。

再構築。再構築段階は図2bに示されています。特徴マップ f \in \mathbb{R}^{C \times H \times W} は、まずH × Wの特徴トークンに分割されます。計算コストを削減するため、これらのトークンをトランスフォーマーに投入する前に、1×1畳み込みが適用されて次元が削減されます。また、トランスフォーマーから出力される際に、別の1×1畳み込みにより次元が回復されます。トランスフォーマーエンコーダーは、入力特徴トークンを潜在特徴空間に埋め込みます。各エンコーダー層は、マルチヘッドアテンション、フィードフォワードネットワーク（FFN）、残差接続、正規化を含む標準アーキテクチャ[33]に従います。トランスフォーマーデコーダーは、補助クエリ埋め込みを含む標準アーキテクチャ[33]に従います。補助クエリは、入力特徴トークンと同じサイズの学習済み埋め込みです。トランスフォーマーデコーダーは、マルチヘッド自己注意とエンコーダー-デコーダー注意メカニズムを使用して、この学習済みクエリ埋め込みを特徴トークンに再構築します。トランスフォーマーが置換不変であるため、学習済み位置埋め込み[9]が含まれています。
比較。通常サンプルのみの場合、モデルはバックボーンから抽出された特徴量fと再構築された特徴量f ^ $\in R^{C\times H\times W}$間のMSE損失、$L_{norm}$で訓練されます。

$$\mathcal{L}_{norm} = \frac{1}{H \times W} ||\boldsymbol{f} - \hat{\boldsymbol{f}}||_2^2. \tag{1}$$

推論。まず、特徴差分マップ d(i, u) を次のように定義します。

$$\boldsymbol{d}(i,u) = \boldsymbol{f}(i,u) - \hat{\boldsymbol{f}}(i,u), \tag{2}$$

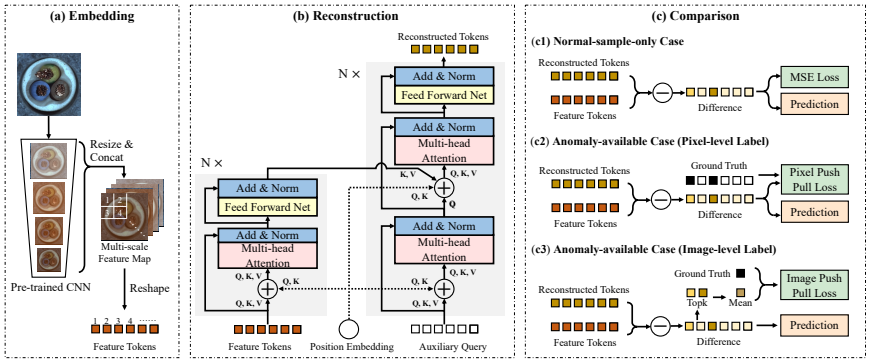

Fig. 2: **Overview of our method.** (a) Embedding: a pre-trained CNN backbone is applied to extract the multi-scale features. (b) Reconstruction: a transformer is utilized to reconstruct the feature tokens with an auxiliary learnable query embedding. (c) Comparison: our approach is compatible with both normal-sample-only case and anomaly-available case. The anomaly score maps are obtained through the differences between extracted and reconstructed features.

The features from *layer1* to *layer5* are resized to the same size, then concatenated together to form a multi-scale feature map, $\boldsymbol{f} \in \mathbb{R}^{C\times H\times W}$. Note that here we define *layer* as the combination of stages with the same size of features. We adopt multi-scale feature map because feature maps from different layers have different levels of receptive fields thus are sensitive to different anomalies.

Reconstruction. The reconstruction stage is shown in Fig. 2b. The feature map, $\boldsymbol{f} \in \mathbb{R}^{C\times H\times W}$, is first split to $H \times W$ feature tokens. To reduce the computation consumption, a 1×1 convolution is applied to reduce the dimension of these tokens before they are fed into the transformer. Also, their dimensions are recovered by another 1×1 convolution when output by transformer. The transformer encoder embeds the input feature tokens into a latent feature space. Each encoder layer follows the standard architecture [33] with multi-head attention, feed forward network (FFN), residual connection, and normalization. The transformer decoder follows the standard architecture [33] with an auxiliary query embedding. The auxiliary query is a learned embedding with the same size of the input feature tokens. The transformer decoder transforms this learned query embedding to reconstruct the feature tokens using multi-head self-attention and encoder-decoder attention mechanisms. The learned position embedding [9] is included because transformer is permutation-invariant.

Comparison. In normal-sample-only case, the model is trained with the MSE loss, $\mathcal{L}_{norm}$, between the backbone extracted features, $\boldsymbol{f}$, and the reconstructed features, $\hat{\boldsymbol{f}} \in \mathbb{R}^{C\times H\times W}$, as follows,

$$\mathcal{L}_{norm} = \frac{1}{H\times W}||\boldsymbol{f} - \hat{\boldsymbol{f}}||_2^2. \tag{1}$$

Inference. We first define the feature difference map, $\boldsymbol{d}(i,u)$, as,

$$\boldsymbol{d}(i,u) = \boldsymbol{f}(i,u) - \hat{\boldsymbol{f}}(i,u), \tag{2}$$