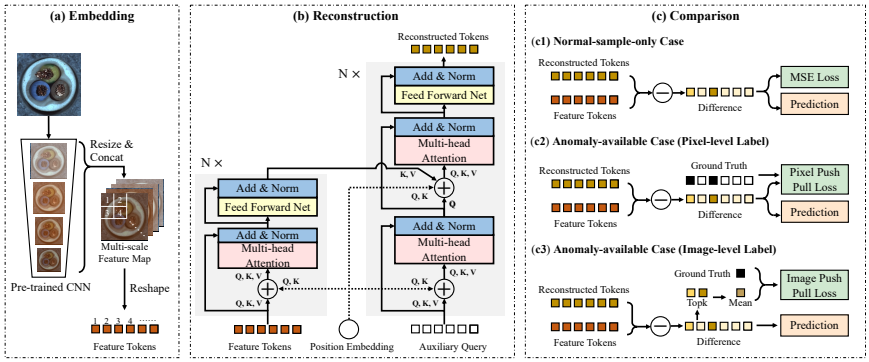

Fig. 2: **Overview of our method.** (a) Embedding: a pre-trained CNN backbone is applied to extract the multi-scale features. (b) Reconstruction: a transformer is utilized to reconstruct the feature tokens with an auxiliary learnable query embedding. (c) Comparison: our approach is compatible with both 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,

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