

3.1 特徴抽出

当社の方法の第一段階は、強力な画像レベルの特徴抽出です。同じ特徴は後でピクセルレベルの画像アラインメントにも使用されます。特徴抽出には複数のオプションがあります。最も一般的に使用されるオプションは、自己教師付き特徴学習、つまり入力正常画像から直接特徴を学習する手法です。この方法は魅力的ですが、小さなトレーニングデータセットで学習された特徴が、高品質な類似度測定として十分であるかどうかは明らかではありません。Bergmanら [4] の分析では、自己教師付き特徴は異常検出の目的においてImageNetで訓練されたResNet特徴よりも劣ることが示されています。そのため、私たちはImageNetデータセットで事前訓練されたResNet特徴抽出器を使用しました。画像レベルの特徴量として、最後の畳み込み層のグローバルプーリング後に得られる特徴ベクトルを使用しました。グローバル特徴量抽出器を F とし、与えられた画像 $x_{\{i\}}$ に対して抽出された特徴量を $f_{\{i\}}$ と表します：

$$f_i = F(x_i) \quad (1)$$

初期化時に、すべてのトレーニング画像（すべて正常）の特徴量が計算され保存されます。推論時には、ターゲット画像の特徴量のみが抽出されます。

3.2 K近傍法による正常画像の検索

当社の方法の最初の段階は、DN2 [4] を使用して異常を含む画像を選択することです。テスト画像 y に対して、トレーニングセットからその K 近傍の正常画像 $N_{\{k\}}$ ($f_{\{y\}}$) を取得します。距離は、画像レベルの特徴表現間のユークリッド距離で測定されます。

$$d(y) = \frac{1}{K} \sum_{f \in N_K(f_y)} \|f - f_y\|^2 \quad (2)$$

この段階で画像は正常または異常としてラベル付けされます。正の分類は、 kN 距離が閾値 τ より大きいことを確認することで決定されます。ほとんどの画像は正常であり、異常と指定される画像は少数であることが想定されています。

3.3 イメージアライメントによるサブイメージ異常検出

画像レベル段階で異常とラベル付けされた後、目的は1つまたは複数の異常のピクセルを特定しセグメント化することです。画像が誤って異常と分類された場合、目的は異常としてマークされるピクセルをゼロにすることです。

動機付けのアイデアとして、テスト画像を取得した正常画像に一致させることを検討しましょう。テスト画像と正常画像の差異を特定することで、異常なピクセルを検出することを目指します。この単純な方法はいくつかの欠点があります。i) 対象物に複数の正常な部分が存在すると仮定しています。

3.1 Feature Extraction

The first stage of our method is the extraction of strong image level features. The same features are later used for pixel-level image alignment. There are multiple options for extracting features. The most commonly used option is self-supervised feature learning, that is, learning features from scratch directly on the input normal images. Although it is an attractive option, it is not obvious that the features learned on small training datasets will indeed be sufficient for serving as high-quality similarity measures. The analysis performed in Bergman et al. [4] illustrates that self-supervised features underperform ImageNet-trained ResNet features for the purposes of anomaly detection. We therefore used a ResNet feature extractor pre-trained on the ImageNet dataset. As image-level features we used the feature vector obtained after global-pooling the last convolutional layer. Let us denote the global feature extractor F , for a given image x_i , we denote the extracted features f_i :

$$f_i = F(x_i) \quad (1)$$

At initialization, the features for all training images (which are all normal) are computed and stored. At inference, only the features of the target image are extracted.

3.2 K Nearest Neighbor Normal Image Retrieval

The first stage in our method is determining which images contain anomalies using DN2 [4]. For a given test image y , we retrieve its K nearest normal images from the training set, $N_K(f_y)$. The distance is measured using the Euclidean metric between the image-level feature representations.

$$d(y) = \frac{1}{K} \sum_{f \in N_K(f_y)} \|f - f_y\|^2 \quad (2)$$

Images are labelled at this stage as normal or anomalous. Positive classification is determined by verifying if the kNN distance is larger than a threshold τ . It is expected that most images are normal, and only few images are designated as anomalous.

3.3 Sub-image Anomaly Detection via Image Alignment

After being labelled as anomalous at the image-level stage, the objective is to locate and segment the pixels of one or multiple anomalies. In the case that the image was falsely classified as anomalous, our objective would be to mark no pixels as anomalous.

As a motivational idea, let us consider aligning the test image to a retrieved normal image. By finding the differences between the test and normal image, we would hope to detect the anomalous pixels. This naive method has several flaws i) assume that there are multiple normal parts the object may possibly