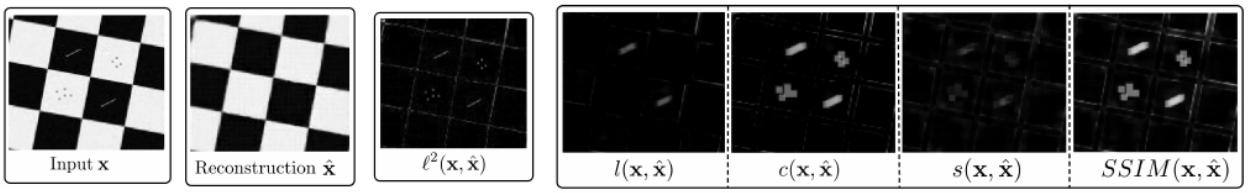

図3：`²よりもSSIMが欠陥のセグメンテーションに優れることを示す単純な例。（a）128×128のチェッカーボードパターンに、欠陥を模擬する灰色の線と点を含む。（b）欠陥のないチェッカーボードパターンで訓練された`² -オートエンコーダーによる入力画像xの出力再構築x^。オートエンコーダーにより欠陥が除去されている。（c）`² -残差マップ。明るい色は入力と再構築の間の不一致が大きいことを示します。（d）輝度 l 、コントラスト c 、構造 s 、およびそれらの点積から得られる最終的なSSIM残差マップの残差。`² -誤差マップとは対照的に、SSIMは再構築されたエッジ周辺のわずかな不正確さよりも、視覚的に目立つ擾乱に優先順位を置きます。

Q(z|x)から空間残差マップを生成する方法は、Q(z|x)から抽出したN個の潜在サンプルz_1、z_2、…、z_Nを復号化し、ピクセルごとの再構築確率R_{VAE} = P(x|z_1、z_2、… . , z_N) を評価することです。これはAnとCho（2015）で説明されている方法です。

3.1.3. 特徴マッチングオートエンコーダー。DosovitskiyとBrox（2016）は、標準的なオートエンコーダーへの別の拡張を提案しました。これは、入力画像xとその再構築x^から特徴を抽出し、それらを等しくする条件を課すことで、生成される再構築の品質を向上させます。F :$^{k\times h\times w}_{R}$ →$^{f}_{R}$を、入力画像からf次元の特徴ベクトルを取得する特徴抽出関数とします。次に、オートエンコーダの損失関数に正則化項を追加することで、特徴マッチングオートエンコーダ（FM-AE）損失が得られます。

$$L_{\mathrm{FM}}(\mathbf{x},\hat{\mathbf{x}}) = L_2(\mathbf{x},\hat{\mathbf{x}}) + \lambda\|F(\mathbf{x}) - F(\hat{\mathbf{x}})\|_2^2 \ , \qquad (3)$$

ここで、λ > 0 は2つの損失項間の重み付け係数を表す。F は、画像分類タスクで事前訓練されたCNNの最初の層を使用してパラメータ化できます。評価時、x と x^ のピクセルごとの `² -距離を比較することで、残差マップ $R_{FM}$ が得られます。より鋭く現実的な再構築が、標準の `² -オートエンコーダーと比較してより良い残差マップを生成するとの期待があります。

3.1.4. SSIM オートエンコーダー。私たちは、VAEsやFM-AEsのようなより複雑なアーキテクチャを採用しても、無監督欠陥セグメンテーションタスクにおいて、確定的な`² -オートエンコーダーよりも残差マップの改善が満足のいくものでないことを示します。これらはすべて、隣接するピクセル間の現実的でない独立性を仮定するピクセル単位の評価指標に基づいています。したがって、入力と再構築間の構造的差異を検出できません。画像領域間の局所的な相互依存関係を捕捉するように損失関数と評価関数を適応させることで、上述のすべてのアーキテクチャを大幅に改善できます。セクション3.2では、オートエンコーダーの損失関数および評価指標として構造的類似性指標SSIM（x, x^）を採用し、残差マップ R_{SSIM} を取得する理由を具体的に説明します。

3.2. 構造的類似性

SSIM指標（Wang et al., 2004）は、2つのK×K画像パッチpとq間の距離を測定し、輝度l(p, q)、コントラストc(p, q)、構造s(p, q)の類似性を考慮します：

$$\mathrm{SSIM}(\mathbf{p},\mathbf{q}) = l(\mathbf{p},\mathbf{q})^{\alpha} c(\mathbf{p},\mathbf{q})^{\beta} s(\mathbf{p},\mathbf{q})^{\gamma} \ , \qquad (4)$$

ここで、α、β、γ $\in_R$ は、3つの項の重みを決定するユーザー定義の定数です。輝度測定値 l(p , q) は、パッチの平均強度 μ_p と μ_q を比較することで推定されます。コントラスト測定値 c(p , q) はパッチの分散 σ_p^2 と σ_q^2 の関数です。構造測定値 s(p , q) は2つのパッチの共分散 σ_{pq} を考慮します。3つの測定値は次のように定義されます：

$$l(\mathbf{p},\mathbf{q}) = \frac{2\mu_{\mathbf{p}}\mu_{\mathbf{q}} + c_1}{\mu_{\mathbf{p}}^2 + \mu_{\mathbf{q}}^2 + c_1} \qquad (5)$$

$$c(\mathbf{p},\mathbf{q}) = \frac{2\sigma_{\mathbf{p}}\sigma_{\mathbf{q}} + c_2}{\sigma_{\mathbf{p}}^2 + \sigma_{\mathbf{q}}^2 + c_2} \qquad (6)$$

$$s(\mathbf{p},\mathbf{q}) = \frac{2\sigma_{\mathbf{pq}} + c_2}{2\sigma_{\mathbf{p}}\sigma_{\mathbf{q}} + c_2} \ . \qquad (7)$$

定数 c_1 と c_2 は数値的安定性を確保するためのもので、通常は c_1 = 0.01 と c_2 = 0.03 に設定されます。式（5）～（7）を式（4）に代入すると、SSIM は次のように与えられます

$$\mathrm{SSIM}(\mathbf{p},\mathbf{q}) = \frac{(2\mu_{\mathbf{p}}\mu_{\mathbf{q}} + c_1)(2\sigma_{\mathbf{pq}} + c_2)}{(\mu_{\mathbf{p}}^2 + \mu_{\mathbf{q}}^2 + c_1)(\sigma_{\mathbf{p}}^2 + \sigma_{\mathbf{q}}^2 + c_2)} \ . \qquad (8)$$

SSIM(p, q) ∈ [-1, 1] です。特に、SSIM(p, q) = 1 は、p と q が同一である場合のみ成立します（Wang et al., 2004）。図2は、SSIM指数を構成する3つの類似性関数の異なる認識を示しています。各パッチペア p と q は、ピクセルあたり 0.25 の定数 `² 残差を持ち、したがって3つのケースそれぞれに低い欠陥スコアを割り当てます。一方、SSIM はパッチの平均、分散、共分散の変動に敏感であり、比較関数の一つにおいて各パッチペアに低い類似性を割り当てます。
イメージ x とその再構築 x^ 間の構造的類似性を計算するには、イメージ上に K × K のウィンドウを移動させ、各ピクセル位置で SSIM 値を計算します。式（8）は微分可能であるため、勾配降下法で最適化される深層学習アーキテクチャの損失関数として使用できます。

図 3 は、SSIM が `² などのピクセル単位の誤差関数に比べて欠陥のセグメンテーションにおいて優れていることを示しています。欠陥のないチェッカーボードパターン（さまざまなスケールと方向）で`² -オートエンコーダーを訓練した後、灰色の線と点で欠陥を模擬した画像（図3(a)）に適用します。図3(b)は、オートエンコーダーが生成した対応する再構築画像を示し、入力画像から欠陥を除去しています。残りの2つのサブ図は、

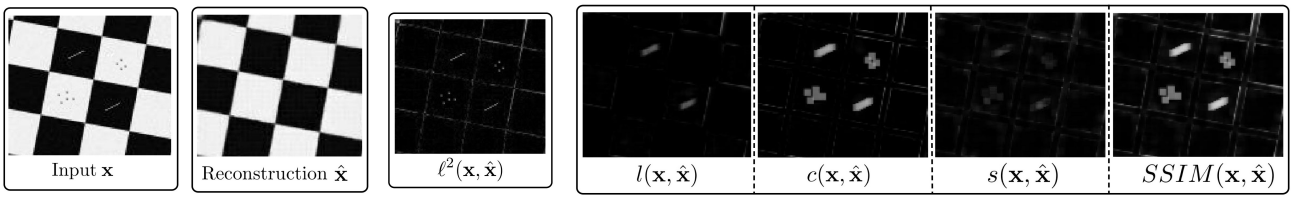

Figure 3: A toy example illustrating the advantages of SSIM over ℓ^2 for the segmentation of defects. **(a)** 128×128 checkerboard pattern with gray strokes and dots that simulate defects. **(b)** Output reconstruction $\hat{\mathbf{x}}$ of the input image $\mathbf{x}$ by an ℓ^2-autoencoder trained on defect-free checkerboard patterns. The defects have been removed by the autoencoder. **(c)** ℓ^2-residual map. Brighter colors indicate larger dissimilarity between input and reconstruction. **(d)** Residuals for luminance l, contrast c, structure s, and their pointwise product that yields the final SSIM residual map. In contrast to the ℓ^2-error map, SSIM gives more importance to the visually more salient disturbances than to the slight inaccuracies around reconstructed edges.

$Q(\mathbf{z}|\mathbf{x})$ that yields a spatial residual map is to decode N latent samples $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N$ drawn from $Q(\mathbf{z}|\mathbf{x})$ and to evaluate the per-pixel reconstruction probability $R_{VAE} = P(\mathbf{x}|\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_N)$ as described by An and Cho (2015).

3.1.3. Feature Matching Autoencoder. Another extension to standard autoencoders was proposed by Dosovitskiy and Brox (2016). It increases the quality of the produced reconstructions by extracting features from both the input image $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}}$ and enforcing them to be equal. Consider $F : \mathbb{R}^{k \times h \times w} \to \mathbb{R}^f$ to be a feature extractor that obtains an f-dimensional feature vector from an input image. Then, a regularizer can be added to the loss function of the autoencoder, yielding the feature matching autoencoder (FM-AE) loss

$$L_{\text{FM}}(\mathbf{x}, \hat{\mathbf{x}}) = L_2(\mathbf{x}, \hat{\mathbf{x}}) + \lambda \|F(\mathbf{x}) - F(\hat{\mathbf{x}})\|_2^2 \ , \qquad (3)$$

where $\lambda > 0$ denotes the weighting factor between the two loss terms. F can be parameterized using the first layers of a CNN pretrained on an image classification task. During evaluation, a residual map R_{FM} is obtained by comparing the per-pixel ℓ^2-distance of $\mathbf{x}$ and $\hat{\mathbf{x}}$. The hope is that sharper, more realistic reconstructions will lead to better residual maps compared to a standard ℓ^2-autoencoder.

3.1.4. SSIM Autoencoder. We show that employing more elaborate architectures such as VAEs or FM-AEs does not yield satisfactory improvements of the residial maps over deterministic ℓ^2-autoencoders in the unsupervised defect segmentation task. They are all based on per-pixel evaluation metrics that assume an unrealistic independence between neighboring pixels. Therefore, they fail to detect structural differences between the inputs and their reconstructions. By adapting the loss and evaluation functions to capture local inter-dependencies between image regions, we are able to drastically improve upon all the aforementioned architectures. In Section 3.2, we specifically motivate the use of the strucutural similarity metric $\text{SSIM}(\mathbf{x}, \hat{\mathbf{x}})$ as both the loss function and the evaluation metric for autoencoders to obtain a residual map R_{SSIM}.

3.2. Structural Similarity

The SSIM index (Wang et al., 2004) defines a distance measure between two $K \times K$ image patches $\mathbf{p}$ and $\mathbf{q}$, taking into account their similarity in luminance $l(\mathbf{p}, \mathbf{q})$, contrast $c(\mathbf{p}, \mathbf{q})$, and structure $s(\mathbf{p}, \mathbf{q})$:

$$\text{SSIM}(\mathbf{p}, \mathbf{q}) = l(\mathbf{p}, \mathbf{q})^\alpha c(\mathbf{p}, \mathbf{q})^\beta s(\mathbf{p}, \mathbf{q})^\gamma \ , \qquad (4)$$

where $\alpha, \beta, \gamma \in \mathbb{R}$ are user-defined constants to weight the three terms. The luminance measure $l(\mathbf{p}, \mathbf{q})$ is estimated by comparing the patches' mean intensities $\mu_\mathbf{p}$ and $\mu_\mathbf{q}$. The contrast measure $c(\mathbf{p}, \mathbf{q})$ is a function of the patch variances $\sigma_\mathbf{p}^2$ and $\sigma_\mathbf{q}^2$. The structure measure $s(\mathbf{p}, \mathbf{q})$ takes into account the covariance $\sigma_{\mathbf{pq}}$ of the two patches. The three measures are defined as:

$$l(\mathbf{p}, \mathbf{q}) = \frac{2\mu_\mathbf{p}\mu_\mathbf{q} + c_1}{\mu_\mathbf{p}^2 + \mu_\mathbf{q}^2 + c_1} \qquad (5)$$

$$c(\mathbf{p}, \mathbf{q}) = \frac{2\sigma_\mathbf{p}\sigma_\mathbf{q} + c_2}{\sigma_\mathbf{p}^2 + \sigma_\mathbf{q}^2 + c_2} \qquad (6)$$

$$s(\mathbf{p}, \mathbf{q}) = \frac{2\sigma_{\mathbf{pq}} + c_2}{2\sigma_\mathbf{p}\sigma_\mathbf{q} + c_2} \ . \qquad (7)$$

The constants c_1 and c_2 ensure numerical stability and are typically set to $c_1 = 0.01$ and $c_2 = 0.03$. By substituting (5)-(7) into (4), the SSIM is given by

$$\text{SSIM}(\mathbf{p}, \mathbf{q}) = \frac{(2\mu_\mathbf{p}\mu_\mathbf{q} + c_1)(2\sigma_{\mathbf{pq}} + c_2)}{(\mu_\mathbf{p}^2 + \mu_\mathbf{q}^2 + c_1)(\sigma_\mathbf{p}^2 + \sigma_\mathbf{q}^2 + c_2)} \ . \qquad (8)$$

It holds that $\text{SSIM}(\mathbf{p}, \mathbf{q}) \in [-1, 1]$. In particular, $\text{SSIM}(\mathbf{p}, \mathbf{q}) = 1$ if and only if $\mathbf{p}$ and $\mathbf{q}$ are identical (Wang et al., 2004). Figure 2 shows the different perceptions of the three similarity functions that form the SSIM index. Each of the patch pairs $\mathbf{p}$ and $\mathbf{q}$ has a constant ℓ^2-residual of 0.25 per pixel and hence assigns low defect scores to each of the three cases. SSIM on the other hand is sensitive to variations in the patches' mean, variance, and covariance in its respective residual map and assigns low similarity to each of the patch pairs in one of the comparison functions.

To compute the structural similarity between an entire image $\mathbf{x}$ and its reconstruction $\hat{\mathbf{x}}$, one slides a $K \times K$ window across the image and computes a SSIM value at each pixel location. Since (8) is differentiable, it can be employed as a loss function in deep learning architectures that are optimized using gradient descent.

Figure 3 indicates the advantages SSIM has over per-pixel error functions such as ℓ^2 for segmenting defects. After training an ℓ^2-autoencoder on defect-free checkerboard patterns of various scales and orientations, we apply it to an image (Figure 3(a)) that contains gray strokes and dots that simulate defects. Figure 3(b) shows the corresponding reconstruction produced by the autoencoder, which removes the defects from the input image. The two remaining subfigures display the residual maps when