



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 $\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N$ drawn from $Q(\mathbf{z}|\mathbf{x})$ and to evaluate the per-pixel reconstruction probability $R_{VAE} = P(\mathbf{x}|\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N)$ 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 $F: \mathbb{R}^{k \times h \times w} \rightarrow \mathbb{R}^f$ 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

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

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 residual 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 structural similarity metric $SSIM(\mathbf{x}, \hat{\mathbf{x}})$ 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 $s(\mathbf{p}, \mathbf{q})$:

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

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:

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

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

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

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

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

It holds that $SSIM(\mathbf{p}, \mathbf{q}) \in [-1, 1]$. In particular, $SSIM(\mathbf{p}, \mathbf{q}) = 1$ 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