A Comprehensive Report on Mathematical, Statistical, and Information-Theoretic Methods for Matrix Comparison in Defect Detection

Section 1: Foundational Methodologies for Matrix Comparison

This section establishes the baseline for matrix comparison, starting with direct methods and progressing to more sophisticated global similarity metrics. These techniques form the building blocks for more advanced local and feature-based analyses.

1.1 Pre-processing and Normalization for Comparison

Before any meaningful comparison can occur, the raw matrix data must be prepared to ensure that detected differences are genuine anomalies and not artifacts of the imaging process. The input data, structured as (x, y, B, G, R), can be processed into one or more 2D matrices for analysis. This typically involves converting the color information into a single grayscale intensity matrix, I(x, y), or analyzing three separate matrices for the red, green, and blue channels: R(x, y), G(x, y), and B(x, y).

A critical pre-processing step is intensity normalization. Variations in lighting, sensor gain, or exposure time can cause two otherwise identical surfaces to produce matrices with different intensity ranges. Comparing these raw matrices would yield spurious differences. To mitigate this, pixel values should be scaled to a consistent range, such as `` or [-1, 1]. This normalization is mathematically essential for the stability and correctness of many comparison algorithms, including matrix norms and the Structural Similarity Index (SSIM), as it prevents results from being skewed by irrelevant global intensity shifts.[1]

1.2 Direct Comparison: Element-wise Subtraction and Boolean Analysis

The most direct method for identifying differences between two matrices is element-wise subtraction. This approach requires that the two matrices, A (reference) and B (test), be of the same dimensions and properly registered (aligned).

Procedure and Mathematical Formulation

Given two normalized matrices A and B of size m x n, the difference matrix D is computed as:

$$D_{ij}=A_{ij}-B_{ij}$$

Each element D_ij represents the change in intensity at the corresponding pixel location (i, j). In an ideal scenario where A is a defect-free reference and B is a test image, non-zero elements in D correspond to potential defects.

To isolate significant differences from background noise, a binary defect mask M can be generated by applying a threshold τ to the absolute values of the difference matrix:

$$M_{ij}=\begin{cases}1 & \text{if } |D_{ij}|>\tau \\ 0 & \text{otherwise}\end{cases}$$

The resulting matrix M highlights regions of potential defects. This method is computationally inexpensive and can be implemented efficiently using basic loops, list comprehensions, or vectorized operations in libraries like NumPy and SciPy.[3] Similar functionalities are available in MATLAB and SymPy for symbolic or numerical comparisons.[5]

Limitations

While simple, element-wise subtraction is highly sensitive to even minor misalignments (registration errors) and random noise. A slight shift of just one pixel can result in the entire image being flagged as different, leading to a high rate of false positives. Therefore, this method is best suited for highly controlled environments where images are perfectly registered and noise levels are minimal.

1.3 Quantifying Global Difference with Matrix Norms

Matrix norms distill the entire difference matrix D into a single scalar value, providing a quantitative measure of the overall dissimilarity between two matrices A and B. This is useful for a quick, global assessment of whether a significant difference exists.

A matrix norm ||D|| is a function that assigns a non-negative real number to a matrix, satisfying specific properties such as positivity, definiteness, absolute homogeneity, and the triangle inequality.[7] For comparative analysis, the following norms are particularly relevant:

- L1-Norm (Maximum Absolute Column Sum): Defined as the largest sum of the absolute values of elements in any single column.
  $$||D||_1 = \max_j \sum_i |D_{ij}|$$
  This norm is highly sensitive to errors that are concentrated vertically. Consequently, it is an effective detector for defects such as vertical scratches, which would produce a column in the difference matrix with a large sum of absolute values.8
- L-infinity Norm (Maximum Absolute Row Sum): Defined as the largest sum of the absolute values of elements in any single row.
  $$||D||_\infty = \max_i \sum_j |D_{ij}|$$
  Analogous to the L1-norm, the L-infinity norm is most sensitive to errors concentrated horizontally, making it well-suited for detecting horizontal scratches or similar linear defects.8
- Frobenius Norm (Euclidean or L2-Norm): Defined as the square root of the sum of the squares of all elements.
  $$||D||_F = \left(\sum_{i=1}^{m} \sum_{j=1}^{n} |D_{ij}|^2\right)^{1/2}$$
  The Frobenius norm is the most widely used matrix norm and is equivalent to the standard Euclidean vector norm applied to the matrix elements as if they were unrolled into a single long vector. It measures the total "energy" of the difference matrix and provides a good general-purpose measure of dissimilarity, sensitive to both large, localized errors and widespread, low-level differences.8

The choice of norm allows for a targeted analysis. If specific defect geometries are expected, selecting the corresponding norm (e.g., L1 for vertical defects) can enhance detection sensitivity.

1.4 Structural Similarity Index (SSIM) and its Components

The Structural Similarity Index (SSIM) offers a more sophisticated approach to image comparison, designed to align better with human perception of quality and similarity. Unlike pixel-based measures like Mean Squared Error (MSE) or matrix norms, SSIM evaluates degradation as a perceived change in structural information.[12]

Core Components of SSIM

SSIM decomposes the comparison of two image windows, x and y, into three distinct components: luminance, contrast, and structure.[12]

1. Luminance Comparison (l): Measures the similarity of the mean pixel intensity.
   $l(x,y)=\frac{2\mu_x\mu_y+C_1}{\mu_x^2+\mu_y^2+C_1}$
2. Contrast Comparison (c): Measures the similarity of the standard deviation (a measure of contrast).
   $c(x,y)=\frac{2\sigma_x\sigma_y+C_2}{\sigma_x^2+\sigma_y^2+C_2}$
3. Structure Comparison (s): Measures the correlation of the two windows after their mean intensities have been subtracted.
   $s(x,y)=\frac{\sigma_{xy}+C_3}{\sigma_x\sigma_y+C_3}$

Here, μ represents the mean, σ the standard deviation, and σ_{xy} the covariance. The constants C_1, C_2, and C_3 are small values added to stabilize the division when denominators are close to zero.

The final SSIM score is a weighted product of these three components:

$SSIM(x,y)=[l(x,y)]^{\alpha}\cdot[c(x,y)]^{\beta}\cdot[s(x,y)]^{\gamma}$

A commonly used simplified version sets $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$.[15] SSIM is typically applied over a sliding window, generating a similarity map (
ssimmap) where each pixel value represents the local similarity. Regions with low SSIM values are indicative of potential defects.[16]

Component-wise Defect Analysis

A single, combined SSIM score indicates the presence of a difference but does not describe its nature. A more powerful application of SSIM for defect detection involves analyzing the three component maps—$l(x,y)$, $c(x,y)$, and $s(x,y)$—independently. This deconstruction allows for a rudimentary, non-machine-learning-based classification of defects.

The rationale is that different types of defects affect the components differently.

- A **scratch or dig** is primarily a disruption of local structure. It would cause a significant drop in the **structure component s(x,y)** while potentially having a smaller impact on local luminance or contrast.
- A **dark blob or stain**, on the other hand, is mainly a change in local brightness and contrast. It would register most strongly as a low value in the **luminance l(x,y) and contrast c(x,y) maps**.

By computing and examining these three separate maps, one can not only locate a defect but also infer its physical characteristics. This provides a much richer analysis than a single dissimilarity score, directly addressing the goal of identifying and understanding different defect types like scratches, digs, and blobs.

Section 2: Strategies for Comparing Matrices of Unequal Dimensions

A critical challenge in this analysis is that the input matrices are not always of the same size. Direct comparison methods like element-wise subtraction and matrix norms are undefined for matrices of different dimensions. This section explores two primary philosophies for overcoming this hurdle: transforming the matrices to make them conformable or using methods that are inherently invariant to size.

2.1 Image Registration and Resampling for Conformability

Image registration is the process of geometrically aligning two or more images into a common coordinate system, a necessary prerequisite for any element-wise comparison.[18] For matrices of different dimensions, this involves resampling one matrix to match the dimensions of the other.

Strategies for Achieving Conformability

- **Padding and Cropping:** These are the simplest approaches. A smaller matrix can be padded with a constant value (e.g., zero or the image mean) to match the size of a larger one. Conversely, a larger matrix can be cropped to match a smaller one. While computationally efficient, padding can introduce artificial edges, and cropping discards potentially valuable information.[21]

- **Resizing via Interpolation:** This is a more sophisticated method that estimates new pixel values when an image is scaled up or down.[22] The choice of interpolation algorithm significantly impacts the quality of the resized image and the preservation of details. Key techniques include:
  - **Nearest-Neighbor Interpolation:** Assigns the value of the single nearest pixel in the source image to the new pixel location. It is the fastest method but tends to produce blocky, pixelated results (aliasing).[22] The formula is effectively
    $I'(x', y') = I(\text{round}(x), \text{round}(y))$.
  - **Bilinear Interpolation:** Calculates the value of a new pixel as a weighted average of the four nearest pixels (a 2x2 neighborhood) in the original image. This produces a smoother output than nearest-neighbor but at the cost of some blurring.[22]
  - **Bicubic Interpolation:** Extends the concept further by using a 4x4 neighborhood of 16 pixels and fitting a cubic spline to them to determine the new pixel value. It yields higher-quality, smoother images but is more computationally intensive.[22]
  - **Lanczos Resampling:** This method uses a windowed sinc function for interpolation, which is theoretically closer to the ideal resampling filter. It excels at preserving details, especially when downscaling, but can introduce "ringing" artifacts (halos) near sharp edges.[22]

The Registration-Artifact Trade-off

While registration is necessary to enable many powerful comparison techniques, it comes with a significant and unavoidable trade-off. The goal is to detect small, subtle defects like faint scratches or digs. These defects are high-frequency spatial details. Interpolation methods, by their very nature of averaging neighboring pixels, act as low-pass filters. This filtering process inherently blurs the image, which can diminish the intensity of a small defect or even erase it completely from the data.

This creates a fundamental dilemma: to use powerful global comparison tools, one must register the images, but the act of registration may destroy the evidence being sought. This implies that a robust defect detection pipeline cannot rely solely on methods that require registration. It must be complemented by techniques that can operate on the original, unaltered matrices, such as feature-based or localized approaches. A multi-pronged strategy is therefore essential.

2.2 Comparison via Fixed-Length Feature Vectors

An alternative approach that circumvents the dimensionality problem is to transform each matrix, regardless of its size, into a fixed-length feature vector. The comparison is then performed between these standardized vectors, not the matrices themselves.[25]

Procedure

1. For each input matrix M_i, compute a feature vector v_i of a predefined length k.
2. Compare any two vectors, v_i and v_j, using standard vector distance metrics such as Euclidean distance, Cosine similarity, or Manhattan distance.

Examples of features that can be used to construct such a vector include:

- A histogram of pixel intensities, using a fixed number of bins.
- A vector composed of global statistical moments (e.g., mean, variance, skewness, kurtosis).
- A vector of texture descriptors, such as properties derived from a Gray-Level Co-occurrence Matrix (GLCM).

The primary limitation of this method is the complete loss of spatial information. While it can detect that two images have different overall statistical properties, it cannot identify the location, shape, or structure of a defect.

2.3 Advanced Statistical Approaches: MDS and DISTATIS

Multidimensional Scaling (MDS) and its generalization, DISTATIS, are advanced techniques from multivariate statistics designed to analyze and compare sets of distance matrices.[28] These methods are particularly relevant when dealing with multiple samples that are expected to follow a similar trend.

The DISTATIS Procedure

DISTATIS is designed to analyze several distance matrices computed on the same set of objects. Its goal is to find a consensus or "best aggregate" of the original matrices.[29] The procedure is as follows:

1. **Feature Identification and Distance Matrix Creation:** Identify a set of I common keypoints or features across all images. For each of the T images, compute an I x I distance matrix D_[t], where the entry (i, j) is the distance between keypoint i and keypoint j.
2. **Cross-Product Transformation:** Transform each distance matrix D_[t] into a cross-product matrix S_[t], which is a standard step in classical MDS.
3. **Compromise Matrix Computation:** Calculate a "compromise" matrix S_c as a weighted average of the individual S_[t] matrices. This compromise matrix represents the best possible consensus of the geometric structure across all the input images.
4. **Analysis and Projection:** The compromise matrix S_c is analyzed using Principal Component Analysis (PCA) to create a low-dimensional consensus space. The original matrices can then be projected onto this space to visualize how much each one deviates from the group consensus.

Establishing a "Golden Standard" from Multiple Samples

The concept of a "compromise" matrix from DISTATIS provides a powerful strategy for this defect detection problem. Instead of relying on a single, potentially imperfect reference image, one can leverage a collection of known "good" (defect-free) images to create a statistically robust "golden standard."

The process would be:

1. Take a set of N known good images {G_1, G_2,..., G_N}.
2. Apply the DISTATIS methodology to this set to compute a single compromise matrix C. This matrix C represents a stable, averaged model of a defect-free fiber optic surface, with noise and minor individual variations filtered out.
3. The defect detection problem then simplifies from a potentially noisy one-to-one comparison (A vs. B) to a more stable one-to-standard comparison (Test Image T vs. Compromise Standard C).
4. Significant deviations of a test image's structure from this golden standard would provide strong evidence of a defect.

This approach transforms the problem from simple difference detection to true anomaly detection against a statistically derived norm, offering significantly greater robustness.

Section 3: Feature-Based Comparison via Statistical and Textural Descriptors

This section transitions from direct matrix comparisons to indirect methods that first characterize image content using statistical and textural features. These features are often more robust to noise and minor geometric distortions than raw pixel values. The comparison is then performed on these descriptive features.

3.1 Statistical Moments for Shape and Intensity Distribution Analysis

Moments are quantitative measures that describe the shape of a distribution. When applied to an image's intensity histogram, they summarize the overall properties of pixel brightness. When applied to the spatial coordinates of pixels, they describe the geometric shape of an object or region.[30]

First to Fourth Order Statistical Moments

These moments are calculated from the normalized intensity histogram $p(z_i)$, where z_i is the intensity level and $p(z_i)$ is its probability.

- **Mean (1st Moment):** The average intensity level, $\mu = \Sigma z_i * p(z_i)$. It provides a basic measure of overall brightness and is sensitive to large, uniformly bright or dark defects like blobs.[33]
- **Variance (2nd Moment):** The spread of intensities around the mean, $\sigma^2 = \Sigma (z_i - \mu)^2 * p(z_i)$. It measures global contrast and can indicate textural complexity.[33]
- **Skewness (3rd Moment):** The asymmetry of the intensity distribution, $\gamma = \Sigma (z_i - \mu)^3 * p(z_i) / \sigma^3$. A symmetric distribution has a skewness of zero. A defect like a dark scratch or a shadow can introduce a "tail" to the histogram, resulting in non-zero skewness.[33]
- **Kurtosis (4th Moment):** The "peakedness" or "tailedness" of the distribution, $\kappa = \Sigma (z_i - \mu)^4 * p(z_i) / \sigma^4$. A high kurtosis (leptokurtic) indicates a distribution with a sharp peak and heavy tails, often caused by outliers. Small, intense defects can act as outliers in the intensity distribution, leading to a high local kurtosis.[33]

Geometric and Central Moments

These moments consider the spatial distribution of pixel intensities.

- **Raw Moments:** Defined as $M_{pq} = \Sigma_x \Sigma_y x^p * y^q * I(x, y)$, where $I(x, y)$ is the intensity at coordinates (x, y). The zeroth moment M_{00} is the total mass (sum of intensities) of the image, and the first moments M_{10} and M_{01} are used to find the centroid $(\bar{x}, \bar{y})$.[37]
- **Central Moments:** Defined as $\mu_{pq} = \Sigma_x \Sigma_y (x - \bar{x})^p * (y - \bar{y})^q * I(x, y)$. By calculating moments about the centroid, they become invariant to translation, meaning an object's central moments are the same regardless of its position in the image.[37]
- **Orientation from Second-Order Moments:** The second-order central moments (μ_{20}, μ_{02}, μ_{11}) can be used to construct a covariance matrix for an image region. The eigenvectors of this matrix correspond to the major and minor axes of the intensity distribution, allowing for the calculation of the region's orientation. This is highly applicable for characterizing elongated defects like scratches.[37]

Hu Moment Invariants for Robust Shape Description

A significant challenge in defect detection is sensitivity to minor geometric variations. A slight rotation or scaling of the camera could cause a perfect surface to appear different from the reference. Hu moment invariants are a set of seven values, derived from second and third-order central moments, that are mathematically constructed to be invariant to translation, scaling, and rotation.[37]

This invariance provides a powerful method for robust comparison. By computing the seven Hu moments for a region in a test image, one obtains a 7-dimensional feature vector that acts as a "fingerprint" of the region's shape. This vector can be compared to the corresponding vector from a reference image. A significant difference between the two vectors would indicate a true structural change—a defect—rather than a simple geometric misalignment. This makes Hu moments an excellent candidate for a feature-based comparison strategy that is resilient to the typical variations in industrial imaging.

3.2 Texture Analysis for Surface Anomaly Detection

Defects on a surface like a fiber optic cable can be fundamentally understood as localized disruptions of an otherwise uniform texture. Therefore, methods from texture analysis are central to this problem.[40]

Gray-Level Co-occurrence Matrix (GLCM)

The GLCM is a powerful statistical tool that characterizes texture by capturing second-order statistics—the spatial relationships between pairs of pixels.

- **Procedure:** A GLCM is a matrix where the entry $P(i, j)$ counts the number of times a pixel with intensity i co-occurs with a pixel with intensity j at a specified distance and orientation.[44]
- **Derived Features:** Comparing entire GLCMs is cumbersome. Instead, a set of scalar features are derived from the matrix to summarize the texture. Key features relevant to defect detection include [44]:
  - **Contrast:** $\Sigma_{i,j} (i-j)^2 * P(i,j)$. Measures the amount of local variation. The $(i-j)^2$ term heavily weights pixel pairs with large intensity differences, making this feature highly sensitive to sharp edges, such as those found in **scratches and digs**.
  - **Homogeneity (Inverse Difference Moment):** $\Sigma_{i,j} P(i,j) / (1 + (i-j)^2)$. Measures the closeness of the distribution of GLCM elements to the diagonal. The weighting factor gives more importance to pairs with similar intensity. Homogeneous regions will have high values, making this feature sensitive to the presence of non-uniform **blobs or stains**.
  - **Energy (Angular Second Moment):** $\Sigma_{i,j} P(i,j)^2$. A measure of textural uniformity. High energy indicates a texture with few, repetitive gray levels.
  - **Correlation:** $\Sigma_{i,j} (i-\mu_i)(j-\mu_j)P(i,j) / (\sigma_i\sigma_j)$. Measures the linear dependency of gray levels. It indicates the presence of linear structures in the texture.

By computing these GLCM features within a sliding window across an image, one can generate feature maps. Comparing these maps to those of a defect-free reference allows for the localization and characterization of texture anomalies, i.e., defects.[48]

Other Texture Descriptors

- **Local Binary Patterns (LBP):** A computationally efficient and powerful texture descriptor. It works by thresholding the neighbors of each pixel and encoding the result as a binary number. A histogram of these LBP codes serves as the texture feature vector. LBP is robust to monotonic illumination changes.[40]
- **Gabor Filters:** A bank of Gabor filters, which are oriented, frequency-selective filters, can be convolved with an image to analyze its texture. The energy of the filter responses at various orientations and frequencies forms a feature vector

that effectively captures textural information.[33]

Section 4: Frequency, Scale, and Structural Component Analysis

This section delves into methods that transform the image matrices into alternative domains—such as frequency, scale, or component space—where differences corresponding to defects can be more easily isolated and characterized.

4.1 2D Fourier Transform for Spectral Anomaly Detection

The 2D Fast Fourier Transform (FFT) is a fundamental tool that decomposes an image into its constituent sine and cosine waves of varying frequencies and orientations.[61] The result is a frequency-domain representation where the center corresponds to low frequencies (overall structure) and the periphery corresponds to high frequencies (fine details, edges, noise).

Procedure and Application

1. **Transformation:** Compute the 2D FFT of the reference image A and the test image B.
2. **Magnitude Comparison:** The power spectrum, |FFT(A)|^2, shows the energy at each spatial frequency. A regular, periodic texture like that of a fiber optic surface will produce distinct, sharp peaks in the spectrum. Defects disrupt this regularity:
   - **Scratches and Digs:** These are sharp, high-frequency features that will introduce a broad spread of energy across the high-frequency regions of the spectrum.
   - Blobs and Stains: These are typically smooth, low-frequency features that will alter the energy distribution near the center of the spectrum.
     Comparing the power spectra of A and B can reveal these anomalous frequency components.63
3. **Phase Correlation:** This technique is used for registration. The cross-power spectrum is computed, and its inverse FFT reveals a peak whose location corresponds to the translational shift between the two images.

The Critical Role of Phase Information

While many applications of FFT for image processing focus on filtering the magnitude spectrum, this approach discards critical information. The magnitude spectrum indicates *which* frequencies are present, but the **phase spectrum** encodes *where* these frequencies are located in the image.[65] Since defects are localized spatial events, their primary impact is on the phase information. A comprehensive comparison using FFT must therefore analyze not just the magnitude but also the phase. Comparing the full complex-valued FFT results or analyzing the phase spectra separately provides a more complete method for detecting and localizing structural differences.

4.2 Wavelet Transform for Multi-Resolution Defect Identification

The Wavelet Transform (WT) overcomes a key limitation of the FFT by providing simultaneous frequency and spatial localization. This makes it an exceptionally powerful tool for analyzing non-stationary signals like images containing localized defects.[67]

Procedure and Defect Detection

A 2D Discrete Wavelet Transform (DWT) recursively decomposes an image into a set of coefficients at multiple scales.[71] At each level of decomposition, four sub-bands are produced:

- **LL (Approximation):** A down-sampled, low-pass filtered version of the image, capturing its coarse structure.
- **LH (Horizontal Detail):** A high-pass filtered version that highlights vertical edges.
- **HL (Vertical Detail):** A high-pass filtered version that highlights horizontal edges.
- **HH (Diagonal Detail):** A high-pass filtered version that highlights diagonal edges.

Defects are typically high-frequency anomalies and thus manifest as coefficients with large magnitudes in the detail sub-bands (LH, HL, HH).[67] This property allows for both detection and characterization:

- **Detection:** By comparing the detail coefficients of a test image to a reference, or simply by thresholding the coefficients, defects can be isolated.

- **Characterization:** The orientation of a defect can be inferred from the sub-band in which it has the most energy. A vertical scratch will produce strong coefficients in the HL (horizontal detail) sub-band, a horizontal scratch in the LH sub-band, and a small dig or blob in the HH sub-band.

The choice of the "mother wavelet" is crucial. Simple wavelets like **Haar** are effective at detecting sharp, step-like discontinuities. Smoother wavelets like **Daubechies (dbN)** or **Symlets (symN)** may be more suitable for analyzing more complex textures and avoiding artifacts.[71]

4.3 Singular Value Decomposition (SVD) for Structural Change Detection

Singular Value Decomposition (SVD) is a powerful matrix factorization technique that decomposes any matrix A into the product of three other matrices: $A = U\Sigma V^T$. Here, U and V are orthogonal matrices representing rotation and reflection, and Σ is a diagonal matrix containing the non-negative singular values of A in descending order.[76]

Procedure for Comparison

The vector of singular values, $\sigma = [\sigma_1, \sigma_2, \ldots, \sigma_r]$, serves as a compact and rotation-invariant signature of the matrix's structure. A simple and robust comparison method involves:

1. Computing the SVD for a reference matrix A and a test matrix B to obtain their singular value vectors, σ_A and σ_B.
2. Calculating the distance between these two signature vectors using a standard vector norm, such as $||\sigma_A - \sigma_B||_2$. A large distance indicates a significant structural difference between the two matrices.

Defect Characterization via the Singular Value Spectrum

The utility of SVD extends beyond a single dissimilarity score. The magnitude of the singular values directly corresponds to their importance in representing the image's structure.[78]

- The **largest singular values** correspond to the principal components of the image, capturing its dominant, large-scale patterns and energy.
- The **smallest singular values** correspond to the finer details, subtle variations,

and noise within the image.

This property allows for a more nuanced analysis of defects based on their scale.

- A **large-scale defect**, such as a widespread stain, a significant change in texture pattern, or a large blob, will perturb the fundamental structure of the image. This will cause a noticeable change in the largest singular values.
- A **small, localized defect**, such as a tiny scratch, dig, or isolated pixel anomaly, constitutes a fine detail. Its presence will primarily affect the smallest singular values.

Therefore, instead of comparing the entire singular value vectors, a partitioned comparison can be performed. For example, one could compute the difference norm for the top 10% of singular values and separately for the bottom 50%. A large difference in the former suggests a large-scale anomaly, while a large difference in the latter points to a small, localized defect. This approach provides a powerful, non-spatial method to infer the scale and nature of the detected anomaly.

Section 5: Information-Theoretic Measures for Distributional Comparison

This section reframes the matrix comparison problem from a geometric or structural perspective to a probabilistic one. Here, the pixel intensity values within an image or a region are treated as samples drawn from a probability distribution. Defects are then identified as significant deviations between the probability distribution of a test image and that of a reference. This approach is particularly powerful for detecting changes in texture, color, and overall image statistics.

5.1 Measuring Image Complexity with Shannon Entropy

Shannon entropy is a fundamental concept from information theory that quantifies the uncertainty, randomness, or complexity of a system. When applied to an image's intensity histogram $p(z_i)$, the entropy is calculated as:

$H = -\sum_i p(z_i)\log_2(p(z_i))$
where p(z_i) is the probability of a pixel having intensity level i.[81]

Application to Anomaly Detection

A simple, uniform texture (e.g., a perfectly smooth surface) has a very narrow histogram and thus low entropy. A complex, random-looking texture will have a wider histogram and higher entropy. A defect introduces new, unexpected pixel values into a region, thereby increasing the complexity and entropy of its local histogram.[82] This principle can be operationalized by computing entropy in a sliding window across the image to create an "entropy map." Regions with anomalously high entropy can be flagged as potential defects.[84]

Normalized vs. Absolute Entropy

A critical consideration is that the absolute value of Shannon entropy depends on the number of possible outcomes—in this case, the number of gray levels or histogram bins. An image quantized to 256 levels has a higher maximum possible entropy ($\log_2(256) = 8$) than one quantized to 64 levels ($\log_2(64) = 6$). This makes direct comparison of entropy values between images with different bit depths or quantization schemes misleading.[84]

To create a robust and comparable measure, **normalized Shannon entropy** should be used. This is calculated by dividing the computed entropy H by the maximum possible entropy $H_{max} = \log_2(N)$, where N is the number of quantization levels. The resulting value, between 0 and 1, represents the complexity relative to the maximum possible randomness, making it a stable metric that is independent of the specific quantization choices.[85]

5.2 Comparing Intensity Distributions with Histogram Divergence Measures

Divergence measures provide a formal way to quantify the "distance" or difference between two probability distributions, P and Q. In this context, P and Q are the normalized intensity histograms of two images or image regions.

- Kullback-Leibler (KL) Divergence: Also known as relative entropy, KL divergence measures the information lost when distribution Q is used to approximate the true distribution P.

$$D_{KL}(P||Q)=\sum_x P(x)\log\left(\frac{P(x)}{Q(x)}\right)$$

KL divergence is asymmetric, meaning D_KL(P||Q) ≠ D_KL(Q||P), and can be infinite if Q(x) = 0 for any x where P(x) > 0. This makes it somewhat fragile for direct comparison but useful for measuring how well a reference model Q describes a test sample P.87

- Jensen-Shannon (JS) Divergence: JS divergence is a symmetrized and smoothed version of KL divergence, which resolves its main practical issues.

$$JSD(P||Q)=\frac{1}{2}D_{KL}(P||M)+\frac{1}{2}D_{KL}(Q||M)$$

where M = (P + Q) / 2 is the average distribution.90 JSD is symmetric, always has a finite value, and its square root is a true mathematical metric, making it a much more robust and reliable choice for quantifying the dissimilarity between two histograms.91 It is an excellent candidate for defect detection, where a stable, bounded score is highly desirable.92

5.3 Advanced Histogram Distance Metrics

Beyond KL and JS divergence, other metrics offer different sensitivities.

- Chi-Squared Distance: A classical statistical test for comparing observed frequencies to expected ones. The formula for comparing two histograms P and Q is often given as:

$$\chi^2(P,Q)=\frac{1}{2}\sum_i \frac{(P(i)-Q(i))^2}{P(i)+Q(i)}$$

This metric is particularly sensitive to differences in bins with low counts, making it useful for detecting rare events or subtle changes in the tails of a distribution.98

- **Earth Mover's Distance (EMD) / Wasserstein Metric:** EMD represents a fundamentally different approach to histogram comparison. Instead of comparing bins independently, it measures the minimum "work" required to transform one distribution into the other, where "work" is the amount of "mass" moved multiplied by the "ground distance" between the bins.[101]

This "cross-bin" comparison makes EMD uniquely powerful. Consider a defect that causes a slight darkening, shifting pixel values from an average of 150 to 140.

- Bin-wise metrics like Chi-Squared or JSD would see this as a complete change: mass disappears from bin 150 and appears in bin 140. They do not recognize that these bins are "close" to each other.

- EMD, however, uses the ground distance |150 - 140| = 10. The work required to move the mass is small, reflecting the subtle nature of the change. If the shift were to an intensity of 50, the ground distance would be |150 - 50| = 100, and the EMD would be much larger.

This property makes EMD highly sensitive to the overall *shape* of the distribution and perceptually more relevant for detecting defects like stains, subtle color shifts, or gradual intensity changes that other metrics might misinterpret.[103] The power of the Wasserstein metric in capturing structural information is further evidenced by its successful use as a loss function in autoencoders to improve image reconstruction quality.[106]

Section 6: Advanced Methodologies for Local and Topological Defect Detection

This section introduces cutting-edge methodologies that are particularly well-suited for identifying the small, localized, and structurally complex defects relevant to this analysis. These techniques move beyond global comparisons to pinpoint anomalies with high precision.

6.1 Localized Analysis via Sliding Window Techniques

The sliding window technique is a powerful and versatile framework that bridges the gap between global and local analysis. It operates by applying a comparison metric within a small, localized window that is systematically moved across the image, allowing for the creation of a defect map.[108]

Procedure

1. **Define Window and Step Size:** A window of a fixed size (e.g., 32x32 pixels) and a step size (the number of pixels the window moves at each iteration) are defined. Smaller step sizes lead to higher resolution but increased computation.
2. **Iterate Across Images:** The window is slid across the reference image A and the

test image B in tandem.
3. **Local Comparison:** At each position i, the sub-matrices a_i and b_i within the window are extracted. A dissimilarity score s_i = metric(a_i, b_i) is computed.
4. **Construct Defect Map:** The resulting scores s_i are assembled into a map where the value at each location corresponds to the local dissimilarity. High values on this map indicate potential defect locations.

This framework is highly flexible, as the metric used within the window can be any of the methods discussed previously, including:

- **Statistical Metrics:** Comparing the mean, variance, skewness, or kurtosis of the pixel distributions within the windows.[111]
- **Texture Features:** Comparing vectors of GLCM properties (e.g., contrast, homogeneity) or LBP histograms.
- **Information-Theoretic Measures:** Calculating the JS Divergence or Earth Mover's Distance between the local histograms.

This approach is one of the most practical and effective for this problem, as it directly addresses the need to find and localize small defects.[108]

6.2 Non-Local Means (NLM) for Robust Patch-Based Comparison

Non-Local Means (NLM) is a sophisticated algorithm, originally for denoising, that operates on the principle of image redundancy. It assumes that for any given patch in a natural image, many similar patches exist elsewhere in the image.[118] A pixel's value is reconstructed by a weighted average of all other pixels, where the weights are determined by the similarity of their surrounding patches.

Mathematical Formulation

The restored value NL[I](p) for a pixel p in an image I is:

$$NL[I](p)=C(p)1q\in I\Sigma w(p,q)I(q)$$

where C(p) is a normalization constant and the weight w(p, q) is a function of the similarity between the patch centered at p and the patch centered at q. A common weighting function is:

$$w(p,q)=\exp(-h2\mid\mid Patch(p)-Patch(q)\mid\mid 22)$$

Here, h is a filtering parameter that controls the degree of smoothing.

Application to Anomaly Detection

A defect is, by definition, an anomalous, non-redundant pattern. A patch containing a defect will be dissimilar to the vast majority of other patches in a normal image. This property can be leveraged for comparison.[119]

1. Take a reference (defect-free) image A and a test image B.
2. For a patch P_B at some location in the test image B, its NLM reconstruction P'_B is computed by averaging patches from the *reference image A*. The weights are based on the similarity of patches in A to the test patch P_B.
3. The anomaly score for that location is the reconstruction error: ||P_B - P'_B||_2.
4. If P_B is a normal, defect-free patch, it will be similar to many patches in A, and the reconstruction error will be low. If P_B contains a defect, it will be dissimilar to the normal patches in A, resulting in a poor reconstruction and a high error score. This creates a powerful anomaly detector.

6.3 Topological Data Analysis: Detecting Defects with Persistent Homology

Persistent Homology (PH) is a cutting-edge method from the field of topological data analysis (TDA) that provides a way to quantify the "shape" of data. It analyzes data by tracking the evolution of topological features—specifically connected components (0D holes), loops (1D holes), and voids (2D holes)—across multiple scales or resolutions.[122] This provides a fundamentally different and mathematically intense approach to defect detection.

Procedure for Image Analysis

1. **Point Cloud Creation:** The image is first converted into a point cloud. This can be done by thresholding the image and using the coordinates of pixels above the threshold, or by selecting "landmark" pixels based on texture features like LBP.[59]
2. **Filtration Construction:** A sequence of nested simplicial complexes (a generalization of networks with points, edges, triangles, etc.) is built from the point cloud. This is done by defining a distance parameter ε and connecting any two points that are within distance ε of each other. As ε increases, more connections are formed, and the complex "grows".[122]
3. **Persistent Homology Computation:** The algorithm tracks the "birth" and

"death" of topological features as ε increases.
  - A **connected component (Betti-0)** is "born" when a point appears. It "dies" when it merges with another component.
  - A **loop (Betti-1)** is "born" when a set of edges connects to form a cycle. It "dies" when the interior of that loop is filled in by triangles.
4. **Persistence Diagram Visualization:** The output is a persistence diagram, a 2D scatter plot where each point (birth_ε, death_ε) represents a single topological feature. The persistence of a feature is death_ε - birth_ε. Features with high persistence (far from the diagonal line y=x) are considered robust topological features of the data, while those with low persistence (close to the diagonal) are considered topological "noise".[59]

Defects as Topological Anomalies

This framework provides a novel way to define and detect defects by treating them as anomalies in the image's topology.

- A defect-free fiber optic surface should have a relatively simple and predictable topology. Its persistence diagram would be "clean," likely showing one highly persistent connected component and very little else.
- A **scratch** is a thin, linear feature. Topologically, it can either break a single connected component into two, or it can form a long, persistent **1D loop (Betti-1 feature)** that does not exist in the reference image.[128]
- A **dig or blob** is a distinct, isolated region. This will appear as a new, persistent **0D connected component (Betti-0 feature)** that is born and dies at different ε values than the main surface component.[123]

By computing the persistence diagram for a local region and comparing it to the diagram of a "normal" region (using metrics like the bottleneck or Wasserstein distance between diagrams), one can robustly detect defects. This method is powerful because it is concerned with the abstract connectivity and shape of features, rather than their precise pixel values, making it potentially resilient to noise and illumination changes.[123]

Section 7: Synthesis and Recommendations for Implementation

This report has detailed an exhaustive array of classical mathematical, statistical, and information-theoretic methods for comparing matrices, with a specific focus on detecting defects in fiber optic imagery. No single method is universally superior; the optimal approach depends on the specific defect types, computational constraints, and desired level of robustness. This section synthesizes these findings into a practical framework and proposes a multi-stage pipeline for implementation.

7.1 A Comparative Framework for Method Selection

The following table provides a comprehensive comparison of the discussed techniques, designed to aid in the selection of an appropriate combination of methods for a defect detection system.

Method	Category	Primary Defect Sensitivity	Handles Unequal Sizes?	Computational Complexity	Robustness to Noise/Alignment	Mathematical Rigor
Element-wise Difference	Global/Direct	General, high-frequency changes	No	Low (O(N))	Low	Low
Frobenius Norm	Global/Direct	General difference energy	No	Low (O(N))	Low	Medium
L1 / L-inf Norm	Global/Direct	Vertical / Horizontal streaks	No	Low (O(N))	Low	Medium
SSIM (Global)	Global/Perceptual	Structural, luminance, contrast	No	Medium (O(N))	Medium	Medium
SSIM (Component Maps)	Local/Perceptual	Characterizes scratches, blobs	Requires Registration	Medium (O(N))	Medium	High
Hu	Global/Fe	Shape,	Yes	Low	High	High

Moment Invariants	ature	structure		(O(N))		
GLCM Features (Contrast)	Local/Texture	Edges, scratches, digs	Yes (in window)	Medium	Medium	High
GLCM Features (Homogeneity)	Local/Texture	Blobs, stains, smooth areas	Yes (in window)	Medium	Medium	High
2D Fourier Transform (FFT)	Transform	Periodic patterns, global texture	No	Medium (O(N log N))	Medium	High
Wavelet Transform (DWT)	Transform/Local	Localized edges, scratches, blobs	Yes	Medium (O(N))	High	High
Singular Value Decomposition (SVD)	Transform/Global	Global structure, fine details	No	High (O(N^3))	High	High
Jensen-Shannon Divergence	Info-Theoretic	Intensity/Color distribution	Yes (on histograms)	Low (on histograms)	Medium	High
Earth Mover's Distance (EMD)	Info-Theoretic	Subtle shifts in distribution, stains	Yes (on histograms)	High	High	High
Non-Local Means (NLM)	Local/Patch-based	Non-redundant patches, anomalies	Yes (in window)	Very High (O(N^2))	Very High	High
Persistent Homolog	Topological	Holes (scratches),	Yes (on point clouds)	High	Very High	Very High

y (PH)		componen ts (blobs)				

7.2 Recommended Procedures for a Fiber Optic Defect Detection Pipeline

A robust and efficient defect detection system should employ a multi-stage pipeline that balances computational cost with analytical depth. The following tiered approach is recommended:

1. **Stage 1: Global Anomaly Screening (Fast Filter)**
   - **Objective:** To quickly identify images with gross defects or significant deviations from the norm without performing expensive pixel-level comparisons.
   - **Procedure:**
     - For each incoming image matrix, do not perform registration.
     - Extract a fixed-length global feature vector. This vector should include **global statistical moments** (mean, variance, skewness, kurtosis of the entire intensity histogram) and the seven **Hu moment invariants**.
     - Compare this feature vector to a pre-computed "golden standard" vector derived by averaging the feature vectors from a large set of known good samples. The Mahalanobis distance is recommended for this comparison as it accounts for correlations between features.
     - If the distance exceeds a high, conservative threshold, flag the image as potentially defective and pass it to the next stage. This step efficiently filters out obvious failures.
2. **Stage 2: Localized Defect Mapping (Primary Detection)**
   - **Objective:** To precisely locate potential defects on the images that pass the initial screening.
   - **Procedure:**
     - Employ a **sliding window** approach on the original, non-registered image to preserve fine details.
     - Within each window, compute a rich local feature vector. A powerful combination would be:
       - **GLCM Features:** Specifically **Contrast** (for edges/scratches) and **Homogeneity** (for blobs/stains).
       - **Wavelet Transform Features:** The energy of the detail coefficients (LH, HL, HH) from a 1- or 2-level DWT. This captures oriented details

that GLCM might miss.

        - Compare this local feature vector to an average feature vector from corresponding locations in the set of "good" images (the golden standard).
        - The resulting map of dissimilarity scores will highlight the precise locations of potential defects.
3. **Stage 3: Defect Characterization and Confirmation (Intense Analysis)**
    - **Objective:** To analyze the small regions flagged in Stage 2 with the most computationally intensive and rigorous methods to confirm the presence of a defect and classify its type.
    - **Procedure:**
        - For each flagged region of interest (ROI):
            - **Persistent Homology:** Convert the ROI into a point cloud based on its intensity values. Compute its persistence diagram. A defect-free region should have a simple topological signature (e.g., one persistent 0D feature). The appearance of new, persistent 0D features (indicating isolated **blobs/digs**) or 1D features (indicating **scratches/holes**) provides strong, mathematically rigorous confirmation of a defect. The characteristics of the persistence diagram can be used to classify the defect type.
            - **SSIM Component Analysis:** If a higher-resolution reference is available, register the ROI from the test image to the corresponding ROI from the reference. Compute the individual luminance (l), contrast (c), and structure (s) maps. Analysis of which component shows the largest deviation provides further evidence for classification (e.g., low structure suggests a scratch; low luminance suggests a dark blob).

This tiered strategy effectively leverages the strengths of multiple mathematical frameworks. It uses computationally inexpensive methods for broad screening and reserves the most powerful and "intense" techniques, such as Persistent Homology, for targeted analysis where they can provide the most value. This creates a pipeline that is not only robust and accurate but also computationally feasible for a real-world application.

Works cited

1. Estimating the Fundamental Matrix, accessed June 17, 2025, https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/MOHR_TRIGGS/node50.html
2. Utilizing the scale-invariant feature transform algorithm to align ..., accessed June

17, 2025, https://academic.oup.com/bioinformatics/article/40/3/btae064/7601449
3. Python – Element wise Matrix Difference | GeeksforGeeks, accessed June 17, 2025, https://www.geeksforgeeks.org/python-element-wise-matrix-difference/
4. Element-wise comparison of SymPy Matrices - python - Stack Overflow, accessed June 17, 2025, https://stackoverflow.com/questions/77929603/element-wise-comparison-of-sympy-matrices
5. Array Comparison with Relational Operators - MATLAB & Simulink - MathWorks, accessed June 17, 2025, https://www.mathworks.com/help/matlab/matlab_prog/array-comparison-with-relational-operators.html
6. Fastest way to compare two matrices element by element - Mathematica Stack Exchange, accessed June 17, 2025, https://mathematica.stackexchange.com/questions/163754/fastest-way-to-compare-two-matrices-element-by-element
7. Matrix norms, accessed June 17, 2025, https://nucinkis-lab.cc.ic.ac.uk/HELM/workbooks/workbook_30/30_4_matrx_norms.pdf
8. Matrix norm - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Matrix_norm
9. norm - Vector and matrix norms - MATLAB - MathWorks, accessed June 17, 2025, https://www.mathworks.com/help/matlab/ref/norm.html
10. MATRIX NORMS – Linear Algebra and Applications - Pressbooks.pub, accessed June 17, 2025, https://pressbooks.pub/linearalgebraandapplications/chapter/matrix-norms/
11. What is the difference between the Frobenius norm and the 2-norm of a matrix? - Mathematics Stack Exchange, accessed June 17, 2025, https://math.stackexchange.com/questions/33083/what-is-the-difference-between-the-frobenius-norm-and-the-2-norm-of-a-matrix
12. Structural Similarity Index - NI, accessed June 17, 2025, https://www.ni.com/docs/en-US/bundle/ni-vision-concepts-help/page/structural_similarity_index.html
13. Structural similarity index measure - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Structural_similarity_index_measure
14. [2006.13846] Understanding SSIM - ar5iv - arXiv, accessed June 17, 2025, https://ar5iv.labs.arxiv.org/html/2006.13846
15. Max and min values of the structural similarity (SSIM) function S(x, a) on the L2 sphere SR - University of Waterloo, accessed June 17, 2025, https://www.math.uwaterloo.ca/~ervrscay/talks/post206_iciar12.pdf
16. Structural similarity index — skimage 0.25.2 documentation, accessed June 17, 2025, https://scikit-image.org/docs/0.25.x/auto_examples/transform/plot_ssim.html
17. ssim - Structural similarity (SSIM) index for measuring image quality - MATLAB - MathWorks, accessed June 17, 2025, https://www.mathworks.com/help/images/ref/ssim.html

18. Global image registration using a symmetric block-matching approach - PMC, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC4478989/
19. Image Registration Techniques, accessed June 17, 2025, https://engrxiv.org/preprint/download/127/299
20. An Iterative Image Registration Technique Using a Scale-Space Model - Texas A&M University, accessed June 17, 2025, https://psi.engr.tamu.edu/wp-content/uploads/2018/01/lee2011iterative.pdf
21. mrgrid — MRtrix3 3.0 documentation, accessed June 17, 2025, https://mrtrix.readthedocs.io/en/dev/reference/commands/mrgrid.html
22. What is Image Resizing? A Computer Vision Guide. - Roboflow Blog, accessed June 17, 2025, https://blog.roboflow.com/image-resizing/
23. Comparative Analysis of Different Interpolation Schemes In Image Processing, accessed June 17, 2025, https://www.researchgate.net/publication/303940746_Comparative_Analysis_of_Different_Interpolation_Schemes_In_Image_Processing
24. Comparison gallery of image scaling algorithms - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Comparison_gallery_of_image_scaling_algorithms
25. Feature extraction from matrices with unequal size - Cross Validated - Stack Exchange, accessed June 17, 2025, https://stats.stackexchange.com/questions/323788/feature-extraction-from-matrices-with-unequal-size
26. How to compare elements in multiple vectors (of different sizes) against each other?, accessed June 17, 2025, https://stackoverflow.com/questions/49382863/how-to-compare-elements-in-multiple-vectors-of-different-sizes-against-each-ot
27. Vector DB feature comparison matrix : r/mlops - Reddit, accessed June 17, 2025, https://www.reddit.com/r/mlops/comments/197z0p1/vector_db_feature_comparison_matrix/
28. Multidimensional scaling - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Multidimensional_scaling
29. DISTATIS How to analyze multiple distance matrices - The University of Texas at Dallas, accessed June 17, 2025, https://personal.utdallas.edu/~herve/Abdi-DISTATIS2007-pretty.pdf
30. Moments in Statistics: A Comprehensive Guide - Shiksha Online, accessed June 17, 2025, https://www.shiksha.com/online-courses/articles/moments-in-statistics/
31. Moment (mathematics) - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Moment_(mathematics)
32. Image Statistics, accessed June 17, 2025, https://www.csee.umbc.edu/~caban1/Fall2010/CMSC691/Schedule_files/Docs/08-ImageStatistics.pdf
33. Lecture (4): Texture features and analysis (part one), accessed June 17, 2025, https://www.uomustansiriyah.edu.iq/media/lectures/5/5_2024_11_20!08_06_26_PM.pdf
34. image operations, skewness and kurtosis - MATLAB Answers - MathWorks,

accessed June 17, 2025, https://www.mathworks.com/matlabcentral/answers/15307-image-operations-skewness-and-kurtosis
35. Difference Between Skewness and Kurtosis - Analytics Vidhya, accessed June 17, 2025, https://www.analyticsvidhya.com/blog/2021/05/shape-of-data-skewness-and-kurtosis/
36. Skewness and Kurtosis — Explanation in detail along with Cheat-sheet... | Towards AI, accessed June 17, 2025, https://towardsai.net/p/l/skewness-and-kurtosis%E2%80%8A-%E2%80%8Aexplanation-in-detail-along-with-cheat-sheet
37. Image moment - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Image_moment
38. Tracking object orientation with image moments, accessed June 17, 2025, https://raphael.candelier.fr/?blog=Image%20Moments
39. Image Moments - Breadboard Gremlins - WordPress.com, accessed June 17, 2025, https://breadboardgremlins.wordpress.com/image-moments/
40. Texture - Washington, accessed June 17, 2025, https://courses.cs.washington.edu/courses/cse576/book/ch7.pdf
41. Image texture - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Image_texture
42. Texture analysis | Images as Data Class Notes - Fiveable, accessed June 17, 2025, https://library.fiveable.me/images-as-data/unit-4/texture-analysis/study-guide/6rcT9p8U0KbNBwq9
43. Texture Analysis, accessed June 17, 2025, http://www.cyto.purdue.edu/cdroms/micro2/content/education/wirth06.pdf
44. Texture Analysis Using Gray-Level Co-Occurrence Matrix - MATLAB & - MathWorks, accessed June 17, 2025, https://www.mathworks.com/help/images/texture-analysis-using-the-gray-level-co-occurrence-matrix-glcm.html
45. Gray level co-occurence matrix · ImageFeatures - JuliaImages, accessed June 17, 2025, https://juliaimages.org/ImageFeatures.jl/stable/tutorials/glcm/
46. GLCM Texture: A Tutorial - lapi - UNAM, accessed June 17, 2025, https://lapi.fi-p.unam.mx/wp-content/uploads/Tutorial-GLCM.pdf
47. Building Change Detection Based on a Gray-Level Co-Occurrence Matrix and Artificial Neural Networks - MDPI, accessed June 17, 2025, https://www.mdpi.com/2504-446X/6/12/414
48. Textural Fabric Defect Detection using Adaptive Quantized Gray-level Co-occurrence Matrix and Support Vector Description Data - Science Alert, accessed June 17, 2025, https://scialert.net/fulltext/?doi=itj.2012.673.685
49. (PDF) Crowd Abnormality Detection Using Optical Flow and GLCM ..., accessed June 17, 2025, https://www.researchgate.net/publication/356034556_Crowd_Abnormality_Detection_Using_Optical_Flow_and_GLCM-Based_Texture_Features
50. Spatial assessments in texture analysis: what the radiologist needs to know -

PMC, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC10484588/
51. Fabric Defect Classification Based on LBP and GLCM - Global Science Press, accessed June 17, 2025, https://global-sci.com/article/86533/fabric-defect-classification-based-on-lbp-and-glcm
52. An Automated Surface Defect Inspection System Using Local Binary Patterns and Co-Occurrence Matrix Textures based on SVM Classifier - Jordan Journal of Electrical Engineering, accessed June 17, 2025, https://jjee.ttu.edu.jo/3644/
53. Fabric Defect Detection based on GLCM and Gabor filter: A Comparison - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/236732989_Fabric_Defect_Detection_based_on_GLCM_and_Gabor_filter_A_Comparison
54. Apple Surface Defect Detection Based on Gray Level Co-Occurrence Matrix and Retinex Image Enhancement - MDPI, accessed June 17, 2025, https://www.mdpi.com/2076-3417/13/22/12481
55. Automatic Fabric Inspection using GLCM-based Jensen-Shannon Divergence - Informatica, An International Journal of Computing and Informatics, accessed June 17, 2025, https://www.informatica.si/index.php/informatica/article/download/3015/1721
56. Surface Defect Detection Methods for Industrial Products: A Review - MDPI, accessed June 17, 2025, https://www.mdpi.com/2076-3417/11/16/7657
57. Image Classification Without Neural Networks: A Practical Guide - DigitalOcean, accessed June 17, 2025, https://www.digitalocean.com/community/tutorials/image-classification-without-neural-networks
58. Defect Detection in Tire X-Ray Images: Conventional Methods Meet Deep Structures - arXiv, accessed June 17, 2025, https://arxiv.org/html/2402.18527v1
59. (PDF) Persistent Homology for Image Analysis - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/341136357_Persistent_Homology_for_Image_Analysis
60. Texture Analysis - MATLAB & Simulink - MathWorks, accessed June 17, 2025, https://www.mathworks.com/help/images/texture-analysis-1.html
61. Image Processing with Fourier Transform - YouTube, accessed June 17, 2025, https://www.youtube.com/watch?v=oACegp4iGi0
62. Fourier Transform - Gwyddion, accessed June 17, 2025, http://gwyddion.net/documentation/user-guide-en/fourier-transform.html
63. Understanding fourier transform layers. : r/GIMP - Reddit, accessed June 17, 2025, https://www.reddit.com/r/GIMP/comments/1aplm07/understanding_fourier_transform_layers/
64. Image Processing with Python - Application of Fourier Transformation, accessed June 17, 2025, https://towardsdatascience.com/image-processing-with-python-application-of-fourier-transformation-5a8584dc175b/

65. Introduction to Image Processing with 2D Fourier Transform - YouTube, accessed June 17, 2025, https://www.youtube.com/watch?v=tlwIWjeuu8U
66. 2D Fourier Transform Explained with Examples - YouTube, accessed June 17, 2025, https://m.youtube.com/watch?v=3gAZ0U66AEA
67. Image based defect detection Algorithm by use of wavelet ..., accessed June 17, 2025, https://www.researchgate.net/publication/290543003_Image_based_defect_detection_Algorithm_by_use_of_wavelet_transformation
68. (PDF) A Review of Wavelet Analysis and Its Applications: Challenges and Opportunities, accessed June 17, 2025, https://www.researchgate.net/publication/361020478_A_Review_of_Wavelet_Analysis_and_Its_Applications_Challenges_and_Opportunities
69. (PDF) Wavelet transform for image analysis - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/4092523_Wavelet_transform_for_image_analysis
70. A Tutorial of the Wavelet Transform - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/265633800_A_Tutorial_of_the_Wavelet_Transform
71. Fabric Defect Detection using Discrete Wavelet Transform - IRJET, accessed June 17, 2025, https://www.irjet.net/archives/V6/i6/IRJET-V6I6700.pdf
72. A wavelet-based image fusion tutorial - Docta Complutense, accessed June 17, 2025, https://docta.ucm.es/rest/api/core/bitstreams/771b1ec0-a2c8-4aac-a680-a4a99ed382cd/content
73. Research on Image Denoising in Edge Detection Based on Wavelet ..., accessed June 17, 2025, https://www.mdpi.com/2076-3417/13/3/1837
74. (PDF) Visual Diagnostics Based on Image Wavelet Transform - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/228945342_Visual_Diagnostics_Based_on_Image_Wavelet_Transform
75. Discrete wavelet transform for image texture analysis - Stack Overflow, accessed June 17, 2025, https://stackoverflow.com/questions/54611018/discrete-wavelet-transform-for-image-texture-analysis
76. Singular value decomposition - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Singular_value_decomposition
77. Singular Value Decomposition (SVD) - GeeksforGeeks, accessed June 17, 2025, https://www.geeksforgeeks.org/singular-value-decomposition-svd/
78. Singular Value Decomposition: Applications to Image Processing, accessed June 17, 2025, https://www.lagrange.edu/academics/undergraduate/undergraduate-research/_images/18-Citations2020.Compton.pdf
79. Comparative Analysis of Singular Value Decomposition (SVD) and Wavelet Difference Reduction (WDR) based Image Compression - Research India

Publications, accessed June 17, 2025, https://www.ripublication.com/irph/ijert_spl17/ijertv10n1spl_94.pdf
80. SVD Image Compression, Explained - Dennis Miczek, accessed June 17, 2025, https://dmicz.github.io/machine-learning/svd-image-compression/
81. An Information-Theoretic Framework for Image Complexity, accessed June 17, 2025, https://imae.udg.edu/~rigau/Publications/Rigau05A.pdf
82. www.mdpi.com, accessed June 17, 2025, https://www.mdpi.com/1099-4300/27/2/166#:~:text=The%20Normalized%20Shannon%20Entropy%20definition%20of%20visual%20complexity%20begins%20with,distribution%20of%20colors%20%5B15%5D.
83. Measuring complexity of images using Multiscale Entropy | JOV - Journal of Vision, accessed June 17, 2025, https://jov.arvojournals.org/article.aspx?articleid=2750283
84. Perceptual Complexity as Normalized Shannon Entropy - MDPI, accessed June 17, 2025, https://www.mdpi.com/1099-4300/27/2/166
85. (PDF) Perceptual Complexity as Normalized Shannon Entropy - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/388727833_Perceptual_Complexity_as_Normalized_Shannon_Entropy
86. Information Entropy Measure for Evaluation of Image Quality - PMC, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC3043833/
87. Kullback–Leibler divergence - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence
88. Metrics Used to Compare Histograms - Krystian Safjan's Blog, accessed June 17, 2025, https://safjan.com/metrics-to-compare-histograms/
89. Kullback-Leibler Divergence | GeeksforGeeks, accessed June 17, 2025, https://www.geeksforgeeks.org/kullback-leibler-divergence/
90. Jensen-Shannon Divergence — JSD • philentropy, accessed June 17, 2025, https://drostlab.github.io/philentropy/reference/JSD.html
91. Jensen–Shannon divergence - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Jensen%E2%80%93Shannon_divergence
92. Image Threshold Segmentation with Jensen-Shannon Divergence and Its Application - IAENG, accessed June 17, 2025, https://www.iaeng.org/IJCS/issues_v49/issue_1/IJCS_49_1_21.pdf
93. Jensen Shannon Divergence: Intuition and Practical Application ..., accessed June 17, 2025, https://arize.com/blog-course/jensen-shannon-divergence/
94. Automatic Detection of Texture-defects using Texture-periodicity and Jensen-Shannon Divergence - Korea Science, accessed June 17, 2025, https://koreascience.kr/article/JAKO201222340312054.page?vJhi6srpW=F2sf2S&lang=ko
95. Automatic Detection of Texture-defects using Texture-periodicity and Jensen-Shannon Divergence - Journal of Information Processing Systems (JIPS), accessed June 17, 2025, https://jips-k.org/digital-library/2012/8/2/359
96. Automatic Detection of Texture-defects using Texture-periodicity and Jensen-Shannon Divergence - Journal of Information Processing Systems (JIPS),

accessed June 17, 2025, https://jips-k.org/digital-library/manuscript/file/22541/JIPS-2012-8-2-359.pdf
97. Automatic Detection of Texture-defects using Texture-periodicity and Jensen-Shannon Divergence | Request PDF - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/264108183_Automatic_Detection_of_Texture-defects_using_Texture-periodicity_and_Jensen-Shannon_Divergence
98. Chi-Square Distance in Python - Tutorialspoint, accessed June 17, 2025, https://www.tutorialspoint.com/chi-square-distance-in-python
99. Chi-square distance in Python - GeeksforGeeks, accessed June 17, 2025, https://www.geeksforgeeks.org/chi-square-distance-in-python/
100. Comparing two histograms using Chi-Square distance - Cross Validated - Stack Exchange, accessed June 17, 2025, https://stats.stackexchange.com/questions/184101/comparing-two-histograms-using-chi-square-distance
101. The Earth Mover's Distance, accessed June 17, 2025, https://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/RUBNER/emd.htm
102. Wasserstein metric - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Wasserstein_metric
103. What is Earth Mover's Distance - Activeloop, accessed June 17, 2025, https://www.activeloop.ai/resources/glossary/earth-movers-distance/
104. (PDF) The Earth Mover's Distance as a Metric for Image Retrieval, accessed June 17, 2025, https://www.researchgate.net/publication/220659330_The_Earth_Mover's_Distance_as_a_Metric_for_Image_Retrieval
105. Wasserstein Distance and Textual Similarity - neptune.ai, accessed June 17, 2025, https://neptune.ai/blog/wasserstein-distance-and-textual-similarity
106. Incorporating Wasserstein metric in Autoencoder Training for Effective Image Compression - University of Waterloo, accessed June 17, 2025, https://uwaterloo.ca/computational-mathematics/sites/default/files/uploads/documents/research_paper_antonina_rudakova.pdf
107. [1612.00181] Monge's Optimal Transport Distance for Image Classification - arXiv, accessed June 17, 2025, https://arxiv.org/abs/1612.00181
108. What is Sliding Window in Object Detection: Complete Overview of Methods & Tools, accessed June 17, 2025, https://supervisely.com/blog/how-sliding-window-improves-neural-network-models/
109. Sliding Windows for Object Detection with Python and OpenCV ..., accessed June 17, 2025, https://pyimagesearch.com/2015/03/23/sliding-windows-for-object-detection-with-python-and-opencv/
110. Sliding Window Technique - GeeksforGeeks, accessed June 17, 2025, https://www.geeksforgeeks.org/window-sliding-technique/
111. Advanced Algorithms: Median in Sliding Window - HackerNoon, accessed June 17, 2025,

https://hackernoon.com/coding-interview-prep-mastering-median-statistics-in-sliding-windows
112. SWSSL: Sliding window-based self-supervised learning for anomaly detection in high-resolution images, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC10766076/
113. Anomaly Detection Using a Sliding Window Technique and Data Imputation with Machine Learning for Hydrological Time Series - MDPI, accessed June 17, 2025, https://www.mdpi.com/2073-4441/13/13/1862
114. Dive into Time-Series Anomaly Detection: A Decade Review - arXiv, accessed June 17, 2025, https://arxiv.org/html/2412.20512v1
115. Anomaly Detection Using a Sliding Window Technique and Data Imputation with Machine Learning for Hydrological Time Series - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/353001246_Anomaly_Detection_Using_a_Sliding_Window_Technique_and_Data_Imputation_with_Machine_Learning_for_Hydrological_Time_Series
116. How to build robust anomaly detectors with machine learning - Ericsson, accessed June 17, 2025, https://www.ericsson.com/en/blog/2020/4/anomaly-detection-with-machine-learning
117. Anomaly Detection in Time Series - Neptune.ai, accessed June 17, 2025, https://neptune.ai/blog/anomaly-detection-in-time-series
118. Non-local means - Wikipedia, accessed June 17, 2025, https://en.wikipedia.org/wiki/Non-local_means
119. An adaptive non-local means filter for denoising live-cell images and improving particle detection - PMC - PubMed Central, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC3087503/
120. pmc.ncbi.nlm.nih.gov, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC10529561/#:~:text=The%20nonlocal%20means%20reconstructs%20an,around%20the%20locations%20in%20consideration.
121. Image Anomalies: a Review and Synthesis of Detection Methods - IPOL Development Server, accessed June 17, 2025, http://dev.ipol.im/~adavy/Image_anomalies__a_review_and_synthesis_of_detection_methods.pdf
122. Persistent Homology: Theory and Practice, accessed June 17, 2025, https://pub.ista.ac.at/~edels/Papers/2012-11-PHTheoryPractice.pdf
123. TDA in Image Analysis: A Persistent Homology Primer, accessed June 17, 2025, https://www.numberanalytics.com/blog/tda-image-analysis-persistent-homology-primer
124. Unlocking Persistent Homology - Number Analytics, accessed June 17, 2025, https://www.numberanalytics.com/blog/ultimate-guide-interleaving-distance-persistent-homology
125. Persistent homology: a step-by-step introduction for newcomers, accessed June 17, 2025, https://www.math.uri.edu/~thoma/comp_top__2018/stag2016.pdf

126. Mastering Persistent Homology - Number Analytics, accessed June 17, 2025, https://www.numberanalytics.com/blog/mastering-persistent-homology-transform
127. Persistent Homology Based Graph Convolution Network for Fine-Grained 3D Shape Segmentation - CVF Open Access, accessed June 17, 2025, https://openaccess.thecvf.com/content/ICCV2021/papers/Wong_Persistent_Homology_Based_Graph_Convolution_Network_for_Fine-Grained_3D_Shape_ICCV_2021_paper.pdf
128. Persistent Homology for Defect Detection in Non-Destructive Evaluation of Materials, accessed June 17, 2025, https://www.math.uni-hamburg.de/home/iske/papers/ndt_persistent.pdf
129. (PDF) Persistent homology for defect detection in non-destructive evaluation of materials, accessed June 17, 2025, https://www.researchgate.net/publication/305209461_Persistent_homology_for_defect_detection_in_non-destructive_evaluation_of_materials
130. Leveraging Persistent Homology Features for Accurate Defect Formation Energy Predictions via Graph Neural Networks | Chemistry of Materials, accessed June 17, 2025, https://pubs.acs.org/doi/10.1021/acs.chemmater.4c03028
131. Persistent Homology in Medical Image Processing: A Literature Review - medRxiv, accessed June 17, 2025, https://www.medrxiv.org/content/10.1101/2025.02.21.25322669v1
132. Persistent Homology in Medical Image Processing: A Literature Review - ResearchGate, accessed June 17, 2025, https://www.researchgate.net/publication/389331382_Persistent_Homology_in_Medical_Image_Processing_A_Literature_Review
133. Persistent Homology in Medical Image Processing: A Literature Review | medRxiv, accessed June 17, 2025, https://www.medrxiv.org/content/10.1101/2025.02.21.25322669v1.full-text
134. Persistent Homology Tools for Image Analysis - the Buckingham E-Archive of Research (BEAR), accessed June 17, 2025, http://bear.buckingham.ac.uk/467/1/DPhil%20_Final_ForPrint_22_2_2020.pdf
135. Leveraging Persistent Homology Features for Accurate Defect Formation Energy Predictions via Graph Neural Networks - PubMed Central, accessed June 17, 2025, https://pmc.ncbi.nlm.nih.gov/articles/PMC11866744/
136. Unlocking Material Secrets with Persistent Homology - Number Analytics, accessed June 17, 2025, https://www.numberanalytics.com/blog/material-science-persistent-homology-guide
137. Advancing Materials Research with Persistent Homology - Number Analytics, accessed June 17, 2025, https://www.numberanalytics.com/blog/advancing-materials-research-persistent-homology
138. Introduction to Persistent Homology - YouTube, accessed June 17, 2025, https://www.youtube.com/watch?v=2PSqWBIrn90
139. Persistent homology features and multiple topologies for image analysis - SPIE

Digital Library, accessed June 17, 2025, https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11399/113990A/Persistent-homology-features-and-multiple-topologies-for-image-analysis/10.1117/12.2567052.full
140. Leveraging Persistent Homology Features for Accurate Defect Formation Energy Predictions via Graph Neural Networks - arXiv, accessed June 17, 2025, https://arxiv.org/html/2407.05204v1